A CATEGORICAL APPROACH TO LINEAR LOGIC, GEOMETRY OF PROOFS AND FULL COMPLETENESS

By
Esfandiar Haghverdi Candidate, B.Sc., M.S.
May 2000

A Thesis
submitted to the School of Graduate Studies and Research
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy in Mathematics[1]

[1]The Ph.D. Program is a joint program with Carleton University, administered by the Ottawa-Carleton Institute of Mathematics and Statistics

Dedicated to the memory of my father

Abstract

The major contributions of this thesis are in the areas of Geometry of Interaction (GoI) and full completeness for models of linear logic. Geometry of interaction was introduced by Girard in the late 80's. It provides a new semantics of computation that captures the dynamical aspects of computation; a feature which is missing in denotational semantics. Subsequently Abramsky and Jagadeesan introduced a categorical interpretation of this program using domain theoretical categories. Recently Abramsky introduced a general categorical GoI construction which leads to models of computation (combinatory algebras) based on GoI Situations that essentially consist of a traced symmetric monoidal category with additional structure.

In this thesis we develop this general programme in all details. We also prove that this construction can be done in a more general framework, namely based on symmetric monoidal closed categories.

We also introduce a class of categories, called *unique decomposition categories* (UDC), that axiomatise the so-called particle-style GoI along with providing a computational calculus. Unique decomposition categories are motivated by and generalise the partially additive categories of Manes and Arbib. These models are special cases of the general construction which capture an important class of examples. They provide us with dataflow-like computational analysis.

We also establish connections with the work of Girard/Danos/Regnier on dynamic algebras. This formalises the path-based approach to GoI, the so-called *path-semantics*. The paths in proof-nets used in the work of Danos/Regnier correspond to morphisms in UDC-based models.

Full completeness for models of linear logic was introduced and studied by Abramsky and Jagadeesan. Full completeness establishes the tightest connection between syntax and semantics: it is completeness with respect to proofs instead of the traditional completeness with respect to provability. Recently Hyland and his student Tan introduced the double glueing method in Tan's PhD thesis. Tan's work provides the first steps towards an axiomatic approach to full completeness problems for linear logic.

In this thesis we construct a class of models for the multiplicative fragment of linear logic with the MIX rule, based on partially additive categories together with the Int construction of Joyal, Street and Verity and the double glueing construction. We also prove that such models are fully complete for the multiplicative fragment of linear logic with the MIX rule.

Acknowledgements

I would like to thank my supervisor Philip Scott with whom I had the pleasure of working. His continuous help, encouragement and support during the last four years have meant a great deal to me. He has also introduced me to many great mathematicians and scientists for which I am very grateful.

I would like to express my deepest gratitude to Prakash Panangaden for his encouragement and keen interest in my work and progress over the last four years. His diversity of intellectual interests and incisive mental energy have been a great source of stimulation and inspiration to me.

I would also like to thank Richard Blute and Thierry Giordano for sharing their knowledge and expertise with me during many helpful discussions.

I would like to express my appreciation to Samson Abramsky for introducing me to his analysis of Girard's GoI programme. He has thus provided me with a most interesting research subject.

Many thanks to the entire secretarial staff of the Department of Mathematics and Statistics. I have always been delighted by their kindness and good humour.

I would like to thank my mother who has always kept me in her thoughts and prayers.

It is so hard to find words strong enough to express my gratitude and thankfulness to my wife Manijeh, the love of my life and my best friend. She has always been by my side and has shared my happiness and sorrow during the many ups and downs in the last 4 years. I will always treasure her for the long hours and sleepless nights that she has spent listening to me talking about my work. Her numerous encouraging words have kept me going at times when I felt I was at the end of my rope. I owe

this thesis to her belief in me, her patience and her never ending love and support.

The commutative diagrams in this thesis are drawn using Paul Taylor's `diagrams` macro.

Contents

Chapter 1

Introduction

Linear logic was invented by Girard in 1987. Linear logic is a constructive and resource sensitive logic which is a refinement of classical and intuitionistic logic. Girard introduced many novelties including a graphical proof syntax and a wide range of mathematical models. Linear logic has had many impacts on theoretical computer science. These include but are not limited to logic programming, database theory, automated reasoning, semantics of programming languages, and program efficiency. Linear logic is the first attempt to solve the problem of parallelism at the logical level. Since its introduction several mathematical models have been introduced for different fragments of linear logic by Girard, Barr, Kleisli, R. Blute among others. Several of these models are constructed on the categories of Banach spaces (Girard, Barr, Kleisli) and Topological vector spaces (Blute & Scott).

In the late 1980's Girard initiated his "Geometry of Interaction" (GoI) programme in [46] and continued his investigation in a series of papers [47, 45, 48]. The goal of this programme was to provide a mathematical analysis of the cut elimination process in linear logic proofs. This new interpretation replaces graph-rewriting by an analysis of information flow in proof-nets. From a computational point of view this is the analysis of β-reduction and has proved quite fruitful in the analysis of optimal reduction strategies for the λ-calculus. GoI has also brought forward a new perspective for the semantics of computation which places it in a kind of "middle ground" between imperative/procedural, denotational/operational approaches to the semantics of programming languages. This new view helps to model the computational dynamics which is absent in denotational semantics and manages to offer a mathematical analysis which is lacking in operational semantics.

In Girard's terms, GoI provides a semantics free from the twin drawbacks of *reductionism* and *subjectivism*. In his approach, Girard uses operator algebras (specifically, the C^*-algebra of bounded linear operators on ℓ^2) to interpret multiplicatives, second order quantifiers and exponentials. Girard's *execution formula* $EX(u, \sigma) = (1 - \sigma^2)u(1 - \sigma u)^{-1}(1 - \sigma^2)$ gives the interpretation of the elimination of cuts (represented by σ) in a proof represented by u. Termination of the normalization/cut elimination process is interpreted as the nilpotency of σu, and the part $u(1 - \sigma u)^{-1}$ is the candidate for *computation*.

In subsequent work, Danos and Regnier [33, 35, 36, 99] have applied the execution formula to proof nets and λ-terms and presented the ideas in the GoI programme in terms of properties of paths on proof nets and λ-terms. Danos and Regnier's work has thus helped to illustrate cut elimination as a process of information flow in proof nets.

Abramsky and Jagadeesan gave an alternative implementation of the GoI programme based on the category of domains, where the execution formula was constructed using the notion of *feedback* in dataflow networks [5]. Subsequently Abramsky provided a categorical notion of *geometry of interaction situation* based on the concept of traced monoidal categories with additional structure. He showed how to define models of computation (combinatory algebras) from a geometry of interaction situation [2]. We will discuss this programme, which we call the Abramsky Programme, in more detail in Chapter 6. Abramsky introduced his programme in his Siena lecture notes [2]. In his notes, the programme was sketched and worked out for the specific example of the category of sets and partial injective functions.

One of the most active recent topics in theoretical computer science is the area of full completeness theorems. Full completeness establishes the tightest connection between the logic and the model under consideration: completeness at the level of proofs (=programs) rather than just traditional completeness with respect to provability. This is deeply connected to "Full Abstraction" theorems in programming language theory. From an algebraic point of view full completeness is related to fully-faithful representation theorems for logical categories (in our case, to representation theorems for various *-autonomous and tensor categorical models of linear logics.) We discuss full completeness problem in more detail in Chapter 10.

In this thesis we aim at achieving the following goals:

- To develop Abramsky's Programme in all details and in its most general form. More specifically, we shall provide the proofs for the passage from a GoI Situation to a combinatory algebra, as well as provide proofs for a more general case: the passage from symmetric monoidal closed categories with additional structure to combinatory algebras.

- To give an axiomatisation for the particle-style GoI, in the terminology of [1]. This will be a categorical setting which provides us with a computational (dataflow, token-pushing) analysis of the GoI programme and a computational calculus. For this purpose, we introduce *unique decomposition categories*, a generalisation of partially additive categories of Manes and Arbib [90] which provide the necessary algebraic structure for our purposes. Unique decomposition categories are tensor categories endowed with an additive structure.

- To give categorical models for GoI-based *path-semantics* of Danos and Regnier. This amounts to defining categorical models for dynamic algebras. In view of such categorical models, a path in a proof net becomes a morphism in the model category.

- To construct categorical models for the multiplicative fragment of linear logic based on our categories, that is partially additive and unique decomposition categories. More importantly to prove that such models are fully complete. In this way, we will have a whole new class of models for the multiplicative fragment of linear logic which are also tightly connected to syntax of linear logic in a non-trivial way (i.e., are fully complete.)

1.1 Outline of Thesis

The chapters of the thesis are organised as follows:

Chapter 2: Background

In this chapter we recall symmetric monoidal closed, compact closed and traced symmetric monoidal categories together with some examples. We also prove a new normal form theorem for arrows in traced categories. We also discuss two compact closure constructions on traced categories, namely the Int construction of Joyal, Street and Verity [67] and the $\mathcal{G}$ construction of Abramsky and prove that these yield isomorphic compact closed categories. We next review different fragments of linear logic and discuss their categorical models.

Chapter 3: Partially Additive Categories

We recall the partially additive categories (PAC) of Manes and Arbib [90] together with several examples. In particular, we discuss the category **SRel** of measurable spaces and stochastic kernels and prove that it is a partially additive category. This is followed by a discussion of the dagger (iteration) operation in a PAC. We prove that the dagger operation satisfies several important properties and show that every PAC is canonically traced, where the trace is induced by the dagger operation. We also prove that the trace in a PAC is uniform.

Chapter 4: Unique Decomposition Categories

Unique Decomposition Categories (UDCs) are introduced in this chapter. These categories are generalisations of PACs. We also discuss and prove important properties of such categories, relate them to PACs and discuss the existence of traces. This is followed by several examples that will be used later in the thesis.

Chapter 5: Geometry of Interaction

We review Girard's Geometry of Interaction (GoI) programme. We also discuss Abramsky and Jagadeesan's [5] domain theoretical implementation of GoI. This is followed by a detailed discussion of Abramsky's programme, that is the construction of models of computation from a traced category and more generally, from a symmetric monoidal closed category.

Chapter 6: Linear Combinatory Algebras

In this chapter we provide all the proofs necessary for the realisation of Abramsky's Programme. This includes the general case of this programme based on symmetric monoidal closed categories and the specific case of traced monoidal categories.

Chapter 7: Particle-style Semantics

In this chapter we work with UDCs that we introduced in Chapter 4. We show how to construct a combinatory algebra based on a traced UDC. This will provide us with an axiomatisation for the particle-style GoI.

Chapter 8: Examples

This chapter contains several examples of GoI Situations based on traced UDCs. We give a detailed discussion in the case of the category of sets and partial injective functions, including the definition of combinators as maps on the set of natural numbers. We also give examples of some computations with combinators in the context of traced UDCs to illustrate the computational aspect of our model.

Chapter 9: Dynamic Algebras

In this chapter we define categorical models for dynamic algebras which we call *dynamic situations*. We construct specific examples of dynamic situations and show that Girard's original operator algebraic model of a dynamic algebra [47] and Danos' *small model* [33] are dynamic situations obtained using the categories $\mathbf{Hilb}_2$ (Hilbert spaces and partial isometries induced by partial injective functions) and **PInj** (sets and partial injective functions) respectively.

Chapter 10: Full Completeness

In this chapter we give a review of the notion of full completeness in the context of categorical models, together with a brief historical review of the relevant work in the literature. We recall functorial polymorphism which will be our semantic setting in this chapter. We also recall the double glueing construction of Hyland and Tan [104] which will be used to obtain models of linear logic. The rest of the chapter contains the proof of full completeness of our models for the multiplicative fragment of linear logic with the MIX rule.

Chapter 11: Conclusions and Further Work

Finally, this chapter contains our concluding remarks and a discussion of possible directions for further study.

Appendix A: Diagrams for Trace Axioms

Graphical representation of trace axioms.

Appendix B: Derivation of GoI Combinators

This appendix contains the necessary diagrams and graphical calculus used to obtain the definition of combinators in a linear combinatory algebra based on a GoI Situation, from the general case.

Appendix C: Reduction to Normal Forms

More diagrams illustrating the reduction of combinators to their normal forms.

Chapter 2

Background

2.1 Traced Monoidal Categories

In this section we recall the definitions of monoidal, compact closed, *-autonomous and traced symmetric monoidal categories and state some properties of trace and a normal form theorem for arrows in traced symmetric monoidal categories which are new. We also discuss Abramsky's $\mathcal{G}$ construction and show that it is equivalent to the Int construction of Joyal, Street and Verity in the case of traced symmetric monoidal categories.

Monoidal (tensor) categories are found in the areas of logic [78, 86], mathematical physics [41], quantum groups and knot theory [71], computer science [103], etc. Here we recall the basic definitions. For details and notation see [85].

Definition 2.1.1 *A monoidal category* is a tuple $(\mathbb{C}, I, \otimes, \alpha, \lambda, \rho)$ consisting of a category $\mathbb{C}$, an object I of $\mathbb{C}$ called the *unit*, a bifunctor $\otimes : \mathbb{C} \times \mathbb{C} \longrightarrow \mathbb{C}$ (called the *tensor product*), and natural isomorphisms α, λ, ρ with components

$$\alpha_{A,B,C} : A \otimes (B \otimes C) \cong (A \otimes B) \otimes C \quad \lambda_A : I \otimes A \cong A \quad \rho_A : A \otimes I \cong A$$

such that the diagrams

$$
\begin{tikzcd}
A \otimes (B \otimes (C \otimes D)) \arrow[r, "\alpha"] \arrow[d, "1_A \otimes \alpha"'] & (A \otimes B) \otimes (C \otimes D) \arrow[r, "\alpha"] & ((A \otimes B) \otimes C) \otimes D \\
A \otimes ((B \otimes C) \otimes D) \arrow[rr, "\alpha"] & & (A \otimes (B \otimes C)) \otimes D \arrow[u, "\alpha \otimes 1_D"']
\end{tikzcd}
$$

and

$$
\begin{tikzcd}
A \otimes (I \otimes B) \arrow[rr, "\alpha"] \arrow[dr, "1_A \otimes \lambda"'] & & (A \otimes I) \otimes B \arrow[dl, "\rho \otimes 1_B"] \\
& A \otimes B &
\end{tikzcd}
$$

commute.

A *symmetric* monoidal category is a monoidal category equipped with a natural isomorphism σ with components $\sigma_{A,B} : A \otimes B \longrightarrow B \otimes A$ such that the diagram

$$
\begin{tikzcd}
A \otimes (B \otimes C) \arrow[r, "\alpha"] \arrow[d, "1_A \otimes \sigma"'] & (A \otimes B) \otimes C \arrow[r, "\sigma"] & C \otimes (A \otimes B) \arrow[d, "\alpha"] \\
A \otimes (C \otimes B) \arrow[r, "\alpha"] & (A \otimes C) \otimes B \arrow[r, "\sigma \otimes 1_B"] & (C \otimes A) \otimes B
\end{tikzcd}
$$

commutes and $\sigma_{B,A}\sigma_{A,B} = 1_{A\otimes B}$ and $\rho_A = \lambda_A \sigma_{A,I}$.

A monoidal category is *strict* when $\alpha_{A,B,C}, \lambda_A$, and ρ_A are identities for all objects A, B, and C of $\mathbb{C}$.

Familiar examples of symmetric monoidal categories are the category **Set** of sets and functions, the category **Top** of topological spaces and continuous maps, the category **Ab** of Abelian groups and group homomorphisms. For more examples see [85] and [28].

Definition 2.1.2 A symmetric monoidal category $\mathbb{C}$ is *closed* iff, for each object B in $\mathbb{C}$, the functor $- \otimes B : \mathbb{C} \longrightarrow \mathbb{C}$ has a specified right adjoint which we denote by $B \multimap - : \mathbb{C} \longrightarrow \mathbb{C}$. Hence we have an isomorphism $\mathbb{C}(A \otimes B, C) \cong \mathbb{C}(A, B \multimap C)$ which is natural for all A and C in $\mathbb{C}$.

Familiar examples include the category **Vec** of vector spaces and linear transformations and **Ab**, etc., for more examples see [85, 28]. Cartesian closed categories (CCCs) are important examples of closed categories. CCCs have many applications in logic, proof theory [78], topos theory [86] and they play a fundamental role in the semantics of programming languages [103]. Examples of CCCs include the category **Set**, the functor category $\mathbf{Set}^{\mathbb{C}^{op}}$ of presheaves on $\mathbb{C}$, the category $\mathbf{Sh}(X)$ of sheaves on a topological space X.

The following definition of a compact closed category is due to Kelly and Laplaza [73].

Definition 2.1.3 A *compact closed category* is a symmetric monoidal category in which every object A has a left dual A^*, i.e. there exist a *unit* $\eta_A : I \longrightarrow A \otimes A^*$ and a *counit* $\epsilon_A : A^* \otimes A \longrightarrow I$ such that the following triangles commute.

$$\begin{tikzcd}
A \cong I \otimes A \arrow[r, "\eta_A \otimes 1_A"] \arrow[dr, "1_A"'] & A \otimes A^* \otimes A \arrow[d, "1_A \otimes \epsilon_A"] \\
& A \otimes I \cong A
\end{tikzcd}$$

$$\begin{tikzcd}
A^* \cong A^* \otimes I \arrow[r, "1_{A^*} \otimes \eta_A"] \arrow[dr, "1_{A^*}"'] & A^* \otimes A \otimes A^* \arrow[d, "\epsilon_A \otimes 1_{A^*}"] \\
& I \otimes A^* \cong A^*
\end{tikzcd}$$

Example 2.1.4

The category **Rel** of sets and relations is a compact closed category. The tensor product is given by the cartesian product of sets and the tensor unit is the one element set. $A^* = A$ for a set A and given $R : A \longrightarrow B$, $R^* : B \longrightarrow A$ is the converse relation. Unit and counit of adjunction can be viewed as subsets of $A \times A$ and in particular they are the identity relation on A. $\square$

Example 2.1.5

The category **FDVec** of finite dimensional vector spaces over a field k and linear transformations is a compact closed category. Tensor product is given by the algebraic tensor product of vector spaces and the unit is the field k. V^* is the set of linear functionals on V. Given a linear transformation $f : V \longrightarrow W$, $f^* : W^* \longrightarrow V^*$ is defined by $f^*(g) = gf$ for $g \in W^*$. Given V with basis $\{e_i\}_{i=1}^n$ and dual basis $\{\varepsilon_j\}_{j=1}^n$ for V^* (i.e. $\varepsilon_j(e_i) = 1$ if $i = j$ and 0 otherwise), $\eta_V : k \longrightarrow V \otimes V^*$ is given by

$$\eta_V(1) = \sum_{i=1}^{n} e_i \otimes \varepsilon_i.$$

The counit $\epsilon_V : V^* \otimes V \longrightarrow k$ is given by

$$\epsilon_V(\theta \otimes v) = \theta(v).$$

$\square$

We will encounter examples of compact closed categories constructed by certain compact closure operator, later in the thesis.

Definition 2.1.6 A **-autonomous category* $\mathbb{C}$ is a symmetric monoidal category such that there exists a full and faithful functor $(-)^\perp : \mathbb{C}^{op} \longrightarrow \mathbb{C}$ with $\mathbb{C}(A \otimes B, C^\perp) \xrightarrow{\cong} \mathbb{C}(A, (B \otimes C)^\perp)$, defined and natural for all objects A, B and C in $\mathbb{C}$.

*-autonomous categories were introduced by Barr in [14]. See also [15, 17]. The following result is folklore.

Theorem 2.1.7 *A *-autonomous category may be equivalently specified as a symmetric monoidal closed category* $(\mathbb{C}, \otimes, I, \multimap)$ *with an object* $\perp$ *(called the* dualising object*) such that the canonical morphism* $A \longrightarrow (A \multimap \perp) \multimap \perp$ *is an isomorphism, for all objects* A *in* $\mathbb{C}$.

Example 2.1.8

The category **SL** of complete sup semilattices is *-autonomous. The objects are complete lattices and morphisms are order and sup preserving maps. Given objects X and Y, the homset $\mathbf{SL}(X, Y)$ is itself a complete lattice with pointwise order and sup. Let $\mathbf{2}$ be the two element lattice, $0 < 1$. Then any $f_\alpha : X \longrightarrow \mathbf{2}$ satisfies

$$f_\alpha(x) = \begin{cases} 0 & \text{if } x \leq x_\alpha \\ 1 & \text{otherwise.} \end{cases}$$

for some unique element x_α in X (namely $x_\alpha = sup\{x | f_\alpha(x) = 0\}$.) Thus there is a bijection between elements x_α in X and elements $f_\alpha \in \mathbf{SL}(X, \mathbf{2})$. Furthermore if $x_\alpha \leq x_\beta$ then $f_\beta \leq f_\alpha$. Therefore, $\mathbf{SL}(X, \mathbf{2}) \cong X^{op}$, the lattice with the same elements as X and the order reversed. Hence **SL** is *-autonomous with autonomous structure induced by the hom functor and dualising object **2**. □

The examples **Rel** and **FDVec** we saw above are also *-autonomous. In fact,

Proposition 2.1.9 *A category* $\mathbb{C} = (\mathbb{C}, \otimes, I, (-)^*)$ *is compact closed iff it is *-autonomous and* $(B \otimes C)^* \cong B^* \otimes C^*$ *for all objects* B *and* C *in* $\mathbb{C}$.

An interesting and important example of a *-autonomous category can be obtained by considering the reflexive objects in topological vector spaces [14]. More specifically, consider the category **TVec** whose objects are vector spaces equipped with linear topologies, and whose morphisms are linear continuous maps. Barr showed that **TVec** is a symmetric monoidal closed category, when $V \multimap W$ is defined to be the vector space of linear continuous maps, with pointwise convergence topology. Let $V^\perp$ denote $V \multimap k$ where k is the underlying field. Lefschetz proved that the embedding

$V \longrightarrow V^{\perp\perp}$ is always a bijection, but need not be an isomorphism. Barr proved that the full subcategory **RTVec** of reflexive objects in **TVec** is a *-autonomous category.

For more examples of *-autonomous categories see [15, 17, 24]. Later in the thesis we will see more examples of *-autonomous categories that can be obtained from compact closed categories using a *double glueing* construction (see Chapter 10.)

We continue by recalling some more facts on *-autonomous and compact closed categories that will be of use later in the thesis. Our presentation here follows [104].

Theorem 2.1.10 *Let $\mathbb{C} = (\mathbb{C}, \otimes, I, (-)^{\perp})$ be a *-autonomous category. Then there exists another bifunctor $\parr : \mathbb{C} \times \mathbb{C} \longrightarrow \mathbb{C}$, called the* par *product, with unit $\perp$, which is dual to the tensor product in the sense of De Morgan. That is, there exist isomorphisms*

$$(A \otimes B)^{\perp} \xrightarrow{\cong} A^{\perp} \parr B^{\perp}$$

or equivalently

$$(A \parr B)^{\perp} \xrightarrow{\cong} A^{\perp} \otimes B^{\perp},$$

defined and natural for all objects A and B in $\mathbb{C}$.

It follows easily that $A \parr B \cong A^{\perp} \multimap B$ for all A, B in $\mathbb{C}$.

Theorem 2.1.11 *Let $\mathbb{C}$ be a *-autonomous category. Then $\mathbb{C}$ possesses* weak distributivity *natural transformations*

$$w^{L}_{ABC} : A \otimes (B \parr C) \longrightarrow (A \otimes B) \parr C$$

$$w^{R}_{ABC} : A \otimes (B \parr C) \longrightarrow (A \otimes C) \parr B$$

such that a number of coherence diagrams commute.

A discussion on weakly distributive categories can be found in [31].

Theorem 2.1.12 (Cockett and Seely[31]) *The notions of symmetric weakly distributive categories with negation and *-autonomous categories coincide.*

The term "with negation" means the addition of two families of maps

$$\eta_A : I \longrightarrow A^\perp \parr A \qquad \epsilon_A : A \otimes A^\perp \longrightarrow \perp$$

subject to a number of coherence equations.

Proposition 2.1.13 *Let $\mathbb{C} = (\mathbb{C}, \otimes, I, (-)^*)$ be a compact closed category. Then, $A \parr B \cong A \otimes B$ for all objects A and B in $\mathbb{C}$ and $I^* \cong I$.*

Definition 2.1.14 A *monoidal functor* from a monoidal category $\mathbb{C}$ to a monoidal category $\mathbb{D}$ is a triple (F, φ, φ_I) where $F : \mathbb{C} \longrightarrow \mathbb{D}$ is a functor, φ is a natural transformation with components $\varphi_{A,B} : FA \otimes FB \longrightarrow F(A \otimes B)$ and $\varphi_I : I \longrightarrow FI$ is a $\mathbb{D}$-morphism such that the diagrams

$$\begin{tikzcd}
FA \otimes (FB \otimes FC) \arrow[r, "1 \otimes \varphi"] \arrow[d, "\alpha"'] & FA \otimes F(B \otimes C) \arrow[r, "\varphi"] & F(A \otimes (B \otimes C)) \arrow[d, "F\alpha"] \\
(FA \otimes FB) \otimes FC \arrow[r, "\varphi \otimes 1"] & F(A \otimes B) \otimes FC \arrow[r, "\varphi"] & F((A \otimes B) \otimes C)
\end{tikzcd}$$

and

$$\begin{tikzcd}
I \otimes FA \arrow[r, "\lambda"] \arrow[d, "\varphi_I \otimes 1"'] & FA \\
FI \otimes FA \arrow[r, "\varphi"] & F(I \otimes A) \arrow[u, "F\lambda"']
\end{tikzcd}
\qquad
\begin{tikzcd}
FA \otimes I \arrow[r, "\rho"] \arrow[d, "1 \otimes \varphi_I"'] & FA \\
FA \otimes FI \arrow[r, "\varphi"] & F(A \otimes I) \arrow[u, "F\rho"']
\end{tikzcd}$$

commute. A monoidal functor is said to be *strong* when φ is a natural isomorphism and φ_I is an isomorphism, and *strict* when φ and φ_I are identities. This means essentially that F preserves the monoidal structure on the nose.

A monoidal functor $F : \mathbb{C} \longrightarrow \mathbb{D}$ with $\mathbb{C}$ and $\mathbb{D}$ symmetric monoidal categories is *symmetric* if the following diagram commutes:

$$\begin{tikzcd} FA \otimes FB \arrow[r, "\varphi_{A,B}"] \arrow[d, "\sigma_{FA,FB}"'] & F(A \otimes B) \arrow[d, "F\sigma_{A,B}"] \\ FB \otimes FA \arrow[r, "\varphi_{B,A}"] & F(B \otimes A) \end{tikzcd}$$

A monoidal natural transformation m from (F, φ, φ_I) to (G, ψ, ψ_I) is a natural transformation $m : F \Longrightarrow G$ such that the following diagrams commute:

$$\begin{tikzcd} FA \otimes FB \arrow[r, "\varphi_{A,B}"] \arrow[d, "m_A \otimes m_B"'] & F(A \otimes B) \arrow[d, "m_{A\otimes B}"] \\ GA \otimes GB \arrow[r, "\psi_{A,B}"] & G(A \otimes B) \end{tikzcd} \qquad \begin{tikzcd} I \arrow[r, "\varphi_I"] \arrow[dr, "\psi_I"'] & FI \arrow[d, "m_I"] \\ & GI \end{tikzcd}$$

A monoidal *pointwise* natural transformation is a family of maps $m_A : FA \longrightarrow GA$ such that the naturality diagram commutes for morphisms of the form $f : I \longrightarrow A$, i.e. we have $Gfm_I = m_A Ff$, for $f : I \longrightarrow A$, A an object of $\mathbb{C}$.

In the sequel and throughout the thesis, monoidal functor will mean strong monoidal functor, unless explicitly stated otherwise.

Definition 2.1.15 A *comonoid* in a monoidal category $\mathbb{C}$ is a triple (C, e, d) with C an object of $\mathbb{C}$ and $e : C \longrightarrow I$ and $d : C \longrightarrow C \otimes C$ morphisms in $\mathbb{C}$ such that the following diagrams commute:

$$\begin{tikzcd} C \otimes C \arrow[d, "1_C \otimes d"'] & C \arrow[l, "d"'] \arrow[r, "d"] & C \otimes C \arrow[d, "d \otimes 1_C"] \\ C \otimes (C \otimes C) \arrow[rr, "\alpha"] & & (C \otimes C) \otimes C \end{tikzcd}$$

$$\begin{tikzcd} & C \arrow[dl, "\lambda^{-1}"'] \arrow[d, "d"] \arrow[dr, "\rho^{-1}"] & \\ I \otimes C & C \otimes C \arrow[l, "e \otimes 1_C"'] \arrow[r, "1_C \otimes e"] & C \otimes I \end{tikzcd}$$

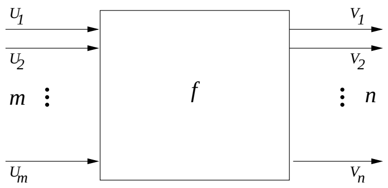


Figure 1: Graphical Representation of f

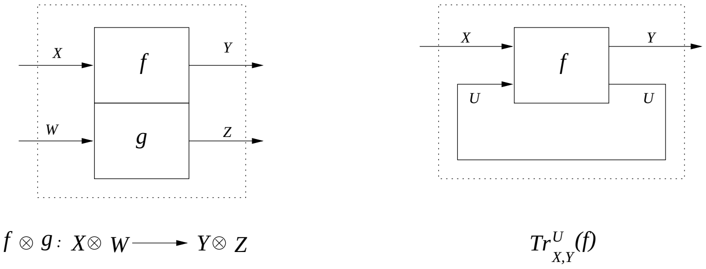


Figure 2: I/O Interfaces

The comonoid C is said to be *commutative* iff $\mathbb{C}$ is symmetric monoidal and

$$\begin{tikzcd} C \arrow[r, "d"] \arrow[dr, "d"'] & C \otimes C \arrow[d, "\sigma"] \\ & C \otimes C \end{tikzcd}$$

commutes.

For readability and without loss of generality we consider strict monoidal categories. It is well known that every monoidal category is equivalent to a strict one [85]. This is in fact equivalent to Mac Lane's coherence theorem for monoidal categories (see [85], XI.3)

Notation: We introduce the following graphical notation: an arrow $U_1 \otimes \cdots \otimes U_m \xrightarrow{f} V_1 \otimes \cdots \otimes V_n$ is represented as a box as in Figure 1. We sometimes emphasize the I/O (Input/Output) interface of a tensor or a trace using a dotted box as in Figure 2. We omit writing the labels on the lines when it is clear. Identity is represented by a straight line and symmetry by crossed lines.

Joyal, Street and Verity [67] introduced the notion of abstract trace on a balanced monoidal category (a monoidal category with braidings and twists.) This trace can be interpreted in various contexts where it could be called contraction, feedback, Markov trace or braid closure. The notion of trace is used to analyse the cyclic structures encountered in mathematics and physics, most notably in knot theory. Since their introduction, traced monoidal categories have found applications in different areas in computer science, for example model theory of cyclic lambda calculi [59] and categorical frameworks for the semantics of asynchronous communication in networks of parallel processes [102]. In this thesis we will see another example of the application of traced monoidal categories to construct categorical models for the analysis of information flow in proofs of linear logic (GoI). For the purposes of this thesis we will be interested in traced *symmetric* monoidal case, that is we take the braiding to be the symmetry.

Definition 2.1.16 A *traced symmetric monoidal category* is a symmetric monoidal category $(\mathbb{C}, I, \otimes, \sigma)$ with a family of functions $Tr^U_{X,Y} : \mathbb{C}(X \otimes U, Y \otimes U) \longrightarrow \mathbb{C}(X, Y)$ pictured in Figure 2, called a *trace*, subject to the following axioms:

- **Natural** in X, $Tr^U_{X,Y}(f)g = Tr^U_{X',Y}(f(g \otimes 1_U))$ where $f : X \otimes U \longrightarrow Y \otimes U$, $g : X' \longrightarrow X$,
- **Natural** in Y, $gTr^U_{X,Y}(f) = Tr^U_{X,Y'}((g \otimes 1_U)f)$ where $f : X \otimes U \longrightarrow Y \otimes U$, $g : Y \longrightarrow Y'$,
- **Dinatural** in U, $Tr^U_{X,Y}((1_Y \otimes g)f) = Tr^{U'}_{X,Y}(f(1_X \otimes g))$ where $f : X \otimes U \longrightarrow Y \otimes U'$, $g : U' \longrightarrow U$,
- **Vanishing (I,II)**, $Tr^I_{X,Y}(f) = f$ and $Tr^{U\otimes V}_{X,Y}(g) = Tr^U_{X,Y}(Tr^V_{X\otimes U,Y\otimes U}(g))$ for $f : X \otimes I \longrightarrow Y \otimes I$ and $g : X \otimes U \otimes V \longrightarrow Y \otimes U \otimes V$,
- **Superposing**,

  $$Tr^U_{X,Y}(f) \otimes g = Tr^U_{X\otimes W,Y\otimes Z}((1_Y \otimes \sigma_{U,Z})(f \otimes g)(1_X \otimes \sigma_{W,U}))$$

  for $f : X \otimes U \longrightarrow Y \otimes U$ and $g : W \longrightarrow Z$,
- **Yanking**, $Tr^U_{U,U}(\sigma_{U,U}) = 1_U$.

See Appendix A for a graphical representation of trace axioms.

A monoidal functor $F = (F, \varphi, \varphi_I) : \mathbb{C} \longrightarrow \mathbb{D}$ with $\mathbb{C}$ and $\mathbb{D}$ traced symmetric monoidal categories is *traced* if it is symmetric and satisfies

$$Tr^{FU}_{FA,FB}(\varphi^{-1}_{B,U}(Ff)\varphi_{A,U}) \quad = \quad F(Tr^{U}_{A,B}(f))$$

where $A \otimes U \xrightarrow{f} B \otimes U$ and $FA \otimes FU \xrightarrow{\varphi_{A,U}} F(A \otimes U) \xrightarrow{Ff} F(B \otimes U) \xrightarrow{\varphi^{-1}_{B,U}} FB \otimes FU$. In the case of a strict functor this means that F preserves the trace on the nose.

Example 2.1.17

1. The category **Rel** is traced. Let $R : X \times U \longrightarrow Y \times U$ be a morphism in **Rel**. Then $Tr^U_{X,Y}(R) : X \longrightarrow Y$ is defined by: $Tr^U_{X,Y}(R)(x, y) = \exists u.R(x, u, y, u)$.

2. The category **FDVec** is traced. Given a linear transformation $f : V \otimes U \longrightarrow W \otimes U$ where U, V, W are vector spaces with bases $\{u_i\}, \{v_j\}, \{w_k\}$. $Tr^U_{V,W}(f) : V \longrightarrow W$ is given by

$$Tr^U_{V,W}(f)(v_i) = \sum_{j,k} a^{kj}_{ij} w_k \qquad \text{where } f(v_i \otimes u_j) = \sum_{k,m} a^{km}_{ij} w_k \otimes u_m.$$

This reduces to the usual trace of $f : U \longrightarrow U$ when V and W are one dimensional.
3. Note that both **Rel** and **FDvec** are compact closed categories. More generally, every compact closed category is canonically traced as follows [67]: given $f : A \otimes U \longrightarrow B \otimes U$ in a compact closed category $\mathbb{C}$,

$Tr^U_{A,B}(f) =$

$$A \xrightarrow{1 \otimes \eta_U} A \otimes U \otimes U^* \xrightarrow{f \otimes 1_{U^*}} B \otimes U \otimes U^* \xrightarrow{1 \otimes \sigma} B \otimes U^* \otimes U \xrightarrow{1_U \otimes \epsilon_U} B$$

$\square$

Proposition 2.1.18 *The set of axioms for traced symmetric monoidal categories given above is equivalent to one where superposing is replaced by the following axiom.*

$$g \otimes Tr^U_{X,Y}(f) = Tr^U_{W \otimes X, Z \otimes Y}(g \otimes f)$$

for $f : X \otimes U \longrightarrow Y \otimes U$ *and* $g : W \longrightarrow Z$ *.*

Proof. Let $\mathbb{C}$ be a symmetric monoidal category with a trace operator subject to axioms naturality, dinaturality, vanishing and yanking. Then the following are equivalent:

$$(1) \qquad Tr^U_{X,Y}(f) \otimes g = Tr^U_{X \otimes W, Y \otimes Z}((1_Y \otimes \sigma_{U,Z})(f \otimes g)(1_X \otimes \sigma_{W,U}))$$

$$(2) \qquad g \otimes Tr^U_{X,Y}(f) = Tr^U_{W \otimes X, Z \otimes Y}(g \otimes f)$$

for $f : X \otimes U \longrightarrow Y \otimes U$ and $g : W \longrightarrow Z$. We will refer to (1) and (2) above as Superposing axiom.

(1) $\Longrightarrow$ (2):

$$\begin{aligned}
g \otimes Tr^U_{X,Y}(f) &= \sigma_{Y,Z}(Tr^U_{X,Y}(f) \otimes g)\sigma_{W,X} \\
&= \sigma_{Y,Z}(Tr^U_{X \otimes W, Y \otimes Z}((1_Y \otimes \sigma_{U,Z})(f \otimes g)(1_X \otimes \sigma_{W,U})))\sigma_{W,X} \\
&= Tr^U_{W \otimes X, Z \otimes Y}((\sigma_{Y,Z} \otimes 1_U)(1_Y \otimes \sigma_{U,Z})(f \otimes g)(1_X \otimes \sigma_{W,U})(\sigma_{W,X} \otimes 1_U)) \\
&= Tr^U_{W \otimes X, Z \otimes Y}((\sigma_{Y,Z} \otimes 1_U)(1_Y \otimes \sigma_{U,Z})(1_Y \otimes \sigma_{Z,U})(\sigma_{Z,Y} \otimes 1_U)(g \otimes f) \\
&\quad (\sigma_{X,W} \otimes 1_U)(1_X \otimes \sigma_{U,W})(1_X \otimes \sigma_{W,U})(\sigma_{W,X} \otimes 1_U)) \\
&= Tr^U_{W \otimes X, Z \otimes Y}(g \otimes f).
\end{aligned}$$

(2) $\Longrightarrow$ (1):

$$\begin{aligned}
Tr^U_{X,Y}(f) \otimes g &= \sigma_{Z,Y}(g \otimes Tr^U_{X,Y}(f))\sigma_{X,W} \\
&= \sigma_{Z,Y}(Tr^U_{W \otimes X, Z \otimes Y}(g \otimes f))\sigma_{X,W} \\
&= Tr^U_{X \otimes W, Y \otimes Z}((\sigma_{Z,Y} \otimes 1_U)(g \otimes f)(\sigma_{X,W} \otimes 1_U)) \\
&= Tr^U_{X \otimes W, Y \otimes Z}((\sigma_{Z,Y} \otimes 1_U)(\sigma_{Y,Z} \otimes 1_U)(1_Y \otimes \sigma_{U,Z})(f \otimes g) \\
&\quad (1_X \otimes \sigma_{W,U})(\sigma_{W,X} \otimes 1_U)(\sigma_{X,W} \otimes 1_U)) \\
&= Tr^U_{X \otimes W, Y \otimes Z}((1_Y \otimes \sigma_{U,Z})(f \otimes g)(1_X \otimes \sigma_{W,U}))
\end{aligned}$$

$\square$

Remark 2.1.19 Following [67] we will mainly use geometric proofs. Two-dimensional reasoning is valid on the progressive parts of the diagrams because of the results of [66] for symmetric monoidal categories. The reasoning in parts involving trace is deduced from the axioms of trace. With these provisos, geometric reasoning is completely rigorous. Indeed, as remarked in [67], p. 450, "Algebraic proofs can be constructed from the geometric ones, but algebraic proofs seem only to obfuscate the intuition." As an example, we will give algebraic and geometric proofs for the next proposition.

