

A cleaner operational semantics

Mikkel Nygaard

October 17th, 2001

Abstract

The ordering on terms used in the equivalence proof of Section 3.4 of the progress report is not well-founded and the operational semantics suffers from a confusion between terms considered as terms and terms considered as terms-in-context. We address both issues below.

1 Patterns

General patterns are given by the grammar

$$p, q ::= 1 \mid x \mid a.p \mid (p, -) \mid (-, q) \mid p \otimes q$$

The associated judgments have the form $x_1 : \mathbb{P}_1, \dots, x_n : \mathbb{P}_n \Vdash p : \mathbb{P}$ where the variables of p are exactly the x_i with no repetitions. Such a judgment is interpreted as a functor $(\mathbb{P}_1 \otimes \dots \otimes \mathbb{P}_n)_\perp \xrightarrow{p} \mathbb{P}_\perp$ according to the following formation rules with interpretations:

$$\begin{array}{c} \overline{\Vdash 1 : \mathbb{P}} \\ \overline{x : \mathbb{P} \Vdash x : \mathbb{P}} \\ \frac{\Pi \Vdash p : \mathbb{P}_{a\perp}}{\Pi \Vdash a.p : \bigoplus_{\alpha} \mathbb{P}_{\alpha\perp}} \\ \frac{\Pi \Vdash p : \mathbb{P}}{\Pi \Vdash (p, -) : \mathbb{P} \& \mathbb{Q}} \\ \frac{\Pi \Vdash p : \mathbb{P} \quad \Lambda \Vdash q : \mathbb{Q}}{\Pi, \Lambda \Vdash p \otimes q : \mathbb{P} \otimes \mathbb{Q}} \end{array} \qquad \begin{array}{c} \overline{\mathbb{O}_\perp \xrightarrow{\perp \mapsto \perp} \mathbb{P}_\perp} \\ \overline{\mathbb{P}_\perp \xrightarrow{1} \mathbb{P}_\perp} \\ \frac{\Pi_\perp \xrightarrow{p} \mathbb{P}_{a\perp}}{\Pi_\perp \xrightarrow{\text{in}_a \circ p} \bigoplus_{\alpha} \mathbb{P}_{\alpha\perp} \xrightarrow{[-]} (\bigoplus_{\alpha} \mathbb{P}_{\alpha\perp})_\perp} \\ \frac{\Pi_\perp \xrightarrow{p} \mathbb{P}_\perp}{\Pi_\perp \xrightarrow{p} \mathbb{P}_\perp \xrightarrow{(\text{in}_1)_\perp} (\mathbb{P} \& \mathbb{Q})_\perp} \\ \frac{\Pi_\perp \xrightarrow{p} \mathbb{P}_\perp \quad \Lambda_\perp \xrightarrow{q} \mathbb{Q}_\perp}{(\Pi \otimes \Lambda)_\perp \cong \Pi_\perp \times \Lambda_\perp \xrightarrow{p \times q} \mathbb{P}_\perp \times \mathbb{Q}_\perp \cong (\mathbb{P} \otimes \mathbb{Q})_\perp} \end{array}$$

– where $[-] : \mathbb{P} \rightarrow \mathbb{P}_\perp$ maps P to $[P]$ and where for any functor $F : \mathbb{P} \rightarrow \mathbb{Q}$, we define the strict extension $F_\perp : \mathbb{P}_\perp \rightarrow \mathbb{Q}_\perp$ by $F_\perp \perp = \perp$ and $F_\perp [P] = [FP]$. Given a well-formed pattern $\Pi \Vdash p : \mathbb{P}$, we may construct the map $p^* : \Pi_\perp \rightarrow \mathbb{P}_\perp$ of **Cocont** to be

$$p^* X = \widehat{\Pi}_\perp (y_{\Pi_\perp}(p-), X) \stackrel{\text{Yoneda}}{\cong} X(p-)$$

and then

$$\llbracket [t > p \Rightarrow u] \rrbracket = \int^{P \in \Pi_{\perp}} (p^* \circ [t])P . uP$$

for $\Gamma \vdash t : \mathbb{P}$ and $\Pi, \Delta \vdash u : \mathbb{Q}$.

1.1 Strict patterns

The pattern sublanguage obtained by omitting $a.p$ is interpreted by strict functors $F_{\perp} : \mathbb{P}_{\perp} \rightarrow \mathbb{Q}_{\perp}$ according to the above. Of course, we may simplify such interpretations to just $F : \mathbb{P} \rightarrow \mathbb{Q}$ and doing so allows us to define a map $s^* : \Pi \rightarrow \mathbb{P}$ of **Cocont** and obtain

$$\llbracket [t > s \Rightarrow u] \rrbracket \cong \int^{P \in \Pi_{\perp}} [s^* \circ t]P . uP \cong u \circ_{\mathbf{Cocont}} (s^* \circ_{\mathbf{Cocont}} t)$$

for any strict pattern $s ::= 1 \mid x \mid (s, -) \mid (-, r) \mid s \otimes r$ – and t and u as above.

