Linear Realizability

Naohiko Hoshino

RIMS, Kyoto University naophiko@kurims.kyoto-u.ac.jp

Abstract. We define a notion of relational linear combinatory algebra (rLCA) which is a generalization of a linear combinatory algebra defined by Abramsky, Haghverdi and Scott. We also define a category of assemblies as well as a category of modest sets which are realized by rLCA. This is a linear style of realizability in a way that duplicating and discarding of realizers is allowed in a controlled way. Both categories form linear-non-linear models and their coKleisli categories have a natural number object. We construct some examples of rLCA's which have some relations to well known PCA's.

1 Introduction

A category of realizability with respect to a partial combinatory algebra (PCA) A, for example assemblies $\mathbf{Ass}(A)$ or modest sets $\mathbf{Mod}(A)$, is a category of sets and functions that are implemented by the calculating system A. For computer science, these categories are models of the (second order) lambda calculus and PCF [Jac99],[Lon95]. Moreover, a realizability model provides a strong normalization proof of the second order lambda calculus [HO93]. In this paper we develop a linear variant of realizability by using another algebra in place of PCA, anticipating models of (2nd order) linear lambda calculus.

Properties of these categories, for example being cartesian closed categories (CCC), mainly come from the combinatory completeness of PCA. The combinatory completeness is informally stated as "for any lambda term t, there is an element which works like t", which allows arbitrary copying and discarding of terms. For our purpose of giving a linear variant of realizability, we should consider an algebra in which we can restrict this copying or discarding.

First, such an algebra should at least have elements b,i which satisfy $bxyz = x(yz)$ and $ix = x$ since when we use this algebra as a realizer of a category of assemblies, this algebra must realize identity and composition of two realizers of some functions.

An algebra which is called BCI algebra if it is applied with the above two elements and also an element c satisfying $cxyz = (xz)y$ is called BCI algebra, has combinatory completeness for the untyped pure linear lambda calculus. In the untyped pure linear lambda calculus, no terms are copied or discarded and in fact a category of assemblies of the untyped pure linear lambda calculus forms a symmetric monoidal closed category (SMCC). So a category of assemblies of a BCI algebra is a model of the typed pure linear lambda calculus.

The pure linear lambda calculus is too simple, however. We seek for a category of assemblies that is a model of the typed linear lambda calculus which has an exponential comonad !, so we should add some structures corresponding to the exponential comonad to BCI algebras. One algebra that has such a structure is *linear combinatory algebra(LCA)* in [AHS02]. This algebra is a BCI algebra which has an operator ! and some elements which work like natural transformations for the exponential comonad ! of a linear category [Bie95]. LCA's have good relations to SK algebras and there are many important examples of LCA's. For example, Abramsky and Lenisa used an LCA constructed from a set of partial involutions to show that the PER model of this algebra is fully complete w.r.t the fragment of System F consisting of ML-types [AL05].

In [Lon95], Longley introduced applicative morphisms, which are morphisms between applicative structures. Using applicative morphisms, we define relational LCA (rLCA) which is a generalization of LCA. Although the notion of rLCA seems slightly strange as a combinatory algebra, the definition of rLCA seems suitable because there are examples of rLCA's that have an adjoint pair (defined in Def 10) between important PCA's (cf. Sect 5.4, Sect 5.5) and no LCA has an adjoint pair between them (cf. Proposition 11).

The structure of this paper is as follows. In the Sect. 2, we recall the notions of combinatory algebras, applicative morphisms and categories of assemblies and modest sets. In Sect. 3 we show assemblies and modest sets realized by a BCI algebra form SMCC's. In Sect. 4 we show assemblies and modest sets realized by an rLCA form adjoint models. Sect. 5 is for some examples of rLCA's.

2 Background

2.1 Combinatory algebras

In this section we recall some notions such as partial combinatory algebra, BCI algebra and their combinatory completeness.

Definition 1. *Let* $(A, \cdot)$ *be a pair of a set* A *and a partial binary application* $\cdot : A \times A \rightharpoonup A$. $(A, \cdot)$ *is a* partial combinatory algebra *(PCA) if* A *has elements* s, k *which satisfy:*

$$s \cdot x \downarrow, \;\; s \cdot x \cdot y \downarrow, \;\; s \cdot x \cdot y \cdot z \simeq (x \cdot z) \cdot (y \cdot z)$$
$$k \cdot x \downarrow, \;\; k \cdot x \cdot y \simeq x$$

where $x \cdot y \downarrow$ *means that the value of* $x \cdot y$ *is defined and* $\simeq$ *means that if one side of the equation is defined then the other side is also defined and are equal.*

If the application of a PCA is total then we call this PCA an SK algebra.

Definition 2. *Let* $(A, \cdot)$ *be a pair of a set* A *and a total binary application* $\cdot : A \times A \to A$. $(A, \cdot)$ *is a* BCI algebra *when* A *has elements* b, c, i *satisfying:*

$$b \cdot x \cdot y \cdot z = x \cdot (y \cdot z) \qquad c \cdot x \cdot y \cdot z = (x \cdot z) \cdot y \qquad i \cdot x = x$$

for all $x, y, z \in A$.

In the following we write b, c, i for elements of a BCI algebra as above and s, k for such elements of an SK algebra.

Let A be a PCA or a BCI algebra, a *polynomial* over A is a syntactic expression generated by variables, elements of A and applications. The applicative structure of A induces an evident denotational relation between polynomials and elements of A.

Proposition 1. *Let $(A, \cdot)$ be a PCA and M be a polynomial over A. Then there exists a polynomial $\lambda^* x.M$ whose free variables are just those of M excluding x, which is defined as a polynomial over A and $(\lambda^* x.M)a \simeq M[a/x]$ for all $a \in A$.*

Proposition 2. *Let $(A, \cdot)$ be a BCI algebra and M be a polynomial over A in which x appears exactly once. Then there exists a polynomial $\lambda^* x.M$ whose free variables are just those of M excluding x such that $(\lambda^* x.M)a = M[a/x]$.*

In the following, for elements $p, q, r, \cdots, s$ of a BCI algebra or a PCA, we write $p \cdot q \cdot r \cdots s$ for $(\cdots((p \cdot q) \cdot r) \cdots s)$, and we write $[p, q]$ for $\lambda^* x.xpq$, and

$$\texttt{let } [u, v] = x \texttt{ in } r(u, v)$$

for $x(\lambda^* u.\lambda^* v.r(u, v))$ where $r(u, v)$ is a polynomial containing u,v as its free variables and having exactly one occurrence of each u and v. From these definitions, we have

$$\texttt{let } [u, v] = [p, q] \texttt{ in } r(u, v) = r(p, q) \ .$$

Remark 1. Although it is natural to define "partial BCI algebra" which may be defined in a similar way of PCA, we consider only total ones in this paper: we can construct a total BCI algebra from a partial one by adding $\bot$ and define a new application by

$$a \bullet b = \begin{cases} a \cdot b & a \text{ and } b \text{ are not } \bot \text{ and } a \cdot b \text{ is defined} \\ \bot & \text{else} \end{cases}$$

and a category of assemblies, defined in Sect 2.3, realized by the partial BCI algebra is a full subcategory of a category of assemblies realized by the new total BCI algebra.