Proposition 2.1.20 *(Generalized Yanking) Let $\mathbb{C}$ be a traced symmetric monoidal category. Let $f : X \longrightarrow Y$ and $g : Y \longrightarrow Z$ be given. Then*

$$gf = Tr^Y_{X,Z}(\sigma_{Y,Z}(f \otimes g)).$$

Proof.

$$\begin{aligned} gf &= g1_Y f && (1)\\ &= gTr^Y_{Y,Y}(\sigma_{Y,Y})f && (2)\\ &= Tr^Y_{Y,Z}((g \otimes 1_Y)\sigma_{Y,Y})f && (3)\\ &= Tr^Y_{X,Z}((g \otimes 1_Y)\sigma_{Y,Y}(f \otimes 1_Y)) && (4)\\ &= Tr^Y_{X,Z}(\sigma_{Y,Z}(1_Y \otimes g)(f \otimes 1_Y)) && (5)\\ &= Tr^Y_{X,Z}(\sigma_{Y,Z}(f \otimes g)). && (6) \end{aligned}$$

Equations 1,2,3,4,5, and 6 respectively correspond to the diagrams in Figure 3. For clarity, we draw 1_Y as a straight line, omitting the box.

□

In particular one can give an equivalent axiomatisation of traced symmetric monoidal categories with Yanking replaced by Generalized Yanking.

Proposition 2.1.21 *The set of axioms for traced symmetric monoidal categories given above is equivalent to one where Yanking is replaced by the following axiom.*

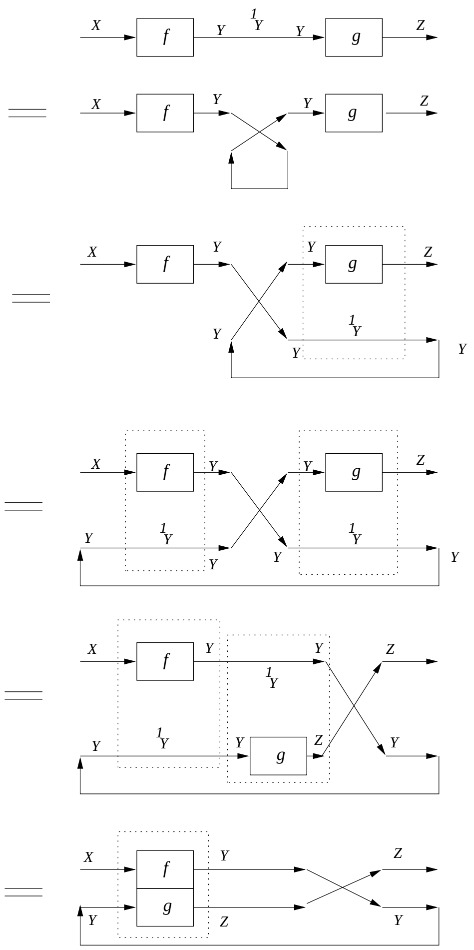


Figure 3: Graphical Proof of Proposition 2.1.20

Generalised yanking:

$$gf = Tr^{Y}_{X,Z}(\sigma_{Y,Z}(f \otimes g))$$

for $f : X \longrightarrow Y$ *and* $g : Y \longrightarrow Z$.

Proof. Let $\mathbb{C}$ be a symmetric monoidal category with a trace operator subject to axioms naturality, dinaturality, vanishing and superposing. Then the following are equivalent:

(1) $\qquad Tr^{U}_{U,U}(\sigma_{U,U}) = 1_U$

(2) $gf = Tr^{Y}_{X,Z}(\sigma_{Y,Z}(f \otimes g))$

for $f : X \longrightarrow Y$ and $g : Y \longrightarrow Z$.

(1) $\Longrightarrow$ (2): Proposition 2.1.20.

(2) $\Longrightarrow$ (1): Let X, Y, Z be U and $f = g = 1_U$, then

$$Tr^{U}_{U,U}(\sigma_{U,U}) = Tr^{U}_{U,U}(\sigma_{U,U}(f \otimes g)) = 1_U$$

□

We now give a normal form theorem for arrows in a traced symmetric monoidal category.

Theorem 2.1.22 *Let $\mathbb{C}$ be a traced symmetric monoidal category and T be a set of arrows in $\mathbb{C}$. Then, any expression E built from arrows in T using the tensor product, composition, and trace can be represented as $Tr(\pi F \tau)$ where F consists of a tensor product of arrows in T and π and τ are constructed from symmetry and identity maps using composition and tensor (i.e., π and τ are permutations.)*

Proof. By induction on the structure of the expression E:

- *Basis step :* Let $E = f : X \longrightarrow Y$ in T. Then $f = Tr^{I}_{X,Y}(f)$.
- Let $E = E_1 \otimes E_2$ with $E_1 : X \longrightarrow Y$ and $E_2 : X' \longrightarrow Y'$, by inductive hypothesis, $E_1 = Tr^{U}_{X,Y}(\pi_1 F_1 \tau_1)$ and $E_2 = Tr^{U'}_{X',Y'}(\pi_2 F_2 \tau_2)$. Hence

  $E_1 \otimes E_2 = Tr^{U}_{X,Y}(\pi_1 F_1 \tau_1) \otimes Tr^{U'}_{X',Y'}(\pi_2 F_2 \tau_2)$

  $= Tr^{U'}_{X\otimes X',Y\otimes Y'}(Tr^{U}_{X,Y}(\pi_1 F_1 \tau_1) \otimes (\pi_2 F_2 \tau_2))$ using Superposing,

  $= Tr^{U'}_{X\otimes X',Y\otimes Y'}(Tr^{U}_{X\otimes X'\otimes U',Y\otimes Y'\otimes U'}((1_Y \otimes \sigma_{U,Y'\otimes U'})(\pi_1 F_1 \tau_1 \otimes \pi_2 F_2 \tau_2)$

  $(1_X \otimes \sigma_{X'\otimes U',U})))$ using Superposing,

$= Tr^{U' \otimes U}_{X \otimes X', Y \otimes Y'}((1_Y \otimes \sigma_{U, Y' \otimes U'})(\pi_1 F_1 \tau_1 \otimes \pi_2 F_2 \tau_2)(1_X \otimes \sigma_{X' \otimes U', U}))$ using Vanishing II,
$= Tr^{U' \otimes U}_{X \otimes X', Y \otimes Y'}(\pi F \tau)$

where $\pi = (1_Y \otimes \sigma_{U,Y' \otimes U'})(\pi_1 \otimes \pi_2)$, $F = F_1 \otimes F_2$ and $\tau = (\tau_1 \otimes \tau_2)(1_X \otimes \sigma_{X' \otimes U', U})$.

- Let $E = E_2 E_1 : X \longrightarrow Z$ where $E_1 : X \longrightarrow Y$ and $E_2 : Y \longrightarrow Z$.

$$\begin{aligned} E = E_1 E_2 &= Tr^{Y}_{X,Z}(\sigma_{Y,Z}(E_1 \otimes E_2)) \text{ by Proposition 2.1.20,} \\ &= Tr^{Y}_{X,Z}(\sigma_{Y,Z} Tr^{U}_{X \otimes Y, Y \otimes Z}(\pi F \tau)) \text{ using previous part,} \\ &= Tr^{Y}_{X,Z}(Tr^{U}_{X \otimes Y, Z \otimes Y}((\sigma_{Y,Z} \otimes 1_U)(\pi F \tau))) \text{ using Naturality,} \\ &= Tr^{Y \otimes U}_{X,Z}(\pi' F \tau) \text{ using Vanishing II.} \end{aligned}$$

where $\pi' = (\sigma_{Y,Z} \otimes 1_U)\pi$.

- Let $E = Tr^{U}_{X,Y}(E_1)$ for $E_1 : X \otimes U \longrightarrow Y \otimes U$.

$$\begin{aligned} E &= Tr^{U}_{X,Y}(E_1) \\ &= Tr^{U}_{X,Y}(Tr^{U'}_{X \otimes U, Y \otimes U}(\pi F \tau)) \text{ by inductive hypothesis} \\ &= Tr^{U \otimes U'}_{X,Y}(\pi F \tau) \text{ using Vanishing II.} \end{aligned}$$

$\square$

2.2 Int Construction

The Int construction was introduced by Joyal, Street and Verity in [67]. It is used to construct a free tortile monoidal category from a given traced balanced monoidal category. In this thesis we will work with traced symmetric monoidal categories and hence in this case the main result of [67] reads as follows. Recall from Example 2.1.15 that every compact closed category is canonically traced, given $f : A \otimes U \longrightarrow B \otimes U$ in a compact closed category $\mathbb{C}$,

$Tr^{U}_{A,B}(f) =$

$$A \xrightarrow{1 \otimes \eta_U} A \otimes U \otimes U^* \xrightarrow{f \otimes 1_{U^*}} B \otimes U \otimes U^* \xrightarrow{1 \otimes \sigma} B \otimes U^* \otimes U \xrightarrow{1_U \otimes \epsilon_U} B$$

and every monoidal full subcategory of a compact closed category is traced. Moreover, every traced symmetric monoidal category $\mathbb{C}$ arises in this way, that is it is a monoidal full subcategory of a compact closed category, namely $Int\,\mathbb{C}$. Informally, $Int\,\mathbb{C}$ is the "free compact closure" of $\mathbb{C}$. More formally:

Theorem 2.2.1 (Joyal, Street & Verity) [1] *Suppose $\mathbb{C}$ is a traced symmetric monoidal category and $\mathbb{D}$ is a compact closed category. Then there exists a compact closed category $Int\,\mathbb{C}$ such that for all traced monoidal functors $F : \mathbb{C} \longrightarrow \mathbb{D}$, there exists a symmetric monoidal functor $K : Int\,\mathbb{C} \longrightarrow \mathbb{D}$ which is unique up to monoidal natural isomorphism with the property $KN \cong F$, where $N : \mathbb{C} \longrightarrow Int\,\mathbb{C}$ is the full faithful inclusion functor.*

We next proceed by recalling the Int construction which is motivated by the construction of integers $\mathbb{Z}$ from natural numbers $\mathbb{N}$ and hence the name. Let $\mathbb{C}$ be a traced symmetric monoidal category. The category $Int\,\mathbb{C}$ is defined as follows:

- Objects: Pairs of objects from $\mathbb{C}$, e.g. (A^+, A^-) where A^+ and A^- are objects of $\mathbb{C}$.
- Arrows: An arrow $f : (A^+, A^-) \longrightarrow (B^+, B^-)$ in $Int\,\mathbb{C}$ is $f : A^+ \otimes B^- \longrightarrow B^+ \otimes A^-$ in $\mathbb{C}$.
- Identity: $1_{(A^+,A^-)} = 1_{A^+ \otimes A^-}$.
- Composition: Given $f : (A^+, A^-) \longrightarrow (B^+, B^-)$ and $g : (B^+, B^-) \longrightarrow (C^+, C^-)$, $gf : (A^+, A^-) \longrightarrow (C^+, C^-)$ is given by:

  $gf = Tr^{B^-}_{A^+ \otimes C^-, C^+ \otimes A^-}((1_{C^+} \otimes \sigma_{B^-,A^-})(g \otimes 1_{A^-})(1_{B^+} \otimes \sigma_{A^-,C^-})(f \otimes 1_{C^-})\ (1_{A^+} \otimes \sigma_{C^-,B^-}))$

  This can be represented graphically as in Figure 4.
- Tensor: $(A^+, A^-) \otimes (B^+, B^-) = (A^+ \otimes B^+, B^- \otimes A^-)$ and for $f : (A^+, A^-) \longrightarrow (B^+, B^-)$ and $g : (C^+, C^-) \longrightarrow (D^+, D^-)$, $f \otimes g$ is defined to be the following

[1] *Note that this is the version of the original theorem for the case of symmetric monoidal categories.*

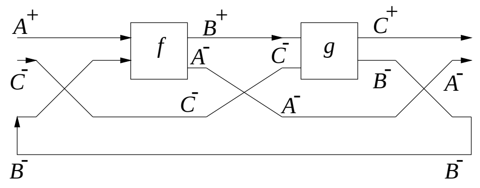


Figure 4: Composition in $Int\,\mathbb{C}$

composite: $A^+ \otimes C^+ \otimes D^- \otimes B^- \xrightarrow{\sigma\otimes\sigma} C^+ \otimes A^+ \otimes B^- \otimes D^- \xrightarrow{1\otimes f\otimes 1} C^+ \otimes B^+ \otimes A^- \otimes D^- \xrightarrow{\sigma\otimes\sigma} B^+ \otimes C^+ \otimes D^- \otimes A^- \xrightarrow{1\otimes g\otimes 1} B^+ \otimes D^+ \otimes C^- \otimes A^-$

- Unit: (I, I).

Proposition 2.2.2 *Let $\mathbb{C}$ be a traced symmetric monoidal category , $Int\,\mathbb{C}$ is a compact closed category. Moreover, $N : \mathbb{C} \longrightarrow Int\,\mathbb{C}$ with $N(A) = (A, I)$ and $N(f) = f$ is a full and faithful embedding.*

Proof. (Sketch) This is just a specialisation of the proof that appears in [67] and we will not repeat it here. However we give the main morphisms of the closed structure. For any two objects (A^+, A^-) and (B^+, B^-) in $Int\,\mathbb{C}$,

$$\sigma_{(A^+,A^-),(B^+,B^-)} =_{def} \sigma_{A^+,B^+} \otimes \sigma_{A^-,B^-}.$$

The left dual of (A^+, A^-), $(A^+, A^-)^* = (A^-, A^+)$. The unit is given by

$$\eta_{(A^+,A^-)} : (I, I) \longrightarrow (A^+, A^-) \otimes (A^+, A^-)^* =_{def} 1_{A^+\otimes A^-}$$

and counit is

$$\epsilon_{(A^+,A^-)} : (A^+, A^-)^* \otimes (A^+, A^-) \longrightarrow (I, I) =_{def} 1_{A^-\otimes A^+}.$$

The internal homs are given by

$$(A^+, A^-) \multimap (B^+, B^-) = (B^+, B^-) \otimes (A^+, A^-)^* = (B^+ \otimes A^-, A^+ \otimes B^-).$$

$\square$

2.3 $\mathcal{G}$ Construction

In this section we recall Abramsky's $\mathcal{G}$ construction [1]. It was pointed out in [1] that this construction captures the Geometry of Interaction construction for the multiplicative fragment of linear logic and that Girard's execution formula corresponds to the composition in the $\mathcal{G}(\mathbb{C})$ category. We will use this construction extensively in the first part of the thesis. We will hence describe this construction and show that it is equivalent to the Int construction of the previous section.

Definition 2.3.1 (The *Geometry of Interaction construction*) Given a traced symmetric monoidal category $\mathbb{C}$ we define a new category $\mathcal{G}(\mathbb{C})$, as follows:

- Objects: Pairs of objects from $\mathbb{C}$, e.g. (A^+, A^-) where A^+ and A^- are objects of $\mathbb{C}$.
- Arrows: An arrow $f : (A^+, A^-) \longrightarrow (B^+, B^-)$ in $\mathcal{G}(\mathbb{C})$ is $f : A^+ \otimes B^- \longrightarrow A^- \otimes B^+$ in $\mathbb{C}$.
- Identity: $1_{(A^+,A^-)} = \sigma_{A^+,A^-}$.
- Composition: Composition is given by symmetric feedback. Given $f : (A^+, A^-) \longrightarrow (B^+, B^-)$ and $g : (B^+, B^-) \longrightarrow (C^+, C^-)$, $gf : (A^+, A^-) \longrightarrow (C^+, C^-)$ is given by:

$$gf = Tr^{B^- \otimes B^+}_{A^+ \otimes C^-, A^- \otimes C^+}(\beta(f \otimes g)\alpha)$$

where

$$\alpha = (1_{A^+} \otimes 1_{B^-} \otimes \sigma_{C^-,B^+})(1_{A^+} \otimes \sigma_{C^-,B^-} \otimes 1_{B^+})$$

$$\beta = (1_{A^-} \otimes 1_{C^+} \otimes \sigma_{B^+,B^-})(1_{A^-} \otimes \sigma_{B^+,C^+} \otimes 1_{B^-})(1_{A^-} \otimes 1_{B^+} \otimes \sigma_{B^-,C^+}).$$

This can be represented graphically as in Figure 5. An informal picture displaying gf is also given below. This latter picture (Figure 6) represents the *symmetric feedback*.

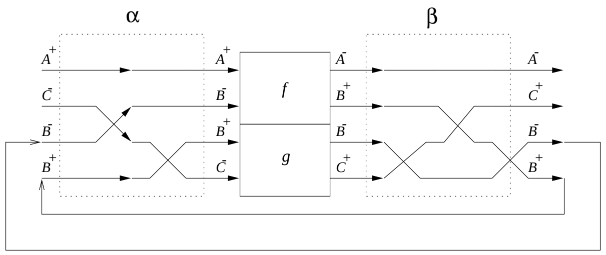


Figure 5: Composition in $\mathcal{G}(\mathbb{C})$

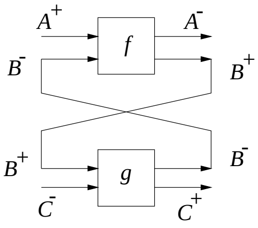


Figure 6: Symmetric Feedback

- Tensor: $(A^+, A^-) \otimes (B^+, B^-) = (A^+ \otimes B^+, A^- \otimes B^-)$ and for $f : (A^+, A^-) \longrightarrow (B^+, B^-)$ and $g : (C^+, C^-) \longrightarrow (D^+, D^-)$,

$$f \otimes g = (1_{A^-} \otimes \sigma_{B^+,C^-} \otimes 1_{D^+})(f \otimes g)(1_{A^+} \otimes \sigma_{C^+,B^-} \otimes 1_{D^-}).$$

- Unit: (I, I).

Remark 2.3.2 We have given a definition for α and β above, however, any other permutations $A^+ \otimes C^- \otimes B^- \otimes B^+ \xrightarrow{\cong} A^+ \otimes B^- \otimes B^+ \otimes C^-$ and $A^- \otimes B^+ \otimes B^- \otimes C^+ \xrightarrow{\cong} A^- \otimes C^+ \otimes B^- \otimes B^+$ for α and β respectively will yield the same result for gf due to coherence.

Proposition 2.3.3 *Let $\mathbb{C}$ be a traced symmetric monoidal category. Then $\mathcal{G}(\mathbb{C})$ defined as in Definition 2.3.1 is a compact closed category. Moreover, $N : \mathbb{C} \longrightarrow \mathcal{G}(\mathbb{C})$ with $N(A) = (A, I)$ and $N(f) = f$ is a full and faithful embedding.*

Proof. For any two objects (A^+, A^-) and (B^+, B^-) in $\mathcal{G}(\mathbb{C})$, $\sigma_{(A^+,A^-),(B^+,B^-)} =_{def} (1_{A^-} \otimes \sigma_{B^+,B^-} \otimes 1_{A^+})(\sigma_{B^+,A^-} \otimes \sigma_{A^+,B^-})(1_{B^+} \otimes \sigma_{A^+,A^-} \otimes 1_{B^-})(\sigma_{A^+,B^+} \otimes \sigma_{B^-,A^-})$. This makes $\mathcal{G}(\mathbb{C})$ into a symmetric monoidal category.

The dual of (A^+, A^-) is given as $(A^+, A^-)^* = (A^-, A^+)$ where the unit

$$\eta_{(A^+,A^-)} : (I, I) \longrightarrow (A^+, A^-) \otimes (A^+, A^-)^* =_{def} \sigma_{A^-,A^+}$$

and counit

$$\epsilon_{(A^+,A^-)} : (A^+, A^-)^* \otimes (A^+, A^-) \longrightarrow (I, I) =_{def} \sigma_{A^-,A^+}.$$

It is easily verified that the unit and counit are natural transformations and the triangles of Definition 2.1.3 commute. Hence $\mathcal{G}(\mathbb{C})$ is a compact closed category.

The internal homs are given as

$$(A^+, A^-) \multimap (B^+, B^-) = (B^+, B^-) \otimes (A^+, A^-)^* = (B^+ \otimes A^-, B^- \otimes A^+).$$

$\square$

Proposition 2.3.4 *Let $\mathbb{C}$ be a traced symmetric monoidal category, then $\mathcal{G}(\mathbb{C}) \cong Int\,\mathbb{C}$*

Proof. Define the functor $F : \mathcal{G}(\mathbb{C}) \longrightarrow Int\,\mathbb{C}$ as follows: $F(A^+, A^-) = (A^+, A^-)$ and $F(f) = \sigma_{A^-,B^+}f$ for $f : (A^+, A^-) \longrightarrow (B^+, B^-)$. $F(1_{(A^+,A^-)}) = F(\sigma_{A^+,A^-}) = \sigma_{A^-,A^+}\sigma_{A^+,A^-} = 1_{A^+\otimes A^-} = 1_{F(A^+,A^-)}$. The proof of $F(gf) = F(g)F(f)$ is given in Figure 7: both (a) and (b) reduce to (c). F is a symmetric monoidal functor with $\varphi = 1_{A^+\otimes B^+} \otimes \sigma_{A^-,B^-}$ and $\varphi_I = 1_I$. Moreover, F has an inverse G which is the identity on objects and $G(f) = \sigma_{B^+,A^-}f$ for $f : (A^+, A^-) \longrightarrow (B^+, B^-)$ in $Int\,\mathbb{C}$.

$\square$

Notation: We write $A \lhd B(f, g)$ for A, B objects of a category $\mathbb{C}$ and $f : A \longrightarrow B$ and $g : B \longrightarrow A$ morphisms in $\mathbb{C}$ to mean that A is a retract of B i.e., $gf = 1_A$.

Lemma 2.3.5

- *Let $A^+ \cong B^+$ and $A^- \cong B^-$ in $\mathbb{C}$, then $(A^+, A^-) \cong (B^+, B^-)$ in $\mathcal{G}(\mathbb{C})$.*
- *If $A^+ \lhd B^+(f_1, g_1)$ and $A^- \lhd B^-(f_2, g_2)$ in $\mathbb{C}$, then $(A^+, A^-) \lhd (B^+, B^-)(\sigma_{B^+,A^-}(f_1 \otimes g_2), \sigma_{A^+,B^-}(g_1 \otimes f_2))$ in $\mathcal{G}(\mathbb{C})$.*

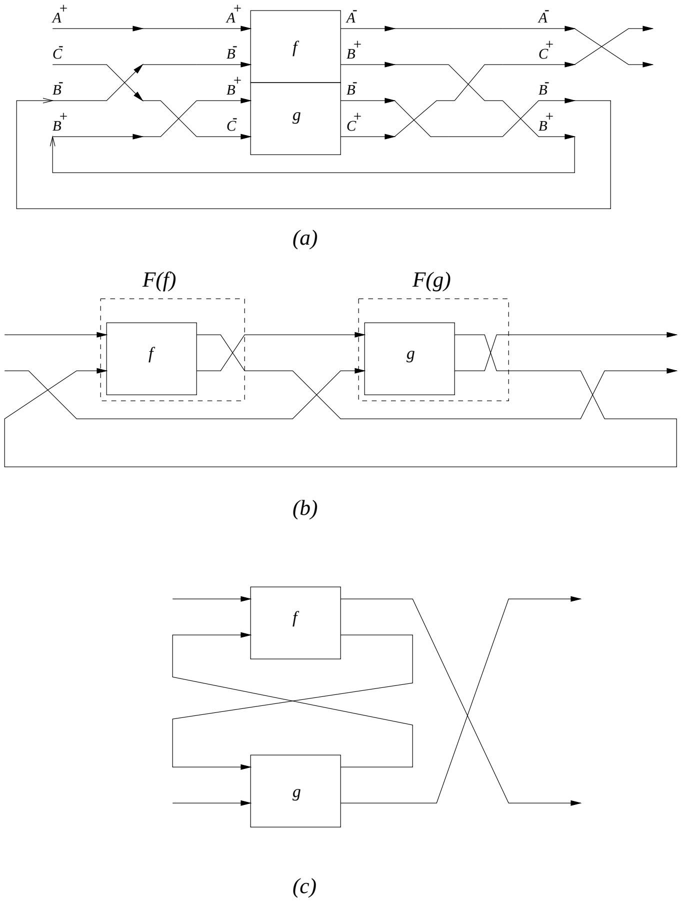


Figure 7: $F(gf) = F(g)F(f)$

Proof. First we observe that if $f : (A^+, A^-) \longrightarrow (B^+, B^-) = \sigma_{B^+,A^-}(f_1 \otimes f_2)$ and $g : (B^+, B^-) \longrightarrow (C^+, C^-) = \sigma_{C^+,B^-}(g_1 \otimes g_2)$, with $f_1 : A^+ \longrightarrow B^+, f_2 : B^- \longrightarrow A^-, g_1 : B^+ \longrightarrow C^+$ and $g_2 : C^- \longrightarrow B^-$, then composition is given by

$$gf = \sigma_{C^+,A^-}(g_1 f_1 \otimes f_2 g_2).$$

Suppose that $f_1 : A^+ \xrightarrow{\cong} B^+$ and $g_2 : A^- \xrightarrow{\cong} B^-$. Define $f : (A^+, A^-) \longrightarrow (B^+, B^-)$ by $f = \sigma_{B^+,A^-}(f_1 \otimes g_2^{-1})$ and $g : (B^+, B^-) \longrightarrow (A^+, A^-)$ by $g = \sigma_{A^+,B^-}(f_1^{-1} \otimes g_2)$. Then, $gf = \sigma_{A^+,A^-}(1_{A^+} \otimes 1_{A^-}) = \sigma_{A^+,A^-} = 1_{(A^+,A^-)}$. Similarly $fg = 1_{(B^+,B^-)}$ and hence $(A^+, A^-) \cong (B^+, B^-)$.

Now suppose $A^+ \lhd B^+(f_1, g_1)$ and $A^- \lhd B^-(f_2, g_2)$, define $f : (A^+, A^-) \longrightarrow (B^+, B^-)$ by $f = \sigma_{B^+,A^-}(f_1 \otimes g_2)$ and define $g : (B^+, B^-) \longrightarrow (A^+, A^-)$ by $g = \sigma_{A^+,B^-}(g_1 \otimes f_2)$. Then

$$\begin{aligned} gf &= \sigma_{A^+,A^-}(g_1 f_1 \otimes g_2 f_2) \\ &= \sigma_{A^+,A^-}(1_{A^+} \otimes 1_{A^-}) \\ &= \sigma_{A^+,A^-} \\ &= 1_{(A^+,A^-)} \end{aligned}$$

Hence $(A^+, A^-) \lhd (B^+, B^-)(f, g)$. □

2.4 Linear Logic and its Categorical Models

Linear logic invented by Girard [44], is a refinement of classical and intuitionistic logics. Linear logic is a constructive and resource sensitive logic. To see the resource sensitivity of linear logic we need to discuss Gentzen's analysis of Hilbert's proof theory [50, 103]. Gentzen introduced a fundamental reformulation of the syntax, called *sequent calculus*. A sequent is denoted by $\Gamma \vdash \Delta$, where Δ and Γ are finite sequences (possibly empty) of formulas, for example,

$$A_1, A_2, \cdots, A_m \vdash B_1, B_2, \cdots, B_n$$

Gentzen also introduced formal rules for generating sequents, the so-called *derivable* sequents. Gentzen broke down the manipulation of logic into two classes of rules applied to sequents: *structural rules* and *logical rules.* Two of the structural rules that are important for our discussion here are the contraction and weakening rules:

$$
\begin{array}{lcc}
(\textit{contraction}) & \dfrac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta} & \dfrac{\Gamma \vdash \Delta, B, B}{\Gamma \vdash \Delta, B} \\
(\textit{weakening}) & \dfrac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta} & \dfrac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, B}
\end{array}
$$

Let's consider *intuitionistic* sequents, i.e. those of the form $A_1, A_2, \cdots, A_m \vdash B$ with one conclusion. We can give a functional interpretation: view the sequent as a function $t(x_1, x_2, \cdots, x_m) : B$ with arguments $x_i \in A_i$ for all $i = 1, \cdots, m$. Then, (*contraction*) can be interpreted as duplicating the arguments of a function, i.e. setting two input variables equal and (*weakening*) is interpreted as introducing *dummy* variables. Yet another way of looking at these rules (reading from conclusion upward to hypothesis) is to view (*contraction*) as duplication and (*weakening*) as erasing.

In linear logic one is not allowed to duplicate or erase a formula (resource). Thus, these rules are removed and as a consequence we get two conjunctions: *tensor* ($\otimes$) and *with* ($\&$) and two disjunctions: *par* ($⅋$) and *plus* ($\oplus$). These are grouped into *multiplicatives* ($\otimes$, $⅋$, $\multimap$) and *additives* ($\oplus$, $\&$). There is also an involutive negation $(-)^{\perp}$ which relates these groups by De Morgan duality. The linear implication above ($\multimap$) is a defined multiplicative connective: $A \multimap B =_{def} A^{\perp} ⅋ B$. Linear implication represents real life (causal) implication. That is, from A and $A \multimap B$ we get B (modus ponens) and A is no longer there; compare this to implication in mathematics where from A and $A \Rightarrow B$ we get B and A *is still true.* To regain the power of intuitionistic logic, Girard introduced a new connective, the *exponential* connective, !. This indicates that weakening and contraction can be applied to $!A$. In this way we get back these structural rules in a controlled fashion. From a resource point of view, a hypothesis $!A$ can be reused arbitrarily. This gives a decomposition of "$\Rightarrow$" into more basic notions:

$$A \Rightarrow B = (!A) \multimap B.$$

Finally, classical and intuitionistic logics can be faithfully translated into linear logic [44, 105]. For more on linear logic see the excellent review [49].

2.4.1 The Multiplicative Fragment of Linear Logic

In this thesis we will be interested in the multiplicative fragment of linear logic (MLL) and the multiplicative and exponential fragment of intuitionistic linear logic (MEILL).

Formulas in MLL are built from *propositional atoms*, $\alpha, \beta, \ldots$, their linear negations, $\alpha^\perp, \beta^\perp, \cdots$ and the constants (or *units*) $\mathbf{1}$ and $\perp$ by the connectives $\otimes$ and $⅋$. Finite sequences of formulas are denoted by capital Greek letters $\Delta, \Gamma, \ldots$. A *literal* is either an atom or its linear negation. Notice that one can remove the (*exchange*) rule (see below) and instead work with *multisets* of formulas.

Below we give the sequent calculus presentation for MLL. We will use one sided sequents, that is sequents are of the form $\vdash \Delta$, general sequents $\Gamma \vdash \Delta$ can be mimicked as $\vdash \Gamma^\perp, \Delta$. This is a fundamental feature of linear negation: it transports the formulas from one side of the sequent to the other side.

Identity/Negation

$$\frac{}{\vdash \alpha, \alpha^\perp}\,(identity) \qquad\qquad \frac{\vdash \Gamma, A \quad \vdash A^\perp, \Delta}{\vdash \Gamma, \Delta}\,(cut)$$

Structure

$$\frac{\vdash \Gamma}{\vdash \tau(\Gamma)}\,(exchange : \tau \text{ is a permutation of } \Gamma)$$

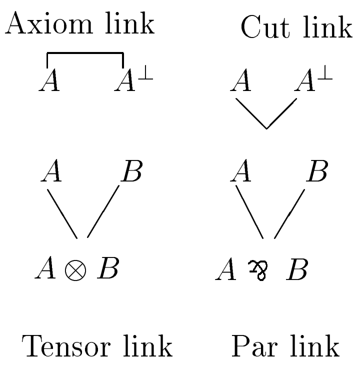
Logic

$$\frac{}{\vdash \mathbf{1}}\,(\mathbf{1}) \qquad\qquad \frac{\vdash \Gamma}{\vdash \Gamma, \perp}\,(\perp)$$

$$\frac{\vdash \Gamma, A \quad \vdash B, \Delta}{\vdash \Gamma, A \otimes B, \Delta}\,(\otimes) \qquad\qquad \frac{\vdash \Gamma, A, B}{\vdash \Gamma, A ⅋ B}\,(⅋)$$

There is a cut elimination procedure which is Church-Rosser and strongly normalising for MLL without units. In particular, any proof of a sequent in MLL without units can be reduced to a unique cut-free proof of the same sequent [49].

Girard introduced proof nets [44] as a graphical representation of proofs. Proof nets are the natural deduction system for linear logic. They provide a parallel syntax in that they have multiple conclusions and they suppress the commuting rules in sequent calculus. That is, the sequent calculus proofs that are equivalent by commuting rules are translated into a single proof net. Proof nets are defined using proof structures. A *proof structure* is simply a graph with formula occurrences as vertices. The edges are built via links of the form



However, not every proof structure is a proof net. The correctness criterion to pick the correct ones was given by Girard in [44] using the notion of a "trip" on a proof structure. Girard introduced the notion of dataflow (a particle, information, etc.) through a proof structure. This view will be explored more fully in this thesis. Girard's correctness criterion was simplified by Danos and Regnier using *switchings* [34].

A *DR-switching* for a cut-free proof structure is the assignment of left or right to each par link. Given a switching S for a cut-free proof structure we define the associated graph to be the deletion of the par-edges *not* selected by the switching S.

Theorem 2.4.1 (Danos and Regnier) *A cut-free proof structure is a proof net iff for every DR-switching, the associated DR-graph is acyclic and connected.*

We also recall the following important results due to Girard [44].

Theorem 2.4.2 (Girard) *If π is a proof of $\vdash \Gamma$ in the sequent calculus for MLL then we can naturally associate with it a proof net $N(\pi)$ whose multiset of terminal formulas is precisely Γ.*

Theorem 2.4.3 (Girard) *If β is a proof net, then there exists a proof π in the sequent calculus of MLL such that $N(\pi) = \beta$.*

2.4.2 The Multiplicative and Exponential Fragment of ILL

Formulas in MEILL are built from *propositional atoms*, $\alpha, \beta, \ldots$, and the constant (or *unit*) $\mathbf{1}$ by the connectives $\otimes$, $\multimap$ and $!$. Finite sequences of formulas are denoted by capital Greek letters $\Delta, \Gamma, \ldots$. Notice that one can remove the (*exchange*) rule (see below) and instead use *multisets* of formulas. Below we give a sequent calculus presentation for MEILL. Note that in the intuitionistic linear logic, linear negation is not a primitive connective and we deal with sequents of the form $\Gamma \vdash A$ for A a single formula.

Identity

$$\frac{}{\alpha \vdash \alpha}\ (identity) \qquad\qquad \frac{\Gamma \vdash A \quad A, \Delta \vdash B}{\Gamma, \Delta \vdash B}\ (cut)$$

Structure

$$\frac{\Gamma \vdash B}{\tau(\Gamma) \vdash B}\ (exchange : \tau \text{ is a permutation of } \Gamma)$$

Logic

$$\frac{\Gamma \vdash A}{\Gamma, \mathbf{1} \vdash A}\ (\mathcal{L}\mathbf{1}) \qquad \frac{}{\vdash \mathbf{1}}\ (\mathcal{R}\mathbf{1})$$

$$\frac{\Gamma, A, B \vdash C}{\Gamma, A \otimes B \vdash C}\ (\mathcal{L}\otimes) \qquad \frac{\Gamma \vdash A \quad \Delta \vdash B}{\Gamma, \Delta \vdash A \otimes B}\ (\mathcal{R}\otimes)$$

$$\frac{\Gamma \vdash A \quad \Delta, B \vdash C}{\Gamma, \Delta, A \multimap B \vdash C}\ (\mathcal{L} \multimap) \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash A \multimap B}\ (\mathcal{R} \multimap)$$

$$\frac{!\Gamma \vdash A}{!\Gamma \vdash !A}\ (ofcourse) \qquad \frac{\Gamma \vdash B}{\Gamma, !A \vdash B}\ (weakening)$$

$$\frac{\Gamma, A \vdash B}{\Gamma, !A \vdash B}\ (dereliction) \qquad \frac{\Gamma, !A, !A \vdash B}{\Gamma, !A \vdash B}\ (contraction)$$

2.4.3 Categorical Models

In this section we will recall and discuss the categorical models of MLL and MEILL. First let's consider MLL. Seely [101] has shown that *-autonomous categories are models of MLL. Tensor and par are interpreted by tensor and par (dual of tensor) in a *-autonomous category $\mathbb{C}$. Linear negation corresponds to the involution on $\mathbb{C}$ and a multiset Γ is interpreted as a par product $⅋\ \Gamma$. A sequent $\vdash \Gamma$ is interpreted as a morphism $I \longrightarrow ⅋\ \Gamma$. Now let's look at this in more detail.

Suppose the propositional atoms in the logic are interpreted by the objects in $\mathbb{C}$. The *interpretation* function $[\![-]\!]$ is defined inductively as:

$$[\![\mathbf{1}]\!] = I; \quad [\![\bot]\!] = \bot; \quad [\![\alpha^\perp]\!] = [\![\alpha]\!]^\perp;$$

$$[\![A \otimes B]\!] = [\![A]\!] \otimes [\![B]\!];$$

$$[\![A, B]\!] = [\![A ⅋ B]\!] = [\![A]\!] ⅋ [\![B]\!];$$

$$[\![\vdash A_1, \cdots, A_n]\!] = \text{ a morphism } I \longrightarrow [\![A_1]\!] ⅋ \cdots ⅋ [\![A_n]\!].$$

We now sketch how the axioms and inference rules are interpreted in a *-autonomous category $\mathbb{C}$.

The identity sequents $\vdash \alpha, \alpha^\perp$ are interpreted as morphisms $I \longrightarrow X \parr X^\perp$ where X is the interpretation of α. These morphisms are obtained by transposition of the identity on X, i.e. using the bijection $\mathbb{C}(I \otimes X, X) \cong \mathbb{C}(I, X \multimap X)$.

The $(\mathbf{1})$ rule is expressed by 1_I and the $(\perp)$ rule is expressed via composition with the inverse of the right unit isomorphism of $\parr$.

The (cut) rule is expressed by composition. $[\![\vdash \Gamma, A]\!]$ = a morphism $I \longrightarrow (\parr \Gamma) \parr A$ which is equivalent to a morphism $(\parr \Gamma)^\perp \longrightarrow A$ and a morphism $I \longrightarrow A^\perp \parr (\parr \Delta)$ is equivalent to a morphism $A \longrightarrow (\parr \Delta)$. Composing these two morphisms gives a morphism $(\parr \Gamma)^\perp \longrightarrow (\parr \Delta)$, equivalent to $I \longrightarrow (\parr \Gamma) \parr (\parr \Delta) = \parr (\Gamma, \Delta)$.

The $(exchange)$ rule is expressed through the symmetry and associativity of the par and tensor products in $\mathbb{C}$.

The $(\otimes)$ rule is essentially taking the tensor product of the morphisms $(\parr \Gamma)^\perp \longrightarrow A$ and $(\parr \Delta)^\perp \longrightarrow B$. Finally, the $(\parr)$ rule does nothing.