2 Environments

Let e range over finite lists – called environments – of the form

$$t_1 > s_1, \dots, t_n > s_n$$

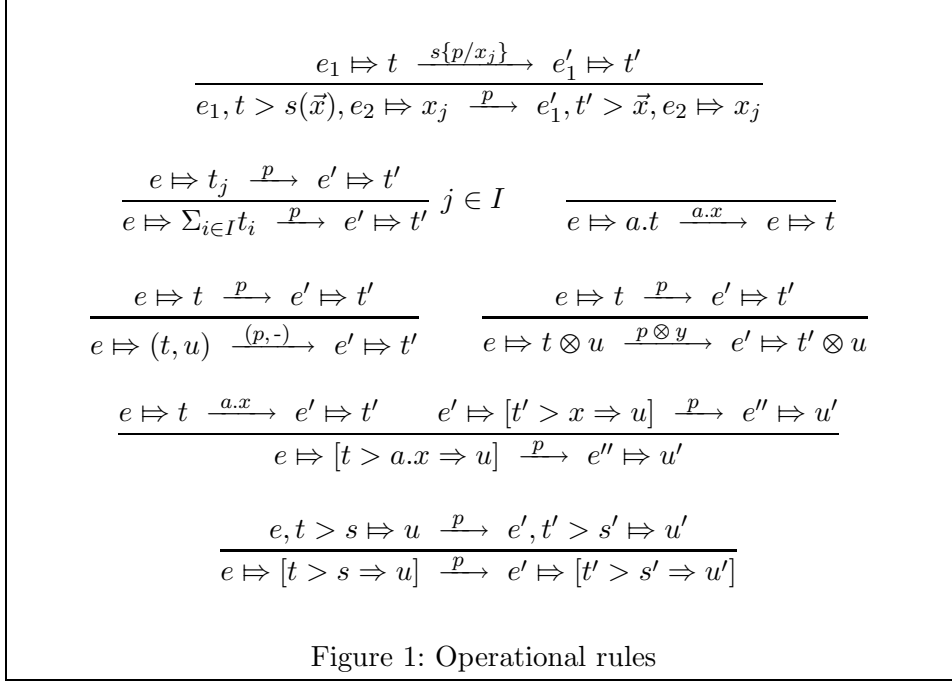
The empty environment will be written ε . For convenience, we'll assume that the variables occurring in the s_1, \dots, s_n are distinct. This restriction is not essential, but makes the notation simpler. An environment exports variables bound to presheaves. This can be formalised using a judgment like

$$e : x_1 : \mathbb{P}_1, \dots, x_n : \mathbb{P}_n$$

which is interpreted by a presheaf over $\mathbb{P}_1 \otimes \dots \otimes \mathbb{P}_n$. The formation rules and denotational semantics of environments are as follows:

$$\begin{array}{l} \frac{}{\varepsilon : \mathbb{O}} \qquad \qquad \qquad \emptyset \in \widehat{\mathbb{O}} \\ \\ \frac{e : \Gamma, x : \mathbb{P}}{e : \Gamma} \qquad \qquad \qquad (\text{via } \mathbb{P} \xrightarrow{0} \mathbb{O} \text{ and } \Gamma \otimes \mathbb{O} \cong \Gamma) \\ \\ \frac{e : \Gamma, x : \mathbb{P}, y : \mathbb{Q}, \Delta}{e : \Gamma, y : \mathbb{Q}, x : \mathbb{P}, \Delta} \qquad \qquad \qquad (\text{via } \mathbb{P} \otimes \mathbb{Q} \cong \mathbb{Q} \otimes \mathbb{P}) \\ \\ \frac{e : \Gamma, \Delta \quad \Gamma \vdash t : \mathbb{P} \quad \Pi \Vdash s : \mathbb{P}}{e, t > s : \Pi, \Delta} \quad \frac{e \in \widehat{\Gamma \otimes \Delta} \quad \Gamma \xrightarrow{t} \mathbb{P} \quad \Pi \xrightarrow{s} \mathbb{P}}{((s^* \circ t) \otimes 1_{\Delta})(e) \in \widehat{\Pi \otimes \Delta}} \end{array}$$

A term t in environment e will be written $e \Vdash t$. For $e : \Gamma, \Delta$ and $\Gamma \vdash t : \mathbb{P}$, we give the judgment $\vdash e \Vdash t : \mathbb{P}; \Delta$ the denotation $\llbracket e \Vdash t \rrbracket = (\llbracket t \rrbracket \otimes 1_{\Delta}) \llbracket e \rrbracket$ so that $\llbracket e \Vdash t \rrbracket \cong \llbracket [e \Rightarrow t \otimes \vec{x}] \rrbracket$ where \vec{x} are the variables exported by e but not free in t (and where $[\varepsilon \Rightarrow t] \equiv t$).



3 Operational rules – atomic steps

In the operational semantics it is inconvenient to allow general patterns. Fortunately, the full generality is not needed since we have the following isomorphisms

$$\begin{array}{ll}
[t > 1 \Rightarrow u] \cong u & \\
[t > a.p \Rightarrow u] \cong [t > a.x \Rightarrow [x > p \Rightarrow u]] & x \text{ fresh} \\
[t > (p, -) \Rightarrow u] \cong [t > (x, -) \Rightarrow [x > p \Rightarrow u]] & x \text{ fresh} \\
[t > p \otimes q \Rightarrow u] \cong [t > x \otimes y \Rightarrow [x > p \Rightarrow [y > q \Rightarrow u]]] & x, y \text{ fresh}
\end{array}$$

– effectively reducing the necessary patterns to $a.x$ and

$$s ::= x \mid (x, -) \mid (-, y) \mid x \otimes y$$

Consider the set of operational rules in Figure 1. The metavariables p, q range over atomic patterns,

$$p, q ::= a.x \mid (p, -) \mid (-, q) \mid p \otimes y \mid x \otimes q$$

In the first rule, the variable x_j is not bound by e_2 . The vector \vec{x} is either x_1 or $x_1 \otimes x_2$ with j being 1 or 2. Notice that x_j may well change its type going from left to right. The last rule says that the “point of control” of a term is found by moving strict patterns into the environment. In other words, the point of control is the match body to the far right looking through strict matches. It can be proven by rule induction that if $e \Vdash t \xrightarrow{p} e' \Vdash t'$ then e and e' are lists of the same length, so the last rule is general enough.

