Interaction Equivalence (and Improvement)

Adrienne Lancelot, University of Bologna

j.w.w. Beniamino Accattoli[1], Giulio Manzonetto[2], Gabriele Vanoni[2]
(and Guy McCusker[3])

[1]Inria & LIX, École Polytechnique
[2]Université Paris Cité, CNRS, IRIF
[3]Univerity of Bath, UK

PERR, July 24th 2026

Program Equivalences

When are two programs equivalent?

In the λ-calculus, natural equivalences arise by identifying terms with the same (possibly infinite) normal form (aka Böhm tree equivalence).

This is not enough, some programs behave in the same way but do not have exactly the same (infinite) normal form.

Contextual Equivalence

t u

When are two λ-terms equivalent? [Mor68]

$$t \equiv^{\mathrm{ctx}} u \quad \text{if for all contexts } C\,. \quad [\; C\langle t\rangle \Downarrow \quad \Leftrightarrow \quad C\langle u\rangle \Downarrow \;]$$

Head contextual equivalence is an **equational theory**,
where $\mathcal{O}$:= having a head normal form.

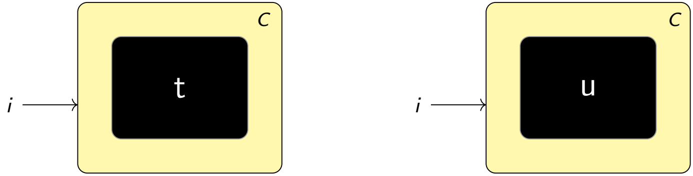
1. *Equivalence Relation*;
2. *Context-Closed*:

$$\text{if } t \equiv^{\mathrm{ctx}} u \text{ then } \forall C,\ C\langle t\rangle \equiv^{\mathrm{ctx}} C\langle u\rangle;$$

3. *Invariance*:

$$\text{if } t =_\beta u \text{ then } t \equiv^{\mathrm{ctx}} u.$$

Contextual Equivalence



When are two λ-terms equivalent? [Mor68]

$$t \equiv^{\mathrm{ctx}} u \quad \text{if for all contexts } C\,. \;\; [\; C\langle t\rangle \Downarrow \;\; \Leftrightarrow \;\; C\langle u\rangle \Downarrow \;]$$

Head contextual equivalence is an **equational theory**,
where $\mathcal{O}$:= having a head normal form.

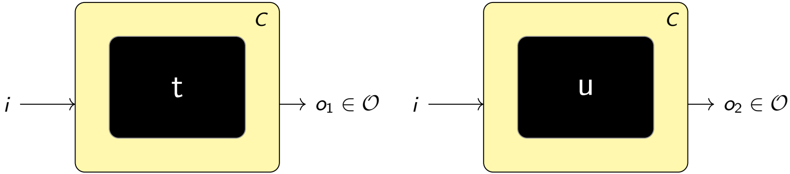
1. *Equivalence Relation*;
2. *Context-Closed*:

$$\text{if } t \equiv^{\mathrm{ctx}} u \text{ then } \forall C,\; C\langle t\rangle \equiv^{\mathrm{ctx}} C\langle u\rangle;$$

3. *Invariance*:

$$\text{if } t =_\beta u \text{ then } t \equiv^{\mathrm{ctx}} u.$$

Contextual Equivalence



When are two λ-terms equivalent? (It depends on the $\mathcal{O}$bservables) [Mor68]

$$t \equiv^{\mathrm{ctx}} u \quad \text{if for all contexts } C\,. \quad [\ C\langle t\rangle \Downarrow_{\mathcal{O}} \quad \Leftrightarrow \quad C\langle u\rangle \Downarrow_{\mathcal{O}}\ ]$$

Head contextual equivalence is an **equational theory**, where $\mathcal{O}$:= having a head normal form.

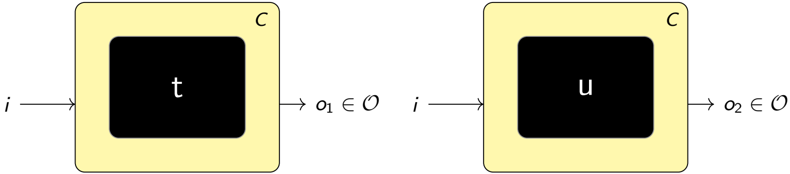
1. *Equivalence Relation*;
2. *Context-Closed*:

$$\text{if } t \equiv^{\mathrm{ctx}} u \text{ then } \forall C,\ C\langle t\rangle \equiv^{\mathrm{ctx}} C\langle u\rangle;$$

3. *Invariance*:

$$\text{if } t =_{\beta} u \text{ then } t \equiv^{\mathrm{ctx}} u.$$

Contextual Equivalence



When are two λ-terms equivalent? (It depends on the $\mathcal{O}$bservables) [Mor68]

$$t \equiv^{\mathrm{ctx}} u \quad \text{if for all contexts } C\,. \quad [\ C\langle t\rangle \Downarrow_{\mathcal{O}} \quad \Leftrightarrow \quad C\langle u\rangle \Downarrow_{\mathcal{O}}\ ]$$

Head contextual equivalence is an **equational theory**, where $\mathcal{O}$:= having a head normal form.