Proposition 2.4.4 (Seely, [101]) *The above interpretation yields a sound model of MLL in any *-autonomous category $\mathbb{C}$. That is, the induced interpretation of proofs is preserved by cut-elimination: if Π reduces to Π' (by cut-elimination) then $[\![\Pi]\!] = [\![\Pi']\!]$ as morphisms in $\mathbb{C}$.*

We will next recall the categorical model for MEILL. This was first introduced by Seely [101]. However it was later modified by [18] and [19]. We closely follow [30].

A categorical model of MEILL, which we call a *linear category*, consists of the following data:

1. A symmetric monoidal closed category $(\mathbb{C}, I, \otimes, \multimap)$,

2. A symmetric (not necessarily strong) monoidal functor $(!, \varphi, \varphi_I) : \mathbb{C} \longrightarrow \mathbb{C}$,

3. Monoidal natural transformations:

   (i) $der : ! \Longrightarrow id$, where id is the identity functor,

(ii) $\delta : ! \Longrightarrow !!$,

(iii) $weak : ! \Longrightarrow \mathcal{K}_I$ where $\mathcal{K}_I$ is the constant I functor and

(iv) $con : ! \Longrightarrow ! \otimes !$

such that

- $(!, der, \delta)$ is a comonad.
- for each object A, the maps $weak_A$ and con_A are maps of coalgebras.
- for each object A, the triple $(!A, weak_A, con_A)$ is a commutative comonoid.
- for each object A, the map δ_A is a map of commutative comonoids.

Suppose that the propositional atoms are interpreted by objects of $\mathbb{C}$, the interpretation function $[\![-]\!]$ is defined inductively as follows:

$$[\![\mathbf{1}]\!] = I$$

$$[\![A \multimap B]\!] = [\![A]\!] \multimap [\![B]\!]$$

$$[\![A, B]\!] = [\![A \otimes B]\!] = [\![A]\!] \otimes [\![B]\!]$$

$$[\![!A]\!] = !([\![A]\!])$$

A sequence Γ of formulas is interpreted as $\otimes\Gamma$ and a sequent $\Gamma \vdash A$ is interpreted as a morphism $\otimes\Gamma \longrightarrow A$ (We use A, B instead of $[\![A]\!]$ and $[\![B]\!]$.) Similarly to MLL, we interpret axioms and inference rules as generating arrows, as follows:

The identity sequents $\alpha \vdash \alpha$ are interpreted by identity morphisms $X \longrightarrow X$ where X is the interpretation of α.

The (*cut*) rule is expressed by composition: given the morphisms $f : \otimes\Gamma \longrightarrow A$ and $g : A \otimes (\otimes\Delta) \longrightarrow B$ we get a morphism $\otimes\Gamma \otimes (\otimes\Delta) \xrightarrow{f \otimes 1} A \otimes (\otimes\Delta) \xrightarrow{g} B$.

The (*exchange*) rule is expressed using the symmetry and associativity morphisms of the tensor product.

The ($\mathcal{R}\mathbf{1}$) rule is just 1_I and the ($\mathcal{L}\mathbf{1}$) rule is expressed by composition with the right unit isomorphism.

The ($\mathcal{L}\otimes$) rule does nothing. The ($\mathcal{R}\otimes$) rule is interpreted by taking the tensor product of the morphisms $\otimes\Gamma \longrightarrow A$ and $\otimes\Delta \longrightarrow B$.

The ($\mathcal{L} \multimap$) rule is expressed using the morphisms $f : \otimes\Gamma \longrightarrow A$ and $g : (\otimes\Delta) \otimes B \longrightarrow C$ as follows:

$(\otimes\Gamma) \otimes (\otimes\Delta) \otimes (A \multimap B) \xrightarrow{f\otimes 1} A \otimes (\otimes\Delta) \otimes (A \multimap B) \xrightarrow{\sigma\otimes 1} (\otimes\Delta) \otimes A \otimes (A \multimap B) \xrightarrow{1\otimes\sigma} (\otimes\Delta) \otimes (A \multimap B) \otimes A \xrightarrow{1\otimes ev} (\otimes\Delta) \otimes B \xrightarrow{g} C$

The ($\mathcal{R} \multimap$) rule is just taking the transpose of the morphism $(\otimes\Gamma) \otimes A \longrightarrow B$.

The (*of course*) rule is expressed by using functoriality of $!$ and δ (comultiplication natural transformation.) We demonstrate this for $\Gamma = B, C$ the general case follows trivially. Given $f : !B\otimes !C \longrightarrow A$ we get a morphism as follows:

$$!B\otimes !C \xrightarrow{\delta_B\otimes\delta_C} !!B\otimes !!C \xrightarrow{\varphi} !(!B\otimes !C) \xrightarrow{!f} !A$$

The (*weakening*) rule is expressed by composition with $weak_A$. Given $f : \otimes\Gamma \longrightarrow B$ we get a morphism

$$(\otimes\Gamma)\otimes !A \xrightarrow{1\otimes weak_A} (\otimes\Gamma) \otimes I \xrightarrow{\cong} \otimes\Gamma \xrightarrow{f} B$$

The (*dereliction*) rule is expressed by composition with der_A. Given $f : (\otimes\Gamma) \otimes A \longrightarrow B$ we get a morphism

$$(\otimes\Gamma)\otimes !A \xrightarrow{1\otimes der_A} (\otimes\Gamma) \otimes A \xrightarrow{f} B$$

The (*contraction*) rule is expressed by composition with con_A. Given $f : (\otimes\Gamma)\otimes !A\otimes !A \longrightarrow B$, we get a morphism

$$(\otimes\Gamma)\otimes !A \xrightarrow{1\otimes con_A} (\otimes\Gamma)\otimes !A\otimes !A \xrightarrow{f} B$$

Proposition 2.4.5 (Seely [101], Bierman [20]) *The above interpretation yields a sound model of MEILL in any linear category. That is, the induced interpretation of proofs is preserved by cut-elimination.*

2.4.4 The MIX Rule

An interesting variant of MLL is obtained by adding the MIX rule,

$$\frac{\vdash \Gamma \quad \vdash \Delta}{\vdash \Gamma, \Delta}\ (MIX).$$

The MIX rule and its categorical interpretation are discussed in [32]. Our presentation here follows [104].

The MIX rule can be specified in several equivalent forms.

Proposition 2.4.6 (Cockett & Seely [32]) *The equivalent forms of the MIX rule are* $\perp \vdash \mathbf{1}$ *and* $A \otimes B \vdash A \parr B$*.*

Fleury and Retoré [38] provided a correctness criterion for proof nets in MLL+MIX. Simply, the acyclicity condition remains and the connectedness condition is removed. This fits in nicely with the intuitive idea that an MLL proof net associated with $\vdash \Gamma$ and an MLL proof net associated with $\vdash \Delta$ can be juxtaposed to yield an MLL+MIX proof net associated with $\vdash \Gamma, \Delta$.

A categorical model of MLL supports the MIX rule, if there exists a *unary MIX morphism* $m : \perp \longrightarrow \mathbf{1}$ such that the following diagram commutes.

$$\begin{tikzcd}
 & \perp \otimes \perp \arrow[dl, "m \otimes 1_\perp"'] \arrow[dr, "1_\perp \otimes m"] & \\
\mathbf{1} \otimes \perp \cong \perp \arrow[rr, "1_\perp"] & & \perp \cong \perp \otimes \mathbf{1}
\end{tikzcd}$$

Given morphisms $f : X \longrightarrow \perp$ and $g : Y \longrightarrow \perp$, we can construct $X \otimes Y \xrightarrow{f \otimes g} \perp \otimes \perp \xrightarrow{\tilde{m}} \perp$, where $\tilde{m} : \perp \otimes \perp \longrightarrow \perp$ is the morphism in the diagram above. Furthermore, m induces the *binary MIX morphisms* $z_{AB} : A \otimes B \longrightarrow A \parr B$, natural in A and B as follows:

$$\cfrac{\cfrac{\cfrac{\vdash A^\perp, A \quad \vdash B^\perp, B}{\vdash A^\perp, B^\perp, A, B}\ (MIX)}{\vdash A^\perp \parr B^\perp, A \parr B}\ (\parr), \text{twice}}{A \otimes B \vdash A \parr B}$$

Chapter 3

Partially Additive Categories

In this chapter we recall the definitions of partially additive monoids and categories enriched over such monoids, partially additive categories. Partially additive categories were defined and used by Manes and Arbib to provide an algebraic semantics for programming languages [90]. Our interest in partially additive categories is primarily due to the fact that they provide a canonical method for fixed-point construction, which in turn yields canonical constructions for trace and composition in geometry of interaction categories.

Definition 3.0.7 A *partially additive monoid* is a pair (M, Σ), where M is a nonempty set and Σ is a partial function which maps countable families in M to elements of M (we say that $\{x_i\}_{i\in I}$ is *summable* if $\sum_{i\in I} x_i$ is defined)[1] subject to the following axioms:

1. *Partition-Associativity Axiom.* If $\{x_i\}_{i\in I}$ is a countable family and if $\{I_j\}_{j\in J}$ is a (countable) partition of I , then $\{x_i\}_{i\in I}$ is summable if and only if $\{x_i\}_{i\in I_j}$ is summable for every $j \in J$ and $\sum_{i\in I_j} x_i$ is summable for $j \in J$. In that case,

$$\textstyle\sum_{i\in I} x_i = \sum_{j\in J}(\sum_{i\in I_j} x_i)$$

2. *Unary Sum Axiom.* Any family $\{x_i\}_{i\in I}$ in which I is a singleton is summable and $\sum_{i\in I} x_i = x_j$ if $I = \{j\}$.

[1]Throughout, "countable" means finite or denumerable. All index sets are countable.

3. *Limit Axiom.* If $\{x_i\}_{i \in I}$ is a countable family and if $\{x_i\}_{i \in F}$ is summable for every finite subset F of I then $\{x_i\}_{i \in I}$ is summable.

We observe the following facts about partially additive monoids:

(i) Axiom 1 implies that every subfamily of a summable family is summable. Suppose $\{x_i\}_{i \in I}$ is summable and let $K \subseteq I$. Consider the partition $I = I_1 \cup I_2$ where $I_1 = K$ and $I_2 = I - K$, then by the partition-associativity axiom, $\sum_{i \in K} x_i$ is summable.

(ii) Axioms 1 and 2 imply that the empty family is summable. We denote $\sum_{i \in \emptyset} x_i$ by 0, which is an additive identity for summation. In fact, 0 is a countable additive identity. To see this, first note that $I_j = \emptyset$ is allowed for any number of j (finite or denumerably infinite). Now suppose $\{x_i\}_{i \in I}$ is a summable family and J is any countable set disjoint from I such that $x_i = 0$ for all $i \in J$. For $j \in I \cup J$ define

$$I_j = \begin{cases} \{j\} & \text{if } j \in I, \\ \emptyset & \text{if } j \in J. \end{cases}$$

Clearly $\{I_j\}_{j \in I \cup J}$ partitions I and by the partition-associativity axiom $\sum_{j \in I \cup J} (\sum_{i \in I_j} x_i)$ exists and equals $\sum_{i \in I} x_i$ but $\sum_{i \in I_j} x_i = x_j$ if $j \in I$ and is 0 if $j \in J$ and hence the result.

(iii) Axiom 1 implies the obvious equations of commutativity and associativity for the sum (when defined). More generally, $\sum_i x_{\varphi(i)}$ is defined for any permutation φ of I whenever $\sum_i x_i$ exists and $\sum_i x_{\varphi(i)} = \sum_i x_i$; just consider the partition $\{\varphi(j)\}_{j \in I}$.

(iv) In case Σ is totally defined, (M, Σ_2) forms an Abelian monoid, where Σ_2 is the restriction of Σ to binary families, i.e. $\Sigma_2 : M \times M \longrightarrow M$.

(v) There are no additive inverses. Indeed, let $\{x_i\}_{i \in I}$ be a summable family with $\sum_{i \in I} x_i = 0$ and set $y = \sum_{j \in I - \{i\}} x_j$ for some $i \in I$. Then, $y + x_i = 0$ and $x_i = x_i + (y + x_i) + (y + x_i) + \cdots = (x_i + y) + (x_i + y) + \cdots = 0$. Thus, $x_i = 0$

for all $i \in I$. This property has been called the *positivity property* by Benson and Manes [91].

We will also need to consider partially additive monoids without requiring the limit axiom. We call such structures Σ-*monoids*. Such structures have been considered by Benson and Manes [91], where they are called positive monoids. Kuroš has also studied such structures in [75]. Clearly, a partially additive monoid is a Σ-monoid. All the properties mentioned above for partially additive monoids are also true for Σ-monoids. We will further study such structures in the next chapter.

A doubly indexed family f in a partially additive monoid M, $f : I \times J \longrightarrow M$, is denoted as $\{f_{ij}\}_{i \in I, j \in J}$ or simply $\{f_{ij}\}$ if the index sets are clear from the context. Such a family is summable iff $\sum_{i \in I}(\sum_{j \in J} f_{ij})$ exists and in that case $\sum_{i,j} f_{ij} = \sum_{i \in I}(\sum_{j \in J} f_{ij})$. It follows, using Axiom 1, that for a summable family $\{f_{ij}\}$, $\sum_{i \in I}(\sum_{j \in J} f_{ij}) = \sum_{j \in J}(\sum_{i \in I} f_{ij})$.

Here are some examples of partially additive monoids.

Example 3.0.8

1. $M =$**PInj**(X, Y), the set of partial injective functions from X to Y. A family $\{f_i\}_{i \in I} \in$**PInj**(X, Y) is said to be summable iff $Dom(f_i) \cap Dom(f_j) = \emptyset$ and $Codom(f_i) \cap Codom(f_j) = \emptyset$ for all $i \neq j$. In that case, $(\sum_i f_i)(x) = f_j(x)$ if $x \in Dom(f_j)$ for some $j \in I$ and undefined, otherwise.

2. $M =$**Pfn**(X, Y), the set of partial functions from X to Y. A family $\{f_i\}_{i \in I} \in M$ is summable iff $Dom(f_i) \cap Dom(f_j) = \emptyset$ for all $i \neq j$. In that case, $(\sum_i f_i)(x) = f_j(x)$ if $x \in Dom(f_j)$ for some $j \in I$ and undefined, otherwise. We denote this partially additive monoid by $(\mathbf{Pfn}(X, Y), \Sigma^{di})$.

3. Let $M =$**Pfn**(X, Y) and define a family $\{f_i\}_{i \in I}$ to be summable when $f_i(x) = f_j(x)$ for all $x \in Dom(f_i) \cap Dom(f_j)$ and all $i, j \in I$. We call such families *coherently overlapping* partial functions. This additive structure is called *the overlap sum* and will be denoted by Σ^{ov}. For a summable family $\{f_i\}$,

$(\sum^{ov} f_i)(x) = f_j(x)$ if there exists a $j \in I$ such that $x \in Dom(f_j)$ and undefined, otherwise. Notice that in sharp contrast to the previous example the overlap sum is countably idempotent, i.e. for a family $\{f_i\}_{i\in I}$ with $f_i = f$ for all $i \in I$, $\sum_{i\in I} f_i = f$. However, clearly such a family is not even summable in the additive structure given in the previous example.

4. $M =$**Rel**$_+(X, Y)$, the set of relations from a set X to a set Y. Any family $\{R_i\}_{i\in I} \in$**Rel**$_+(X, Y)$ is summable with $\sum_i R_i = \bigcup_{i\in I} R_i$.

5. A countably complete poset with Σ as supremum is a partially additive monoid in which all families are summable and the sum is countably idempotent, that is for $x_i = x$ for all $i \in I$, $\sum_{i\in I} x_i = x$. A related *nonexample* is ω-complete posets with summable families defined as countable chains and Σ as supremum. For example, let $x, y \leq z$ with x, y incomparable, then $x + (y + z)$ is defined but $x + y + z$ is not.

6. Let M be an ω-complete poset satisfying *countable bounded completeness*, that is given a countable subset D of M, if D has an upper bound then it has a least upper bound. Let the summable families be the bounded families and Σ be the supremum. Then, (M, Σ) is a partially additive monoid. The additive identity is of course $sup(\emptyset)$.

7. Let $M = [0, 1]$ and given $\{x_i\}_{i\in I}$, define $\sum_i x_i = min(S, 1)$ for I finite, where S is the usual sum in real numbers, and for I infinite let $\sum_i x_i$ be the limit of the net of finite partial sums. (M, Σ) forms a partially additive monoid.

Definition 3.0.9 The category of partially additive monoids, **PAMon**, has as objects partially additive monoids (M, Σ). Its arrows $(M, \Sigma) \xrightarrow{f} (M', \Sigma')$ are maps from M to M' which preserve the sum, in the sense that for all summable families $\{x_i\}_{i\in I}$ in M, $\{f(x_i)\}_{i\in I}$ is summable in M' and $f(\sum_{i\in I} x_i) = \sum'_{i\in I} f(x_i)$. Composition and identities are inherited from **Set**.

Note that Σ-monoids and additive maps also form a category which we will denote by Σ**Mon**.

Observe that the categories **PAMon** and Σ**Mon** have finite products: given (M_1, Σ_1) and (M_2, Σ_2), their product is $(M_1 \times M_2, \Sigma)$ where $\sum_{i\in I}(x_i, y_i) = (\sum_1 x_i, \sum_2 y_i)$ for all summable families $\{(x_i, y_i)\}_{i\in I}$ in $M_1 \times M_2$. The zero object $\mathbf{0}$ is $(\{0\}, \sum)$ in which all families are summable, with sum equal to 0. In particular, **PAMon** and Σ**Mon** are symmetric monoidal categories with product as the tensor.

We can also define finite coproducts in **PAMon** and Σ**Mon** as follows: given (M_1, Σ_1) and (M_2, Σ_2), their coproduct is (M, Σ) where $M = M_1 - \{0\} \uplus M_2 - \{0\} \uplus \{0\}$, i.e. M is the disjoint union of M_1 and M_2 with the 0 elements identified.

A **PAMon**-category $\mathbb{C}$ is a category enriched in **PAMon**; that is, the homsets are enriched with an additive structure such that composition distributes over addition from left and right. More specifically, for all $f : W \longrightarrow X, h : Y \longrightarrow Z$ and for all summable families $\{g_i\}_{i\in I}$ in $\mathbb{C}(X, Y)$, $\{g_i f\}_{i\in I}$ and $\{hg_i\}_{i\in I}$ are also summable and

$$(\textstyle\sum_{i\in I} g_i)f = \sum_{i\in I} g_i f$$

$$h(\textstyle\sum_{i\in I} g_i) = \sum_{i\in I} hg_i$$

Note that such categories have non-empty homsets and automatically have zero morphisms, namely $0_{XY} : X \longrightarrow Y = \sum_{i\in\emptyset} f_i$ for $f_i \in \mathbb{C}(X, Y)$. Note that this does not imply the existence of the zero object; however existence of the initial or terminal object will guarantee the existence of the zero object in the presence of zero morphisms [90].

Notation: We will use $+$ for the addition operation on the homsets. We use $\oplus$ for coproduct and $\tilde{\oplus}$ for biproduct.

Definition 3.0.10 Let $\mathbb{C}$ be a **PAMon**-category with countable coproducts $\bigoplus_{i\in I} X_i$. We define the following:

(i) A countable family $\{f_i\}_I : Y \longrightarrow X_i$ is *quasi summable* iff the family $\{in_i f_i\}_I : Y \longrightarrow \bigoplus_{i\in I} X_i$ is summable.

(ii) *Quasi projections* $\rho_j : \bigoplus_{i \in I} X_i \longrightarrow X_j$ for all $j \in I$ are defined as follows: $\rho_j in_k = 1_{X_j}$ if $k = j$ and $0_{X_k X_j}$ otherwise. Note that ρ_j exists for all $j \in I$ since $\mathbb{C}$ has zero morphisms.

Definition 3.0.11 A *partially additive category* $\mathbb{C}$ is a **PAMon**-category with countable coproducts which satisfies the following axioms:

1. *Compatible Sum Axiom*: If $\{f_i\}_I \in \mathbb{C}(X, Y)$ is a countable family and there exists $f : X \longrightarrow I.Y$ such that

$$\begin{tikzcd} X \arrow[r, "f"] \arrow[dr, "f_i"'] & I.Y \arrow[d, "\rho_i"] \\ & Y \end{tikzcd}$$

commutes (we say the f_i are *compatible*), then $\sum_{i \in I} f_i$ exists.

2. *Untying Axiom*: If $f + g : X \longrightarrow Y$ exists then so does $in_1 f + in_2 g : X \longrightarrow Y + Y$.

The dual of a partially additive category (PAC) is a **PAMon**-category with countable products which satisfies the dual of the above axioms.

We will next review the main theorems of PACs, namely (i) the unique decomposition property, (ii) the existence of the iteration (dagger) operation, and (iii) the uniqueness of the additive structure.

Proposition 3.0.12 (Manes and Arbib) *Given $f : X \longrightarrow \bigoplus_{i \in I} Y_i$ in a partially additive category. There exists a unique family $f_i : X \longrightarrow Y_i$ with $f = \sum_{i \in I} in_i f_i$, namely, $f_i = \rho_i f$.*

Proof. [90], Proposition 15, page 80. $\square$

Corollary 3.0.13 *Given $f : \bigoplus_{j\in J} X_j \longrightarrow \bigoplus_{i\in I} Y_i$ in a partially additive category, there exists a unique family $\{f_{ij}\}_{i\in I, j\in J} : X_j \longrightarrow Y_i$ with $f = \sum_{i\in I, j\in J} in_i f_{ij} \rho_j$, namely, $f_{ij} = \rho_i f in_j$.*

Proof. In any PAC, $\sum_{i\in I} in_i\rho_i$ exists and

$$\textstyle\sum_{i\in I} in_i\rho_i = 1 : \bigoplus_{i\in I} X_i \longrightarrow \bigoplus_{i\in I} X_i.$$

To see this, note that by the theorem above $1_{\bigoplus_I X_i}$ can be uniquely written as $1 = \sum_{i\in I} in_i\rho_i 1 = \sum_{i\in I} in_i\rho_i$.

Now let $f = 1_{\bigoplus_I Y_i} f 1_{\bigoplus_J X_j} = (\sum_i in_i\rho_i) f (\sum_j in_j\rho_j) = (\sum_i in_i\rho_i f)(\sum_j in_j\rho_j) = \sum_{ij} in_i\rho_i f in_j\rho_j = \sum_{ij} in_i f_{ij}\rho_j$.

For uniqueness, suppose there is another family $\{g_{kl}\}_{k\in I, l\in J}$ such that $f = \sum_{kl} in_k g_{kl}\rho_l$. Then $f_{ij} = \rho_i f in_j = \sum_{kl} \rho_i in_k g_{kl}\rho_l in_j = g_{ij}$ for all i, j.

□

Based on this proposition, every morphism $f : \bigoplus_J X_j \longrightarrow \bigoplus_I Y_i$ can be represented by its components. When I and J are finite, we will use the corresponding matrices to represent morphisms, for example f above with $|I| = m$ and $|J| = n$ is represented by an $m \times n$ matrix.

$$f = \begin{bmatrix} f_{11} & \dots & f_{1n} \\ \vdots & \vdots & \vdots \\ f_{m1} & \dots & f_{mn} \end{bmatrix}$$

Given $f : \bigoplus_K X_k \longrightarrow \bigoplus_J Y_j$ and $g : \bigoplus_J Y_j \longrightarrow \bigoplus_I Z_i$, let $h = gf$, then $h_{ik} = \rho_i h in_k = \rho_i (gf) in_k = \rho_i g(\sum_{j\in J} in_j\rho_j) f in_k = \sum_{j\in J} (\rho_i g in_j\rho_j) f in_k = \sum_{j\in J} g_{ij}\rho_j f in_k = \sum_{j\in J} g_{ij} f_{jk}$.

Therefore, the composition of morphisms in a PAC with finite coproducts in their domain and codomain corresponds to matrix multiplication of their matricial representations.

Remark. Note that although any morphism $f : \bigoplus_J X_j \longrightarrow \bigoplus_I Y_i$ can be represented by the unique family $\{f_{ij}\}_{i\in I, j\in J}$ of its components, the converse is not necessarily true, that is to say given a family $\{f_{ij}\}$ there may not be a morphism $f : \bigoplus_J X_j \longrightarrow \bigoplus_I Y_i$ satisfying $f = \sum_{ij} in_i f_{ij}\rho_j$. However, in case such an f exists it will be unique.

Theorem 3.0.14 (Manes and Arbib) *Given a map $f : X \longrightarrow Y \oplus X$ in a partially additive category. The sum*

$$f^\dagger = \textstyle\sum_{n=0}^{\infty} f_1 f_2^n : X \longrightarrow Y$$

exists, where $f_1 : X \longrightarrow Y$ and $f_2 : X \longrightarrow X$ are the components of f.

Proof. See [90] Theorem 24, page 83. $\square$

$f^\dagger$ is called the *iterate* of f. We will look at the properties of iteration in more detail. These properties will provide every PAC with a trace operator which is induced by the iteration operation.

Theorem 3.0.15 (Manes and Arbib, [90]) *The addition operation of a partially additive category is unique: If $\mathbb{C}$ is a partially additive category, then a family $\{f_i\}_{i\in I}$ in $\mathbb{C}(X, Y)$ is summable iff it is compatible. In that case, the $f : X \longrightarrow I.Y$ with $\rho_i f = f_i$ is unique, and $\sum f_i = X \xrightarrow{f} I.Y \xrightarrow{\sigma} Y$, where σ is defined by $\sigma in_i = 1_Y$.*

Proposition 3.0.16 *Let $\mathbb{C}$ be a PAC and $\{f_i\}_{i\in I} : Y \longrightarrow X_i$ be a quasi summable family. Then there exists a unique $f : Y \longrightarrow \bigoplus_{i\in I} X_i$ such that the following diagram commutes:*

$$\begin{tikzcd}
\bigoplus_{i\in I} X_i \arrow[r, "\rho_i"] & X_i \\
Y \arrow[u, "f"] \arrow[ur, "f_i"'] &
\end{tikzcd}$$

Proof. Let $f = \sum_{j\in I} in_j f_j : Y \longrightarrow \bigoplus_{i\in I} X_i$. Then $\rho_i f = \rho_i(\sum_{j\in I} in_j f_j) = \sum_{j\in I} \rho_i in_j f_j = f_i$.

Suppose $h : Y \longrightarrow \bigoplus_{i\in I} X_i$ is another morphism such that $\rho_i h = f_i$. h can be uniquely written as $h = \sum_{i\in I} in_i h_i$ for $h_i : Y \longrightarrow X_i$. Hence, we have $h_i = \rho_i h = f_i$ for all $i \in I$, and therefore $h = f$, which implies the desired result. $\square$

Theorem 3.0.17 *Let $\mathbb{C}$ be a category. $\mathbb{C}$ is a PAC with all families summable iff $\mathbb{C}$ has countable biproducts.*

Proof. Suppose $\mathbb{C}$ is a PAC, then it has countable coproducts $\bigoplus_{i\in I} X_i$. We will show that $(\bigoplus_{i\in I} X_i, \{\rho_i\}_I)$ is a product. Given an object Y and a family $\{f_i\}_{i\in I}$ in $\mathbb{C}(Y, X_i)$, let $f = \sum_{i\in I} in_i f_i : Y \longrightarrow \bigoplus_{i\in I} X_i$. Note that f exists since all families are summable, thus $\{f_i\}$ is a quasi summable family. By Proposition 3.0.16, there exists a unique morphism $f = \sum_{i\in I} in_i f_i : Y \longrightarrow \bigoplus_{i\in I} X_i$ such that $\rho_i f = f_i$. Also, note that $\rho_i in_j = \delta_{ij}$ for all i and j by definition of quasi projections and $\sum_I in_i \rho_i = 1_{\bigoplus_I X_i}$ (see the proof of Corollary 3.0.13.) Hence, $\mathbb{C}$ has countable biproducts. In particular, $\mathbb{C}$ is a semiadditive category.

Now let $\mathbb{C}$ be a category with countable biproducts. $\mathbb{C}$ has a zero object and hence families of zero morphisms. Given a family $\{f_i\}_I$ of morphisms in $\mathbb{C}(X, Y)$ with countable I, define $\sum_{i\in I} f_i = \nabla_Y(\tilde{\bigoplus}_{i\in I} f_i)\Delta_X$ where $\Delta_X : X \longrightarrow I.X$ and $\nabla_Y : I.Y \longrightarrow Y$ are defined by: $\pi_i\Delta_X = 1_X$ and $\nabla_Y in_i = 1_Y$ for all $i \in I$ and define $\sum_{i\in\emptyset} f_i = 0_{XY}$. Note that $I.X$ and $I.Y$ denote the biproduct of $|I|$ copies of X and Y respectively. This makes every homset $\mathbb{C}(X, Y)$ into a commutative monoid with $\sum_{i\in\emptyset} f_i$ as the additive identity. More generally, using the associativity and symmetry isomorphisms of biproduct, it follows that every homset $\mathbb{C}(X, Y)$ is a partially additive monoid with all families summable (see [85, 96] for similar proofs in the case of additive categories.) Also, composition distributes over addition.

As all families are summable, the compatible sum and untying axioms are trivially true. Quasi projections are just the projection morphisms of the biproduct and $\pi_i in_i = 1$ and $\pi_i in_j = 0$ for $i \neq j$ and $\sum_{i\in I} in_i \pi_i = 1_{\tilde{\bigoplus}_I X_i}$ by definition of biproduct. Here $\pi_i : \tilde{\bigoplus}_{i\in I} X_i \longrightarrow X_i$ and $in_i : X_i \longrightarrow \tilde{\bigoplus}_{i\in I} X_i$. $\square$

Definition 3.0.18 A functor $T : \mathbb{C} \longrightarrow \mathbb{D}$ between two PACs is said to be *additive* if $T_{X,Y} : \mathbb{C}(X, Y) \longrightarrow \mathbb{D}(TX, TY)$ is a **PAMon** morphism for all objects X, Y in $\mathbb{C}$. That is, for every summable family $\{f_i\}$ in $\mathbb{C}(X, Y)$, $\{T_{X,Y}(f_i)\}$ is summable in $\mathbb{D}(TX, TY)$ and $T_{X,Y}(\sum_i f_i) = \sum_i T_{X,Y}(f_i)$.

3.1 Iteration

By Theorem 3.0.14 we know that a morphism $f : X \longrightarrow Y \oplus X$ in a PAC $\mathbb{C}$ induces a morphism $f^\dagger : X \longrightarrow Y$ called the iterate (or dagger) of f. Hence we can define a family of operations $\dagger_{X,Y} : \mathbb{C}(X, Y \oplus X) \longrightarrow \mathbb{C}(X,Y)$ which to each f associates its iterate $f^\dagger$ (we also use the notation $\dagger_{X,Y}(f)$.) We will show that this operation induces a trace operator on $\mathbb{C}$. Therefore we continue by studying some interesting and useful properties of the dagger operation.

Lemma 3.1.1 *Let $\mathbb{C}$ be a PAC and $f : X \longrightarrow Y \oplus X$ a morphism in $\mathbb{C}$. Also let f_1, f_2 and $f^\dagger$ be as in Theorem 3.0.14. Then, the equation $f^\dagger = f^\dagger f_2 + f_1$ holds. Moreover, $f^\dagger$ is a solution of the* Elgot equation $[1_Y, \zeta]f = \zeta$ *where $\zeta : X \longrightarrow Y$.*

Proof. Consider the equation

$$\begin{aligned} f^\dagger f_2 + f_1 &= (\textstyle\sum_{n=0}^{\infty} f_1 f_2^n) f_2 + f_1 \\ &= \textstyle\sum_{n=1}^{\infty} f_1 f_2^n + f_1 \\ &= \textstyle\sum_{n=0}^{\infty} f_1 f_2^n = f^\dagger \end{aligned}$$

$$\begin{aligned} [1_Y, f^\dagger] f &= [1_Y, f^\dagger](in_1 f_1 + in_2 f_2) \\ &= [1_Y, f^\dagger] in_1 f_1 + [1_Y, f^\dagger] in_2 f_2 \\ &= f_1 + f^\dagger f_2 = f^\dagger. \end{aligned}$$

□

We say that $\dagger$ satisfies the *fixed-point identity*, if $[1_Y, f^\dagger]f = f^\dagger$ holds for all $f : X \longrightarrow Y \oplus X$.

Lemma 3.1.2 (Star sum identity) *Let $\mathbb{C}$ be a PAC and $f, g : X \longrightarrow X$ be morphisms in $\mathbb{C}$. Then, $\sum_{n\in\omega}(f+g)^n$, if it exists, is given by*

$$\textstyle\sum_{n\in\omega}(f+g)^n = \displaystyle\sum_{q\in\omega}(\sum_{m\in\omega} g^m)(\sum_{p\in\omega} f g^p)^q.$$

Proof. Observe that $\sum_{n\in\omega}(f+g)^n = 1+f+g+ff+fg+gf+gg+fff+\cdots$, the sum of all products of f's and g's. We can view every term as either having no occurrence of f or being the product of arbitrary copies of g partitioned by occurrences of f. For example, $fffgg = f^3g^2 = g^0fg^0fg^0fg^2$, $gg = g^2$ with no occurrence of f and $fff = g^0fg^0fg^0fg^0$. On the other hand, a typical term of $\sum_{q\in\omega}(\sum_{m\in\omega} g^m)(\sum_{p\in\omega} fg^p)^q$ is $g^j(fg^{i_1})(fg^{i_2})\cdots(fg^{i_k})$, a product of g's partitioned by occurrences of f or no occurrence of f, i.e. with $q = 0$. $\square$

The identity in Lemma 3.1.2 is called the *star sum* identity in the theory of Kleene algebras [21].

Proposition 3.1.3 *Let $\mathbb{C}$ be a PAC, with $\dagger_{X,Y} : \mathbb{C}(X, Y \oplus X) \longrightarrow \mathbb{C}(X, Y)$. Then, $\dagger$ satisfies the following properties:*

- *Naturality in Y*

$$\begin{array}{ccc} \mathbb{C}(X, Y\oplus X) & \xrightarrow{\dagger_{X,Y}} & \mathbb{C}(X,Y) \\ {\scriptstyle \mathbb{C}(1_X,\, f\oplus 1_X)}\downarrow & & \downarrow{\scriptstyle \mathbb{C}(1_X,\, f)} \\ \mathbb{C}(X, Y'\oplus X) & \xrightarrow{\dagger_{X,Y'}} & \mathbb{C}(X,Y') \end{array}$$

or in other words, for all $h : X \longrightarrow Y \oplus X$:

$$f \dagger_{X,Y}(h) = \dagger_{X,Y'}((f \oplus 1_X)h) \tag{7}$$

where $f : Y \longrightarrow Y'$.

- *Dinaturality in X*

$$\begin{tikzcd}
 & \mathbb{C}(X, Y \oplus X) \arrow[r, "\dagger_{X,Y}"] & \mathbb{C}(X, Y) \arrow[dd, "\mathbb{C}(k, 1_Y)"] \\
\mathbb{C}(X, Y \oplus U) \arrow[ur, "\mathbb{C}(1_X, 1_Y \oplus k)"] \arrow[dr, "\mathbb{C}(k, 1_{Y \oplus U})"'] & k : U \to X & \\
 & \mathbb{C}(U, Y \oplus U) \arrow[r, "\dagger_{U,Y}"] & \mathbb{C}(U, Y)
\end{tikzcd}$$

or in other words, for all $h : X \to Y \oplus U$

$$(\dagger_{X,Y}((1_Y \oplus k)h))k \;=\; \dagger_{U,Y}(hk)\ . \tag{8}$$

- *Double dagger identity*

$$\begin{tikzcd}
\mathbb{C}(X, Y \oplus X \oplus X) \arrow[r, "\dagger_{X,Y \oplus X}"] \arrow[d, "\mathbb{C}(1_X, 1_Y \oplus \nabla_X)"'] & \mathbb{C}(X, Y \oplus X) \arrow[d, "\dagger_{X,Y}"] \\
\mathbb{C}(X, Y \oplus X) \arrow[r, "\dagger_{X,Y}"] & \mathbb{C}(X, Y)
\end{tikzcd}$$

where $\nabla_X = [1_X, 1_X]$ *or in other words, for all* $h : X \longrightarrow Y \oplus X \oplus X$

$$\dagger_{X,Y}(\dagger_{X,Y \oplus X}(h)) = \dagger_{X,Y}((1_Y \oplus \nabla_X)h). \tag{9}$$

- *Bekič Identity*

  Given $f : Y \longrightarrow X \oplus Y \oplus Z$ *and* $g : Z \longrightarrow X \oplus Y \oplus Z$*, define* $h = [1_{X \oplus Y}, \dagger_{Z,X \oplus Y}(g)]f : Y \longrightarrow X \oplus Y$*. We have*

$$\dagger_{Y \oplus Z,X}([f, g]) = [1_X, \dagger_{Y,X}(h)][in_2^{X,Y}, \dagger_{Z,X \oplus Y}(g)]. \tag{10}$$

- *Uniformity or functoriality [90, 7, 21]*

If $\begin{tikzcd} X \arrow[r, "f"] \arrow[d, "h"'] & Y \oplus X \arrow[d, "k \oplus h"] \\ X' \arrow[r, "g"'] & Y' \oplus X' \end{tikzcd}$ *commutes then* $\begin{tikzcd} X \arrow[r, "f^\dagger"] \arrow[d, "h"'] & Y \arrow[d, "k"] \\ X' \arrow[r, "g^\dagger"'] & Y' \end{tikzcd}$ *commutes too.*

Proof. We will sometimes use $f^\dagger$ instead of $\dagger(f)$.

1. Naturality in Y: Write $h_1 : X \longrightarrow Y$ and $h_2 : X \longrightarrow X$ for the components of h.

$$\begin{aligned} f\ \dagger_{X,Y}(h) &= f(\textstyle\sum_n h_1 h_2^n) \\ &= \textstyle\sum_n (fh_1) h_2^n \end{aligned}$$

$$\begin{aligned} ((f \oplus 1_X)h)^\dagger &= ((f \oplus 1_X)(in_1 h_1 + in_2 h_2))^\dagger \\ &= ([in_1 f, in_2 1_X](in_1 h_1 + in_2 h_2))^\dagger \\ &= ([in_1 f, in_2 1_X] in_1 h_1 + [in_1 f, in_2 1_X] in_2 h_2)^\dagger \\ &= (in_1 f h_1 + in_2 h_2)^\dagger \\ &= \textstyle\sum_n (fh_1) h_2^n. \end{aligned}$$

Hence the identity follows.