4 Type correctness

Proposition 4.1 *Assume $\vdash e \Rightarrow t : \mathbb{P}; \Delta$. If $e \Rightarrow t \xrightarrow{p} e' \Rightarrow t'$, then p is a path with $\Pi \Vdash p : \mathbb{P}$ for some Π , and $\vdash e' \Rightarrow t' : \Pi; \Delta$.*

Proof: By rule induction. We look at each rule in turn:

Variable The rule is

$$\frac{e_1 \Rightarrow t \xrightarrow{s\{p/x_j\}} e'_1 \Rightarrow t'}{e_1, t > s(\vec{x}), e_2 \Rightarrow x_j \xrightarrow{p} e'_1, t' > \vec{x}, e_2 \Rightarrow x_j}$$

with the variables of p and s disjoint. Assume that $\vdash e_1, t > s, e_2 \Rightarrow x_j : \mathbb{P}; \Delta$. Then $\vdash e_1 \Rightarrow t : \mathbb{Q}; \Delta'$ and $\Lambda_1, x_j : \mathbb{P}, \Lambda_2 \Vdash s : \mathbb{Q}$ for some $\mathbb{Q}, \Delta', \Lambda_1, \Lambda_2$. By the induction hypothesis, $\Lambda \Vdash s\{p/x_j\} : \mathbb{Q}$ for some Λ , but this means that $\Pi \Vdash p : \mathbb{P}$ for some Π and $\Lambda \equiv \Lambda_1, \Pi, \Lambda_2$. By further use of the induction hypothesis, $\vdash e'_1 \Rightarrow t' : \Lambda; \Delta'$ and so, if we change the type of x_j to Π , we get $\Lambda_1, x_j : \Pi, \Lambda_2 \Vdash \vec{x} : \Lambda$, so that $\vdash e'_1, t' > \vec{x}, e_2 \Rightarrow x_j : \Pi; \Delta$ as wanted.

Sum The rule is

$$\frac{e \Rightarrow t_j \xrightarrow{p} e' \Rightarrow t'}{e \Rightarrow \Sigma_{i \in I} t_i \xrightarrow{p} e' \Rightarrow t'} \quad j \in I$$

Assume $\vdash e \Rightarrow \Sigma_{i \in I} t_i : \mathbb{P}; \Delta$. Then for each $j \in I$, we have $\vdash e \Rightarrow t_j : \mathbb{P}; \Delta$, and so by the induction hypothesis, $\Pi \Vdash p : \mathbb{P}$ for some Π and $\vdash e' \Rightarrow t' : \Pi; \Delta$, as wanted.

Prefix The rule is:

$$\frac{}{e \Rightarrow a.t \xrightarrow{a.x} e \Rightarrow t}$$

Assume $\vdash e \Rightarrow a.t : \bigoplus_{\alpha} \mathbb{P}_{\alpha \perp}; \Delta$. Then $\vdash e \Rightarrow t : \mathbb{P}_a; \Delta$ and since $a.x$ has type $x : \mathbb{P}_a \Vdash a.x : \bigoplus_{\alpha} \mathbb{P}_{\alpha \perp}$ we are done.

Pair The left rule is:

$$\frac{e \Rightarrow t \xrightarrow{p} e' \Rightarrow t'}{e \Rightarrow (t, u) \xrightarrow{(p, -)} e' \Rightarrow t'}$$

Assume $\vdash e \Rightarrow (t, u) : \mathbb{P} \& \mathbb{Q}; \Delta$. Then $\vdash e \Rightarrow t : \mathbb{P}; \Delta$ and so by the induction hypothesis, $\Pi \Vdash p : \mathbb{P}$ for some Π and $\vdash e' \Rightarrow t' : \Pi; \Delta$. Since $\Pi \Vdash (p, -) : \mathbb{P} \& \mathbb{Q}$ we are done. The proof for the right rule is symmetric.

Tensor The left rule is:

$$\frac{e \Rightarrow t \xrightarrow{p} e' \Rightarrow t'}{e \Rightarrow t \otimes u \xrightarrow{p \otimes y} e' \Rightarrow t' \otimes u}$$

Assume $\vdash e \Rightarrow t \otimes u : \mathbb{P} \otimes \mathbb{Q}; \Delta$. Then for some Γ we have $\Gamma \Vdash u : \mathbb{Q}$ and $\vdash e \Rightarrow t : \mathbb{P}; \Gamma, \Delta$. By the induction hypothesis, $\Pi \Vdash p : \mathbb{P}$ for some Π and $\vdash e' \Rightarrow t' : \Pi; \Gamma, \Delta$. But then $\Pi, y : \mathbb{Q} \Vdash p \otimes y : \mathbb{P} \otimes \mathbb{Q}$ and $\vdash e' \Rightarrow t' \otimes u : \Pi \otimes \mathbb{Q}; \Delta$ as wanted. The proof for the right rule is symmetric.

