The Recursive Index Calculus and Its Translation From Rω

AH & JGM

October 16, 2023

1 Design considerations of the formal index calculus

1.1 The necessity of Set-in-Set

Denote the translation of Rω type τ to the index calculus as $[\![\tau]\!]$. The big stars of systems R&c. are record and variants. Consider how we might translate the former. We first establish the translation of kinds so that we may assert that translated types are in the meaning of their translated kinds. (This is a sensible verification.) The following seems reasonable.

$$\begin{aligned}
[\![\star]\!]_\kappa &= \star \\
[\![\mathsf{L}]\!]_\kappa &= \top \\
[\![\kappa_1 \to \kappa_2]\!]_\kappa &= [\![\kappa_1]\!] \to [\![\kappa_2]\!] \\
[\![\mathsf{R}^\kappa]\!]_\kappa &= \exists i : Nat.(\mathrm{Fin}\ i \to [\![\kappa]\!])
\end{aligned}$$

(Nevermind, for now, the complexity of existentially quantifying Nats within kinds.) Recall that records (in Rω) live in kind $\star$ but, as shown in our Agda denotation, should translate to functions which map finite indices to types. Below is a sensible type and translation for such an idea. (As $[\![\rho]\!]$ is a dependent product, let its first and second projections be defined as usual.)

$$[\![\Pi\,\rho]\!] = \forall i : \text{Nat}.\text{Fin}\,[\![\rho]\!].1 \to [\![\rho]\!].2\,i$$

Yet, as $\Pi\,\rho : \star$ (in Rω), we must conclude that the type $\forall i : \text{Nat}.\text{Fin}\,[\![\rho]\!].1 \to [\![\rho]\!].2\,i$ has kind $\star$ in μIx. I don't see a way around this. Further, it implies to me that we would like to flatten types and kinds and let types quantify over types. That is to say, we want a two-level stratification of terms and types only, omitting kinds. The next two subsections argue for this change.

1.2 Flattening kinds and types

Revisit now the translation of rows.

$$[\![\mathsf{R}^{\kappa}]\!]_{\kappa} = \exists i : \text{Nat}.(\text{Fin}\,i \to [\![\kappa]\!])$$

The existential quantification of $i : \text{Nat}$ is ambiguous. We seem to need to either (i) permit the dependent, existential quantification over *type* Nat, or (ii) lift Nat indices to kinds and permit the existential quantification of indices in kinds. In the latter case, we would further need to represent the *application* of Fin to *index variable* i—and also, of course, need to represent existential quantification in kinds (and represent Fin!). This becomes cumbersome quickly, enough to beg the question: why distinguish types from kinds at all?

1.2.1 The inadequacy of Xi and Pfenning [1999]

Described above by (i) is the approach advocated by Xi and Pfenning [1999]. Xi and Pfenning [1999] only permits types to be indexed by a declared set of index objects, each with index sort—for example, vectors are indexed by index objects with index sort Nat. The syntax of (some simple) indices is given below.

$$\begin{array}{lrcl} \text{Index Sorts} & \gamma & ::= & \text{Nat} \mid \top \mid ... \\ \text{Index Objects} & \iota & ::= & i \mid () \mid ... \end{array}$$

Rather than types we have *families of types*, each indexed by *index objects*. For example, an *intlist* might be a family type family indexed by Nat, representing its length. Consider the typing of vector concatenation.

$$\texttt{concat} : \Pi m : \mathrm{Nat}.\Pi n : \mathrm{Nat}.\mathit{intlist}\ m \to \mathit{intlist}\ n \to \mathit{intlist}\ (m+n)$$

This machinery is not sufficient to type many Rω terms. To illustrate, presume a base set of index sorts γ and index objects ι to be as defined below. (For convenience, let the index objects of ι inhabit both Nat and Fin ι.)

$$\begin{array}{lcl} \gamma & ::= & \mathrm{Nat} \mid \mathrm{Fin}\,\iota \\ \iota & ::= & i \mid 0 \mid \mathrm{Suc}\,\iota \end{array}$$

Now consider an example translation of record concatenation from Rω to μIx. In Rω, we have:

$$\texttt{concat} : \forall (z_1 z_2 z_3 : \mathsf{R}^{\star}). z_1 \cdot z_2 \sim z_3 \Rightarrow \Pi\, z_1 \to \Pi\, z_2 \to \Pi\, z_3$$

