Destination calculus

A linear λ–calculus for pure, functional memory updates

ARNAUD SPIWACK, Modus Create, France
THOMAS BAGREL, LORIA/Inria, France and Modus Create, France

We present the destination calculus, a linear λ–calculus for pure, functional memory updates. We introduce the syntax, type system, and operational semantics of the destination calculus, and prove type safety formally in the Coq proof assistant.

We show how the principles of the destination calculus can form a theoretical ground for destination-passing style programming in functional languages. In particular, we detail how the present work can be applied to Linear Haskell to lift the main restriction of DPS programming in Haskell as developed in [1]. We illustrate this with a range of pseudo-Haskell examples.

ACM Reference Format:
Arnaud Spiwack and Thomas Bagrel. 2024. Destination calculus: A linear λ–calculus for pure, functional memory updates. In . ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 TERM AND VALUE SYNTAX

var, x, y	Term-level variable name
k	Index for ranges

POPL'25, January 19 – 25, 2025, Denver, Colorado

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

$\mathtt{hvar},\ \mathtt{h}$	$::=$			Hole or destination name ($\mathbb{N}$)
	$\mid$	$\mathtt{h}{+}\mathtt{h}'$	M	
	$\mid$	$\mathtt{h}[\mathtt{H}\underline{\pm}\mathtt{h}']$	M	Shift by $\mathtt{h}'$ if $\mathtt{h} \in \mathtt{H}$
	$\mid$	$\mathit{max}(\mathtt{H})$	M	Maximum of a set of holes
$\mathtt{hvars},\ \mathtt{H}$	$::=$			Set of hole names
	$\mid$	$\{\mathtt{h}_1, \ldots, \mathtt{h}_k\}$		
	$\mid$	$\mathtt{H}_1 \cup \mathtt{H}_2$	M	Union of sets
	$\mid$	$\mathtt{H}\underline{\pm}\mathtt{h}'$	M	Shift all names from $\mathtt{H}$ by $\mathtt{h}'$.
	$\mid$	$\mathit{hvars}(\Gamma)$	M	Hole names of a context (requires $\mathtt{ctx}$
	$\mid$	$\mathit{hvars}(C)$	M	Hole names of an evaluation context
$\mathit{term},\ t,\ u$	$::=$			Term
	$\mid$	v		Value
	$\mid$	$\mathtt{x}$		Variable
	$\mid$	$t \triangleright t'$		Application
	$\mid$	$t\ ;u$		Pattern-match on unit
	$\mid$	$t \triangleright \mathtt{case}_{\mathtt{m}}\,\{\mathtt{Inl}\ \mathtt{x}_1 \mapsto u_1\,,\ \mathtt{Inr}\ \mathtt{x}_2 \mapsto u_2\}$		Pattern-match on sum
	$\mid$	$t \triangleright \mathtt{case}_{\mathtt{m}}\,(\mathtt{x}_1\,,\,\mathtt{x}_2) \mapsto u$		Pattern-match on product
	$\mid$	$t \triangleright \mathtt{case}_{\mathtt{m}}\ \mathtt{E}_{\mathtt{n}}\ \mathtt{x} \mapsto u$		Pattern-match on exponential
	$\mid$	$t \triangleright \mathtt{map}\ \mathtt{x} \mapsto t'$		Map over the right side of ampar t
	$\mid$	$\mathtt{to}_{\ltimes}\ u$		Wrap u into a trivial ampar
	$\mid$	$\mathtt{from}_{\ltimes}\ t$		Convert ampar with no dest remaining
	$\mid$	$t \triangleleft ()$		Fill destination with unit
	$\mid$	$t \triangleleft \mathtt{Inl}$		Fill destination with left variant
	$\mid$	$t \triangleleft \mathtt{Inr}$		Fill destination with right variant
	$\mid$	$t \triangleleft \mathtt{E}_{\mathtt{m}}$		Fill destination with exponential const

	$\mid$	$t \triangleleft (,)$		Fill destination with product constructor
	$\mid$	$t \triangleleft (\lambda \mathtt{x}\ {}_{\mathtt{m}}\!\mapsto\ u)$		Fill destination with function
	$\mid$	$t \triangleleft\!\bullet\ t'$		Fill destination with root of ampar t'
	$\mid$	$t[\mathtt{x} \coloneqq v]$	M	
	$\mid$	$\mathtt{alloc}$	M	
	$\mid$	$\mathtt{from}'_{\ltimes}$	M	
	$\mid$	$t \triangleleft t'$	M	
	$\mid$	${}^{\mathrm{s}}\lambda \mathtt{x}\ {}_{\mathtt{m}}\!\mapsto\ u$	M	Lambda abstraction
	$\mid$	${}^{s}\mathtt{Inl}\,t$	M	Left variant for sum
	$\mid$	${}^{s}\mathtt{Inr}\,t$	M	Right variant for sum
	$\mid$	${}^{s}\mathtt{E}_{\mathtt{m}}\ t$	M	Exponential
	$\mid$	${}^{s}(t_1\ ,\ t_2)$	M	Product

val, v	$::=$			Term value
	$\mid$	$-\mathtt{h}$		Hole
	$\mid$	$+\mathtt{h}$		Destination
	$\mid$	$()$		Unit
	$\mid$	${}^{\mathrm{V}}\lambda \mathtt{x}\ {}_{\mathtt{m}}\!\mapsto\ u$		Lambda abstraction
	$\mid$	$\mathtt{Inl}\ v$		Left variant for sum
	$\mid$	$\mathtt{Inr}\ v$		Right variant for sum
	$\mid$	$\mathtt{E}_{\mathtt{m}}\ v$		Exponential
	$\mid$	$(v_1\ ,\ v_2)$		Product
	$\mid$	${}_{\mathtt{H}}\langle v_2\ ⨾\ v_1\rangle$		Ampar
	$\mid$	$v[\mathtt{H}\underline{\pm}\mathtt{h}']$	M	Shift hole names inside v by $\mathtt{h}'$ if they belong to $\mathtt{H}$.