1. *Equivalence Relation*;
2. *Context-Closed*:
$$\text{if } t \equiv^{\mathrm{ctx}} u \text{ then } \forall C,\ C\langle t\rangle \equiv^{\mathrm{ctx}} C\langle u\rangle;$$
3. *Invariance*:
$$\text{if } t =_\beta u \text{ then } t \equiv^{\mathrm{ctx}} u.$$

Sands' improvement & cost equivalence

Sands' **cost equivalence** [San99] is defined as:

$$t \equiv^{\text{cost}} u \quad \text{if } \forall C, \forall k \geq 0\,. \quad [\ C\langle t\rangle \Downarrow_{\mathcal{O}}^{k} \quad \Leftrightarrow \quad C\langle u\rangle \Downarrow_{\mathcal{O}}^{k}\ ]$$

Again, we specify to the λ-calculus, where $\mathcal{O} := \Downarrow_{\mathrm{h}}$.

Not an equational theory!
$\mathtt{I} =_{\beta} \mathtt{I}\,\mathtt{I}$ but $\mathtt{I} \not\equiv^{\text{cost}} \mathtt{I}\mathtt{I}$

Sands' improvement & cost equivalence

Sands' **cost equivalence** [San99] is defined as:

$$t \equiv^{\text{cost}} u \quad \text{if } \forall C, \forall k \geq 0\,. \;\; [\; C\langle t\rangle \Downarrow_{\mathcal{O}}^{k} \quad \Leftrightarrow \quad C\langle u\rangle \Downarrow_{\mathcal{O}}^{k} \;]$$

Again, we specify to the λ-calculus, where $\mathcal{O} := \Downarrow_{\text{h}}$.

Not an equational theory!
$\mathtt{I} =_{\beta} \mathtt{I}\,\mathtt{I}$ but $\mathtt{I} \not\equiv^{\text{cost}} \mathtt{II}$

Interaction Quantities in Programming

How many $\leq$ does it take to check that a list is sorted?

```
let rec is_sorted  l compare = match l with
|a::b::t -> if compare(a,b) then false
else is_sorted (b::t) compare
|_-> let n = fibonacci(42) in true
```

```
let rec my_compare a b = # returns false if b < a
if a < 0 and b < 0 then my_compare(-a,-b)
elif b < 0 then
let k = fact(42) in
false
elif a < 0 then true
else (b-a >= 0)
```

The best of both worlds?

Can we build a cost-sensitive equational theory?

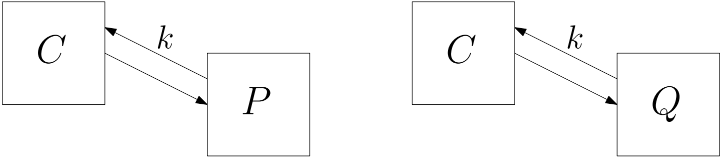
How can we measure the interaction between a program and a context modulo the internal dynamics?

Our contribution: a framework to identify internal and interaction steps for the untyped λ-calculus
$\rightarrow$ checkers λ-calculus

The best of both worlds?

Can we build a cost-sensitive equational theory?

How can we measure the interaction between a program and a context modulo the internal dynamics?



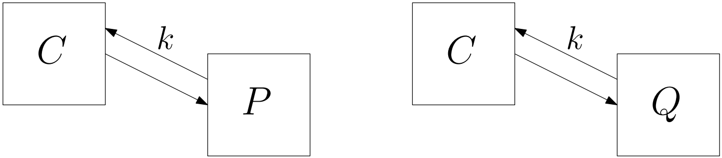
$$P \simeq Q$$

Our contribution: a framework to identify internal and interaction steps for the untyped λ-calculus
$\rightarrow$ checkers λ-calculus

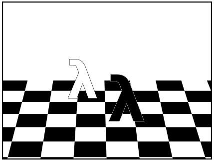
The best of both worlds?

Can we build a cost-sensitive equational theory?

How can we measure the interaction between a program and a context modulo the internal dynamics?



$$P \simeq Q$$



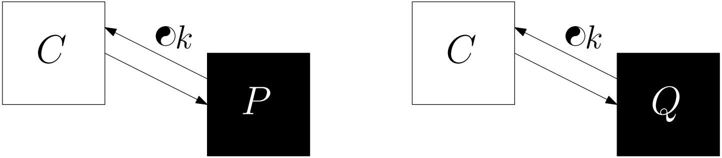
Our contribution: a framework to identify internal and interaction steps for the untyped λ-calculus

$\rightarrow$ checkers λ-calculus

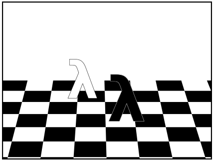
The best of both worlds?

Can we build a cost-sensitive equational theory?

How can we measure the interaction between a program and a context modulo the internal dynamics?



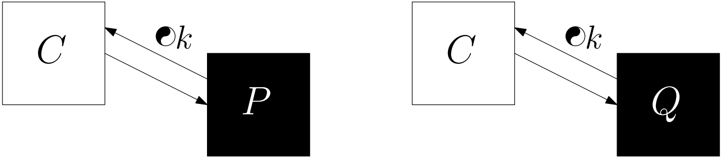
$$P \equiv^{\mathrm{int}} Q$$



Our contribution: a framework to identify internal and interaction steps for the untyped λ-calculus
$\rightarrow$ checkers λ-calculus

What should be the meaning of a program

Interaction Equivalence is an equational theory!



$$P \equiv^{\text{int}} Q$$

- Duality between Program/Context reminiscent of Game Semantics
- Modeling communication $P \mid C$ akin to π-calculus and LTS

"The meaning of a program should express its history of access to resources which are not local to it." – Milner 1975

Inspecting Black Boxes

Contextual equivalences are hard to check! $\forall$-quantifier

Intensional equivalences are easy to check!

