

Linearly-used Continuations in the Enriched Effect Calculus

Jeff Egger ^{*1}, Rasmus Ejlers Møgelberg ^{**2}, and Alex Simpson ^{*1}

¹ LFCS, School of Informatics, University of Edinburgh, Scotland, UK

² IT University of Copenhagen, Copenhagen, Denmark

Abstract. The *enriched effect calculus* is an extension of Moggi’s computational metalanguage with a selection of primitives from linear logic. In this paper, we present an extended case study within the enriched effect calculus: the linear usage of continuations. We show that established call-by-value and call-by name linearly-used CPS translations are uniformly captured by a single generic translation of the enriched effect calculus into itself. As a main syntactic theorem, we prove that the generic translation is involutive up to isomorphism. As corollaries, we obtain full completeness results for the original call-by-value and call-by-name translations. The main syntactic theorem is proved using a category-theoretic semantics for the enriched effect calculus. We show that models are closed under a natural *dual model* construction. The canonical linearly-used CPS translation then arises as the unique (up to isomorphism) map from the syntactic initial model to its own dual. This map is an equivalence of models. Thus the initial model is self-dual.

1 Introduction

The *continuations monad* $((-) \rightarrow R) \rightarrow R$ is commonly used to model control effects. Following Moggi’s idea of interpreting call-by-value programs in the Kleisli category [11, 12], a call-by-value program from X to Y is interpreted as a *continuation transformer* $(Y \rightarrow R) \rightarrow (X \rightarrow R)$. In an influential paper [3], Berdine *et al.* observe that, in many programming situations, continuation transformers satisfy an additional property: their argument, the continuation $Y \rightarrow R$, is used *linearly*. Thus a call-by-value program can be more informatively modelled as a linear function $(Y \rightarrow R) \multimap (X \rightarrow R)$, corresponding to a Kleisli map for the *linearly-used continuations monad* $((-) \rightarrow R) \multimap R$.

One goal of the present paper is to address the question: what is the natural type-theoretic (and semantic) context for modelling linearly-used continuations? With the presence of both intuitionistic (\rightarrow) and linear (\multimap) arrows, *intuitionistic linear type theory (ILL)* (and its model theory) seems a natural answer. Indeed, ILL has been used as the basis of a systematic study of linearly-used

* Research supported by EPSRC research grant “Linear Observations and Computational Effects”

** Research supported by the Danish Agency for Science, Technology and Innovation.

continuations by Hasegawa. In [6], he presents a continuation passing style (CPS) translation of Moggi’s call-by-value computational λ -calculus into ILL, using the linearly-used continuations monad, and establishes a full completeness result for this. A follow-up paper [7] considers call-by-name.

In this paper, we use a more general type theory, the *enriched effect calculus (EEC)* [4], as a context for modelling linearly-used continuations. On the one hand, it can be seen as a fragment of ILL and, as such, its models strictly generalise models of ILL. On the other hand, it is a conservative extension of the standard calculi for modelling computational effects (Moggi’s *computational metalanguage* [12], and Levy’s *call-by-push-value (CBPV)* [10]) with a selection of constructs from linear logic. In fact, any *adjunction model* of CBPV (and hence any model of Moggi’s computational metalanguage) expands to a model of EEC [4]. This provides an abundant supply of computationally interesting models of EEC that are not models of ILL.

The paper begins with a brief presentation of the enriched effect calculus in Section 2. This is followed, in Section 3, by the treatment of linearly-used continuations within EEC. The starting point is the observation that Hasegawa’s call-by-value [6] and call-by-name [7] linearly-used CPS translations of simply-typed λ -calculus both fall in that fragment of ILL corresponding to EEC. The first contribution of the paper is to show that, using EEC, we can recover these translations in a particularly interesting way. This is achieved by identifying a single canonical linearly-used CPS-translation of the entire enriched effect calculus into itself. Hasegawa’s call-by-value and call-by-name translations are derived from this by composing the canonical translation with the standard call-by-value and call-by-name encodings of typed λ -calculus into effect calculi (cf. Moggi [12], Filinski [5], Levy [10]).

The canonical linearly-used CPS-translation of EEC into itself possesses a remarkable property, unexpected in the context of CPS translations: it is involutive up to isomorphism. That is, the translation of a translated term equals the original term modulo type isomorphism. This property is the main syntactic theorem of the paper. As corollaries, we obtain full-completeness results for the call-by-value and call-by-name linearly-used CPS translations into EEC, mirroring Hasegawa’s results for the translations into ILL.

The second half of the paper provides a semantic context for the first. Section 4 reviews the notion of EEC model given in [4]. Following this, Section 5 gives a semantic account of the canonical linearly-used CPS translation. We show that the linearly-used continuations monad forms the basis of a duality of EEC models. The *dual model* of a model can be viewed as a linearly-used continuations model constructed over the original. The “dual” terminology is justified by every model being isomorphic to its own double dual. Thus, surprisingly, every model of the enriched effect calculus arises as a linearly-used continuations model relative to some other model.

Finally, we specialise the dual model construction to the syntactic model of EEC given by typed terms modulo equality. By the universal property of this model [4], there is a unique (up to isomorphism) morphism of models from the

$$\begin{array}{c}
\frac{}{\Gamma, x:A \mid - \vdash x:A} \quad \frac{}{\Gamma \mid z:\underline{A} \vdash z:\underline{A}} \quad \frac{}{\Gamma \mid \Delta \vdash *: 1} \\
\frac{\Gamma \mid \Delta \vdash t:A \quad \Gamma \mid \Delta \vdash u:B}{\Gamma \mid \Delta \vdash \langle t, u \rangle: A \times B} \quad \frac{\Gamma \mid \Delta \vdash t: A \times B}{\Gamma \mid \Delta \vdash \text{fst}(t): A} \quad \frac{\Gamma \mid \Delta \vdash t: A \times B}{\Gamma \mid \Delta \vdash \text{snd}(t): B} \\
\frac{\Gamma, x:A \mid \Delta \vdash t: B}{\Gamma \mid \Delta \vdash \lambda x:A. t: A \rightarrow B} \quad \frac{\Gamma \mid \Delta \vdash s: A \rightarrow B \quad \Gamma \mid - \vdash t: A}{\Gamma \mid \Delta \vdash s(t): B} \\
\frac{\Gamma \mid - \vdash t: A}{\Gamma \mid - \vdash !t: !A} \quad \frac{\Gamma \mid \Delta \vdash t: !A \quad \Gamma, x:A \mid - \vdash u: \underline{B}}{\Gamma \mid \Delta \vdash \text{let } !x \text{ be } t \text{ in } u: \underline{B}} \\
\frac{\Gamma \mid z:\underline{A} \vdash t: \underline{B}}{\Gamma \mid - \vdash \lambda z:\underline{A}. t: \underline{A} \multimap \underline{B}} \quad \frac{\Gamma \mid - \vdash s: \underline{A} \multimap \underline{B} \quad \Gamma \mid \Delta \vdash t: \underline{A}}{\Gamma \mid \Delta \vdash s[t]: \underline{B}} \\
\frac{\Gamma \mid - \vdash t: A \quad \Gamma \mid \Delta \vdash u: \underline{B}}{\Gamma \mid \Delta \vdash !t \otimes u: !A \otimes \underline{B}} \quad \frac{\Gamma \mid \Delta \vdash s: !A \otimes \underline{B} \quad \Gamma, x:A \mid z:\underline{B} \vdash t: \underline{C}}{\Gamma \mid \Delta \vdash \text{let } !x \otimes z \text{ be } s \text{ in } t: \underline{C}} \\
\frac{\Gamma \mid \Delta \vdash t: \underline{0}}{\Gamma \mid \Delta \vdash \text{image}(t): \underline{A}} \quad \frac{\Gamma \mid \Delta \vdash t: \underline{A}}{\Gamma \mid \Delta \vdash \text{inl}(t): \underline{A} \oplus \underline{B}} \quad \frac{\Gamma \mid \Delta \vdash t: \underline{B}}{\Gamma \mid \Delta \vdash \text{inr}(t): \underline{A} \oplus \underline{B}} \\
\frac{\Gamma \mid \Delta \vdash s: \underline{A} \oplus \underline{B} \quad \Gamma \mid x:\underline{A} \vdash t: \underline{C} \quad \Gamma \mid y:\underline{B} \vdash u: \underline{C}}{\Gamma \mid \Delta \vdash \text{case } s \text{ of } (\text{inl}(x). t; \text{inr}(y). u): \underline{C}}
\end{array}$$

Fig. 1. Typing rules for the effect calculus

syntactic model to its dual. This morphism is an equivalence of models. Thus the syntactic model is *self-dual*. Furthermore, the morphism identified from the syntactic model to its dual is nothing other than the canonical linearly-used CPS translation of EEC into itself from Section 3, and it is the equivalence property of this morphism that proves the involutivity of the translation. Thus we obtain a semantic proof of the main syntactic theorem of Section 3.