2.2 Applicative morphisms

In [Lon95], applicative morphisms and a preorder between them are defined for PCA's. Here, the same definitions of applicative morphisms and preorder are given for BCI algebras and PCA's.

Definition 3. *Let A,B be BCI algebras or PCA's. An* applicative morphism γ *from A to B is a total relation from A to B such that :*

$$\exists r \in B, \forall p, q \in A, pq \textit{ is defined } \Rightarrow \forall s \in \gamma(p), t \in \gamma(q)\ rst \in \gamma(pq) \ .$$

If the domain A is a BCI algebra, "pq is defined " is not necessary since BCI algebra is total. This r is called a realizer for γ and we say γ is realized by r.

We say an applicative morphism γ is *functional* when $\gamma(p)$ is always a singleton and we say an adjoint pair $\delta \dashv \gamma$ is functional when both δ and γ are functional. If γ is functional and $\gamma(p) = \{q\}$, we write $\gamma(p) = q$ and $\gamma(p)$ for q in equations.

Definition 4. *Let A,B be BCI algebras or PCA's and $\gamma, \delta : A \to B$ be applicative morphisms. We write $\gamma \preceq \delta$ if there exists $r \in B$ such that:*

$$\forall p \in A, q \in \gamma(p).\ rq \in \delta(p)\ .$$

We say that two applicative morphisms γ, δ are equivalent if $\gamma \preceq \delta$ and $\delta \preceq \gamma$. It is proved in [Lon95] that PCA's, applicative morphisms and preorder between them form a preorder enriched category. This is also the case for BCI algebras. We have to check that the constructions of realizers in the proof of [Lon95] can also be carried out for BCI algebras. In fact all "lambda term"s in the proof are linear. Notice that an identity on a BCI algebra or a PCA A is $\{(a,a)|a \in A\}$.

Proposition 3. *BCI algebras, PCA's and applicative morphisms and preorder between them form a preorder enriched category.*

2.3 Assemblies, Modest sets

Categories of *assemblies* and *modest sets* are defined for PCA's in [Lon95]. We can also define these categories for BCI algebras.

Definition 5. *Let A be a BCI algebra. The category of assemblies* $\mathbf{Ass}(A)$ *consists of*

- *objects :* $(X, ||-||_X)$ *where X is a set and* $||-||_X$ *is a map from X to a set of non empty subsets of A.*
- *arrows :* $f : (X, ||-||_X) \to (Y, ||-||_Y)$ *where f is a realizable map from X to Y .*

Here a realizable map is a map such that there exists $r \in A$ which satisfies for all $a \in ||x||_X$, $ra \in ||fx||_Y$. We say r is a realizer of f or f is realized by r.

Modest sets $\mathbf{Mod}(A)$ is a full subcategory of $\mathbf{Ass}(A)$ whose object $(X, ||-||_X)$ satisfies:

$$x \neq y \Rightarrow ||x||_X \bigcap ||y||_X = \phi\ .$$

We sometimes omit $||-||_X$ and write X for an object of assemblies or modest sets. We write $|X|$ for the underlying set of X, and we write $|f|$ for the underlying map of f.

2.4 Categorical models of intuitionistic linear logic

We recall some models of intuitionistic linear logic.

Definition 6. [Bie95] *A* linear category *is an SMCC with a monoidal comonad* $(!, \epsilon, \delta, m_{A,B}, m_I)$ *such that*

- *There are two distinguished monoidal natural transformations with components* $e_A :!A \to I$ *and* $d_A :!A \to !A \otimes !A$ *for every free* !*-coalgebra* $(!A, \delta_A)$ *which form a commutative comonoid and are coalgebra morphisms,*
- *Whenever* $f : (!A, \delta_A) \to (!B, \delta_B)$ *is a coalgebra morphism between free coalgebras, then it is also a comonoidal morphism.*

Definition 7. [Ben94] *A* linear-non-linear model *is a symmetric monoidal adjunction* $F \dashv G : \mathcal{L} \to \mathcal{C}$ *where* $\mathcal{C}$ *is a CCC and* $\mathcal{L}$ *is an SMCC.*

3 Realizability of BCI algebras

3.1 Assemblies and modest sets realized by BCI algebras

Proposition 4. *Let* A *be a BCI algebra, then* $\mathbf{Ass}(A)$ *is a symmetric monoidal closed category.*

Proof. (outline) We may take for objects X,Y in $\mathbf{Ass}(A)$,

- $|I| = \{*\}$ and $||*||_I = \{i\}$
- $|X \otimes Y| = |X| \times |Y|$ and $||(x, y)||_{X \otimes Y} = \{[p, q] | p \in ||x||_X, q \in ||y||_Y\}$
- $|X \multimap Y| = \{f : |X| \to |Y| \; | f$ is realizable. $\}$ and $||f||_{X \multimap Y} = \{r | r$ realize $f\}$.