This (hypothetically) translates to (the sketch I have in my head of) the μIx type:

$$\begin{aligned} \texttt{concat} :& \forall^i mnl : \mathit{Nat}. \\ & \forall (z_1 : \mathrm{Fin}\, m \to \star)(z_2 : \mathrm{Fin}\, n \to \star)(z_3 : \mathrm{Fin}\, l \to \star). \\ & [\![z_1 \cdot z_2\ z_3]\!] \to \\ & (\forall (i : \mathrm{Fin}\, m) \to z_1\ i) \\ & (\forall (i : \mathrm{Fin}\, n) \to z_2\ i) \\ & \forall (i : \mathrm{Fin}\, l). z_3\ i \end{aligned}$$

Let $\forall^i$ denote the quantification of indices in types and $[\![z_1 \cdot z_2 \sim z_3]\!]$ denote the (yet-undefined) translation of Rω predicates to μIx types. The glaring incompatibility of Xi and Pfenning [1999] is that Rω is higher-order. So, it is not clear what the quantification of z_1 over (Fin $m \to \star$) means. Is (Fin $m \to \star$) a kind? Xi and Pfenning [1999] permits only the quantification over indices, and the use of those indices in types. The authors do not permit $F\omega$ (or just System F)-style quantification over type variables. And it is not clear how to lift their calculus to higher-order; I suspect, also, nontrivial.

1.3 Let types quantify types

A lot of our problems go away when committing to an impredicative, dependent, term-and-type-stratified type theory. So, this is what I suggest we do. Firstly, the higher-order nature of $F\omega$ is given for free, as there are no more kinds and thus all type-level quantification is over types. W.r.t. mechanizational ease, we substantially reduce the overlap in ASTs.

What we are left with is more less MLTT with (built-in) finite naturals. Call it $\lambda^{\Pi,\Sigma}$ for now. It is described formally in the next section (but it has nothing in and of itself novel.)

2 The formalized index calculus

2.1 Syntax

Term variables x Type variables α Index variables i

$$
\begin{array}{llcl}
\text{Index Sorts} & \gamma & ::= & \mathrm{Nat} \mid \mathrm{Ix}\ \iota \\
\text{Index Objects} & \iota & ::= & i \mid \mathrm{Zero} \mid \mathrm{Suc}\,\iota \\
\text{Types} & \tau, \upsilon & ::= & \gamma \mid \iota \mid \star \mid \alpha \mid \top \mid \Pi i : \tau.\upsilon \mid \Sigma i : \tau.\upsilon \\
 & & & \tau\,\upsilon \mid \tau \sim \upsilon \\
\text{Terms} & M, N & ::= & x \mid () \mid \lambda x : \tau.M \mid M\ N \mid (M, N) \mid M.1 \mid M.2 \\
\text{Environments} & \Gamma & ::= & \varepsilon \mid \Gamma, x : \tau
\end{array}
$$

Figure 1: Syntax

2.2 Typing

$$\boxed{\Gamma \vdash \iota : \gamma}$$

$$\frac{\vdash \Gamma \quad i : \gamma \in \Gamma}{\Gamma \vdash i : \gamma}$$

$$\frac{\vdash \Gamma}{\Gamma \vdash \mathrm{Zero} : \mathrm{Nat}} \qquad \frac{\Gamma \vdash \iota : \mathrm{Nat}}{\Gamma \vdash \mathrm{Suc}\,\iota : \mathrm{Nat}}$$

$$\frac{\vdash \Gamma}{\Gamma \vdash \mathrm{Zero} : \mathrm{Ix}\,1} \qquad \frac{\Gamma \vdash n : \mathrm{Nat} \quad \Gamma \vdash \iota : \mathrm{Ix}\,n}{\Gamma \vdash \mathrm{Suc}\,\iota : \mathrm{Ix}\,(\mathrm{Suc}\,n)} \qquad \frac{\Gamma \vdash n : \mathrm{Nat} \quad \Gamma \vdash \iota : \mathrm{Ix}\,n}{\Gamma \vdash \iota : \mathrm{Ix}\,(\mathrm{Suc}\,n)}$$

$$\boxed{\vdash \Gamma}$$

$$(\textsc{c-emp})\ \frac{}{\vdash \varepsilon} \qquad (\textsc{c-var})\ \frac{\vdash \Gamma \quad \Gamma \vdash \tau\ \mathrm{Type}}{\vdash \Gamma, x : \tau}$$

$$\boxed{\Gamma \vdash \tau : \upsilon}$$

$$(\textsc{k-}\star\textsc{-in-}\star)\ \frac{\vdash \Gamma}{\Gamma \vdash \star : \star} \qquad (\textsc{k-}\top)\ \frac{\vdash \Gamma}{\Gamma \vdash \top : \star} \qquad (\textsc{k-var})\ \frac{\vdash \Gamma \quad x : \tau \in \Gamma}{\Gamma \vdash x : \tau}$$