2 The enriched effect calculus

The *enriched effect calculus* [4] is an extension of Moggi's computational metalanguage [12] with constructors from linear type theory. Similar to Filinski's effect PCF [5] and Levy's CBPV [10], it has two notions of types: *value types* and *computation types*. We use α, β, \dots to range over a set of *value type constants*, and $\underline{\alpha}, \underline{\beta}, \dots$ to range over a disjoint set of *computation type constants*. We then use A, B, \dots to range over *value types*, and $\underline{A}, \underline{B}, \dots$ to range over *com-*

$\Gamma \mid \Delta \vdash t = * : 1$	if $\Gamma \mid \Delta \vdash t : 1$
$\Gamma \mid \Delta \vdash \text{fst}(\langle t, u \rangle) = t : \mathbf{A}$	if $\Gamma \mid \Delta \vdash t : \mathbf{A}$ and $\Gamma \mid \Delta \vdash t : \mathbf{B}$
$\Gamma \mid \Delta \vdash \text{snd}(\langle t, u \rangle) = u : \mathbf{B}$	if $\Gamma \mid \Delta \vdash t : \mathbf{A}$ and $\Gamma \mid \Delta \vdash t : \mathbf{B}$
$\Gamma \mid \Delta \vdash \langle \text{fst}(t), \text{snd}(t) \rangle = t : \mathbf{A} \times \mathbf{B}$	if $\Gamma \mid \Delta \vdash t : \mathbf{A} \times \mathbf{B}$
$\Gamma \mid \Delta \vdash (\lambda x : \mathbf{A}. t)(u) = t[u/x] : \mathbf{B}$	if $\Gamma, x : \mathbf{A} \mid \Delta \vdash t : \mathbf{B}$ and $\Gamma \mid - \vdash u : \mathbf{A}$
$\Gamma \mid \Delta \vdash \lambda x : \mathbf{A}. (t(x)) = t : \mathbf{A} \rightarrow \mathbf{B}$	if $\Gamma \mid \Delta \vdash t : \mathbf{A} \rightarrow \mathbf{B}$ and $x \notin \Gamma, \Delta$
$\Gamma \mid - \vdash \text{let } !x \text{ be } !t \text{ in } u = u[t/x] : \underline{\mathbf{B}}$	if $\Gamma \mid - \vdash t : \mathbf{A}$ and $\Gamma, x : \mathbf{A} \mid - \vdash u : \underline{\mathbf{B}}$
$\Gamma \mid \Delta \vdash \text{let } !x \text{ be } t \text{ in } u[!x/y] = u[t/y] : \underline{\mathbf{B}}$	if $\Gamma \mid \Delta \vdash t : !\mathbf{A}$ and $\Gamma \mid y : !\mathbf{A} \vdash u : \underline{\mathbf{B}}$
$\Gamma \mid \Delta \vdash (\lambda x : \underline{\mathbf{A}}. t)[u] = t[u/x] : \underline{\mathbf{B}}$	if $\Gamma \mid x : \underline{\mathbf{A}} \vdash t : \underline{\mathbf{B}}$ and $\Gamma \mid \Delta \vdash u : \underline{\mathbf{A}}$
$\Gamma \mid - \vdash \underline{\lambda} x : \underline{\mathbf{A}}. (t[x]) = t : \underline{\mathbf{A}} \multimap \underline{\mathbf{B}}$	if $\Gamma \mid - \vdash t : \underline{\mathbf{A}} \multimap \underline{\mathbf{B}}$ and $x \notin \Gamma$
$\Gamma \mid \Delta \vdash \text{let } !x \otimes y \text{ be } !t \otimes s \text{ in } u = u[t, s/x, y] : \underline{\mathbf{C}}$	if $\Gamma \mid - \vdash t : \mathbf{A}$ and $\Gamma \mid \Delta \vdash s : \underline{\mathbf{B}}$ and $\Gamma, x : \mathbf{A} \mid y : \underline{\mathbf{B}} \vdash u : \underline{\mathbf{C}}$
$\Gamma \mid \Delta \vdash \text{let } !x \otimes y \text{ be } t \text{ in } u[!x \otimes y/z] = u[t/z] : \underline{\mathbf{C}}$	if $\Gamma \mid \Delta \vdash t : !\mathbf{A} \otimes \underline{\mathbf{B}}$ and $\Gamma \mid z : !\mathbf{A} \otimes \underline{\mathbf{B}} \vdash u : \underline{\mathbf{C}}$
$\Gamma \mid x : \underline{\mathbf{0}} \vdash t = \text{image}(x) : \underline{\mathbf{A}}$	if $\Gamma \mid x : \underline{\mathbf{0}} \vdash t : \underline{\mathbf{A}}$
$\Gamma \mid \Delta \vdash \text{case } \text{inl}(t) \text{ of } (\text{inl}(x). u; \text{inr}(y). u')$ $\quad = u[t/x] : \underline{\mathbf{C}}$	if $\Gamma \mid x : \underline{\mathbf{A}} \vdash u : \underline{\mathbf{C}}$ and $\Gamma \mid y : \underline{\mathbf{B}} \vdash u' : \underline{\mathbf{C}}$ and $\Gamma \mid \Delta \vdash t : \underline{\mathbf{A}}$
$\Gamma \mid \Delta \vdash \text{case } \text{inr}(t) \text{ of } (\text{inl}(x). u; \text{inr}(y). u')$ $\quad = u'[t/y] : \underline{\mathbf{C}}$	if $\Gamma \mid x : \underline{\mathbf{A}} \vdash u : \underline{\mathbf{C}}$ and $\Gamma \mid y : \underline{\mathbf{B}} \vdash u' : \underline{\mathbf{C}}$ and $\Gamma \mid \Delta \vdash t : \underline{\mathbf{B}}$
$\Gamma \mid \Delta \vdash \text{case } t \text{ of } (\text{inl}(x). u[\text{inl}(x)/z]; \text{inr}(y). u[\text{inr}(y)/z])$ $\quad = u[t/z] : \underline{\mathbf{C}}$	if $\Gamma \mid \Delta \vdash t : \underline{\mathbf{A}} \oplus \underline{\mathbf{B}}$ and $\Gamma \mid z : \underline{\mathbf{A}} \oplus \underline{\mathbf{B}} \vdash u : \underline{\mathbf{C}}$

Fig. 2. Equality rules for the enriched effect calculus

putation types, which are specified by the grammar below:

$$\begin{aligned} \mathbf{A} ::= & \alpha \mid \underline{\alpha} \mid 1 \mid \mathbf{A} \times \mathbf{B} \mid \mathbf{A} \rightarrow \mathbf{B} \mid !\mathbf{A} \mid \underline{\mathbf{A}} \multimap \underline{\mathbf{B}} \mid !\mathbf{A} \otimes \underline{\mathbf{B}} \mid \underline{\mathbf{0}} \mid \underline{\mathbf{A}} \oplus \underline{\mathbf{B}} \\ \underline{\mathbf{A}} ::= & \underline{\alpha} \mid 1 \mid \underline{\mathbf{A}} \times \underline{\mathbf{B}} \mid \mathbf{A} \rightarrow \underline{\mathbf{B}} \mid !\mathbf{A} \mid !\mathbf{A} \otimes \underline{\mathbf{B}} \mid \underline{\mathbf{0}} \mid \underline{\mathbf{A}} \oplus \underline{\mathbf{B}} . \end{aligned}$$

The type constructor $!$ plays the role of Moggi's monadic type constructor T and Levy's F . Our notation has been chosen to exhibit the enriched effect calculus as a fragment of intuitionistic linear logic, a view which will be enhanced by the typing rules below.

Note that computation types form a subset of the value types (which is not the case in CBPV [10]). The reader is referred to [4] for further comparisons with other calculi and explanations of the enriched effect calculus.

The enriched effect calculus has two typing judgements:

$$(i) \quad \Gamma \mid - \vdash t : \mathbf{B} \qquad (ii) \quad \Gamma \mid z : \underline{\mathbf{A}} \vdash t : \underline{\mathbf{B}} ,$$

where Γ is a context of value-type assignments to variables. On the right of Γ is a *stoup*, which may either be empty, as in the case of judgement (i), or may

consist of a unique type assignment $x : \underline{\mathbf{A}}$, in which case the type on the right of the turnstyle is also required to be a computation type, as in (ii). The typing rules are given in Figure 1. In them, Δ ranges over an arbitrary (possibly empty) stoup, and the rules are only applicable in the case of typing judgements that conform to (i) or (ii) above.

Rules for equalities between typed terms are presented in Figure 2. They are to be considered in addition to the expected (typed) congruence and α -equivalence rules.

There is a standard call-by-value translation of typed λ -calculus into Moggi's computational metalanguage, Filinski's effect PCF [5], and Levy's CBPV [10]. Similarly, there is a standard call-by-name translation into the latter two, which exploits the existence of computation types. We recall these using the syntax of our effect calculus. As a source calculus, we use the simply-typed λ -calculus with types σ, τ, \dots given by:

$$\sigma ::= \alpha \mid 1 \mid \sigma \times \tau \mid \sigma \rightarrow \tau .$$

The call-by-value interpretation translates a type σ into a value type σ^{cbv} , and the call-by-name interpretation into a computation type σ^{cbn} .

$$\begin{array}{ll} \alpha^{\text{cbv}} = \alpha & \alpha^{\text{cbn}} = \underline{\alpha} \\ 1^{\text{cbv}} = 1 & 1^{\text{cbn}} = 1 \\ (\sigma \times \tau)^{\text{cbv}} = \sigma^{\text{cbv}} \times \tau^{\text{cbv}} & (\sigma \times \tau)^{\text{cbn}} = \sigma^{\text{cbn}} \times \tau^{\text{cbn}} \\ (\sigma \rightarrow \tau)^{\text{cbv}} = \sigma^{\text{cbv}} \rightarrow !\tau^{\text{cbv}} & (\sigma \rightarrow \tau)^{\text{cbn}} = \sigma^{\text{cbn}} \rightarrow \tau^{\text{cbn}} . \end{array}$$

Here, we assume that each type constant α of the typed λ -calculus, is included as a value-type constant, and has an associated computation-type constant $\underline{\alpha}$.