For example, an isomorphism $\rho_X : X \otimes I \to X$ is $(x, *) \mapsto x$ realized by $\lambda^* x.\mathtt{let}\ [p, q] = x\ \mathtt{in}\ qp$ and an evaluation $ev_{X,Y} : (X \multimap Y) \otimes X \to Y$ is $(f, x) \mapsto fx$ realized by $\lambda^* x.\mathtt{let}\ [p, q] = x\ \mathtt{in}\ pq$ □

Proposition 5. *Inclusion functor* $J : \mathbf{Mod}(A) \to \mathbf{Ass}(A)$ *has a left adjoint* Δ*. Hence* $\mathbf{Mod}(A)$ *is a reflective full subcategory of* $\mathbf{Ass}(A)$*.*

Proof. Let X be an object of $\mathbf{Mod}(A)$. We define $\simeq$ as a transitive closure of $x \sim y$ where $x \sim y$ iff $||x||_X \bigcap ||y||_X \neq \phi$ and $|\Delta X| = |X|/\simeq$, $||[x]||_{\Delta X} = \bigcup_{y \simeq x} ||y||_X$ where $[x]$ is an equivalence class of x. If f is a morphism of $X \to Y$ then $|\Delta f|([x]) = [fx]$ realized by a realizer r of f. If $a \in ||x|| \bigcap ||y||$ then $ra \in ||fx|| \bigcap ||fy||$ hence this is well defined. Let $\eta : 1 \to J\Delta$ be $\eta_X(x) = [x]$ and $\epsilon : \Delta J \to 1$ be the identity; both are realized by i. These natural transformations form unit and counit of the adjunction. □

Lemma 1. *Let* X *be an object in* $\mathbf{Mod}(A)$ *and* Y *be an object in* $\mathbf{Ass}(A)$*. Then* $\eta_{Y \multimap JX} : Y \multimap JX \to J\Delta(Y \multimap JX)$ *is an isomorphism.*

Proof. We show $Y \multimap JX$ is a modest set. Let $f, g \in |Y \multimap JX|$. If $r \in ||f|| \bigcap ||g||$ then for any $a \in ||y||_Y$, $ra \in ||fx||_{JX}$ and $ra \in ||gx||_{JX}$. Since X is a modest set, this implies $fx = gx$. Hence if $f \neq g$ then $||f||_{Y \multimap JX} \bigcap ||g||_{Y \multimap JX} = \phi$. □

In general, if a reflective full subcategory $\Delta \dashv J : \mathcal{C} \to \mathcal{D}$ of an SMCC $\mathcal{D}$ satisfies that for any object X of $\mathcal{C}$ and Y of $\mathcal{D}$, $\eta_{Y \multimap JX} : Y \multimap JX \simeq J\Delta(Y \multimap JX)$ then $\mathcal{C}$ forms an SMCC, whose monoidal product is $\Delta(JX \otimes JY)$, the unit is ΔI, the exponential is $\Delta(JX \multimap JY)$ and $\Delta \dashv J$ is a monoidal adjunction.

Therefore $\mathbf{Mod}(A)$ is an SMCC and J,Δ are monoidal functors. Moreover if $\mathbf{Ass}(A)$ has (co)limits then $\mathbf{Mod}(A)$ also has (co)limits.

3.2 Functors from applicative morphisms

In this section, we use A and B for a BCI algebra or a PCA. If γ is an applicative morphism from A to B, we can construct a functor $\gamma_* : \mathbf{Ass}(A) \to \mathbf{Ass}(B)$. This definition is the same as the one in [Lon95]

Definition 8. *Let $\gamma : A \to B$ be an applicative morphism. The functor γ_* sends $(X, ||-||_X)$ in $\mathbf{Ass}(A)$ to $(X, \gamma(||-||_X))$ in $\mathbf{Ass}(B)$ and a morphism $f : X \to Y$ to $f : \gamma_* X \to \gamma_* Y$ whose underlying map is $|f|$.*

If $f : X \to Y$ is realized by $s \in B$ then $\gamma_*(f)$ is realized by $\lambda^* x.rs'x$ where r is a realizer of γ and s' is an element of $\gamma(s)$.

Proposition 6. *Let $\gamma : A \to B$ be an applicative morphism. Then γ_* is a lax monoidal functor from $\mathbf{Ass}(A) \to \mathbf{Ass}(B)$.*

Proof. Underlying maps of two natural transformations $m_{X,Y} : \gamma_*(X) \otimes \gamma_*(Y) \to \gamma_*(X \otimes Y)$ and $m_I : I \to \gamma_*(I)$ are both identity. If we choose an element $a \in \gamma(\lambda^* xy.[x, y])$ and a realizer r of γ, $\lambda^* pq.r(rap)q$ realizes m_X, Y. A realizer of m_I is $\lambda^* x.xi'$ where i' is an element of $\gamma(i)$. It is easy to see that these natural transformations satisfy coherence diagrams. □

If $\gamma \preceq \delta$ are applicative morphisms related by the preorder between them, then there is a monoidal natural transformation from γ_* to δ_*.

Definition 9. *If $\gamma \preceq \delta : A \to B$ are applicative morphisms related by the preorder, $\alpha_* : \gamma_* \to \delta_* : \mathbf{Ass}(A) \to \mathbf{Ass}(B)$ is a natural transformation such that an underlying map of α_{*X} is identity.*

From the definition of preorder, we can see that a realizer of $\gamma \preceq \delta$ realizes $\alpha_{*X} : \gamma_*(X) \to \delta_*(X)$. For any morphism $f : X \to Y$, $|\alpha_{*Y}\gamma_*(f)|$ is $|f|$ and $|\delta_*(f)\alpha_{*X}|$ is also $|f|$. Hence, $\alpha_{*Y}\gamma_*(f) = \delta_*(f)\alpha_{*X}$. It is easy to see that this is a monoidal natural transformation.

This construction is a 2-functor from a 2-category of BCI algebras and PCA's to a 2-category of categories of assemblies realized by BCI algebras and PCA's.

4 Realizability of relational combinatory algebras

4.1 Relational linear combinatory algebras

In Sect. 2.2 we recalled the definitions of applicative morphisms and preorder between them. In this section we define adjoint pair and comonadic applicative morphism. The same definition of adjoint pair for PCA's is given in [Lon95].

Definition 10. *Let A,B be BCI algebras or PCA's.* $\delta \dashv \gamma : A \to B$ *is an* adjoint pair *if* $\delta : B \to A$ *and* $\gamma : A \to B$ *are applicative morphisms satisfying* $\delta\gamma \preceq 1_A$ *and* $1_B \preceq \gamma\delta$*.*

Definition 11. *Let A be a BCI algebra or a PCA.* $\gamma : A \to A$ *is a* comonadic applicative morphism *if* γ *is an applicative morphism satisfying* $\gamma \preceq 1_A$ *and* $\gamma \preceq \gamma\gamma$*. Notice that any comonadic applicative morphism* γ *is equivalent to* $\gamma\gamma$ *and* γ_* *is idempotent.*

Just as a comonad functor can be constructed from an adjunction, we can construct a comonadic applicative morphism from an adjoint pair.