$$(\textsc{k-}\forall)\ \frac{\Gamma, \alpha : \kappa \vdash \tau : \star}{\Gamma \vdash \forall \alpha : \kappa . \tau : \star} \qquad (\textsc{k-}\exists)\ \frac{\Gamma, \alpha : \tau \vdash \upsilon : \star}{\Gamma \vdash \exists \alpha : \tau . \upsilon : \star}$$

$$(\textsc{k-}\Pi)\ \frac{\Gamma, \alpha : \kappa \vdash \tau : \star}{\Gamma \vdash \forall \alpha : \kappa . \tau : \star} \qquad (\textsc{k-}\Sigma)\ \frac{\Gamma, \alpha : \kappa \vdash \tau : \star}{\Gamma \vdash \exists \alpha : \kappa . \tau : \star}$$

$$(\textsc{k-}{\to}I)\ \frac{\Gamma, \alpha : \kappa_1 \vdash \tau : \kappa_2}{\Gamma \vdash \lambda \alpha : \kappa_1 . \tau : \kappa_1 \to \kappa_2} \qquad (\textsc{k-}{\to}E)\ \frac{\Gamma \vdash \tau_1 : \kappa_1 \to \kappa_2 \quad \Gamma \vdash \tau_2 : \kappa_1}{\Gamma \vdash \tau_1\, \tau_2 : \kappa_2}$$

$$(\textsc{k-}\sim)\ \frac{\Gamma \vdash \tau_1 : \star \quad \Gamma \vdash \tau_2 : \star}{\Gamma \vdash \tau_1 \sim \tau_2 : \star} \ (\textsc{k-Ix})\ \frac{\Gamma \vdash \tau : \mathrm{Nat}}{\Gamma \vdash \mathrm{Ix}\ \tau : \star}$$

$$(\textsc{k-}\mu)\ \frac{\Gamma \vdash \tau : \star \to \star}{\Gamma \vdash \mu\tau : \star} \qquad (\textsc{k-}\nu)\ \frac{\Gamma \vdash \tau : \star \to \star}{\Gamma \vdash \nu\tau : \star}$$

Figure 2: Contexts and kinding.

2.3 Typing

3 Adding Recursion to Rω

The static semantics of Rω, as defined in Hubers and Morris [2023], are given in Appendix A for reference. We define only the syntax and rules necessary for least- and greatest-fixed points with general term-level recursion, which are routine.

$$\boxed{\Gamma \vdash \tau : \kappa}$$

$$\frac{\Gamma \vdash \tau : \star \to \star}{\Gamma \vdash \mu\tau : \star} \qquad \frac{\Gamma \vdash \tau : \star \to \star}{\Gamma \vdash \nu\tau : \star}$$

$$\boxed{\Gamma \vdash M : \tau}$$

$$\ldots$$

4 Translating μIx from Rω

Rules for the System F_ω fragment of Rω have a trivial correspondence to the F_ω fragment of μIx and are omitted. The syntax and typing judgments on the left are that of Rω (see Appendix A); on the right are μIx.

$$\boxed{[\![\kappa]\!]}$$

$$
\begin{aligned}
[\![\star]\!] &= \star \\
[\![\mathsf{L}]\!] &= \star \\
[\![\kappa_1 \to \kappa_2]\!] &= [\![\kappa_1]\!] \to [\![\kappa_2]\!] \\
[\![\mathsf{R}^\kappa]\!] &= \mathrm{Nat} \to [\![\kappa]\!]
\end{aligned}
$$