On terms, the cbv translation maps a judgement $x_1 : \sigma_1, \dots, x_n : \sigma_n \vdash t : \tau$ to

$$x_1 : \sigma_1^{\text{cbv}}, \dots, x_n : \sigma_n^{\text{cbv}} \mid - \vdash t^{\text{cbv}} : !\tau^{\text{cbv}} .$$

We omit the (routine) details. The cbn translation rather trivially maps a judgement $x_1 : \sigma_1, \dots, x_n : \sigma_n \vdash t : \tau$ to

$$x_1 : \sigma_1^{\text{cbn}}, \dots, x_n : \sigma_n^{\text{cbn}} \mid - \vdash t^{\text{cbn}} : \tau^{\text{cbn}} .$$

3 Linearly-used continuations

In [6], Hasegawa studies a translation from typed λ -calculus into intuitionistic linear type theory (ILL) which implements a continuation passing semantics in which continuations are used linearly. In [7], he defines a corresponding call-by-name translation (actually for a variant of Parigot's $\lambda\mu$ -calculus [13]). The call-by-value interpretation translates a type σ into a linear type $\sigma^{\text{cbv}\mathbb{R}}$, and the call-by-name interpretation translates σ to $\sigma^{\text{cbn}\mathbb{R}}$, as defined in Figure 3. Both translations are defined with respect to a distinguished type constant $\underline{\mathbb{R}}$, which

$$\begin{array}{ll}
\alpha^{\text{cbv}_R} = \alpha & \alpha^{\text{cbn}_R} = \underline{\alpha} \\
1^{\text{cbv}_R} = 1 & 1^{\text{cbn}_R} = \underline{0} \\
(\sigma \times \tau)^{\text{cbv}_R} = \sigma^{\text{cbv}_R} \times \tau^{\text{cbv}_R} & (\sigma \times \tau)^{\text{cbn}_R} = \sigma^{\text{cbn}_R} \oplus \tau^{\text{cbn}_R} \\
(\sigma \rightarrow \tau)^{\text{cbv}_R} = \sigma^{\text{cbv}_R} \rightarrow ((\tau^{\text{cbv}_R} \rightarrow \underline{R}) \rightarrow \underline{R}) & (\sigma \rightarrow \tau)^{\text{cbn}_R} = !(\sigma^{\text{cbn}_R} \rightarrow \underline{R}) \otimes \tau^{\text{cbn}_R} .
\end{array}$$

Fig. 3. Cbv and cbn linearly-used CPS translations of typed λ -calculus.

acts as a result type for continuations. Whereas Hasegawa's translations are into ILL, we give them into the enriched effect calculus, for which it is crucial that R is a computation type, hence the underlining. The important point is that Hasegawa's type translations fall into the EEC fragment of ILL.

For terms, Hasegawa [6, 7] gives translations into ILL. Once again, these translations land in the fragment of ILL given by the enriched effect calculus, and we shall just give them directly as translations into the latter. On terms, the cbv translation maps a judgement $x_1 : \sigma_1, \dots, x_n : \sigma_n \vdash t : \tau$ to

$$x_1 : \sigma_1^{\text{cbv}_R}, \dots, x_n : \sigma_n^{\text{cbv}_R} \mid - \vdash t^{\text{cbv}_R} : (\tau^{\text{cbv}_R} \rightarrow \underline{R}) \rightarrow \underline{R} .$$

The cbn translation maps a typing judgement $x_1 : \sigma_1, \dots, x_n : \sigma_n \vdash t : \tau$ to

$$x_1 : \sigma_1^{\text{cbn}_R} \rightarrow \underline{R}, \dots, x_n : \sigma_n^{\text{cbn}_R} \rightarrow \underline{R} \mid - \vdash t^{\text{cbn}_R} : \tau^{\text{cbn}_R} \rightarrow \underline{R} .$$

For lack of space, we do not give the details of the term translations, see [6, 7].

We now present the canonical linearly-used CPS translation of the entire EEC into itself. This translation maps any value type A to a value type $A^{\mathcal{V}_R}$ and any computation type \underline{B} to a computation type $\underline{B}^{\mathcal{C}_R}$, as defined in Figure 4. Note, that the result type \underline{R} is treated differently from the other computation-type constants, and $\underline{\alpha}$ ranges only over the latter. For each computation type \underline{A} , we obtain an isomorphism $i_{\underline{A}} : (\underline{A}^{\mathcal{C}_R} \rightarrow \underline{R}) \rightarrow \underline{A}^{\mathcal{V}_R}$, whose easy definition we omit.

Next, we define the translation of terms. We translate a typing judgement $\Gamma \mid - \vdash t : A$ as:

$$\Gamma^{\mathcal{V}_R} \mid - \vdash t^{\mathcal{V}_R} : A^{\mathcal{V}_R} ,$$

where $\Gamma^{\mathcal{V}_R}$ is the context obtained by applying $(-)^{\mathcal{V}_R}$ to every type in Γ . A typing judgement $\Gamma \mid z : \underline{A} \vdash t : \underline{B}$ is translated to:

$$\Gamma^{\mathcal{V}_R} \mid z : \underline{B}^{\mathcal{C}_R} \vdash t^{\mathcal{C}_R} : \underline{A}^{\mathcal{C}_R} .$$

The two translations are given in Figure 5. In this figure, each line corresponds to one of the typing rules in Figure 1. Observe that each typing rule that mentions Δ has two cases in Figure 5: one for empty stoup, and one for non-empty stoup. Also note that, in Figure 5, we always write z for the stoup variable.

We now establish the main properties of the translation. Since it is defined compositionally on the term structure, a straightforward induction proves:

$$\begin{array}{ll}
\alpha^{\mathcal{V}_R} = \alpha & \alpha^{C_R} = \alpha \\
\underline{\alpha}^{\mathcal{V}_R} = \underline{\alpha}^{C_R} \multimap \underline{R} & \underline{1}^{C_R} = \underline{0} \\
1^{\mathcal{V}_R} = 1 & (\underline{A} \times \underline{B})^{C_R} = \underline{A}^{C_R} \oplus \underline{B}^{C_R} \\
(\underline{A} \times \underline{B})^{\mathcal{V}_R} = \underline{A}^{\mathcal{V}_R} \times \underline{B}^{\mathcal{V}_R} & (\underline{A} \rightarrow \underline{B})^{C_R} = !\underline{A}^{\mathcal{V}_R} \otimes \underline{B}^{C_R} \\
(\underline{A} \rightarrow \underline{B})^{\mathcal{V}_R} = \underline{A}^{\mathcal{V}_R} \rightarrow \underline{B}^{\mathcal{V}_R} & (!\underline{A})^{C_R} = \underline{A}^{\mathcal{V}_R} \multimap \underline{R} \\
(!\underline{A})^{\mathcal{V}_R} = (!\underline{A})^{C_R} \multimap \underline{R} & (\underline{A} \otimes \underline{B})^{C_R} = \underline{A}^{\mathcal{V}_R} \rightarrow \underline{B}^{C_R} \\
(\underline{A} \multimap \underline{B})^{\mathcal{V}_R} = \underline{B}^{C_R} \multimap \underline{A}^{C_R} & \underline{0}^{C_R} = 1 \\
(!\underline{A} \otimes \underline{B})^{\mathcal{V}_R} = (!\underline{A} \otimes \underline{B})^{C_R} \multimap \underline{R} & (\underline{A} \oplus \underline{B})^{C_R} = \underline{A}^{C_R} \times \underline{B}^{C_R} \\
\underline{0}^{\mathcal{V}_R} = \underline{0}^{C_R} \multimap \underline{R} & \underline{R}^{C_R} = !1 \\
(\underline{A} \oplus \underline{B})^{\mathcal{V}_R} = (\underline{A} \oplus \underline{B})^{C_R} \multimap \underline{R} & \\
\underline{R}^{\mathcal{V}_R} = \underline{R} &
\end{array}$$

Fig. 4. Linearly-used CPS translation of enriched effect calculus types.

Proposition 1 (Soundness).

1. If $\Gamma \mid - \vdash t = u : \underline{A}$ then $\Gamma^{\mathcal{V}_R} \mid - \vdash t^{\mathcal{V}_R} = u^{\mathcal{V}_R} : \underline{A}^{\mathcal{V}_R}$.
2. If $\Gamma \mid z : \underline{A} \vdash t = u : \underline{B}$ then $\Gamma^{\mathcal{V}_R} \mid z : \underline{B}^{C_R} \vdash t^{C_R} = u^{C_R} : \underline{A}^{C_R}$.

A fundamental property is that, for the entire enriched effect calculus, the linearly-used CPS translation is involutive up to type isomorphism.