Proposition 7. *Let A,B be BCI algebras or PCA's and* $\delta \dashv \gamma : A \to B$ *is an adjoint pair. Then* $\epsilon = \delta\gamma : A \to A$ *is a comonadic applicative morphism.*

Proof. Since $\delta \dashv \gamma$, $\delta\gamma \preceq 1_A$ and $1_B \preceq \gamma\delta$. Hence $\epsilon \preceq 1_A$ and $\epsilon = \delta\gamma \preceq \delta\gamma\delta\gamma = \epsilon\epsilon$. □

We define 'relational LCA', which is an analogue of linear category.

Definition 12. *A* relational linear combinatory algebra *(rLCA)* $(A, !)$ *consists of a BCI algebra A and a comonadic applicative morphism* $! : A \to A$ *such that*

$$! \preceq [!, !] \quad ! \preceq k_i$$

where $[!, !]$ *is an applicative morphism such that* $[!, !](p) = \{[u, v] | u, v \in !(p)\}$ *which is realized by*

$$\lambda^* pq.\texttt{let } [p_1, p_2] = p \texttt{ in let } [q_1, q_2] = q \texttt{ in } [rp_1q_1, rp_2q_2]$$

using a realizer r *of* $!$*; and* k_i *is an applicative morphism such that* $k_i(a) = \{i\}$ *which is realized by* i*.*

In the same way as we can construct a linear category from a linear-non-linear model, we can construct an rLCA from an adjoint pair between a BCI algebra and a PCA.

Proposition 8. *Let A be a BCI algebra, B be a PCA and* $\delta \dashv \gamma : A \to B$ *be an adjoint pair. Then* $(A, !)$ *is an rLCA where* $! = \delta\gamma : A \to A$*.*

Proof. From Proposition 7, ! is a comonadic applicative morphism. Let s be a realizer for $\delta\gamma \preceq 1_A$, t be a realizer for $1_B \preceq \gamma\delta$, r_δ a realizer for δ and r_γ a realizer for γ.

Choose $u \in \gamma(\lambda^* xy.[x, y])$ and $v \in \delta(\lambda^* x.r_\gamma(r_\gamma u(tx))(tx))$ and we will show that $! \preceq [!, !]$ is realized by $\lambda^* x.s(r_\delta vx)$; let $p' \in \delta\gamma(p)$. Then $(\lambda^* x.s(r_\delta vx))p' \simeq s(r_\delta vp')$. There exists $p'' \in \gamma(p)$ such that $p' \in \delta(p'')$ and $r_\delta vp'$ is an element of $\delta(r_\gamma(r_\gamma u(tp''))(tp''))$ if $r_\gamma(r_\gamma u(tp''))(tp'')$ is defined. Since $tp'' \in \gamma\delta\gamma(p)$ there exists $q, q' \in \delta\gamma(p)$,

$$r_\gamma(r_\gamma u(tp''))(tp'') \in \gamma((\lambda^* xy.[x, y])qq')$$

if $(\lambda^* xy.[x,y])qq'$ is defined. Since a BCI algebra A is total, this $(\lambda^* xy.[x,y])qq'$ is defined and is equal to $[q,q']$. Hence, $r_\delta vp'$ is an element of $\delta\gamma([q,q'])$ and $t(r_\delta vp') \in \{[q,q']\} \subseteq [!,!](p)$.

$! \preceq k_i$ is realized by $\lambda^* x.t(r_\delta hx)$ where h is an element of $\delta(\lambda^* x.i')$ taking an element $i' \in \gamma(i)$: If $p' \in \delta\gamma(p)$, there exists $q \in \gamma(p)$ and $r_\delta hp' \in \delta((\lambda^* x.i')q)$ if $(\lambda^* x.i')q$ is defined. In fact this is defined and equal to i'. Hence $r_\delta hp'$ is an element of $\delta\gamma(i)$, $t(r_\delta hp')$ is an element of $\{i\}$. □

It seems not the case in general that we can construct a PCA from an rLCA. However if we restrict applicative morphisms to be functional, we can construct a PCA. We recall the notion of *linear combinatory algebra* (LCA) which appears in [AHS02].

Definition 13. *A BCI algebra $(A,\cdot)$ is a* linear combinatory algebra *if it has a map $! : A \to A$ and $k, w, d, \delta, f \in A$ satisfying:*

$$kx!y = x \quad \delta!x = !!x \quad d!x = x \quad wx!y = x!y!y \quad f!x!y = !(xy)$$

If we define γ as $\gamma(p) = \{!p\}$, this is a comonadic applicative morphism. $! \preceq 1$ is realized by d, $! \preceq !!$ is realized by δ and a realizer of γ is f. Hence (A,γ) is an rLCA since $! \preceq [!,!]$ is realized by $\lambda^* z.w(\lambda^* xy.[x,y])z$ and $! \preceq k_i$ is realized by $\lambda^* x.kix$. Hence we can think LCA as a special case of rLCA whose ! is functional.

Proposition 9. *Let $\delta \dashv \gamma : A \to B$ be a functional adjoint pair from a BCI algebra A to a PCA B. Then $(A,!)$ is an LCA where $!p = p'$ iff $\delta\gamma(p) = \{p'\}$.*

Proof. From Proposition 8 and since LCA is a special case of rLCA. □

If we have an LCA then we can construct a functional adjoint pair.

Lemma 2. *Let $(A,!)$ be an LCA then an applicative structure $(A_!, \bullet)$ is an SK algebra where $|A_!| = A$ and $p \bullet q = p \cdot !q$.*

Proof. Let s and k' be elements of $A_!$ such that $s = \lambda^* xyz.w(\lambda^* uv.dxu(fy(\delta v)))z$ and $k' = \lambda^* xy.k(dx)y$. Then s and k' satisfy $s!x!y!z = x(!z)!(y(!z))$ and $k'!x!y = k(d(!x))(!y) = x$. □

Proposition 10. *Let $(A,!)$ be an LCA. Then there is a functional adjoint pair between A and $A_!$.*

Proof. Let $\rho : A \to A_!$ and $\sigma : A_! \to A$ be applicative morphisms such that $\rho(p) = p$ and $\sigma(p) = !p$. ρ and σ are realized by $\lambda^* xy.(dx)(dy)$ and $\lambda^* xy.fx(\delta y)$ respectively and $1_{A_!} \preceq \rho\sigma$ is realized by i, $\sigma\rho \preceq 1_A$ is realized by d. □

The next lemma can be proved as Theorem 3.1.8 and Corollary 3.1.9 of [Lon95].

