Operational Algorithmic Game Semantics

Benedict Bunting Andrzej Murawski

Department of Computer Science
University of Oxford

UNIVERSITY OF OXFORD

LICS, 2023

Overview

- We study **contextual equivalence** for a **call-by-push-value** language with first-order state.
- We build a fully abstract model for this language using **Operational Game Semantics**.
- We refine this model into an automaton, for selected types.
- Using this we derive an algorithm for determining if two terms are contextually equivalent.
- We subsume all decidability results for finitary Idealised Algol and RML with iteration, achieved using finite alphabet automata.

Call-By-Push-Value (Levy, 1999, TLCA)

$$\begin{array}{lcll} \text{Value Type} & \sigma & \triangleq & U\underline{\tau} \mid \text{Unit} \mid \text{Int} \mid \text{Ref} \\ \text{Computation Type} & \underline{\tau} & \triangleq & F\sigma \mid \sigma \to \underline{\tau} \end{array}$$

$$\frac{}{\Sigma;\Gamma \vdash^{v} () : \text{Unit}} \qquad \frac{n \in \{0, \cdots, \max\}}{\Sigma;\Gamma \vdash^{v} \widehat{n} : \text{Int}} \qquad \frac{(x,\sigma) \in \Gamma}{\Sigma;\Gamma \vdash^{v} x : \sigma}$$

$$\frac{\Sigma;\Gamma \vdash^{c} M : \underline{\tau}}{\Sigma;\Gamma \vdash^{v} \text{thunk } M : U\underline{\tau}} \qquad \frac{\Sigma;\Gamma \vdash^{v} V : U\underline{\tau}}{\Sigma;\Gamma \vdash^{c} \text{force } V : \underline{\tau}} \qquad \frac{\Sigma;\Gamma \vdash^{v} V : \sigma}{\Sigma;\Gamma \vdash^{c} \text{return } V : F\sigma}$$

$$\frac{\Sigma;\Gamma \vdash^{c} M : F\sigma \qquad \Sigma;\Gamma, x : \sigma \vdash^{c} N : \underline{\tau}}{\Sigma;\Gamma \vdash^{c} M \text{ to } x.N : \underline{\tau}} \qquad \frac{\Sigma;\Gamma, x : \sigma \vdash^{c} M : \underline{\tau}}{\Sigma;\Gamma \vdash^{c} \lambda x^{\sigma}.M : \sigma \to \underline{\tau}}$$

$$\frac{\Sigma;\Gamma \vdash^{c} M : \sigma \to \underline{\tau} \qquad \Sigma;\Gamma \vdash^{v} V : \sigma}{\Sigma;\Gamma \vdash^{c} MV : \underline{\tau}} \qquad \frac{\Sigma;\Gamma \vdash^{v} V : \text{Int}}{\Sigma;\Gamma \vdash^{c} \text{ref } V : F\text{Ref}}$$

$$\frac{\Sigma;\Gamma \vdash^{v} V_{\text{read}} : UF\text{Int} \qquad \Sigma;\Gamma \vdash^{v} V_{\text{write}} : U(\text{Int} \to F\text{Unit})}{\Sigma;\Gamma \vdash^{v} \text{MkVar } V_{\text{read}}\ V_{\text{write}} : \text{Ref}}$$

CBPV: Contextual Pre-order and Equivalence

Termination: $(M, h) \Downarrow_{ter}$

$$
\begin{array}{llcl}
\text{Value Context} & V_C & \triangleq & \text{thunk } C \mid \text{MkVar } V_C\ V \mid \text{MkVar } V\ V_C \\
\text{Context} & C & \triangleq & \bullet \mid \text{force } V_C \mid \text{return } V_C \mid \lambda x^{\sigma}.C \mid \text{let } x \text{ be } V_C.M \\
 & & & \mid \text{let } x \text{ be } V.C \mid C \text{ to } x.M \mid M \text{ to } x.C \\
 & & & \mid CV \mid MV_C \mid !V_C \mid V_C := V \\
 & & & \mid \text{while}\, C\, \text{do}\, M \mid \text{while}\, M\, \text{do}\, C \\
 & & & \mid \text{case } V \text{ of } (M_i)_{i<j}, C, (M_i)_{j<i}
\end{array}
$$

Definition

Given computations $\Gamma \vdash^c M_1, M_2 : \underline{\tau}$, we define $\Gamma \vdash^c M_1 \lesssim^{\mathrm{CBPV}}_{ter} M_2$ to hold, when for all contexts $\vdash^k C : \underline{\tau} \implies F\sigma$, we have $(C[M_1], \emptyset) \Downarrow_{ter}$ implies $(C[M_2], \emptyset) \Downarrow_{ter}$.