Theorem 1 (Involution property). *For every value type \underline{A} , there is an isomorphism $j_{\underline{A}} : \underline{A}^{\mathcal{V}_R \mathcal{V}_R} \rightarrow \underline{A}$, and, for every computation type \underline{A} , there is a linear isomorphism $k_{\underline{A}} : \underline{A}^{C_R C_R} \multimap \underline{A}$, such that:*

1. If $\Gamma \mid - \vdash t : \underline{A}$ then $t = j_{\underline{A}}(t^{\mathcal{V}_R \mathcal{V}_R}) [j^{-1}(\Gamma)]$, where we write $[j^{-1}(\Gamma)]$ for the substitution $[j^{-1}(x) / x]_{x: C \in \Gamma}$.
2. If $\Gamma \mid z : \underline{A} \vdash t : \underline{B}$ then $t = k_{\underline{B}}(t^{C_R C_R}) [k_{\underline{A}}^{-1}(z) / z] [j^{-1}(\Gamma)]$.

For the proof, suitable isomorphisms $j_{\underline{A}}$ and $k_{\underline{A}}$ are easily defined by induction on the structure of types. Given these, it should, in principle, be a routine unwinding of definitions to verify the equalities in items 1 and 2. However, because of the intricacy of Figure 5, the terms $t^{\mathcal{V}_R \mathcal{V}_R}$ and $t^{C_R C_R}$ are extremely unwieldy, and such a direct verification seems impractical. Instead, we shall give a more manageable (as well as more conceptually appealing) proof of the theorem using categorical model theory, in Section 5.

Theorem 2 (Full-completeness). *The linearly-used CPS translation is full and faithful, i.e.:*

1. If $\Gamma \mid - \vdash t, u : \underline{A}$ are two terms of the same type and $t^{\mathcal{V}_R} = u^{\mathcal{V}_R}$ (i.e., the equality is provable in context) then $t = u$.

$$\begin{aligned}
& \Gamma^{\mathcal{V}_R}, x: \mathbf{A}^{\mathcal{V}_R} \mid - \vdash x^{\mathcal{V}_R} = x \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{A}^{\mathcal{C}_R} \vdash z^{\mathcal{C}_R} = z \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash *^{\mathcal{V}_R} = * \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{Q} \vdash *^{\mathcal{C}_R} = \underline{\text{image}}(z) \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash \langle t, u \rangle^{\mathcal{V}_R} = \langle t^{\mathcal{V}_R}, u^{\mathcal{V}_R} \rangle \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{A}^{\mathcal{C}_R} \oplus \mathbf{B}^{\mathcal{C}_R} \vdash \langle t, u \rangle^{\mathcal{C}_R} = \underline{\text{case}} z \text{ of } (\underline{\text{inl}}(z). t^{\mathcal{C}_R}; \underline{\text{inr}}(z). u^{\mathcal{C}_R}) \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash \text{fst}(t)^{\mathcal{V}_R} = \text{fst}(t^{\mathcal{V}_R}) \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{A}^{\mathcal{C}_R} \vdash \text{fst}(t)^{\mathcal{C}_R} = t^{\mathcal{C}_R}[\underline{\text{inl}}(z)/z] \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash \text{snd}(t)^{\mathcal{V}_R} = \text{snd}(t^{\mathcal{V}_R}) \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{A}^{\mathcal{C}_R} \vdash \text{snd}(t)^{\mathcal{C}_R} = t^{\mathcal{C}_R}[\underline{\text{inr}}(z)/z] \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash (\lambda x: \mathbf{A}. t)^{\mathcal{V}_R} = \lambda x: \mathbf{A}^{\mathcal{V}_R}. t^{\mathcal{V}_R} \\
& \Gamma^{\mathcal{V}_R} \mid z: !\mathbf{A}^{\mathcal{V}_R} \otimes \mathbf{B}^{\mathcal{C}_R} \vdash (\lambda x: \mathbf{A}. t)^{\mathcal{C}_R} = \text{let } !x \otimes z \text{ be } z \text{ in } (t^{\mathcal{C}_R}) \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash (s(t))^{\mathcal{V}_R} = s^{\mathcal{V}_R}(t^{\mathcal{V}_R}) \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{B}^{\mathcal{C}_R} \vdash (s(t))^{\mathcal{C}_R} = s^{\mathcal{C}_R}[!t^{\mathcal{V}_R} \otimes z / z] \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash (!t)^{\mathcal{V}_R} = \underline{\lambda} f: \mathbf{A}^{\mathcal{V}_R} \rightarrow \mathbf{R}. f(t^{\mathcal{V}_R}) \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash (\text{let } !x \text{ be } t \text{ in } u)^{\mathcal{V}_R} = i_{\mathbf{B}}(\underline{\lambda} z: \mathbf{B}^{\mathcal{C}_R}. t^{\mathcal{V}_R}[\lambda x: \mathbf{A}^{\mathcal{V}_R}. i_{\mathbf{B}}^{-1}(u^{\mathcal{V}_R})[z]]) \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{B}^{\mathcal{C}_R} \vdash (\text{let } !x \text{ be } t \text{ in } u)^{\mathcal{C}_R} = t^{\mathcal{C}_R}[(\lambda x: \mathbf{A}^{\mathcal{V}_R}. i_{\mathbf{B}}^{-1}(u^{\mathcal{V}_R})[z]) / z] \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash (\underline{\lambda} z: \mathbf{A}. t)^{\mathcal{V}_R} = \underline{\lambda} z: \mathbf{B}^{\mathcal{C}_R}. t^{\mathcal{C}_R} \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash (s[t])^{\mathcal{V}_R} = i_{\mathbf{B}}(\underline{\lambda} z: \mathbf{B}^{\mathcal{C}_R}. i_{\mathbf{A}}^{-1}(t^{\mathcal{V}_R})[s^{\mathcal{V}_R}[z]]) \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{B}^{\mathcal{C}_R} \vdash (s[t])^{\mathcal{C}_R} = t^{\mathcal{C}_R}[s^{\mathcal{V}_R}[z] / z] \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash (!t \otimes u)^{\mathcal{V}_R} = \underline{\lambda} z: \mathbf{A}^{\mathcal{V}_R} \rightarrow \mathbf{B}^{\mathcal{C}_R}. i_{\mathbf{B}}^{-1}(u^{\mathcal{V}_R})[z(t^{\mathcal{V}_R})] \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{A}^{\mathcal{V}_R} \rightarrow \mathbf{B}^{\mathcal{C}_R} \vdash (!t \otimes u)^{\mathcal{C}_R} = u^{\mathcal{C}_R}[z(t^{\mathcal{V}_R}) / z] \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash (\text{let } !x \otimes z \text{ be } s \text{ in } t)^{\mathcal{V}_R} = i_{\mathbf{C}}(\underline{\lambda} z: \mathbf{C}^{\mathcal{C}_R}. s^{\mathcal{V}_R}[\lambda x: \mathbf{A}^{\mathcal{V}_R}. t^{\mathcal{C}_R}]) \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{C}^{\mathcal{C}_R} \vdash (\text{let } !x \otimes z \text{ be } s \text{ in } t)^{\mathcal{C}_R} = s^{\mathcal{C}_R}[(\lambda x: \mathbf{A}^{\mathcal{V}_R}. t^{\mathcal{C}_R}) / z] \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash (\underline{\text{image}}(t))^{\mathcal{V}_R} = i_{\mathbf{A}}(\underline{\lambda} z: \mathbf{A}^{\mathcal{C}_R}. t^{\mathcal{V}_R}[*]) \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{A}^{\mathcal{C}_R} \vdash (\underline{\text{image}}(t))^{\mathcal{C}_R} = t^{\mathcal{C}_R}[*/z] \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash (\underline{\text{inl}}(t))^{\mathcal{V}_R} = \underline{\lambda} z: \mathbf{A}^{\mathcal{C}_R} \times \mathbf{B}^{\mathcal{C}_R}. i_{\mathbf{A}}^{-1}(t^{\mathcal{V}_R})[\text{fst}(z)] \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{A}^{\mathcal{C}_R} \times \mathbf{B}^{\mathcal{C}_R} \vdash (\underline{\text{inl}}(t))^{\mathcal{C}_R} = t^{\mathcal{C}_R}[\text{fst}(z) / z] \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash (\underline{\text{inr}}(t))^{\mathcal{V}_R} = \underline{\lambda} z: \mathbf{A}^{\mathcal{C}_R} \times \mathbf{B}^{\mathcal{C}_R}. i_{\mathbf{A}}^{-1}(t^{\mathcal{V}_R})[\text{snd}(z)] \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{A}^{\mathcal{C}_R} \times \mathbf{B}^{\mathcal{C}_R} \vdash (\underline{\text{inr}}(t))^{\mathcal{C}_R} = t^{\mathcal{C}_R}[\text{snd}(z) / z] \\
& \Gamma^{\mathcal{V}_R} \mid - \vdash (\underline{\text{case}} s \text{ of } (\underline{\text{inl}}(x). t; \underline{\text{inr}}(y). u))^{\mathcal{V}_R} = i_{\mathbf{C}}(\underline{\lambda} z: \mathbf{C}^{\mathcal{C}_R}. s^{\mathcal{V}_R}[\langle t^{\mathcal{C}_R}, u^{\mathcal{C}_R} \rangle]) \\
& \Gamma^{\mathcal{V}_R} \mid z: \mathbf{C}^{\mathcal{C}_R} \vdash (\underline{\text{case}} s \text{ of } (\underline{\text{inl}}(x). t; \underline{\text{inr}}(y). u))^{\mathcal{C}_R} = s^{\mathcal{C}_R}[\langle t^{\mathcal{C}_R}, u^{\mathcal{C}_R} \rangle / z]
\end{aligned}$$

Fig. 5. Linearly-used CPS translation of terms.