Prefix match The rule is:

$$\frac{e \Vdash t \xrightarrow{a.x} e' \Vdash t' \quad e' \Vdash [t' > x \Rightarrow u] \xrightarrow{p} e'' \Vdash u'}{e \Vdash [t > a.x \Rightarrow u] \xrightarrow{p} e'' \Vdash u'}$$

Assume $\vdash e \Vdash [t > a.x \Rightarrow u] : \mathbb{P}; \Delta$. Then $\Gamma, x : \mathbb{P}_a \vdash u : \mathbb{P}$ and $\vdash e \Vdash t : \bigoplus_{\alpha} \mathbb{P}_{\alpha\perp}; \Gamma, \Delta$ for some $\Gamma, \bigoplus_{\alpha} \mathbb{P}_{\alpha\perp}, \Delta$. By the induction hypothesis for the left premise, we have $\vdash e' \Vdash t' : \mathbb{P}_a; \Gamma, \Delta$ and so $\vdash e' \Vdash [t' > x \Rightarrow u] : \mathbb{P}; \Delta$. The induction hypothesis for the right premise now yields $\Pi \Vdash p : \mathbb{P}$ and $\vdash e'' \Vdash u' : \Pi; \Delta$ as wanted.

Strict match The rule is:

$$\frac{e, t > s \Vdash u \xrightarrow{p} e', t' > s' \Vdash u'}{e \Vdash [t > s \Rightarrow u] \xrightarrow{p} e' \Vdash [t' > s' \Rightarrow u']}$$

Assume $\vdash e \Vdash [t > s \Rightarrow u] : \mathbb{P}; \Delta$. Then $\vdash e, t > s \Vdash u : \mathbb{P}, \Delta$ and so by the induction hypothesis, $\Pi \Vdash p : \mathbb{P}$ and $\vdash e', t' > s' \Vdash u' : \Pi; \Delta$. But then $\vdash e' \Vdash [t' > s' \Rightarrow u'] : \Pi; \Delta$ as wanted.

By rule induction, the proof is complete. \square

5 The size of terms

To enable an equivalence proof based on a well-founded relation \succ on terms, we define an ordinal $|t|$, measuring the size of the raw term t , by structural induction on t :

$$\begin{aligned} |x| &= 1 \\ |\Sigma_{i \in I} t_i| &= (\sup_{i \in I} |t_i|) \oplus 1 \\ |a.t| &= |t| \oplus 1 \\ |(t, u)| &= |t| \oplus |u| \\ |t \otimes u| &= |t| \oplus |u| \\ |[t > p \Rightarrow u]| &= |t| \oplus |p| \oplus |u| \end{aligned}$$

– where the size of a pattern, $|p|$, is the number of variables in p and \oplus is the natural addition of ordinals [Levy: Basic Set Theory, 1979]. The need for ordinals arises because of the possibly infinite sum. The operation \oplus is associative, has identity 0 and is strictly monotone in each argument. Notice that for all t , $|t| > 0$. We extend the definition to environments and terms in environments:

$$\begin{aligned} |\varepsilon| &= 0 \\ |e, t > s| &= |e| \oplus |t| \oplus |s| \\ |e \Vdash t| &= |e| \oplus |t| \end{aligned}$$

Then for all e and t we have $|e \Vdash t| = |[e \Rightarrow t]|$. We can now prove

Lemma 5.1 Suppose $e \vDash t \xrightarrow{p} e' \vDash t'$. Then $|e \vDash t| > |e' \vDash t'|$.

Proof: By rule induction. We consider each rule in turn.

Variable The rule is

$$\frac{e_1 \vDash t \xrightarrow{s\{p/x_j\}} e'_1 \vDash t'}{e_1, t > s(\vec{x}), e_2 \vDash x_j \xrightarrow{p} e'_1, t' > \vec{x}, e_2 \vDash x_j}$$

Notice that $|s| = |\vec{x}|$. By IH, we have $|e_1 \vDash t| > |e'_1 \vDash t'|$, and so

$$\begin{aligned} & |e_1, t > s, e_2 \vDash x_j| \\ &= |e_1 \vDash t| \oplus |s| \oplus |e_2| \oplus 1 \\ &> |e'_1 \vDash t'| \oplus |s| \oplus |e_2| \oplus 1 \quad \text{strict monotonicity} \\ &= |e'_1, t' > \vec{x}, e_2 \vDash x_j| \end{aligned}$$

as wanted.

Sum The rule is

$$\frac{e \vDash t_j \xrightarrow{p} e' \vDash t'}{e \vDash \Sigma_{i \in I} t_i \xrightarrow{p} e' \vDash t'} \quad j \in I$$

By IH, we have $|e \vDash t_j| > |e' \vDash t'|$, and so

$$\begin{aligned} & |e \vDash \Sigma_{i \in I} t_i| \\ &= |e| \oplus \sup_{i \in I} |t_i| \oplus 1 \\ &> |e| \oplus \sup_{i \in I} |t_i| \quad \text{strict monotonicity, identity zero} \\ &\geq |e| \oplus |t_j| \quad \text{property of sup, monotonicity} \\ &= |e \vDash t_j| \\ &> |e' \vDash t'| \end{aligned}$$

as wanted.

Prefix The rule is

$$\frac{}{e \vDash a.t \xrightarrow{a.x} e \vDash t}$$

We have

$$\begin{aligned} & |e \vDash a.t| \\ &= |e| \oplus |t| \oplus 1 \\ &> |e| \oplus |t| \quad \text{strict monotonicity, identity zero} \\ &= |e \vDash t| \end{aligned}$$

as wanted.