Lemma 3. *There is no SK algebra which has a decidable equality.*

Here a PCA A has a decidable equality when there is an element $d \in A$ such that

$$dxy = \begin{cases} \lambda^* uv.u & \text{if } x = y \\ \lambda^* uv.v & \text{if } x \neq y \end{cases}$$

The generalization of LCA to rLCA enables us to treat PCA which has a decidable equality.

Proposition 11. *If a PCA A has a decidable equality then there is no functional adjoint pair from any BCI algebra to A.*

Proof. We suppose there is a functional adjoint pair $\delta \dashv \gamma : B \to A$ where B is a BCI algebra. By Proposition 10, we have a SK algebra $B_{\delta\gamma}$. We show $B_{\delta\gamma}$ has decidable equality. Then by the above lemma, we obtain contradiction.

Let c be an element of A which decide the equality of A. Then

$$\lambda^* xy.d((((\delta(c) * x) * y) * \delta\gamma(true)) * \delta\gamma(false))$$

decides the equality of $B_{\delta\gamma}$ where $x * y = r_\delta xy$ for a realizer r_δ of δ, d is a realizer of $\delta\gamma \preceq 1$, $true$ and $false$ is $\lambda^* uv.u$ and $\lambda^* uv.v$ of $B_{\delta\gamma}$ and $\lambda^* xy. \cdots$ is a lambda abstraction of B. □

4.2 Assemblies and modest sets realized by rLCA's

Let $(A, !)$ be an rLCA. As we have seen in the previous section, we have a monoidal comonad $!_* : \mathbf{Ass}(A) \to \mathbf{Ass}(A)$ with natural transformations w_X : $!_* X \to I$ which sends x to $*$ and $c_X : !_* X \to !_* X \otimes !_* X$ which sends x to (x, x).

Proposition 12. *Let A be an rLCA then $\mathbf{Ass}(A)$ is finitely complete and cocomplete and there are natural isomorphisms $!_* X \otimes !_* Y \simeq !_*(X \times Y)$ and $I \simeq !_* 1$.*

Proof. Definitions of terminal, initial, equalizer and coequalizer are the same in [Lon95].

For products, let X,Y be objects of $\mathbf{Ass}(A)$. Then we define $X \times Y$ as

$$|X \times Y| = |X| \times |Y|$$
$$||(x, y)||_{X \times Y} = \{[a, [p, q]] | \exists r, s. p \in !r, q \in !s, ra \in ||x||_X, sa \in ||y||_Y\}$$

First projection $\pi : X \times Y \to X$ which sends (x, y) to x and is realized by

$$\lambda^* x.\mathtt{let}\ [t, u] = x\ \mathtt{in}\ \mathtt{let}\ [v, w] = u\ \mathtt{in}\ (kw)(dv)t$$

where k is a realizer of $! \preceq k_i$ and d is a realizer of $! \preceq 1_A$. Second projection π' is similar. Let $f : Z \to X$ and $g : Z \to Y$ be morphisms realized by m, n respectively. Then we have a map $h : Z \to X \times Y$ which sends z to (fz, gz). This morphism is realized by $\lambda^* x.[x, [m', n']]$ where $m' \in !m$ and $n' \in !n$. h satisfies $\pi h = f$, $\pi' h = g$ and we can see uniqueness from underlying maps of these morphisms.

$!_*(\langle\theta(d_X \otimes w_Y), \theta'(w_X \otimes d_Y)\rangle)m_{X,Y}(\delta_X \otimes \delta_Y)$ and $(!_*\pi \otimes !_*\pi')c_{X\times Y}$ form an isomorphism of $!_*X \otimes !_*Y \simeq !_*(X \times Y)$, $!_*(u)m_I$ and w_1 form an isomorphism of $I \simeq !_*1$ since underlying maps of those morphisms are identity. Here $\theta : X \otimes I \simeq X$ and $\theta' : I \otimes Y \simeq Y$, m is a natural transformation of $!_*$ and $u : I \to 1$ is a unique morphism.

For coproducts, let X,Y be objects of $\mathbf{Ass}(A)$. If k realizes $! \preceq k_i$ and d realizes $! \preceq 1_A$ then

$$
\begin{aligned}
|X+Y| &= |X| + |Y| \\
||(0,x)||_{X+Y} &= \{[p,r] | r \in ||x||_X\} \\
||(1,y)||_{X+Y} &= \{[q,s] | s \in ||y||_Y\}
\end{aligned}
$$

where p is $\lambda^* xy.(ky)(dx)$ and q is $\lambda^* xy.(kx)(dy)$. Inclusion $in_1 : X \to X+Y$ sends x to $(0,x)$. This morphism is realized by $\lambda^* x.[p,x]$. Inclusion $in_2 : Y \to X+Y$ is similar. Let m,n realize $f : X \to Z$ and $g : Y \to Z$ respectively. Then a morphism $h : X+Y \to Z$ which sends $(0,x)$ to fx and $(1,y)$ to gy is realized by $\lambda^* x.\texttt{let}\ [u,v] = x\ \texttt{in}\ (um'n')v$ where $m' \in !m$ and $n' \in !n$. For example, if we apply $[p,a] \in ||(0,x)||$ to this realizer then we get $pm'n'a$. From definition of p, this is $(kn')(dm')a = ma \in ||fx||$ since $kn' = i$, $dm' = m$. □

Hence the coKleisli category $\mathbf{Ass}(A)_{!_*}$ is a CCC and $\mathbf{Ass}(A)_{!_*}$ also has finite coproducts since $!_*$ is an idempotent comonad on $\mathbf{Ass}(A)$. Initial object is the same one of $\mathbf{Ass}(A)$ and a coproduct of X and Y is $!_*X + !_*Y$.

Proposition 13. *Let $(A, !)$ be an rLCA. $\mathbf{Ass}(A)$ and $\mathbf{Ass}(A)_{!_*}$ form a linear-non-linear model.*

Proof. By Proposition 12, $\mathbf{Ass}(A)_{!_*}$ is a CCC. We show the left adjoint G is strong monoidal since if the left adjoint is strong monoidal then the adjunction is monoidal. G is a strong monoidal functor since $\alpha : !_*X \otimes !_*Y \to !_*(X \times Y)$ and $\beta : I \to !_*1$ where α and β are isomorphisms given in Proposition 12. Required diagrams commute since underlying maps of these morphisms are identity. □

Since there is a monoidal adjunction $\Delta \dashv J : \mathbf{Mod}(A) \to \mathbf{Ass}(A)$, $\Delta !_* J$ is a monoidal comonad on $\mathbf{Mod}(A)$. If X is a modest set and $a \in ||x||_{!_*X} \cap ||y||_{!_*X}$ then $ra \in ||x||_X \cap ||y||_X$ where r is a realizer of $! \preceq 1_A$, hence $x = y$. Hence if X is a modest set then $!_*JX$ is a modest set and $!_*JX \simeq J\Delta !_* JX$.