2. If $\Gamma \mid z : \underline{\mathbf{A}} \vdash t, u : \underline{\mathbf{B}}$ and $t^{\mathcal{C}_{\mathbb{R}}} = u^{\mathcal{C}_{\mathbb{R}}}$ then $t = u$.
3. If $\Gamma^{\mathcal{V}_{\mathbb{R}}} \mid - \vdash t : \mathbf{A}^{\mathcal{V}_{\mathbb{R}}}$ then there exists $\Gamma \mid - \vdash u : \mathbf{A}$ such that $t = u^{\mathcal{V}_{\mathbb{R}}}$.
4. If $\Gamma^{\mathcal{V}_{\mathbb{R}}} \mid z : \underline{\mathbf{B}}^{\mathcal{C}_{\mathbb{R}}} \vdash t : \underline{\mathbf{A}}^{\mathcal{C}_{\mathbb{R}}}$ is a valid typing judgement, then there exists $\Gamma \mid z : \underline{\mathbf{A}} \vdash u : \underline{\mathbf{B}}$ such that $t = u^{\mathcal{C}_{\mathbb{R}}}$.

Proof. For statement 1, suppose $\Gamma \mid - \vdash t, u : \mathbf{A}$ and $t^{\mathcal{V}_{\mathbb{R}}} = u^{\mathcal{V}_{\mathbb{R}}}$. Then:

$$\begin{aligned}
t &= j_{\mathbf{A}}(t^{\mathcal{V}_{\mathbb{R}}}) [j_{\mathcal{C}}^{-1}(x) / x]_{x: \mathcal{C} \in \Gamma} && \text{(Theorem 1.1)} \\
&= j_{\mathbf{A}}(u^{\mathcal{V}_{\mathbb{R}}}) [j_{\mathcal{C}}^{-1}(x) / x]_{x: \mathcal{C} \in \Gamma} && \text{(Proposition 1)} \\
&= u && \text{(Theorem 1.1) .}
\end{aligned}$$

For statement 3, suppose $\Gamma^{\mathcal{V}_{\mathbb{R}}} \mid - \vdash t : \mathbf{A}^{\mathcal{V}_{\mathbb{R}}}$. Define $u = j_{\mathbf{A}}(t^{\mathcal{V}_{\mathbb{R}}}) [j^{-1}(\Gamma)]$. Then:

$$\begin{aligned}
u^{\mathcal{V}_{\mathbb{R}}} &= j_{\mathbf{A}^{\mathcal{V}_{\mathbb{R}}}}(u^{\mathcal{V}_{\mathbb{R}}}) [j^{-1}(\Gamma^{\mathcal{V}_{\mathbb{R}}})] && \text{(Theorem 1.1)} \\
&= j_{\mathbf{A}^{\mathcal{V}_{\mathbb{R}}}}((j_{\mathbf{A}}^{-1}(u) [j(\Gamma)])^{\mathcal{V}_{\mathbb{R}}}) [j^{-1}(\Gamma^{\mathcal{V}_{\mathbb{R}}})] && \text{(Theorem 1.1)} \\
&= j_{\mathbf{A}^{\mathcal{V}_{\mathbb{R}}}}(t^{\mathcal{V}_{\mathbb{R}}}) [j^{-1}(\Gamma^{\mathcal{V}_{\mathbb{R}}})] && \text{(Definition } u) \\
&= t && \text{(Theorem 1.1) .}
\end{aligned}$$

The proofs of items 2 and 4 are similar. \square

We end this section by showing how the linearly-used CPS translation of the enriched effect calculus into itself subsumes the call-by-value and call-by-name linearly-used CPS translations of Figure 3, from [7, 6]. Indeed, these are obtained uniformly by precomposing the translation on the enriched effect calculus with the standard call-by-value and call-by-name translations.

Theorem 3 (Recovering $(\cdot)^{\text{cbv}_{\mathbb{R}}}$ and $(\cdot)^{\text{cbn}_{\mathbb{R}}}$).

1. For every simple type σ , we have $\sigma^{\text{cbv}_{\mathbb{R}}} = (\sigma^{\text{cbv}})^{\mathcal{V}_{\mathbb{R}}}$; and, for every simply-typed term $\Theta \vdash t : \sigma$, we have $t^{\text{cbv}_{\mathbb{R}}} = (t^{\text{cbv}})^{\mathcal{V}_{\mathbb{R}}}$.
2. For every simple type σ , there is a linear isomorphism $r_{\sigma} : (\sigma^{\text{cbn}})^{\mathcal{C}_{\mathbb{R}}} \multimap \sigma^{\text{cbn}_{\mathbb{R}}}$ such that, for every $\Theta \vdash t : \sigma$, it holds that $t^{\text{cbn}_{\mathbb{R}}} = p_{\tau^{\text{cbn}}}^{-1}((t^{\text{cbn}})^{\mathcal{V}_{\mathbb{R}}})[p(\Theta)]$, where p_{σ} is the isomorphism $i_{\sigma^{\text{cbn}}} \circ (r_{\sigma} \multimap \underline{\mathbf{R}}) : (\sigma^{\text{cbn}_{\mathbb{R}}} \multimap \underline{\mathbf{R}}) \rightarrow (\sigma^{\text{cbn}})^{\mathcal{V}_{\mathbb{R}}}$.

That proofs are by induction on the structure of σ and t .

Statement 2 of the theorem is more complex than one would like because the types $(\sigma^{\text{cbn}})^{\mathcal{C}_{\mathbb{R}}}$ and $\sigma^{\text{cbn}_{\mathbb{R}}}$ are not syntactically identical. The difficulty derives from $((\sigma \rightarrow \tau)^{\text{cbn}})^{\mathcal{C}_{\mathbb{R}}} = !(\sigma^{\text{cbn}})^{\mathcal{V}_{\mathbb{R}}} \otimes (\tau^{\text{cbn}})^{\mathcal{C}_{\mathbb{R}}} \cong !((\sigma^{\text{cbn}})^{\mathcal{C}_{\mathbb{R}}} \multimap \underline{\mathbf{R}}) \otimes (\tau^{\text{cbn}})^{\mathcal{C}_{\mathbb{R}}}$. But identity does not hold because we do not have $\underline{\mathbf{A}}^{\mathcal{V}_{\mathbb{R}}} = \underline{\mathbf{A}}^{\mathcal{C}_{\mathbb{R}}} \multimap \underline{\mathbf{R}}$. This technicality could be avoided by changing to a syntax for EEC with no overloading between value and computation types, as in Levy's CBPV [10]. However, then the syntax of EEC terms would become more complex.

Hasegawa [6] proves full-completeness for the call-by-value linearly-used CPS translation of Moggi's computational λ -calculus [11] into ILL. He also has a similar result for the call-by-name translation of [7] restricted to the simply-typed λ -calculus (private communication). The corollary below adapts this to

our identical translations into EEC. In fact, our results follow from Hasegawa's (and not vice versa!). What is interesting is our method of proof, which applies the results we have established for our canonical translation. In the statement, we write λ_c for the equational theory of Moggi's computational λ -calculus.

Corollary 1 (Full completeness of $(\cdot)^{\text{cbv}_R}$ and $(\cdot)^{\text{cbn}_R}$).

1. If $\Theta \vdash t, u: \tau$ and $\Theta^{\text{cbv}_R} \mid - \vdash t^{\text{cbv}_R} = u^{\text{cbv}_R}: (\tau^{\text{cbv}_R} \rightarrow \underline{R}) \multimap \underline{R}$ then $\Theta \vdash_{\lambda_c} t = u: \tau$.
2. If $\Theta^{\text{cbv}_R} \mid - \vdash u: (\tau^{\text{cbv}_R} \rightarrow \underline{R}) \multimap \underline{R}$ then there exists a simply-typed term $\Theta \vdash t: \tau$ such that $u = t^{\text{cbv}_R}$.
3. If $\Theta \vdash t, u: \tau$ and $\Theta^{\text{cbn}_R} \mid - \vdash t^{\text{cbn}_R} = u^{\text{cbn}_R}: \tau^{\text{cbn}_R}$ then $\Theta \vdash_{\beta\eta} t = u: \tau$.
4. If $\Theta^{\text{cbn}_R} \mid - \vdash u: \tau^{\text{cbn}_R}$ then there exists a simply-typed term $\Theta \vdash t: \tau$ such that $u = t^{\text{cbn}_R}$.