We write $\cong^{\mathrm{CBPV}}_{ter}$ for the equivalence induced by $\lesssim^{\mathrm{CBPV}}_{ter}$.

CBPV: Contextual Equivalence Example

Example

Let $\Gamma = \{f : U(UF\mathrm{Int} \to F\mathrm{Unit})\}$

$$
\begin{aligned}
M_1 &\triangleq (\mathrm{force}\ f)\ \mathrm{thunk}\ (\mathrm{return}\ \widehat{1}) \\
M_2 &\triangleq \mathrm{ref}\,\widehat{0}\ \mathrm{to}\ x.x := \widehat{1}; (\mathrm{force}\ f)\ \mathrm{thunk}\ (!x\ \mathrm{to}\ z.\mathrm{return}\ z)
\end{aligned}
$$

Then $M_1 \cong_{ter}^{\mathrm{CBPV}} M_2$

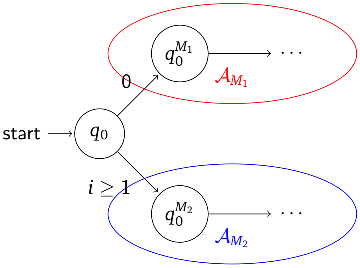
Game Semantics

- Idea: model a term as a strategy for a game of question and answer between it and its context.
- Originally due to Abramsky, Jagadeesan and Malacaria, Hyland and Ong, and Nickau (all early 90's).
- $\Gamma \vdash M \simeq_{ctx} N : \Theta \iff [\![\Gamma \vdash M : \Theta]\!] = [\![\Gamma \vdash N : \Theta]\!]$
- $[\![\Gamma \vdash M\,N : \Theta_2]\!] = \langle [\![\Gamma \vdash M : \Theta_1 \to \Theta_2]\!], [\![\Gamma \vdash N : \Theta_1]\!] \rangle ; \mathrm{ev}$

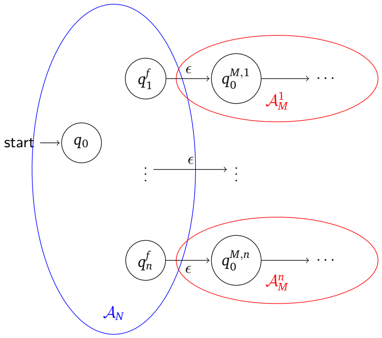
Algorithmic Game Semantics

- Use automata to represent strategies (and so terms).
- $M_1 \simeq_{ctx} M_2$ iff $\mathcal{L}(\mathcal{A}_{M_1}) = \mathcal{L}(\mathcal{A}_{M_2})$
- Early results by (Ghica and McCusker, 2000, ICALP), (Ong, 2002, LICS), (Murawski and Walukiewicz, 2005, FoSSaCS), (Hopkins et al., 2011, ICALP).

$\mathcal{A}_{\text{if } x^\beta \text{ then } M_1 \text{ else } M_2}$



$\mathcal{A}_{\text{let } x^\beta = N \text{ in } M}$



Operational Game Semantics

- Strategies represented as sets of traces (concrete!)
- ...generated by a labelled transition system (LTS) derived from the operational semantics.
- Most relevant results by (Laird, 2007, ICALP), (Jaber, 2015, FoSSaCS), we follow (Jaber and Murawski, 2021, ESOP).
- Other related work: (Jeffrey and Rathke, 2005, ESOP), (Lassen and Levy, 2007, CSL).
- We claim it is 'easier' to obtain an automaton from OGS.

Example

$\bar{f}(r\, w\, \widehat{4}, c_1)\ w(\widehat{1}, c_2)\ \bar{c_2}(())\ c_1(())\ \bar{c}(g)\ g(h, c_3)\ \bar{h}(\epsilon, c_4)\ c_4(\widehat{2})\ \bar{c_3}(\widehat{3})$

OGS: Transition System $\mathcal{L}_{\text{CBPV}}$

$$C_M^{\rho,c} \triangleq (\langle M\{\rho\}, c, \emptyset, N_O, \emptyset, [N_O \mapsto \emptyset]\rangle, \bot)$$

$(P\tau)$ $\langle M, c, \gamma, \phi, h, H\rangle \xrightarrow{\tau} \langle N, c, \gamma, \phi, h', H\rangle$
when $(M, h) \to (N, h')$