$$\boxed{[\![\Gamma \vdash \tau : \kappa]\!]}$$

$$
\begin{aligned}
[\![\frac{\Gamma \vdash \rho : \mathsf{R}^\kappa}{\Gamma \vdash \Pi\rho : \kappa}]\!] &= \forall n : \mathrm{Nat}.\mathit{Ix}\ n \to [\![\Gamma \vdash \rho : \mathsf{R}^\kappa]\!]\ n \\
[\![\frac{\Gamma \vdash \rho : \mathsf{R}^\kappa}{\Gamma \vdash \Sigma\rho : \kappa}]\!] &= \exists n : \mathrm{Nat}.[\![\Gamma \vdash \rho : \mathsf{R}^\kappa]\!]\ n \\
[\![\frac{\vdash \Gamma}{\Gamma \vdash \ell : \mathsf{L}}]\!] &= \top \\
[\![\frac{\Gamma \vdash \xi : \mathsf{L}}{\Gamma \vdash \lfloor \xi \rfloor : \star}]\!] &= \top \\
[\![\frac{\Gamma \vdash \xi : \mathsf{L} \quad \Gamma \vdash \tau : \kappa}{\Gamma \vdash \xi \triangleright \tau : \kappa}]\!] &= [\![\Gamma \vdash \tau : \kappa]\!] \\
[\![\frac{\Gamma \vdash \rho : \mathsf{R}^{\kappa_1 \to \kappa_2} \quad \Gamma \vdash \tau : \kappa_1}{\Gamma \vdash \rho\,\tau : \mathsf{R}^{\kappa_2}}]\!] &= \lambda n : \mathrm{Nat}.[\![\Gamma \vdash \rho : \mathsf{R}^{\kappa_1 \to \kappa_2}]\!]\ n\ [\![\Gamma \vdash \tau : \kappa_1]\!] \\
[\![\frac{\Gamma \vdash \phi : \kappa_1 \to \kappa_2 \quad \Gamma \vdash \rho : \mathsf{R}^{\kappa_1}}{\Gamma \vdash \phi\,\rho : \mathsf{R}^{\kappa_2}}]\!] &= \lambda n : \mathrm{Nat}.[\![\Gamma \vdash \phi : \kappa_1 \to \kappa_2]\!]\ n\ [\![\Gamma \vdash \rho : \mathsf{R}^{\kappa_1}]\!] \\
[\![\frac{\Gamma \vdash_{\mathcal{T}} \{\overline{\xi \triangleright \tau}\} : \mathsf{R}^\kappa}{\Gamma \vdash \{\overline{\xi \triangleright \tau}\} : \mathsf{R}^\kappa}]\!] &= [\![\Gamma \vdash_{\mathcal{T}} \{\overline{\xi \triangleright \tau}\} : \mathsf{R}^\kappa]\!] \\
[\![\frac{\Gamma \vdash \pi \quad \Gamma, \pi \vdash \tau : \star}{\Gamma \vdash \pi \Rightarrow \tau : \star}]\!] &= [\![\pi]\!] \to [\![\tau]\!]
\end{aligned}
$$

$$\boxed{[\![\Gamma \vdash \pi]\!]}$$

...

Figure 3: A compositional translation of Rω to μIx

References

Alex Hubers and J. Garrett Morris. Generic programming with extensible data types; or, making ad hoc extensible data types less ad hoc. *CoRR*, abs/2307.08759, 2023. doi: 10.48550/arXiv.2307.08759. URL `https://doi.org/10.48550/arXiv.2307.08759`.

Hongwei Xi and Frank Pfenning. Dependent types in practical programming. In *POPL '99, Proceedings of the 26th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, San Antonio, TX, USA, January 20-22, 1999*, pages 214–227, 1999. doi: 10.1145/292540.292560. URL `https://doi.org/10.1145/292540.292560`.

A The static semantics of Rω

A.1 Syntax

The syntax of R$\omega(\mathcal{T})$ is given in Figure 4.

Term variables x Type variables α Labels ℓ Directions $d \in \{\mathsf{L}, \mathsf{R}\}$

$$
\begin{array}{llcl}
\text{Kinds} & \kappa & ::= & \star \mid \mathsf{L} \mid \mathsf{R}^{\kappa} \mid \kappa \to \kappa \\
\text{Predicates} & \pi, \psi & ::= & \rho \lesssim_d \rho \mid \rho \odot \rho \sim \rho \\
\text{Types} & \phi, \tau, \upsilon, \rho, \xi & ::= & \alpha \mid (\to) \mid \pi \Rightarrow \tau \mid \forall \alpha : \kappa.\tau \mid \lambda \alpha : \kappa.\tau \mid \tau\,\tau \\
 & & \mid & \ell \mid \lfloor \xi \rfloor \mid \xi \triangleright \tau \mid \{\tau_1, \ldots, \tau_n\} \mid \Pi\rho \mid \Sigma\rho \\
\text{Terms} & M, N & ::= & x \mid \lambda x : \tau.M \mid M\;N \mid \Lambda \alpha : \kappa.M \mid M\;[\tau] \\
 & & \mid & \ell \mid M \triangleright M \mid M/M \mid \mathsf{prj}_d\,M \mid M \mathbin{+\!\!+} M \mid \mathsf{inj}_d\,M \mid M \triangledown M \\
 & & \mid & \mathsf{syn}_\phi\,M \mid \mathsf{ana}_\phi\,M \mid \mathsf{fold}\,M\;M\;M\;M \\
\text{Environments} & \Gamma & ::= & \varepsilon \mid \Gamma, \alpha : \kappa \mid \Gamma, x : \tau \mid \Gamma, \pi
\end{array}
$$