2. Dinaturality in X: Write $h = in_1 h_1 + in_2 h_2$ for unique $h_1 : X \longrightarrow Y$ and $h_2 : X \longrightarrow U$.

$$\begin{aligned} (hk)^\dagger &= ((in_1 h_1 + in_2 h_2)k)^\dagger \\ &= (in_1 h_1 k + in_2 h_2 k)^\dagger \\ &= \textstyle\sum_{n=0}^\infty (h_1 k)(h_2 k)^n. \end{aligned}$$

$$\begin{aligned}
((1_Y \oplus k)h)^\dagger k &= ((1_Y \oplus k)(in_1 h_1 + in_2 h_2))^\dagger k \\
&= ([in_1, in_2 k](in_1 h_1 + in_2 h_2))^\dagger k \\
&= ([in_1, in_2 k] in_1 h_1 + [in_1, in_2 k] in_2 h_2)^\dagger k \\
&= (in_1 h_1 + in_2 k h_2)^\dagger k \\
&= (\textstyle\sum_{n=0}^{\infty} h_1 (k h_2)^n) k \\
&= (h_1 + h_1(kh_2) + h_1(kh_2)(kh_2) + \cdots) k \\
&= h_1 k + h_1(kh_2)k + h_1(kh_2)(kh_2)k + \cdots \\
&= h_1 k + (h_1 k)(h_2 k) + (h_1 k)(h_2 k)^2 + \cdots \\
&= \textstyle\sum_{n=0}^{\infty} (h_1 k)(h_2 k)^n
\end{aligned}$$

Hence the identity follows.

3. Double dagger identity:

Using the matricial representation of $h : X \longrightarrow Y \oplus X \oplus X$, we have (all indices range over ω):

$$\dagger_{X,Y\oplus X}(h) = \textstyle\sum_n \begin{bmatrix} h_1 \\ h_2 \end{bmatrix} h_3^n$$

and

$$\dagger_{X,Y}(\dagger_{X,Y\oplus X}(h)) = \dagger_{X,Y}\left(\begin{bmatrix} \sum_n h_1 h_3^n \\ \sum_n h_2 h_3^n \end{bmatrix}\right) = \textstyle\sum_q (\sum_m h_1 h_3^m)(\sum_p h_2 h_3^p)^q.$$

On the other hand,

$$\dagger_{X,Y}((1_Y \oplus \nabla_X)h) = \dagger_{X,Y}\left(\begin{bmatrix} 1_Y & 0 & 0 \\ 0 & 1_X & 1_X \end{bmatrix}\begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix}\right) = \dagger\left(\begin{bmatrix} h_1 \\ h_2 + h_3 \end{bmatrix}\right) = \textstyle\sum_n h_1 (h_2 + h_3)^n.$$

Finally, $\sum_n h_1(h_2 + h_3)^n = \sum_q (\sum_m h_1 h_3^m)(\sum_p h_2 h_3^p)^q$ by Lemma 3.1.2.

4. Bekič identity:

It is known that Bekič identity follows from the fixed-point, naturality, dinaturality and double dagger identities [21]. In [21], the Bekič identity is called the pairing identity.

5. Uniformity:

First note that

$$\begin{aligned}(k \oplus h)f &= [in_1 k, in_2 h](in_1 f_1 + in_2 f_2) \\ &= in_1 k f_1 + in_2 h f_2 \\ &= gh \\ &= (in_1 g_1 + in_2 g_2)h \\ &= in_1 g_1 h + in_2 g_2 h\end{aligned}$$

Therefore, by uniqueness of decomposition

$$k f_1 = g_1 h \qquad h f_2 = g_2 h \tag{11}$$

$$\begin{aligned}k f^\dagger &= k(\textstyle\sum_n f_1 f_2^n) \\ &= \textstyle\sum_n (k f_1) f_2^n \\ &= \textstyle\sum_n (g_1 h) f_2^n \text{ using equation (11)}\end{aligned}$$

$$\begin{aligned}g^\dagger h &= \textstyle\sum_n (g_1 g_2^n) h \\ &= g_1 h + g_1 g_2 h + g_1 g_2^2 h + \cdots \\ &= g_1 h + g_1 h f_2 + g_1 h f_2^2 + g_1 h f_2^3 + \cdots \text{ using equation (11) repeatedly,} \\ &= \textstyle\sum_n (g_1 h) f_2^n\end{aligned}$$

□

Proposition 3.1.4 *Every partially additive category is a traced symmetric monoidal category, where given $f : X \oplus U \longrightarrow Y \oplus U$,*

$$Tr^U_{X,Y} = f_{11} + \textstyle\sum_{n \in \omega} f_{12} f_{22}^n f_{21}$$

and f_{ij} are the components of f.

Proof. Given $f : X \oplus U \longrightarrow Y \oplus U$, let $f_1 =_{def} fin_1^{X,U} : X \longrightarrow Y \oplus U$ and $f_2 =_{def} fin_2^{X,U} : U \longrightarrow Y \oplus U$. By Theorem 3.0.14, $f_2^\dagger = \sum_{n=0}^{\infty} f_{12} f_{22}^n$. Also, by Proposition 3.0.12, f_1 can be written as $f_1 = in_1 f_{11} + in_2 f_{21}$. Define $Tr_{X,Y}^U(f) = [1_Y, f_2^\dagger] f_1$. Then $Tr_{X,Y}^U(f) = [1_Y, f_2^\dagger](in_1 f_{11} + in_2 f_{21}) = f_{11} + (\sum_{n=0}^{\infty} f_{12} f_{22}^n) f_{21} = f_{11} + \sum_{n=0}^{\infty} f_{12} f_{22}^n f_{21}$. We need to verify the axioms of trace:

1. Naturality in X:
$Tr_{X,Y}^U(f)g = Tr_{X',Y}^U(f(g \oplus 1_U))$ where $f : X \oplus U \longrightarrow Y \oplus U, g : X' \longrightarrow X$. let $h = f(g \oplus 1)$, then $h_1 = hin_1 = f([in_1 g, in_2])in_1 = fin_1 g = f_1 g$. Also $h_2 = hin_2 = f([in_1 g, in_2])in_2 = fin_2 = f_2$. Therefore, $Tr_{X',Y}^U(h) = [1_Y, h_2^\dagger]h_1 = [1_Y, f_2^\dagger]f_1 g = Tr_{X,Y}^U(f)g$.

2. Naturality in Y :
$gTr_{X,Y}^U(f) = Tr_{X,Y'}^U((g \oplus 1_U)f)$ where $f : X \oplus U \longrightarrow Y \oplus U, g : Y \longrightarrow Y'$. Let $h = (g \oplus 1)f$, then $h_1 = hin_1 = [in_1 g, in_2]fin_1 = [in_1 g, in_2]f_1$ and $h_2 = hin_2 = (g \oplus 1)fin_2 = (g \oplus 1)f_2$. Note that $h_2^\dagger = ((g \oplus 1)f_2)^\dagger = gf_2^\dagger$ by the naturality property of dagger. Therefore, $Tr_{X,Y'}^U(h) = [1_{Y'}, h_2^\dagger]h_1 = [1_{Y'}, gf_2^\dagger][in_1 g, in_2]f_1 = [g, gf_2^\dagger]f_1 = g[1_Y, f_2^\dagger]f_1 = gTr_{X,Y}^U(f)$.

3. Dinaturality in U:
$Tr_{X,Y}^U((1_Y \oplus g)f) = Tr_{X,Y}^{U'}(f(1_X \oplus g))$ where $f : X \oplus U \longrightarrow Y \oplus U'$ and $g : U' \longrightarrow U$. Let $h = (1_Y \oplus g)f$, then $h_1 = [in_1, gin_2]f_1$ and $h_2 = (1_Y \oplus g)fin_2 = (1_Y \oplus g)f_2$. Therefore, $Tr_{X,Y}^U(h) = [1_Y, h_2^\dagger]h_1 = [1_Y, ((1_Y \oplus g)f_2)^\dagger][in_1, gin_2]f_1 = [1_Y, ((1_Y \oplus g)f_2)^\dagger g]f_1 = [1_Y, (f_2 g)^\dagger]f_1$, by dinaturality property of dagger.

Now let $l = f(1 \oplus g)$, then $l_1 = f[in_1, in_2 g]in_1 = fin_1 = f_1$ and $l_2 = f(1_X \oplus g)in_2 = f[in_1, in_2 g]in_2 = f_2 g$. Hence, $Tr(l) = [1, (f_2 g)^\dagger]f_1$.

4.1. Vanishing I:
$Tr_{X,Y}^I(f) = f$ where $f : X \oplus I \longrightarrow Y \oplus I$ and I is the zero object. We use $0_{IY} : I \longrightarrow Y$ to denote the unique map (zero morphism) from I to Y. $Tr_{X,Y}^I(f) = [1_Y, 0_{IY}]fin_1^{X,I} = [1_Y, 0_{IY}]f1_X = 1_Y f = f$.

4.2 Vanishing II:
$Tr^{U\oplus V}_{X,Y}(f) = Tr^{U}_{X,Y}(Tr^{V}_{X\oplus U,Y\oplus U}(f))$ where $f : X\oplus U\oplus V \longrightarrow Y\oplus U\oplus V$. We will suppress the subscripts and simply write Tr^U and Tr^V instead. Note that $Tr^V(f) = [1_{Y\oplus U},(fin_2^{X\oplus U,V})^\dagger]fin_1^{X\oplus U,V}$ and $Tr^U(Tr^V(f)) = [1_Y,(Tr^V(f)in_2^{X,U})^\dagger]Tr^V(f)in_1^{X,U}$.

On the other hand $Tr^{U\oplus V}(f) = [1_Y,(fin_2^{X,U\oplus V})^\dagger]fin_1^{X,U\oplus V}$. Note that $fin_2^{X,U\oplus V} : U\oplus V \longrightarrow Y\oplus U\oplus V$ and hence can be expressed as $[F,G]$ where:

$$F : U \longrightarrow Y\oplus U\oplus V = fin_2^{X,U\oplus V}in_1^{U,V}$$

$$G : V \longrightarrow Y\oplus U\oplus V = fin_2^{X,U\oplus V}in_2^{U,V} = fin_2^{X\oplus U,V}$$

$$\begin{aligned}
Tr^{U\oplus V}(f) &= [1_Y,(fin_2^{X,U\oplus V})^\dagger]fin_1^{X,U\oplus V}\\
&= [1_Y,[F,G]^\dagger]fin_1^{X,U\oplus V}\\
&= [1_Y,[1_Y,([1_{Y\oplus U},G^\dagger]F)^\dagger][in_2^{Y,U},G^\dagger]]fin_1^{X,U\oplus V} \text{ using Bekič identity of dagger}\\
&= [1_Y,([1_{Y\oplus U},G^\dagger]F)^\dagger][1_{Y\oplus U},G^\dagger]fin_1^{X,U\oplus V}\\
&= [1_Y,([1_{Y\oplus U},(fin_2^{X\oplus U,V})^\dagger]fin_2^{X,U\oplus V}in_1^{U,V})^\dagger][1_{Y\oplus U},(fin_2^{X\oplus U,V})^\dagger]fin_1^{X\oplus U,V}in_1^{X,U}\\
&= [1_Y,([1_{Y\oplus U},(fin_2^{X\oplus U,V})^\dagger]fin_2^{X,U\oplus V}in_1^{U,V})^\dagger]Tr^V(f)in_1^{X,U}\\
&= [1_Y,([1_{Y\oplus U},(fin_2^{X\oplus U,V})^\dagger]fin_1^{X\oplus U,V}in_2^{X,U})^\dagger]Tr^V(f)in_1^{X,U}\\
&= [1_Y,(Tr^V(f)in_2^{X,U})^\dagger]Tr^V(f)in_1^{X,U}\\
&= Tr^U(Tr^V(f)).
\end{aligned}$$

5. Superposing:
Recall from Proposition 2.1.18 that the superposing axiom is equivalent to the following in the presence of the other axioms: $g\oplus Tr^U_{X,Y}(f) = Tr^U_{W\oplus X,Z\oplus Y}(g\oplus f)$ where $f : X\oplus U \longrightarrow Y\oplus U$ and $g : W \longrightarrow Z$. Let $h = g\oplus f$, then $h_1 = hin_1^{W\oplus X,U} = g\oplus fin_1^{X,U} = g\oplus f_1$. Also, $h_2^\dagger = hin_2^{W\oplus X,U} = ((g\oplus f)in_2^{W\oplus X,U})^\dagger =$

$(in_2^{Z,Y} \oplus 1_U) f in_2^{X,U} = (in_2^{Z,Y} \oplus 1_U) f_2$.

$$
\begin{aligned}
Tr^U_{W\oplus X, Z\oplus Y}(g \oplus f) &= [1_{Z\oplus Y}, ((in_2^{Z,Y} \oplus 1_U) f_2)^\dagger](g \oplus f_1) \\
&= [1_{Z\oplus Y}, in_2^{Z,Y} f_2^\dagger](g \oplus f_1) \qquad \text{by naturality property of dagger} \\
&= [1_Z \oplus 1_Y, in_2^{Z,Y} f_2^\dagger](g \oplus f_1) \\
&= [[in_1^{Z,Y}, in_2^{Z,Y}], in_2^{Z,Y} f_2^\dagger](g \oplus f_1) \\
&= [in_1^{Z,Y}, [in_2^{Z,Y}, in_2^{Z,Y} f_2^\dagger]](g \oplus f_1) \\
&= [in_1^{Z,Y} g, in_2^{Z,Y} [1_Y, f_2^\dagger] f_1] \\
&= [in_1^{Z,Y} g, in_2^{Z,Y} Tr^U_{X,Y}(f)] \\
&= g \oplus Tr^U_{X,Y}(f)
\end{aligned}
$$

6. Yanking :
$Tr^U_{U,U}(\sigma_{U,U}) = 1_U$. $\sigma_1 = \sigma in_1$ and $\sigma_2 = \sigma in_2$.

Therefore,

$$
\begin{aligned}
Tr^U_{U,U}(\sigma) &= [1_U, (\sigma in_2)^\dagger] \sigma in_1 \\
&= [(\sigma in_2)^\dagger, 1_U] in_1 \\
&= (\sigma in_2)^\dagger \\
&= [1_U, (\sigma in_2)^\dagger] \sigma in_2, \ \text{using the fixed-point identity for dagger} \\
&= [(\sigma in_2)^\dagger, 1_U] in_2 \\
&= 1_U.
\end{aligned}
$$

□

Note that the proof given above does not use the additive structure of the underlying category $\mathbb{C}$. In particular, we conclude a more general theorem:

Theorem 3.1.5 *Let $\mathbb{C}$ be a category with finite coproducts and have a dagger operator $(\dagger)$ satisfying the fixed-point, naturality, dinaturality and double dagger identities (see Proposition 3.1.3.) Then, $\mathbb{C}$ is a traced symmetric monoidal category.*

A trace operator in a traced symmetric monoidal category $\mathbb{C}$ is said to be *uniform* [59] if for any $h : U \longrightarrow U'$, f and g, whenever

$$\begin{tikzcd} X \otimes U \arrow[r, "f"] \arrow[d, "1_X \otimes h"'] & Y \otimes U \arrow[d, "1_Y \otimes h"'] \\ X \otimes U' \arrow[r, "g"] & Y \otimes U' \end{tikzcd}$$

commutes, then $Tr^U_{X,Y}(f) = Tr^{U'}_{X,Y}(g)$.

Proposition 3.1.6 *The trace operator in a PAC is uniform.*

Proof. Let $f : X \oplus U \longrightarrow Y \oplus U$, $g : X \oplus U' \longrightarrow Y \oplus U'$ and $h : U \longrightarrow U'$ be morphisms in a PAC $\mathbb{C}$ such that $(1_Y \oplus h)f = g(1_X \oplus h)$. We need to show that $Tr^U_{X,Y}(f) = Tr^{U'}_{X,Y}(g)$. Recall that $Tr^U_{X,Y} = [1, f_2^\dagger]f_1$ with $f_1 = f in_1^{X,U}$ and $f_2 = f in_2^{X,U}$ and that the dagger operator in a PAC is uniform, see Proposition 3.1.3.

First observe that $(1_Y \oplus h)f_1 = (1_Y \oplus h)f in_1^{X,U} = g(1_X \oplus h)in_1^{X,U} = g in_1^{X,U'} = g_1$ and $(1_Y \oplus h)f_2 = (1_Y \oplus h)f in_2^{X,U} = g(1_X \oplus h)in_2^{X,U} = g in_2^{X,U'} h = g_2 h$. Therefore by uniformity property of dagger it follows that $f_2^\dagger = g_2^\dagger h$.

$$\begin{aligned} Tr^U_{X,Y}(f) &= [1_Y, f_2^\dagger]f_1 \\ &= [1_Y, g_2^\dagger h]f_1 \\ &= [1_Y, g_2^\dagger](1_Y \oplus h)f_1 \\ &= [1_Y, g_2^\dagger]g_1 \\ &= Tr^{U'}_{X,Y}(g). \end{aligned}$$

□

We remark that we have not used the additive structure of $\mathbb{C}$ in the proof of the theorem above. Hence, we have a more general result.

Proposition 3.1.7 *Let $\mathbb{C}$ be a category with finite coproducts and an iteration operator that satisfies the fixed-point, naturality, dinaturality, double dagger and uniformity identities. Then, $\mathbb{C}$ is a traced monoidal category with a uniform trace.*

Note that the uniformity property is not necessarily true in every traced symmetric monoidal category, for example it does not hold in $\mathbf{Rel}_\times$. Hence, this gives a criterion to distinguish between trace operators. The role played by the uniformity property of trace remains to be seen! See also [102].

We close this section by commenting on the dual of iteration operator on $\mathbb{C}^{op}$ where $\mathbb{C}$ is a PAC. By dualising the arguments above about iteration (dagger) we have the following:

Proposition 3.1.8 *Let $\mathbb{C}$ be a PAC. Then $\mathbb{C}^{op}$ has a fixed-point operator*

$$\mathsf{Y}_{A,B} : \mathbb{C}(A \times B, B) \longrightarrow \mathbb{C}(A, B)$$

satisfying the following properties:

- *Fixed-point identity*

$$h < 1_A, \mathsf{Y}_{A,B}(h) >= \mathsf{Y}_{A,B}(h)$$

  for $h : A \times B \longrightarrow B$.

- *Naturality in A*

$$\mathsf{Y}_{A,B}(h)f = \mathsf{Y}_{A',B}(h(f \times 1_B))$$

  for $f : A' \longrightarrow A$ and $h : A \times B \longrightarrow B$.

- *Dinaturality in B*

$$\mathsf{Y}_{A,B}(kh) = k\mathsf{Y}_{A,C}(h(1_A \times k))$$

  for $k : C \longrightarrow B$ and $h : A \times B \longrightarrow C$.

- *Diagonal property*

$$\mathsf{Y}_{A,B}(\mathsf{Y}_{A\times B,B}(h)) = \mathsf{Y}_{A,B}(h(1_A \times \Delta_B))$$

  for $h : A \times B \times B \longrightarrow B$. Here $\Delta_B =< 1_B, 1_B >$.

- *Bekič*

$$\mathsf{Y}_{A,B\times C}(< f,g >) =< \mathsf{Y}_{A,B}(h), \mathsf{Y}_{A\times B,C}(g) < 1_A, \mathsf{Y}_{A,B}(h) >>$$

for $f : A \times B \times C \longrightarrow B$ *and* $g : A \times B \times C \longrightarrow C$ *and where* $h = f < 1_{A\times B}, \mathsf{Y}_{A\times B,C}(g) >$.

Fixed-points play a central role in domain theory, where traditionally the least-fixed-point operator for continuous endofunctions on complete partial orders is used [93]. Recently, there has been considerable interest in developing a more general axiomatic account of the constructions of domain theory [98]. The axioms above for a fixed-point operator are the axiomatisation of abstract fixed-point operator introduced by Plotkin and Simpson [98]. For more on fixed-point and dagger operators see [22, 21, 89].

Hasegawa in his PhD thesis [58], studies the correspondence between the trace and fixed-point operators in a cartesian category. The author of this thesis has proven Theorem 3.1.5 in complete ignorance of Hasegawa's work. In view of the results in [58], we can state the following results whose proofs are simply obtained by dualising the arguments of the respective theorems for fixed-point identities given in [58].

Proposition 3.1.9 *Let $\mathbb{C}$ be a category with finite coproducts and a trace operator. Then $\mathbb{C}$ has a dagger operator satisfying the fixed-point, naturality, dinaturality and double dagger identities.*

Proof. Let $f : X \longrightarrow Y \oplus X$ be given, we define

$$f^\dagger =_{def} Tr^X_{X,Y}([f, f]).$$

As mentioned above the identities stated in the theorem can be obtained by dualising the arguments for a fixed-point operator. □

Finally in view of the results above we have a nice correspondence between dagger and trace operators in categories with finite coproducts.

Proposition 3.1.10 *Let $\mathbb{C}$ be a category with finite coproducts. There is a bijective correspondence between the dagger and trace operators whenever either one exists.*

Having established the connection between the dagger and trace operators, it becomes interesting to know the uniqueness of trace operation on a given PAC. In the categories **Pfn** and $\mathbf{Pfn}_D$ (see the next section) there is a unique dagger operation satisfying the Elgot equation (see [7] for a proof) and hence there is a unique trace operator on these categories.

3.2 Examples

1. Consider the category **Pfn** of sets and partial functions. Recall that a partial function from X to Y is a function from $Dom(f) \subseteq X$ to Y. Given $f : X \longrightarrow Y$ and $g : Y \longrightarrow Z$, $gf : X \longrightarrow Z$ is defined by

$$Dom(gf) = \{x \in X | x \in Dom(f), f(x) \in Dom(g)\}$$

$$(gf)(x) = g(f(x)) \text{ for } x \in Dom(gf).$$

The additive structure was given in Example 3.0.8, the zero morphism $0_{XY} : X \longrightarrow Y$ is the everywhere undefined partial function. Note that $Dom(gf) \subseteq Dom(f)$ and it easily follows that composition distributes over sum. **Pfn** has countable coproducts given by disjoint union.

To verify the compatible sum axiom, let $\{f_i\}_{i \in I}$ be a compatible family in $\mathbb{C}(X,Y)$ then $Dom(f_i) \cap Dom(f_j) = \emptyset$ for all $i \neq j$. Suppose otherwise, then $Dom(\rho_i) \cap Dom(\rho_j)$ will be nonempty, since $f_i = \rho_i f$ and $f_j = \rho_j f$. However, $Dom(\rho_i) = \{(y,i)|y \in Y\}$ and $Dom(\rho_j) = \{(y,j)|y \in Y\}$ with $i \neq j$. Hence $\{f_i\}_I$ is summable.

Let $f, g : X \longrightarrow Y$ be summable, then $Dom(in_1 f) \cap Dom(in_2 g) \subseteq Dom(f) \cap Dom(g) = \emptyset$. Therefore, $in_1 f$ and $in_2 g$ are summable. This verify the untying axiom.

A closely related example is the category $\mathbf{Pfn}_D$ for a set D where objects are sets and a morphism from X to Y is a partial function from $X \times D$ to $Y \times D$. The additive structure is the same as in **Pfn** with disjoint union as the coproduct.

The reader may be curious as to why we did not use the overlap sum structure on **Pfn** or $\mathbf{Pfn}_D$. The reason can be seen by recalling the uniqueness of additive structure in a PAC; Theorem 3.0.15. More specifically, the untying axiom fails for overlap sum.

2. Consider the category $\mathbf{Rel}_+$ of sets and binary relations. Given morphisms $R : X \longrightarrow Y$ and $S : Y \longrightarrow Z$ the composition $SR : X \longrightarrow Z$ is given by

$$(x,z) \in SR \text{ iff } \exists y \in Y.(x,y) \in R \text{ and } (y,z) \in S.$$

$1_X = \{(x,x)|x \in X\}$.

We have already defined the additive structure for the homsets in Example 3.0.8. Hence, for any X and Y, $\mathbf{Rel}_+(X,Y)$ is a partially additive monoid with all families summable and the zero morphism is the empty relation ($\emptyset \subseteq X \times Y$). Note that $\mathbf{Rel}_+$ has countable coproducts given by the disjoint union. Clearly composition distributes over sum and compatibility and untying axioms are trivially true since all families are summable.

A closely related example is the category $\mathbf{Rel}_D$ for a set D. Objects are sets and a morphism from X to Y is a relation from $X \times D$ to $Y \times D$. The additive structure is as in $\mathbf{Rel}_+$.

3. The following example appears in [90] where it is called $\mathbf{FwR}_{(M,\circ,e)}$ and in [89] where it is called $\mathbf{Pfn}_M$. We have chosen the latter notation. The objects are sets and an arrow from X to Y is a partial function from $X \longrightarrow Y \times M$, where $(M,\circ,e)$ is a monoid. If $f(x)$ is defined and equals (y,a), then $a \in M$ is interpreted as the reliability [90] or the attribute of f [89]. Given $f : X \longrightarrow Y$ and $g : Y \longrightarrow Z$, $Dom(gf)$ is the same as in **Pfn** and if $f(x) = (y,a)$ and $g(y) = (z,b)$, then $gf(x) = (z,ba)$. $1_X(x) = (x,e)$.

The additive structure is the same as in **Pfn**. Also $\mathbf{Pfn}_M$ has countable coproducts given by disjoint union. The necessary axioms of a PAC follow using similar arguments as in the case of **Pfn**.

5. Consider the category **SL** of complete sup semilattices and order and sup preserving maps. The additive structure is given by sup operation (see Example 3.0.8(5).)

Note that **SL** has countable biproducts given by cartesian product of sets with pointwise ordering. Hence, by Theorem 3.0.17, **SL** is a PAC with all families summable.

6. Consider the category **SRel** of stochastic relations with measurable spaces $(X, \mathcal{F}_X)$ as objects and stochastic kernels as arrows. An arrow $f : (X, \mathcal{F}_X) \longrightarrow (Y, \mathcal{F}_Y)$ is a map $f : X \times \mathcal{F}_Y \longrightarrow [0, 1]$ such that $f(\cdot, B) : X \longrightarrow [0, 1]$ is a bounded measurable function for fixed $B \in \mathcal{F}_Y$ and $f(x, \cdot) : \mathcal{F}_Y \longrightarrow [0, 1]$ is a subprobability measure (i.e., σ-additive, set function, $f(x, \emptyset) = 0$ and $f(x, Y) \leq 1$). The identity morphism $1_X : (X, \mathcal{F}_X) \longrightarrow (X, \mathcal{F}_X)$ is $1_X : X \times \mathcal{F}_X \longrightarrow [0, 1]$ and is defined by

$$1_X(x, A) = \delta(x, A) = \begin{cases} 1, & \text{if } x \in A; \\ 0, & \text{if } x \notin A. \end{cases}$$

For A fixed, $\delta(x, A)$ is the characteristic function of A and for x fixed, $\delta(x, A)$ is the Dirac distribution. Finally, composition is defined as follows: given $f : (X, \mathcal{F}_X) \longrightarrow (Y, \mathcal{F}_Y)$ and $g : (Y, \mathcal{F}_Y) \longrightarrow (Z, \mathcal{F}_Z)$, $gf : (X, \mathcal{F}_X) \longrightarrow (Z, \mathcal{F}_Z)$ is given by

$$gf(x, C) = \int_Y g(y, C)\, f(x, dy),$$

where we are using $f(x, \cdot)$ as the measure for the integration and the function being integrated is the measurable function $g(\cdot, C)$.

This category is based on the work of Giry [52] and Lawvere [80]. In fact **SRel** is the Kleisli category of certain functor Π over the category **Mes** of measurable spaces and measurable functions. The category **SRel** was defined by Panangaden [95] with slight modifications on Giry's work. This category first appeared in [1]. **SRel** was proven to be a PAC jointly by Panangaden [95] and the author. It was shown to be a traced symmetric monoidal category directly (without using partially additive structure) in [1]. We will include the proof of **SRel** being a PAC for the sake of completeness, the reader can find more details on **SRel** and the measure theory background in [95].

We will sometimes refer to objects in **SRel** by their underlying sets, so we write X for $(X, \mathcal{F}_X)$.

Proposition 3.2.1 *The category* **SRel** *has countable coproducts.*

Proof. Given a family $\{(X_i, \mathcal{F}_{X_i})\}_{i\in I}$ of objects in **SRel**, the coproduct $(X, \mathcal{F}_X)$ is defined as follows. The set X is the disjoint union of the X_i. The σ-field on X is generated by the measurable sets of each component. Thus, a measurable set in $\mathcal{F}_X$ is of the form $\biguplus_{i\in I} A_i$, where $A_i \in \mathcal{F}_{X_i}$ for all $i \in I$. The injections $in_i : X_i \longrightarrow X$ are $in_i(x, \biguplus_{k\in I} A_k) = \delta(x, A_i)$. Given an object $(Y, \mathcal{F}_Y)$ and a family $f_j : X_j \longrightarrow Y$ with $j \in I$, the mediating morphism $f : X \longrightarrow Y$ is defined by $f((x, i), B) = f_i(x, B)$. We check the required commutativity

$$\begin{aligned}
(f in_j)(x, B) &= \int_X f((x', i), B) in_j(x, d(x', i)) \\
&= \int_X f((x', i), B)\delta(x, d(x', j)) \\
&= \int_{X_j} f_j(x', B)\delta(x, dx'), \text{ the integrals over } X_k \text{ for } k \neq j \text{ are } 0 \\
&= f_j(x, B).
\end{aligned}$$

Suppose $g : X \longrightarrow Y$ is another morphism such that $g in_j = f_j$ for all $j \in I$. Then,

$$\begin{aligned}
f((x, j), B) &= f_j(x, B) \\
&= g in_j(x, B) \\
&= \int_X g((x', i), B) in_j(x, d(x', i)) \\
&= \int_X g((x', i), B)\delta(x, d(x', j)) \\
&= \int_{X_j} g_j(x', B)\delta(x, dx') \\
&= g_j(x, B) \\
&= g((x, j), B)
\end{aligned}$$

for all $x \in X$ and $B \in \mathcal{F}_Y$. Thus, $g = f$. □

A family $\{f_i\}_{i\in I}$ in $\mathbf{SRel}(X, Y)$ is summable if

$$\textstyle\sum_i f_i(x, Y) \leq 1 \text{ for all } x \in X.$$

In that case, $(\sum_i f_i)(x, B) = \sum_i f_i(x, B)$ where the latter is the usual sum of real numbers. Partition-associativity follows from the fact that all numbers are nonnegative and hence we are dealing with absolute convergence. The unary sum axiom is trivially true. We need to verify the limit axiom.

Let $\{f_i\}_I$ be a summable family in $\mathbf{SRel}(X, Y)$ and suppose that all finite subfamilies are summable. The sums $\sum_{i=0}^{n} f_i(x, Y)$ are bounded by 1 for all $x \in X$. Now $\sum_{i=0}^{\infty} f_i(x, Y)$ is the limit of the net of finite partial sums all bounded by 1, hence $\sum_{i=0}^{\infty} f_i(x, Y)$ is bounded by 1 and $\sum_{i=0}^{\infty} f_i(x, Y)$ exists. The distributivity of composition over sum follows from the monotone convergence theorem [95].

We next define the quasi projections: $\rho_j : \bigoplus_I X_i \longrightarrow X_j$ is defined by $\rho_j((x, k), A_j) = \delta(x, A_j)$ if $k = j$ and 0 otherwise. The zero object is $(\emptyset, \mathcal{F}_\emptyset)$, because given $(X, \mathcal{F}_X)$, $f : (\emptyset, \mathcal{F}_\emptyset) \longrightarrow (X, \mathcal{F}_X)$ is the unique empty function and $g : (X, \mathcal{F}_X) \longrightarrow (\emptyset, \mathcal{F}_\emptyset)$ is the unique function such that $g(x, \emptyset) = 0$ for all $x \in X$. The zero morphism $0_{XY} : (X, \mathcal{F}_X) \longrightarrow (Y, \mathcal{F}_Y)$ is defined by $0_{XY}(x, B) = 0$ for all $x \in X$ and $B \in \mathcal{F}_Y$.

We need to verify the compatible sum and untying axioms. Let $\{f_i\}_I$ be a compatible family in $\mathbf{SRel}(X, Y)$,

$$\begin{aligned}
\textstyle\sum_{i \in I} f_i(x, Y) &= \textstyle\sum_{i \in I} \rho_i f(x, Y) \\
&= \textstyle\sum_{i \in I} \displaystyle\int_{I.Y} \rho_i((y, k), Y) f(x, d(y, k)) \\
&= \textstyle\sum_{i \in I} \displaystyle\int_{Y} \delta(y, Y) f(x, dy), \text{ integral is over the } i\text{th summand} \\
&= \textstyle\sum_{i \in I} f(x, in_i(Y)) \\
&= f(x, I.Y), \ \sigma\text{-additivity of the measure } f(x, .).
\end{aligned}$$

Here $I.Y$ denotes the coproduct of $|I|$ copies of Y. Now as $f(x, I.Y) \leq 1$, we conclude that the family $\{f_i\}_I$ is summable.

For the untying axiom, suppose $f, g : X \longrightarrow Y$ are summable,

$$\begin{aligned}
(in_1 f + in_2 g)(x, Y \uplus Y) &= in_1 f(x, Y \uplus Y) + in_2 g(x, Y \uplus Y) \\
&= \int_Y in_1(y, Y \uplus Y) f(x, dy) + \int_Y in_2(y, Y \uplus Y) g(x, dy) \\
&= \int_Y \delta(y, Y) f(x, dy) + \int_Y \delta(y, Y) g(x, dy) \\
&= f(x, Y) + g(x, Y) \\
&= (f + g)(x, Y) \leq 1, \text{ since } f \text{ and } g \text{ are summable.}
\end{aligned}$$

Therefore, $in_1 f$ and $in_2 g$ are summable.

Thus we conclude that

Proposition 3.2.2 *The category* **SRel** *of measurable spaces and stochastic kernels is a PAC.*

Chapter 4

Unique Decomposition Categories

In the previous chapter, we introduced partially additive categories and discussed several facts about such categories. In this chapter we generalise PACs into categories that we call *unique decomposition categories* (UDCs). Homsets in UDCs are also enriched with a partial infinitary operation Σ.

Some of the ideas used in the definition of UDCs can be traced back to early works of Kuroš [75]. Unique decomposition categories carry the necessary algebraic structure to capture the main example of Geometry of Interaction, namely the category of sets and partial injective functions, which is not a partially additive category (see Example 4.0.15.)

As was pointed out in the previous chapter, Σ-monoids are the same as partially additive monoids save for the *limit axiom.* Here is the formal definition.

Definition 4.0.3 A Σ-*monoid* consists of a pair (M, Σ) where M is a nonempty set and Σ is a partial operation on the countable families in M (we say that $\{x_i\}_{i\in I}$ is summable if $\sum_{i\in I} x_i$ is defined), subject to the following axioms:

1. *Partition-Associativity Axiom.* If $\{x_i\}_{i\in I}$ is a countable family and if $\{I_j\}_{j\in J}$ is a (countable) partition of I , then $\{x_i\}_{i\in I}$ is summable if and only if $\{x_i\}_{i\in I_j}$ is summable for every $j \in J$ and $\sum_{i\in I_j} x_i$ is summable for $j \in J$. In that case,

$$\textstyle\sum_{i\in I} x_i = \sum_{j\in J}(\sum_{i\in I_j} x_i)$$

2. *Unary Sum Axiom.* Any family $\{x_i\}_{i\in I}$ in which I is a singleton is summable and $\sum_{i\in I} x_i = x_j$ if $I = \{j\}$.

Similar axiomatisations can be found in Bourbaki ([29], Chapter 3, Section 5) where infinite sums for commutative Hausdorff topological groups are defined. Kuroš [75] also gives an axiomatisation of infinite sums partially defined over sets, specifically homsets of categories. Axiom II in [75] is the same as our first axiom. Higgs [61] uses similar axioms, however our first axiom corresponds to his second and third axioms and is stronger. We first encountered such structures in the work of Manes and Benson [91] where they are called *positive monoids*.

All the properties that we discussed for partially additive monoids in the previous chapter are also true for Σ-monoids.

We have already seen that Σ-monoids form a symmetric monoidal category (with product as the tensor), called $\Sigma\mathbf{Mon}$.

A $\Sigma\mathbf{Mon}$-category $\mathbb{C}$ is a category enriched in $\Sigma\mathbf{Mon}$, that is the homsets are enriched with an additive structure such that composition distributes over addition from left and right. Note that such categories have non-empty homsets and automatically have zero morphisms, namely $0_{XY} : X \longrightarrow Y = \sum_{i\in\emptyset} f_i$ for $f_i \in \mathbb{C}(X,Y)$. Note that this does not imply the existence of the zero object; however existence of the initial or terminal object will guarantee the existence of a zero object in the presence of zero morphisms.

Clearly, every partially additive monoid is a Σ-monoid. However, the converse is not true. Here are some examples:

Example 4.0.4

1. Let M be a poset satisfying countable bounded completeness, that is given a countable subset D of M, if D has an upper bound then it has a least upper bound. Let the summable families be the bounded families and Σ be the supremum. Then, (M,Σ) is a Σ-monoid. The additive identity is of course $sup(\emptyset)$. Note that the limit axiom is not true in general: for example, consider the family $x_1 \leq x_2 \leq \cdots$, an unbounded ascending chain. Clearly all finite subfamilies

are bounded and hence summable, whereas the family itself is not. Therefore, (M, Σ) is not a partially additive monoid.

2. Consider the set $\mathbb{R}^+$ of all non-negative real numbers and define $\sum_{i \in I} x_i$ to be the arithmetic sum for I finite and the limit of the net of finite partial sums for I infinite. The summable families are absolutely convergent series. This is another example of a Σ-monoid which is not a partially additive monoid, because the limit axiom fails: just consider the family $\{x_i\}_{i \in I}$ with I countable infinite and $x_i = 1$ for all $i \in I$.

Definition 4.0.5 A *unique decomposition category* (UDC) $\mathbb{C}$ is a symmetric monoidal Σ**Mon**-category which satisfies the following axiom[1]:
(A) For all $j \in I$ there are morphisms *quasi injection*: $\iota_j : X_j \longrightarrow \otimes_I X_i$, and *quasi projection*: $\rho_j : \otimes_I X_i \longrightarrow X_j$, such that

1. $\rho_k \iota_j = 1_{X_j}$ if $j = k$ and $0_{X_j X_k}$ otherwise.
2. $\sum_{i \in I} \iota_i \rho_i = 1_{\otimes_I X_i}$.

Proposition 4.0.6 (Matricial Representation) *Given $f : \otimes_J X_j \longrightarrow \otimes_I Y_i$ in a UDC with $|I| = m$ and $|J| = n$, there exists a unique family $\{f_{ij}\}_{i \in I, j \in J} : X_j \longrightarrow Y_i$ with $f = \sum_{i \in I, j \in J} \iota_i f_{ij} \rho_j$, namely, $f_{ij} = \rho_i f \iota_j$.*

Proof. $f = 1_{\otimes_I Y_i} f 1_{\otimes_J X_j} = (\sum_i \iota_i \rho_i) f (\sum_j \iota_j \rho_j) = (\sum_i \iota_i \rho_i f)(\sum_j \iota_j \rho_j) = \sum_{ij} \iota_i \rho_i f \iota_j \rho_j = \sum_{ij} \iota_i f_{ij} \rho_j$. For uniqueness, suppose there is another family $\{g_{kl}\}_{k \in I, l \in J}$ such that $f = \sum_{kl} \iota_k g_{kl} \rho_l$. Then, $f_{ij} = \rho_i f \iota_j = \sum_{kl} \rho_i \iota_k g_{kl} \rho_l \iota_j = g_{ij}$ for all $i \in I$ and $j \in J$. $\square$

Based on this proposition, every morphism $f : \otimes_J X_j \longrightarrow \otimes_I Y_i$ can be represented by its components. We will use the corresponding matrices to represent morphisms, for example f above (with $|I| = m$ and $|J| = n$) is represented by an $m \times n$ matrix.

$$f = \begin{bmatrix} f_{11} & \cdots & f_{1n} \\ \vdots & \vdots & \vdots \\ f_{m1} & \cdots & f_{mn} \end{bmatrix}$$

[1] We use the following notational conventions: $I.X$ for $X^{\otimes I}$, $\otimes_I X_i$ for $\otimes_{i \in I} X_i$ and $\{X_i\}_I$ for $\{X_i\}_{i \in I}$. All index sets are finite when we discuss UDCs, unless explicitly stated otherwise.