(PA) $\langle \text{return } V, c, \gamma, \phi, h, H\rangle \xrightarrow{\bar{c}(A)} \langle \gamma \cdot \gamma', \phi \uplus \nu(A), h, H, H(c) \uplus \nu(A)\rangle$
when $c : \sigma$, $(A, \gamma') \in \mathbf{AVal}_\sigma(V)$

(PQ) $\langle K[(\text{force } f)\,\overrightarrow{V}], c, \gamma, \phi, h, H\rangle \xrightarrow{\bar{f}(\overrightarrow{A}, c')/(c', (K, c))} \langle \gamma \cdot \gamma', \phi \uplus \phi', h, H, H(f) \uplus \nu(\overrightarrow{A})\rangle$
when $f : U\underline{\tau}$, $(\overrightarrow{A}, \gamma') \in \mathbf{AVal}(\overrightarrow{V})$, $\sigma = \mathbf{RType}(\underline{\tau})$, $c' : \sigma$ and $\phi' = \nu(\overrightarrow{A}) \uplus \{c'\}$

(OA) $\langle \gamma, \phi, h, H, Fn\rangle \xrightarrow{c(A), (c, (K, c'))} \langle K[\text{return } A], c', \gamma, \phi \uplus \nu(A), h, H \cdot [\nu(A) \mapsto Fn]\rangle$
when $c : \sigma$, $A : \sigma$

(OQ) $\langle \gamma, \phi, h, H, Fn\rangle \xrightarrow{f(\overrightarrow{A}, c)} \langle (\text{force } V)\,\overrightarrow{A}, c, \gamma, \phi \uplus \phi', h, H \cdot [\phi' \mapsto Fn]\rangle$
when $f \in Fn$, $f : U\underline{\tau}$, $\overrightarrow{A} \in \mathbf{ASeq}(\underline{\tau})$, $\sigma = \mathbf{RType}(\underline{\tau})$, $c : \sigma$, $\gamma(f) = V$ and $\phi' = \nu(\overrightarrow{A}) \uplus \{c\}$

Given $N \subseteq$ Names, $[N \mapsto \mathcal{V}]$ stands for the map $[n \mapsto \mathcal{V} \mid n \in N]$.

Full Abstraction

Theorem (Full Abstraction)

For any CBPV *computations* $\Gamma \vdash^c M_1, M_2 : F\sigma$, *then* $\Gamma \vdash^c M_1 \lesssim_{ter}^{\mathrm{CBPV}} M_2$ *iff* $\mathbf{Tr}_{\mathrm{CBPV}}(\Gamma \vdash^c M_1) \subseteq \mathbf{Tr}_{\mathrm{CBPV}}(\Gamma \vdash^c M_2)$.

Example

$$M_i \triangleq \text{force}\, f \text{ to } g_1.\text{force}\, f \text{ to } g_2.\text{force}\, g_i$$

$\bar{f}(\epsilon, c_1)\ c_1(g_1)\ \bar{f}(\epsilon, c_2)\ c_2(g_2)\ \bar{g_1}(\epsilon, c_3)\ c_3(())\ \bar{c_0}(())$ is in $\mathbf{Tr}(\mathsf{C}_{M_1}^{\rho,c_0})$ but not $\mathbf{Tr}(\mathsf{C}_{M_2}^{\rho,c_0})$

$\bar{f}(\epsilon, c_1)\ c_1(g_1)\ \bar{f}(\epsilon, c_2)\ c_2(g_2)\ \bar{g_2}(\epsilon, c_3)\ c_3(())\ \bar{c_0}(())$ is in $\mathbf{Tr}(\mathsf{C}_{M_2}^{\rho,c_0})$ but not $\mathbf{Tr}(\mathsf{C}_{M_1}^{\rho,c_0})$

Finite Alphabet Traces

For $M_i = \text{force } f \text{ to } g_1.\text{force } f \text{ to } g_2.\text{force } g_i$, traces have the form

$$\mathtt{t}_i = \bar{f}(\epsilon, c_1)\ c_1(g_1)\ \bar{f}(\epsilon, c_2)\ c_2(g_2)\ \bar{g_i}(\epsilon, c_3)\ c_3(())\ \bar{c_0}(())$$

$$\mathbf{Base}^{\Delta}(\mathbf{Tr}(\mathsf{C}^{\rho,c_0}_{M_i})) = \{\mathtt{t} = \bar{f}(\epsilon, d)\ d(g)\ \bar{f}(\epsilon, d)\ d(g)\ \bar{g}(\epsilon, e)\ e(())\ \bar{c_0}(())\}$$

