Towards an algebra for real-time programs

Brijesh Dongol[1] Ian J. Hayes[2] Larissa Meinicke[3] Kim Solin[4]

[1,2,3,4]School of Information Technology and Electrical Engineering,
The University of Queensland

[1]Department of Computer Science,
The University of Sheffield

[4]Department of Software Engineering,
Gotland University

September 22, 2012

Background

- Working on interval-based models for reasoning about real-time systems
- Have hybrid properties, i.e., mixture of continuous and discrete properties
- Aiming for realistic assumptions to ensure implementability
- Trying not to assume too much is "instantaneous"
- Weakening assumptions leads to increase in complexity

Goals

- Main question: What algebra does our model give rise to?
  - Begin with interval predicates (this paper)
  - Moving towards programming frameworks (e.g., real-time action systems)
- Secondary questions: Can we use an algebra to simplify proofs in the model, improve insights, etc.?
- There are related algebraic approaches to reasoning about hybrid systems — in particular we build on work by Peter Höfner and Bernhard Möller

A model for real-time programs

- Several authors have proposed the use of intervals as a way to reason about real-time/hybrid systems
- Brief overview of our model

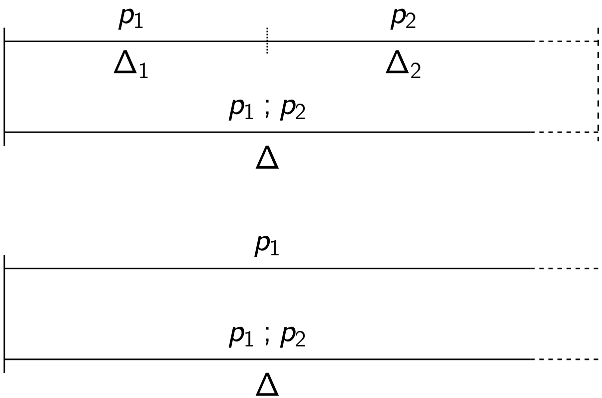
$$
\begin{array}{rcl}
\mathit{Time} & \mathrel{\widehat{=}} & \mathbb{R} \\
\mathit{State} & \mathrel{\widehat{=}} & \mathit{Var} \to \mathit{Val} \\
\mathit{Stream} & \mathrel{\widehat{=}} & \mathit{Time} \to \mathit{State} \\
\mathit{Interval} & \mathrel{\widehat{=}} & \left\{ \Delta \subseteq \mathit{Time} \;\middle|\; \begin{array}{r} \forall t_1, t_2 \in \Delta, t \in \mathit{Time} \bullet \\ t_1 \leq t \leq t_2 \Rightarrow t \in \Delta \end{array} \right\} \\
\mathit{StatePred} & \mathrel{\widehat{=}} & \mathit{State} \to \mathbb{B} \\
\mathit{IntvPred} & \mathrel{\widehat{=}} & \mathit{Interval} \to \mathit{Stream} \to \mathbb{B}
\end{array}
$$

Chop operator

- For interval-based logics the chop operator (denoted ';') is useful
- We use '.' for function application

- For interval predicates p_1 and p_2, interval Δ and stream s, we say $(p_1 ; p_2).\Delta.s$ holds iff either
  - Δ can be split into adjoining intervals Δ_1 and Δ_2 such that both $p_1.\Delta_1.s$ and $p_2.\Delta_2.s$ hold, or
  - the least upper bound of Δ is ∞ and $p_1.\Delta.s$ holds

Chop operator



Chop operator

- Chop allows one to model sequential composition and iteration
- However, reasoning across the boundary of two adjoining intervals can be problematic, e.g., if we want to specify $\boxdot c \; ; \boxdot \neg c$