Figure 4: Syntax

A.2 Types and Kinds

Figure 5 gives rules for context formation ($\vdash \Gamma$), kinding ($\Gamma \vdash \tau : \kappa$), and predicate formation ($\Gamma \vdash \pi$), parameterized by row theory $\mathcal{T}$.

$$\boxed{\vdash \Gamma}$$

$$(\textsc{c-emp})\ \frac{}{\vdash \varepsilon} \qquad (\textsc{c-tvar})\ \frac{\vdash \Gamma}{\vdash \Gamma, \alpha : \kappa} \qquad (\textsc{c-var})\ \frac{\vdash \Gamma \quad \Gamma \vdash \tau : \star}{\vdash \Gamma, x : \tau} \qquad (\textsc{c-pred})\ \frac{\vdash \Gamma \quad \Gamma \vdash \pi}{\vdash \Gamma, \pi}$$

$$\boxed{\Gamma \vdash \tau : \kappa}\ \boxed{\Gamma \vdash \pi}$$

$$(\textsc{k-var})\ \frac{\vdash \Gamma \quad \alpha : \kappa \in \Gamma}{\Gamma \vdash \alpha : \kappa} \qquad (\textsc{k-}(\to))\ \frac{\vdash \Gamma}{\Gamma \vdash (\to) : \star \to \star \to \star} \qquad (\textsc{k-}\Rightarrow)\ \frac{\Gamma \vdash \pi \quad \Gamma, \pi \vdash \tau : \star}{\Gamma \vdash \pi \Rightarrow \tau : \star}$$

$$(\textsc{k-}\forall)\ \frac{\Gamma, \alpha : \kappa \vdash \tau : \star}{\Gamma \vdash \forall \alpha : \kappa . \tau : \star} \qquad (\textsc{k-}{\to}I)\ \frac{\Gamma, \alpha : \kappa_1 \vdash \tau : \kappa_2}{\Gamma \vdash \lambda \alpha : \kappa_1 . \tau : \kappa_1 \to \kappa_2} \qquad (\textsc{k-}{\to}E)\ \frac{\Gamma \vdash \tau_1 : \kappa_1 \to \kappa_2 \quad \Gamma \vdash \tau_2 : \kappa_1}{\Gamma \vdash \tau_1\, \tau_2 : \kappa_2}$$

$$(\textsc{k-lab})\ \frac{\vdash \Gamma}{\Gamma \vdash \ell : \mathsf{L}} \qquad (\textsc{k-sing})\ \frac{\Gamma \vdash \xi : \mathsf{L}}{\Gamma \vdash \lfloor \xi \rfloor : \star} \qquad (\textsc{k-lty})\ \frac{\Gamma \vdash \xi : \mathsf{L} \quad \Gamma \vdash \tau : \kappa}{\Gamma \vdash \xi \triangleright \tau : \kappa} \qquad (\textsc{k-row})\ \frac{\Gamma \vdash_{\mathcal{T}} \{\overline{\xi \triangleright \tau}\} : \mathsf{R}^{\kappa}}{\Gamma \vdash \{\overline{\xi \triangleright \tau}\} : \mathsf{R}^{\kappa}}$$

$$(\textsc{k-}\Pi)\ \frac{\Gamma \vdash \rho : \mathsf{R}^{\kappa}}{\Gamma \vdash \Pi \rho : \kappa} \qquad (\textsc{k-}\Sigma)\ \frac{\Gamma \vdash \rho : \mathsf{R}^{\kappa}}{\Gamma \vdash \Sigma \rho : \kappa} \qquad (\textsc{k-lift}_1)\ \frac{\Gamma \vdash \rho : \mathsf{R}^{\kappa_1 \to \kappa_2} \quad \Gamma \vdash \tau : \kappa_1}{\Gamma \vdash \rho\, \tau : \mathsf{R}^{\kappa_2}}$$

$$(\textsc{k-lift}_2)\ \frac{\Gamma \vdash \phi : \kappa_1 \to \kappa_2 \quad \Gamma \vdash \rho : \mathsf{R}^{\kappa_1}}{\Gamma \vdash \phi\, \rho : \mathsf{R}^{\kappa_2}} \qquad (\textsc{k-}\lesssim_d)\ \frac{\Gamma \vdash \rho_i : \mathsf{R}^{\kappa}}{\Gamma \vdash \rho_1 \lesssim_d \rho_2} \qquad (\textsc{k-}\odot)\ \frac{\Gamma \vdash \rho_i : \mathsf{R}^{\kappa}}{\Gamma \vdash \rho_1 \odot \rho_2 \sim \rho_3}$$

Figure 5: Contexts and kinding.