ectx, c	$::=$		Evaluation context component
	$\mid$	$\square \triangleright t'$	Application
	$\mid$	$v \triangleright \square$	Application
	$\mid$	$\square\ ;\ u$	Pattern-match on unit
	$\mid$	$\square \triangleright \mathtt{case}_{\mathtt{m}}\ \{\mathtt{Inl}\ \mathtt{x}_1 \mapsto u_1\ ,\ \mathtt{Inr}\ \mathtt{x}_2 \mapsto u_2\}$	Pattern-match on sum
	$\mid$	$\square \triangleright \mathtt{case}_{\mathtt{m}}\ (\mathtt{x}_1\ ,\ \mathtt{x}_2) \mapsto u$	Pattern-match on product
	$\mid$	$\square \triangleright \mathtt{case}_{\mathtt{m}}\ \mathtt{E}_{\mathtt{n}}\ \mathtt{x} \mapsto u$	Pattern-match on exponential
	$\mid$	$\square \triangleright \mathtt{map}\ \mathtt{x} \mapsto t'$	Map over the right side of ampar
	$\mid$	$\mathtt{to}_{\ltimes}\ \square$	Wrap into a trivial ampar
	$\mid$	$\mathtt{from}_{\ltimes}\ \square$	Convert ampar with no dest remaining int
	$\mid$	$\square \triangleleft ()$	Fill destination with unit
	$\mid$	$\square \triangleleft\ \mathtt{Inl}$	Fill destination with left variant
	$\mid$	$\square \triangleleft\ \mathtt{Inr}$	Fill destination with right variant
	$\mid$	$\square \triangleleft\ \mathtt{E}_{\mathtt{m}}$	Fill destination with exponential construct
	$\mid$	$\square \triangleleft (,)$	Fill destination with product constructor
	$\mid$	$\square \triangleleft (\lambda \mathtt{x}\ {}_{\mathtt{m}}\!\mapsto\ u)$	Fill destination with function
	$\mid$	$\square \triangleleft\!\bullet\ t'$	Fill destination with root of ampar
	$\mid$	$v \triangleleft\!\bullet \square$	Fill destination with root of ampar
	$\mid$	${}^{\mathrm{op}}_{\mathtt{H}}\langle v_2\ ⨾\ \square$	Open ampar. Only new addition to term sh

ectxs, C	::=			Evaluation context stack
	$\mid$	□		Represent the empty stack / "identity" evaluation context
	$\mid$	$C \circ c$		Push c on top of C
	$\mid$	$C[\mathsf{h} :=_{\mathsf{H}} v]$	M	Fill h in C with value v (that may contain holes)

2 TYPE SYSTEM

`type`, T, U	::=			Type
	$\mid$	$\mathsf{1}$		Unit
	$\mid$	$\mathsf{T}_1 \oplus \mathsf{T}_2$		Sum
	$\mid$	$\mathsf{T}_1 \otimes \mathsf{T}_2$		Product
	$\mid$	$!_{\mathsf{m}}\,\mathsf{T}$		Exponential
	$\mid$	$\mathsf{U} \ltimes \mathsf{T}$		Ampar type (consuming T yields U)
	$\mid$	$\mathsf{T}\ {}_{\mathsf{m}}\!\!\rightarrow \mathsf{U}$		Function
	$\mid$	$\lfloor \mathsf{T} \rfloor^{\mathsf{m}}$		Destination

`mode`, m, n	::=			Mode (Semiring)
	$\mid$	pa		Pair of a multiplicity and age
	$\mid$	☠		Error case (incompatible types, multiplicities, or ages)
	$\mid$	$\mathsf{m}_1 \cdot \ldots \cdot \mathsf{m}_{\mathsf{k}}$	M	Semiring product

`mul`, p	::=			Multiplicity (first component of modality)
	$\mid$	$\mathsf{1}$		Linear. Neutral element of the product
	$\mid$	ω		Non-linear. Absorbing for the product
	$\mid$	$\mathsf{p}_1 . \ldots . \mathsf{p}_{\mathsf{k}}$	M	Semiring product

`age`, a	::=			Age (second component of modality)
	$\mid$	ν		Born now. Neutral element of the product
	$\mid$	$\uparrow$		One scope older
	$\mid$	∞		Infinitely old / static. Absorbing for the product
	$\mid$	$\mathsf{a}_1 \cdot \ldots \cdot \mathsf{a}_{\mathsf{k}}$	M	Semiring product

ctx, Γ, Δ, Π	::=			Typing context
	$\mid$	$\mathsf{x} :_{\mathsf{m}} \mathsf{T}$		
	$\mid$	$+\mathsf{h} :_{\mathsf{m}} \lfloor \mathsf{T} \rfloor^{\mathsf{n}}$		
	$\mid$	$-\mathsf{h} : \mathsf{T}^{\mathsf{n}}$		
	$\mid$	$\mathsf{m} \cdot \Gamma$	M	Multiply each binding by m
	$\mid$	$\Gamma_1 + \Gamma_2$	M	Sum contexts Γ_1 and Γ_2. Duplicate keys with incompatible values
	$\mid$	$\Gamma_1,\ \Gamma_2$	M	Disjoint sum/union of contexts Γ_1 and Γ_2.
	$\mid$	$-\Gamma$	M	Transforms dest bindings into a hole bindings (requires `ctx_Dest`
	$\mid$	$-^{-1}\Gamma$	M	Transforms hole bindings into dest bindings with left mode $\mathsf{1}\nu$ (re
	$\mid$	$\Gamma[\mathsf{H} \pm \mathsf{h}']$	M	Shift hole/dest names by h' if they belong to H

