Automated Temporal Verification via Rewriting Conditional Dependent Effects

Yahui Song and Wei-Ngan Chin @ APLAS 2019

National University of Singapore

Overview

We propose a new solution for temporal verification using a term rewriting system on *Conditional Dependent Effects.*

- **Temporal Effects:** extended regular expressions.
- **Term Rewriting System (t.r.s):** solves regular expressions inequalities by iterated checking of their partial derivatives.
- **Proof System:** soundly deal with mixed inductive and co-inductive definitions by using cyclic proof principle.

Benefits:

1) Provides an expressive temporal specification of all the possible finite/infinite event sequences of a program.
2) Soundly automates the modular local temporal verification.

Temporal Effects

```
send n =
    if n == 0 then event [Done];
    else event [Send]; send (n - 1);
```

Specifications:

Φ' = (Send* · Done, Send^w) *[Martain 2014]*

Φ'' = (Send^n · Done, Send^w) *[Yoji 2018]*

Φ_{pre} = True ∧ Ready · _*

Φ_{post}(n) = (n ≥ 0 ∧ Send^n · Done) ∨ (n < 0 ∧ Send^w)

Hoare-style Forward Verifier: {Φ_{pre}} e {Φ_{post}}

Φ_{then} = n = 0 ∧ Done

Φ_{else} = n /= 0 ∧ Send · Φ_{post}(n-1)

$\Phi_{if\text{-}else}$ = Φ_{then} ∨ Φ_{else}

Effects Entailment Checking:

GOAL: To check $\Phi_{if\text{-}else} \sqsubseteq \Phi_{post}(n)$.

$$(n = 0 \wedge \mathrm{Done}) \vee (n > 0 \wedge \mathrm{Send} \cdot \mathrm{Send}^{n-1} \cdot \mathrm{Done}) \vee (n < 0 \wedge \mathrm{Send}^{w})$$
$$\sqsubseteq$$
$$(n \geq 0 \wedge \mathrm{Send}^{n} \cdot \mathrm{Done}) \vee (n < 0 \wedge \mathrm{Send}^{w})$$

Challenges:

- Dealing with mixed inductive and co-inductive definitions.
- Regular expression with branching properties

Regular Expressions Containment Problem

A : a finite set of alphabet

E = ⊥ | ϵ | a | E · E | E ∨ E

For r, s ∈ E, to check if r ≼ s is valid

PSPACE Complete

Translation into automata (State Explosion)

Term Rewriting System (t.r.s) i.e. symbolic decision procedure

Formal Specifications

$$
\begin{array}{rrcl}
(\textit{Effects}) & \Phi & ::= & \pi \wedge \mathtt{es} \mid \Phi_1 \vee \Phi_2 \mid \exists x.\Phi \\
(\textit{Event Sequences}) & \mathtt{es} & ::= & \bot \mid \epsilon \mid \underline{a} \mid \mathtt{es}_1 \cdot \mathtt{es}_2 \mid \mathtt{es}_1 \vee \mathtt{es}_2 \mid \boxed{\_ \mid \mathtt{es}^{t} \mid \mathtt{es}^{\omega}} \\
(\textit{Pure Formulae}) & \pi & ::= & \mathtt{True} \mid \mathtt{False} \mid A(t_1, t_2) \mid \pi_1 \wedge \pi_2 \mid \pi_1 \vee \pi_2 \\
 & & & \mid \neg\pi \mid \pi_1 \Rightarrow \pi_2 \mid \forall x.\pi \mid \exists x.\pi \\
(\textit{Integer Terms}) & t & ::= & n \mid \star \mid x \mid t_1 + t_2 \mid t_1 - t_2 \\
(\textit{Residue}) & \gamma_R & ::\in & \Phi \\
(\textit{Entailment Context}) & \Gamma & ::= & \emptyset \mid \Gamma,\ \phi_1 \sqsubseteq \phi_1 \quad (\phi_1, \phi_2 \in \Phi)
\end{array}
$$