Proof. We only give arguments for call-by-value. It can be shown that the mapping $(\cdot)^{\text{cbv}}$ from the computational λ -calculus into EEC is full and faithful.³ For statement 1, suppose $\Theta^{\text{cbv}_R} \mid - \vdash t^{\text{cbv}_R} = u^{\text{cbv}_R}: (\tau^{\text{cbv}_R} \rightarrow \underline{R}) \multimap \underline{R}$. Then, by Theorem 3.1, $(\Theta^{\text{cbv}})^{\mathcal{V}_R} \mid - \vdash (t^{\text{cbv}})^{\mathcal{V}_R} = (u^{\text{cbv}})^{\mathcal{V}_R}: ((\tau^{\text{cbv}})^{\mathcal{V}_R} \rightarrow \underline{R}) \multimap \underline{R}$. But we have $((\tau^{\text{cbv}})^{\mathcal{V}_R} \rightarrow \underline{R}) \multimap \underline{R} = (!\tau^{\text{cbv}})^{\mathcal{V}_R}$. So, by Theorem 2.1, $\Theta^{\text{cbv}} \mid - \vdash t^{\text{cbv}} = u^{\text{cbv}}: !\tau^{\text{cbv}}$ and statement 1 follows from the faithfulness of $(\cdot)^{\text{cbv}}$. For statement 2, by Theorem 2.3 there exists u' such that $\Theta^{\text{cbv}} \mid - \vdash u': !\tau^{\text{cbv}}$ and $(u')^{\mathcal{V}_R} = u$. Statement 2 now follows from the fullness of $(\cdot)^{\text{cbv}}$. \square

4 Models

We review the notion of *model* of the enriched effect calculus [4]. This is defined in terms of *enriched category theory* [8]. Recall that, given a chosen monoidal category \mathbf{V} , a \mathbf{V} -enriched category (or, \mathbf{V} -category) \mathbf{C} is given by a collection of objects, with, for every pair of objects $A, B \in \mathbf{C}$, a specified *hom-object* $\mathbf{C}(A, B) \in \mathbf{V}$, together with families of morphisms in \mathbf{V} supplying \mathbf{C} with its identity maps and an associative composition. When \mathbf{V} is the category of sets, a \mathbf{V} -category is just an ordinary (locally small) category. When modelling the enriched effect calculus, it is natural to ask for the category of linear maps between computation types to be enriched over the category of value types, since the value types $\underline{A} \multimap \underline{B}$ act as hom-objects. In the sequel, we shall assume some basic knowledge of enriched category theory; see [8] for a detailed account.

We shall consider enrichment only with respect to categories \mathbf{V} that are cartesian closed (we write B^A or $[A \rightarrow B]$ for functions spaces). Any such category is self-enriched. We say that a \mathbf{V} -enriched category \mathbf{C} has $(\mathbf{V}\text{-})$ *powers*⁴ if, for all objects $A \in \mathbf{V}$ and $\underline{B} \in \mathbf{C}$, there exists an object $\underline{B}^A \in \mathbf{C}$ with isomorphisms

$$\mathbf{C}(\underline{C}, \underline{B}^A) \xrightarrow{\cong} [A \rightarrow \mathbf{C}(\underline{C}, \underline{B})]$$

³ E.g., the term model of λ_c [11] fully embeds in a model of CBPV [10], and every model of CBPV fully embeds in a model of EEC [4] (the assumptions of sums [10] and cartesian closedness [4] are not needed for the embeddings).

⁴ Kelly writes *cotensors* (resp., *tensors*) where we write *powers* (resp., *copowers*).

\mathbf{V} -natural in objects \underline{C} of \mathbf{C} . The dual property is that of having $(\mathbf{V}\text{-})\text{copowers}$: for all $A \in \mathbf{V}$ and $\underline{B} \in \mathbf{C}$, there must exist an object $A \cdot \underline{B}$ of \mathbf{C} with isomorphisms

$$\mathbf{C}(A \cdot \underline{B}, \underline{C}) \rightarrow [A \rightarrow \mathbf{C}(\underline{B}, \underline{C})]$$

\mathbf{V} -natural in \underline{C} . An *enriched* adjunction $F \dashv U$ between \mathbf{V} -functors $F: \mathbf{D} \rightarrow \mathbf{C}$ and $U: \mathbf{C} \rightarrow \mathbf{D}$ requires the existence of isomorphisms in \mathbf{V}

$$\mathbf{C}(F(A), \underline{B}) \rightarrow [A \rightarrow U(\underline{B})]$$

which are \mathbf{V} -natural in A and \underline{B} .

Definition 1. An *enriched-effect-calculus model* comprises: a cartesian closed category \mathbf{V} , with \mathbf{V} -enriched finite products, coproducts, powers, copowers, and a \mathbf{V} -adjunction $F \dashv U: \mathbf{C} \rightarrow \mathbf{V}$.

We shall loosely specify models as $F \dashv U: \mathbf{C} \rightarrow \mathbf{V}$, without making the other structure (which is, in any case, determined up to isomorphism) explicit.

To interpret the enriched effect calculus in a model, value types A are interpreted as objects $\mathbf{V}[\underline{A}]$ of \mathbf{V} , and computation types \underline{A} are interpreted as pairs $(\mathbf{C}[\underline{A}], s_{\underline{A}})$ where $\mathbf{C}[\underline{A}]$ is an object of \mathbf{C} , and $s_{\underline{A}}: U(\mathbf{C}[\underline{A}]) \rightarrow \mathbf{V}[\underline{A}]$ is an isomorphism in \mathbf{V} . The reader is referred to [4] for further details.

In [4], the equational theory of the enriched effect calculus is shown to be sound and complete with respect to interpretations in models. Completeness is proved via a syntactic model construction. Since this model will play an important role in Section 5, we recall its definition.

The category \mathbf{V}_{Syn} has as objects value types and as morphisms from A to B terms of the form $x: A \mid - \vdash t: B$ identified up to the equality theory of EEC. Composition is given by substitution. The \mathbf{V}_{Syn} -enriched category \mathbf{C}_{Syn} has as objects all computation types and as object of morphisms from \underline{A} to \underline{B} the value type $\underline{A} \multimap \underline{B}$. Powers and copowers are given by the constructions $A \rightarrow \underline{B}$ and $!A \otimes \underline{B}$, respectively. The right adjoint is the forgetful functor from computation types to value types, with its action on morphisms defined by the coercion $(\underline{A} \multimap \underline{B}) \rightarrow (\underline{A} \rightarrow \underline{B})$. The left adjoint is given by $!$.

The syntactic model is characterised by a universal property. To formulate this, one needs a notion of morphism of models. Essentially, a morphism from $F \dashv U: \mathbf{C} \rightarrow \mathbf{V}$ to $F' \dashv U': \mathbf{C}' \rightarrow \mathbf{V}'$, is given by a pair of functors $S: \mathbf{V} \rightarrow \mathbf{V}'$ and $T: \mathbf{C} \rightarrow \mathbf{C}'$ (jointly) preserving the structure. There are, however, two complicating factors. First, morphisms need only preserve structure up to isomorphism rather than identity. (This choice is both mathematically natural and essential to the results of Section 5.) Second, \mathbf{C} and \mathbf{C}' are enriched over two different categories, which leads to subtle requirements regarding how the functor T enriches. These issues are treated in [4]. There is a notion of coherent natural isomorphism (henceforth, *cni*) between morphisms of models; hence, the category of models is naturally enriched in that of groupoids, **Grpd**.

Theorem 4 ([4, Theorem 3]). *Given an enriched-effect-calculus model $F \dashv U: \mathbf{C} \rightarrow \mathbf{V}$ and families of objects, $\mathbf{V}[\underline{\alpha}]$ in \mathbf{V} and $\mathbf{C}[\underline{\alpha}]$ in \mathbf{C} , indexed by type*

constants, there exists a morphism of models from the syntactic model $F_{\text{Syn}} \dashv U_{\text{Syn}}: \mathbf{C}_{\text{Syn}} \rightarrow \mathbf{V}_{\text{Syn}}$ to $F \dashv U: \mathbf{C} \rightarrow \mathbf{V}$ that extends the given interpretation of type constants, and this morphism is unique up to cni.

We remark, that the proof of the theorem produces a morphism that preserves type constants up to equality. Nevertheless, uniqueness (up to isomorphism) holds relative to the wider class of morphisms that only preserve type constants up to isomorphism.

5 Dual models

Given a model $F \dashv U: \mathbf{C} \rightarrow \mathbf{V}$ of the enriched effect calculus and an object $\underline{\mathbf{R}}$ of \mathbf{C} , one can define another model $F^{\underline{\mathbf{R}}} \dashv U^{\underline{\mathbf{R}}}: \mathbf{C}^{\text{op}} \rightarrow \mathbf{V}$ by:

$$F^{\underline{\mathbf{R}}} := \underline{\mathbf{R}}^{(-)} \qquad U^{\underline{\mathbf{R}}} := \mathbf{C}(-, \underline{\mathbf{R}}) .$$

Indeed, \mathbf{C}^{op} is trivially \mathbf{V} -enriched. Its enriched powers and copowers are given by copowers and powers in \mathbf{C} , respectively; similarly also products and coproducts. And it is standard that $\underline{\mathbf{R}}^{(-)} \dashv \mathbf{C}(-, \underline{\mathbf{R}})$ is an enriched adjunction. We call the constructed model the $\underline{\mathbf{R}}$ -dual of $F \dashv U: \mathbf{C} \rightarrow \mathbf{V}$.

There are obvious similarities between the dual model construction and the linearly-used CPS translation on enriched effect calculus types, defined in Fig. 4. For example, $F^{\underline{\mathbf{R}}}$ corresponds to the $(-) \rightarrow \underline{\mathbf{R}}$ action of the translation of $!(-)$ to computation types. Similarly, $U^{\underline{\mathbf{R}}}$ implements the feature that the translation, $\underline{\mathbf{A}}^{\mathbf{V}_{\underline{\mathbf{R}}}}$, of $\underline{\mathbf{A}}$ as a value type is obtained (modulo the isomorphism $i_{\underline{\mathbf{A}}}$) as $\underline{\mathbf{A}}^{\mathbf{C}_{\underline{\mathbf{R}}}} \multimap \underline{\mathbf{R}}$, where $\underline{\mathbf{A}}^{\mathbf{C}_{\underline{\mathbf{R}}}}$ is the interpretation of $\underline{\mathbf{A}}$ as a computation type. Furthermore, the induced monad $\mathbf{C}(\underline{\mathbf{R}}^{(-)}, \underline{\mathbf{R}})$ on \mathbf{V} corresponds to the linearly-used continuations monad $((-) \rightarrow \underline{\mathbf{R}}) \multimap \underline{\mathbf{R}}$.