From Proposition 5, $\mathbf{Mod}(A)$ is also finite complete and cocomplete. For any object X,Y in $\mathbf{Mod}(A)$, $\Delta !_* J(X \times Y) \simeq \Delta !_*(JX \times JY) \simeq \Delta(!_*JX \otimes !_*JY) \simeq \Delta !_* JX \bar{\otimes} \Delta !_* JY$ and $\Delta !_* J1 \simeq \Delta !_* 1 \simeq \Delta I \simeq \bar{I}$ since Δ is a left adjoint and especially strong monoidal where we write the monoidal product of $\mathbf{Mod}(A)$ as $\bar{\otimes}$ and the unit of $\mathbf{Mod}(A)$ as $\bar{I}$. This means $\mathbf{Mod}(A)$ satisfies Proposition 12 and its coKleisli category $\mathbf{Mod}(A)_{!_*}$ is a CCC. $\mathbf{Mod}(A)_{!_*}$ also has finite coproducts since $\Delta !_* J$ is idempotent and $\mathbf{Mod}(A)$ has finite coproducts.

Let $G' \dashv J'$ be an adjunction of $\mathbf{Mod}(A) \to \mathbf{Mod}(A)_{!_*}$. Then $G' = \Delta GL$ where $L : \mathbf{Mod}(A)_{!_*} \to \mathbf{Ass}(A)_{!_*}$ is a comparison functor of comonad $\Delta !_* J$. L is a strong monoidal functor since LX is JX for any object X of $\mathbf{Mod}(A)_{!_*}$ and a product of X and Y in $\mathbf{Mod}(A)_{!_*}$ is the product of $\mathbf{Mod}(A)$ and J preserves finite products. Δ and G are also strong monoidal functors by Proposition 13 and since Δ is a left adjoint. Hence G' is a strong monoidal functor and $G' \dashv J'$ is a monoidal adjunction.

Proposition 14. *Let $(A, !)$ be an rLCA. $\mathbf{Mod}(A)$ and its coKleisli category $\mathbf{Mod}(A)_{!_*}$ form a linear-non-linear model.*

4.3 Natural number object in Ass(A)

Let A be an rLCA. We construct a natural number object of $\mathbf{Ass}(A)_{!_*}$. First we define some notations.

Definition 14. *Let P be a polynomial over A. If P has a variable x which appears in P exactly once. Then $\lambda^* x.P$ is defined as*

$$\begin{aligned} \lambda^* x.x &= i \\ \lambda^* x.PQ &= b \cdot (\lambda^* x.P) \cdot Q \quad (\textit{if } P \textit{ has } x) \\ \lambda^* x.PQ &= c \cdot P \cdot (\lambda^* x.Q) \quad (\textit{if } Q \textit{ has } x) \end{aligned}$$

For any polynomial P over A, $\lambda^* !x.P$ is defined as

$$\begin{aligned} \lambda^* !x.x &= d \\ \lambda^* !x.a &= k \cdot a \\ \lambda^* !x.y &= k \cdot y \quad (\textit{if } y \neq x) \\ \lambda^* !x.PQ &= w \cdot (\lambda^* xy.((\lambda^* !x.P) \cdot x) \cdot ((\lambda^* !x.Q) \cdot y)) \end{aligned}$$

When free variables of a polynomial P are only x, $\lambda^* !x.P$ has no free variables. Then we treat $\lambda^* !x.P$ as an element of A. By induction of the definition, we can see for any $a' \in !a$, $(\lambda^* !x.P)a' = P[a/x]$.

Let $\bar{n}$ be $\lambda^* !fx.\overbrace{f(\cdots(f}^{n} x)\cdots)$. We define $|N|$ as a set of natural numbers and $||n||_N = \{\bar{n}\}$. $0 : 1 \to N$ which sends $*$ to $0 \in |N|$ is realized by $\lambda^* !x.\bar{0}$. $S : N \to N$ is realized by $\lambda^* !n.w(\lambda^* xy.((\lambda^* !f.cf)x)(ny))$ since

$$\overline{n+1} = \lambda^* !f.\overbrace{(cf)(\cdots((cf}^{n+1} i)\cdots) = w(\lambda^* xy.((\lambda^* !f.cf)x)(\bar{n}y))$$

In $\mathbf{Ass}(A)_{!_*}$, given $x : 1 \to X$ and $f : X \to X$, a morphism $h : N \to X$ which sends n to $f^n(x)$ is realized by $\lambda^* !n.nra$ where r is an element of $!s$ for a realizer s of f and a is an element of $||x||_X$. By an induction of the definition of $\bar{n}$, we can show h is well defined and uniqueness follows from that $|N|$ is a natural number object of $\mathbf{Set}$. Notice that N is a modest set if A has more than two elements: Let a, b are two different elements of A and B be an object such that $|B| = 2$ and $||0||_B = \{a\}$, $||1||_B = \{b\}$. For any $n < m \in N$, since N is a natural number object, $f : N \to B$ which sends i to 0 if $i \leq n$ and 1 if $i > n$ is realizable. Hence $\bar{n} \neq \bar{m}$.

We have $!_*(1) \simeq I$ since $!_* : \mathbf{Ass}(A)_{!_*} \to \mathbf{Ass}(A)$ is a left adjoint. Hence for any $x : I \to X$,$f : X \to X$ in $\mathbf{Ass}(A)$, there exists a unique morphism $h :!_*N \to X$ such that $h!_*(0) = x$ and $h!_*S = fh$.

5 Examples of rLCA

5.1 Linear lambda calculus

The untyped linear lambda calculus is defined in [Sim05]. Terms of the untyped linear lambda calculus is defined as

$$t = x|tt|\lambda x.t|\lambda !x.t|!t$$

t of $\lambda x.t$ is required to have exactly one x which is not in any scope of !. A set of closed terms up to reductions given in [Sim05] forms an LCA.