Given $f : \otimes_K X_k \longrightarrow \otimes_J Y_j$ and $g : \otimes_J Y_j \longrightarrow \otimes_I Z_i$, let $h = gf$, then $h_{ik} = \rho_i h \iota_k = \rho_i (gf)\iota_k = \rho_i g(\sum_{j\in J} \iota_j \rho_j) f \iota_k = \sum_{j\in J}(\rho_i g \iota_j \rho_j) f \iota_k = \sum_{j\in J}(g_{ij}\rho_j f \iota_k) = \sum_{j\in J} g_{ij} f_{jk}$.

Therefore, the composition of morphisms in a UDC corresponds to matrix multiplication of their matricial representations.

Remark. Note that although any morphism $f : \otimes_J X_j \longrightarrow \otimes_I Y_i$ can be represented by the unique family $\{f_{ij}\}$ of its components, the converse is not necessarily true, that is to say given a family $\{f_{ij}\}$ with I, J finite there may not be a morphism $f : \otimes_J X_j \longrightarrow \otimes_I Y_i$ satisfying $f = \sum_{ij} \iota_i f_{ij} \rho_j$. However, in case such an f exists it will be unique.

Definition 4.0.7 A functor $T : \mathbb{C} \longrightarrow \mathbb{D}$ between two UDCs is said to be *additive* if $T_{X,Y} : \mathbb{C}(X,Y) \longrightarrow \mathbb{D}(TX,TY)$ is a Σ**Mon** morphism for all objects X, Y in $\mathbb{C}$. That is, for every summable family $\{f_i\}_I$ in $\mathbb{C}(X,Y)$ with countable I, $\{T_{X,Y}(f_i)\}_I$ is summable in $\mathbb{D}(TX,TY)$ and $T_{X,Y}(\sum_i f_i) = \sum_i T_{X,Y}(f_i)$.

Theorem 4.0.8 *Let $\mathbb{C}$ be a category with countable biproducts, then $\mathbb{C}$ is a UDC (with biproduct as tensor) in which all families of morphisms are summable.*

Proof. Let $\mathbb{C}$ be a category with countable biproducts. $\mathbb{C}$ has zero object and hence families of zero morphisms. Given a family $\{f_i\}_I$ of morphisms in $\mathbb{C}(X,Y)$ with countable I, define $\sum_{i\in I} f_i = \nabla_Y(\tilde{\bigoplus}_{i\in I} f_i)\Delta_X$ where $\Delta_X : X \longrightarrow I.X$ and $\nabla_Y : I.Y \longrightarrow Y$ are defined by: $\pi_i \Delta_X = 1_X$ and $\nabla_Y in_i = 1_Y$ for all $i \in I$ and $\sum_{i\in\emptyset} f_i = 0_{XY}$. Here $I.X$ and $I.Y$ denote biproduct of $|I|$ copies of X and Y respectively. This makes every homset $\mathbb{C}(X,Y)$ into a commutative monoid with $\sum_{i\in\emptyset} f_i$ as the additive identity. More generally, using the associativity and symmetry isomorphisms of biproduct it follows that every homset $\mathbb{C}(X,Y)$ is a Σ-monoid with all families summable. Also, composition distributes over addition.

We need to verify axiom A of Definition 4.0.5. Given any family $\{X_i\}_I$ in $\mathbb{C}$, let $\iota_j = in_j$ and $\rho_j = \pi_j$ for all $j \in I$. Here in_j and π_j are the biproduct morphisms of $(\tilde{\bigoplus}_I X_i, \{\pi_j\}_I, \{in_j\}_I)$. By definition of biproduct we have $\pi_k in_j = 1_{X_j}$ if $j = k$ and $0_{X_j X_k}$ else. We also have $\pi_j(\sum_{i\in I} in_i \pi_i) = \pi_j$ for all $j \in I$ and therefore $\sum_{i\in I} in_i \pi_i = 1_{\tilde{\bigoplus}_I X_i}$. □

Proposition 4.0.9 *Let $\mathbb{C}$ be a UDC. If all finite families of morphisms in $\mathbb{C}$ are summable then $\mathbb{C}$ has finite biproducts.*

Proof. Let $\mathbb{C}$ be a UDC with all finite families summable. We show that given a family $\{X_i\}_I$, $(\otimes_I X_i, \{\iota_j\}_I)$ is a coproduct. Given any object Y and a family $\{f_j\}_I$ with $f_j : X_j \longrightarrow Y$ let $f = \sum_{j \in I} f_j \rho_j$, f exists since all finite families are summable. Also, $f\iota_j = (\sum_j f_j \rho_j)\iota_j = \sum_j (f_j \rho_j \iota_j) = f_j$ for all $j \in I$. Let $h : \otimes_I X_i \longrightarrow Y$ be such that $h\iota_j = f_j$ for all $j \in I$, there exists a unique family $\{h_j\}_{j\in I}$ with $h_j : X_j \longrightarrow Y$ such that $h = \sum_j h_j \rho_j$. $f_j = h\iota_j = h_j$ for all $j \in I$. Thus $f = h$ and there exists a unique f such that $f\iota_j = f_j$ for all $j \in I$.

Similarly, we show that given a family $\{X_i\}_I$, $(\otimes_I X_i, \{\rho_j\}_I)$ is a product. Given any object Y and a family $\{f_j\}_I$ with $f_j : Y \longrightarrow X_j$ let $f = \sum_{j\in I} \iota_j f_j$, f exists since all finite families are summable. Also, $\rho_j f = \rho_j(\sum_j \iota_j f_j) = \sum_j (\rho_j \iota_j f_j) = f_j$ for all $j \in I$. Let $h : Y \longrightarrow \otimes_I X_i$ be such that $\rho_j h = f_j$ for all $j \in I$, there exists a unique family $\{h_j\}_{j\in I}$ with $h_j : Y \longrightarrow X_j$ such that $h = \sum_j \iota_j h_j$. $f_j = \rho_j h = h_j$ for all $j \in I$. Thus $f = h$ and there exists a unique f such that $\rho_j f = f_j$ for all $j \in I$.

Also $\rho_k \iota_j = 1_{X_j}$ if $j = k$ and $0_{X_j X_k}$ otherwise and $\sum_{i\in I} \iota_i \rho_i = 1_{\otimes_I X_i}$, since $\mathbb{C}$ is a UDC. Therefore, $(\otimes_I X_i, \{\rho_j\}_I, \{\iota_j\}_I)$ is a biproduct. □

Lemma 4.0.10 *Let $\mathbb{C}$ be a UDC and $f : X \otimes U \longrightarrow X \otimes U$ be given by* $\begin{bmatrix} A & B \\ C & D \end{bmatrix}$, *then $\sum_{n\in\omega} f^n$, if it exists, is given by* $\begin{bmatrix} E & F \\ G & H \end{bmatrix}$, *where:*

- $E = \sum_{m\in\omega}(A + \sum_{n\in\omega} BD^n C)^m$
- $F = \sum_{l\in\omega} EBD^l$
- $G = \sum_{l\in\omega} D^l CE$
- $H = \sum_{l\in\omega} D^l + \sum_{p,q\in\omega} D^p CEBD^q$

Proof. Let f be as in the statement of the lemma, then we can represent f as a directed graph where the vertices are labelled by X and U and the edges are labelled by the components of the matricial representation of f, see the figure below.

With this representation, f^n corresponds to the matrix that represents the paths of length n between the vertices in the graph and $\sum_{n\in\omega} f^n$ is the sum over all paths of length n for all $n \in \omega$.

The components of $\sum_{n\in\omega} f^n$ can be obtained by simply following the edges on the graph in the figure below.

- $X \longrightarrow X, \quad E = \sum_{m\in\omega}(A + \sum_{n\in\omega} BD^nC)^m$

- $X \longrightarrow U, \quad G = \sum_{l\in\omega} D^lCE$

- $U \longrightarrow X, \quad F = \sum_{l\in\omega} EBD^l$

- $U \longrightarrow U, \quad H = \sum_{l\in\omega} D^l + \sum_{p,q\in\omega} D^pCEBD^q$

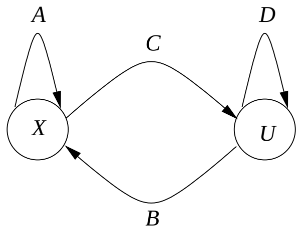


Figure 8: Graphical Representation of f

$\square$

Proposition 4.0.11 (Execution/Trace Formula) *Let $\mathbb{C}$ be a unique decomposition category such that for every X, Y, and U and $f : X \otimes U \longrightarrow Y \otimes U$, the sum $f_{11} + \sum_{n=0}^{\infty} f_{12}f_{22}^n f_{21}$ exists, where f_{ij} are the components of f. Then, $\mathbb{C}$ is traced and*

$$Tr^U_{X,Y}(f) = f_{11} + \sum_{n=0}^{\infty} f_{12}f_{22}^n f_{21}.$$

Before proceeding to the proof of this proposition, let's look at some examples of trace computation. Let $\mathbb{C}$ be a traced UDC and hence given any $f : X \otimes U \longrightarrow Y \otimes U$, $Tr^U_{X,Y}(f)$ exists. Consider the following examples:

1. Let $f : X \otimes U \longrightarrow Y \otimes U$ be given by $\begin{bmatrix} g & 0 \\ h & 0 \end{bmatrix}$. Then

$$Tr^U_{X,Y}(f) = Tr^U_{X,Y}\left(\begin{bmatrix} g & 0 \\ h & 0 \end{bmatrix}\right) = g + \textstyle\sum_n 00^n h = g + 0h = g + 0 = g.$$

2. Let $f : X \otimes U \longrightarrow Y \otimes U$ be given by $\begin{bmatrix} g & 0 \\ 0 & h \end{bmatrix}$. Then

$$Tr^U_{X,Y}(f) = Tr^U_{X,Y}\left(\begin{bmatrix} g & 0 \\ 0 & h \end{bmatrix}\right) = g + \textstyle\sum_n 0h^n 0 = g + 0 = g.$$

3. Let $f : X \otimes U \longrightarrow Y \otimes U$ be given by $\begin{bmatrix} 0 & g \\ h & 0 \end{bmatrix}$. Then

$$Tr^U_{X,Y}(f) = Tr^U_{X,Y}\left(\begin{bmatrix} 0 & g \\ h & 0 \end{bmatrix}\right) = 0 + \textstyle\sum_n g0^n h = 0 + g1h = gh.$$

Proof.

1. Naturality in X:

$Tr^U_{X,Y}(f)g = Tr^U_{X',Y}(f(g \otimes 1_U))$ where $f : X \otimes U \longrightarrow Y \otimes U, g : X' \longrightarrow X$. We rewrite this in matricial form, so we have $f(g \otimes 1_U) = \begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix} \begin{bmatrix} g & 0 \\ 0 & 1 \end{bmatrix}$.

Therefore, $Tr^U_{X',Y}(f(g \otimes 1_U)) = f_{11}g + \sum_n f_{12}(f_{22})^n(f_{21}g)$ which is the same as $Tr^U_{X,Y}(f)g$.

2. Naturality in Y :

$gTr^U_{X,Y}(f) = Tr^U_{X,Y'}((g \otimes 1_U)f)$ where $f : X \otimes U \longrightarrow Y \otimes U, g : Y \longrightarrow Y'$.

$(g \otimes 1_U)f = \begin{bmatrix} g & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix}$.

Therefore, $Tr^U_{X,Y'}((g \otimes 1_U)f) = gf_{11} + \sum_n (gf_{12})(f_{22})^n f_{21}$ which is the same as $gTr^U_{X,Y}(f)$.

3. Dinaturality in U:

$Tr^U_{X,Y}((1_Y \otimes g)f) = Tr^{U'}_{X,Y}(f(1_X \otimes g))$ where $f : X \otimes U \longrightarrow Y \otimes U'$ and $g : U' \longrightarrow U$.

$(1_Y \otimes g)f = \begin{bmatrix} f_{11} & f_{12} \\ gf_{21} & gf_{22} \end{bmatrix}$ and $f(1_X \otimes g) = \begin{bmatrix} f_{11} & f_{12}g \\ f_{21} & f_{22}g \end{bmatrix}$.

Therefore, $Tr^U_{X,Y}((1_Y \otimes g)f) = f_{11} + \sum_n f_{12}(gf_{22})^n(gf_{21}) = f_{11} + \sum_n (f_{12}g)(f_{22}g)^n f_{21} = Tr^{U'}_{X,Y}(f(1_X \otimes g))$.

4.1. Vanishing I:

$Tr^I_{X,Y}(f) = f$ where $f : X \otimes I \longrightarrow Y \otimes I$.

$Tr^I_{X,Y}(f) = Tr^I_{X,Y}\left(\begin{bmatrix} f & 0_{IY} \\ 0_{XI} & 1_I \end{bmatrix}\right) = f + \sum_n 0_{IY} 1_I^n 0_{XI} = f$.

4.2 Vanishing II:

$Tr^{U \otimes V}_{X,Y}(f) = Tr^U_{X,Y}(Tr^V_{X \otimes U, Y \otimes U}(f))$ where $f : X \otimes U \otimes V \longrightarrow Y \otimes U \otimes V$. Let f be given by $\begin{bmatrix} L & M & N \\ P & A & B \\ Q & C & D \end{bmatrix}$.

Then $Tr^U Tr^V\left(\begin{bmatrix} L & M & N \\ P & A & B \\ Q & C & D \end{bmatrix}\right) = Tr^U\left(\begin{bmatrix} L & M \\ P & A \end{bmatrix} + \sum_n \begin{bmatrix} N \\ B \end{bmatrix} D^n \begin{bmatrix} Q & C \end{bmatrix}\right)$

$= Tr^U\left(\begin{bmatrix} L + \sum_n ND^nQ & M + \sum_n ND^nC \\ P + \sum_n BD^nQ & A + \sum_n BD^nC \end{bmatrix}\right)$

$= L + \sum_n ND^nQ + (M + \sum_n ND^nC)(\sum_m (A + \sum_n BD^nC)^m)(P + \sum_n BD^nQ)$

$= L + \sum_n ND^nQ + (M + \sum_n ND^nC)E(P + \sum_n BD^nQ)$ for E as in Lemma 4.0.10

$= L + \sum_n ND^nQ + MEP + ME\sum_n BD^nQ + \sum_n ND^nCEP + \sum_m ND^mCE\sum_n BD^nQ$

$= L + MEP + \sum_n MEBD^nQ + \sum_n ND^nCEP + N(\sum_n D^n + \sum_{m,n} D^mCEBD^n)Q$

$= L + MEP + MFQ + NGP + NHQ$ for E, F, G and H as in Lemma 4.0.10

$$= L + \begin{bmatrix} M & N \end{bmatrix} \begin{bmatrix} E & F \\ G & H \end{bmatrix} \begin{bmatrix} P \\ Q \end{bmatrix}$$

$$= L + \sum_n \begin{bmatrix} M & N \end{bmatrix} \begin{bmatrix} A & B \\ C & D \end{bmatrix}^n \begin{bmatrix} P \\ Q \end{bmatrix} = Tr^{U\otimes V}\left(\begin{bmatrix} L & M & N \\ P & A & B \\ Q & C & D \end{bmatrix}\right) = Tr^{U\otimes V}_{X,Y}(f).$$

5. Superposing :

$g \otimes Tr^U_{X,Y}(f) = Tr^U_{W\otimes X, Z\otimes Y}(g \otimes f)$ where $f : X \otimes U \longrightarrow Y \otimes U$ and $g : W \longrightarrow Z$.

$$Tr^U_{W\otimes X,Z\otimes Y}(g\otimes f) = Tr^U_{W\otimes X,Z\otimes Y}\left(\begin{bmatrix} g & 0 & 0 \\ 0 & f_{11} & f_{12} \\ 0 & f_{21} & f_{22} \end{bmatrix}\right) = \begin{bmatrix} g & 0 \\ 0 & f_{11} \end{bmatrix} + \sum_n \begin{bmatrix} 0 \\ f_{12} \end{bmatrix} f_{22}^n$$

$$\begin{bmatrix} 0 & f_{21} \end{bmatrix} = \begin{bmatrix} g & 0 \\ 0 & f_{11} + \sum_n f_{12} f_{22}^n f_{21} \end{bmatrix} = \begin{bmatrix} g & 0 \\ 0 & Tr^U_{X,Y}(f) \end{bmatrix} = g \otimes Tr^U_{X,Y}(f).$$

6. Yanking:

$$Tr^U_{U,U}(\sigma_{U,U}) = Tr^U_{U,U}\left(\begin{bmatrix} 0 & 1_U \\ 1_U & 0 \end{bmatrix}\right) = 1_U.$$ □

Proposition 4.0.12 *The trace operator in a traced UDC is uniform.*

Proof. Let $f : X \otimes U \longrightarrow Y \otimes U$ and $g : X \otimes U' \longrightarrow Y \otimes U'$ be two morphisms in $\mathbb{C}$. We have $(1_Y \otimes h)f = g(1_X \otimes h)$. Thus, $\begin{bmatrix} 1 & 0 \\ 0 & h \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} \\ f_{21} & f_{22} \end{bmatrix} = \begin{bmatrix} g_{11} & g_{12} \\ g_{21} & g_{22} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & h \end{bmatrix}$

Therefore, we have the following equalities:

$$f_{11} = g_{11}, \quad f_{12} = g_{12}h, \quad hf_{21} = g_{21}, \quad hf_{22} = g_{22}h.$$

Now

$$
\begin{aligned}
Tr^U_{X,Y}(f) &= f_{11} + \textstyle\sum_n f_{12} f_{22}^n f_{21} \\
&= g_{11} + \textstyle\sum_n (g_{12}h) f_{22}^n f_{21}, \text{ using the identities above,} \\
&= g_{11} + g_{12}hf_{21} + g_{12}hf_{22}f_{21} + g_{12}hf_{22}^2 f_{21} + \cdots \\
&= g_{11} + g_{12}g_{21} + g_{12}g_{22}hf_{21} + g_{12}g_{22}^2 hf_{21} + \cdots, \text{ using the identities above,} \\
&= g_{11} + g_{12}g_{21} + g_{12}g_{22}g_{21} + g_{12}g_{22}^2 g_{21} + \cdots, \text{ using the identities above,} \\
&= g_{11} + \textstyle\sum_n g_{12} g_{22}^n g_{21} \\
&= Tr^{U'}_{X,Y}(g).
\end{aligned}
$$

□

Proposition 4.0.13 *Every PAC is a traced UDC with coproduct as tensor.*

Proof. Let $\mathbb{C}$ be a PAC. Let the tensor product be coproduct and given a family $\{X_i\}_I$ in $\mathbb{C}$ choose $\iota_j = in_j$ and ρ_j as defined in Definition 3.0.10. Clearly $\mathbb{C}$ is a Σ**Mon** enriched category since every partially additive monoid is already a Σ-monoid. To show that $\mathbb{C}$ is a UDC we need to verify axiom A of Definition 4.0.5. Observe that $\rho_k in_j = 1_{X_j}$ for $j = k$ and 0 else by definition of ρ_k and $\sum_{i\in I} in_i\rho_i = 1_{\bigoplus_I X_i}$ by Corollary 3.0.13. Therefore, axiom A holds. Moreover, in every PAC the sum $f_{11} + \sum_n f_{12}f_{22}^n f_{21}$ exists for any $f : X \oplus U \longrightarrow Y \oplus U$, where f_{ij} are the components of f. Hence by Proposition 4.0.11 $\mathbb{C}$ is a traced UDC. □

The condition for the existence of a trace in a UDC can be significantly simplified in the presence of the limit axiom.

Proposition 4.0.14 *Let $\mathbb{C}$ be a UDC in which every homset is a partially additive monoid. If for every X, Y, U and $f : X \otimes U \longrightarrow Y \otimes U$, the sum $f_{12} + gf_{22}$ exists for every $g : U \longrightarrow Y$ and the sum $f_{11} + h$ exists, where $h = \sum_{n=0}^{\infty} f_{12}f_{22}^n f_{21}$, then $\mathbb{C}$ is traced with $Tr(f) = f_{11} + \sum_{n=0}^{\infty} f_{12}f_{22}^n f_{21}$. Here f_{11}, f_{12}, f_{21} and f_{22} are the components of f.*

Proof. Note that f_{12} is summable by the unary sum axiom. Given that $f_{12} + gf_{22}$ exists for all $g : U \longrightarrow Y$, we let $g = f_{12}$ and hence $f_{12} + f_{12}f_{22}$ exists. Next let $g = f_{12} + f_{12}f_{22}$ and hence $f_{12} + f_{12}f_{22} + f_{12}f_{22}^2$ exists. Proceeding in this way we conclude that the sum $\sum_{k=0}^{n} f_{12}f_{22}^k$ exists for every n and hence, by the limit axiom $\sum_{n=0}^{\infty} f_{12}f_{22}^n$ exists. Note that $h = \sum_n f_{12}f_{22}^n f_{21} = (\sum_n f_{12}f_{22}^n)f_{21}$ exists since composition distributes over sum.

Finally, $Tr_{X,Y}^U(f) = f_{11}+\sum_n f_{12}f_{22}^n f_{21} = f_{11}+h$ exists by hypothesis and therefore $\mathbb{C}$ is traced by Proposition 4.0.11. □

As we saw above every PAC is a UDC and hence the PAC examples in the previous chapter are also UDC examples. Below we give more examples of UDCs that are not PACs.

Example 4.0.15

1. Consider the category **PInj** of sets and partial injective functions. Define $X \otimes Y = X \uplus Y$, note that this does not give a coproduct, indeed **PInj** does not have coproducts. The partially additive structure is given in Example 3.0.8 in the previous chapter. Define $\rho_j : \biguplus_{i \in I} X_i \longrightarrow X_j$ by $\rho_j(x,j) = x$, and $\rho_j(x,i)$ undefined for $i \neq j$. $\iota_j : X_j \longrightarrow \biguplus_{i \in I} X_i$ is defined by $\iota_j(x) = (x,j)$. It can be easily seen that $\rho_k \iota_j = 1_{X_j}$ if $j = k$ and $0_{X_j X_k}$ otherwise. Note that $Dom(\iota_i\rho_i) \cap Dom(\iota_j\rho_j) = \emptyset$ and $Codom(\iota_i\rho_i) \cap Codom(\iota_j\rho_j) = \emptyset$ for $i \neq j$ and hence the family $\{\iota_i\rho_i\}_{i \in I}$ is summable. Also, $(\sum_i \iota_i\rho_i)((x,j)) = \iota_j(x) = (x,j) = 1_{\biguplus_I X_i}(x,j)$.

   Given $f : X \uplus U \longrightarrow Y \uplus U$ in **PInj** we show that the sum $f_{11} + \sum_n f_{12}f_{22}^n f_{21}$ exists. That is to say, the family of functions appearing in the sum have pairwise disjoint domains and codomains.

   (i) Suppose $x \in Dom(f_{11}) \cap Dom(f_{12}f_{22}^j f_{21})$ for some $j \in \omega$, then $f_{11}(x) = y$ for some $y \in Y$ and hence $f((x,1)) = (y,1)$. On the other hand, $f_{12}f_{22}^j f_{21}(x) = y'$ for some $y' \in Y$ and hence $f_{21}(x) = u$ for some $u \in U$ and therefore $f((x,1)) = (u,2)$, a contradiction.

Now let $x \in Dom(f_{12}f_{22}^{j}f_{21}) \cap Dom(f_{12}f_{22}^{k}f_{21})$ for $k < j$. Let $f_{21}(x) = u$ and $f_{22}^{k}(u) = u'$, then $u' \in Dom(f_{12}) \cap Dom(f_{22}^{j-k})$. Then $f_{12}(u') = y$ for some $y \in Y$ and since $j - k > 0$, $f_{22}(u') = u''$ for some $u'' \in U$, hence $f((u', 2)) = (y, 1)$ and $f((u', 2)) = (u'', 2)$, a contradiction.

(ii) We next show that the maps have disjoint codomains by proving that the inverse maps have disjoint domains. Consider first $Dom(f_{11}^{-1})$ and $Dom(f_{21}^{-1}f_{22}^{-j}f_{12}^{-1})$ for some $j \in \omega$ and suppose there exist a $y \in Dom(f_{11}^{-1}) \cap Dom(f_{21}^{-1}f_{22}^{-j}f_{12}^{-1})$, then $f_{11}(x) = y$ for some $x \in X$ and therefore $f((x, 1)) = (y, 1)$, also $f_{12}(u) = y$ for some $u \in U$ and hence $f((u, 2)) = (y, 1)$ contradicting the injectivity of f.

Next consider $Dom(f_{21}^{-1}f_{22}^{-j}f_{12}^{-1})$ and $Dom(f_{21}^{-1}f_{22}^{-k}f_{12}^{-1})$ for $k < j$ and let there be a $y \in Dom(f_{21}^{-1}f_{22}^{-j}f_{12}^{-1}) \cap Dom(f_{21}^{-1}f_{22}^{-k}f_{12}^{-1})$, and let $u = f_{22}^{-k}f_{12}^{-1}(y)$,then $u \in Dom(f_{21}^{-1}) \cap Dom(f_{22}^{-(j-k)})$ and since $j - k > 0$, u will also be in $Dom(f_{22}^{-1})$. Then, $f_{21}(x) = u$ and $f_{22}(u') = u$ for some $x \in X$ and $u' \in U$, which implies $f((u', 2)) = (u, 2)$ and $f((x, 1)) = (u, 2)$, contradicting the injectivity of f. Hence, $Tr(f)$ exists.

2. Consider the category $\mathbf{Pfn}_{ov}$ where the objects and morphisms are the same as in **Pfn**, but the additive structure is given by the overlap sum (see Example 3.0.8.) As was noted in the previous chapter $\mathbf{Pfn}_{ov}$ is not a partially additive category, because the untying axiom fails. However, $\mathbf{Pfn}_{ov}$ is a UDC. The tensor product is the coproduct (given by disjoint union). Quasi injections are the injection morphisms and quasi projections are defined as in the case of **PInj**. Clearly, Axiom (A) holds. Moreover, $\mathbf{Pfn}_{ov}$ is a traced UDC. The existence of the sum $Tr(f) = f_{11} + \sum_n f_{12}f_{22}^{n}f_{21}$ for an $f : X \uplus U \longrightarrow Y \uplus U$ follows from part (i) of the proof in the previous example.

3. Consider the category **IRel** of sets and injective relations. Let the tensor be the disjoint union of sets. Notice that this is not a coproduct in **IRel**. The additive structure is defined as follows. A family $\{R_i\}_I$ in $\mathbf{IRel}(X, Y)$ is summable if $Codom(R_i) \cap Codom(R_j) = \emptyset$ for all $i \neq j$. For a summable family $\{R_i\}_I$,

$\sum_{i\in I} R_i = \cup_I R_i$. This makes every homset into a Σ-monoid and composition distributes over sum. For every $i \in I$, $\iota_i : X_i \longrightarrow \biguplus_I X_i$ is the relation $\iota_i = \{(x,(x,i))|x \in X_i\}$ and $\rho_i : \biguplus_I X_i \longrightarrow X_i$ is given by the relation $\rho_i = \{((x,i),x)|x \in X_i\}$. Clearly, $\rho_k\iota_j = 0$ for $j \neq k$ and 1_{X_j} otherwise and $\sum_{i\in I} \iota_i\rho_i = 1_{\biguplus_I X_i}$. Moreover, **IRel** is a traced UDC. The existence of the sum $Tr(R) = R_{11} + \sum_n R_{12}R_{22}^n R_{21}$ for an $R : X \uplus U \longrightarrow Y \uplus U$ follows from part (ii) of the proof in the case of **PInj** above.

4. This example will provide the connection to operator algebraic models. Given a set X let $\ell_2(X)$ be the set of all complex valued functions a on X for which the (unordered) sum $\sum_{x\in X} |a(x)|^2$ is finite. $\ell_2(X)$ is a Hilbert space and its norm is given by $||a|| = (\sum_{x\in X} |a(x)|^2)^{1/2}$ and its inner product is given by $< a, b >= \sum_{x\in X} a(x)\overline{b(x)}$ for $a, b \in \ell_2(X)$ [68].

   Barr observed that this construction can be made into a functor, ℓ_2 [16].[2] There is a contravariant faithful functor ℓ_2 :$\mathbf{PInj}^{op} \longrightarrow$**Hilb** where **Hilb** is the category which has Hilbert spaces as objects and linear contractions (norm ≤ 1) as morphisms. For a set X, $\ell_2(X)$ is defined as above and given $f : X \longrightarrow Y$ in **PInj**, $\ell_2(f) : \ell_2(Y) \longrightarrow \ell_2(X)$ is defined by

$$\ell_2(f)(b)(x) = \begin{cases} b(f(x)), & \text{if } x \in Dom(f); \\ 0, & \text{otherwise.} \end{cases}$$

   Hence, this gives a correspondence between partial injective functions and partial isometries on Hilbert spaces (see also [48, 1].)

PInj(X,Y)	**Hilb**$(\ell_2(Y),\ell_2(X))$
f	$\ell_2(f)$
partial injective function	partial isometry
total	isometry
total and surjective	unitary
$X = Y$ and f is identity on $Dom(f)$	projection

[2] Our presentation here is slightly different from Barr's original one in [16].

It can be shown that $\ell_2(X \times Y) \cong \ell_2(X) \otimes \ell_2(Y)$, where $\otimes$ is the tensor product of Hilbert spaces (see [68], page 142 for a proof.) Also $\ell_2(X \uplus Y) \cong \ell_2(X) \oplus \ell_2(Y)$ where $\oplus$ is the direct sum of Hilbert spaces. To see this, let $a \in \ell_2(X)$ and $b \in \ell_2(Y)$, define $q_{a,b} : X \uplus Y \longrightarrow \mathbb{C}$ by $q_{a,b}(x,1) = a(x)$ and $q_{a,b}(y,2) = b(y)$, clearly $q_{a,b} \in \ell_2(X \uplus Y)$. The linear transformation $U : \ell_2(X \uplus Y) \longrightarrow \ell_2(X) \oplus \ell_2(Y)$ given by $U(q_{a,b}) = (a,b)$ is an isomorphism.

Consider the category $\mathbf{Hilb}_2$ whose objects are of the form $\ell_2(X)$ for a set X and a morphism $u : \ell_2(X) \longrightarrow \ell_2(Y)$ is of the form $\ell_2(f)$ for some partial injective function $f : Y \longrightarrow X$ (recall ℓ_2 is contravariant.) Hence, $\mathbf{Hilb}_2$ is a subcategory of $\mathbf{Hilb}$.

For $\ell_2(X)$ and $\ell_2(Y)$ in $\mathbf{Hilb}_2$, the Hilbert space tensor product $\ell_2(X) \otimes \ell_2(Y)$ yields a tensor product in $\mathbf{Hilb}_2$. This follows from the isomorphism $\ell_2(X) \otimes \ell_2(Y) \cong \ell_2(X \times Y)$ and the fact that $X \times Y$ is a tensor product in $\mathbf{PInj}$.

Similarly for $\ell_2(X)$ and $\ell_2(Y)$ in $\mathbf{Hilb}_2$, the direct sum $\ell_2(X) \oplus \ell_2(Y)$ yields a tensor product in $\mathbf{Hilb}_2$. This follows from the isomorphism $\ell_2(X) \oplus \ell_2(Y) \cong \ell_2(X \uplus Y)$ and the fact that $X \uplus Y$ is a tensor product in $\mathbf{PInj}$. Notice however that, although $\ell_2(X) \oplus \ell_2(Y)$ is the direct sum (biproduct) of the Hilbert spaces $\ell_2(X)$ and $\ell_2(Y)$ in $\mathbf{Hilb}$, it is only a tensor product in $\mathbf{Hilb}_2$, as otherwise this would imply that $X \uplus Y$ is the coproduct in $\mathbf{PInj}$, a contradiction.

The additive structure on $\mathbf{PInj}$ makes $\mathbf{Hilb}_2$ into a UDC as follows. We take $\oplus$ as the tensor product with unit $\ell_2(\emptyset)$. We define a sum for operators in $\mathbf{Hilb}_2(\ell_2(X), \ell_2(Y))$. Given a family $\{\ell_2(f_i)\}_I \in \mathbf{Hilb}_2(\ell_2(X), \ell_2(Y))$ with $\{f_i\}_I \in \mathbf{PInj}(Y, X)$, we say that $\{\ell_2(f_i)\}$ is summable if $\{f_i\}$ is summable in $\mathbf{PInj}$ and in that case $\sum_i \ell_2(f_i) =_{def} \ell_2(\sum_i f_i)$. Clearly, this definition makes ℓ_2 an additive functor. Quasi injections and projections are the ℓ_2 images of quasi projections and injections in $\mathbf{PInj}$. Clearly Axiom (A) holds.

Moreover, $\mathbf{Hilb}_2$ is traced. Given $u : \ell_2(X) \oplus \ell_2(U) \longrightarrow \ell_2(Y) \oplus \ell_2(U)$,

$$Tr(u) = \ell_2(Tr^U_{Y,X}(f))$$

where $u = \ell_2(f)$ with $f : Y \uplus U \longrightarrow X \uplus U$.

Chapter 5

Geometry of Interaction

> *Geometry of Interaction is a new kind of semantics of computation free from the twin drawbacks of reductionism (which leads to static modelisation) and subjectivism (which leads to syntactical abuses, in other terms, bureaucracy.)*
>
> *Jean-Yves Girard*

In this chapter we will briefly review Girard's Geometry of Interaction (GoI) programme and the categorical implementation of it given by Abramsky and Jagadeesan [5]. We will be using the references that we cite and our presentation closely follows the original texts. We will next discuss Abramsky's general GoI Construction that we call Abramsky's Programme. It is this programme that we will develop in all details in the next chapter.

5.1 What is Girard's GoI?

Girard introduced the geometry of interaction programme in late 80's. The first proposal for this programme appeared in [46]. This was followed by a series of papers [47, 45, 48] further investigating different aspects of this programme.

Geometry of interaction is a new kind of semantics. To realise what has been achieved and how it differs from the extant approaches we need to look at more traditional forms of semantics.

Denotational semantics tries to find a model for formulas and proofs of logic. For example, in a categorical model [78], the formulas are denoted by objects and proofs by morphisms in a model category. However, the soundness of the model means that the convertibility of proofs with respect to the cut-elimination process is represented by equations that hold true between morphisms. That is, if Π, Π' are proofs of a sequent $\Gamma \vdash A$ and if we have a reduction $\Pi \succ \Pi'$, (by cut-elimination), then their interpretations $[\![\,-\,]\!]$ in any model category denote equal morphisms, i.e. $[\![\,\Pi\,]\!] = [\![\,\Pi'\,]\!] : [\![\,\Gamma\,]\!] \longrightarrow [\![\,A\,]\!]$. It is in this sense that denotational semantics is *static*. Girard [46] gives a useful analogy with mechanics: the denotational approach to computation in computer science is analogous to statics in mechanics. The fact that the denotational semantics is kept constant during a computation can be compared to the existence of static invariants like mass in classical mechanics. But the core of mechanics is *dynamics*, where other invariants of a dynamical nature, like energy play a prominent role. It is the same with computation: it is the dynamics of computation that must be modeled in a mathematical form.

The closest to the idea of dynamics is *syntax*, which makes all the necessary distinctions of sense and is finite. The standpoint that Girard takes against contenting oneself with syntax (subjectivism) is the fact that it contains irrelevant information. This information is very often of a temporal nature. Thus the idea is to find out what is hidden behind syntax, without going to denotational semantics and the answer Girard proposes is

geometry of interaction.

Hence the motivation for this programme is to find out the geometrical meaning of the *Hauptsatz* [50], i.e. to find out what is hidden behind the syntactical manipulations it involves. For a fuller and much more elaborate account of the ideas above see [46, 48].

The first implementation of this programme was given by Girard [47], based on the C^*-algebra of bounded linear operators on the space ℓ^2 of square summable sequences. More specifically, to each proof-as-program Π of system $\mathbb{F}$ [43], (also known as *polymorphic λ-calculus* [100]), one associates a pair (u, σ) of partial symmetries. Here u represents the proof and is defined inductively following the rules of sequent calculus and σ represents the cuts of the proof, i.e. *dynamics*. The dynamics is introduced as a way of eliminating σ using the *execution formula*:

$$EX(u, \sigma) = (1 - \sigma^2)u(1 - \sigma u)^{-1}(1 - \sigma^2).$$

This formula makes sense (converges) if σu is nilpotent, since σu has unit norm. Thus, the nilpotency of σu corresponds to strong normalisation.

We will give an example below, but before doing so we need to discuss the interpretation of proofs in more detail. The presentation used is the usual sequent calculus, however here one needs to keep track of the cuts already made. Therefore a sequent is of the form $\vdash [\Delta], \Gamma$, where Δ is the list of the cuts that have been made during the proof of $\vdash \Gamma$. A proof Π of $\vdash [\Delta], \Gamma$ is represented by means of an operator u in the C^*-algebra $\mathcal{M}_{2m+n}(\mathbb{B}(\ell^2))$, where m, n are the number of formulas in Δ and Γ respectively. Finally, the matrix $\sigma = (\sigma_{ij})$ is defined by
$\sigma_{2i,2i-1} = \sigma_{2i-1,2i} = 1$ for $i = 1, \ldots, m$ and all other entities are 0. Here is an example:

Consider the proof

$$\frac{\vdash A, A^{\perp} \quad \vdash A^{\perp}, A}{\vdash [A], A^{\perp}, A}$$

where we have applied the cut rule to two instances of the axiom $\vdash A, A^{\perp}$. The interpretation of $\vdash A, A^{\perp}$ is given by the 2 by 2 symmetry matrix (here u_1 and u_2) and u is the interpretation of the cut rule on these sequents. Note that u represents the sequent $\vdash [A], A^{\perp}, A$ and hence it is a 4 by 4 matrix: $m = 1$ and $n = 2$. Note also that the execution formula can be written as

$$EX(u, \sigma) = (1 - \sigma^2)(\textstyle\sum_{n=0}^{\infty} u(\sigma u)^n)(1 - \sigma^2).$$

For this example, we have $u(\sigma u)^2 = \mathbf{0}$, the zero matrix. Finally, cut-elimination applied to our example yields $\vdash A, A^{\perp}$, whose denotation is obtained by removing the first 2(=$2m$) rows and columns of the matrix $EX(u, \sigma)$.

$$u_1 = u_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \qquad u = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},$$

$$\sigma = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \qquad u + u\sigma u = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}, \qquad u(\sigma u)^2 = \mathbf{0}$$

$$EX(u, \sigma) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Girard proved the following soundness theorem for the GoI interpretation of system $\mathbb{F}$ in the C^*-algebraic setting.

Theorem 5.1.1 (Girard [47])

(i) If (u, σ) is the interpretation of a proof Π of a sequent $\vdash [\Delta], \Gamma$ then σu is nilpotent.