Always definition

Definition

For state predicate c, time t and stream s, define

$$(c@t).s \ \ \widehat{=} \ \ c.(s.t)$$

Definition

For state predicate c and interval Δ, define

$$(\boxdot c).\Delta \ \ \widehat{=} \ \ \forall t : \Delta \bullet c@t$$

Always definition

Definition

For state predicate c, time t and stream s, define

$$(c@t).s \quad \widehat{=} \quad c.(s.t)$$

Definition

For state predicate c and interval Δ, define

$$(\boxdot c).\Delta \quad \widehat{=} \quad \forall t : \Delta \bullet c@t$$

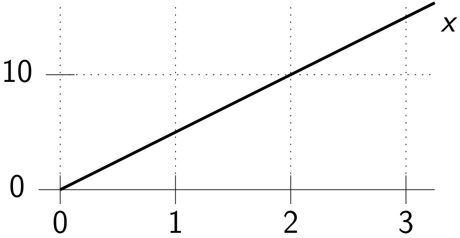
Definition

For variable x, time t and stream s, define

$$(x@t).s \quad \widehat{=} \quad (s.t).x$$

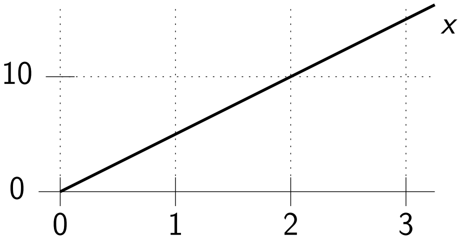
Example 1

- Consider continuous variable x where $x@0 = 0$ and $\boxdot(\mathring{x} = 5).[0, 3]$



Example 1

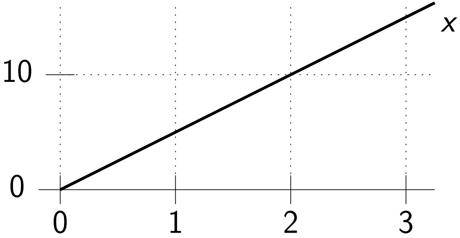
- Consider continuous variable x where $x@0 = 0$ and $\boxdot(\mathring{x} = 5).[0, 3]$



- We have $x@1 = 5$

Example 1

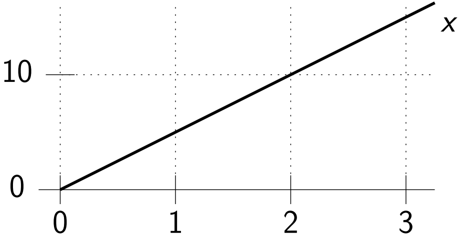
- Consider continuous variable x where $x@0 = 0$ and $\boxdot(\mathring{x} = 5).[0,3]$



- We have $x@1 = 5$
- Hence, $(\boxdot(x < 5) \,;\, \boxdot(x \geq 5)).[0,2]$ should hold

Example 1

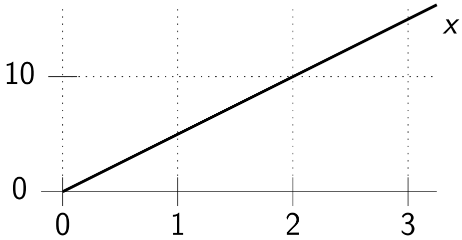
- Consider continuous variable x where $x@0 \ = \ 0$ and $\boxdot(\mathring{x} = 5).[0, 3]$



- We have $x@1 \ = \ 5$
- Hence, $(\boxdot(x < 5) \, ; \boxdot(x \geq 5)).[0, 2]$ should hold because
  - $\boxdot(x < 5).[0, 1)$ and
  - $\boxdot(x \geq 5).[1, 2]$

Example 1

- Consider continuous variable x where $x@0 \;=\; 0$ and $\boxdot(\mathring{x} = 5).[0, 3]$



- We have $x@1 \;=\; 5$
- Hence, $(\boxdot(x < 5) \,;\, \boxdot(x \geq 5)).[0, 2]$ should hold because
  - $\boxdot(x < 5).[0, 1)$ and
  - $\boxdot(x \geq 5).[1, 2]$
- However, $\boxdot(x < 5).[0, 1]$ does not hold

Lesson learnt

- Can be difficult to formalise ';' if we restrict ourselves to closed intervals only
- Allow intervals to be open/closed at either end

Consequences of ';' with closed intervals

- Duration calculus:
  - All finite length intervals are closed
  - $\boxdot c$ weakened to $\mathit{AlmostAlways}(c)$
  - $\mathit{AlmostAlways}(c)$ holds in Δ iff the times in Δ for which c is *false* form a set of measure 0

Consequences of ';' with closed intervals

- Duration calculus:
  - All finite length intervals are closed
  - $\boxdot c$ weakened to $\mathit{AlmostAlways}(c)$
  - $\mathit{AlmostAlways}(c)$ holds in Δ iff the times in Δ for which c is *false* form a set of measure 0
- Höfner and Möller's hybrid algebra:
  - All finite length intervals are closed
  - A new (relaxed) compatibility relation defined at point of composition between two adjoining intervals

Example 2

Can we deduce $\boxdot(x \geq 5).[0, 3]$ using

- $\boxdot(x \geq 5).[0, 2)$ and
- $\boxdot(x \geq 5).(2, 3]$?