$$\boxed{\tau \equiv \tau} \; \boxed{\pi \equiv \pi}$$

$$(\text{E-REFL})\ \frac{}{\tau \equiv \tau} \qquad (\text{E-SYM})\ \frac{\tau_1 \equiv \tau_2}{\tau_2 \equiv \tau_1} \qquad (\text{E-TRANS})\ \frac{\tau_1 \equiv \tau_2 \quad \tau_2 \equiv \tau_3}{\tau_1 \equiv \tau_3} \qquad (\text{E-}\beta)\ \frac{}{(\lambda\alpha : \kappa.\tau)\,\upsilon \equiv \tau[\upsilon/\alpha}$$

$$(\text{E-}\xi_{\Rightarrow})\ \frac{\pi_1 \equiv \pi_2 \quad \tau_1 \equiv \tau_2}{\pi_1 \Rightarrow \tau_1 \equiv \pi_2 \Rightarrow \tau_2} \qquad (\text{E-}\xi_{\forall})\ \frac{\tau[\gamma/\alpha] \equiv \upsilon[\gamma/\beta]}{\forall\alpha : \kappa.\tau \equiv \forall\beta : \kappa.\upsilon}\ (\gamma \notin \mathit{fv}(\tau, \upsilon)) \qquad (\text{E-}\xi_{\text{APP}})\ \frac{\tau_i \equiv \upsilon_i}{\tau_1\,\tau_2 \equiv \upsilon_1\,\upsilon_2}$$

$$(\text{E-}\xi_{\triangleright})\ \frac{\xi_1 \equiv \xi_2 \quad \tau_1 \equiv \tau_2}{\xi_1 \triangleright \tau_1 \equiv \xi_2 \triangleright \tau_2} \qquad (\text{E-ROW})\ \frac{\{\overline{\xi_i \triangleright \tau_i}\} \equiv_{\mathcal{T}} \{\overline{\xi'_j \triangleright \tau'_j}\}}{\{\overline{\xi_i \triangleright \tau_i}\} \equiv \{\overline{\xi'_j \triangleright \tau'_j}\}} \qquad (\text{E-}\xi_{\lfloor\cdot\rfloor})\ \frac{\xi_1 \equiv \xi_2}{\lfloor \xi_1 \rfloor \equiv \lfloor \xi_2 \rfloor}$$

$$(\text{E-LIFT}_1)\ \frac{}{\{\xi \triangleright \phi\}\,\tau \equiv \{\xi \triangleright \phi\,\tau\}} \qquad (\text{E-LIFT}_2)\ \frac{}{\phi\,\{\xi \triangleright \tau\} \equiv \{\xi \triangleright \phi\,\tau\}}$$

$$(\text{E-}\xi_{\Pi\Sigma})\ \frac{\rho_1 \equiv \rho_2}{K\,\rho_1 \equiv K\,\rho_2} \qquad (\text{E-LIFT}_3)\ \frac{}{(K\,\rho)\,\tau \equiv K\,(\rho\,\tau)} \qquad (\text{E-SING})\ \frac{}{K\{\xi \triangleright \tau\} \equiv \xi \triangleright \tau} \qquad (K \in \{\Pi, \Sigma\})$$

$$(\text{E-}\xi_{\lesssim_d})\ \frac{\tau_i \equiv \upsilon_i}{\tau_1 \lesssim_d \tau_2 \equiv \upsilon_1 \lesssim_d \upsilon_2} \qquad (\text{E-}\xi_{\odot})\ \frac{\tau_i \equiv \upsilon_i}{\tau_1 \odot \tau_2 \sim \tau_3 \equiv \upsilon_1 \odot \upsilon_2 \sim \upsilon_3}$$

Figure 6: Type and predicate equivalence

A.3 Terms

$$\boxed{\Gamma \vdash M : \tau}$$

$$(\textsc{t-var})\ \frac{\vdash \Gamma \quad x : \tau \in \Gamma}{\Gamma \vdash x : \tau} \qquad (\textsc{t-}{\to}I)\ \frac{\Gamma \vdash \tau_1 : \star \quad \Gamma, x : \tau_1 \vdash M : \tau_2}{\Gamma \vdash \lambda x : \tau_1 . M : \tau_1 \to \tau_2} \qquad (\textsc{t-}{\to}E)\ \frac{\Gamma \vdash M_1 : \tau_1 \to \tau_2 \quad \Gamma \vdash M_2}{\Gamma \vdash M_1\, M_2 : \tau_2}$$