(ii) If Γ does not use the symbols "?" or "$\exists$", and Ξ is any cut-free proof of $\vdash \Gamma$ with interpretation v, obtained from Π by using the standard Gentzen reduction steps in any order*, then $EX(u, \sigma) = v$.*

The GoI interpretation was extended to untyped λ-calculus by Danos in his PhD thesis [33]. Danos proved that in the class of pure-nets, i.e. nets designed to represent λ-terms, (σu) is nilpotent *iff* the associated pure-net represents a strongly normalising term. Danos and Regnier further extended the GoI interpretation to define a *path-semantics* for proofs(=programs). The idea is that a proof net is represented by a set of paths and the execution formula is an *invariant* of reduction. We will discuss this view in more detail in Chapter 9.

Hence the

$$\text{nilpotency} \ = \ \text{strong normalisation}$$

correspondence was established for system $\mathbb{F}$ and λ-terms. However, there were some disconcerting issues left: the nilpotency condition is not adapted to recursive programming, neither is it in harmony with non-terminating algorithms. This issue was addressed by Girard in [45]. Consider the central part of the execution formula,

$$CP(u, \sigma) = u(1 - \sigma u)^{-1}$$

which represents the *computation*. Girard realised that the most general solution to the problem of existence of this term without asking for the nilpotency of (σu) is to demand that it exist as an *unbounded* operator (in the sense of a closed densely defined one.) One uses the *weak topology* on $\mathbb{B}(\ell^2)$ which is the topology of pointwise convergence, when ℓ^2 is endowed with its own weak topology. Girard proposed the notion of *weak nilpotency*, i.e. convergence to 0 with respect to the weak topology on $\mathbb{B}(\ell^2)$. He proved that $CP(u, \sigma)$ makes sense as a densely defined unbounded operator exactly when (σu) is weakly nilpotent and showed that fixed-point operators à la ML satisfy this property. Malacaria and Regnier were able to prove the weak nilpotency for the pure λ-calculus in [88]. The proof is a corollary of the main result of that paper stating that no cyclic path in a pure net can be repeated consecutively.

Finally in [48], Girard extended the geometry of interaction to the full case including the additives and constants. He also proved a nilpotency theorem for this semantics and its soundness with respect to a slight modification of familiar sequent calculus in the case of exponential-free conclusions.

In this thesis we will be interested in the multiplicative and exponential fragments of linear logic and hence will not further discuss the latter work. We continue our review by discussing the categorical implementation given for GoI by Abramsky and Jagadeesan [5].

5.2 A Categorical Implementation

Abramsky and Jagadeesan gave the first categorical realisation for GoI in [5]. Their formalisation is based on domain theory and arises from the construction of a categorical model of linear logic. The execution formula is replaced by a least fixed-point operator, essentially a generalisation of Kahn's semantics for feedback in dataflow networks [69, 70].

The construction in [5] uses subcategories of the category **Predom** of ω-complete posets and continuous maps, called $\mathcal{GI}$-categories, to construct a model of linear logic, that is a *-autonomous category with products.

Next, a type-free $\mathcal{GI}$-model is used to give an interpretation of geometry of interaction. A type-free model consists of an object D in a $\mathcal{GI}$-category such that it is a domain equipped with certain retractions. The idea is to use this as a "universal domain", using the retractions to internalise the definitions of the linear proofs, with all types denoted by D. One thus associates to a proof Π of $\vdash [\Delta], \Gamma$ a **Predom** morphism $f : D^{2m+n} \longrightarrow D^{2m+n}$. One defines the *feedback formula*, $FB(f, \sigma) : D^n \longrightarrow D^n$, by

$$FB(f,\sigma)(\vec{x}) = \pi' f(f'(\vec{x}), \vec{x})$$

$$f'(\vec{x}) = Y[\lambda \vec{u}.\pi(\sigma \times 1) f(\vec{u}, \vec{x})]$$

where $\pi(\vec{u}, \vec{x}) = \vec{x}$, $\pi'(\vec{u}, \vec{x}) = \vec{u}$ and Y is the familiar Tarski least fixed-point operator. The permutation $\sigma : D^{2m} \longrightarrow D^{2m}$ defined by $\sigma(x_1, x_2, \cdots, x_{2m-1}, x_{2m}) = (x_2, x_1, \cdots, x_{2m}, x_{2m-1})$ represents the cuts. The feedback formula can be written as: $FB(f, \sigma) = \pi(\bigvee_k f^{(k)})$, where $f^{(k)} : D^n \longrightarrow D^{2m+n}$ is defined inductively by

$$f^{(0)} = \perp$$

$$f^{(k+1)} = (\sigma \times 1) f <\pi' f^{(k)}, 1> .$$

The dynamics of cut-elimination is modeled by the sequence of iterations to the fixed-point, $f^{(0)}, f^{(1)}, \cdots$ and strong normalisation is represented by a *finite convergence* property, $f^{(k)} = f^{(k+1)} = \bigvee_{k \in \omega} f^{(k)}$.

Finally we have the following theorem:

Theorem 5.2.1 (Abramsky & Jagadeesan [5]) *Let Π be a proof of a sequent $\vdash [\Delta], \Gamma$ in second order classical linear logic, with $[\![\Pi]\!] = f$. Then,*

1. *If there are no occurrences of* $⅋$ *either explicitly in* Γ *or in any of the witnessing formulas used in the existential quantifier rule to introduce occurrences of* $\exists$ *in* Γ*-in particular, if neither* $⅋$ *nor* $\exists$ *occurs in* Γ*-then if* Π *reduces to* Π' *by any sequence of contractions, with* $[\![\Pi']\!] = g$*, then* $FB(f, \sigma_f) = FB(g, \sigma_g)$*. In particular, if* Π' *is any cut-free proof obtained from* Π *by cut-elimination, then* $FB(f, \sigma_f) = g$*.*

2. f *has finite convergence property:* $(\exists k \in \omega)[f^{(k)} = f^{(k+1)}]$.

5.3 General GoI Construction

The ideas and techniques used in [5] and the notion of abstract *trace* and Int construction, introduced by Joyal, Street and Verity [67], led to a more abstract formalisation of GoI data in the notion of *GoI Situation* introduced by Abramsky in [2]. GoI Situations give an abstract embodiment of the essential ingredients of GoI for the multiplicative and exponential fragment. Furthermore, in his Siena lecture [2] Abramsky introduced a *General GoI Construction*, that we will call Abramsky's programme. This program was sketched in [2] and worked out on the category **PInj**. In this thesis (Chapter 6), we develop this program in full detail and generality. The idea of this program is that a GoI Situation can be used to construct models of computation, i.e. combinatory algebras which are models of untyped combinatory logic. This construction first yields a structure called a *Linear Combinatory Algebra* (LCA). It is then a short step to a combinatory algebra. This construction can be made in a more general context, namely one can start with a *Weak Linear Category* (WLC), which essentially consists of a symmetric monoidal closed category together with a symmetric monoidal functor and some additional structure. The beauty of working with a GoI Situation at the basis of the program, is that the notion of trace comes to play an important role in the model. It is used to define the binary application operator in the LCA. Below we give a schematic description of Abramsky's programme.

Abramsky's Programme:

$$\begin{tikzcd}
\boxed{\text{GoI Situation}} \arrow[d, "\mathcal{G}"] \\
\boxed{\text{Weak Linear Categories}} \arrow[d, "\mathcal{G}(\mathbb{C})(I,V)"] \\
\boxed{\text{Linear Combinatory Algebra}} \arrow[d, "\text{standardisation}"] \\
\boxed{\text{Combinatory Algebra}} \arrow[d, "\text{quotienting}"] \\
\boxed{\lambda\text{-algebra}}
\end{tikzcd}$$

Chapter 6

Linear Combinatory Algebras

Abramsky in his categorical interpretation of Girard's Geometry of Interaction (GoI) observed that the GoI programme was constructing a Linear Combinatory Algebra (LCA) (to be defined in this chapter). In the previous chapter, we outlined this programme. In this chapter, we develop this programme in full detail and generality. We start with a weak linear category and show how to construct an LCA. A weak linear category comes very close to being a model of linear logic, however we relax some of the conditions and hence we get a weaker structure. We will see that this is exactly what we need to implement the GoI programme in a categorical setting. This construction is quite general and indeed captures the CA construction from a given CCC [77, 9, 78].

We will next consider WLCs that are constructed using a GoI Situation which essentially consists of a traced symmetric monoidal category with some extra structure. It is precisely here that the connection to Girard's execution formula is established and captured by the trace operator in a traced monoidal category. Subsequently, one can construct a CA, a model of untyped combinatory logic, from the given LCA. This corresponds to the translation of intuitionistic logic into LL using $!A \multimap B$ as the translation of the intuitionistic implication $A \Rightarrow B$.

This chapter is organised as follows: we first give a short review of combinatory logic, next we recall the definition of weak linear categories and linear combinatory algebras. Next, we explain the construction of a linear combinatory algebra from a weak linear category. In Section 6.3, we focus on weak linear categories that can be obtained from a GoI Situation.

6.1 Combinatory Algebras

The presentation in this section follows the texts by several authors [13, 77, 62, 56].

Combinatory logic (CL) was invented by Schönfinkel in 1924 and independently by Curry in 1930. The idea was to provide a non-set-theoretical foundation for mathematics based on the abstract notion of function and application, the same goal that motivated Church's invention of λ-calculus in the 30's. In combinatory logic there are no bound variables and hence the complications of λ-calculus are avoided. Combinatory terms are built from a binary operation $\bullet$ (application), infinitely many variables and the constants S, K and I. The terms built solely from constants (no variables) are called *combinators*. Originally Shönfinkel used S and K only and I was defined as $I = (S \bullet K) \bullet K$. The equality relation on combinatory terms is defined as the least congruence generated by the reductions: ($\bullet$ is associated to the left)

$$\begin{aligned} S \bullet x \bullet y \bullet z &\succ (x \bullet z) \bullet (y \bullet z) \\ K \bullet x \bullet y &\succ x \\ I \bullet x &\succ x \end{aligned}$$

The fundamental property of combinatory logic is the so-called *combinatory completeness*: given an arbitrary combinatory term with free variables contained in the set $\{x_1, x_2, \cdots, x_n\}$, denoted by $M(\vec{x})$, there exists a constant term (combinator) F such that:

$$F \bullet \vec{x} = M(\vec{x}).$$

This says that every "polynomial" $M(\vec{x})$ can be written in the form $F \bullet \vec{x}$ for some closed term F. In this sense $\{S, K\}$ forms a basis for combinatory terms. In this thesis we will use a different basis for combinatory terms that we discuss later.

To make the connection with proof theory we will consider typed combinators. There is a well-known connection between simply-typed λ-calculus and the natural deduction formulation of propositional intuitionistic logic called the *Curry-Howard isomorphism* [63, 78]. The Curry-Howard isomorphism establishes a correspondence:

$$\begin{aligned}\text{proof} &\leftrightarrow \lambda\text{-term}\\ \text{formula} &\leftrightarrow \text{type}\\ \text{normalisation} &\leftrightarrow \beta\text{-reduction.}\end{aligned}$$

Similarly, the Curry-Howard isomorphism establishes a connection between typed combinatory logic and *Hilbert-style* (as opposed to natural deduction) formulations of propositional intuitionistic logic.

A proof in a Hilbert-style formulation of logic is a sequence of formulas each of which is either an axiom or follows from previous formulas in the sequence by the application of an inference rule. The formula proved is the last element of the sequence. Here we will consider the implicational fragment of intuitionistic logic. There is only one inference rule, namely *modus ponens*:

$$\frac{\alpha \Rightarrow \beta \quad \alpha}{\beta}\, MP$$

and there are two axiom schemes, which are the principal types of the combinators S and K:

$(S):\ (\alpha \Rightarrow (\beta \Rightarrow \gamma)) \Rightarrow ((\alpha \Rightarrow \beta) \Rightarrow (\alpha \Rightarrow \gamma)),$

$(K):\ \alpha \Rightarrow (\beta \Rightarrow \alpha)$

Intuitively, we think of S and K as functions and their types as determining domain and codomain. For example, $K : \alpha \Rightarrow (\beta \Rightarrow \alpha)$ is the function such that for any $x \in \alpha, y \in \beta$, $(K{\bullet}x){\bullet}y = x$.

Finally we have the following correspondence:

$$\begin{aligned}\text{proof} &\leftrightarrow \text{typed CL-term}\\ \text{modus ponens} &\leftrightarrow \text{application operation}\\ \text{axioms} &\leftrightarrow \text{the constants } K \text{ and } S.\end{aligned}$$

With the Curry-Howard isomorphism at hand we can identify the implicational fragment of several substructural logics, including of course linear logic. We first introduce a different basis (the original one used by Curry) and their principal types. We will write xy for $x{\cdot}y$ and $\cdot$ is associated to the left.

$$\begin{array}{lcll} Ix & \succ & x & \text{identity} \\ Bxyz & \succ & x(yz) & \text{composition or cut} \\ Cxyz & \succ & xzy & \text{exchange or commutator} \\ Wxy & \succ & xyy & \text{duplication} \\ Kxy & \succ & x & \text{discarding} \end{array}$$

The principal types for these combinators are:

$$\begin{array}{lcl} I & : & \alpha \Rightarrow \alpha \\ B & : & (\beta \Rightarrow \gamma) \Rightarrow ((\alpha \Rightarrow \beta) \Rightarrow (\alpha \Rightarrow \gamma)) \\ C & : & (\beta \Rightarrow (\alpha \Rightarrow \gamma)) \Rightarrow (\alpha \Rightarrow (\beta \Rightarrow \gamma)) \\ W & : & (\alpha \Rightarrow (\alpha \Rightarrow \beta)) \Rightarrow (\alpha \Rightarrow \beta) \\ K & : & \alpha \Rightarrow (\beta \Rightarrow \alpha) \end{array}$$

As noted above, the set $\{I, B, C, W, K\}$ forms another basis for combinatory logic terms. Hence, S is expressible in terms of these combinators, indeed

$$S = B(B(BW)C)(BB).$$

A close investigation of the types reveals the correspondence with Gentzen's sequent calculus formulation. In particular, notice that W corresponds to the contraction rule and K corresponds to the weakening rule (see Chapter 2.) Also a remark concerning resource manipulation is in order. Notice that the combinators B,C and I use each of their arguments exactly once. K does not use one of its arguments (y above) and W duplicates one of its arguments (y above). Therefore, it is natural to expect that the combinators B,C and I are *linear* combinators (see the discussion in Chapter 2.)

Indeed, the combinators B, C and I give a Hilbert-style axiomatisation of the $\{\multimap\}$ fragment of intuitionistic linear logic, the combinators B, C and K give a Hilbert-style axiomatisation of the $\{\multimap\}$ fragment of intuitionistic Affine logic and B, C, I, W give a Hilbert-style axiomatisation of implicational fragment of relevance logic. Finally, B, C, K, W give a Hilbert-style axiomatisation of the implicational fragment of intuitionistic logic [11, 105].

We close this section by discussing the model theory of combinatory logic. *Combinatory algebras* (CA) are models of combinatory logic. A combinatory algebra $(M, \cdot, K, S)$ consists of a set M together with a binary operation $\cdot$ on M and two distinguished elements K and S of M such that:

$$(K \cdot x) \cdot y = x \text{ and } ((S \cdot x) \cdot y) \cdot z = (x \cdot z) \cdot (y \cdot z)$$

for all $x, y, z \in M$. Combinatory algebras are never associative, never commutative, never finite and never recursive [13, 56]. Nevertheless, they are interesting and important structures in theoretical computer science. A closely related structure is a λ-algebra: a combinatory algebra which satisfies all provable equations of the λ-calculus. Not all combinatory algebras are λ-algebras. Curry showed that by adding five extra axioms, that we call *Curry axioms*, to CL the resulting theory is equivalent to the λ-calculus (see [13], Chapter 7 and [56], Chapter 4.) Hence one can get a λ-algebra from a combinatory algebra via quotienting by these axioms.

6.2 Weak Linear Categories

In this section we introduce *Weak Linear Categories* (WLC) and *Linear Combinatory Algebras* (LCA) and construct such algebras from WLCs. We also show how to obtain (standard) combinatory algebras from linear ones. The latter corresponds to Girard's translation of intuitionistic logic into linear logic. We next consider a special class of WLCs which are obtained from traced symmetric monoidal categories using the $\mathcal{G}$ construction of Abramsky which we discussed in Chapter 2.

Definition 6.2.1 A *Weak Linear Category (WLC)* $(\mathbb{C}, !)$ consists of the following data:

1. A symmetric monoidal closed category $\mathbb{C}$,

2. A symmetric (not necessarily strong) monoidal functor $! : \mathbb{C} \longrightarrow \mathbb{C}$,

3. The following monoidal *pointwise* natural transformations (pointwise naturality suffices for the construction in Theorem 6.2.5):

   (a) $der :! \Longrightarrow Id$ (dereliction)

   (b) $\delta :! \Longrightarrow !!$ (comultiplication)

   (c) $con :! \Longrightarrow ! \otimes !$ (contraction)

   (d) $weak :! \Longrightarrow \mathcal{K}_I$ (weakening, $\mathcal{K}_I$ is the constant I functor)

Note that we do not require $(!, der, \delta)$ form a comonad, nor that $(!A, weak_A, con_A)$ form a comonoid (cf. Chapter 2, linear categories.)

Definition 6.2.2 A *reflexive* object in a WLC $(\mathbb{C}, !)$ is an object V of $\mathbb{C}$ with the following retracts:

1. $V \multimap V \lhd V$
2. $!V \lhd V$
3. $I \lhd V$

Since cartesian closed categories (CCC) are symmetric monoidal closed categories (SMCC), all the usual domain theoretic constructions of reflexive objects in CCCs [78] also yield reflexive objects in the WLC-sense, as follows:

Proposition 6.2.3 *Let $\mathbb{C}$ be a CCC and V be a reflexive object in $\mathbb{C}$, i.e., $V^V \lhd V$. Then $(\mathbb{C}, Id)$ is a WLC and V is a reflexive object in the WLC-sense.*

Proof. Any CCC is an SMCC. Id is a symmetric monoidal functor from $\mathbb{C}$ to itself and the following are monoidal natural transformations:

1. $der_A = 1_A$

2. $\delta_A = 1_A$

3. $con_A =< 1_A, 1_A >$

4. $weak_A = f : A \longrightarrow T$; the unique map from A to terminal object T.

It can be easily shown that $V^V \lhd V$ implies $T \lhd V$ [9]. Moreover, $V \multimap V = V^V \lhd V$, $!V = Id(V) = V \lhd V$ and $I = T \lhd V$ and hence V is a reflexive object in the WLC-sense. □

Definition 6.2.4 A *Linear Combinatory Algebra* $(\mathcal{A}, \cdot, \mathbf{!})$ consists of the following data:

- An applicative structure $(\mathcal{A}, \cdot)$, i.e. a nonempty set $\mathcal{A}$ and a binary operation $\cdot$ on $\mathcal{A}$.
- A unary operation $\mathbf{!} : \mathcal{A} \longrightarrow \mathcal{A}$
- Distinguished elements $B, C, I, K, W, D, \delta, F$ of $\mathcal{A}$,

satisfying the following identities (we associate $\cdot$ to the left and write xy for $x \cdot y$, $x \cdot \mathbf{!}y = x \cdot (\mathbf{!}y)$, etc.) for all variables x, y, z ranging over A.

1. $Bxyz = x(yz)$ Composition, Cut
2. $Cxyz = (xz)y$ Exchange
3. $Ix = x$ Identity
4. $Kx\mathbf{!}y = x$ Weakening
5. $Wx\mathbf{!}y = x\mathbf{!}y\mathbf{!}y$ Contraction
6. $Dx\mathbf{!}y = xy$ Dereliction
7. $\delta\mathbf{!}x = \mathbf{!!}x$ Comultiplication
8. $F\mathbf{!}x\mathbf{!}y = \mathbf{!}(xy)$ Monoidal Functoriality.

The principal types for the combinators above are given as:

$$
\begin{array}{rcl}
I & : & \alpha \multimap \alpha \\
B & : & (\beta \multimap \gamma) \multimap ((\alpha \multimap \beta) \multimap (\alpha \multimap \gamma)) \\
C & : & (\beta \multimap (\alpha \multimap \gamma)) \multimap (\alpha \multimap (\beta \multimap \gamma)) \\
K & : & \alpha \multimap (!\beta \multimap \alpha) \\
W & : & (!\alpha \multimap (!\alpha \multimap \beta)) \multimap (!\alpha \multimap \beta) \\
\delta & : & !\alpha \multimap !!\alpha \\
D & : & (\alpha \multimap \beta) \multimap (!\alpha \multimap \beta) \\
F & : & !(\alpha \multimap \beta) \multimap (!\alpha \multimap !\beta)
\end{array}
$$

The typed linear combinatory logic corresponds to a Hilbert style axiomatisation of the $\{!, \multimap\}$ fragment of intuitionistic linear logic [2],[11, 105], in the same way that typed combinatory logic gives a Hilbert style axiomatisation of the implicational fragment of intuitionistic logic (see the discussion in Section 6.1.)

Let $(\mathbb{C}, !)$ be a WLC and V be a reflexive object in $\mathbb{C}$ with retracts $V \multimap V \lhd V(r,s)$ and $!V \lhd V(p,q)$ where (r,s) and (p,q) are the retraction morphisms. We define the following operations on the set $\mathbb{C}(I, V)$:

- Given $f, g \in \mathbb{C}(I, V)$, $f \bullet g = ev(sf \otimes g)$
- Given $f \in \mathbb{C}(I, V)$, $\mathbf{!}f = p\,!f\,\varphi_I$ where $\varphi_I : I \longrightarrow !I$ and $! = (!, \varphi, \varphi_I)$.

Theorem 6.2.5 *Let* $(\mathbb{C}, !)$ *be a WLC and* V *be a reflexive object in* $\mathbb{C}$ *with retracts* $V \multimap V \lhd V(r,s)$ *and* $!V \lhd V(p,q)$. *Then* $(\mathbb{C}(I,V), \bullet, \mathbf{!})$ *with* $\bullet$ *and* $\mathbf{!}$ *defined as above is a linear combinatory algebra.*

Proof. Let $\mathbb{C}$ be a SMCC and $! = (!, \varphi, \varphi_I)$, we first recall some properties of the evaluation map and transposition that we will be using. Given $f : A \otimes B \longrightarrow C$ in $\mathbb{C}$ we denote its transpose by $\Lambda_{A,C}(f) : A \longrightarrow B \multimap C$. Λ is natural in A and C, i.e.

$$\Lambda_{A',C}(h(f \otimes 1_B)) = \Lambda_{A,C}(h)f$$

$$\Lambda_{A,C'}(gh) = (1_B \multimap g)\Lambda_{A,C}(h)$$

for all $h : A \otimes B \longrightarrow C$, $f : A' \longrightarrow A$ and $g : C \longrightarrow C'$.

The evaluation map $ev_{A,B} : (A \multimap B) \otimes A \longrightarrow B$ has the universal property: $ev_{A,B}(\Lambda(f) \otimes 1_A) = f$ for $f : C \otimes A \longrightarrow B$ and $ev_{A,B}(g \otimes 1_A) = h$ for $g : C \longrightarrow A \multimap B$. Here $h : C \otimes A \longrightarrow B$ is such that $\Lambda(h) = g$. $ev_{A,B}$ is also natural in B and dinatural in A, i.e. The following diagrams commute:

$$\begin{tikzcd}
(A \multimap B) \otimes A \arrow[r, "ev_{A,B}"] \arrow[d, "(1_A \multimap f) \otimes 1_A"'] \arrow[dr, phantom, "B \xrightarrow{f} B'"] & B \arrow[d, "f"] \\
(A \multimap B') \otimes A \arrow[r, "ev_{A,B'}"'] & B'
\end{tikzcd}$$

$$
\begin{tikzcd}
 & (A \multimap B) \otimes A \arrow[r, "ev_{A,B}"] & B \arrow[dd, "1_B"] \\
(A' \multimap B) \otimes A \arrow[ur, "(f \multimap 1_B) \otimes 1_A"] \arrow[dr, "1_{A' \multimap B} \otimes f"'] & A \xrightarrow{f} A' & \\
 & (A' \multimap B) \otimes A' \arrow[r, "ev_{A',B}"'] & B
\end{tikzcd}
$$

The principal types for the combinators, given earlier in this section, will guide us in defining the combinators.

Recall that we have the following retractions: $!V \lhd V(p,q)$, $V \multimap V \lhd V(r,s)$ and $I \lhd V$.

We proceed with the definition of the combinators. In each case, we first define the combinator, next we give the diagrams necessary for the verification of the respective identity for the combinator being defined

Identity Combinator I:

$$I =_{def} I \xrightarrow{\Lambda(1_V)} V \multimap V \xrightarrow{r} V$$

The following diagram commutes for all $x : I \longrightarrow V$:

$$
\begin{tikzcd}
(V \multimap V) \otimes V \arrow[r, "ev"] & V \\
I \otimes V = V \arrow[u, "\Lambda(1_V) \otimes 1_V"] \arrow[ur, "1_V"'] & \\
I \otimes I = I \arrow[u, "1_I \otimes x"', "= x"] &
\end{tikzcd}
$$

and therefore, $I \cdot x = ev(sI \otimes x) = ev(\Lambda(1_V) \otimes x) = x$.

Weakening Combinator K:

$$K =_{def} I \xrightarrow{\Lambda^2(k)} V \multimap (!V \multimap V) \xrightarrow{1 \multimap (q \multimap 1)} V \multimap (V \multimap V) \xrightarrow{1 \multimap r} V \multimap V \xrightarrow{r} V$$

where $k = 1_V \otimes weak_V$ and $weak_V : !V \longrightarrow I$. The following diagrams commute for all $x, y : I \longrightarrow V$:

$$\begin{tikzcd}
(V \multimap (!V \multimap V)) \otimes V \arrow[r, "ev"] & !V \multimap V \\
I \otimes V = V \arrow[u, "\Lambda^2(k) \otimes 1_V"] \arrow[ur, "\Lambda(k)"'] & \\
I \otimes I \arrow[u, "1 \otimes x = x"] &
\end{tikzcd}$$

(1)

$$\begin{tikzcd}
(!V \multimap V) \otimes !V \arrow[r, "ev"] & V \\
I \otimes !V = !V \arrow[u, "\Lambda(k)x \otimes 1"] \arrow[r, "\Lambda^{-1}(\Lambda(k)x) = h"] & V \arrow[u, "1"'] \\
I \otimes I \arrow[u, "1 \otimes !y\varphi_I"] \arrow[r, "x"] & V \arrow[u, "1"']
\end{tikzcd}
\qquad
\begin{tikzcd}
(V \multimap V) \otimes V \arrow[r, "ev"] & V \\
(V \multimap (!V \multimap V)) \otimes V \arrow[u, "(1 \multimap \alpha) \otimes 1"] \arrow[r, "ev"] & !V \multimap V \arrow[u, "\alpha"'] \\
I \otimes I \arrow[u, "\Lambda^2(k) \otimes x"] \arrow[r, "="] & I \otimes I \arrow[u, "\Lambda(k)x"']
\end{tikzcd}$$

(2) (3)

$\alpha = r(q \multimap 1)$ in diagram (3) above, also diagram (3) uses diagram (1). To see the commutativity of diagram (2) above, consider

$$\begin{aligned} h &= \Lambda^{-1}(\Lambda(k)x) \\ &= \Lambda^{-1}(\Lambda(k(x \otimes 1))) \qquad \text{naturality of } \Lambda \\ &= k(x \otimes 1) \\ &= (1 \otimes weak_V)(x \otimes 1) = (x \otimes weak_V) \end{aligned}$$

Hence

$$\begin{aligned} h(1 \otimes !y\varphi_I) &= (x \otimes weak_V !y\varphi_I) \\ &= (x \otimes weak_I \varphi_I) \qquad \text{naturality of } weak \\ &= (x \otimes 1_I) \qquad weak \text{ is a monoidal natural transformation} \\ &= x. \end{aligned}$$

Given $x, y : I \longrightarrow V$, we have

$$\begin{aligned} K{\bullet}x &= ev(sK \otimes x) \\ &= ev((1 \multimap r(q \multimap 1))\Lambda^2(k) \otimes x) \qquad sr = 1 \\ &= ev((1 \multimap \alpha)\Lambda^2(k) \otimes x) \qquad \text{where } \alpha = r(q \multimap 1) \\ &= \alpha\Lambda(k)x \qquad \text{diagram (3)} \\ &= r(q \multimap 1)\Lambda(k)x. \end{aligned}$$

$$\begin{aligned} (K{\bullet}x){\bullet}!y &= ev(s(K{\bullet}x) \otimes !y) \\ &= ev((q \multimap 1)\Lambda(k)x \otimes p!y\varphi_I) \qquad sr = 1 \\ &= ev(((q \multimap 1) \otimes 1)(1 \otimes p)(\Lambda(k)x \otimes !y\varphi_I)) \\ &= ev(1 \otimes q)(1 \otimes p)(\Lambda(k)x \otimes !y\varphi_I) \qquad \text{dinaturality of } ev \\ &= ev(1 \otimes qp)(\Lambda(k)x \otimes !y\varphi_I) \\ &= ev(\Lambda(k)x \otimes !y\varphi_I) \qquad qp = 1 \\ &= x. \qquad \text{diagram (2)} \end{aligned}$$

Composition Combinator B:

$B =_{def} I \xrightarrow{\Lambda^3(b)} (V \multimap V) \multimap ((V \multimap V) \multimap (V \multimap V)) \xrightarrow{s \multimap (s \multimap r)} V \multimap (V \multimap V) \xrightarrow{1 \multimap r}$ $V \multimap V \xrightarrow{r} V$

where $b = (V \multimap V) \otimes (V \multimap V) \otimes V \xrightarrow{1 \otimes ev} (V \multimap V) \otimes V \xrightarrow{ev} V$

The following diagrams commute for all $x, y, z : I \longrightarrow V$:

$$
\begin{tikzcd}
(V \multimap V) \multimap ((V \multimap V) \multimap (V \multimap V)) \otimes (V \multimap V) \arrow[r, "ev"] & ((V \multimap V) \multimap (V \multimap V)) \\
I \otimes (V \multimap V) \arrow[u, "\Lambda^3(b) \otimes 1"] \arrow[r, "="] & I \otimes (V \multimap V) \arrow[u, "\Lambda^2(b)"'] \\
I \otimes I \arrow[u, "1 \otimes sx"] & (1)
\end{tikzcd}
$$

$$
\begin{tikzcd}
((V \multimap V) \multimap (V \multimap V)) \otimes (V \multimap V) \arrow[r, "ev"] & V \multimap V \\
I \otimes (V \multimap V) \arrow[u, "\Lambda^2(b)sx \otimes 1"] \arrow[r, "="] & I \otimes (V \multimap V) \arrow[u, "\Lambda^{-1}(\Lambda^2(b)sx) = h_1"'] \\
I \otimes I \arrow[u, "1 \otimes sy"] & (2)
\end{tikzcd}
$$

$$
\begin{tikzcd}
(V \multimap V) \otimes V \arrow[r, "ev"] & V \\
I \otimes V \arrow[u, "h_1 sy \otimes 1"] \arrow[ur, "h_2 = \Lambda^{-1}(h_1 sy)"'] & \\
I \otimes I \arrow[u, "1 \otimes z"] &
\end{tikzcd}
\qquad (3)
$$

$$
\begin{aligned}
h_1 sy &= \Lambda^{-1}(\Lambda(\Lambda(b))sx)sy \\
&= \Lambda^{-1}(\Lambda(\Lambda(b)(sx \otimes 1))sy && \text{naturality of } \Lambda \\
&= \Lambda(b)(sx \otimes 1)sy
\end{aligned}
$$

$$
\begin{aligned}
h_2 z &= \Lambda^{-1}(h_1 sy)z \\
&= \Lambda^{-1}(\Lambda(b)(sx \otimes 1)sy)z \\
&= \Lambda^{-1}(\Lambda(b((sx \otimes 1)sy \otimes 1))z && \text{naturality of } \Lambda \\
&= b((sx \otimes 1)sy \otimes 1)z
\end{aligned}
$$

$$
\begin{tikzcd}
(V \multimap V) \otimes V \arrow[rr, "ev"] & & V \arrow[dd, "1"'] \\
((V \multimap V) \multimap V) \otimes V \arrow[u, "(s \multimap 1) \otimes 1"] \arrow[d, "1 \otimes s"'] & V \xrightarrow{s} V \multimap V & \\
((V \multimap V) \multimap V) \otimes (V \multimap V) \arrow[rr, "ev"] & & V \\
((V \multimap V) \multimap ((V \multimap V) \multimap (V \multimap V))) \otimes V \multimap V \arrow[u, "(1 \multimap r(s \multimap r)) \otimes 1"] \arrow[rr, "ev"] & & (V \multimap V) \multimap (V \multimap V) \arrow[u, "r(s \multimap r)"'] \\
I \arrow[u, "\Lambda^3(b) \otimes sx"] \arrow[rr, "="] & & I \arrow[u, "\Lambda^2(b)sx"']
\end{tikzcd}
$$

(4)

$$
\begin{tikzcd}
(V \multimap V)\otimes V \arrow[rr, "ev"] & & V \arrow[dd, "1"'] \\
((V \multimap V) \multimap V)\otimes V \arrow[u, "(s \multimap 1)\otimes 1"] \arrow[d, "1 \otimes s"'] & V \xrightarrow{s} V \multimap V & \\
((V \multimap V) \multimap V)\otimes (V \multimap V) \arrow[rr, "ev"] & & V \\
((V \multimap V) \multimap (V \multimap V))\otimes (V \multimap V) \arrow[u, "(1 \multimap r)\otimes 1"] \arrow[rr, "ev"] & & (V \multimap V) \arrow[u, "r"] \\
I \arrow[u, "\Lambda^2(b)sx \otimes sy"] \arrow[rr, "="] & & I \arrow[u, "h_1 sy"]
\end{tikzcd}
$$

(5)

$$
\begin{aligned}
B{\bullet}x &= ev(sB \otimes x) \\
&= ev((s \multimap r(s \multimap r))\Lambda^3 b \otimes x) \qquad sr = 1 \\
&= ev((s \multimap 1)\otimes 1)(1 \multimap r(s \multimap r)\otimes 1)(\Lambda^3(b)\otimes x) \\
&= ev(1 \otimes s)(1 \multimap r(s \multimap r)\otimes 1)(\Lambda^3(b)\otimes x) \qquad \text{dinaturality of } ev \\
&= ev(1 \multimap r(s \multimap r)\otimes s)(\Lambda^3(b)\otimes x) \\
&= ev(1 \multimap r(s \multimap r)\otimes 1)(1 \otimes s)(\Lambda^3(b)\otimes x) \\
&= r(s \multimap r)ev(1 \otimes s)(\Lambda^3(b)\otimes x) \qquad \text{diagram (4)} \\
&= r(s \multimap r)ev(\Lambda^3(b)\otimes sx) \\
&= r(s \multimap r)\Lambda^2(b)sx \qquad \text{diagram (1)}
\end{aligned}
$$

$$
\begin{aligned}
B{\bullet}x{\bullet}y &= (B{\bullet}x){\bullet}y \\
&= ev(s(B{\bullet}x) \otimes y) \\
&= ev((s \multimap r)\Lambda^2(b)sx \otimes y) \qquad sr = 1 \\
&= ev((s \multimap 1) \otimes 1)((1 \multimap r) \otimes 1)(\Lambda^2(b)sx \otimes y) \\
&= ev(1 \otimes s)((1 \multimap r) \otimes 1)(\Lambda^2(b)sx \otimes y) \qquad \text{diagram (4)} \\
&= ev((1 \multimap r) \otimes s)(\Lambda^2(b)sx \otimes y) \\
&= ev((1 \multimap r) \otimes 1)(1 \otimes s)(\Lambda^2(b)sx \otimes y) \\
&= rev(1 \otimes s)(\Lambda^2(b)sx \otimes y) \qquad \text{diagram (5)} \\
&= rev(\Lambda^2(b)sx \otimes sy) \\
&= rh_1sy \qquad \text{diagram (2)} \\
&= r\Lambda(b)(sx \otimes 1)sy
\end{aligned}
$$

$$
\begin{aligned}
B{\bullet}x{\bullet}y{\bullet}z &= (B{\bullet}x{\bullet}y){\bullet}z \\
&= ev(s(B{\bullet}x{\bullet}y) \otimes z) \\
&= ev(\Lambda(b)(sx \otimes 1)sy \otimes z) \qquad sr = 1 \\
&= ev(h_1sy \otimes z) \\
&= h_2z \qquad \text{diagram (3)} \\
&= b((sx \otimes 1)sy \otimes 1)z \\
&= ev(1 \otimes ev)((sx \otimes 1)sy \otimes 1)z \\
&= ev(1 \otimes ev)((sx \otimes 1)(1_I \otimes sy) \otimes 1)(1_I \otimes z) \\
&= ev(1 \otimes ev)(sx \otimes sy \otimes 1)(1 \otimes z) \\
&= ev(1 \otimes ev)(sx \otimes sy \otimes z) \\
&= ev(sx \otimes ev(sy \otimes z)) \\
&= ev(sx \otimes y{\bullet}z) \qquad \text{definition of } {\bullet} \\
&= x{\bullet}(y{\bullet}z) \qquad \text{definition of } {\bullet}
\end{aligned}
$$

Exchange Combinator C :

$$C =_{def} I \xrightarrow{\Lambda^3(c)} (V \multimap (V \multimap V)) \multimap (V \multimap (V \multimap V)) \xrightarrow{(1 \multimap s) \multimap (1 \multimap r)}$$

$(V \multimap V) \multimap (V \multimap V) \xrightarrow{s \multimap r} V \multimap V \xrightarrow{r} V$

where $c = (V \multimap (V \multimap V)) \otimes V \otimes V \xrightarrow{1 \otimes \sigma} (V \multimap (V \multimap V)) \otimes V \otimes V \xrightarrow{ev \otimes 1}$ $(V \multimap V) \otimes V \xrightarrow{ev} V$

The following diagrams commute for all $x, y, z : I \longrightarrow V$:

$$\begin{tikzcd}
((V \multimap (V \multimap V)) \multimap (V \multimap (V \multimap V))) \otimes (V \multimap (V \multimap V)) \arrow[r, "ev"] & V \multimap (V \multimap V) \\
I \otimes (V \multimap (V \multimap V)) \arrow[u, "\Lambda^3(c) \otimes 1"] \arrow[r, "\Lambda^2(c)"] & V \multimap (V \multimap V) \arrow[u, "1"'] \\
I \otimes I \arrow[u, "1 \otimes (1 \multimap s)sx"] & (1)
\end{tikzcd}$$

$$\begin{tikzcd}
(V \multimap (V \multimap V)) \otimes V \arrow[r, "ev"] & V \multimap V \\
I \otimes V \arrow[u, "\Lambda^2(c)(1 \multimap s)sx \otimes 1"'] \arrow[r, "h_1 = \Lambda^{-1}(\Lambda^2(c)(1 \multimap s)sx)"'] & V \multimap V \arrow[u, "1"'] \\
I \otimes I \arrow[u, "1 \otimes y"] & (2)
\end{tikzcd}$$

$$\begin{tikzcd}
(V \multimap V) \otimes V \arrow[r, "ev"] & V \\
I \otimes V \arrow[u, "h_1 y \otimes 1"] \arrow[ur, "h_2 = \Lambda^{-1}(h_1 y)"'] & \\
I \otimes I \arrow[u, "1 \otimes z"] & (3)
\end{tikzcd}$$

$$\begin{tikzcd}
(V \multimap V) \otimes V \arrow[r, "ev"] & V \arrow[dd, "1"] \\
((V \multimap (V \multimap V)) \multimap V) \otimes V \arrow[u, "((1 \multimap s)s \multimap 1) \otimes 1"] \arrow[d, "1 \otimes (1 \multimap s)s"'] & \\
((V \multimap (V \multimap V)) \multimap V) \otimes (V \multimap (V \multimap V)) \arrow[r, "ev"] & V \\
((V \multimap (V \multimap V)) \multimap (V \multimap (V \multimap V))) \otimes (V \multimap (V \multimap V)) \arrow[u, "(1 \multimap (r(1 \multimap r))) \otimes 1"] \arrow[r, "ev"] & V \multimap (V \multimap V) \arrow[u, "r(1 \multimap r)"']
\end{tikzcd}$$