$\rightarrow$ Böhm tree equivalence, normal form bisimilarity, set of approximants, building a strategy from a λ-term, executing an operational game semantics, etc.

Second contribution:
interaction equivalence is exactly Böhm tree equivalence

Also answering an **open question** in the λ-calculus: which contextual equivalence can match Böhm tree equivalence?

Inspecting Black Boxes

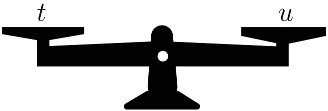
Contextual equivalences are hard to check! ∀-quantifier

Intensional equivalences are easy to check!

→ Böhm tree equivalence, normal form bisimilarity, set of approximants, building a strategy from a λ-term, executing an operational game semantics, etc.

Second contribution:

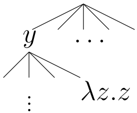
interaction equivalence is exactly Böhm tree equivalence



Interaction Equivalence

$\Leftrightarrow$

$$BT(t) = \lambda x_1 \ldots x_k.x = BT(u)$$



Also answering an **open question** in the λ-calculus: which contextual equivalence can match Böhm tree equivalence?

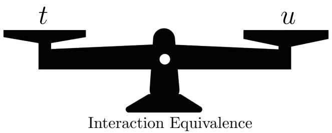
Inspecting Black Boxes

Contextual equivalences are hard to check! $\forall$-quantifier

Intensional equivalences are easy to check!

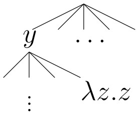
$\rightarrow$ Böhm tree equivalence, normal form bisimilarity, set of approximants, building a strategy from a λ-term, executing an operational game semantics, etc.

Second contribution:
interaction equivalence is exactly Böhm tree equivalence



$\Leftrightarrow$

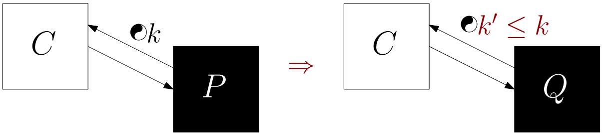
$$BT(t) = \lambda x_1 \ldots x_k . x = BT(u)$$



Also answering an **open question** in the λ-calculus: which contextual equivalence can match Böhm tree equivalence?

Interaction Improvement

Another way to compare interactions:

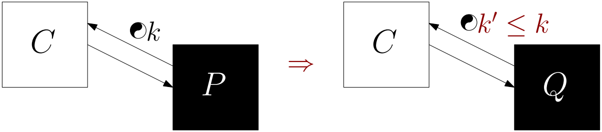


$$P \sqsubseteq^{\mathrm{int\cdot imp}} Q$$

Third contribution towards optimizations:

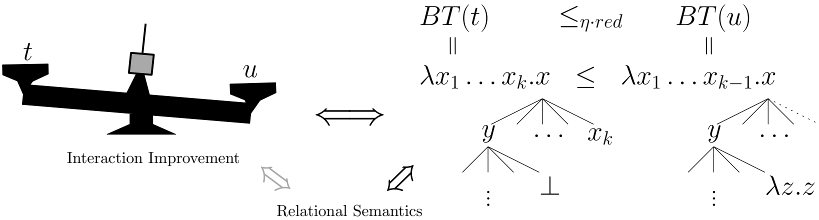
Interaction Improvement

Another way to compare interactions:

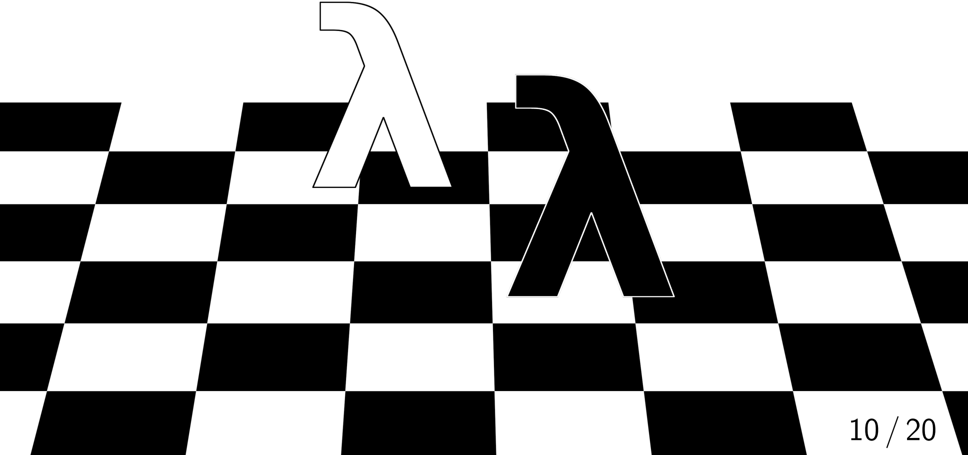


$$P \sqsubseteq^{\mathrm{int\cdot imp}} Q$$

Third contribution towards optimizations:



The Checkers λ-Calculus



The Checkers λ-Calculus

$$
\begin{array}{llllll}
\Lambda \ni t, u & := & x & \mid \lambda x.t & \mid tu \\
\Lambda_{\bullet\circ} \ni \mathfrak{t}, \mathfrak{u} & := & x & \mid \lambda_{\bullet} x.\mathfrak{t} & \mid \mathfrak{t} \bullet \mathfrak{u} \\
 & & & \mid \lambda_{\circ} x.\mathfrak{t} & \mid \mathfrak{t} \circ \mathfrak{u}
\end{array}
$$

Silent Steps:

$(\lambda_{\bullet} x.\mathfrak{t}) \bullet \mathfrak{u} \mapsto_{\beta\tau} \mathfrak{t}\{x := \mathfrak{u}\}$

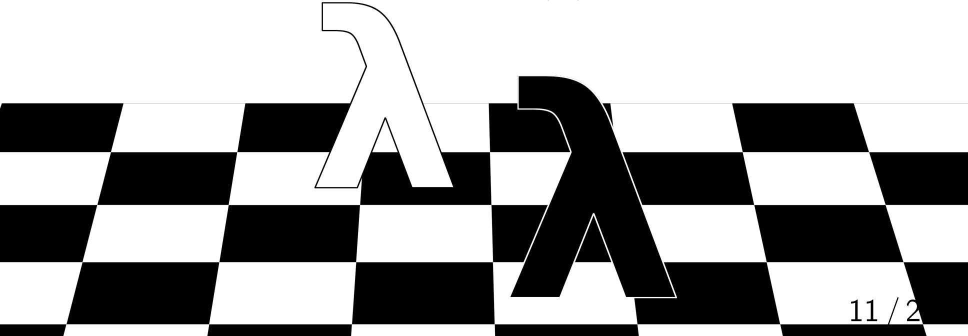
$(\lambda_{\circ} x.\mathfrak{t}) \circ \mathfrak{u} \mapsto_{\beta\tau} \mathfrak{t}\{x := \mathfrak{u}\}$

Interaction Steps:

$(\lambda_{\bullet} x.\mathfrak{t}) \circ \mathfrak{u} \mapsto_{\beta◐} \mathfrak{t}\{x := \mathfrak{u}\}$

$(\lambda_{\circ} x.\mathfrak{t}) \bullet \mathfrak{u} \mapsto_{\beta◐} \mathfrak{t}\{x := \mathfrak{u}\}$

Intuition: $\overline{C}^{\circ}\langle t^{\bullet}\rangle$



The Checkers λ-Calculus

$$
\begin{array}{rclll}
\Lambda \ni t, u & := & x & \mid \lambda x.t & \mid tu \\
\Lambda_{\bullet\circ} \ni \mathfrak{t}, \mathfrak{u} & := & x & \mid \lambda_{\bullet} x.\mathfrak{t} & \mid \mathfrak{t} \bullet \mathfrak{u} \\
& & & \mid \lambda_{\circ} x.\mathfrak{t} & \mid \mathfrak{t} \circ \mathfrak{u}
\end{array}
$$

Silent Steps:

$(\lambda_{\bullet} x.\mathfrak{t}) \bullet \mathfrak{u} \mapsto_{\beta\tau} \mathfrak{t}\{x := \mathfrak{u}\}$

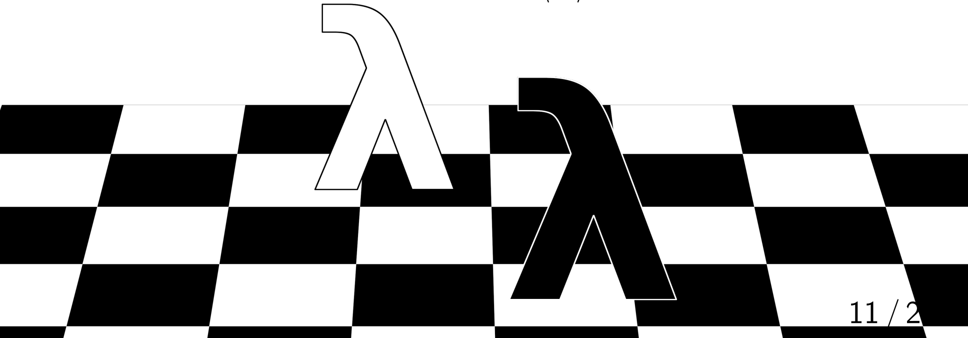
$(\lambda_{\circ} x.\mathfrak{t}) \circ \mathfrak{u} \mapsto_{\beta\tau} \mathfrak{t}\{x := \mathfrak{u}\}$

Interaction Steps:

$(\lambda_{\bullet} x.\mathfrak{t}) \circ \mathfrak{u} \mapsto_{\beta\LEFTcircle} \mathfrak{t}\{x := \mathfrak{u}\}$

$(\lambda_{\circ} x.\mathfrak{t}) \bullet \mathfrak{u} \mapsto_{\beta\LEFTcircle} \mathfrak{t}\{x := \mathfrak{u}\}$