$\boxed{\Gamma \Vdash v : T}$ *(Typing of values (raw))*

TyR-val-H
$$\dfrac{}{-h : T^{1\nu} \Vdash -h : T}$$

TyR-val-D
$$\dfrac{}{+h :_{1\nu} \lfloor T \rfloor^{n} \Vdash +h : \lfloor T \rfloor^{n}}$$

TyR-val-U
$$\dfrac{}{\Vdash () : 1}$$

TyR-val-F
$$\dfrac{\Delta + x :_{m} T \vdash u : U}{\Delta \Vdash {}^{v}\lambda x\ {}_{m}\!\mapsto\ u : T\ {}_{m}\!\!\to U}$$

TyR-val-L
$$\dfrac{\Gamma \Vdash v_1 : T_1}{\Gamma \Vdash \texttt{Inl}\ v_1 : T_1 \oplus T_2}$$

TyR-val-R
$$\dfrac{\Gamma \Vdash v_2 : T_2}{\Gamma \Vdash \texttt{Inr}\ v_2 : T_1 \oplus T_2}$$

TyR-val-P
$$\dfrac{\begin{array}{c}\Gamma_1 \Vdash v_1 : T_1\\ \Gamma_2 \Vdash v_2 : T_2\end{array}}{\Gamma_1 + \Gamma_2 \Vdash (v_1 , v_2) : T_1 \otimes T_2}$$

TyR-val-E
$$\dfrac{\Gamma \Vdash v' : T}{n\cdot\Gamma \Vdash \mathsf{E}_{n}\ v' : !_{n}\, T}$$

TyR-val-A
$$\dfrac{\begin{array}{c}\texttt{LinOnly}\ \Delta_3\\ \texttt{FinAgeOnly}\ \Delta_3\\ 1\uparrow\cdot\Delta_1,\ \Delta_3 \Vdash v_1 : T\\ \Delta_2,\ (-\Delta_3) \Vdash v_2 : U\end{array}}{\Delta_1,\ \Delta_2 \Vdash {}_{hvars(-\Delta_3)}\langle v_2 \,,\, v_1 \rangle : U \ltimes T}$$

$\boxed{\Pi \vdash t : T}$ *(Typing of terms)*

Ty-term-Val
$$\dfrac{\begin{array}{c}\texttt{DisposableOnly}\ \Pi\\ \Delta \Vdash v : T\end{array}}{\Pi,\ \Delta \vdash v : T}$$

Ty-term-Var
$$\dfrac{\begin{array}{c}\texttt{DisposableOnly}\ \Pi\\ 1\nu \ <:\ m\end{array}}{\Pi,\ x :_{m} T \vdash x : T}$$

Ty-term-App
$$\dfrac{\begin{array}{c}\Pi_1 \vdash t : T\\ \Pi_2 \vdash t' : T\ {}_{m}\!\!\to U\end{array}}{m\cdot\Pi_1 + \Pi_2 \vdash t \triangleright t' : U}$$

Ty-term-PatU
$$\dfrac{\Pi_1 \vdash t : 1 \qquad \Pi_2 \vdash u : U}{\Pi_1 + \Pi_2 \vdash t \,;\, u : U}$$

Ty-term-PatS
$$\dfrac{\begin{array}{c}\Pi_1 \vdash t : T_1 \oplus T_2\\ \Pi_2,\ x_1 :_{m} T_1 \vdash u_1 : U\\ \Pi_2,\ x_2 :_{m} T_2 \vdash u_2 : U\end{array}}{m\cdot\Pi_1 + \Pi_2 \vdash t \triangleright \mathsf{case}_{m}\, \{\texttt{Inl}\ x_1 \mapsto u_1\, ,\ \texttt{Inr}\ x_2 \mapsto u_2\} : U}$$

Ty-term-PatP
$$\dfrac{\begin{array}{c}\Pi_1 \vdash t : T_1 \otimes T_2\\ \Pi_2,\ x_1 :_{m} T_1,\ x_2 :_{m} T_2 \vdash u : U\end{array}}{m\cdot\Pi_1 + \Pi_2 \vdash t \triangleright \mathsf{case}_{m}\, (x_1 , x_2) \mapsto u : U}$$

Ty-term-PatE
$$\dfrac{\begin{array}{c}\Pi_1 \vdash t : !_{n}\, T\\ \Pi_2,\ x :_{m\cdot n} T \vdash u : U\end{array}}{m\cdot\Pi_1 + \Pi_2 \vdash t \triangleright \mathsf{case}_{m}\, \mathsf{E}_{n}\, x \mapsto u : U}$$

Ty-term-Map
$$\dfrac{\begin{array}{c}\Pi_1 \vdash t : U \ltimes T\\ 1\uparrow\cdot\Pi_2,\ x :_{1\nu} T \vdash t' : T'\end{array}}{\Pi_1 + \Pi_2 \vdash t \triangleright \mathsf{map}\ x \mapsto t' : U \ltimes T'}$$

Ty-term-ToA
$$\dfrac{\Pi \vdash u : U}{\Pi \vdash \texttt{to}_{\ltimes}\ u : U \ltimes 1}$$

Ty-term-FromA
$$\dfrac{\Pi \vdash t : U \ltimes (!_{1\infty}\, T)}{\Pi \vdash \texttt{from}_{\ltimes}\ t : U \otimes (!_{1\infty}\, T)}$$

Ty-term-FillU
$$\dfrac{\Pi \vdash t : \lfloor 1 \rfloor^{n}}{\Pi \vdash t \triangleleft () : 1}$$

Ty-term-FillL
$$\dfrac{\Pi \vdash t : \lfloor T_1 \oplus T_2 \rfloor^{n}}{\Pi \vdash t \triangleleft \texttt{Inl} : \lfloor T_1 \rfloor^{n}}$$

Ty-term-FillR
$$\dfrac{\Pi \vdash t : \lfloor T_1 \oplus T_2 \rfloor^{n}}{\Pi \vdash t \triangleleft \texttt{Inr} : \lfloor T_2 \rfloor^{n}}$$

