A Unary Semigroup Trace Algebra

Pedro Ribeiro

UNIVERSITY of York

Department of Computer Science

www.cs.york.ac.uk/robostar/

RAMiCS, 26th October 2020

Outline

Motivation

Traces

Typically finite sequences. Concrete examples include:

- Events: $\langle a, b, \ldots, c \rangle$.
- Events and refusals: $\langle a, \{e\}, \ldots, c \rangle$.
- Events, refusals and time: $\langle (\langle a \rangle, \{e\}), (\langle\rangle, \{a, e\}), \ldots \rangle$.

Besides events, may record events refused, and across time.

Motivation

Prefix relation

Sequentiality may be different, for example:

- Events: $s \leq t \mathrel{\widehat{=}} \exists z \bullet s \frown z = t$.
- Events and refusals: $\langle a, \{e\} \rangle \leq \langle a, \{e\} \rangle$
- Events, refusals and time: $\langle (\langle a \rangle, \{e\}) \rangle \leq \langle (\langle a \rangle, \{f\}) \rangle$

In common: reflexive, transitive, and optionally anti-symmetric.

Monoid Trace Algebra

Algebra $(\mathcal{T}, \widehat{}, \varepsilon)$ [1]
Useful for *unification* of semantics in Isabelle/UTP.

- **TA1. Associativity** (semigroup): $(a \widehat{} b) \widehat{} c = a \widehat{} (b \widehat{} c)$
- **TA2. Monoid** (zero laws): $\varepsilon \widehat{} a = a \widehat{} \varepsilon = a$
- **TA3. Zero sum** (left): $(a \widehat{} b = \varepsilon) \Rightarrow a = \varepsilon$
- **TA4. Left-cancellative**: $(a \widehat{} b = a \widehat{} c) \Rightarrow b = c$

Prefix relation
An order provided concatenation satisfies the above axioms.

$$a \leq b \mathrel{\widehat{=}} \exists\, c \bullet a \widehat{} c = b$$

[1] Simon Foster et al. "Unifying theories of time with generalised reactive processes". In: *Information Processing Letters* 135 (2018), pp. 47 –52.