$$\begin{aligned}
\mathbf{Rename}^{\Delta}(\mathbf{Tr}(\mathsf{C}^{\rho,c_0}_{M_1})) = \{ & \bar{f}(\epsilon, d)\ d(g)\ \bar{f}(\epsilon, d)\ d(g)\ \bar{g}(\epsilon, e)\ e(())\ \bar{c_0}(()), \\
& \bar{f}(\epsilon, d)\ d(\hat{g})\ \bar{f}(\epsilon, d)\ d(g)\ \bar{\hat{g}}(\epsilon, e)\ e(())\ \bar{c_0}(()), \\
& \bar{f}(\epsilon, d)\ d(g)\ \bar{f}(\epsilon, d)\ d(\hat{g})\ \bar{g}(\epsilon, e)\ e(())\ \bar{c_0}(())\} \\
\mathbf{Rename}^{\Delta}(\mathbf{Tr}(\mathsf{C}^{\rho,c_0}_{M_2})) = \{ & \bar{f}(\epsilon, d)\ d(g)\ \bar{f}(\epsilon, d)\ d(g)\ \bar{g}(\epsilon, e)\ e(())\ \bar{c_0}(()), \\
& \bar{f}(\epsilon, d)\ d(\hat{g})\ \bar{f}(\epsilon, d)\ d(g)\ \bar{g}(\epsilon, e)\ e(())\ \bar{c_0}(()), \\
& \bar{f}(\epsilon, d)\ d(g)\ \bar{f}(\epsilon, d)\ d(\hat{g})\ \bar{\hat{g}}(\epsilon, e)\ e(())\ \bar{c_0}(())\}
\end{aligned}$$

Corollary

For PTR-*computations* $\Gamma \vdash^c M_1, M_2 : F\sigma$*, continuation name* $c : \sigma$*, a* Γ*-assignment* ρ *and* $\Delta = (\mathrm{TB}, \mathrm{CB}, \rho, c, \mathrm{Suc_T}, \mathrm{Suc_C})$*, we have* $\mathbf{Tr}(\mathsf{C}^{M_1}_{\rho,c}) = \mathbf{Tr}(\mathsf{C}^{M_2}_{\rho,c})$ *iff* $\mathbf{Rename}^{\Delta}(\mathbf{Tr}(\mathsf{C}^{M_1}_{\rho,c})) = \mathbf{Rename}^{\Delta}(\mathbf{Tr}(\mathsf{C}^{M_2}_{\rho,c}))$.

Recycling Scheme I

$$N_i \triangleq (\text{force } f)\ (\text{thunk return } \widehat{1}) \text{ to } g_1.$$
$$(\text{force } f)\ (\text{thunk return } \widehat{2}) \text{ to } g_2.\text{force } g_i$$

Need to distinguish occurrences of the same name during reduction; use **indexing**.

- Consider $t\ f(\overrightarrow{A}, c)\ t'\ \bar{c}(A')\ t''$, when A' has a non-thunk type
- Names introduced in $f(\overrightarrow{A}, c)\ t'$ can never appear in t''

$$\langle \text{return } V, c^i, \gamma, h, H, i_h, \eta, \mu, l\rangle \xrightarrow{\bar{c}(V)} \langle \gamma_{<\eta'}, h_{<i'_h}, H_{<\eta'}, H(c^i), i'_h, \eta', \mu_{<\eta'}, l\rangle$$

when $c \neq c_0$ and $(i'_h, \eta') = \mu(c^i)$

Recycling Scheme II

- Consider $t\ f(\overrightarrow{A}, c)\ t'\ \bar{g}(\overrightarrow{A'}, d)\ t''\ d(A)$ when $\overrightarrow{A'}$ contains no thunk names, g originated in $\overrightarrow{A}$
- Then names introduced in $f(\overrightarrow{A}, c)\ t'\ \bar{g}(\overrightarrow{A'}, d)$ cannot be used in t''

$$(PQ)\quad \langle K[(\text{force } f^i)\overrightarrow{V}], c'^j, \gamma, h, H, i_h, \eta, \mu, l\rangle \xrightarrow{\bar{f}(\overrightarrow{V}, c)/(c^0, (K, c'^j), P)} \langle \gamma_{<\eta'}, h_{<i'_h}, H_{<\eta'}, H(f^i), i'_h, \eta', \mu_{<\eta'}, l\rangle$$

when f is a level 2 name, and $(i'_h, \eta') = \mu(f^i)$, $\mathrm{Succ}_{\mathrm{C}}(f) = c$,
and $P = (i_h, \eta, \gamma_{\geq\eta'}, h_{\geq i'_h}, H_{\geq\eta'}, \mu_{\geq\eta'})$