(4)

$$\begin{tikzcd}
(V \multimap V) \otimes V \arrow[r, "ev"] & V \\
(V \multimap (V \multimap V)) \otimes V \arrow[u, "(1 \multimap r) \otimes 1"] \arrow[r, "ev"] & V \multimap V \arrow[u, "r"']
\end{tikzcd}
\qquad
\begin{tikzcd}
(V \multimap V) \otimes V \arrow[r, "ev"] \arrow[d, "(1 \multimap s) \otimes 1"'] & V \arrow[d, "s"] \\
(V \multimap (V \multimap V)) \otimes V \arrow[r, "ev"] & V \multimap V
\end{tikzcd}$$

(5) (6)

$$\begin{aligned}
h_1 y &= \Lambda^{-1}(\Lambda(\Lambda(c))(1 \multimap s)sx)y \\
&= \Lambda^{-1}(\Lambda(\Lambda(c)((1 \multimap s)sx \otimes 1))y && \text{naturality of } \Lambda \\
&= \Lambda(c)((1 \multimap s)sx \otimes 1)y
\end{aligned}$$

$$\begin{aligned}
h_2 &= \Lambda^{-1}(h_1 y) \\
&= \Lambda^{-1}(\Lambda(c)((1 \multimap s)sx \otimes 1)y) \\
&= \Lambda^{-1}(\Lambda(c(((1 \multimap s)sx \otimes 1)y \otimes 1)) && \text{naturality of } \Lambda \\
&= c(((1 \multimap s)sx \otimes 1)y \otimes 1)
\end{aligned}$$

$$
\begin{aligned}
C{\bullet}x &= ev(sC \otimes x) \\
&= ev(((1 \multimap s)s \multimap r(1 \multimap r))\Lambda^3(c) \otimes x) \qquad sr = 1 \\
&= ev(((1 \multimap s)s \multimap 1) \otimes 1)((1 \multimap r(1 \multimap r)) \otimes 1)(\Lambda^3(c) \otimes x) \\
&= ev(1 \otimes (1 \multimap s)s)((1 \multimap r(1 \multimap r)) \otimes 1)(\Lambda^3(c) \otimes x) \qquad \text{diagram (4)} \\
&= ev((1 \multimap r(1 \multimap r)) \otimes (1 \multimap s)s)(\Lambda^3(c) \otimes x) \\
&= ev((1 \multimap r(1 \multimap r)) \otimes 1)(1 \otimes (1 \multimap s)s)(\Lambda^3(c) \otimes x) \\
&= r(1 \multimap r)ev(1 \otimes (1 \multimap s)s)(\Lambda^3(c) \otimes x) \qquad \text{diagram (4)} \\
&= r(1 \multimap r)ev(\Lambda^3(c) \otimes (1 \multimap s)sx) \\
&= r(1 \multimap r)\Lambda^2(c)(1 \multimap s)sx \qquad \text{diagram (1)}
\end{aligned}
$$

$$
\begin{aligned}
C{\bullet}x{\bullet}y &= (C{\bullet}x){\bullet}y \\
&= ev(s(C{\bullet}x) \otimes y) \\
&= ev((1 \multimap r)\Lambda^2(c)(1 \multimap s)sx \otimes y) \qquad sr = 1 \\
&= ev((1 \multimap r) \otimes 1)(\Lambda^2(c)(1 \multimap s)sx \otimes y) \\
&= rev(\Lambda^2(c)(1 \multimap s)sx \otimes y) \qquad \text{diagram (5)} \\
&= rh_1y \qquad \text{diagram (2)} \\
&= r\Lambda(c)((1 \multimap s)sx \otimes 1)y
\end{aligned}
$$

$$
\begin{aligned}
C{\bullet}x{\bullet}y{\bullet}z &= (C{\bullet}x{\bullet}y){\bullet}z \\
&= ev(s(C{\bullet}x{\bullet}y) \otimes z) \\
&= ev(\Lambda(c)((1 \multimap s)sx \otimes 1)y \otimes z) \qquad sr = 1 \\
&= h_2z \qquad \text{diagram (3)} \\
&= c(((1 \multimap s)sx \otimes 1)y \otimes 1)z
\end{aligned}
$$

$$
\begin{aligned}
&= ev(ev \otimes 1)(1 \otimes \sigma)(((1 \multimap s)sx \otimes 1)y \otimes 1)z \qquad \text{definition of } c \\
&= ev(ev \otimes 1)(1 \otimes \sigma)(((1 \multimap s)sx \otimes 1)(1_I \otimes y) \otimes 1)(1_I \otimes z) \\
&= ev(ev \otimes 1)(1 \otimes \sigma)((1 \multimap s)sx \otimes y \otimes z) \\
&= ev(ev \otimes 1)((1 \multimap s)sx \otimes \sigma(y \otimes z)) \\
&= ev(ev \otimes 1)((1 \multimap s)sx \otimes z \otimes y) \\
&= ev(ev((1 \multimap s)sx \otimes z) \otimes y) \\
&= ev(sev(sx \otimes z) \otimes y) \qquad \text{diagram (6)} \\
&= ev(s(x{\bullet}z) \otimes y) \qquad \text{definition of } \bullet \\
&= (x{\bullet}z){\bullet}y \qquad \text{definition of } \bullet
\end{aligned}
$$

Contraction Combinator W:

$W =_{def} I \xrightarrow{\Lambda^2(w)} (!V \multimap (!V \multimap V)) \multimap (!V \multimap V) \xrightarrow{1 \multimap (q \multimap 1)} (!V \multimap (!V \multimap V)) \multimap (V \multimap V) \xrightarrow{(p \multimap (p \multimap 1)) \multimap 1} (V \multimap (V \multimap V)) \multimap (V \multimap V) \xrightarrow{(1 \multimap s) \multimap r} (V \multimap V) \multimap V \xrightarrow{s \multimap 1} V \multimap V \xrightarrow{r} V$

where $w = (!V \multimap (!V \multimap V)) \otimes !V \xrightarrow{1 \otimes con_V} (!V \multimap (!V \multimap V)) \otimes (!V \otimes !V) \xrightarrow{ev \otimes 1} (!V \multimap V) \otimes !V \xrightarrow{ev} V$

$$
\begin{tikzcd}
(!V \multimap (!V \multimap V)) \multimap (!V \multimap V) \otimes !V \multimap (!V \multimap V) \arrow[r, "ev"] & !V \multimap V \\
I \otimes !V \multimap (!V \multimap V) \arrow[u, "\Lambda^2(w) \otimes 1"] \arrow[r, "\Lambda(w)"] & !V \multimap V \arrow[u, "1"'] \\
I \otimes I \arrow[u, "1 \otimes (p \multimap (p \multimap 1)s)sx"] & (1)
\end{tikzcd}
$$

$$\begin{tikzcd}
(!V \multimap V)\otimes !V \arrow[r, "ev"] & V \\
I\otimes !V \arrow[u, "\Lambda(w)(p \multimap (p \multimap 1)s)sx \otimes 1"] \arrow[r, "h"] & V \arrow[u, "1"'] \\
I \otimes I \arrow[u, "1\otimes !y\varphi_I"] & (2)
\end{tikzcd}$$

$$\begin{aligned}
h &= \Lambda^{-1}(\Lambda(w)(p \multimap (p \multimap 1)s)sx) \\
&= \Lambda^{-1}(\Lambda(w((p \multimap (p \multimap 1)s)sx \otimes 1))) && \text{naturality of } \Lambda \\
&= w((p \multimap (p \multimap 1)s)sx \otimes 1)
\end{aligned}$$

$$\begin{tikzcd}
!I \arrow[r, "con_I"] & !I\otimes !I \\
I \arrow[u, "\varphi_I"] \arrow[r, "="] & I \otimes I \arrow[u, "\varphi_I \otimes \varphi_I"']
\end{tikzcd}$$

(3)

$$\begin{tikzcd}
(V \multimap V) \otimes V \arrow[r, "ev"] & V \arrow[dd, "1"'] \\
(!V \multimap (!V \multimap V) \multimap V) \otimes V \arrow[u, "((p \multimap (p \multimap 1)s)s \multimap 1) \otimes 1"] \arrow[d, "1 \otimes (p \multimap (p \multimap 1)s)s"'] & \\
(!V \multimap (!V \multimap V) \multimap V) \otimes !V \multimap (!V \multimap V) \arrow[r, "ev"] & V \\
((!V \multimap (!V \multimap V)) \multimap (!V \multimap V)) \otimes !V \multimap (!V \multimap V) \arrow[u, "(1 \multimap r(q \multimap 1)) \otimes 1"] \arrow[r, "ev"] & !V \multimap V \arrow[u, "r(q \multimap 1)"']
\end{tikzcd}$$

(4)

$$\begin{aligned}
W{\bullet}x &= ev(sW \otimes x) \\
&= ev((p \multimap (p \multimap 1))(1 \multimap s)s \multimap r(q \multimap 1))\Lambda^2(w) \otimes x) \qquad sr = 1 \\
&= ev((p \multimap (p \multimap 1)s)s \multimap r(q \multimap 1))\Lambda^2(w) \otimes x) \\
&= ev((p \multimap (p \multimap 1)s)s \multimap 1)(1 \multimap r(q \multimap 1))\Lambda^2(w) \otimes x) \\
&= ev((((p \multimap (p \multimap 1)s)s \multimap 1) \otimes 1)((1 \multimap r(q \multimap 1)) \otimes 1)(\Lambda^2(w) \otimes x) \\
&= ev(1 \otimes (p \multimap (p \multimap 1)s)s)(((1 \multimap r(q \multimap 1)) \otimes 1)(\Lambda^2(w) \otimes x) \qquad \text{diagram (4)} \\
&= ev((1 \multimap r(q \multimap 1)) \otimes (p \multimap (p \multimap 1)s)s)(\Lambda^2(w) \otimes x) \\
&= ev((1 \multimap r(q \multimap 1)) \otimes 1)(1 \otimes (p \multimap (p \multimap 1)s)s)(\Lambda^2(w) \otimes x) \\
&= r(q \multimap 1)ev(1 \otimes (p \multimap (p \multimap 1)s)s)(\Lambda^2(w) \otimes x) \qquad \text{diagram(4)} \\
&= r(q \multimap 1)ev(\Lambda^2(w) \otimes (p \multimap (p \multimap 1)s)sx) \\
&= r(q \multimap 1)\Lambda(w)(p \multimap (p \multimap 1)s)sx \qquad \text{diagram (1)}
\end{aligned}$$

$$
\begin{aligned}
W{\bullet}x{\bullet}!y &= (W{\bullet}x){\bullet}!y \\
&= ev(s(W{\bullet}x) \otimes !y) \\
&= ev((q \multimap 1)\Lambda(w)(p \multimap (p \multimap 1)s)sx \otimes p!y\varphi_I) \qquad sr = 1 \\
&= ev(((q \multimap 1) \otimes 1)(\Lambda(w)(p \multimap (p \multimap 1)s)sx \otimes p!y\varphi_I) \\
&= ev(1 \otimes q)(\Lambda(w)(p \multimap (p \multimap 1)s)sx \otimes p!y\varphi_I) \qquad \text{dinaturality of } ev \\
&= ev(\Lambda(w)(p \multimap (p \multimap 1)s)sx \otimes !y\varphi_I) \qquad qp = 1 \\
&= h!y\varphi_I \qquad \text{diagram (2)} \\
&= w((p \multimap (p \multimap 1)s)sx \otimes 1)!y\varphi_I \\
&= ev(ev \otimes 1)(1 \otimes con_V)((p \multimap (p \multimap 1)s)sx \otimes !y\varphi_I) \\
&= ev(ev \otimes 1)((p \multimap (p \multimap 1)s)sx \otimes con_V(!y\varphi_I)) \\
&= ev(ev \otimes 1)((p \multimap (p \multimap 1)s)sx \otimes (!y \otimes !y)con_I\varphi_I) \qquad \text{naturality of } con \\
&= ev(ev((p \multimap (p \multimap 1)s)sx \otimes ((!y \otimes !y)con_I\varphi_I)) \\
&= ev(ev((p \multimap (p \multimap 1)s)sx) \otimes !y\varphi_I) \otimes !y\varphi_I \qquad \text{diagram (3)} \\
&= ev(ev((1 \multimap (p \multimap 1)s) \otimes 1)((p \multimap 1)sx \otimes 1)(1 \otimes !y\varphi_I)) \otimes !y\varphi_I) \\
&= ev((p \multimap 1)sev((p \multimap 1)sx \otimes 1)(1 \otimes !y\varphi_I)) \otimes !y\varphi_I) \qquad \text{naturality of } ev \\
&= ev((p \multimap 1)sev((p \multimap 1) \otimes 1)(sx \otimes !y\varphi_I)) \otimes !y\varphi_I) \\
&= ev((p \multimap 1)sev(1 \otimes p)(sx \otimes !y\varphi_I) \otimes !y\varphi_I) \qquad \text{dinaturality of } ev \\
&= ev((p \multimap 1)sev(sx \otimes p!y\varphi_I) \otimes !y\varphi_I) \\
&= ev((p \multimap 1)s(x{\bullet}!y) \otimes !y\varphi_I) \qquad \text{definition of } {\bullet} \\
&= ev(((p \multimap 1) \otimes 1)(s(x{\bullet}!y) \otimes !y\varphi_I)) \\
&= ev(1 \otimes p)(s(x{\bullet}!y) \otimes !y\varphi_I)) \qquad \text{dinaturality of } ev \\
&= ev(s(x{\bullet}!y) \otimes p!y\varphi_I) \\
&= x{\bullet}!y{\bullet}!y \qquad \text{definition of } {\bullet}
\end{aligned}
$$

Comultiplication Combinator δ :

$\delta =_{def} I \xrightarrow{\Lambda(\delta_V)} !V \multimap !!V \xrightarrow{1 \multimap !p} !V \multimap !V \xrightarrow{q \multimap p} V \multimap V \xrightarrow{r} V$ where δ_V : $!V \longrightarrow !!V$ is the component of dereliction monoidal natural transformation at V.

$$
\begin{tikzcd}
(!V \multimap !!V) \otimes !V \arrow[r, "ev"] & !!V \\
I \otimes !V \arrow[u, "\Lambda(\delta_V) \otimes 1"] \arrow[r, "\delta_V"] & !!V \arrow[u, "1"'] \\
I \otimes I \arrow[u, "!x\varphi_I"]
\end{tikzcd}
$$

(1)

$$
\begin{tikzcd}
(V \multimap V) \otimes V \arrow[r, "ev"] & V \arrow[dd, "1"'] \\
(!V \multimap V) \otimes V \arrow[u, "(q \multimap 1) \otimes 1"'] \arrow[d, "1 \otimes q"'] & \\
(!V \multimap V) \otimes !V \arrow[r, "ev"] & V \\
(!V \multimap !!V) \otimes !V \arrow[u, "(1 \multimap p!p) \otimes 1"] \arrow[r, "ev"] & !!V \arrow[u, "p!p"]
\end{tikzcd}
$$

(2)

$$
\begin{tikzcd}
!V \arrow[r, "\delta_V"] & !!V \\
!I \arrow[u, "!x"'] \arrow[r, "\delta_I"] & !!I \arrow[u, "!!x"'] \\
I \arrow[u, "\varphi_I"'] \arrow[r, "="] & I \arrow[u, "!\varphi_I \varphi_I"']
\end{tikzcd}
$$

(3)

$$
\begin{aligned}
\delta \bullet !x &= ev(s\delta \otimes p!x\varphi_I) \\
&= ev((q \multimap p!p)\Lambda(\delta_V) \otimes p!x\varphi_I) \qquad sr = 1 \\
&= ev((q \multimap 1) \otimes 1)((1 \multimap p!p) \otimes 1)(\Lambda(\delta_V) \otimes p!x\varphi_I) \\
&= ev(1 \otimes q)((1 \multimap p!p) \otimes 1)(\Lambda(\delta_V) \otimes p!x\varphi_I) \qquad \text{diagram (2)} \\
&= ev((1 \multimap p!p) \otimes 1)(1 \otimes q)(\Lambda(\delta_V) \otimes p!x\varphi_I) \\
&= p!pev(1 \otimes q)(\Lambda(\delta_V) \otimes p!x\varphi_I) \qquad \text{diagram (2)} \\
&= p!pev(\Lambda(\delta_V) \otimes !x\varphi_I) \qquad qp = 1 \\
&= p!p\delta_V !x\varphi_I \qquad \text{diagram (1)} \\
&= p!p!!x\delta_I\varphi_I \qquad \text{diagram (3)} \\
&= p!p!!x!\varphi_I\varphi_I \qquad \text{diagram (3)} \\
&= \mathbf{!!}x \qquad \text{definition of } \mathbf{!}x
\end{aligned}
$$

Dereliction Combinator D:

$D =_{def} I \xrightarrow{\Lambda^2(d)} (V \multimap V) \multimap (!V \multimap V) \xrightarrow{s \multimap (q \multimap 1)} V \multimap (V \multimap V) \xrightarrow{1 \multimap r} V \multimap V \xrightarrow{r} V$

where $d = (V \multimap V)\otimes !V \xrightarrow{1\otimes der_V} (V \multimap V) \otimes V \xrightarrow{ev} V$.

$$\begin{tikzcd}
((V \multimap V) \multimap (!V \multimap V)) \otimes (V \multimap V) \arrow[r, "ev"] & !V \multimap V \\
I \otimes V \multimap V \arrow[u, "\Lambda^2(d)\otimes 1"] \arrow[r, "\Lambda(d)"] & !V \multimap V \arrow[u, "1"'] \\
I \otimes I \arrow[u, "1\otimes sx"]
\end{tikzcd}$$

(1)

$$\begin{tikzcd}
(!V \multimap V)\otimes !V \arrow[r, "ev"] & V \\
I \otimes !V \arrow[u, "\Lambda(d)sx \otimes 1"'] \arrow[r, "h = \Lambda^{-1}(\Lambda(d)sx)"] & V \arrow[u, "1"] \\
I \otimes I \arrow[u, "1\otimes !y\varphi_I"]
\end{tikzcd}$$

(2)

$$\begin{tikzcd}
(V \multimap V) \otimes V \arrow[r, "ev"] & V \arrow[dd, "1"'] \\
((V \multimap V) \multimap V) \otimes V \arrow[u, "(s \multimap 1)\otimes 1"] \arrow[d, "1 \otimes s"] & \\
((V \multimap V) \multimap V) \otimes (V \multimap V) \arrow[r, "ev"] & V \\
((V \multimap V) \multimap (!V \multimap V)) \otimes V \multimap V \arrow[u, "(1 \multimap (r(q \multimap 1)))\otimes 1"] \arrow[r, "ev"] & !V \multimap V \arrow[u, "r(q \multimap 1)"]
\end{tikzcd}$$

$$\begin{tikzcd}
!V \arrow[r, "der_V"] & V \\
!I \arrow[u, "!y"] \arrow[r, "der_I"] & I \arrow[u, "y"'] \\
I \arrow[u, "\varphi_I"] \arrow[r, "="] & I \arrow[u, "1"']
\end{tikzcd}$$

(4)

(3)

$$\begin{aligned}
h &= \Lambda^{-1}(\Lambda(d)sx) \\
&= \Lambda^{-1}(\Lambda(d(sx \otimes 1))) && \text{naturality of } \Lambda \\
&= d(sx \otimes 1)
\end{aligned}$$

$$
\begin{aligned}
D{\cdot}x &= ev(sD \otimes x) \\
&= ev((s \multimap r(q \multimap 1))\Lambda^2(d) \otimes x) \qquad sr = 1 \\
&= ev((1 \multimap r(q \multimap 1) \otimes 1)((s \multimap 1) \otimes 1)(\Lambda^2(d) \otimes x) \\
&= r(q \multimap 1)ev((s \multimap 1) \otimes 1)(\Lambda^2(d) \otimes x) \qquad \text{diagram (3)} \\
&= r(q \multimap 1)ev(1 \otimes s)(\Lambda^2(d) \otimes x) \qquad \text{diagram (3)} \\
&= r(q \multimap 1)ev(\Lambda^2(d) \otimes sx) \\
&= r(q \multimap 1)\Lambda(d)sx \qquad \text{diagram (1)}
\end{aligned}
$$

$$
\begin{aligned}
D{\cdot}x{\cdot}!y &= (D{\cdot}x){\cdot}!y \\
&= ev(s(D{\cdot}x) \otimes p!y\varphi_I) \\
&= ev((q \multimap 1)\Lambda(d)sx \otimes p!y\varphi_I) \qquad sr = 1 \\
&= ev((q \multimap 1) \otimes 1)(\Lambda(d)sx \otimes p!y\varphi_I) \\
&= ev(1 \otimes q)(\Lambda(d)sx \otimes p!y\varphi_I) \qquad \text{dinaturality of } ev \\
&= ev(\Lambda(d)sx \otimes !y\varphi_I) \qquad qp = 1 \\
&= h!y\varphi_I \qquad \text{diagram (2)} \\
&= d(sx \otimes 1)!y\varphi_I \\
&= ev(1 \otimes der_V)(sx \otimes 1)!y\varphi_I \qquad \text{definition of } d \\
&= ev(sx \otimes der_V)(1 \otimes !y\varphi_I) \\
&= ev(sx \otimes der_V !y\varphi_I) \\
&= ev(sx \otimes y der_I \varphi_I) \qquad \text{diagram (4)} \\
&= ev(sx \otimes y) \qquad \text{diagram (4)} \\
&= x{\cdot}y \qquad \text{definition of } \cdot
\end{aligned}
$$

Functoriality Combinator F:

$F =_{def} I \xrightarrow{\Lambda^2(f)} !(V \multimap V) \multimap (!V \multimap !V) \xrightarrow{!s \multimap (q \multimap p)} !V \multimap (V \multimap V) \xrightarrow{q \multimap 1} V \multimap (V \multimap V) \xrightarrow{1 \multimap r} V \multimap V \xrightarrow{r} V.$

where $f = !(V \multimap V) \otimes !V \xrightarrow{\varphi} !((V \multimap V) \otimes V) \xrightarrow{!ev} !V$

$$\begin{tikzcd}
(!(V \multimap V) \multimap (!V \multimap !V)) \otimes !(V \multimap V) \arrow[r, "ev"] & !V \multimap !V \\
I \otimes !(V \multimap V) \arrow[u, "\Lambda^2(f) \otimes 1"] \arrow[r, "\Lambda(f)"] & !V \multimap !V \arrow[u, "1"'] \\
I \otimes I \arrow[u, "1 \otimes !s!x\varphi_I"] & (1)
\end{tikzcd}$$

$$\begin{tikzcd}
(!V \multimap !V) \otimes !V \arrow[r, "ev"] & !V \\
I \otimes !V \arrow[u, "\Lambda(f)!s!x\varphi_I \otimes 1"] \arrow[r, "h"] & !V \arrow[u, "1"'] \\
I \otimes I \arrow[u, "1 \otimes !y\varphi_I"] & (2)
\end{tikzcd}$$

$$\begin{aligned}
h &= \Lambda^{-1}(\Lambda(f)!s!x\varphi_I) \\
&= \Lambda^{-1}(\Lambda(f(!s!x\varphi_I \otimes 1))) \\
&= f(!s!x\varphi_I \otimes 1)
\end{aligned}$$

$$
\begin{tikzcd}
(V \multimap V) \otimes V \arrow[r, "ev"] & V \arrow[dd, "1"'] \\
(!(V \multimap V) \multimap V) \otimes V \arrow[u, "(!sq \multimap 1) \otimes 1"] \arrow[d, "1 \otimes !sq"'] & \\
(!(V \multimap V) \multimap V) \otimes !(V \multimap V) \arrow[r, "ev"] & V \\
(!(V \multimap V) \multimap (!V \multimap !V)) \otimes !(V \multimap V) \arrow[u, "(1 \multimap r(q \multimap p)) \otimes 1"] \arrow[r, "ev"] & !V \multimap !V \arrow[u, "r(q \multimap p)"']
\end{tikzcd}
$$

(3)

$$
\begin{tikzcd}
I \otimes I \arrow[r, "="] \arrow[d, "\varphi_I \otimes \varphi_I"] & I \arrow[d, "\varphi_I"] \\
!I \otimes !I \arrow[r, "\varphi"] & !(I \otimes I)
\end{tikzcd}
$$

(4)

$$
\begin{aligned}
F \cdot !x &= ev(sF \otimes p!x\varphi_I) \\
&= ev((!sq \multimap r(q \multimap p))\Lambda^2(f) \otimes p!x\varphi_I) \qquad sr = 1 \\
&= ev((!sq \multimap 1) \otimes 1)((1 \multimap r(q \multimap p)) \otimes 1)(\Lambda^2(f) \otimes p!x\varphi_I) \\
&= ev(1 \otimes !sq)((1 \multimap r(q \multimap p)) \otimes 1)(\Lambda^2(f) \otimes p!x\varphi_I) \qquad \text{diagram (3)} \\
&= ev((1 \multimap r(q \multimap p)) \otimes 1)(1 \otimes !sq)(\Lambda^2(f) \otimes p!x\varphi_I) \\
&= r(q \multimap p)ev(\Lambda^2(f) \otimes !s!x\varphi_I) \qquad \text{diagram (3)} \\
&= r(q \multimap p)\Lambda(f)!s!x\varphi_I \qquad \text{diagram (1)}
\end{aligned}
$$

$$
\begin{aligned}
F \cdot !x \cdot !y &= ev(s(F \cdot !x) \otimes !y) \\
&= ev((q \multimap p)\Lambda(f)!s!x\varphi_I \otimes p!y\varphi_I) \qquad sr = 1 \\
&= ev((1 \multimap p) \otimes 1)((q \multimap 1) \otimes 1)(\Lambda(f)!s!x\varphi_I \otimes p!y\varphi_I) \\
&= pev((q \multimap 1) \otimes 1)(\Lambda(f)!s!x\varphi_I \otimes p!y\varphi_I) \qquad \text{naturality of } ev
\end{aligned}
$$

$$
\begin{aligned}
&= pev(1 \otimes q)(\Lambda(f)!s!x\varphi_I \otimes p!y\varphi_I) && \text{dinaturality of } ev \\
&= pev(\Lambda(f)!s!x\varphi_I \otimes !y\varphi_I) && qp = 1 \\
&= ph!y\varphi_I && \text{diagram (2)} \\
&= pf(!s!x\varphi_I \otimes 1)(1 \otimes !y\varphi_I) \\
&= p!ev\varphi(!s!x\varphi_I \otimes 1)(1 \otimes !y\varphi_I) && \text{definition of } f \\
&= p!ev\varphi(!(sx)\varphi_I \otimes 1)(1 \otimes !y\varphi_I) && ! \text{ is a functor} \\
&= p!ev\varphi(!(sx) \otimes !y)(\varphi_I \otimes \varphi_I) \\
&= p!ev!(sx \otimes y)\varphi(\varphi_I \otimes \varphi_I) && ! \text{ is a monoidal functor} \\
&= p!(ev(sx \otimes y))\varphi(\varphi_I \otimes \varphi_I) && ! \text{ is a functor} \\
&= p!(ev(sx \otimes y))\varphi_I && \text{diagram (4)} \\
&= \mathbf{!}(x \bullet y) && \text{definition of } \bullet \text{ and } \mathbf{!}
\end{aligned}
$$

□

Definition 6.2.6 A *(Standard) Combinatory Algebra* consists of a pair $(\mathcal{A}, \bullet_s)$, where $\mathcal{A}$ is a nonempty set and $\bullet_s$ is a binary operation on $\mathcal{A}$, and distinguished elements B_s, C_s, I_s, K_s, and W_s of $\mathcal{A}$ satisfying the following identities for all x, y, z variables ranging over $\mathcal{A}$:

1. $B_s \bullet_s x \bullet_s y \bullet_s z = x \bullet_s (y \bullet_s z)$
2. $C_s \bullet_s x \bullet_s y \bullet_s z = (x \bullet_s z) \bullet_s y$
3. $I_s \bullet_s x = x$
4. $K_s \bullet_s x \bullet_s y = x$
5. $W_s \bullet_s x \bullet_s y = x \bullet_s y \bullet_s y$

It is more common in the literature to use combinators K_s and S_s in the definition of a combinatory algebra where $S_s \bullet_s x \bullet_s y \bullet_s z = (x \bullet_s z) \bullet_s (y \bullet_s z)$. However, as we discussed in Section 6.1, the set $\{B_s, C_s, I_s, K_s, W_s\}$ also forms a basis for the terms of combinatory logic and S_s can be defined in terms of this basis elements, indeed $S_s = B_s{\bullet_s}(B_s{\bullet_s}(B_s{\bullet_s}W_s){\bullet_s}C_s){\bullet_s}(B_s{\bullet_s}B_s)$.

Let $(\mathcal{A}, \bullet, !)$ be a linear combinatory algebra. We define a binary operation $\bullet_s$ on $\mathcal{A}$ as follows: for $\alpha, \beta \in \mathcal{A}$, $\alpha{\bullet_s}\beta =_{def} \alpha{\bullet}!\beta$. [1] Next consider the following elements of $\mathcal{A}$.

1. $B_s =_{def} C{\bullet}(B{\bullet}(B{\bullet}B{\bullet}B){\bullet}(D{\bullet}I)){\bullet}(C{\bullet}((B{\bullet}B){\bullet}F){\bullet}\delta)$
2. $C_s =_{def} D{\bullet}C$
3. $I_s =_{def} D{\bullet}I$
4. $K_s =_{def} D{\bullet}K$
5. $W_s =_{def} D{\bullet}W$

Theorem 6.2.7 *Let $(\mathcal{A}, \bullet, !)$ be a linear combinatory algebra. Then $(\mathcal{A}, \bullet_s)$ with $\bullet_s$ and the elements B_s, C_s, I_s, K_s, W_s as defined above is a combinatory algebra.*

Proof. We verify the necessary equations:

$$\begin{aligned} C_s{\bullet_s}x{\bullet_s}y{\bullet_s}z &= DC!x!y!z \\ &= Cx!y!z \\ &= (x!z)!y \\ &= (x{\bullet_s}z){\bullet_s}y \end{aligned}$$

$$\begin{aligned} I_s{\bullet_s}x &= (DI)!x \\ &= Ix \\ &= x \end{aligned}$$

$$\begin{aligned} K_s{\bullet_s}x{\bullet_s}y &= DK!x!y \\ &= Kx!y \\ &= x \end{aligned}$$

[1] The same observation has been made by Girard (see [44], page 17.)

$$
\begin{aligned}
B_s \bullet_s x \bullet_s y \bullet_s z &= C(B(BBB)(DI))(C((BB)F\delta)!x!y!z \\
&= B(BBB)(DI)!x(C((BB)F\delta)!y!z \\
&= BBB(DI!x)(C((BB)F)\delta)!y!z \\
&= BBB(Ix)(C((BB)F)\delta)!y!z \\
&= BBBx(C((BB)F)\delta)!y!z \\
&= B(Bx)(C((BB)F)\delta)!y!z \\
&= Bx(C((BB)F)\delta!y)!z \\
&= x(C((BB)F)\delta!y!z) \\
&= x((BB)F!y\delta!z) \\
&= x(B(F!y)\delta!z) \\
&= x(F!y(\delta!z)) \\
&= x(F!y(!!z)) \\
&= x(!(y!z)) \\
&= x!(y \bullet_s z) \\
&= x \bullet_s (y \bullet_s z)
\end{aligned}
$$

$$
\begin{aligned}
W_s \bullet_s x \bullet_s y &= DW!x!y \\
&= Wx!y \\
&= x!y!y \\
&= x \bullet_s y \bullet_s y
\end{aligned}
$$

□

We remark that in the case of CCCs (see Proposition 6.2.3 above) the associated linear combinatory algebra agrees with the (standard) combinatory algebra structure, since

$$x \bullet_s y = x \bullet !y = x \bullet y.$$

6.3 General GoI Construction

In this section we will study an important class of examples of WLCs which are constructed using traced symmetric monoidal categories. This will relate the notion of trace to computation and will yield new models of untyped combinatory logic incorporating the trace operator into the binary application operator in a combinatory algebra.

Definition 6.3.1 A *GoI Situation* is a triple $(\mathbb{C}, T, U)$ where:

1. $\mathbb{C}$ is a traced symmetric monoidal category
2. $T : \mathbb{C} \longrightarrow \mathbb{C}$ is a traced symmetric monoidal functor with the following retractions:

   (a) $TT \lhd T$ (e, e') (Comultiplication)

   (b) $Id \lhd T$ (l, l') (Dereliction)

   (c) $T \otimes T \lhd T$ (c, c') (Contraction)

   (d) $\mathcal{K}_I \lhd T$ (w, w') (Weakening). Here $\mathcal{K}_I$ is the constant I functor.

   Note that the retraction morphisms above are monoidal natural transformations.

3. U is an object of $\mathbb{C}$, called *a reflexive object*, with retractions:

   (a) $U \otimes U \lhd U$ (j, k)

   (b) $I \lhd U$

   (c) $TU \lhd U$ (u, v)

Given a GoI Situation $(\mathbb{C}, T, U)$ with $U \otimes U \lhd U(j, k)$ and $TU \lhd U(u, v)$, where (j, k) and (u, v) are the associated retract pairs, consider the homset $\mathcal{G}(\mathbb{C})(I, V)$ where $I = (I, I)$ and $V = (U, U)$. Recall that $\mathcal{G}$ is the compact closure construction we discussed in Chapter 2. Note that by definition (since we are in the strict case) $\mathcal{G}(\mathbb{C})(I, V) = \mathbb{C}(U, U)$.

Theorem 6.3.2 *Let $(\mathbb{C}, T, U)$ be a GoI Situation, then $(\mathcal{G}(\mathbb{C})(I, V), \cdot, \mathbf{!})$ is an LCA, where $V = (U, U)$, $\mathcal{G}(\mathbb{C})(I, V) = \mathbb{C}(U, U)$, $f \cdot g = Tr^U_{U,U}((1_U \otimes g)(kfj))$, and $\mathbf{!}f = u(Tf)v$ for any $f, g \in \mathbb{C}(U, U)$.*

Proof. Note that $\mathcal{G}(\mathbb{C})$ is a compact closed category and hence it is symmetric monoidal closed, see Proposition 2.3.3. Let $T = (T, \psi, \psi_I)$, we define the endofunctor $! : \mathcal{G}(\mathbb{C}) \longrightarrow \mathcal{G}(\mathbb{C})$ as follows: $!(A^+, A^-) = (TA^+, TA^-)$ and given $f : (A^+, A^-) \longrightarrow (B^+, B^-)$,

$$!f =_{def} TA^+ \otimes TB^- \xrightarrow{\cong} T(A^+ \otimes B^-) \xrightarrow{Tf} T(A^- \otimes B^+) \xrightarrow{\cong} TA^- \otimes TB^+.$$

It follows easily that ! is a symmetric monoidal functor. Next, we define the following monoidal *pointwise* natural transformations:

- $der_{(A^+,A^-)} : !(A^+, A^-) \longrightarrow (A^+, A^-) =_{def} \sigma_{A^+,TA^-}(l'_{A^+} \otimes l_{A^-})$ where $A \lhd TA\ (l_A, l'_A)$,
- $\delta_{(A^+,A^-)} : !(A^+, A^-) \longrightarrow !!(A^+, A^-) =_{def} \sigma_{T^2A^+,TA^-}(e'_{A^+} \otimes e_{A^-})$ where $T^2A \lhd TA\ (e_A, e'_A)$,
- $con_{(A^+,A^-)} : !(A^+, A^-) \longrightarrow !(A^+, A^-) \otimes !(A^+, A^-) =_{def} \sigma_{TA^+ \otimes TA^+,TA^-}(c'_{A^+} \otimes c_{A^-})$ where $TA \otimes TA \lhd TA\ (c_A, c'_A)$,
- $weak_{(A^+,A^-)} : !(A^+, A^-) \longrightarrow (I, I) =_{def} \sigma_{I,TA^-}(w'_{A^+} \otimes w_{A^-})$ where $I \lhd TA\ (w_A, w'_A)$.

With the definitions given above, $(\mathcal{G}(\mathbb{C}), !)$ is a WLC. Also, it follows (see Lemma 2.3.5) that $V = (U, U)$ is a reflexive object in $\mathcal{G}(\mathbb{C})$ and hence by Theorem 6.2.5 $(\mathcal{G}(\mathbb{C})(I, V), \cdot, \mathbf{!})$ is an LCA. Given $f, g \in \mathcal{G}(\mathbb{C})(I, V)$, $f \cdot g = ev(sf \otimes g) = ev(\sigma_{U \otimes U,U}(k \otimes j)f \otimes g) = Tr^U_{U,U}((1 \otimes g)(kfj))$ see Figure 9 below. Also $\mathbf{!}f = p!f\varphi_I = (\sigma_{U,TU}(u \otimes v))(\psi^{-1}Tf\psi)(\sigma_{TI,I}\psi_I \otimes \psi_I^{-1}) = (\psi_I^{-1} \otimes u)(\psi^{-1}Tf\psi)(\psi_I \otimes v) = u(Tf)v$ (see Figure 10 below.) In the derivation of $\mathbf{!}f$, we have used the following identities which are consequences of T being a monoidal functor and $\mathbb{C}$ a strict category.

$$\psi_{I,U}(\psi_I \otimes v) = v, \qquad (\psi_I^{-1} \otimes u)\psi_{I,U}^{-1} = u$$

$\square$

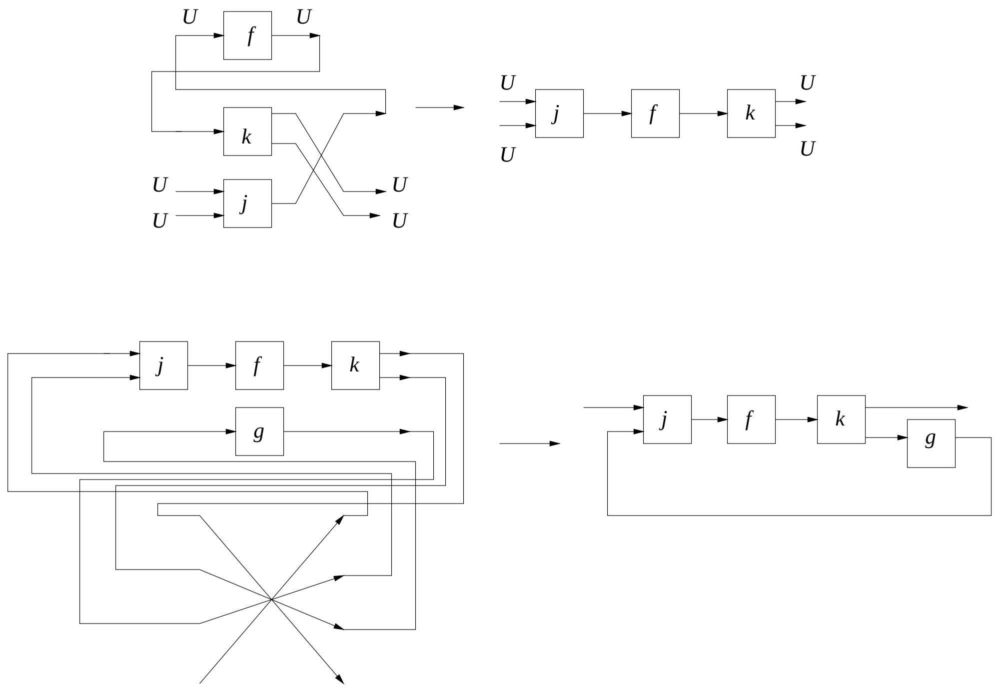


Figure 9: Derivation of $f \bullet g$.