Monads of the form $\mathbf{C}(\underline{\mathbf{R}}^{(-)}, \underline{\mathbf{R}})$ have been called *dual monads* by Lawvere [9], who considers them in the case that $U: \mathbf{C} \rightarrow \mathbf{V}$ exhibits \mathbf{C} as a category of algebras for a monad over its enriching category. Our *dual model* construction performs the analogous operation on general enriched adjunctions, rather than on monads. As we shall see below (Theorem 5), in our setting, the “dual” terminology is particularly appropriate.

Because of the choice of object $\underline{\mathbf{R}}$, the natural context for considering the dual model construction is as an operation on *pointed models*, $(F \dashv U: \mathbf{C} \rightarrow \mathbf{V}, \underline{\mathbf{R}})$, where $\underline{\mathbf{R}}$ is a chosen object of \mathbf{C} . A morphism of pointed models is a morphism of models (S, T) together with an isomorphism $T\underline{\mathbf{R}} \rightarrow \underline{\mathbf{R}}'$. For a pointed model $\mathcal{M} = (F \dashv U: \mathbf{C} \rightarrow \mathbf{V}, \underline{\mathbf{R}})$, the *dual pointed model* \mathcal{M}^{\perp} is defined to be $(F^{\underline{\mathbf{R}}} \dashv U^{\underline{\mathbf{R}}}: \mathbf{C}^{\text{op}} \rightarrow \mathbf{V}, F1)$. The choice of $F1$ may seem arbitrary here, but it is crucial to Theorem 5 below.

Proposition 2. *The dual model construction is a \mathbf{Grpd} -enriched functor on the \mathbf{Grpd} -category of pointed models.*

Proof (outline). A morphism (S, T) is mapped to $(S, T^{\text{op}}: \mathbf{C}^{\text{op}} \rightarrow (\mathbf{C}')^{\text{op}})$, and a cni $(\alpha: S \Rightarrow S', \beta: T \Rightarrow T')$ is mapped to $(\alpha, \beta^{-1}: T^{\text{op}} \Rightarrow (T')^{\text{op}})$. \square

Theorem 5. *For every pointed model \mathcal{M} , we have a natural isomorphism of pointed models between \mathcal{M} and $\mathcal{M}^{\perp\perp}$.*

Proof (outline). We show that the pair of identity functors is the required isomorphism from $\mathcal{M} \rightarrow \mathcal{M}^{\perp\perp}$. The double dual $\mathcal{M}^{\perp\perp}$ turns out to be $(F^{\perp\perp} \dashv U^{\perp\perp} : \mathbf{C} \rightarrow \mathbf{V}, F^{\mathbf{R}}1)$, where $F^{\perp\perp} = (-) \cdot F1$ (this is the power $(F1)^{(-)}$ calculated in \mathbf{C}^{op}) and $U^{\perp\perp} = \mathbf{C}^{\text{op}}(-, F1)$. Then we calculate: $F^{\perp\perp} = (-) \cdot F1 \cong F$, similarly $U^{\perp\perp} = \mathbf{C}^{\text{op}}(-, F1) = \mathbf{C}(F1, -) \cong U$, and also $F^{\mathbf{R}}1 = \underline{\mathbf{R}}^1 \cong \underline{\mathbf{R}}$. \square

Models of intuitionistic linear logic supply a natural collection of models for the enriched effect calculus. A *linear/nonlinear model* [2] consists of a cartesian-closed category \mathbf{V} (the *intuitionistic category*), a symmetric monoidal closed category \mathbf{C} (the *linear category*), and a symmetric monoidal adjunction $F \dashv G : \mathbf{C} \rightarrow \mathbf{V}$. The model has *additives* if \mathbf{C} has finite products and coproducts. It is *classical* if \mathbf{C} is $*$ -autonomous [1]. In [4, Proposition 1], it is shown that every linear/nonlinear model with additives is a model of the enriched effect calculus. Given an object $\underline{\mathbf{R}}$ of \mathbf{C} , the dual model is thus a model of the enriched effect calculus. However, it is not (in general) a linear/nonlinear model (e.g., \mathbf{C}^{op} need not be symmetric monoidal closed). Thus models of the enriched effect calculus are closed under a natural construction which is not available for models of intuitionistic linear logic. However, models of classical linear logic are preserved by the dual model construction, when $\underline{\mathbf{R}}$ is chosen to be the dualizing object. In fact, unsurprisingly, such models are self-dual.

Proposition 3. *If $F \dashv G : \mathbf{C} \rightarrow \mathbf{V}$ is a model of classical linear logic with additives then, defining the pointed model $\mathcal{M} = (F \dashv G : \mathbf{C} \rightarrow \mathbf{V}, \perp)$, where \perp is the dualizing object, it holds that the \mathcal{M} is (pointed) isomorphic to \mathcal{M}^{\perp} .*

Proof (outline). The isomorphism is given by $(Id : \mathbf{V} \rightarrow \mathbf{V}, (-)^* : \mathbf{C} \rightarrow \mathbf{C}^{\text{op}})$ from \mathcal{M} to \mathcal{M}^{\perp} , where $(-)^*$ is the $*$ -autonomous dualizing functor. \square

We next exhibit a more surprising example of self-duality, the syntactic model. As in Section 3, we assume a computation-type constant $\underline{\mathbf{R}}$. We then consider the syntactic model together with $\underline{\mathbf{R}}$ as the chosen object of \mathbf{C}_{Syn} :

$$\mathcal{M}_{\text{Syn}, \underline{\mathbf{R}}} := (F_{\text{Syn}} \dashv U_{\text{Syn}} : \mathbf{C}_{\text{Syn}} \rightarrow \mathbf{V}_{\text{Syn}}, \underline{\mathbf{R}}) .$$

By Theorem 4, there is a unique morphism (up to cni) from the model $F_{\text{Syn}} \dashv U_{\text{Syn}} : \mathbf{C}_{\text{Syn}} \rightarrow \mathbf{V}_{\text{Syn}}$ to the model $F_{\text{Syn}}^{\underline{\mathbf{R}}} \dashv U_{\text{Syn}}^{\underline{\mathbf{R}}} : \mathbf{C}_{\text{Syn}}^{\text{op}} \rightarrow \mathbf{V}_{\text{Syn}}$ given by functors $S_{\underline{\mathbf{R}}} : \mathbf{V}_{\text{Syn}} \rightarrow \mathbf{V}_{\text{Syn}}$ and $T_{\underline{\mathbf{R}}} : \mathbf{C}_{\text{Syn}} \rightarrow \mathbf{C}_{\text{Syn}}^{\text{op}}$ satisfying:

$$S_{\underline{\mathbf{R}}}(\alpha) \cong \alpha \qquad T_{\underline{\mathbf{R}}}(\underline{\alpha}) \cong \begin{cases} !1 & \text{if } \underline{\alpha} = \underline{\mathbf{R}} \\ \underline{\alpha} & \text{otherwise} \end{cases}$$

Obviously, this is a morphism of pointed models from $\mathcal{M}_{\text{Syn}, \underline{\mathbf{R}}}$ to $(\mathcal{M}_{\text{Syn}, \underline{\mathbf{R}}})^{\perp}$.

Theorem 6. *The morphism $(S_{\underline{\mathbf{R}}}, T_{\underline{\mathbf{R}}})$ is an equivalence of pointed models between $\mathcal{M}_{\text{Syn}, \underline{\mathbf{R}}}$ and $(\mathcal{M}_{\text{Syn}, \underline{\mathbf{R}}})^{\perp}$.*