Ty-term-FillP
$$\dfrac{\Pi \vdash t : \lfloor T_1 \otimes T_2 \rfloor^{n}}{\Pi \vdash t \triangleleft (,) : \lfloor T_1 \rfloor^{n} \otimes \lfloor T_2 \rfloor^{n}}$$

Ty-term-FillE
$$\dfrac{\Pi \vdash t : \lfloor !_{n'}\, T \rfloor^{n}}{\Pi \vdash t \triangleleft \mathsf{E}_{n'} : \lfloor T \rfloor^{n'\cdot n}}$$

Ty-term-FillF
$$\dfrac{\begin{array}{c}\Pi_1 \vdash t : \lfloor T\ {}_{m}\!\!\to U \rfloor^{n}\\ \Pi_2,\ x :_{m} T \vdash u : U\end{array}}{\Pi_1 + (1\uparrow\cdot n)\cdot\Pi_2 \vdash t \triangleleft (\lambda x\ {}_{m}\!\mapsto\ u) : 1}$$

Ty-term-FillC
$$\dfrac{\begin{array}{c}\Pi_1 \vdash t : \lfloor U \rfloor^{n}\\ \Pi_2 \vdash t' : U \ltimes T\end{array}}{\Pi_1 + (1\uparrow\cdot n)\cdot\Pi_2 \vdash t \triangleleft\!\bullet\, t' : T}$$

$\boxed{\Delta \dashv C : \mathsf{T}\rightarrowtail\mathsf{U}_0}$ *(Typing of evaluation contexts)*

Ty-ectxs-Id

$$\frac{}{\dashv \square : \mathsf{U}_0\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-AppFoc1

$$\frac{\mathsf{m}\cdot\Delta_1,\ \Delta_2 \dashv C : \mathsf{U}\rightarrowtail\mathsf{U}_0 \qquad \Delta_2 \vdash t' : \mathsf{T}\ {}_{\mathsf{m}}\!\!\rightarrow \mathsf{U}}{\Delta_1 \dashv C \circ (\square \triangleright t') : \mathsf{T}\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-AppFoc2

$$\frac{\mathsf{m}\cdot\Delta_1,\ \Delta_2 \dashv C : \mathsf{U}\rightarrowtail\mathsf{U}_0 \qquad \Delta_1 \vdash v : \mathsf{T}}{\Delta_2 \dashv C \circ (v \triangleright \square) : (\mathsf{T}\ {}_{\mathsf{m}}\!\!\rightarrow \mathsf{U})\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-PatUFoc

$$\frac{\Delta_1,\ \Delta_2 \dashv C : \mathsf{U}\rightarrowtail\mathsf{U}_0 \qquad \Delta_2 \vdash u : \mathsf{U}}{\Delta_1 \dashv C \circ (\square \mathbin{;} u) : 1\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-PatSFoc

$$\frac{\mathsf{m}\cdot\Delta_1,\ \Delta_2 \dashv C : \mathsf{U}\rightarrowtail\mathsf{U}_0 \qquad \Delta_2,\ x_1 :_{\mathsf{m}} \mathsf{T}_1 \vdash u_1 : \mathsf{U} \qquad \Delta_2,\ x_2 :_{\mathsf{m}} \mathsf{T}_2 \vdash u_2 : \mathsf{U}}{\Delta_1 \dashv C \circ (\square \triangleright \mathtt{case}_{\mathsf{m}}\ \{\mathtt{Inl}\ x_1 \mapsto u_1\ ,\ \mathtt{Inr}\ x_2 \mapsto u_2\}) : (\mathsf{T}_1\oplus\mathsf{T}_2)\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-PatPFoc

$$\frac{\mathsf{m}\cdot\Delta_1,\ \Delta_2 \dashv C : \mathsf{U}\rightarrowtail\mathsf{U}_0 \qquad \Delta_2,\ x_1 :_{\mathsf{m}} \mathsf{T}_1,\ x_2 :_{\mathsf{m}} \mathsf{T}_2 \vdash u : \mathsf{U}}{\Delta_1 \dashv C \circ (\square \triangleright \mathtt{case}_{\mathsf{m}}\ (x_1\ ,\ x_2) \mapsto u) : (\mathsf{T}_1\otimes\mathsf{T}_2)\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-PatEFoc

$$\frac{\mathsf{m}\cdot\Delta_1,\ \Delta_2 \dashv C : \mathsf{U}\rightarrowtail\mathsf{U}_0 \qquad \Delta_2,\ x :_{\mathsf{m}\cdot\mathsf{m}'} \mathsf{T} \vdash u : \mathsf{U}}{\Delta_1 \dashv C \circ (\square \triangleright \mathtt{case}_{\mathsf{m}}\ \mathsf{E}_{\mathsf{m}'}\ x \mapsto u) : !_{\mathsf{m}'}\ \mathsf{T}\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-MapFoc

$$\frac{\Delta_1,\ \Delta_2 \dashv C : \mathsf{U} \ltimes \mathsf{T}'\rightarrowtail\mathsf{U}_0 \qquad 1\!\uparrow\!\cdot\Delta_2,\ x :_{1\nu} \mathsf{T} \vdash t' : \mathsf{T}'}{\Delta_1 \dashv C \circ (\square \triangleright \mathtt{map}\ x \mapsto t') : (\mathsf{U} \ltimes \mathsf{T})\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-ToAFoc

$$\frac{\Delta \dashv C : (\mathsf{U} \ltimes 1)\rightarrowtail\mathsf{U}_0}{\Delta \dashv C \circ (\mathtt{to}_{\ltimes}\ \square) : \mathsf{U}\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-FromAFoc

$$\frac{\Delta \dashv C : (\mathsf{U}\otimes(!_{1\infty}\ \mathsf{T}))\rightarrowtail\mathsf{U}_0}{\Delta \dashv C \circ (\mathtt{from}_{\ltimes}\ \square) : (\mathsf{U} \ltimes (!_{1\infty}\ \mathsf{T}))\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-FillUFoc