Pair The rule is

$$\frac{e \Vdash t \xrightarrow{p} e' \Vdash t'}{e \Vdash (t, u) \xrightarrow{(p, -)} e' \Vdash t'}$$

By IH, we have $|e \Vdash t| > |e' \Vdash t'|$, and so

$$\begin{aligned} & |e \Vdash (t, u)| \\ &= |e| \oplus |t| \oplus |u| \\ &> |e| \oplus |t| && \text{strict monotonicity, identity zero} \\ &= |e \Vdash t| \\ &> |e' \Vdash t'| \end{aligned}$$

as wanted.

Tensor The rule is

$$\frac{e \Vdash t \xrightarrow{p} e' \Vdash t'}{e \Vdash t \otimes u \xrightarrow{p \otimes y} e' \Vdash t' \otimes u}$$

By IH, we have $|e \Vdash t| > |e' \Vdash t'|$, and so

$$\begin{aligned} & |e \Vdash t \otimes u| \\ &= |e \Vdash t| \oplus |u| \\ &> |e' \Vdash t'| \oplus |u| && \text{strict monotonicity} \\ &= |e' \Vdash t' \otimes u| \end{aligned}$$

as wanted.

Prefix match The rule is

$$\frac{e \Vdash t \xrightarrow{a.x} e' \Vdash t' \quad e' \Vdash [t' > x \Rightarrow u] \xrightarrow{p} e'' \Vdash u'}{e \Vdash [t > a.x \Rightarrow u] \xrightarrow{p} e'' \Vdash u'}$$

By IH, we have $|e \Vdash t| > |e' \Vdash t'|$ and $|e' \Vdash [t' > x \Rightarrow u]| > |e'' \Vdash u'|$, and so

$$\begin{aligned} & |e \Vdash [t > a.x \Rightarrow u]| \\ &= |e \Vdash t| \oplus 1 \oplus |u| \\ &> |e' \Vdash t'| \oplus 1 \oplus |u| && \text{strict monotonicity} \\ &= |e' \Vdash [t' > x \Rightarrow u]| \\ &> |e'' \Vdash u'| \end{aligned}$$

as wanted.

Strict match The rule is

$$\frac{e, t > s \Vdash u \xrightarrow{p} e', t' > s' \Vdash u'}{e \Vdash [t > s \Rightarrow u] \xrightarrow{p} e' \Vdash [t' > s' \Rightarrow u']}$$

By IH, we have $|e, t > s \Rightarrow u| > |e', t' > s' \Rightarrow u'|$, and so

$$\begin{aligned} & |e \Rightarrow [t > s \Rightarrow u]| \\ &= |e, t > s \Rightarrow u| \\ &> |e', t' > s' \Rightarrow u'| \\ &= |e' \Rightarrow [t' > s' \Rightarrow u']| \end{aligned}$$

as wanted.

The proof is complete. \square

6 A well-founded relation

So transitions are accompanied by a decrease in size. But for the equivalence proof, we really need some ordering on terms in environments that decreases from conclusion to premise, ie. we want a well-founded order \succ such that for each rule

$$\frac{\dots e_u \Rightarrow u \xrightarrow{q} e'_u \Rightarrow u' \dots}{e_t \Rightarrow t \xrightarrow{p} e'_t \Rightarrow t'}$$

we have $e_t \Rightarrow t \succ e_u \Rightarrow u$. An obvious first choice would be

$$e \Rightarrow t \succ e' \Rightarrow t' \iff |e \Rightarrow t| > |e' \Rightarrow t'|$$

but that is not good enough for the last rule

$$\frac{e, t > s \Rightarrow u \xrightarrow{p} e', t' > s' \Rightarrow u'}{e \Rightarrow [t > s \Rightarrow u] \xrightarrow{p} e' \Rightarrow [t' > s' \Rightarrow u']}$$

where the sizes are the same. But here, the *term* part ($[t > s \Rightarrow u]$) in the conclusion is strictly larger than the term part (u) in the premise, and so defining \succ to be the lexicographic order

$$e \Rightarrow t \succ e' \Rightarrow t' \iff \begin{aligned} & |e \Rightarrow t| > |e' \Rightarrow t'| \vee \\ & |e \Rightarrow t| = |e' \Rightarrow t'| \wedge |t| > |t'| \end{aligned}$$

is sufficient:

Lemma 6.1 *For each rule*

$$\frac{\dots e_u \Rightarrow u \xrightarrow{q} e'_u \Rightarrow u' \dots}{e_t \Rightarrow t \xrightarrow{p} e'_t \Rightarrow t'}$$

we have $e_t \Rightarrow t \succ e_u \Rightarrow u$.

Proof: We look at the two interesting cases.

Prefix match The rule is

$$\frac{e \Vdash t \xrightarrow{a.x} e' \Vdash t' \quad e' \Vdash [t' > x \Rightarrow u] \xrightarrow{p} e'' \Vdash u'}{e \Vdash [t > a.x \Rightarrow u] \xrightarrow{p} e'' \Vdash u'}$$

We have

$$\begin{aligned} & |e \Vdash [t > a.x \Rightarrow u]| \\ &= |e \Vdash t| \oplus 1 \oplus |u| \\ &> |e \Vdash t| && \text{strict monotonicity, identity zero} \\ & |e \Vdash [t > a.x \Rightarrow u]| \\ &= |e \Vdash t| \oplus 1 \oplus |u| \\ &> |e' \Vdash t'| \oplus 1 \oplus |u| && \text{Lemma 5.1} \\ &= |e' \Vdash [t' > x \Rightarrow u]| \end{aligned}$$

as wanted.