$$(\textsc{t-}{\equiv})\ \frac{\Gamma \vdash M : \tau \quad \tau \equiv \upsilon}{\Gamma \vdash M : \upsilon} \qquad (\textsc{t-}{\Rightarrow}I)\ \frac{\Gamma \vdash \pi \quad \Gamma, \pi \vdash M : \tau}{\Gamma \vdash M : \pi \Rightarrow \tau} \qquad (\textsc{t-}{\Rightarrow}E)\ \frac{\Gamma \vdash M : \pi \Rightarrow \tau \quad \Gamma \Vdash_{\mathcal{T}} \pi}{\Gamma \vdash M : \tau}$$

$$(\textsc{t-}\forall I)\ \frac{\Gamma, \alpha : \kappa \vdash M : \tau}{\Gamma \vdash \Lambda \alpha : \kappa . M : \forall \alpha : \kappa . \tau} \qquad (\textsc{t-}\forall E)\ \frac{\Gamma \vdash M : \forall \alpha : \kappa . \tau \quad \Gamma \vdash \upsilon : \kappa}{\Gamma \vdash M\, [\upsilon] : \tau[\upsilon / \alpha]}$$

$$(\textsc{t-sing})\ \frac{\vdash \Gamma}{\Gamma \vdash \ell : \lfloor \ell \rfloor} \qquad (\textsc{t-}{\triangleright}I)\ \frac{\Gamma \vdash M_1 : \lfloor \ell \rfloor \quad \Gamma \vdash M_2 : \tau}{\Gamma \vdash M_1 \triangleright M_2 : \ell \triangleright \tau} \qquad (\textsc{t-}{\triangleright}E)\ \frac{\Gamma \vdash M_1 : \ell \triangleright \tau \quad \Gamma \vdash M_2 : \lfloor \ell \rfloor}{\Gamma \vdash M_1 / M_2 : \tau}$$

$$(\textsc{t-}\Pi E)\ \frac{\Gamma \vdash M : \Pi \rho_1 \quad \Gamma \Vdash_{\mathcal{T}} \rho_2 \lesssim_d \rho_1}{\Gamma \vdash \mathsf{prj}_d\, M : \Pi \rho_2} \qquad (\textsc{t-}\Pi I)\ \frac{\Gamma \vdash M_1 : \Pi \rho_1 \quad \Gamma \vdash M_2 : \Pi \rho_2 \quad \Gamma \Vdash_{\mathcal{T}} \rho_1 \odot \rho_2 \sim \rho_3}{\Gamma \vdash M_1 +\!\!+\, M_2 : \Pi \rho_3}$$

$$(\textsc{t-}\Sigma I)\ \frac{\Gamma \vdash M : \Sigma \rho_1 \quad \Gamma \Vdash_{\mathcal{T}} \rho_1 \lesssim_d \rho_2}{\Gamma \vdash \mathsf{inj}_d\, M : \Sigma \rho_2} \qquad (\textsc{t-}\Sigma E)\ \frac{\Gamma \vdash M_1 : \Sigma \rho_1 \to \tau \quad \Gamma \vdash M_2 : \Sigma \rho_2 \to \tau \quad \Gamma \Vdash_{\mathcal{T}} \rho_1 \odot \rho_2 \sim}{\Gamma \vdash M_1 \triangledown M_2 : \Sigma \rho_3 \to \tau}$$

$$(\textsc{t-}\mathsf{ana})\ \frac{\begin{array}{c}\Gamma \vdash \rho : \mathsf{R}^\kappa \quad \Gamma \vdash \phi : \kappa \to \star \\ \Gamma \vdash M : \forall l : \mathsf{L}, u : \kappa, y_1, z, y_2 : \mathsf{R}^\kappa . \,(y_1 \odot \{l \triangleright u\} \sim z, z \odot y_2 \sim \rho) \Rightarrow \lfloor l \rfloor \to \phi\, u \to \tau\end{array}}{\Gamma \vdash \mathsf{ana}_\phi\, M : \Sigma(\phi\, \rho) \to \tau}$$

$$(\textsc{t-}\mathsf{syn})\ \frac{\begin{array}{c}\Gamma \vdash \rho : \mathsf{R}^\kappa \quad \Gamma \vdash \phi : \kappa \to \star \\ \Gamma \vdash M : \forall l : \mathsf{L}, u : \kappa, y_1, z, y_2 : \mathsf{R}^\kappa . \,(y_1 \odot \{l \triangleright u\} \sim z, z \odot y_2 \sim \rho) \Rightarrow \lfloor l \rfloor \to \phi\, u\end{array}}{\Gamma \vdash \mathsf{syn}_\phi\, M : \Pi(\phi\, \rho)}$$

$$(\textsc{t-}\mathsf{fold})\ \frac{\begin{array}{c}M_1 : \forall l : \mathsf{L}, t : \star, y_1, z, y_2 : \mathsf{R}^\kappa . \,(y_1 \odot \{l \triangleright u\} \sim z, z \odot y_2 \sim \rho) \Rightarrow \lfloor l \rfloor \to t \to \upsilon \\ \Gamma \vdash M_2 : \upsilon \to \upsilon \to \upsilon \quad \Gamma \vdash M_3 : \upsilon \quad \Gamma \vdash N : \Pi \rho\end{array}}{\Gamma \vdash \mathsf{fold}\, M_1\, M_2\, M_3\, N : \upsilon}$$