$$\frac{\Delta \dashv C : 1\rightarrowtail\mathsf{U}_0}{\Delta \dashv C \circ (\square \triangleleft ()) : \lfloor 1 \rfloor^{\mathsf{n}}\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-FillLFoc

$$\frac{\Delta \dashv C : \lfloor \mathsf{T}_1 \rfloor^{\mathsf{n}}\rightarrowtail\mathsf{U}_0}{\Delta \dashv C \circ (\square \triangleleft \mathtt{Inl}) : \lfloor \mathsf{T}_1\oplus\mathsf{T}_2 \rfloor^{\mathsf{n}}\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-FillRFoc

$$\frac{\Delta \dashv C : \lfloor \mathsf{T}_2 \rfloor^{\mathsf{n}}\rightarrowtail\mathsf{U}_0}{\Delta \dashv C \circ (\square \triangleleft \mathtt{Inr}) : \lfloor \mathsf{T}_1\oplus\mathsf{T}_2 \rfloor^{\mathsf{n}}\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-FillPFoc

$$\frac{\Delta \dashv C : (\lfloor \mathsf{T}_1 \rfloor^{\mathsf{n}}\otimes\lfloor \mathsf{T}_2 \rfloor^{\mathsf{n}})\rightarrowtail\mathsf{U}_0}{\Delta \dashv C \circ (\square \triangleleft (,)) : \lfloor \mathsf{T}_1\otimes\mathsf{T}_2 \rfloor^{\mathsf{n}}\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-FillEFoc

$$\frac{\Delta \dashv C : \lfloor \mathsf{T} \rfloor^{\mathsf{m}\cdot\mathsf{n}}\rightarrowtail\mathsf{U}_0}{\Delta \dashv C \circ (\square \triangleleft \mathsf{E}_{\mathsf{m}}) : \lfloor !_{\mathsf{m}}\ \mathsf{T} \rfloor^{\mathsf{n}}\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-FillFFoc

$$\frac{\Delta_1,\ (1\!\uparrow\!\cdot\mathsf{n})\cdot\Delta_2 \dashv C : 1\rightarrowtail\mathsf{U}_0 \qquad \Delta_2,\ x :_{\mathsf{m}} \mathsf{T} \vdash u : \mathsf{U}}{\Delta_1 \dashv C \circ (\square \triangleleft (\lambda x\ {}_{\mathsf{m}}\!\!\mapsto u)) : \lfloor \mathsf{T}\ {}_{\mathsf{m}}\!\!\rightarrow \mathsf{U} \rfloor^{\mathsf{n}}\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-FillCFoc1

$$\frac{\Delta_1,\ (1\!\uparrow\!\cdot\mathsf{n})\cdot\Delta_2 \dashv C : \mathsf{T}\rightarrowtail\mathsf{U}_0 \qquad \Delta_2 \vdash t' : \mathsf{U}\ltimes\mathsf{T}}{\Delta_1 \dashv C \circ (\square \triangleleft\!\bullet\, t') : \lfloor\mathsf{U}\rfloor^{\mathsf{n}}\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-FillCFoc2

$$\frac{\Delta_1,\ (1\!\uparrow\!\cdot\mathsf{n})\cdot\Delta_2 \dashv C : \mathsf{T}\rightarrowtail\mathsf{U}_0 \qquad \Delta_1 \vdash v : \lfloor\mathsf{U}\rfloor^{\mathsf{n}}}{\Delta_2 \dashv C \circ (v \triangleleft\!\bullet\, \square) : \mathsf{U}\ltimes\mathsf{T}\rightarrowtail\mathsf{U}_0}$$

Ty-ectxs-AOpenFoc

$$\frac{\begin{array}{c} \mathit{hvars}(C)\ \#\#\ \mathit{hvars}(-\Delta_3) \\ \texttt{LinOnly}\ \Delta_3 \\ \texttt{FinAgeOnly}\ \Delta_3 \\ \Delta_1,\ \Delta_2 \dashv C : (\mathsf{U}\ltimes\mathsf{T}')\rightarrowtail\mathsf{U}_0 \\ \Delta_2,\ -\Delta_3 \Vdash v_2 : \mathsf{U} \end{array}}{1\!\uparrow\!\cdot\Delta_1,\ \Delta_3 \dashv C \circ ({}^{\mathrm{op}}_{\mathit{hvars}(-\Delta_3)}\langle v_2 \,,\, \square) : \mathsf{T}'\rightarrowtail\mathsf{U}_0}$$

$\boxed{\vdash C[t] : \mathsf{T}}$ *(Typing of extended terms (pair of evaluation context and term))*

Ty-eterm-ClosedEterm

$$\frac{\Delta \dashv C : \mathsf{T}\rightarrowtail\mathsf{U}_0 \qquad \Delta \vdash t : \mathsf{T}}{\vdash C[t] : \mathsf{U}_0}$$

3 SMALL-STEP SEMANTICS

$\boxed{C[t] \longrightarrow C'[t']}$ *(Small-step evaluation of terms using evaluation contexts)*

Sem-eterm-AppFoc1

$$\frac{\mathtt{NotVal}\ t}{C[t \triangleright t'] \longrightarrow (C \circ (\square \triangleright t'))[t]}$$

Sem-eterm-AppUnfoc1

$$\frac{}{(C \circ (\square \triangleright t'))[v] \longrightarrow C[v \triangleright t']}$$

Sem-eterm-AppFoc2

$$\frac{\mathtt{NotVal}\ t'}{C[v \triangleright t'] \longrightarrow (C \circ (v \triangleright \square))[t']}$$

Sem-eterm-AppUnfoc2

$$\frac{}{(C \circ (v \triangleright \square))[v'] \longrightarrow C[v \triangleright v']}$$