Intuition: $\overline{C}^{\circ}\langle t^{\bullet}\rangle$



The Checkers λ-Calculus

$$\Lambda \ni t, u \quad := \quad x \quad | \; \lambda x.t \quad | \; tu$$

$$\begin{array}{rcll} \Lambda_{\bullet\circ} \ni \mathfrak{t}, \mathfrak{u} & := & x & | \; \lambda_{\bullet} x.\mathfrak{t} \quad | \; \mathfrak{t} \bullet \mathfrak{u} \\ & & & | \; \lambda_{\circ} x.\mathfrak{t} \quad | \; \mathfrak{t} \circ \mathfrak{u} \end{array}$$

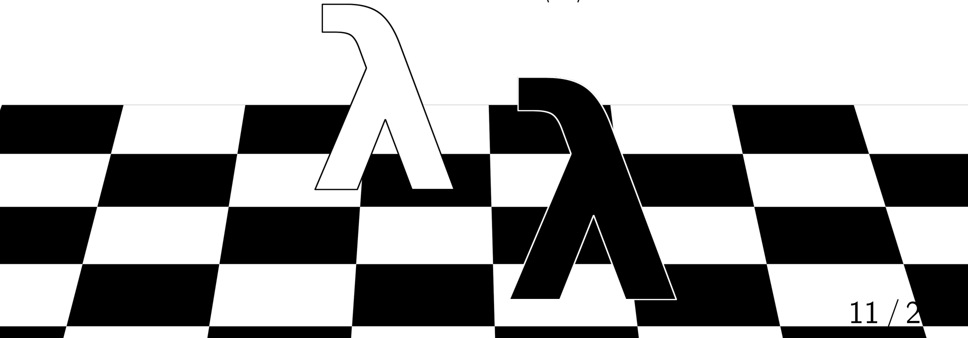
Silent Steps:

$$(\lambda_{\bullet} x.\mathfrak{t}) \bullet \mathfrak{u} \mapsto_{\beta_{\tau}} \mathfrak{t}\{x := \mathfrak{u}\}$$
$$(\lambda_{\circ} x.\mathfrak{t}) \circ \mathfrak{u} \mapsto_{\beta_{\tau}} \mathfrak{t}\{x := \mathfrak{u}\}$$

Interaction Steps:

$$(\lambda_{\bullet} x.\mathfrak{t}) \circ \mathfrak{u} \mapsto_{\beta_{\LEFTcircle}} \mathfrak{t}\{x := \mathfrak{u}\}$$
$$(\lambda_{\circ} x.\mathfrak{t}) \bullet \mathfrak{u} \mapsto_{\beta_{\LEFTcircle}} \mathfrak{t}\{x := \mathfrak{u}\}$$

Intuition: $\overline{C}^{\circ}\langle t^{\bullet} \rangle$



Tags Get Mixed

Intuition: $\overline{C}^{\circ}\langle t^{\bullet}\rangle$

$$C := (\lambda x.\langle\cdot\rangle)\,\mathtt{I}\;\mathtt{I} \qquad t := \lambda y.yx$$

$$\begin{aligned}
\overline{C}^{\circ}\langle t^{\bullet}\rangle \quad &= (\lambda_{\circ} x.\lambda_{\bullet} y.y \bullet x) \circ \mathtt{I}_{\circ} \circ \mathtt{I}_{\circ} \\
&\to_{\beta\tau} (\lambda_{\bullet} y.y \bullet \mathtt{I}_{\circ}) \circ \mathtt{I}_{\circ} \\
&\to_{\beta\text{☯}} \mathtt{I}_{\circ} \bullet \mathtt{I}_{\circ} \\
&\to_{\beta\text{☯}} \mathtt{I}_{\circ}
\end{aligned}$$

Interaction Cost?

Counting interactions depends on the reduction sequence.

$$
\begin{tikzcd}
 & (\lambda_\bullet x.x \bullet x) \bullet (\mathtt{I}_\circ \bullet \mathtt{I}_\bullet) \arrow[dl, "\textbf{1 interaction step}"'] \arrow[dr, "\text{1 silent step}"] & \\
(\lambda_\bullet x.x \bullet x) \bullet \mathtt{I}_\bullet \arrow[dr, "\text{1 silent step}"'] & & (\mathtt{I}_\circ \bullet \mathtt{I}_\bullet) \bullet (\mathtt{I}_\circ \bullet \mathtt{I}_\bullet) \arrow[dl, "\textbf{2 interaction step}"] \\
 & \mathtt{I}_\bullet \bullet \mathtt{I}_\bullet &
\end{tikzcd}
$$

The checkers calculus is **confluent** but does not preserve interaction steps.