$$(OA)\quad \langle \gamma, h, H, Fn, i_h, \eta, \mu, l\rangle \xrightarrow{c(\beta(A)), (c^0, (K, c'^j), P)} \langle K[\text{return } A], c'^j, \gamma\cdot\gamma', h, H', i'_h, \eta', \mu', l'\rangle$$

when $c : \sigma$, $P = (i'_h, \eta'', \gamma', h', H'', \mu'')$, $(A', \eta') \in \mathbf{IVals}^{\Delta}_{\sigma}(c, \eta'')$
and if $l = 1$ then $A = A', l' = 1$
else $A \in \mathbf{Select}(A')$, and $l' = \mathbf{IsMark}(A)$; and
$H' = H \cdot H'' \cdot [\nu(A) \mapsto Fn]$, and $\mu' = \mu \cdot \mu'' \cdot [\nu(A) \mapsto (i_h, \eta)]$

Refined transition system $\mathcal{L}^{\Delta}_{\mathrm{PTR}}$

$(P\tau)$ $\langle M, c^j, \gamma, h, H, i_h, \eta, \mu, l\rangle \xrightarrow{\tau} \langle N, c^j, \gamma_{<\eta'}, h', H_{<\eta'}, i'_h, \eta', \mu_{<\eta'}, l\rangle$
when $(M, h, i_h, \eta) \to_e (N, h', i'_h, \eta')$

(PA) $\langle \mathrm{return}\ V, c_0^0, \gamma, h, H, i_h, \eta, \mu, l\rangle \xrightarrow{\bar{c_0}(\beta(A))} \langle \gamma \cdot \gamma', h, H, H(c_0) \uplus \nu(A), i_h, \eta', \mu, l\rangle$
when $c_0 : \sigma$, $(A, \gamma', \eta') = \mathbf{IVal}^{\Delta}_{\sigma}(c_0, V, \eta)$

(PA) $\langle \mathrm{return}\ V, c^i, \gamma, h, H, i_h, \eta, \mu, l\rangle \xrightarrow{\bar{c}(V)} \langle \gamma_{<\eta'}, h_{<i'_h}, H_{<\eta'}, H(c^i), i'_h, \eta', \mu_{<\eta'}, l\rangle$
when $c \neq c_0$ and $(i'_h, \eta') = \mu(c^i)$

(PQ) $\langle K[(\mathrm{force}\ f^i)\overrightarrow{V}], c'^j, \gamma, h, H, i_h, \eta, \mu, l\rangle \xrightarrow{\bar{f}(\beta(\overrightarrow{A}),c)/(c^0,(K,c'^j))} \langle \gamma \cdot \gamma', h, H, H(f^i) \uplus \nu(\overrightarrow{A}), i_h, \eta, \mu, l\rangle$
when f is not a level 2 name, $(\overrightarrow{A}, \gamma', \eta') \in \mathbf{IVal}^{\Delta}(f, \overrightarrow{V}, \eta)$, and $\mathrm{Succ}_{\mathrm{C}}(f) = c$

(PQ) $\langle K[(\mathrm{force}\ f^i)\overrightarrow{V}], c'^j, \gamma, h, H, i_h, \eta, \mu, l\rangle \xrightarrow{\bar{f}(\overrightarrow{V},c)/(c^0,(K,c'^j),P)} \langle \gamma_{<\eta'}, h_{<i'_h}, H_{<\eta'}, H(f^i), i'_h, \eta', \mu_{<\eta'}, l\rangle$
when f is a level 2 name, and $(i'_h, \eta') = \mu(f^i)$, $\mathrm{Succ}_{\mathrm{C}}(f) = c$, and $P = (i_h, \eta, \gamma_{\geq\eta'}, h_{\geq i'_h}, H_{\geq\eta'}, \mu_{\geq\eta'})$

(OA) $\langle \gamma, h, H, Fn, i_h, \eta, \mu, l\rangle \xrightarrow{c(\beta(A)),(c^0,(K,c'^j))} \langle K[\mathrm{return}\ A], c'^j, \gamma, h, H \cdot [\nu(A) \mapsto Fn], i_h, \eta', \mu, l'\rangle$
when $c : \sigma$, $(A', \eta') \in \mathbf{IVals}^{\Delta}_{\sigma}(c, \eta)$ and if $l = 1$ then $A = A', l' = 1$ else $A \in \mathbf{Select}(A')$, and $l' = \mathbf{IsMark}(A)$