$x ::\in \mathbf{Var}$ $\quad n ::\in \mathbb{Z}$ $\quad (\textit{Event})\ \underline{a} ::\in \Sigma$ $\quad (\textit{Infinity})\ \omega$ $\quad (\textit{Kleene Star})\ \star$

Some Entailment Rules

$$\frac{\Gamma \vdash \Phi_1 \sqsubseteq \Phi \rightsquigarrow \gamma_R^1 \qquad \Gamma \vdash \Phi_2 \sqsubseteq \Phi \rightsquigarrow \gamma_R^2}{\Gamma \vdash \Phi_1 \vee \Phi_2 \sqsubseteq \Phi \rightsquigarrow (\gamma_R^1 \vee \gamma_R^2)}\ [\text{ENT-LHS-OR}]$$

$$\frac{\Gamma \vdash [w/v]\Phi \sqsubseteq_V^k \Phi_1 \rightsquigarrow \gamma_R}{\Gamma \vdash \exists v.\ \Phi \sqsubseteq_V^k \Phi_1 \rightsquigarrow \gamma_R}\ (\text{fresh } w)\ [\text{ENT-LHS-EX}]$$

$$\frac{\pi' = (s = t \oplus n \wedge s \geq 0) \qquad \Gamma \vdash (\pi_1 \wedge \pi') \wedge \mathtt{es}_1{}^{s} \cdot \mathtt{es} \sqsubseteq (\pi_2 \wedge \pi') \wedge \mathtt{es}_2}{\Gamma \vdash \pi_1 \wedge (\mathtt{es}_1{}^{t \oplus n} \cdot \mathtt{es}) \sqsubseteq \pi_2 \wedge \mathtt{es}_2}\ (\text{fresh } s)\ [\text{ENT-LHS-SUBSTITUTE}]$$

$$\frac{\Gamma \vdash ((\pi_1 \wedge t{=}0) \wedge \mathtt{es}) \vee ((\pi_1 \wedge t > 0) \wedge \mathtt{es}_1 \cdot \mathtt{es}_1^{t-1} \cdot \mathtt{es}) \sqsubseteq \pi_2 \wedge \mathtt{es}_2}{\Gamma \vdash \pi_1 \wedge (\mathtt{es}_1{}^{t} \cdot \mathtt{es}) \sqsubseteq \pi_2 \wedge \mathtt{es}_2}\ [\text{ENT-LHS-CASESPLIT}]$$