We consider the **head interaction cost** :

$\mathfrak{t} \Downarrow_{\mathtt{h}}^{\bullet k}$ means $\mathfrak{t}$ $\mathtt{head}$-normalizes with k interaction steps

Counting Interactions in Contextual Equivalence

We define quantitative contextual relations for the checkers calculus.
Let $\mathfrak{t}, \mathfrak{t}' \in \Lambda_{\bullet\circ}$

- **Quantitative Contextual Equivalence:**
  $\mathfrak{t} \equiv_{◐}^{\mathrm{ctx}} \mathfrak{u}$ if,

  $$\forall \text{ colored } \mathfrak{C}, \forall k, \quad [\ \mathfrak{C}\langle\mathfrak{t}\rangle \Downarrow_{\mathtt{h}}^{◐ k} \iff \mathfrak{C}\langle\mathfrak{u}\rangle \Downarrow_{\mathtt{h}}^{◐ k}\ ];$$

- **Quantitative Contextual Preorder:**
  $\mathfrak{t} \sqsubseteq_{◐}^{\mathrm{ctx}} \mathfrak{u}$ [$\mathfrak{u}$ simulates $\mathfrak{t}$] if,

  $$\forall \text{ colored } \mathfrak{C}, \forall k, \quad [\ \mathfrak{C}\langle\mathfrak{t}\rangle \Downarrow_{\mathtt{h}}^{◐ k} \implies \mathfrak{C}\langle\mathfrak{u}\rangle \Downarrow_{\mathtt{h}}^{◐ k}\ ];$$

Key Point: we ignore silent steps and count interaction steps.

Counting Interactions in Contextual Equivalence

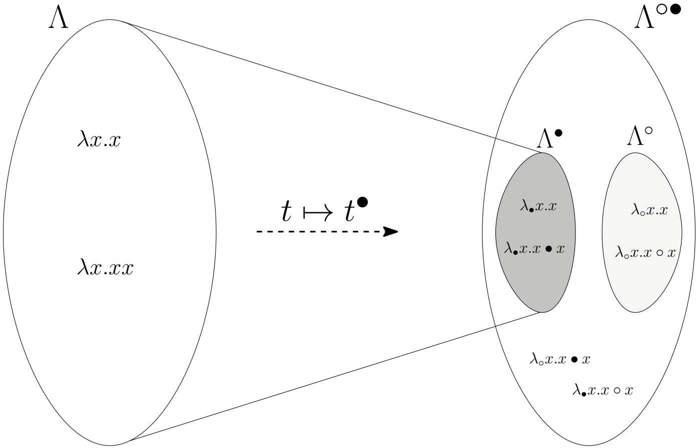
We define quantitative contextual relations for the checkers calculus.
Let $\mathfrak{t}, \mathfrak{t}' \in \Lambda_{\bullet\circ}$

- **Quantitative Contextual Equivalence:**
  $\mathfrak{t} \equiv^{\mathrm{ctx}}_{\bullet} \mathfrak{u}$ if,
  $$\forall \text{ colored } \mathfrak{C}, \forall k, \quad [\ \mathfrak{C}\langle \mathfrak{t} \rangle \Downarrow^{\bullet k}_{\mathtt{h}} \iff \mathfrak{C}\langle \mathfrak{u} \rangle \Downarrow^{\bullet k}_{\mathtt{h}}\ ];$$

- **Quantitative Contextual Preorder:**
  $\mathfrak{t} \sqsubseteq^{\mathrm{ctx}}_{\bullet} \mathfrak{u}$ [$\mathfrak{u}$ simulates $\mathfrak{t}$] if,
  $$\forall \text{ colored } \mathfrak{C}, \forall k, \quad [\ \mathfrak{C}\langle \mathfrak{t} \rangle \Downarrow^{\bullet k}_{\mathtt{h}} \implies \mathfrak{C}\langle \mathfrak{u} \rangle \Downarrow^{\bullet k}_{\mathtt{h}}\ ];$$

Key Point: we ignore silent steps and count interaction steps.

Interaction Equivalence



$$t \sqsubseteq^{\mathrm{int}} u \text{ if } t^\bullet \sqsubseteq^{\mathrm{ctx}}_{\circ\bullet} u^\bullet$$

Note that contexts have both **black or white constructors**.

Interaction Equivalence is an Equational Theory

1. *Equivalence Relation*;
2. *Context-Closed*: if $t \sqsubseteq^{\mathrm{int}} u$ then $\forall C$, $C\langle t\rangle \sqsubseteq^{\mathrm{int}} C\langle u\rangle$;
   $\rightarrow$ Straightforward, as $\mathfrak{C}\langle C^\bullet\rangle$ is a colored context.
3. *Invariance*: if $t =_\beta u$ then $t \sqsubseteq^{\mathrm{int}} u$.
   $\rightarrow$ As in the plain case, it requires some work: rewriting theorems between head and beta and specialized to the checkers calculus.
   For the interaction part, note that $t^\bullet =_{\beta\tau} u^\bullet$.

Interaction Equivalence is an Equational Theory

1. *Equivalence Relation*;
2. *Context-Closed*: if $t \sqsubseteq^{\mathrm{int}} u$ then $\forall C$, $C\langle t\rangle \sqsubseteq^{\mathrm{int}} C\langle u\rangle$;
   $\rightarrow$ Straightforward, as $\mathfrak{C}\langle C^\bullet\rangle$ is a colored context.
3. *Invariance*: if $t =_\beta u$ then $t \sqsubseteq^{\mathrm{int}} u$.
   $\rightarrow$ As in the plain case, it requires some work: rewriting theorems between head and beta and specialized to the checkers calculus.
   For the interaction part, note that $t^\bullet =_{\beta\tau} u^\bullet$.

But... What terms are interaction (in)equivalent?