5.2 BCK algebra with a structure of ω-cppo

This is inspired by examples in [AHS02]. A *BCK algebra* is a BCI algebra which also has an element k which satisfies $kxy = x$. for all x, y. Let A be a BCK algebra with a structure of ω-complete pointed poset (ω-cppo) and the application of A is continuous with this structure, and $\bot x = \bot$ for all $x \in A$. We define $! : A \to A$ as $!a = \mu x.[a, [x, x]]$ where $\mu x.fx$ is the least fixed point of f. Then $(A, !)$ is an LCA.

It is easy to see $! \preceq 1_A$, $! \preceq k_i$ and $! \preceq [!, !]$ and by the following propositions, ! is an applicative morphism and $! \preceq !!$.

If t is constructed from a variable x and elements of A and λ abstractions then t is a continuous function from A to A since the application of A is continuous.

Proposition 15. *! is realized by*

$$f = \mu z.\lambda^* xy.\ \texttt{let}\ [p, [q, r]] = x\ \texttt{in let}\ [u, [v, w]] = y\ \texttt{in}\ [pu, [zqv, zrw]]$$

Proof. Let $T_a = [a, [x, x]]$ for $a \in A$. We have $f\bot\bot = \bot$ since $\bot x = \bot$ for all $x \in A$, and for $a, b \in A$, $fT_a^{n+1}(\bot)T_b^{n+1}(\bot)$ is equal to

$$\texttt{let}\ [p, [q, r]] = T_a^{n+1}(\bot)\ \texttt{in let}\ [u, [v, w]] = T_b^{n+1}(\bot)\ \texttt{in}\ [pu, [fqv, frw]]$$

which is, by induction, $T_{ab}(fT_a^n(\bot)T_b^n(\bot)) = T_{ab}^{n+1}(\bot)$. Hence by continuity of the application,

$$f!a!b = \bigvee_n fT_a^n(\bot)T_b^n(\bot) = \bigvee_n T_{ab}^n(\bot) = !(ab)$$

□

Proposition 16. *$! \preceq !!$ is realized by*

$$\delta = \mu z.\lambda^* x.\texttt{let}\ [p, [q, r]] = x\ \texttt{in let}\ [u, [v, w]] = r\ \texttt{in}\ [w, [zq, zv]]\ .$$

Proof. From properties of the least fixed point,

$$\begin{aligned} !a &= \mu x.\mu y.[a,[x,y]] \quad \cdots (*) \\ &= \mu x.[a,[x,[a,[x,\mu y.[a,[x,y]]]]]] \\ &= \mu x'.\mu x.[a,[x,[a,[x,\mu y.[a,[x',y]]]]]] \\ &= \mu x.[a,[x,[a,[x,\mu y.[a,[!a,y]]]]]] \end{aligned}$$

From $(*)$, $!a = \mu x.[a,[!a,x]]$ and hence $!a = \mu x.[a,[x,[a,[x,!a]]]]$. Let $S_a(x) = [a,[x,[a,[x,!a]]]]$ then $!a = \bigvee_n S_a^n(\bot)$. By induction we can see $\delta S_a^n(\bot) = T_{!a}^n(\bot)$. Hence we have

$$\delta !a = \bigvee_n \delta S_a^n(\bot) = \bigvee_n T_{!a}^n(\bot) = !!a$$

□

For example, let $\mathcal{C}$ be **Rel**(sets and relations) or **Pfn**(sets and partial functions). Then $\mathcal{C}$ has an object $\mathbb{N}$(set of natural numbers) which satisfies $\mathbb{N}\times\mathbb{N}\lhd\mathbb{N}$. Since $\mathcal{C}$ is a traced monoidal category whose monoidal product is coproduct, we can construct a new category by the GoI construction [AHS02], [JSV96]. Let this GoI-category be $\mathcal{G}(\mathcal{C})$. In $\mathcal{G}(\mathcal{C})$, $(\mathbb{N},\mathbb{N})$ is a reflexive object as

$$(\mathbb{N},\mathbb{N})^* \multimap (\mathbb{N},\mathbb{N}) \simeq (\mathbb{N},\mathbb{N}) \otimes (\mathbb{N},\mathbb{N}) \lhd (\mathbb{N},\mathbb{N}) \ .$$

Since $\mathcal{G}(\mathcal{C})$ is an SMCC, $\mathcal{G}(\mathcal{C})(I,(\mathbb{N},\mathbb{N}))$ forms a BCI algebra. However in fact this is a BCK algebra and $\mathcal{G}(\mathcal{C})(I,(\mathbb{N},\mathbb{N})) = \mathcal{C}(\mathbb{N},\mathbb{N})$ forms ω-cppo by inclusion order whose application is continuous and for any $f \in \mathcal{C}(\mathbb{N},\mathbb{N})$, $\phi \cdot f = \phi$. Hence this is an LCA.

By Lemma 2, we can construct an SK algebra from this algebra. If $\mathcal{C}$ is **Rel** then this algebra is isomorphic to $\mathcal{P}(\omega)$.

These examples are already in [AHS02]. However the LCA's we obtain are a little bit different. Comonoidal applicative morphisms of examples in [AHS02] are different from ours. Although, comonads constructed from these applicative morphisms are equivalent and we can think examples of here and ones in [AHS02] are the same.

5.3 $\mathcal{P}(\omega)_{lin}$

$\mathcal{P}(\omega)_{lin}$ is an LCA defined as

$$\begin{aligned} &\mathcal{P}(\omega)_{lin} = \mathcal{P}(\mathbb{N}) && \alpha\cdot\beta = \{n \mid \langle m,n\rangle \in \alpha, m \in \beta\} \\ &!\alpha = \{n \mid e_n \subseteq \alpha\} && \\ &d = \{\langle n,m\rangle \mid m \in e_n\} && \delta = \{\langle n,m\rangle \mid \textstyle\bigcup_{i\in e_m} e_i \subseteq e_n\} \\ &w = \{\langle l,\langle m,n\rangle\rangle \mid l = \langle i,\langle j,n\rangle\rangle \wedge e_i \textstyle\bigcup e_j \subseteq e_m\} && f = \{\langle l,\langle m,n\rangle\rangle \mid e_n \subseteq e_l \cdot e_m\} \end{aligned}$$

here $\langle -,-\rangle$ is a bijection from $\mathbb{N}\times\mathbb{N}$ to $\mathbb{N}$ and e is a bijection from $\mathbb{N}$ to the set of finite subsets of $\mathbb{N}$. If an object X of a compact closed category $\mathcal{C}$ satisfies $X^* = X$ and $X \otimes X \lhd X$ then $\mathcal{C}(I,X)$ form a BCI algebra, especially $\mathbb{N}$

of **Rel**. Hence this $\mathcal{P}(\omega)_{lin}$ is a BCI algebra and forms an LCA with above ! and d, δ, w, f. It is easy to see the SK algebra obtained from this LCA is $\mathcal{P}(\omega)$. This example can be modified to recursive enumerable subsets of $\mathbb{N}$ and we write this rLCA as $\mathcal{P}(\omega)_{lin,re}$.