Unifying Theories of Programming

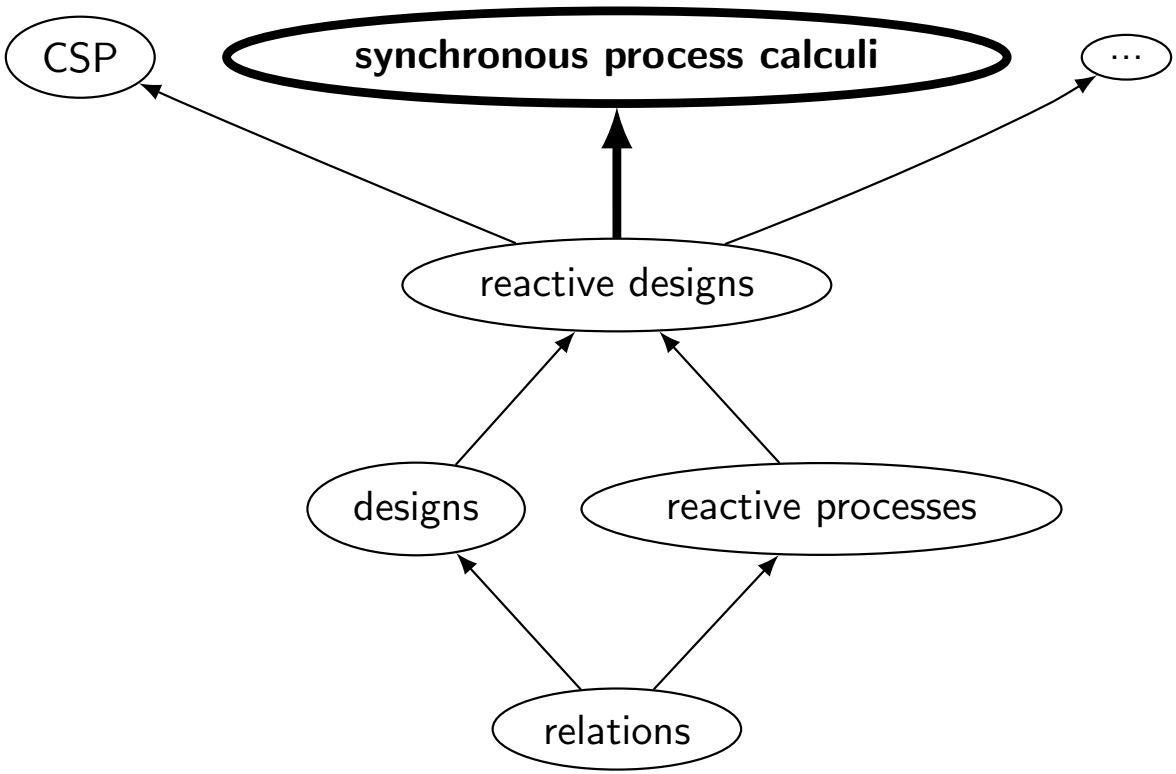

Figure: Simplified view of UTP theory hierarchy.

Unifying Theories of Programming

Alphabet

Alphabetised relations for recording observations.

$tr, tr' : \mathcal{T}$, with tr for before, and tr' for after observations.

Healthiness conditions

Define the valid predicates via fixed points.

$$\mathbf{R1}(P) \mathrel{\widehat{=}} P \wedge tr \leq tr' \qquad \mathbf{R3}(P) \mathrel{\widehat{=}} \mathit{II} \lhd \mathit{wait} \rhd P$$

$$\mathbf{R2}(P) \mathrel{\widehat{=}} P[\varepsilon, tr' - tr / tr, tr'] \qquad \mathbf{R}(P) \mathrel{\widehat{=}} \mathbf{R1} \circ \mathbf{R2} \circ \mathbf{R3}(P)$$

Operators

Define the language for specification of behaviours.

$$\mathit{Skip} \mathrel{\widehat{=}} \mathbf{R}(\mathit{true} \vdash tr' = tr \wedge \neg\, \mathit{wait}')$$

UTP for synchronous process calculi?

Goal: Widen the reactive theory foundation of UTP to accommodate synchronous process calculi.

Challenges:

- **Prefix relation**
  Refusals may change during current time slot.

  Events, refusals and time: $\langle(\langle a\rangle, \{e\})\rangle \leq \langle(\langle a\rangle, \{f\})\rangle$

- **Healthiness conditions**
  Fixed ε not suitable.

  $\mathbf{R2}(P) = P[\langle(\langle\rangle, \mathit{snd} \circ \mathit{last}(tr))\rangle, tr' - tr/tr, tr']$

UTP for synchronous process calculi?

Concatenation and subtraction of slots lifted to slot sequences.

- $(h_1, \_\_) \mathbin{\widehat{}} (h_2, r_2) = (h_1 \mathbin{\widehat{}} h_2, r_2)$
- $(h_1, r_1) - (h_2, \_\_) = (h_1 - h_2, r_1)$

Challenges

- Concatenation is non-injective $\Rightarrow$ pre-order, as $\leq$ defined from $\widehat{}$.
- $\mathbf{R2}(P) = P[\varepsilon, tr' - tr / tr, tr']$ not adequate.
- Slot sequences are non-empty.

Foundations

Solution $(\mathcal{T}, \widehat{}, \Phi)$

- Change **R2**

$$\mathbf{R2}(P) = P[\Phi(tr), tr' - tr / tr, tr']$$

- Drop and change some of the axioms

USTA1. **Associativity** (semigroup): $(a \widehat{} b) \widehat{} c = a \widehat{} (b \widehat{} c)$
USTA2. **Monoid-like** (zero laws): $a \widehat{} \Phi(a) = a$
USTA3. **Left-cancellative**: $(a \widehat{} b = a \widehat{} c) \Rightarrow b = c$
USTA4. **Zero sum** (right): $(a \widehat{} b = \Phi(b)) \Rightarrow b = \Phi(b)$

Unary Semigroup Trace Algebra (USTA)

Algebra $(\mathcal{T}, \widehat{}, \Phi)$
Where Φ is an idempotent function $\Phi : \mathcal{T} \to \mathcal{T}$ (by construction).