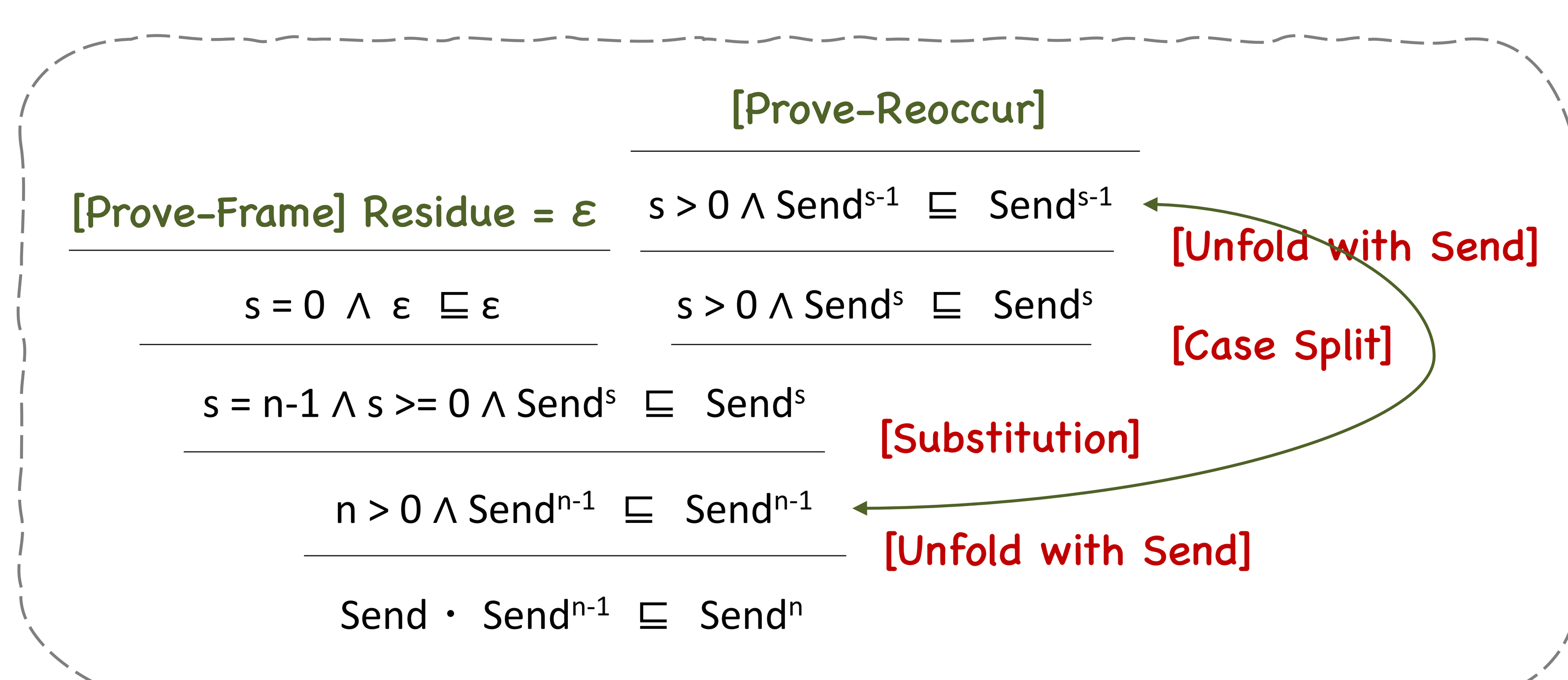
$$\frac{F = \mathtt{fst}_{\pi_1}(\mathtt{es}_1) \qquad \Gamma' = \Gamma,\ \pi_1 \wedge \mathtt{es}_1 \sqsubseteq \pi_2 \wedge \mathtt{es}_2 \qquad \forall f \in F.\ (\Gamma' \vdash D_f^{\pi_1}(\mathtt{es}_1) \sqsubseteq D_f^{\pi_2}(\mathtt{es}_2) \rightsquigarrow \gamma_R^f)}{\Gamma \vdash \pi_1 \wedge \mathtt{es}_1 \sqsubseteq \pi_2 \wedge \mathtt{es}_2 \rightsquigarrow \gamma_R^f}\ [\text{ENT-UNFOLD}]$$

$$\frac{\delta_{\pi_1}(\mathtt{es}_1) \wedge \neg\delta_{\pi_1 \wedge \pi_2}(\mathtt{es}_2)}{\Gamma \vdash \pi_1 \wedge \mathtt{es}_1 \not\sqsubseteq \pi_2 \wedge \mathtt{es}_2}\ [\text{ENT-DISPROVE}]$$

$$\frac{\pi_1 \Rightarrow \pi_2 \qquad \gamma_R = \mathtt{es}_1}{\Gamma \vdash (\pi_1 \wedge \mathtt{es}_1 \sqsubseteq \pi_2 \wedge \epsilon) \rightsquigarrow \gamma_R}\ [\text{ENT-FRAME}]$$

$$\frac{\pi_1 \Rightarrow \pi_2 \qquad (\pi_1 \wedge \mathtt{es}_1 \sqsubseteq \pi_2 \wedge \mathtt{es}_2) \in \Gamma}{\Gamma \vdash \pi_1 \wedge \mathtt{es}_1 \sqsubseteq \pi_2 \wedge \mathtt{es}_2}\ [\text{ENT-REOCCUR}]$$

Example — A Cyclic Proof Tree



Check out our online demo and more information >>>

{yahuis, chinwn}@comp.nus.edu.sg.