Interaction equivalence is not stable by η-equivalence!
η-equivalence: $\lambda x.tx =_\eta t$ if $x \notin \mathtt{fv}(t)$

$$\mathtt{I} := \lambda x.x \not\equiv^{\mathrm{int}} \lambda x.\lambda y.xy =: 1$$

- The **separating** context $\mathfrak{C} := \langle\cdot\rangle \circ z \circ w$ distinguishes 1 and $\mathtt{I}$:

$$\begin{tikzcd}
 & (\lambda_\bullet y.z \bullet y) \circ w \arrow[r, "\mathrm{h}\LEFTcircle"] & z \bullet w \arrow[d, phantom, "\neq"] \\
1_\bullet \circ z \circ w \arrow[ur, bend left, "\mathrm{h}\LEFTcircle"] \arrow[dr, bend right, "\bullet\eta"'] & & \\
 & \mathtt{I}_\bullet \circ z \circ w \arrow[r, "\mathrm{h}\LEFTcircle"] & z \circ w
\end{tikzcd}$$

The interaction preorder strictly refines the contextual preorder.

$$t =_\beta u \implies t \sqsubseteq^{\mathrm{int}} u \underset{\not\Leftarrow}{\implies} t \sqsubseteq^{\mathrm{ctx}} u$$

But... What terms are interaction (in)equivalent?

Interaction equivalence is not stable by η-equivalence!
η-equivalence: $\lambda x.tx =_\eta t$ if $x \notin \mathtt{fv}(t)$

$$\mathtt{I} := \lambda x.x \not\equiv^{\mathrm{int}} \lambda x.\lambda y.xy =: \mathtt{1}$$

- The **separating** context $\mathfrak{C} := \langle\cdot\rangle \circ z \circ w$ distinguishes $\mathtt{1}$ and $\mathtt{I}$:

$$\begin{tikzcd}
 & (\lambda_\bullet y.z \bullet y) \circ w \arrow[r, "\mathbf{h}\LEFTcircle"] & z \bullet w \\
\mathtt{1}_\bullet \circ z \circ w \arrow[ur, bend left, "\mathbf{h}\LEFTcircle"] \arrow[dr, leftrightarrow, bend right, "\bullet\eta"] & & \neq \\
 & \mathtt{I}_\bullet \circ z \circ w \arrow[r, "\mathbf{h}\LEFTcircle"] & z \circ w
\end{tikzcd}$$

The interaction preorder strictly refines the contextual preorder.

$$t =_\beta u \implies t \sqsubseteq^{\mathrm{int}} u \underset{\not\Longleftarrow}{\implies} t \sqsubseteq^{\mathrm{ctx}} u$$

But... What terms are interaction (in)equivalent?

Interaction equivalence is not stable by η-equivalence!
η-equivalence: $\lambda x.tx =_\eta t$ if $x \notin \mathtt{fv}(t)$

$$\mathtt{I} := \lambda x.x \not\equiv^{\mathrm{int}} \lambda x.\lambda y.xy =: \mathtt{1}$$

- The **separating** context $\mathfrak{C} := \langle\cdot\rangle \circ z \circ w$ distinguishes $\mathtt{1}$ and $\mathtt{I}$:

$$\begin{tikzcd}
 & (\lambda_\bullet y.z \bullet y) \circ w \arrow[r, "\mathbf{h}\LEFTcircle"] & z \bullet w \arrow[d, phantom, "\neq"] \\
\mathtt{1}_\bullet \circ z \circ w \arrow[ur, bend left, "\mathbf{h}\LEFTcircle"] \arrow[dr, leftrightarrow, bend right, "\bullet\eta"] & & \\
 & \mathtt{I}_\bullet \circ z \circ w \arrow[r, "\mathbf{h}\LEFTcircle"] & z \circ w
\end{tikzcd}$$

The interaction preorder strictly refines the contextual preorder.

$$t =_\beta u \implies t \sqsubseteq^{\mathrm{int}} u \underset{\not\Longleftarrow}{\Longrightarrow} t \sqsubseteq^{\mathrm{ctx}} u$$

Characterizing the Interaction Preorder

[POPL25] The interaction preorder is characterized by the preorder induced by Böhm trees $\sqsubseteq_{\mathcal{B}}$.

To prove that $\sqsubseteq_{\mathcal{B}} = \sqsubseteq^{\text{int}}$, we develop interaction-based refinements of standard techniques for the λ-calculus:

$$\begin{tikzcd}
t \sqsubseteq_{\mathcal{B}} u \arrow[rr, bend left, "\text{Black and White Intersection types}"] & & t \sqsubseteq^{\text{int}} u \arrow[ll, bend left, "\text{Interaction Böhm-out}"]
\end{tikzcd}$$

Optimize the number of interactions

- **Interaction Preorder:**
  $t \sqsubseteq^{\mathrm{int}} u$ [u simulates t] if,
  $$\forall \text{ colored } \mathfrak{C}, \forall k, \quad [\ \mathfrak{C}\langle t^\bullet\rangle \Downarrow_{\mathtt{h}}^{\bullet k} \implies \mathfrak{C}\langle u^\bullet\rangle \Downarrow_{\mathtt{h}}^{\bullet k}\ ];$$
- **Interaction Improvement:**
  $t \sqsubseteq^{\mathrm{int\cdot imp}} u$ [u improves t] if,
  $$\forall \text{ colored } \mathfrak{C}, \forall k, \quad [\ \mathfrak{C}\langle t^\bullet\rangle \Downarrow_{\mathtt{h}}^{\bullet k} \implies \mathfrak{C}\langle u^\bullet\rangle \Downarrow_{\mathtt{h}}^{\bullet k'} \text{ with } k' \leq k\ ];$$
- It does not change the induced equivalence relation.