Strict match The rule is

$$\frac{e, t > s \Vdash u \xrightarrow{p} e', t' > s' \Vdash u'}{e \Vdash [t > s \Rightarrow u] \xrightarrow{p} e' \Vdash [t' > s' \Rightarrow u']}$$

We have $|e \Vdash [t > s \Rightarrow u]| = |e, t > s \Vdash u|$ and

$$\begin{aligned} & |[t > s \Rightarrow u]| \\ &= |t| \oplus |s| \oplus |u| \\ &> |u| && \text{strict monotonicity, identity zero} \end{aligned}$$

as wanted.

The rule for prefix has no premises and so the wanted property holds vacuously. The remaining rules are handled as in Lemma 5.1. \square

7 Equivalence proof

Theorem 7.1 *Assume $\vdash t : \mathbb{P}$ and let $\Pi \Vdash p : \mathbb{P}$ be an atomic path. Then, summing over all derivations d with conclusion of the form $e \Vdash t \xrightarrow{p} e' \Vdash t'$, for some t' , we have $p^*[t] \cong \Sigma_d[t']$.*

Proof: By well-founded induction on \succ using the induction hypothesis

$$Q(\vdash e \Vdash t : \mathbb{P}; \Delta): \text{ Let } \Pi \Vdash p : \mathbb{P} \text{ be an atomic path and let } \vec{z} \text{ be any subset of the variables in } \Delta. \text{ Then, summing over all derivations } d \text{ with conclusion of the form } e \Vdash t \xrightarrow{p} e' \Vdash t', \text{ for some } e', t', \text{ we have } (p \otimes \vec{z})^*[[e \Rightarrow t \otimes \vec{z}]] \cong \Sigma_d[[e' \Rightarrow t' \otimes \vec{z}]].$$

Because of Proposition 4.1, we need not concern ourselves with questions of well-formedness in what follows. We proceed by case analysis on t :

Variable There is one possible last rule:

$$\frac{e_1 \Vdash t \xrightarrow{s\{p/x_j\}} e'_1 \Vdash t'}{e_1, t > s(\vec{x}), e_2 \Vdash x_j \xrightarrow{p} e'_1, t' > \vec{x}, e_2 \Vdash x_j}$$

Let \vec{y} be the variables exported by e_1 but not free in t . By the IH we have

$$(s\{p/x_j\} \otimes \vec{y})^* [[e_1 \Rightarrow t \otimes \vec{y}]] \cong \Sigma_{d_i} [[e'_1 \Rightarrow t' \otimes \vec{y}]].$$

Therefore,

$$\begin{aligned} & (p \otimes \vec{z})^* [[e_1, t > s, e_2 \Rightarrow x_j \otimes \vec{z}]] \\ & \cong (p \otimes \vec{z})^* [[e_1, t > s \Rightarrow x_j \otimes [e_2 \Rightarrow \vec{z}]]] \\ & \cong [e_1, t > s \Rightarrow p^*[x_j] \times_{\perp} [[e_2 \Rightarrow \vec{z}]]] \\ & \cong [e_1, t > s \Rightarrow [x_j > p(\vec{x}') \Rightarrow [\vec{x}']] \times_{\perp} [[e_2 \Rightarrow \vec{z}]]] \\ & \cong [e_1, t > s \Rightarrow [x_j > p(\vec{x}') \Rightarrow [e_2 \Rightarrow [\vec{x}' \otimes \vec{z}]]]] \\ & \cong [[e_1 \Rightarrow t \otimes \vec{y}] > s \otimes \vec{y} \Rightarrow [x_j > p(\vec{x}') \Rightarrow [e_2 \Rightarrow [\vec{x}' \otimes \vec{z}]]]] \\ & \cong [[e_1 \Rightarrow t \otimes \vec{y}] > s\{p(\vec{x}')/x_j\} \otimes \vec{y} \Rightarrow [e_2 \Rightarrow [\vec{x}' \otimes \vec{z}]]] \\ & \cong \Sigma_{d_i} [[e'_1 \Rightarrow t' \otimes \vec{y}] > \vec{x} \otimes \vec{y} \Rightarrow [e_2 \Rightarrow [x_j \otimes \vec{z}]]] \\ & \cong \Sigma_{d_i} [[e'_1, t' > \vec{x} \Rightarrow [e_2 \Rightarrow [x_j \otimes \vec{z}]]] \\ & \cong \Sigma_{d_i} [[e'_1, t' > \vec{x}, e_2 \Rightarrow x_j \otimes \vec{z}]] \end{aligned}$$

– as wanted.