Example 2

Can we deduce $\boxdot(x \geq 5).[0,3]$ using

- $\boxdot(x \geq 5).[0,2)$ and
- $\boxdot(x \geq 5).(2,3]$?

No! May have $x@2 < 5$.

Lesson learnt

Adjoining intervals should be contiguous across their boundary

Formalising adjoins and chop

Δ_1 *Adjoins* Δ_2 iff

- $\Delta_1 = \{\}$, or
- $\Delta_2 = \{\}$, or
- $\Delta_1 \cap \Delta_2 = \{\}$ and $\Delta_1 \cup \Delta_2 \in \mathit{Interval}$ and $\mathit{lub}.\Delta_1 = \mathit{glb}.\Delta_2$

Formalising adjoins and chop

Δ_1 *Adjoins* Δ_2 iff

- $\Delta_1 = \{\}$, or
- $\Delta_2 = \{\}$, or
- $\Delta_1 \cap \Delta_2 = \{\}$ and $\Delta_1 \cup \Delta_2 \in \mathit{Interval}$ and $\mathit{lub}.\Delta_1 = \mathit{glb}.\Delta_2$

$$(p_1 \;;\; p_2).\Delta.s \quad \widehat{=} \quad \begin{array}{l} \left(\begin{array}{l} \exists \Delta_1, \Delta_2 \bullet (\Delta_1 \ \mathit{Adjoins}\ \Delta_2) \wedge \\ \qquad\qquad\quad (\Delta_1 \cup \Delta_2 = \Delta) \wedge \\ \qquad\qquad\quad p_1.\Delta_1.s \wedge p_2.\Delta_1.s \end{array}\right) \\ \vee \\ (\mathit{lub}.\Delta = \infty \wedge p_1.\Delta.s) \end{array}$$

The algebra of interval predicates

Proposition

$(\mathit{IntvPred}, \vee, ;, \mathsf{False}, \mathsf{Empty})$ *forms a Boolean weak quantale*

The algebra of interval predicates

Proposition

$(\mathit{IntvPred}, \vee, ;, \mathsf{False}, \mathsf{Empty})$ *forms a Boolean weak quantale*

where

- '$\vee$' is lifted disjunction, i.e., $(p_1 \vee p_2).\Delta.s = p_1.\Delta.s \vee p_2.\Delta.s$
- ';' is the chop operator
- $\mathsf{False}.\Delta.s \ \widehat{=}\ \mathit{false}$
- $\mathsf{Empty}.\Delta.s \ \widehat{=}\ (\Delta = \{\})$
- Ordering '$\leq$' is universal implication '$\Rrightarrow$', where

$$p_1 \Rrightarrow p_2 \quad \widehat{=} \quad \forall \Delta, s \bullet p_1.\Delta.s \Rightarrow p_2.\Delta.s$$

- Note that $(p \,;\, \mathsf{False}) \not\equiv \mathsf{False}$

Tests

- Allowing open intervals affects test elements, i.e., a such that $a \leq 1$

Tests

- Allowing open intervals affects test elements, i.e., a such that $a \leq 1$
- If all intervals are closed, test elements correspond to point intervals (Höfner and Möller)

Tests

- Allowing open intervals affects test elements, i.e., a such that $a \leq 1$
- If all intervals are closed, test elements correspond to point intervals (Höfner and Möller)

- In our model:
  - 1 corresponds to Empty
  - The only elements corresponding to tests are False and Empty
  - This is not problematic — we assume guard evaluation takes time
  - $beh.(\textbf{if } b \textbf{ then } S_1 \textbf{ else } S_2 \textbf{ fi}) \ \widehat{=}\ (\circledast b\,;beh.S_1) \vee (\circledast \neg b\,;beh.S_2)$