Figure 7: Typing

Minimal Rows

Figure 8 gives the minimal row theory $\mathcal{M}$.

$$\boxed{\Gamma \vdash_{\mathsf{m}} \rho : \kappa} \; \boxed{\rho \equiv_{\mathsf{m}} \rho}$$

$$(\textsc{k-mrow})\ \frac{\Gamma \vdash \xi : \mathsf{L} \quad \Gamma \vdash \tau : \kappa}{\Gamma \vdash_{\mathsf{m}} \{\xi \triangleright \tau\} : \mathsf{R}^{\kappa}} \qquad (\textsc{e-mrow})\ \frac{\xi \equiv \xi' \quad \tau \equiv \tau'}{\{\xi \triangleright \tau\} \equiv_{\mathsf{m}} \{\xi' \triangleright \tau'\}}$$

$$\boxed{\Gamma \Vdash_{\mathsf{m}} \pi}$$

$$(\textsc{n-ax})\ \frac{\pi \in \Gamma}{\Gamma \Vdash_{\mathsf{m}} \pi} \qquad (\textsc{n-refl})\ \frac{}{\Gamma \Vdash_{\mathsf{m}} \rho \lesssim_d \rho} \qquad (\textsc{n-trans})\ \frac{\Gamma \Vdash_{\mathsf{m}} \rho_1 \lesssim_d \rho_2 \quad \Gamma \Vdash_{\mathsf{m}} \rho_2 \lesssim_d \rho_3}{\Gamma \Vdash_{\mathsf{m}} \rho_1 \lesssim_d \rho_3}$$

$$(\textsc{n-}\equiv)\ \frac{\Gamma \Vdash_{\mathsf{m}} \pi_1 \quad \pi_1 \equiv \pi_2}{\Gamma \Vdash_{\mathsf{m}} \pi_2} \qquad (\textsc{n-}\lesssim\textsc{lift}_1)\ \frac{\Gamma \Vdash_{\mathsf{m}} \rho_1 \lesssim_d \rho_2}{\Gamma \Vdash_{\mathsf{m}} \phi\, \rho_1 \lesssim_d \phi\, \rho_2} \qquad (\textsc{n-}\lesssim\textsc{lift}_2)\ \frac{\Gamma \Vdash_{\mathsf{m}} \rho_1 \lesssim_d \rho_2}{\Gamma \Vdash_{\mathsf{m}} \rho_1\, \tau \lesssim_d \rho_2\, \tau}$$

$$(\textsc{n-}\odot\textsc{lift}_1)\ \frac{\Gamma \Vdash_{\mathsf{m}} \rho_1 \odot \rho_2 \sim \rho_3}{\Gamma \Vdash_{\mathsf{m}} \rho_1\, \tau \odot \rho_2\, \tau \sim \rho_3\, \tau} \qquad (\textsc{n-}\odot\textsc{lift}_2)\ \frac{\Gamma \Vdash_{\mathsf{m}} \rho_1 \odot \rho_2 \sim \rho_3}{\Gamma \Vdash_{\mathsf{m}} \phi\, \rho_1 \odot \phi\, \rho_2 \sim \phi\, \rho_3}$$

$$(\textsc{n-}\odot\lesssim_{\mathsf{L}})\ \frac{\Gamma \Vdash_{\mathsf{m}} \rho_1 \odot \rho_2 \sim \rho_3}{\Gamma \Vdash_{\mathsf{m}} \rho_1 \lesssim_{\mathsf{L}} \rho_3} \qquad (\textsc{n-}\odot\lesssim_{\mathsf{R}})\ \frac{\Gamma \Vdash_{\mathsf{m}} \rho_1 \odot \rho_2 \sim \rho_3}{\Gamma \Vdash_{\mathsf{m}} \rho_2 \lesssim_{\mathsf{R}} \rho_3}$$

Figure 8: Minimal row theory $\mathcal{M} = \langle \vdash_{\mathsf{m}}, \equiv_{\mathsf{m}}, \Vdash_{\mathsf{m}} \rangle$