Sum There is one possible last rule:

$$\frac{e \Vdash t_j \xrightarrow{p} e' \Vdash t'}{e \Vdash \Sigma_{i \in I} t_i \xrightarrow{p} e' \Vdash t'} \quad j \in I$$

If \vec{z} are exported by e and not free in $\Sigma_{i \in I} t_i$, they are not free in t_i for any i and so by IH we have

$$(p \otimes \vec{z})^* [[e \Rightarrow t_i \otimes \vec{z}]] \cong \Sigma_{d_i} [[e' \Rightarrow t' \otimes \vec{z}]].$$

for each $i \in I$. Therefore,

$$\begin{aligned} & (p \otimes \vec{z})^* [[e \Rightarrow \Sigma_{i \in I} t_i \otimes \vec{z}]] \\ & \cong [e \Rightarrow p^*[\Sigma_{i \in I} t_i] \times_{\perp} [\vec{z}]] \\ & \cong [e \Rightarrow (\Sigma_{i \in I} p^*[t_i]) \times_{\perp} [\vec{z}]] \\ & \cong [e \Rightarrow \Sigma_{i \in I} (p^*[t_i] \times_{\perp} [\vec{z}])] \\ & \cong \Sigma_{i \in I} [e \Rightarrow p^*[t_i] \times_{\perp} [\vec{z}]] \\ & \cong \Sigma_{i \in I} (p \otimes \vec{z})^* [[e \Rightarrow t_i \otimes \vec{z}]] \\ & \cong \Sigma_{i \in I} \Sigma_{d_i} [[e' \Rightarrow t' \otimes \vec{z}]] \end{aligned}$$

– as wanted.

Prefix There is one possible last rule:

$$\frac{}{e \mapsto a.t \xrightarrow{a.x} e \mapsto t}$$

We have

$$\begin{aligned} & (b.x \otimes \vec{z})^* [[e \Rightarrow a.t \otimes \vec{z}]] \\ & \cong [e \Rightarrow (b.x)^* [a.t] \times_{\perp} [\vec{z}]] \\ & \cong \begin{cases} [e \Rightarrow [t] \times_{\perp} [\vec{z}]] & \text{if } a \equiv b \\ [e \Rightarrow \emptyset \times_{\perp} [\vec{z}]] & \text{if } a \not\equiv b \end{cases} \\ & \cong \begin{cases} [[e \Rightarrow t \otimes \vec{z}]] & \text{if } a \equiv b \\ \emptyset & \text{if } a \not\equiv b \end{cases} \end{aligned}$$

– as wanted.

Pair There are two possible last rules, one of which is:

$$\frac{e \mapsto t \xrightarrow{p} e' \mapsto t'}{e \mapsto (t, u) \xrightarrow{(p, -)} e' \mapsto t'}$$

If \vec{z} are exported by e and not free in (t, u) , they are not free in t either and so by IH we have

$$(p \otimes \vec{z})^* [[e \Rightarrow t \otimes \vec{z}]] \cong \Sigma_{d_t} [[e' \Rightarrow t' \otimes \vec{z}]].$$

Therefore,

$$\begin{aligned} & ((p, -) \otimes \vec{z})^* [[e \Rightarrow (t, u) \otimes \vec{z}]] \\ & \cong [e \Rightarrow (p, -)^* [(t, u)] \times_{\perp} [\vec{z}]] \\ & \cong [e \Rightarrow p^* [t] \times_{\perp} [\vec{z}]] \\ & \cong (p \otimes \vec{z})^* [[e \Rightarrow t \otimes \vec{z}]] \\ & \cong \Sigma_{d_t} [[e' \Rightarrow t' \otimes \vec{z}]] \end{aligned}$$

– as wanted. The case with the rule involving $(-, q)$ is symmetric.

Tensor There are two possible last rules, one of which is:

$$\frac{e \mapsto t \xrightarrow{p} e' \mapsto t'}{e \mapsto t \otimes u \xrightarrow{p \otimes y} e' \mapsto t' \otimes u}$$

Let \vec{y} be the free variables of u . If \vec{z} are exported by e and not free in $t \otimes u$, then the variables $\vec{y}\vec{z}$ are not free in t and so by IH we have

$$(p \otimes \vec{y}\vec{z})^* [[e \Rightarrow t \otimes \vec{y}\vec{z}]] \cong \Sigma_{d_t} [[e' \Rightarrow t' \otimes \vec{y}\vec{z}]].$$

Let C_u be a suitable map $1 \times_{\perp} ([u] \times_{\perp} 1)$ of **Cocont**. Then,

$$\begin{aligned}
& ((p \otimes y) \otimes \vec{z})^* [[e \Rightarrow (t \otimes u) \otimes \vec{z}]] \\
& \cong [e \Rightarrow (p^*[t] \times_{\perp} [u]) \times_{\perp} [\vec{z}]] \\
& \cong [e \Rightarrow p^*[t] \times_{\perp} ([u] \times_{\perp} [\vec{z}])] \\
& \cong C_u([e \Rightarrow p^*[t] \times_{\perp} ([\vec{y}] \times_{\perp} [\vec{z}])]) \\
& \cong C_u((p \otimes \vec{y}\vec{z})^* [[e \Rightarrow t \otimes \vec{y}\vec{z}]]) \\
& \cong C_u(\Sigma_{d_t} [[e' \Rightarrow t' \otimes \vec{y}\vec{z}]]) \\
& \cong \Sigma_{d_t} [e' \Rightarrow C_u([t'] \times_{\perp} ([\vec{y}] \times_{\perp} [\vec{z}])]) \\
& \cong \Sigma_{d_t} [e' \Rightarrow [t'] \times_{\perp} ([u] \times_{\perp} [\vec{z}])] \\
& \cong \Sigma_{d_t} [e' \Rightarrow ([t'] \times_{\perp} [u]) \times_{\perp} [\vec{z}]] \\
& \cong \Sigma_{d_t} [[e' \Rightarrow (t' \otimes u) \otimes \vec{z}]]
\end{aligned}$$

– as wanted. The case with the rule involving $x \otimes q$ is symmetric.