Iteration: basic properties

- For a Boolean weak quantale $(A, +, \cdot, 0, 1)$, one can define

$$\begin{array}{rcl} a^* & \widehat{=} & (\mu z \bullet az + 1) \\ a^\omega & \widehat{=} & (\nu z \bullet az + 1) \\ a^\infty & \widehat{=} & (\nu z \bullet az) \end{array}$$

  - * is a finite iteration
  - $^\omega$ is an iteration that is either finite or infinite
  - $^\infty$ is an infinite iteration

- Unfolding rules:

$$a^* = aa^* + 1 \qquad a^\omega = aa^\omega + 1 \qquad a^\infty = aa^\infty$$

- Induction rules:

$$\begin{array}{rcl} az + 1 \leq z & \Rightarrow & a^* \leq z \\ z \leq az + 1 & \Rightarrow & z \leq a^\omega \\ z \leq az & \Rightarrow & z \leq a^\infty \end{array}$$

Iteration: some derived properties

Yes:

- $b + ac \leq c \quad \Rightarrow \quad a^* b \leq c$
- $c \leq ac + b \quad \Rightarrow \quad c \leq a^\infty + a^* b$

No:

- $c \leq ac + b \;\Rightarrow\; c \leq a^\omega b$

Iteration: some derived properties

Yes:

- $b + ac \leq c \quad \Rightarrow \quad a^* b \leq c$
- $c \leq ac + b \quad \Rightarrow \quad c \leq a^\infty + a^* b$

No:

- $c \leq ac + b \;\Rightarrow\; c \leq a^\omega b$

Counter-example: Taking $a = 1$ and $b = 0$, equation reduces to $c \leq \top 0$.

Iteration: some derived properties

Yes:

- $b + ac \leq c \quad \Rightarrow \quad a^* b \leq c$
- $c \leq ac + b \quad \Rightarrow \quad c \leq a^\infty + a^* b$

No:

- $c \leq ac + b \;\Rightarrow\; c \leq a^\omega b$

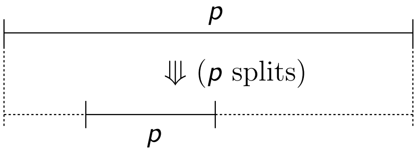
Counter-example: Taking $a = 1$ and $b = 0$, equation reduces to $c \leq \top 0$.

Define positive iteration $a^+ \mathrel{\widehat{=}} aa^*$. Then induction and unfolding rules are:

- $az + a \leq z \quad \Rightarrow \quad a^+ \leq z$
- $a^\infty = a^+ a^\infty$

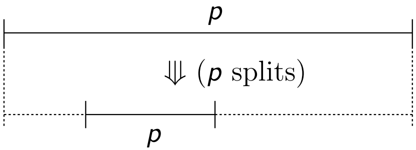
Compositional reasoning

- An interval predicate p splits iff given that p holds over an interval Δ, p holds over all subintervals of Δ

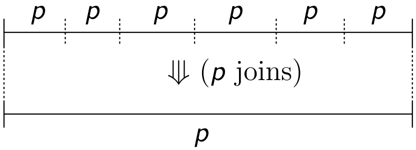


Compositional reasoning

- An interval predicate p splits iff given that p holds over an interval Δ, p holds over all subintervals of Δ



- An interval predicate p joins iff p holds in an interval Δ whenever p^+ holds in Δ



Compositional reasoning

Definition

Suppose $(A, +, \cdot, 0, 1)$ is a Boolean weak quantale and $a \in A$.

- a splits iff $\forall b, c : A \bullet a \curlywedge bc \leq (a \curlywedge b)(a \curlywedge c)$
- a joins iff $\forall b, c : A \bullet (a \curlywedge b)(a \curlywedge c) \leq a \curlywedge bc$

Compositional reasoning

Definition

Suppose $(A, +, \cdot, 0, 1)$ is a Boolean weak quantale and $a \in A$.