(OA) $\langle \gamma, h, H, Fn, i_h, \eta, \mu, l\rangle \xrightarrow{c(\beta(A)),(c^0,(K,c'^j),P)} \langle K[\mathrm{return}\ A], c'^j, \gamma \cdot \gamma', h, H', i'_h, \eta', \mu', l'\rangle$
when $c : \sigma$, $P = (i'_h, \eta'', \gamma', h', H'', \mu'')$, $(A', \eta') \in \mathbf{IVals}^{\Delta}_{\sigma}(c, \eta'')$ and if $l = 1$ then $A = A', l' = 1$
else $A \in \mathbf{Select}(A')$, and $l' = \mathbf{IsMark}(A)$; and $H' = H \cdot H'' \cdot [\nu(A) \mapsto Fn]$, and $\mu' = \mu \cdot \mu'' \cdot [\nu(A) \mapsto (i_h, \eta)]$

(OQ) $\langle \gamma, h, H, Fn, i_h, \eta, \mu, l\rangle \xrightarrow{f(\beta(\overrightarrow{A}),c)} \langle \mathrm{force}\ V\overrightarrow{A}, c^j, \gamma, h, H \cdot [\nu(\overrightarrow{A'}), c^j \mapsto Fn], i_h, \eta', \mu', l\rangle$
when $f^i \in Fn$, $(\overrightarrow{A'}, \eta'') \in \mathbf{IValSeq}^{\Delta}(f, \eta)$, $\mathrm{Succ}_{\mathrm{C}}(f) = c$, $\eta(c) = j, \eta' = \eta''[c \mapsto j+1], \gamma(f^i) = V$, and
if $l = 1$ then $\overrightarrow{A} = \overrightarrow{A'}, l' = 1$ else $\overrightarrow{A} \in \mathbf{Select}(\overrightarrow{A'})$, and $l' = \mathbf{IsMark}(\overrightarrow{A})$; and $\mu' = \mu \cdot [\nu(\overrightarrow{A'}), c^j \mapsto (i_h, \eta)]$

$\mathbf{Tr}_{\mathrm{PTR}}(\Gamma \vdash^c M : F\sigma) \triangleq \{\ (\Delta, t) \mid \Delta$ is a $(\Gamma, F\sigma)$-name scheme,
$t \in \mathbf{Tr}^{\Delta}_{\mathrm{PTR}}(\mathrm{C}^{\mathrm{PTR},\Delta}_{M})$, t is complete$\}$

Correctness and Decidability

Lemma

$\mathbf{Tr}^{\Delta}_{\mathrm{PTR}}(\mathsf{C}^{\mathrm{PTR},\Delta}_{M}) = \mathbf{Rename}^{\Delta}(\mathbf{Tr}_{\mathrm{CBPV}}(\mathsf{C}^{\rho,c_0}_{M}))$.

Theorem (PTR Full Abstraction)

For any PTR *computations* $\Gamma \vdash M_1, M_2 : F\sigma$*, then* $\Gamma \vdash M_1 \lesssim^{\mathrm{CBPV}}_{ter} M_2$ *iff* $\mathbf{Tr}_{\mathrm{PTR}}(\Gamma \vdash M_1) \subseteq \mathbf{Tr}_{\mathrm{PTR}}(\Gamma \vdash M_2)$.

Lemma

For PTR*-computation* $\Gamma \vdash^{c} M : F\sigma$ *and* (Γ, σ)*-name scheme* Δ*, one can effectively construct a deterministic Visibly Pushdown Automaton (Alur and Madhusudan (2004)) accepting* $\mathbf{Tr}^{\Delta}_{\mathrm{PTR}}(\mathsf{C}^{\mathrm{PTR},\Delta}_{M})$*. If* M *is in canonical form, the construction can be carried out in exponential time.*

Theorem

Contextual approximation for the PTR*-fragment of* CBPV *is decidable. For computations in canonical form, it is decidable in exponential time.*

PTR Fragment

- Polarise occurrences of U (corresponding to question actions)
- U^+ for occurrences of U that produce O-questions, and U^- for P-questions
- Problematic types: $FU^+FU^+F\mathrm{Int}$ and $FU^+(U^-(U^+F\mathrm{Int} \to F\mathrm{Unit}) \to F\mathrm{Unit})$
- PTR-fragment: forbids nested occurrences of U^+, while allowing nested occurrences of U^-

Definition

A CBPV computation $\Gamma \vdash^c M : F\sigma^P$ is in the **P-thunk-restricted** (PTR) fragment when all types in Γ can be generated by σ^2 in the grammar below.