Prefix relation
Defined as before: $a \leq b \mathrel{\widehat{=}} \exists\, c \bullet a \widehat{} c = b$.

Theorem

Provided $(\mathcal{T}, \widehat{}, \Phi)$ is a USTA, then $(\mathcal{T}, \leq)$ is a preorder.

Subtraction
Defined analogously to the monoid trace algebra.

$$b - a = \begin{cases} \iota c.\ a \widehat{} c = b & \text{if } a \leq b \\ \Phi(a) & \text{otherwise} \end{cases}$$

Unary Semigroup Trace Algebra (USTA)

Theorem

Provided $\forall x \bullet \Phi(x) = \varepsilon$, *and* $(\mathcal{T}, \widehat{\ }, \Phi)$ *is a unary semigroup trace algebra, then* $(\mathcal{T}, \widehat{\ }, \varepsilon)$ *is a monoid trace algebra.*

Selection of laws: what changes?

- **TP2. Least elements** $\Phi(x) \leq y$
- **TS1.** $x - \Phi(y) = x$
- ~~**TS2.** $\Phi(y) - x = \Phi(y)$~~
- **TS3.** $x - x = \Phi(x)$
- **TS7.** $y \leq x \wedge x - y = \Phi(y) \Leftrightarrow x = y$

Semigroup properties

Restriction semigroups [2]
We have a **unary semigroup**, whose function Φ satisfies some axioms of restriction semigroups, but not all.

P-Ehresmann semigroups

Theorem

$(\mathcal{T}, \widehat{\ }, \Phi)$ is a right P-Ehresmann semigroup.

- **PE1.** $x \mathbin{\widehat{\ }} \Phi(x) = x$
- **PE2.** $\Phi(x \mathbin{\widehat{\ }} y) = \Phi(\Phi(x) \mathbin{\widehat{\ }} y)$
- **PE3.** $\Phi(\Phi(x) \mathbin{\widehat{\ }} \Phi(y)) = \Phi(y) \mathbin{\widehat{\ }} \Phi(x) \mathbin{\widehat{\ }} \Phi(y)$
- **PE4.** $\Phi(x) \mathbin{\widehat{\ }} \Phi(x) = \Phi(x)$

[2] Peter R. Jones. "A common framework for restriction semigroups and regular *-semigroups". In: *Journal of Pure and Applied Algebra* 216.3 (2012), pp. 618 –632.

Trace models

Traces

Can account for all such varieties:

- Events: $\langle a, b, ..., c \rangle$.
- Events and refusals: $\langle a, \{e\}, ..., c \rangle$.
- Events, refusals and time: $\langle (\langle a \rangle, \{e\}), ... \rangle$.

Prefix relation

With different "orders".

- Events and refusals: $\langle a, \{e\} \rangle \leq \langle a, \{e\} \rangle$
- Events, refusals and time: $\langle (\langle a \rangle, \{e\}) \rangle \leq \langle (\langle a \rangle, \{f\}) \rangle$

Trace models

Timed traces

Events, refusals and time: $\langle(\langle a\rangle, \{e\}), \ldots\rangle$.

Stepwise construction

1. Define parametric pairs $\mathcal{P} : \mathcal{H} \times \mathcal{R}$, where $\mathcal{H}$ is a USTA.
2. Define finite non-empty sequences, whose elements are a USTA.
3. Lift USTA (twice) from pairs $\mathcal{P}$ to finite non-empty sequences.

Parametric pairs

Parametric pairs $\mathcal{P} : \mathcal{H} \times \mathcal{R}$ where $\mathcal{H}$ is a USTA $(\mathcal{H}, +_{\mathcal{H}}, \Phi_{\mathcal{H}})$.