5.4 Kleene's first algebra $\mathbf{K_1}$

In [Lon95] it is proved that there is an adjoint pair $\varphi \dashv \psi : \mathcal{P}(\omega)_{re} \to \mathrm{K}_1$. Let $\delta \dashv \gamma$ be an adjoint pair from $\mathcal{P}(\omega)_{lin,re}$ to $\mathcal{P}(\omega)_{re}$ then $\delta\varphi \dashv \psi\gamma$ is an adjoint pair from $\mathcal{P}(\omega)_{lin,re}$ to K_1 and $(\mathcal{P}(\omega)_{lin,re}, \delta\varphi\psi\gamma)$ is an rLCA. Notice that there is no adjoint pair between any BCI algebra and K_1 by Proposition 11.

5.5 Kleene's second algebra $\mathbf{K_2}$

Definition 15. *K_2 is a set of functions from $\mathbb{N}$ to $\mathbb{N}$ whose application is*

$$f \cdot g = \begin{cases} f * g & (\textit{if } f * g \textit{ is a total function}) \\ \textit{undefined} & (\textit{else}) \end{cases}$$

where $f * g(n) = m$ iff

$$\exists k. \forall i < k. f([n, g(0), \cdots, g(i)]) = 0 \wedge f([n, g(0), \cdots, g(k)]) = m + 1$$

here $[-, \cdots, -]$ is a bijection from a set of finite lists of natural numbers to $\mathbb{N}$. For details, see [KV65].

Let $A_{\mathbf{Pfn}}$ be an LCA constructed from **Pfn** in Sec 5.2 and write $(A_{\mathbf{Pfn}})_!$ for the SK algebra constructed as in Lemma 2. Then there is an adjoint pair $\gamma \dashv \delta : (A_{\mathbf{Pfn}})_! \to \mathrm{K}_2$. Since there is an adjoint pair from $A_{\mathbf{Pfn}}$ to $(A_{\mathbf{Pfn}})_!$ we obtain an adjoint pair from $A_{\mathbf{Pfn}}$ to K_2. $\gamma : \mathrm{K}_2 \to (A_{\mathbf{Pfn}})_!$ is defined as $\gamma(f) = \{f\}$ and $\delta : (A_{\mathbf{Pfn}})_! \to \mathrm{K}_2$ is defined as

$$\delta(f) = \{g | \langle g0 \rangle \leq \langle g1 \rangle \leq \cdots \leq f, \bigvee_n \langle gn \rangle = f\}$$

where $\langle - \rangle$ is bijection from $\mathbb{N}$ to a set of partial functions of $\mathbb{N}$ to $\mathbb{N}$ whose domain is finite. The order $\leq$ is an inclusion order of its graph and $\bigvee$ is a union of graphs. Notice that K_2 is another example of PCA that has no adjoint pair between any BCI algebra.

Acknowledgment
I am grateful to Masahito Hasegawa and anonymous reviewers for their valuable advice.

6 Concluding remarks

- In order to model second order linear lambda calculus, we can use a category of partial equivalence relation (PER). By a similar argument of PER realized by PCA, PER realized by an rLCA provides a model of the second order linear lambda calculus.
- We can restrict the modality of ! of rLCA to only $! \preceq [\overbrace{id, id, \cdots, id}^{n}]$ for $n \geq 0$. These modalities are what soft linear logic [Laf04] has. One example is the untyped soft linear lambda calculus whose terms are

$$t = x|tt|\lambda x.t|\lambda !x.t|!t$$

  where t of $\lambda x.t$ is required to have exactly one appearance of x which is not in any scope of ! and t of $\lambda !x.t$ is required to have exactly one appearance of x which is in a scope of at most one ! or to have no x which is in a scope of !. Then the untyped soft linear lambda calculus strongly normalizes in polynomial steps in "weight" of a term and morphisms of a category of assemblies realized by the untyped soft linear lambda calculus is computable in polynomial time in some sense.

Some further considerations are found in the author's MSc thesis [Hos07].

References

[AHS02] Samson Abramsky, Esfandiar Haghverdi, and Philip J. Scott. Geometry of interaction and linear combinatory algebras. *Mathematical Structures in Computer Science*, 12(5):625–665, 2002.

[AL05] Samson Abramsky and Marina Lenisa. Linear realizability and full completeness for typed lambda-calculi. *Ann. Pure Appl. Logic*, 134(2-3):122–168, 2005.

[Ben94] P. N. Benton. A mixed linear and non-linear logic: Proofs, terms and models (extended abstract). In *CSL*, pages 121–135, 1994.

[Bie95] Gavin M. Bierman. What is a categorical model of intuitionistic linear logic? In *TLCA*, pages 78–93, 1995.

[HO93] J. M. E. Hyland and C.-H. Luke Ong. Modified realizability toposes and strong normalization proofs. In *TLCA*, pages 179–194, 1993.

[Hos07] N. Hoshino. Linear realizability. Master's thesis, Kyoto University, 2007.

[Jac99] B. Jacobs. *Categorical Logic and Type Theory.* Number 141 in Studies in Logic and the Foundations of Mathematics. North Holland, Amsterdam, 1999.

[JSV96] André Joyal, Ross Street, and D. Verity. Traced monoidal categories. *Math. Proc. Cambridge Phil. Soc.*, 119(3), 1996.

[KV65] Stephen Cole Kleene and R. E. Vesley. *The Foundations of Intuitionistic Mathematics, especially in relation to recursive functions.* North-Holland Publishing Company, 1965.

[Laf04] Yves Lafont. Soft linear logic and polynomial time. *Theor. Comput. Sci.*, 318(1-2):163–180, 2004.

[Lon95] J. Longley. *Realizability Toposes and Language Semantics.* PhD thesis, Edinburgh University, 1995.

[Sim05] Alex K. Simpson. Reduction in a linear lambda-calculus with applications to operational semantics. In *RTA*, pages 219–234, 2005.