- a splits iff $\forall b, c : A \bullet a \curlywedge bc \leq (a \curlywedge b)(a \curlywedge c)$
- a joins iff $\forall b, c : A \bullet (a \curlywedge b)(a \curlywedge c) \leq a \curlywedge bc$

Höfner and Möller define "submodular" to mean splits and "modular" to mean both splits and joins

Compositional reasoning

Lemma

Suppose $a \in A$ where $(A, +, \cdot, 0, 1)$ is a Boolean weak quantale.

(1) *If a splits, then for any $b \in A$, $a \curlywedge b^* \leq (a \curlywedge b)^*$ holds.*

(2) *If a splits, then for any $b \in A$, $a \curlywedge b^\omega \leq (a \curlywedge b)^\omega$ holds.*

(3) *If a joins, then for any $b \in A$, $(a \curlywedge b)^+ \leq a \curlywedge b^+$ holds.*

Compositional reasoning

Lemma

Suppose $a \in A$ where $(A, +, \cdot, 0, 1)$ is a Boolean weak quantale.

(1) *If a splits, then for any $b \in A$, $a \curlywedge b^* \leq (a \curlywedge b)^*$ holds.*

(2) *If a splits, then for any $b \in A$, $a \curlywedge b^\omega \leq (a \curlywedge b)^\omega$ holds.*

(3) *If a joins, then for any $b \in A$, $(a \curlywedge b)^+ \leq a \curlywedge b^+$ holds.*

Note

- If a joins it is not necessarily true that
  - for any $b \in A$, $(a \curlywedge b)^* \leq a \curlywedge b^*$ holds
  - for any $b \in A$, $(a \curlywedge b)^\omega \leq a \curlywedge b^\omega$ holds

Compositional reasoning

Lemma

Suppose $a \in A$ where $(A, +, \cdot, 0, 1)$ is a Boolean weak quantale.

(1) *If a splits, then for any $b \in A$, $a \curlywedge b^* \leq (a \curlywedge b)^*$ holds.*

(2) *If a splits, then for any $b \in A$, $a \curlywedge b^\omega \leq (a \curlywedge b)^\omega$ holds.*

(3) *If a joins, then for any $b \in A$, $(a \curlywedge b)^+ \leq a \curlywedge b^+$ holds.*

Note

- If a joins it is not necessarily true that
  - for any $b \in A$, $(a \curlywedge b)^* \leq a \curlywedge b^*$ holds
  - for any $b \in A$, $(a \curlywedge b)^\omega \leq a \curlywedge b^\omega$ holds
- Left hand side may iterate zero times and get 1, but on right hand side we already have a

Finite and infinite elements

For a Boolean weak quantale $(A, +, \cdot, 0, 1)$ and $a \in A$ following Höfner and Möller, we have:

- a is purely infinite iff $a0 = a$
- a is purely finite iff $a0 = 0$.
- the largest purely infinite element INF:
  $a \leq \mathsf{INF} \iff a0 = a$
- the largest purely finite element FIN:
  $a \leq \mathsf{FIN} \iff a0 = 0$

INF and FIN in the model

- INF corresponds to interval predicate

$$\lambda \Delta : Interval, s : Stream \bullet lub.\Delta = \infty$$

- FIN corresponds to interval predicate

$$\lambda \Delta : Interval, s : Stream \bullet lub.\Delta \neq \infty$$

Different forms of iteration

Can distinguish between terminating, divergent, Zeno-like and non-terminating elements.

$$\begin{aligned} \mathsf{Term}\, a &\mathrel{\widehat{=}} \mathsf{FIN} \curlywedge a^{*} & \qquad \mathsf{Zeno}\, a &\mathrel{\widehat{=}} \mathsf{FIN} \curlywedge a^{\infty} \\ \mathsf{Diverge}\, a &\mathrel{\widehat{=}} \mathsf{INF} \curlywedge a^{+} & \qquad \mathsf{NonTerm}\, a &\mathrel{\widehat{=}} \mathsf{INF} \curlywedge a^{\infty} \end{aligned}$$

Different forms of iteration

Can distinguish between terminating, divergent, Zeno-like and non-terminating elements.

$$\begin{aligned} \mathsf{Term}\, a &\mathrel{\widehat{=}} \mathsf{FIN} \curlywedge a^* & \qquad \mathsf{Zeno}\, a &\mathrel{\widehat{=}} \mathsf{FIN} \curlywedge a^\infty \\ \mathsf{Diverge}\, a &\mathrel{\widehat{=}} \mathsf{INF} \curlywedge a^+ & \qquad \mathsf{NonTerm}\, a &\mathrel{\widehat{=}} \mathsf{INF} \curlywedge a^\infty \end{aligned}$$