Concatenation & unary function

- $(h_1, \_\_) +_{\mathcal{P}} (h_2, r_2) = (h_1 +_{\mathcal{H}} h_2, r_2)$.
- $\Phi_{\mathcal{P}}(h_1, r_1) = (\Phi_{\mathcal{H}}(h_1), r_1)$

Subtraction

Lemma

Provided $h_2 \leq h_1$, $(h_1, r_1) - (h_2, r_2) = (h_1 - h_2, r_1)$

Lifting the USTA

Theorem

Provided $(\mathcal{H}, +_{\mathcal{H}}, \Phi_{\mathcal{H}})$ *is a USTA then* $(\mathcal{P}, +_{\mathcal{P}}, \Phi_{\mathcal{P}})$ *is a USTA.*

Finite non-empty sequences

Finite non-empty sequences

Datatype $fs ::= One\ \langle\langle \sigma \rangle\rangle \mid Cons\ \langle\langle \sigma \times fs \rangle\rangle$

Concatenation $\frown_{fs}$

Assuming a USTA $(\sigma, \frown_\sigma, \Phi_\sigma)$.

Examples

- A singleton sequence: $\langle a \rangle_{fs}$
- $\langle a \rangle_{fs} \frown_{fs} \langle b \rangle_{fs} = \langle a \frown_\sigma b \rangle_{fs}$
- $\langle a \rangle_{fs} \frown_{fs} \langle b, c, ... \rangle_{fs} = \langle a \frown_\sigma b, c, ... \rangle_{fs}$
- $\langle a, b \rangle_{fs} \frown_{fs} \langle c, ... \rangle_{fs} = \langle a, b \frown_\sigma c, ... \rangle_{fs}$

Sequences of (timed) slots

Unary function

$\Phi_{fs}(x) = \langle \Phi_\sigma(last(x)) \rangle_{fs}$

This is the current time slot, with an empty history.

Example

$$\begin{aligned}
&\Phi_{fs}(\langle(\langle a\rangle, \{b\}), (\langle c\rangle, \{a,b,c\})\rangle_{fs}) \\
&= \langle \Phi_\sigma(last(\langle(\langle a\rangle, \{b\}), (\langle c\rangle, \{a,b,c\})\rangle_{fs}))\rangle_{fs} \\
&= \langle \Phi_{\mathcal{P}}(\langle c\rangle, \{a,b,c\})\rangle_{fs} \\
&= \langle (\Phi_{\mathcal{H}}(\langle c\rangle), \{a,b,c\})\rangle_{fs} \\
&= \langle (\langle\rangle, \{a,b,c\})\rangle_{fs}
\end{aligned}$$

So a fixed point of **R2** is insensitive to the previous history of events and time. $\mathbf{R2}(P) = P[\Phi_{fs}(tr), tr' - tr / tr, tr']$

Lifting

Lifting the USTA (twice)

To non-empty sequences.

Theorem

Provided $(\sigma, +_\sigma, \Phi_\sigma)$ *is a USTA, then* $(fs, \widehat{\ }_{fs}, \Phi_{fs})$ *is a USTA.*

From parametric pairs $\mathcal{P} : \mathcal{H} \times \mathcal{R}$ to non-empty sequences.

Corollary

If $(\mathcal{H}, +_\mathcal{H}, \Phi_\mathcal{H})$ *is a USTA, then* $(fs, \widehat{\ }_{fs}, \Phi_{fs})$ *is a USTA.*

UTP for synchronous process calculi

Mechanised in Isabelle/UTP

- USTA axioms defined via type classes.
- 4 independent axioms vs. circa 26 in [3].
- Established that main results of [3] follow from axioms.
- Hierarchy of algebras.
- Surprisingly little impact on existing proofs.
- Compatible, at least, up to the theory of CSP.

[3] A. Butterfield, A. Sherif and J. C. P. Woodcock. "Slotted *Circus*: A UTP-family of reactive theories". In: *Integrated Formal Methods*. Vol. 4591. LNCS. Springer-Verlag, 2007, pp. 75–97.

Conclusions

Contributions

- Weakened the monoid trace algebra to support pre-orders.
- Widened the foundation for reactive theories in Isabelle/UTP.

Future work

- Integration with main Isabelle/UTP distribution.
- Exploring further weakenings of the algebra.
- Exploit mechanisation of further synchronous process calculi.

Thank you

www.cs.york.ac.uk/robostar/