Sem-eterm-AppRed

$$\frac{}{C[v \triangleright ({}^{\mathsf{v}}\lambda \mathsf{x}\ {}_{\mathsf{m}}\!\mapsto\ u)] \longrightarrow C[u[\mathsf{x} := v]]}$$

Sem-eterm-PatUFoc

$$\frac{\mathtt{NotVal}\ t}{C[t\,;u] \longrightarrow (C \circ (\square\,;u))[t]}$$

Sem-eterm-PatUUnfoc

$$\frac{}{(C \circ (\square\,;u))[v] \longrightarrow C[v\,;u]}$$

Sem-eterm-PatURed

$$\frac{}{C[()\,;u] \longrightarrow C[u]}$$

Sem-eterm-PatSFoc

$$\frac{\mathtt{NotVal}\ t}{C[t \triangleright \mathsf{case}_{\mathsf{m}}\,\{\mathtt{Inl}\ \mathsf{x}_1 \mapsto u_1\,,\ \mathtt{Inr}\ \mathsf{x}_2 \mapsto u_2\}] \longrightarrow (C \circ (\square \triangleright \mathsf{case}_{\mathsf{m}}\,\{\mathtt{Inl}\ \mathsf{x}_1 \mapsto u_1\,,\ \mathtt{Inr}\ \mathsf{x}_2 \mapsto u_2\}))[t]}$$

Sem-eterm-PatSUnfoc

$$\frac{}{(C \circ (\square \triangleright \mathsf{case}_{\mathsf{m}}\,\{\mathtt{Inl}\ \mathsf{x}_1 \mapsto u_1\,,\ \mathtt{Inr}\ \mathsf{x}_2 \mapsto u_2\}))[v] \longrightarrow C[v \triangleright \mathsf{case}_{\mathsf{m}}\,\{\mathtt{Inl}\ \mathsf{x}_1 \mapsto u_1\,,\ \mathtt{Inr}\ \mathsf{x}_2 \mapsto u_2\}]}$$

Sem-eterm-PatLRed

$$\frac{}{C[(\mathtt{Inl}\ v_1) \triangleright \mathsf{case}_{\mathsf{m}}\,\{\mathtt{Inl}\ \mathsf{x}_1 \mapsto u_1\,,\ \mathtt{Inr}\ \mathsf{x}_2 \mapsto u_2\}] \longrightarrow C[u_1[\mathsf{x}_1 := v_1]]}$$

Sem-eterm-PatRRed

$$\frac{}{C[(\mathtt{Inr}\ v_2) \triangleright \mathsf{case}_{\mathsf{m}}\,\{\mathtt{Inl}\ \mathsf{x}_1 \mapsto u_1\,,\ \mathtt{Inr}\ \mathsf{x}_2 \mapsto u_2\}] \longrightarrow C[u_2[\mathsf{x}_2 := v_2]]}$$

Sem-eterm-PatPFoc

$$\frac{\mathtt{NotVal}\ t}{C[t \triangleright \mathsf{case}_{\mathsf{m}}\,(\mathsf{x}_1\,,\mathsf{x}_2) \mapsto u] \longrightarrow (C \circ (\square \triangleright \mathsf{case}_{\mathsf{m}}\,(\mathsf{x}_1\,,\mathsf{x}_2) \mapsto u))[t]}$$

Sem-eterm-PatPUnfoc

$$\frac{}{(C \circ (\square \triangleright \mathsf{case}_{\mathsf{m}}\,(\mathsf{x}_1\,,\mathsf{x}_2) \mapsto u))[v] \longrightarrow C[v \triangleright \mathsf{case}_{\mathsf{m}}\,(\mathsf{x}_1\,,\mathsf{x}_2) \mapsto u]}$$

Sem-eterm-PatPRed

$$\frac{}{C[(v_1\,,v_2) \triangleright \mathsf{case}_{\mathsf{m}}\,(\mathsf{x}_1\,,\mathsf{x}_2) \mapsto u] \longrightarrow C[u[\mathsf{x}_1 := v_1][\mathsf{x}_2 := v_2]]}$$

Sem-eterm-PatEFoc

$$\frac{\mathtt{NotVal}\ t}{C[t \triangleright \mathsf{case}_{\mathsf{m}}\,\mathsf{E}_{\mathsf{n}}\ \mathsf{x} \mapsto u] \longrightarrow (C \circ (\square \triangleright \mathsf{case}_{\mathsf{m}}\,\mathsf{E}_{\mathsf{n}}\ \mathsf{x} \mapsto u))[t]}$$

Sem-eterm-PatEUnfoc

$$\frac{}{(C \circ (\square \triangleright \mathtt{case}_{\mathtt{m}}\, \mathtt{E}_{\mathtt{n}}\, \mathtt{x} \mapsto u))[v] \longrightarrow C[v \triangleright \mathtt{case}_{\mathtt{m}}\, \mathtt{E}_{\mathtt{n}}\, \mathtt{x} \mapsto u]}$$

Sem-eterm-PatERed

$$\frac{}{C[\mathtt{E}_{\mathtt{n}}\, v' \triangleright \mathtt{case}_{\mathtt{m}}\, \mathtt{E}_{\mathtt{n}}\, \mathtt{x} \mapsto u] \longrightarrow C[u[\mathtt{x} \coloneqq v']]}$$

Sem-eterm-MapFoc

$$\frac{\mathtt{NotVal}\ \ t}{C[t \triangleright \mathtt{map}\ \mathtt{x} \mapsto t'] \longrightarrow (C \circ (\square \triangleright \mathtt{map}\ \mathtt{x} \mapsto t'))[t]}$$

Sem-eterm-MapUnfoc

$$\frac{}{(C \circ (\square \triangleright \mathtt{map}\ \mathtt{x} \mapsto t'))[v] \longrightarrow C[v \triangleright \mathtt{map}\ \mathtt{x} \mapsto t']}$$