Lemma

Suppose $(A, +, \cdot, 0, 1)$ is a Boolean weak quantale and $a \in A$. Then each of the following holds.

$$\mathsf{Term}\, a = (\mathsf{FIN} \curlywedge a)^* \tag{1}$$

$$\mathsf{Zeno}\, a = \mathsf{FIN} \curlywedge (\mathsf{FIN} \curlywedge a)^\infty \tag{2}$$

$$\mathsf{Diverge}\, a \leq \mathsf{NonTerm}\, a \tag{3}$$

Properties in the model: Next

- Useful to be able to reason about properties like $next.p$
- $(next.p).\Delta$ holds iff p holds in some interval that immediately follows Δ
- Formally,

$$(next.p).\Delta.s \quad \widehat{=} \quad \exists \Delta' \bullet (\Delta \ Adjoins \ \Delta') \wedge p.\Delta'.s$$

Properties in the model: Next

- Useful to be able to reason about properties like $next.p$
- $(next.p).\Delta$ holds iff p holds in some interval that immediately follows Δ
- Formally,

$$(next.p).\Delta.s \quad \widehat{=} \quad \exists \Delta' \bullet (\Delta \ Adjoins \ \Delta') \wedge p.\Delta'.s$$

- Höfner and Möller use domain and co-domain elements to get algebraic characterisation of *next*

Properties in the model: Next

- Useful to be able to reason about properties like $next.p$
- $(next.p).\Delta$ holds iff p holds in some interval that immediately follows Δ
- Formally,

$$(next.p).\Delta.s \quad \widehat{=} \quad \exists \Delta' \bullet (\Delta \ Adjoins \ \Delta') \wedge p.\Delta'.s$$

- Höfner and Möller use domain and co-domain elements to get algebraic characterisation of *next*
- This is not possible for us — intervals may be open

Properties in the model: Next

- Useful to be able to reason about properties like $next.p$
- $(next.p).\Delta$ holds iff p holds in some interval that immediately follows Δ
- Formally,

$$(next.p).\Delta.s \quad \widehat{=} \quad \exists \Delta' \bullet (\Delta \ Adjoins \ \Delta') \wedge p.\Delta'.s$$

- Höfner and Möller use domain and co-domain elements to get algebraic characterisation of *next*
- This is not possible for us — intervals may be open
- But one can derive properties in the model with the help of algebra

Properties in the model: Previous and Next

Lemma

For any interval predicate p, both of the following hold.

1. *If p splits then $(p \Rightarrow next.p)^+ \wedge \mathrm{Fin} \Rrightarrow (p \Rightarrow next.p)$.*
2. *If p joins then $(p \wedge next.p)^+ \wedge \mathrm{Fin} \Rrightarrow (p \wedge next.p)$.*

Properties in the model: Previous and Next

Lemma

For any interval predicate p, both of the following hold.

1. *If p splits then* $(p \Rightarrow next.p)^+ \wedge \mathrm{Fin} \Rrightarrow (p \Rightarrow next.p)$.
2. *If p joins then* $(p \wedge next.p)^+ \wedge \mathrm{Fin} \Rrightarrow (p \wedge next.p)$.

Note the similarity with unfolding rule when proving loop invariants.

Conclusions

- Properties at and across the boundary between adjoining intervals can be subtle
- The algebraic approach makes reasoning elegant and perspicuous
- Höfner and Möller lay some groundwork (for a closed interval model) that we are (luckily) able to re-use

Future work

- Use these results to prove properties of real-time action systems
- Mechanisation in Isabelle/HOL
  - With Alasdair Armstrong — have encoded a discrete (integer) interval theory into Isabelle/HOL and shown that discrete intervals form a Boolean weak quantale
  - Have Lattice.thy $\rightarrow$ Quantale.thy $\rightarrow$ DiscreteIntvPred.thy
  - Aiming for DiscreteIntvPred.thy $\rightarrow$ Commands.thy $\rightarrow$ Rely-Guarantee.thy

Questions?