Prefix match There is one possible last rule:

$$\frac{e \Rightarrow t \xrightarrow{a.x} e' \Rightarrow t' \quad e' \Rightarrow [t' > x \Rightarrow u] \xrightarrow{p} e'' \Rightarrow u'}{e \Rightarrow [t > a.x \Rightarrow u] \xrightarrow{p} e'' \Rightarrow u'}$$

Let \vec{y} be the variables exported by e but not free in t . By IH we have

$$(a.x \otimes \vec{y})^* [[e \Rightarrow t \otimes \vec{y}]] \cong \Sigma_{d_t} [[e' \Rightarrow t' \otimes \vec{y}]]$$

If \vec{z} are exported by e and not free in $[t > a.x \Rightarrow u]$, then \vec{z} are also not free in $[t' > x \Rightarrow u]$ and so by IH we have

$$(p \otimes \vec{z})^* [[e' \Rightarrow [t' > x \Rightarrow u] \otimes \vec{z}]] \cong \Sigma_{d_u} [[e'' \Rightarrow u' \otimes \vec{z}]]$$

Now,

$$\begin{aligned}
& [e \Rightarrow [t > a.x \Rightarrow u] \otimes \vec{z}] \\
& \cong [e \Rightarrow [[t > y \Rightarrow y] > a.x \Rightarrow u] \otimes \vec{z}] \\
& \cong [e \Rightarrow [t > y \Rightarrow [y > a.x \Rightarrow u]] \otimes \vec{z}] \\
& \cong [e \Rightarrow [t > y \Rightarrow [y > a.x \Rightarrow u] \otimes \vec{z}]] \\
& \cong [[e \Rightarrow t \otimes \vec{y}] > y \otimes \vec{y} \Rightarrow [y > a.x \Rightarrow u] \otimes \vec{z}]
\end{aligned}$$

and so, assuming – possibly by renaming – that x is not among \vec{z} ,

$$\begin{aligned}
& (p \otimes \vec{z})^* [[e \Rightarrow [t > a.x \Rightarrow u] \otimes \vec{z}]] \\
& \cong (p \otimes \vec{z})^* [[[e \Rightarrow t \otimes \vec{y}] > y \otimes \vec{y} \Rightarrow [y > a.x \Rightarrow u] \otimes \vec{z}]] \\
& \cong [[e \Rightarrow t \otimes \vec{y}] > y \otimes \vec{y} \Rightarrow p^* [[y > a.x \Rightarrow u]] \times_{\perp} [\vec{z}]] \\
& \cong [[e \Rightarrow t \otimes \vec{y}] > y \otimes \vec{y} \Rightarrow [y > a.x \Rightarrow p^*[u]] \times_{\perp} [\vec{z}]] \\
& \cong [[e \Rightarrow t \otimes \vec{y}] > a.x \otimes \vec{y} \Rightarrow p^*[u] \times_{\perp} [\vec{z}]] \\
& \cong [[e \Rightarrow t \otimes \vec{y}] > a.x \otimes \vec{y} \Rightarrow (p \otimes \vec{z})^* [u \otimes \vec{z}]] \\
& \cong \Sigma_{d_t} [[e' \Rightarrow t' \otimes \vec{y}] > x \otimes \vec{y} \Rightarrow (p \otimes \vec{z})^* [u \otimes \vec{z}]] \\
& \cong \Sigma_{d_t} (p \otimes \vec{z})^* [[[e' \Rightarrow t' \otimes \vec{y}] > x \otimes \vec{y} \Rightarrow u \otimes \vec{z}]] \\
& \cong \Sigma_{d_t} (p \otimes \vec{z})^* [[e' \Rightarrow [t' > x \Rightarrow u] \otimes \vec{z}]] \\
& \cong \Sigma_{d_t} (p \otimes \vec{z})^* [[e' \Rightarrow [t' > x \Rightarrow u] \otimes \vec{z}]] \\
& \cong \Sigma_{d_t} \Sigma_{d_u} [[e'' \Rightarrow u' \otimes \vec{z}]]
\end{aligned}$$

– as wanted.

Strict match There is one possible last rule:

$$\frac{e, t > s \Rightarrow u \xrightarrow{p} e', t' > s' \Rightarrow u'}{e \Rightarrow [t > s \Rightarrow u] \xrightarrow{p} e' \Rightarrow [t' > s' \Rightarrow u']}$$

We may assume – possibly by renaming – that if \vec{z} are exported by e and not free in $[t > s \Rightarrow u]$, then these variables do not occur in s and so are also not free in u . By the IH we have

$$(p \otimes \vec{z})^* [[e \Rightarrow [t > s \Rightarrow u \otimes \vec{z}]]] \cong \Sigma_{d_u} [[e' \Rightarrow [t' > s' \Rightarrow u' \otimes \vec{z}]]]$$

Therefore,

$$\begin{aligned} & (p \otimes \vec{z})^* [[e \Rightarrow [t > s \Rightarrow u] \otimes \vec{z}]] \\ & \cong (p \otimes \vec{z})^* [[e \Rightarrow [t > s \Rightarrow u \otimes \vec{z}]]] \\ & \cong \Sigma_{d_u} [[e' \Rightarrow [t' > s' \Rightarrow u' \otimes \vec{z}]]] \\ & \cong \Sigma_{d_u} [[e' \Rightarrow [t' > s' \Rightarrow u'] \otimes \vec{z}]] \end{aligned}$$

– as wanted.

The proof is complete. □