$$
\begin{array}{lll}
\sigma^2 & \triangleq & \sigma^1 \mid U\underline{\tau}^2 \\
\sigma^P & \triangleq & \sigma^0 \mid \mathrm{Ref} \mid U\underline{\tau}^P \\
\sigma^1 & \triangleq & \sigma^0 \mid \mathrm{Ref} \mid U\underline{\tau}^1 \\
\sigma^0 & \triangleq & \mathrm{Int} \mid \mathrm{Unit}
\end{array}
\qquad
\begin{array}{lll}
\underline{\tau}^2 & \triangleq & F\sigma^2 \mid \sigma^P \to \underline{\tau}^2 \\
\underline{\tau}^P & \triangleq & F\sigma^0 \mid \sigma^1 \to \underline{\tau}^P \\
\underline{\tau}^1 & \triangleq & F\sigma^1 \mid \sigma^0 \to \underline{\tau}^1
\end{array}
$$

IA and RML

RML type	CBPV value types
$\mathrm{Int}, \mathrm{Unit}, \mathrm{Ref}$	$\mathrm{Int}, \mathrm{Unit}, \mathrm{Ref}$
$\sigma_1 \to \sigma_2$	$U({\sigma_1}^{\mathrm{RML}} \to F{\sigma_2}^{\mathrm{RML}})$

IA type	CBPV computation types
$\underline{\mathrm{expr}}, \underline{\mathrm{com}}$	$F\mathrm{Int}, F\mathrm{Unit}$
$\underline{\mathrm{var}}$	$\mathrm{Int} \to \mathrm{Int} \to F\mathrm{Int}$
$\underline{\tau_1} \to \underline{\tau_2}$	$U{\underline{\tau_1}}^{\mathrm{IA}} \to {\underline{\tau_2}}^{\mathrm{IA}}$

- We provide translations from Idealised Algol and RML into CBPV
- PTR-fragment subsumes
  - Third-Order IA (Murawski and Walukiewicz, 2005, FoSSaCS)
  - O-Strict RML (Hopkins et al., 2011, ICALP)

DFA Fragment

- DFA: $\Gamma \vdash^c M : F\sigma^1$, where each type in Γ is a σ^2

$$
\begin{array}{rcl@{\qquad}rcl}
\sigma^2 & \triangleq & \sigma^1 \mid U\underline{\tau}^1 & \underline{\tau}^1 & \triangleq & F\sigma^2 \mid \sigma^1 \rightarrow \underline{\tau}^1 \\
\sigma^1 & \triangleq & \sigma^0 \mid \mathrm{Ref} \mid U\underline{\tau}^0 & \underline{\tau}^0 & \triangleq & F\sigma^0 \mid \sigma^0 \rightarrow \underline{\tau}^0 \\
\sigma^0 & \triangleq & \mathrm{Int} \mid \mathrm{Unit} & & &
\end{array}
$$

- DFA fragment subsume
  - Second-Order IA (Ghica and McCusker, 2000, ICALP)
  - Bi-Strict $\mathrm{RML}_{\mathrm{f}}^{-}$ (Murawski, 2005, Theor. Comput. Sci.)

Conclusions

- Provide full-abstract trace model for CBPV.
- A new decidability result; first for CBPV.
- Subsume 4 results from the literature.
- A new approach: **Operational Algorithmic Game Semantics**
  - OGS models are (generally) simpler to construct
  - More intuitive
  - Retain operational character suitable for further program analysis tasks
  - We believe that it can be easily adapted for other frameworks
  - Next steps: more powerful automata

Conclusions

- Provide full-abstract trace model for CBPV.
- A new decidability result; first for CBPV.
- Subsume 4 results from the literature.
- A new approach: **Operational Algorithmic Game Semantics**
  - OGS models are (generally) simpler to construct
  - More intuitive
  - Retain operational character suitable for further program analysis tasks
  - We believe that it can be easily adapted for other frameworks
  - Next steps: more powerful automata

Conclusions

- Provide full-abstract trace model for CBPV.
- A new decidability result; first for CBPV.
- Subsume 4 results from the literature.
- A new approach: **Operational Algorithmic Game Semantics**
  - OGS models are (generally) simpler to construct
  - More intuitive
  - Retain operational character suitable for further program analysis tasks
  - We believe that it can be easily adapted for other frameworks
  - Next steps: more powerful automata

Conclusions

- Provide full-abstract trace model for CBPV.
- A new decidability result; first for CBPV.
- Subsume 4 results from the literature.
- A new approach: **Operational Algorithmic Game Semantics**
  - OGS models are (generally) simpler to construct
  - More intuitive
  - Retain operational character suitable for further program analysis tasks
  - We believe that it can be easily adapted for other frameworks
- Next steps: more powerful automata