η-reduction may improve terms, but η-expanding them can never lead to improvement:

$$\lambda y.xy \sqsubseteq^{\mathrm{int\cdot imp}} x \qquad \text{but} \qquad x \not\sqsubseteq^{\mathrm{int\cdot imp}} \lambda y.xy$$

[FoSSaCS26] characterization of the interaction improvement:

$$\begin{array}{ccccc} \sqsubseteq^{\mathrm{int}} & \subsetneq & \sqsubseteq^{\mathrm{int\cdot imp}} & \subsetneq & \sqsubseteq^{\mathrm{ctx}} \\ & & \| & & \\ \sqsubseteq_{\mathcal{B}} & \subsetneq & \sqsubseteq_{\mathcal{B}+\eta red} & \subsetneq & \sqsubseteq_{\mathcal{B}\eta^\infty} \end{array}$$

Optimize the number of interactions

- **Interaction Preorder:**
  $t \sqsubseteq^{\mathrm{int}} u$ [u simulates t] if,
  $\forall$ colored $\mathfrak{C}$, $\forall k$, $[\ \mathfrak{C}\langle t^\bullet\rangle \Downarrow_{\mathtt{h}}^{\bullet k} \implies \mathfrak{C}\langle u^\bullet\rangle \Downarrow_{\mathtt{h}}^{\bullet k}\ ]$;
- **Interaction Improvement:**
  $t \sqsubseteq^{\mathrm{int\cdot imp}} u$ [u improves t] if,
  $\forall$ colored $\mathfrak{C}$, $\forall k$, $[\ \mathfrak{C}\langle t^\bullet\rangle \Downarrow_{\mathtt{h}}^{\bullet k} \implies \mathfrak{C}\langle u^\bullet\rangle \Downarrow_{\mathtt{h}}^{\bullet k'}$ with $k' \leq k\,]$;
- It does not change the induced equivalence relation.

η-reduction may improve terms, but η-expanding them can never lead to improvement:

$$\lambda y.xy \sqsubseteq^{\mathrm{int\cdot imp}} x \quad \text{but} \quad x \not\sqsubseteq^{\mathrm{int\cdot imp}} \lambda y.xy$$

[FoSSaCS26] characterization of the interaction improvement:

$$\begin{array}{ccccc} \sqsubseteq^{\mathrm{int}} & \subsetneq & \sqsubseteq^{\mathrm{int\cdot imp}} & \subsetneq & \sqsubseteq^{\mathrm{ctx}} \\ & & \| & & \\ \sqsubseteq_{\mathcal{B}} & \subsetneq & \sqsubseteq_{\mathcal{B}+\eta red} & \subsetneq & \sqsubseteq_{\mathcal{B}\eta^\infty} \end{array}$$

Conclusion

- **Checkers Calculus**: a calculus to represent interaction between programs
- **Interaction Equivalence**: a cost-sensitive equational theory & the first contextual characterization of Böhm tree equivalence without non determinism in evaluation contexts! (and simple)
- **Interaction improvement**: towards optimizations (& make explicit the costs at play in relational semantics)

Future work:

- Work it out in Call-by-Value/Need, and in effectful extensions

  $\lambda f.f\,() \equiv_{\mathrm{ctx}} \lambda f.f\,()\,;f\,()$
- Prove compiler transformations *optimizing*
- How does it relate to Game Semantics? to process calculi?
- What does our interaction cost represent?

  Communication complexity?

Thank you!

Conclusion

- **Checkers Calculus**: a calculus to represent interaction between programs
- **Interaction Equivalence**: a cost-sensitive equational theory & the first contextual characterization of Böhm tree equivalence without non determinism in evaluation contexts! (and simple)
- **Interaction improvement**: towards optimizations (& make explicit the costs at play in relational semantics)

Future work:

- Work it out in Call-by-Value/Need, and in effectful extensions

  $\lambda f.f\,() \equiv_{\text{ctx}} \lambda f.f\,()\,;f\,()$
- Prove compiler transformations *optimizing*
- How does it relate to Game Semantics? to process calculi?
- What does our interaction cost represent?

  Communication complexity?

Thank you!

Conclusion

- **Checkers Calculus**: a calculus to represent interaction between programs
- **Interaction Equivalence**: a cost-sensitive equational theory & the first contextual characterization of Böhm tree equivalence without non determinism in evaluation contexts! (and simple)
- **Interaction improvement**: towards optimizations (& make explicit the costs at play in relational semantics)

Future work:

- Work it out in Call-by-Value/Need, and in effectful extensions
  $\lambda f.f\,() \equiv_{\mathrm{ctx}} \lambda f.f\,()\,;f\,()$
- Prove compiler transformations *optimizing*
- How does it relate to Game Semantics? to process calculi?
- What does our interaction cost represent?
  Communication complexity?

Thank you!

James Hiram Morris.
Lambda-calculus Models of Programming Languages.
PhD thesis, Massachusetts Institute of Technology, 1968.

David Sands.
Improvement theory and its applications, page 275–306.
Cambridge University Press, USA, 1999.