Sem-eterm-MapRedAOpenFoc

$$\frac{\mathsf{h}' \ = \ \mathit{max}(\mathit{hvars}(C))\mathsf{+1}}{C[{}_{\mathsf{H}}\langle v_2 \,;\, v_1\rangle \triangleright \mathtt{map}\ \mathtt{x} \mapsto t'] \longrightarrow (C \circ ({}^{\mathrm{op}}_{\mathsf{H}\underline{\pm}\mathsf{h}'}\langle v_2[\mathsf{H}\underline{\pm}\mathsf{h}'] \,;\, \square\rangle))[t'[\mathtt{x} \coloneqq v_1[\mathsf{H}\underline{\pm}\mathsf{h}']]]}$$

Sem-eterm-AOpenUnfoc

$$\frac{}{(C \circ^{\mathrm{op}}_{\mathsf{H}} \langle v_2 \,;\, \square\rangle)[v_1] \longrightarrow C[{}_{\mathsf{H}}\langle v_2 \,;\, v_1\rangle]}$$

Sem-eterm-ToAFoc

$$\frac{\mathtt{NotVal}\ \ u}{C[\mathtt{to}_{\ltimes}\ u] \longrightarrow (C \circ (\mathtt{to}_{\ltimes}\ \square))[u]}$$

Sem-eterm-ToAUnfoc

$$\frac{}{(C \circ (\mathtt{to}_{\ltimes}\ \square))[v_2] \longrightarrow C[\mathtt{to}_{\ltimes}\ v_2]}$$

Sem-eterm-ToARed

$$\frac{}{C[\mathtt{to}_{\ltimes}\ v_2] \longrightarrow C[{}_{\{\,\}}\langle v_2 \,;\, ()\rangle]}$$

Sem-eterm-FromAFoc

$$\frac{\mathtt{NotVal}\ \ t}{C[\mathtt{from}_{\ltimes}\ t] \longrightarrow (C \circ (\mathtt{from}_{\ltimes}\ \square))[t]}$$

Sem-eterm-FromAUnfoc

$$\frac{}{(C \circ (\mathtt{from}_{\ltimes}\ \square))[v] \longrightarrow C[\mathtt{from}_{\ltimes}\ v]}$$

Sem-eterm-FromARed

$$\frac{}{C[\mathtt{from}_{\ltimes}\ {}_{\{\,\}}\langle v_2 \,;\, \mathtt{E}_{1\infty}\ v_1\rangle] \longrightarrow C[(v_2\,,\, \mathtt{E}_{1\infty}\ v_1)]}$$

Sem-eterm-FillUFoc

$$\frac{\mathtt{NotVal}\ \ t}{C[t \triangleleft ()] \longrightarrow (C \circ (\square \triangleleft ()))[t]}$$

Sem-eterm-FillUUnfoc

$$\frac{}{(C \circ (\square \triangleleft ()))[v] \longrightarrow C[v \triangleleft ()]}$$

Sem-eterm-FillURed

$$\frac{}{C[\mathsf{+h} \triangleleft ()] \longrightarrow C[\mathsf{h} \coloneqq_{\{\,\}}\ ()][()]}$$

Sem-eterm-FillLFoc

$$\frac{\mathtt{NotVal}\ \ t}{C[t \triangleleft \mathtt{Inl}] \longrightarrow (C \circ (\square \triangleleft \mathtt{Inl}))[t]}$$

Sem-eterm-FillLUnfoc

$$\frac{}{(C \circ (\square \triangleleft \mathtt{Inl}))[v] \longrightarrow C[v \triangleleft \mathtt{Inl}]}$$

Sem-eterm-FillLRed

$$\frac{\mathsf{h}' \ = \ \mathit{max}(\mathit{hvars}(C) \cup \{\mathsf{h}\})\mathsf{+1}}{C[\mathsf{+h} \triangleleft \mathtt{Inl}] \longrightarrow C[\mathsf{h} \coloneqq_{\{\mathsf{h}'+1\}}\ \mathtt{Inl}\ \mathsf{-(h'+1)}][\mathsf{+(h'+1)}]}$$

Sem-eterm-FillRFoc

$$\frac{\mathtt{NotVal}\ \ t}{C[t \triangleleft \mathtt{Inr}] \longrightarrow (C \circ (\square \triangleleft \mathtt{Inr}))[t]}$$

Sem-eterm-FillRUnfoc

$$\frac{}{(C \circ (\square \triangleleft \mathtt{Inr}))[v] \longrightarrow C[v \triangleleft \mathtt{Inr}]}$$

Sem-eterm-FillRRed

$$\frac{\mathsf{h}' \ = \ \mathit{max}(\mathit{hvars}(C) \cup \{\mathsf{h}\})\mathsf{+1}}{C[\mathsf{+h} \triangleleft \mathtt{Inr}] \longrightarrow C[\mathsf{h} \coloneqq_{\{\mathsf{h}'+1\}}\ \mathtt{Inr}\ \mathsf{-(h'+1)}][\mathsf{+(h'+1)}]}$$

$$\textsc{Sem-eterm-FillEFoc}\quad \frac{\mathtt{NotVal}\ \ t}{C[t \triangleleft \mathsf{E}_{\mathsf{m}}] \longrightarrow (C \circ (\square \triangleleft \mathsf{E}_{\mathsf{m}}))[t]}$$

$$\textsc{Sem-eterm-FillEUnfoc}\quad \frac{}{(C \circ (\square \triangleleft \mathsf{E}_{\mathsf{m}}))[v] \longrightarrow C[v \triangleleft \mathsf{E}_{\mathsf{m}}]}$$