Proof (outline). By Proposition 2, $(S_{\underline{R}}, (T_{\underline{R}})^{\text{op}})$ is a morphism from $(\mathcal{M}_{\text{Syn}, \underline{R}})^{\perp}$ to $(\mathcal{M}_{\text{Syn}, \underline{R}})^{\perp\perp}$ and, by (the proof of) Theorem 5, (Id, Id) is a morphism from $(\mathcal{M}_{\text{Syn}, \underline{R}})^{\perp\perp}$ to $\mathcal{M}_{\text{Syn}, \underline{R}}$. The composite endomorphism $(S_{\underline{R}} S_{\underline{R}}, (T_{\underline{R}})^{\text{op}} T_{\underline{R}})$ on $\mathcal{M}_{\text{Syn}, \underline{R}}$ obviously maps value type constants α and computation type constants α , other than \underline{R} , to themselves. In the case of \underline{R} , we have

$$(T_{\underline{R}})^{\text{op}} T_{\underline{R}}(\underline{R}) \cong (T_{\underline{R}})^{\text{op}}(!1) \cong (T_{\underline{R}})(!1) \cong 1 \rightarrow \underline{R} \cong \underline{R} ,$$

where the penultimate isomorphism is because $T_{\underline{R}}$ maps the left adjoint $!(-)$ in $\mathcal{M}_{\text{Syn}, \underline{R}}$ to the left adjoint $(-) \rightarrow \underline{R}$ in $(\mathcal{M}_{\text{Syn}, \underline{R}})^{\perp}$. Thus the endomorphism $(S_{\underline{R}} S_{\underline{R}}, (T_{\underline{R}})^{\text{op}} T_{\underline{R}})$ on $\mathcal{M}_{\text{Syn}, \underline{R}}$ preserves all type constants up to isomorphism. By Theorem 4, this endomorphism is isomorphic to the identity morphism. For the composite endomorphism $(S_{\underline{R}} S_{\underline{R}}, T_{\underline{R}} (T_{\underline{R}})^{\text{op}})$ on $(\mathcal{M}_{\text{Syn}, \underline{R}})^{\perp}$, we have $T_{\underline{R}} (T_{\underline{R}})^{\text{op}} = ((T_{\underline{R}})^{\text{op}} T_{\underline{R}})^{\text{op}}$, which is again isomorphic to the identity because $(T_{\underline{R}})^{\text{op}} T_{\underline{R}}$ is (as shown above). Hence $(S_{\underline{R}}, T_{\underline{R}}): \mathcal{M}_{\text{Syn}, \underline{R}} \rightarrow (\mathcal{M}_{\text{Syn}, \underline{R}})^{\perp}$ and $(S_{\underline{R}}, (T_{\underline{R}})^{\text{op}}): (\mathcal{M}_{\text{Syn}, \underline{R}})^{\perp} \rightarrow \mathcal{M}_{\text{Syn}, \underline{R}}$ together form an equivalence of models. \square

The self-duality of $\mathcal{M}_{\text{Syn}, \underline{R}}$ exhibited above differs in two important respects from the self-duality for classical linear/nonlinear models of Proposition 3. First, $(S_{\underline{R}}, T_{\underline{R}})$ is not an isomorphism. Second, the functor $S_{\underline{R}}$ is not (even isomorphic to) the identity on \mathbf{V}_{Syn} . (Though $S_{\underline{R}} S_{\underline{R}}$ is isomorphic to the identity.)

It is now a straightforward matter to finally prove Theorem 1. The crucial observation is that the morphism $(S_{\underline{R}}, T_{\underline{R}})$ is the generic linearly-used CPS translation of the enriched effect calculus into itself. Indeed, the object actions of $S_{\underline{R}}$ and $T_{\underline{R}}$ are respectively the value-type and computation-type translations of Fig. 4, and the morphism actions are respectively $(\cdot)^{\mathcal{V}_{\underline{R}}}$ and $(\cdot)^{\mathcal{C}_{\underline{R}}}$ from Fig. 5.

Proof of Theorem 1 (outline). We have seen above that $S_{\underline{R}} S_{\underline{R}}$ is isomorphic to the identity on \mathbf{V}_{Syn} , and $(T_{\underline{R}})^{\text{op}} T_{\underline{R}}$ is isomorphic to the identity on \mathbf{C}_{Syn} . We define the $j_{\underline{A}}: \underline{A}^{\mathcal{V}_{\underline{R}} \mathcal{V}_{\underline{R}}} \rightarrow \underline{A}$ to be components of the former natural isomorphism, and the $k_{\underline{A}}: \underline{A}^{\mathcal{C}_{\underline{R}} \mathcal{C}_{\underline{R}}} \rightarrow \underline{A}$ to be components of the latter. Statement 1 of the theorem is equivalent to the naturality of the $j_{\underline{A}}$. The naturality condition which the $k_{\underline{A}}$ are required to satisfy to be part of a cni, is the diagram below.

$$\begin{array}{ccc} (\underline{A} \multimap \underline{B})^{\mathcal{V}_{\underline{R}} \mathcal{V}_{\underline{R}}} & \xrightarrow{j_{\underline{A} \multimap \underline{B}}} & \underline{A} \multimap \underline{B} \\ \downarrow = & & \downarrow k_{\underline{A} \multimap \underline{B}} \\ \underline{A}^{\mathcal{C}_{\underline{R}} \mathcal{C}_{\underline{R}}} \multimap \underline{B}^{\mathcal{C}_{\underline{R}} \mathcal{C}_{\underline{R}}} & \xrightarrow{\underline{A}^{\mathcal{C}_{\underline{R}} \mathcal{C}_{\underline{R}}} \multimap k_{\underline{B}}} & \underline{A}^{\mathcal{C}_{\underline{R}} \mathcal{C}_{\underline{R}}} \multimap \underline{B} \end{array}$$

Hence, if $\Gamma \mid z: \underline{A} \vdash t: \underline{B}$ then

$$\begin{aligned} \lambda z: \underline{A}. t &= j_{\underline{A} \multimap \underline{B}}(\lambda z: \underline{A}. t)^{\mathcal{V}_{\underline{R}} \mathcal{V}_{\underline{R}}} [j^{-1}(\Gamma)] && \text{(statement 1)} \\ &= j_{\underline{A} \multimap \underline{B}}(\lambda z: \underline{A}^{\mathcal{C}_{\underline{R}} \mathcal{C}_{\underline{R}}}. (t^{\mathcal{C}_{\underline{R}} \mathcal{C}_{\underline{R}}})) [j^{-1}(\Gamma)] && \text{(def. } (\cdot)^{\mathcal{V}_{\underline{R}}}) \\ &= (\lambda z: \underline{A}. k_{\underline{B}}(t^{\mathcal{C}_{\underline{R}} \mathcal{C}_{\underline{R}}}) [k_{\underline{A}}^{-1}(z) / z]) [j^{-1}(\Gamma)] && \text{(naturality } k) \end{aligned}$$

proving statement 2 of the theorem. \square

We end the paper by mentioning a couple of issues that there is no space to cover in detail. We call a pointed model $\mathcal{M} = (F \dashv U: \mathbf{C} \rightarrow \mathbf{V}, \underline{R})$ *canonically pointed* if \underline{R} is isomorphic to $F1$. Because we have $F^{\underline{R}}1 = \underline{R}^1 \cong \underline{R}$, we see that \mathcal{M}^\perp is canonically pointed if and only if \mathcal{M} is. We say that $F \dashv U: \mathbf{C} \rightarrow \mathbf{V}$ is *canonically self dual* if the canonically pointed $\mathcal{M} = (F \dashv U: \mathbf{C} \rightarrow \mathbf{V}, F1)$ is self dual. By an argument similar to the proof of Theorem 6, the syntactic model can be shown to be canonically self dual. This has syntactic repercussions for EEC. If the computation-type constant \underline{R} is replaced uniformly by the computation type $!1$ in the canonical linearly-used CPS translation then one still obtains the involutivity property and its consequences.

Finally, we mention that the key to Theorem 5 is that every \mathbf{V} -enriched adjunction $F \dashv U: \mathbf{C} \rightarrow \mathbf{V}$ is isomorphic to a \mathbf{V} -adjunction of the standard form $(-)\cdot \underline{I} \dashv \mathbf{C}(\underline{I}, -)$, by setting $\underline{I} \cong F1$. This fact allows models of the enriched effect calculus to be described more simply as triples $(\mathbf{V}, \mathbf{C}, \underline{I})$, where the adjunction is replaced by simply requiring a specified object \underline{I} of \mathbf{C} . This approach renders many properties of dual models obvious. What becomes less straightforward is to connect semantic and syntactic properties, since the semantic structure is less close to the syntactic primitives.

Acknowledgement. We thank the anonymous referees for helpful suggestions.

References

1. M. Barr. **-autonomous categories*. Springer LNM 752, 1979.
2. P.N. Benton. A mixed linear and non-linear logic: Proofs, terms and models. *Proc. CSL 1994*, Springer LNCS 933, 1995.
3. J. Berdine, P.W. O’Hearn, U. Reddy, H. Thielecke. Linear continuation-passing. *Higher Order and Symbolic Computation*, 15:181–208, 2002.
4. J. Egger, R.E. Møgelberg and A. Simpson. Enriching an effect calculus with linear types. In *Proc. CSL*, Springer LNCS 5771, pp. 240–254, 2009.
5. A. Filinski. *Controlling Effects*. PhD thesis, School of Comp. Sci., CMU, 1996.
6. M. Hasegawa. Linearly used effects: Monadic and CPS transformations into the linear lambda calculus. In *Proc. 6th FLOPS*. Springer LNCS 2441, pp. 167–182. 2002.
7. M. Hasegawa. Semantics of linear continuation-passing in call-by-name. In *Proc. 7th FLOPS*. Springer LNCS 2998, pp. 229–243, 2004.
8. G.M. Kelly. *Basic Concepts of Enriched Category Theory*, volume 64 of *London Math. Society Lecture Note Series*. Cambridge University Press, 1982.
9. F.W. Lawvere. Ordinal sums and equational doctrines. *Seminar on Triples and Categorical Homology Theory (ETH, Zürich)*, pp. 141–155. Springer, 1969.
10. P.B. Levy. *Call-by-push-value. A functional/imperative synthesis*. Semantic Structures in Computation, Springer, 2004.
11. E. Moggi. Computational lambda-calculus and monads. In *Proc. 4th LICS*, pp. 14–23, 1989.
12. E. Moggi. Notions of computation and monads. *Information and Computation*, 93:55–92, 1991.
13. M. Parigot. $\lambda\mu$ -calculus: an algorithmic interpretation of classical natural deduction. In *Proc. LPAR*. Springer LNCS 624, pp.190–201, 1992.