Conclusions

- Provide full-abstract trace model for CBPV.
- A new decidability result; first for CBPV.
- Subsume 4 results from the literature.
- A new approach: **Operational Algorithmic Game Semantics**
  - OGS models are (generally) simpler to construct
  - More intuitive
  - Retain operational character suitable for further program analysis tasks
  - We believe that it can be easily adapted for other frameworks
  - Next steps: more powerful automata

References I

Abramsky, S., Malacaria, P., and Jagadeesan, R. (1994). Full abstraction for PCF. In Hagiya, M. and Mitchell, J. C., editors, *Theoretical Aspects of Computer Software, International Conference TACS '94, Sendai, Japan, April 19-22, 1994, Proceedings*, volume 789 of *Lecture Notes in Computer Science*, pages 1–15. Springer.

Alur, R. and Madhusudan, P. (2004). Visibly Pushdown Languages. In *Proceedings of STOC*, pages 202–211. ACM.

Ghica, D. R. and McCusker, G. (2000). Reasoning about idealized ALGOL using regular languages. In Montanari, U., Rolim, J. D. P., and Welzl, E., editors, *Automata, Languages and Programming, 27th International Colloquium, ICALP 2000, Geneva, Switzerland, July 9-15, 2000, Proceedings*, volume 1853 of *Lecture Notes in Computer Science*, pages 103–115. Springer.

Hopkins, D., Murawski, A. S., and Ong, C.-H. L. (2011). A Fragment of ML Decidable by Visibly Pushdown Automata. In *Proceedings of ICALP*, volume 6756 of *LNCS*, pages 149–161. Springer.

References II

Hyland, J. M. E. and Ong, C. L. (1995). Pi-calculus, dialogue games and PCF. In Williams, J., editor, *Proceedings of the seventh international conference on Functional programming languages and computer architecture, FPCA 1995, La Jolla, California, USA, June 25-28, 1995*, pages 96–107. ACM.

Jaber, G. (2015). Operational Nominal Game Semantics. In *Proceedings of FoSSaCS*, volume 9034 of *LNCS*, pages 264–278. Springer.

Jaber, G. and Murawski, A. S. (2021). Complete trace models of state and control. In *Proceedings of ESOP*, volume 12648 of *LNCS*, pages 348–374. Springer.

Jeffrey, A. and Rathke, J. (2005). Java jr: Fully abstract trace semantics for a core java language. In Sagiv, S., editor, *Programming Languages and Systems, 14th European Symposium on Programming, ESOP 2005, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2005, Edinburgh, UK, April 4-8, 2005, Proceedings*, volume 3444 of *Lecture Notes in Computer Science*, pages 423–438. Springer.

References III

Laird, J. (2007). A Fully Abstract Trace Semantics for General References. In *Proceedings of ICALP*, volume 4596 of *LNCS*, pages 667–679. Springer.

Lassen, S. B. and Levy, P. B. (2007). Typed normal form bisimulation. In Duparc, J. and Henzinger, T. A., editors, *Computer Science Logic, 21st International Workshop, CSL 2007, 16th Annual Conference of the EACSL, Lausanne, Switzerland, September 11-15, 2007, Proceedings*, volume 4646 of *Lecture Notes in Computer Science*, pages 283–297. Springer.

Levy, P. B. (1999). Call-by-push-value: A subsuming paradigm. In Girard, J., editor, *Typed Lambda Calculi and Applications, 4th International Conference, TLCA'99, L'Aquila, Italy, April 7-9, 1999, Proceedings*, volume 1581 of *Lecture Notes in Computer Science*, pages 228–242. Springer.

Murawski, A. S. (2005). Functions with local state: Regularity and Undecidability. *Theor. Comput. Sci.*, 338(1-3):315–349.

References IV

Murawski, A. S. and Walukiewicz, I. (2005). Third-order idealized algol with iteration is decidable. In Sassone, V., editor, *Foundations of Software Science and Computational Structures, 8th International Conference, FOSSACS 2005, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2005, Edinburgh, UK, April 4-8, 2005, Proceedings*, volume 3441 of *Lecture Notes in Computer Science*, pages 202–218. Springer.

Nickau, H. (1994). Hereditarily Sequential Functionals. In *Proceedings of LFCS*, volume 813 of *LNCS*, pages 253–264. Springer.

Ong, C. L. (2002). Observational equivalence of 3rd-order idealized algol is decidable. In *17th IEEE Symposium on Logic in Computer Science (LICS 2002), 22-25 July 2002, Copenhagen, Denmark, Proceedings*, pages 245–256. IEEE Computer Society.