$$\textsc{Sem-eterm-FillERed}\quad \frac{\mathsf{h}' \ = \ \mathit{max}(\mathit{hvars}(C) \cup \{\mathsf{h}\})+1}{C[+\mathsf{h} \triangleleft \mathsf{E}_{\mathsf{m}}] \longrightarrow C[\mathsf{h} \mathrel{:=}_{\{\mathsf{h}'+1\}} \ \mathsf{E}_{\mathsf{m}} -(\mathsf{h}'+1)][+(\mathsf{h}'+1)]}$$

$$\textsc{Sem-eterm-FillPFoc}\quad \frac{\mathtt{NotVal}\ \ t}{C[t \triangleleft (,)] \longrightarrow (C \circ (\square \triangleleft (,)))[t]}$$

$$\textsc{Sem-eterm-FillPUnfoc}\quad \frac{}{(C \circ (\square \triangleleft (,)))[v] \longrightarrow C[v \triangleleft (,)]}$$

$$\textsc{Sem-eterm-FillPRed}\quad \frac{\mathsf{h}' \ = \ \mathit{max}(\mathit{hvars}(C) \cup \{\mathsf{h}\})+1}{C[+\mathsf{h} \triangleleft (,)] \longrightarrow C[\mathsf{h} \mathrel{:=}_{\{\mathsf{h}'+1,\mathsf{h}'+2\}} \ (-(\mathsf{h}'+1)\,,\,-(\mathsf{h}'+2))][(+(\mathsf{h}'+1)\,,\,+(\mathsf{h}'+2))]}$$

$$\textsc{Sem-eterm-FillFFoc}\quad \frac{\mathtt{NotVal}\ \ t}{C[t \triangleleft (\lambda \mathsf{x}\ {}_{\mathsf{m}}\!\mapsto\ u)] \longrightarrow (C \circ (\square \triangleleft (\lambda \mathsf{x}\ {}_{\mathsf{m}}\!\mapsto\ u)))[t]}$$

$$\textsc{Sem-eterm-FillFUnfoc}\quad \frac{}{(C \circ (\square \triangleleft (\lambda \mathsf{x}\ {}_{\mathsf{m}}\!\mapsto\ u)))[v] \longrightarrow C[v \triangleleft (\lambda \mathsf{x}\ {}_{\mathsf{m}}\!\mapsto\ u)]}$$

$$\textsc{Sem-eterm-FillFRed}\quad \frac{}{C[+\mathsf{h} \triangleleft (\lambda \mathsf{x}\ {}_{\mathsf{m}}\!\mapsto\ u)] \longrightarrow C[\mathsf{h} \mathrel{:=}_{\{\,\}} \ {}^{\mathsf{v}}\lambda \mathsf{x}\ {}_{\mathsf{m}}\!\mapsto\ u][()]}$$

$$\textsc{Sem-eterm-FillCFoc1}\quad \frac{\mathtt{NotVal}\ \ t}{C[t \triangleleft\!\bullet\, t'] \longrightarrow (C \circ (\square \triangleleft\!\bullet\, t'))[t]}$$

$$\textsc{Sem-eterm-FillCUnfoc1}\quad \frac{}{(C \circ (\square \triangleleft\!\bullet\, t'))[v] \longrightarrow C[v \triangleleft\!\bullet\, t']}$$

$$\textsc{Sem-eterm-FillCFoc2}\quad \frac{\mathtt{NotVal}\ \ t'}{C[v \triangleleft\!\bullet\, t'] \longrightarrow (C \circ (v \triangleleft\!\bullet\, \square))[t']}$$

$$\textsc{Sem-eterm-FillCUnfoc2}\quad \frac{}{(C \circ (v \triangleleft\!\bullet\, \square))[v'] \longrightarrow C[v \triangleleft\!\bullet\, v']}$$

$$\textsc{Sem-eterm-FillCRed}\quad \frac{\mathsf{h}' \ = \ \mathit{max}(\mathit{hvars}(C) \cup \{\mathsf{h}\})+1}{C[+\mathsf{h} \triangleleft\!\bullet\, {}_{\mathsf{H}}\langle v_2\,,\ v_1\rangle] \longrightarrow C[\mathsf{h} \mathrel{:=}_{(\mathsf{H}\pm\mathsf{h}')} \ v_2[\mathsf{H}\pm\mathsf{h}']][v_1[\mathsf{H}\pm\mathsf{h}']]}$$

4 REMARKS ON THE COQ PROOFS

- Not particularly elegant. Max number of goals observed 232 (solved by a single call to the `congruence` tactic). When you have a computer, brute force is a viable strategy. (in particular, no semiring formalisation, it was quicker to do directly)
- Rules generated by ott, same as in the article (up to some notational difference). Contexts are not generated purely by syntax, and are interpreted in a semantic domain (finite functions).
- Reasoning on closed terms avoids almost all complications on binder manipulation. Makes proofs tractable.
- Finite functions: making a custom library was less headache than using existing libraries (including `MMap`). Existing libraries don't provide some of the tools that we needed, but the most important factor ended up being the need for a modicum of dependency between key and value. There wasn't really that out there. Backed by actual functions for simplicity; cost: equality is complicated.

- Most of the proofs done by author with very little prior experience to Coq.
- Did proofs in Coq because context manipulations are tricky.
- Context sum made total by adding an extra invalid *mode* (rather than an extra context). It seems to be much simpler this way.
- It might be a good idea to provide statistics on the number of lemmas and size of Coq codebase.
- (possibly) renaming as permutation, inspired by nominal sets, make more lemmas don't require a condition (but some lemmas that wouldn't in a straight renaming do in exchange).
- (possibly) methodology: assume a lot of lemmas, prove main theorem, prove assumptions, some wrong, fix. A number of wrong lemma initially assumed, but replacing them by correct variant was always easy to fix in proofs.
- Axioms that we use and why (in particular setoid equality not very natural with ott-generated typing rules).
- Talk about the use and benefits of Copilot.

REFERENCES

[1] Thomas Bagrel. 2024. Destination-passing style programming: a Haskell implementation. https://inria.hal.science/hal-04406360