Definitions of the combinators follow from those given in the proof of Theorem 6.2.5. However, due to the importance of the class of examples arising from a GoI Situation we will include the derivation of these definitions in our graphical calculus. Here we will simply include the diagrams for the combinators. The details leading to there definitions starting from the formulas given in Theorem 6.2.5 can be found in Appendix B.

Notation: We introduce as in Figure 1 the following graphical notation: (see Figure 11)

The combinators are defined as follows:

1. $I =_{def} \alpha\gamma\beta$, where

   (a) $\alpha = j$

   (b) $\beta = k$

   (c) $\gamma = \sigma_{U,U}$. See Figure 12.

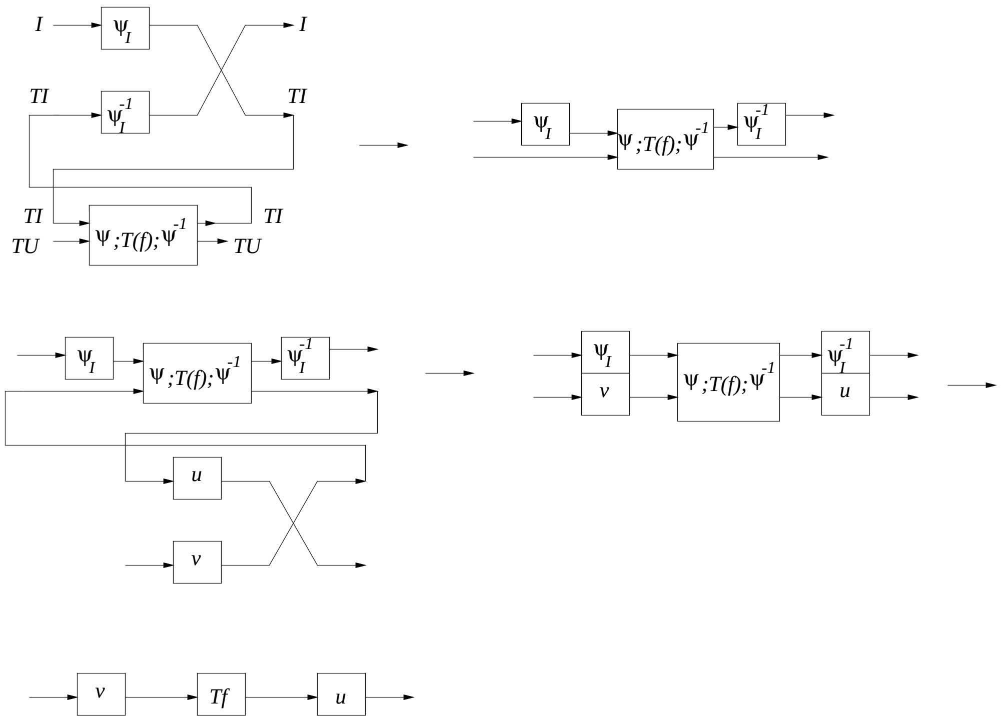


Figure 10: Derivation of $\mathbf{!}f$.

2. $B =_{def} \alpha\gamma\beta$, where

   (a) $\alpha = j(j \otimes 1_U)(j \otimes j \otimes j)$

   (b) $\beta = (k \otimes k \otimes k)(k \otimes 1_U)k$

   (c) $\gamma =$ the permutation π in Figure 13.

3. $C =_{def} \alpha\gamma\beta$, where

   (a) $\alpha = j(j \otimes j)(j \otimes 1_U \otimes j \otimes 1_U)$

   (b) $\beta = (k \otimes 1_U \otimes k \otimes 1_U)(k \otimes k)k$

   (c) $\gamma =$ the permutation π in Figure 14.

4. $K =_{def} \alpha\gamma\beta$, where

   (a) $\alpha = j(j \otimes 1)$

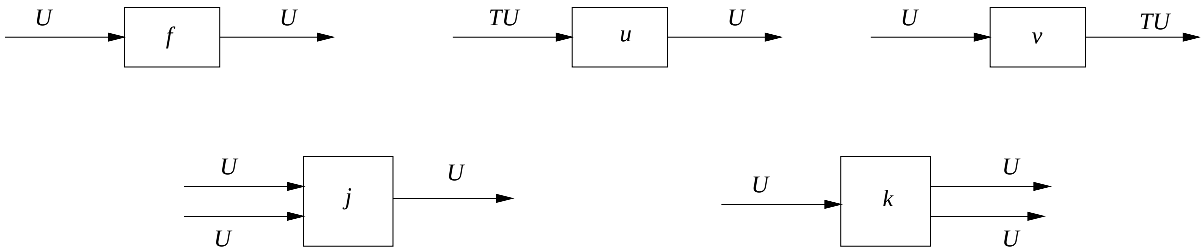


Figure 11: Graphical Representations

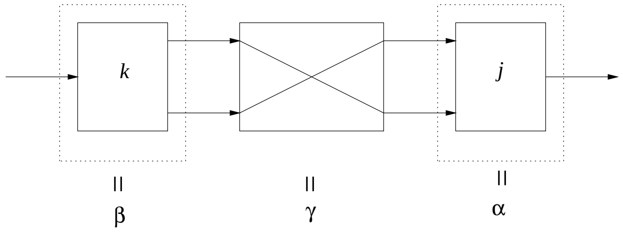


Figure 12: Identity Combinator I

(b) $\beta = (k \otimes 1)k$

(c) $\gamma = \pi(1_U \otimes f_K \otimes 1_U)$, where $f_K = uw_U w'_U v$ and π is the permutation given in Figure 15.

5. $W =_{def} \alpha\gamma\beta$, where

(a) $\alpha = j(1_U \otimes j)(j \otimes j \otimes 1_U)$

(b) $\beta = (k \otimes k \otimes 1_U)(1_U \otimes k)k$

(c) $\gamma = \pi(1_U \otimes g_W \otimes 1_U \otimes f_W)(1 \otimes 1 \otimes 1 \otimes \sigma)$, where $g_W = (u \otimes u)c'_U v$, $f_W =$

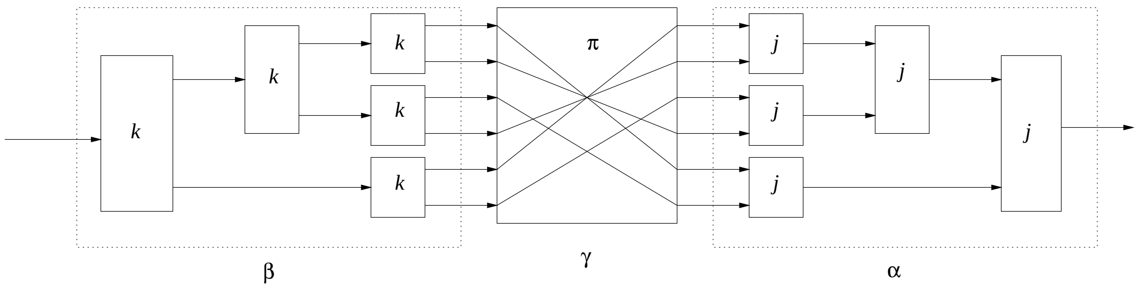


Figure 13: Composition Combinator B

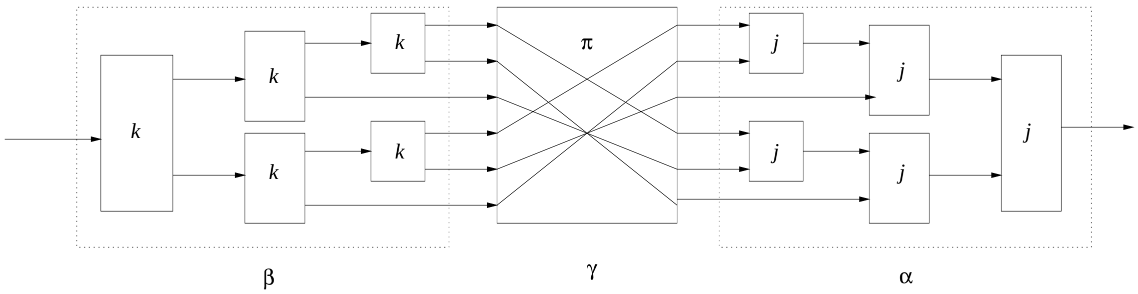


Figure 14: Exchange Combinator C

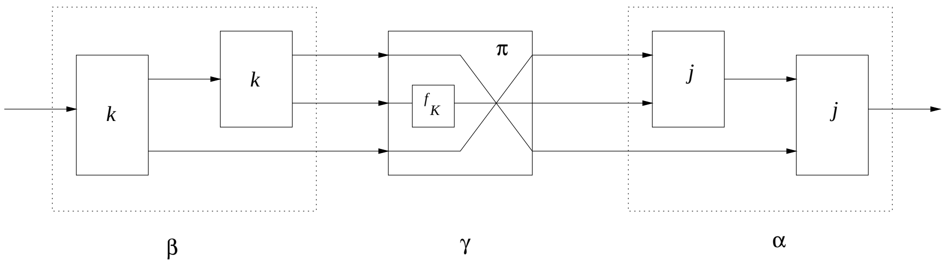


Figure 15: Weakening Combinator K

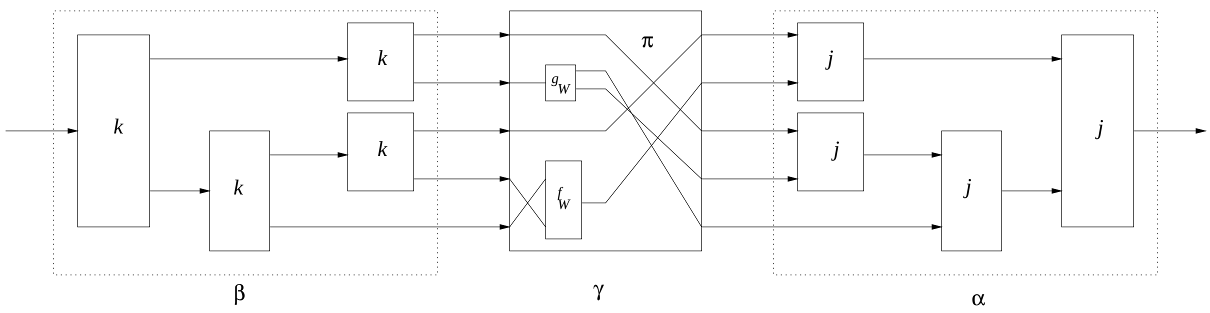


Figure 16: Contraction Combinator W

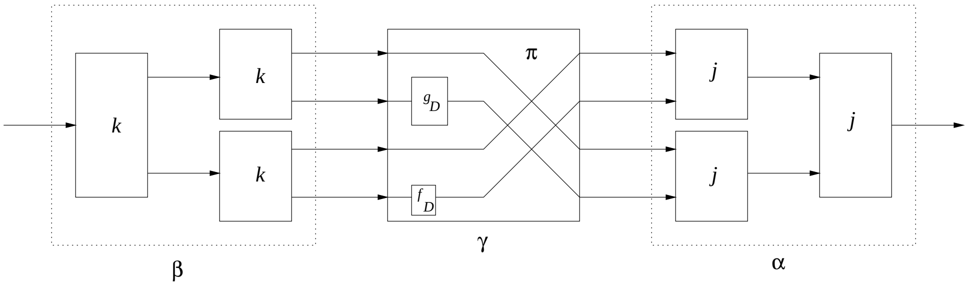


Figure 17: Dereliction Combinator D

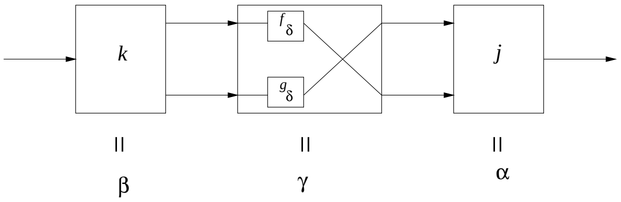


Figure 18: Comultiplication Combinator δ

$uc_U(v \otimes v)$, and π is the permutation given in Figure 16.

6. $D =_{def} \alpha\gamma\beta$, where

   (a) $\alpha = j(j \otimes j)$

   (b) $\beta = (k \otimes k)k$

   (c) $\gamma = \pi(1_U \otimes g_D \otimes 1_U \otimes f_D)$, where $f_D = ul_U$, $g_D = l'_U v$ and π is the permutation given in Figure 17

7. $\delta =_{def} \alpha\gamma\beta$, where

   (a) $\alpha = j$

   (b) $\beta = k$

   (c) $\gamma = \sigma_{U,U}(f_\delta \otimes g_\delta)$, where $f_\delta = ue_U T(v)v$ and $g_\delta = uT(u)e'_U v$. See Figure 18.

8. $F =_{def} \alpha\gamma\beta$, where

   (a) $\alpha = j(j \otimes 1_U)$

(b) $\beta = (k \otimes 1_U)k$

(c) $\gamma = \pi(f_F \otimes g_F)$, where $f_F = uT(j)\psi_{U,U}(v \otimes v)$, $g_F = (u \otimes u)\psi^{-1}_{U,U}T(k)v$ and π is the permutation given in Figure 19.

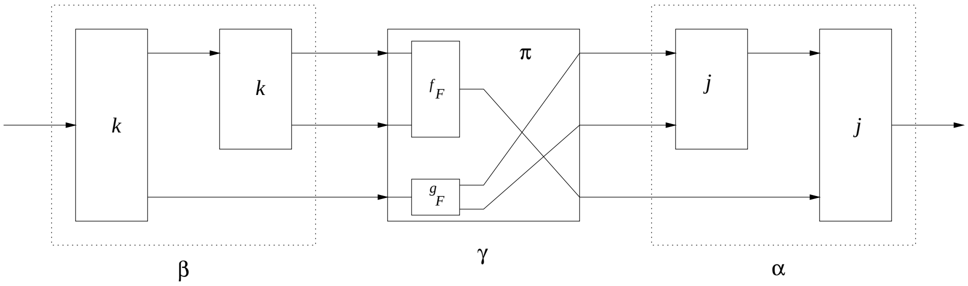


Figure 19: Functoriality Combinator F

There exists an alternative proof of correctness for the combinators in this case (not involving Theorem 6.2.5.) We can use the graphical calculus which involves the rules given by the axioms of traced symmetric monoidal categories, naturality of symmetry morphisms and properties of morphisms used in the definition of combinators to show that combinators satisfy their respective equations. In this approach we consider each combinator as a morphism from $U^n \longrightarrow U^n$ where n is determined by the type of the combinator. We also represent variables in the same way. It is easier to check the correctness of combinators using such a representation. These are given in Appendix C.

Chapter 7

Particle-style Semantics

In the previous chapter we studied WLCs that were constructed from traced symmetric monoidal categories using Abramsky's $\mathcal{G}$ construction. In this chapter we focus on a special class of traced symmetric monoidal categories, namely traced unique decomposition categories introduced in Chapter 4. Such categories provide us with a dataflow-like interpretation and formalise the "particle-style" GoI in the terminology of [1].

7.1 Combinators in UDCs: Dataflow View

As was shown in Chapter 6, linear combinatory algebras can be constructed using traced monoidal categories. This relates the notion of trace to computation and yields new models of untyped combinatory logic, namely combinatory algebras. In this section we show how to define combinators in a unique decomposition category and to construct a linear combinatory algebra from such a category.

Let $\mathbb{C}$ be a traced UDC, $T : \mathbb{C} \longrightarrow \mathbb{C}$ an additive endofunctor and U a reflexive object in $\mathbb{C}$ such that $(\mathbb{C}, T, U)$ forms a GoI Situation. Consider the set $\mathbb{C}(U, U)$. Recall from the previous chapter that we defined the binary operation $\cdot$ and unary operation $!$ on $\mathbb{C}(U, U)$ as follows: given $\alpha, \beta \in \mathbb{C}(U, U)$, $\alpha{\cdot}\beta = Tr^{U}_{U,U}((1_U \otimes \beta)(k\alpha j))$

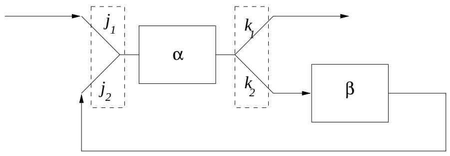


Figure 20: Binary Operation $\bullet$

and $!\alpha = uT(\alpha)v$. In matricial form we have:

$$\alpha{\bullet}\beta = Tr^U_{U,U}\left(\begin{bmatrix} 1 & 0 \\ 0 & \beta \end{bmatrix}\begin{bmatrix} k_1\alpha j_1 & k_1\alpha j_2 \\ k_2\alpha j_1 & k_2\alpha j_2 \end{bmatrix}\right) = k_1\alpha j_1 + \sum_{n\in\omega} k_1\alpha j_2(\beta k_2\alpha j_2)^n(\beta k_2\alpha j_1).$$

where $j = \begin{bmatrix} j_1 & j_2 \end{bmatrix}$ and $k = \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}$. This operation is represented diagrammatically in Figure 20. Note that we have omitted the boxes for j_1, j_2, k_1 and k_2 in order not to load the figure. Recall that $kj = 1_{U\otimes U}$ and hence $k_m j_n = 1_U$ if $m = n$ and 0 otherwise. We will next define the combinators.

7.1.1 Sum-over-Paths: Dataflow Interpretation of Combinators

We explain the process of defining combinators. We purposely use both diagrammatic order $(f; g)$ and algebraic order notation (gf). So (from above) $j_n; k_m = 1_U$ if $m = n$, and 0 else.

Combinator I:

I must satisfy the equation $I{\bullet}x = x$. We view the right hand side of this equation as the required specification; the path from environment Env to x is labelled as 1_U, similarly the path from x to Env is labelled 1_U (see Figure 21(b)). Note that we think of Env as the object U and the connection from Env to x is thus a morphism from U to U namely, 1_U in this case. On the other hand, we view the left hand side of the equation $I{\bullet}x = x$ as the existing structure (circuitry). In this circuit the box representing the combinator I is initially empty and our goal is to design the internal circuitry of I in such a way that we meet the required specification. We proceed as follows (see Figure 21(a)): (1) According to the required specification there must be a path from *Env* to x, hence we put a path with yet unknown label **1** through the box. This provides us with the connection from *Env* to x, namely the path labelled $j_1; \mathbf{1}; k_2$ that will take us from *Env* to x. Now, the specification also requires that the label of this path be 1_U, hence the unknown label, **1**, must be chosen is such a way that $j_1; \mathbf{1}; k_2 = 1_U$, therefore $\mathbf{1} = k_1; j_2$ (using the equations above) (2) Similarly, according to the required specification there must be a path from x to *Env* and hence we put a path labelled **2** through the box for I, thus we have the path labelled $j_2; \mathbf{2}; k_1$ from x to *Env*. Moreover, according to the specification this path must have 1_U as its label; i.e., $j_2; \mathbf{2}; k_1 = 1_U$ and hence $\mathbf{2} = k_2; j_1$. This completes our design for the box I. We have two paths inside the box in parallel with labels $k_1; j_2$ and $k_2; j_1$. We combine this information in a sum which we call *sum-over-paths* formula. Hence, $I = k_1; j_2 + k_2; j_1$ or with categorical order of composition $I = j_2 k_1 + j_1 k_2$. The procedure of design can be compactly represented in a tabular form: here the first and second columns represent the required specification read from the right hand side of the equation. The third column represents the design inside the box with unknown labels (**1** and **2**). Finally the last column provides the last piece of information needed, that is the unknown labels. These can be found by comparing the second and third columns.

Required Specification			
Path	*Label*	*Design*	*Solution*
$Env \longrightarrow x$	1_U	$j_1; \mathbf{1}; k_2$	$\mathbf{1} = k_1; j_2$
$x \longrightarrow Env$	1_U	$j_2; \mathbf{2}; k_1$	$\mathbf{2} = k_2; j_1$

We now verify the correctness of our design for I, namely we would like to verify $I \cdot x = x$ with $I = j_2 k_1 + j_1 k_2$. We will use a dataflow (particle) argument for this verification as follows: Suppose a particle coming from the environment (see Figure 21(a)) arrives at the left hand port of the box I. This would be on the path labeled j_1. Now there are two different paths to choose and only one must be chosen. Since the label of the path on which the particle arrives is j_1 and noting that $j_1; k_2 = 0$ (representing a disconnection) and $j_1; k_1 = 1_U$ (representing connectivity with label 1_U), our particle will choose path **1** (inside the box), labelled $k_1; j_2$ and hence will arrive at the right hand port of the box with a label $j_1; k_1; j_2 = j_2$. At this point the

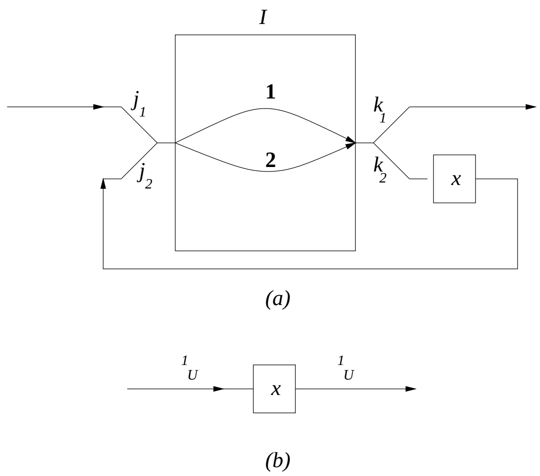


Figure 21: Combinator I

particle is faced with another choice, namely one between paths labelled k_1 and k_2 respectively. However, as the particle has arrived at this point with a label j_2 and since $j_2; k_1 = 0$ represents lack of connectivity, the path labelled k_2 will be chosen taking the particle to the entry of box x. The label of the path from Env to x that the particle has taken is $j_1; k_1; j_2; k_2 = 1_U$ as we expect from the required specification, (see Figure 21(b)). Our particle now passes through x and finds itself on the exit port of x and following the path from there leading to the box for I arrives once again at the left port of the box for I, faced with two choices. This time it chooses path **2** labelled $k_2; j_1$ because $j_2; k_1 = 0$ represents a disconnection and $j_2; k_2 = 1_U$. This way the particle arrives at the right port of the box with a label $j_2; k_2; j_1 = j_1$. This time the way to go will be via the path labelled k_1 which will lead the particle to Env with a label $j_1; k_1 = 1_U$, also in agreement with the required specification. This completes our verification process and establishes the correctness of our design.

Combinator K:

K must satisfy the equation $K \bullet x \bullet !y = x$. Viewing the right hand side of this equation as the required specification for the design of K and the left hand side of the equation as the given circuitry with box K to be designed, we obtain the following information.

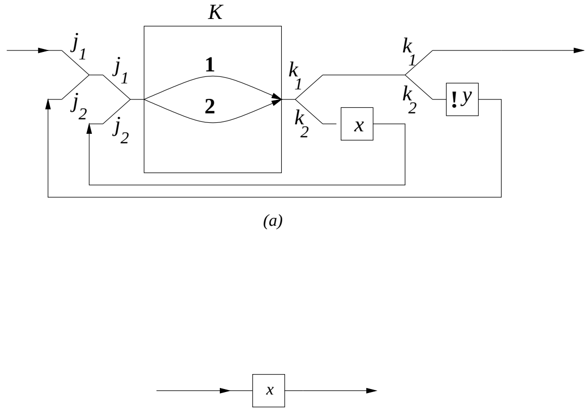


Figure 22: Combinator K

Required Specification			
Path	*Label*	*Design*	*Solution*
$Env \longrightarrow x$	1_U	$j_1; j_1; \mathbf{1}; k_2$	$\mathbf{1} = k_1; k_1; j_2$
$x \longrightarrow Env$	1_U	$j_2; \mathbf{2}; k_1; k_1$	$\mathbf{2} = k_2; j_1; j_1$

Using reasoning similar to the case of combinator I, we find that $K = j_2 k_1^2 + j_1^2 k_2$. We verify the equation of K by following a particle coming from Env through the box; such a particle will necessarily follow the path labelled $j_1; j_1$ and will choose path **1**, labelled $k_1; k_1; j_2$, because $j_1; k_2 = 0$ and $j_1; k_1 = 1_U$, at this point the particle arrives at the junction of k_1 and k_2 and since $j_2; k_2 = 1_U$ and $j_2; k_1 = 0$, it will choose the path labelled by k_2 leading to x. This results in a path with the label $j_1; j_1; k_1; k_1; j_2; k_2 = 1_U$ yielding the desired specification for the path from Env to x. Upon exiting the box x the particle arrives on the path labelled by j_2 and by the same reasoning chooses path **2** and subsequently $k_1; k_1$, exiting to environment, the resulting label being $j_2; k_2; j_1; j_1; k_1; k_1 = 1_U$ as required by the specification.

Combinator B:

B must satisfy the equation $B{\bullet}x{\bullet}y{\bullet}z = x{\bullet}(y{\bullet}z)$. Proceeding as in the previous cases:

Required Specification			
Path	*Label*	*Design*	*Solution*
$Env \longrightarrow x$	j_1	$j_1; j_1; j_1; \mathbf{1}; k_2$	$\mathbf{1} = k_1; k_1; k_1; j_1; j_2$
$x \longrightarrow y$	$k_2; j_1$	$j_2; \mathbf{2}; k_1; k_2$	$\mathbf{2} = k_2; k_2; j_1; j_2; j_1$
$y \longrightarrow z$	k_2	$j_2; j_1; \mathbf{3}; k_1; k_1; k_2$	$\mathbf{3} = k_1; k_2; k_2; j_2; j_1; j_1$
$z \longrightarrow y$	j_2	$j_2; j_1; j_1; \mathbf{4}; k_1; k_2$	$\mathbf{4} = k_1; k_1; k_2; j_2; j_2; j_1$
$y \longrightarrow x$	$k_1; j_2$	$j_2; j_1; \mathbf{5}; k_2$	$\mathbf{5} = k_1; k_2; k_1; j_2; j_2$
$x \longrightarrow Env$	k_1	$j_2; \mathbf{6}; k_1; k_1; k_1$	$\mathbf{6} = k_2; k_1; j_1; j_1; j_1$

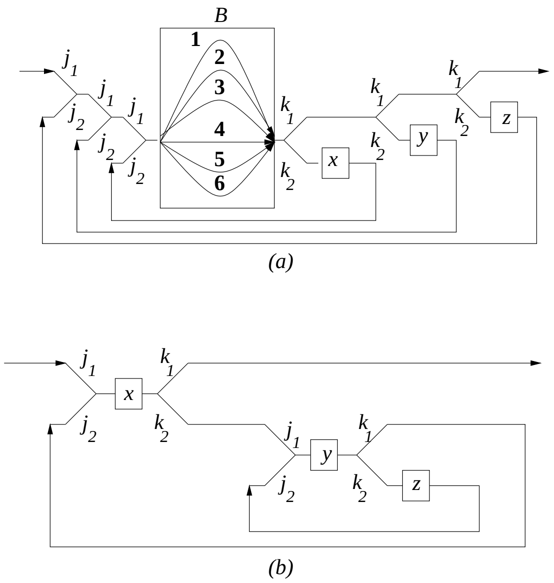


Figure 23: Combinator B

Hence $B = j_2 j_1 k_1^3 + j_1 j_2 j_1 k_2^2 + j_1^2 j_2 k_2^2 k_1 + j_1 j_2^2 k_2 k_1^2 + j_2^2 k_1 k_2 k_1 + j_1^3 k_1 k_2$.

A particle coming from Env will follow the path $j_1; j_1; j_1$ and hence will choose path **1** in the box which will lead it to x with label $j_1; j_1; j_1; k_1; k_1; k_1; j_1; j_2; k_2 = j_1$ as expected. Similarly a particle leaving x will choose path **2** which will lead to y with label $j_2; k_2; k_2; j_1; j_2; j_1; k_1; k_2 = k_2; j_1$ which agrees with the specification. The verification of the correctness for the remaining paths is exactly similar to these arguments.

Combinator C:

C must satisfy the equation $C{\bullet}x{\bullet}y{\bullet}z = x{\bullet}z{\bullet}y$. Proceeding as in the previous cases:

Required Specification			
Path	*Label*	*Design*	*Solution*
$Env \longrightarrow x$	$j_1; j_1$	$j_1; j_1; j_1; \mathbf{1}; k_2$	$\mathbf{1} = k_1; k_1; k_1; j_1; j_1; j_2$
$x \longrightarrow y$	$k_1; k_2$	$j_2; \mathbf{2}; k_1; k_2$	$\mathbf{2} = k_2; k_1; k_2; j_2; j_1$
$x \longrightarrow z$	k_2	$j_2; \mathbf{3}; k_1; k_1; k_2$	$\mathbf{3} = k_2; k_2; j_2; j_1; j_1$
$z \longrightarrow x$	j_2	$j_2; j_1; j_1; \mathbf{4}; k_2$	$\mathbf{4} = k_1; k_1; k_2; j_2; j_2$
$y \longrightarrow x$	$j_2; j_1$	$j_2; j_1; \mathbf{5}; k_2$	$\mathbf{5} = k_1; k_2; j_2; j_1; j_2$
$x \longrightarrow Env$	$k_1; k_1$	$j_2; \mathbf{6}; k_1; k_1; k_1$	$\mathbf{6} = k_2; k_1; k_1; j_1; j_1; j_1$

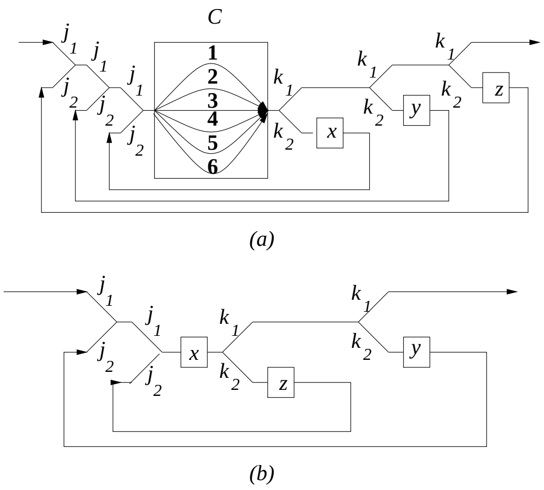


Figure 24: Combinator C

Hence $C = j_2 j_1^2 k_1^3 + j_1 j_2 k_2 k_1 k_2 + j_1^2 j_2 k_2^2 + j_2^2 k_2 k_1^2 + j_2 j_1 j_2 k_2 k_1 + j_1^3 k_1^2 k_2$.

The verification process is analogues to the case of combinator B.

Combinator W:

W must satisfy the equation $W{\bullet}x{\bullet}!y = x{\bullet}!y{\bullet}!y$. Proceeding as in the previous cases:

Required Specification			
Path	*Label*	*Design*	*Solution*
$Env \longrightarrow x$	$j_1; j_1$	$j_1; j_1; \mathbf{1}; k_2$	$\mathbf{1} = k_1; k_1; j_1; j_1; j_2$
$x \longrightarrow \mathbf{!}y$ (first copy)	$k_2; f_{W1}$	$j_2; \mathbf{2}; k_1; k_2$	$\mathbf{2} = k_2; k_2; f_{W1}; j_2; j_1$
$x \longrightarrow \mathbf{!}y$ (second copy)	$k_1; k_2; f_{W2}$	$j_2; \mathbf{3}; k_1; k_2$	$\mathbf{3} = k_2; k_1; k_2; f_{W2}; j_2; j_1$
$\mathbf{!}y$ (first copy) $\longrightarrow x$	$g_{W1}; j_2$	$j_2; j_1; \mathbf{4}; k_2$	$\mathbf{4} = k_1; k_2; g_{W1}; j_2; j_2$
$\mathbf{!}y$ (second copy) $\longrightarrow x$	$g_{W2}; j_2; j_1$	$j_2; j_1; \mathbf{5}; k_2$	$\mathbf{5} = k_1; k_2; g_{W2}; j_2; j_1; j_2$
$x \longrightarrow Env$	$k_1; k_1$	$j_2; \mathbf{6}; k_1; k_1$	$\mathbf{6} = k_2; k_1; k_1; j_1; j_1$

The first copy of $\mathbf{!}y$ is the one on the left in Figure 25(b). Hence

$$W = j_2 j_1^2 k_1^2 + j_1 j_2 f_{W1} k_2^2 + j_1 j_2 f_{W2} k_2 k_1 k_2 + j_2^2 g_{W1} k_2 k_1 + j_2 j_1 j_2 g_{W2} k_2 k_1 + j_1^2 k_1^2 k_2.$$

Where $f_{W1} = uc_U(v \otimes v)\iota_1$, $f_{W2} = uc_U(v \otimes v)\iota_2$, $g_{W1} = \rho_1(u \otimes u)c'_U v$, and $g_{W2} = \rho_2(u \otimes u)c'_U v$. The verification is done similarly to the previous two cases. However, notice that we have replaced the $\mathbf{!}y$'s in the specification by $f_{W1}; \mathbf{!}y; g_{W1}$ and $f_{W2}; \mathbf{!}y; g_{W2}$. This has been done in order to distinguish between the two copies of $\mathbf{!}y$. However, we need to justify this encoding. Indeed we have:

$$\begin{aligned} g_{W1}\mathbf{!}y f_{W1} &= \rho_1(u \otimes u)c'_U v u T(y) v u c_U(v \otimes v)\iota_1 \\ &= \rho_1(u \otimes u)c'_U T(y) c_U(v \otimes v)\iota_1 \\ &= \rho_1(u \otimes u)c'_U c_U(Ty \otimes Ty)(v \otimes v)\iota_1 \qquad \text{naturality of } c_U \\ &= \rho_1(u \otimes u)(Ty \otimes Ty)(v \otimes v)\iota_1 \\ &= uTyv \\ &= \mathbf{!}y \end{aligned}$$

Similarly $g_{W2}\mathbf{!}y f_{W2} = \mathbf{!}y$.

Combinator D:

D must satisfy the equation $D{\bullet}x{\bullet}\mathbf{!}y = x{\bullet}y$. We proceed as in the previous cases, however notice that in defining the specification for combinator D we have replaced y by $f_D; \mathbf{!}y; g_D$, where $f_D = ul_U$ and $g_D = l'_U v$. First, note that $y = f_D; \mathbf{!}y; g_D$ as elements in $\mathbb{C}(U, U)$.

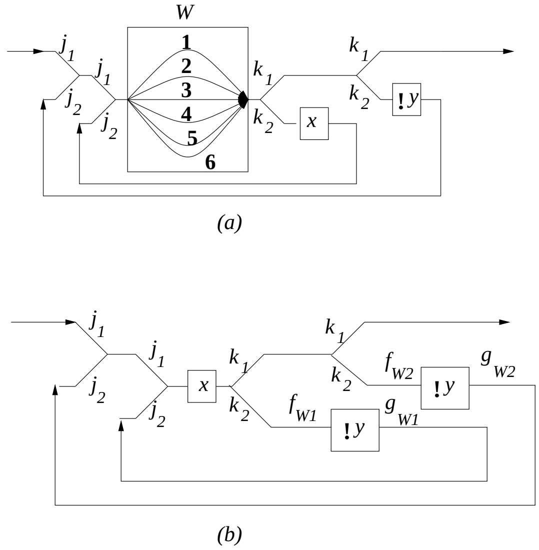


Figure 25: Combinator W

$$
\begin{aligned}
g_D\mathbf{!}yf_D &= l'_U vuT(y)vul_U \\
&= l'_U T(y) l_U \\
&= l'_U l_U y, \qquad \text{naturality of } l_U \\
&= y
\end{aligned}
$$

We use this particular coding because the required specification (right hand side of the equation for D) is given in terms of y, whereas we are only given $\mathbf{!}y$ for our design (see Figure 26(a).) Intuitively, we can think of $\mathbf{!}y$ as infinitely many copies of y and the coding here chooses a single copy out of the infinitely many given copies of y.

Required Specification			
Path	*Label*	*Design*	*Solution*
$Env \longrightarrow x$	j_1	$j_1; j_1; \mathbf{1}; k_2$	$\mathbf{1} = k_1; k_1; j_1; j_2$
$x \longrightarrow \mathbf{!}y$	$k_2; f_D$	$j_2; \mathbf{2}; k_1; k_2$	$\mathbf{2} = k_2; k_2; f_D; j_2; j_1$
$\mathbf{!}y \longrightarrow x$	$g_D; j_2$	$j_2; j_1; \mathbf{3}; k_2$	$\mathbf{3} = k_1; k_2; g_D; j_2; j_2$
$x \longrightarrow Env$	k_1	$j_2; \mathbf{4}; k_1; k_1$	$\mathbf{4} = k_2; k_1; j_1; j_1$

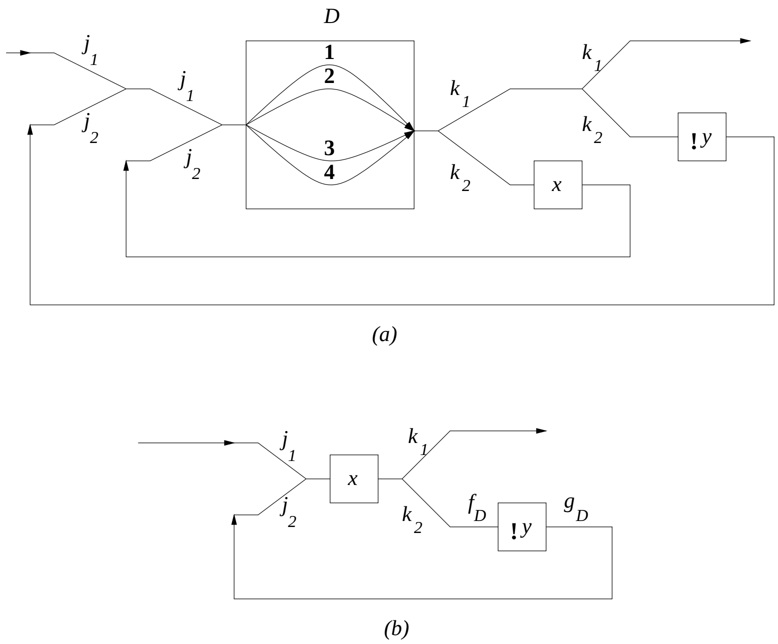


Figure 26: Combinator D

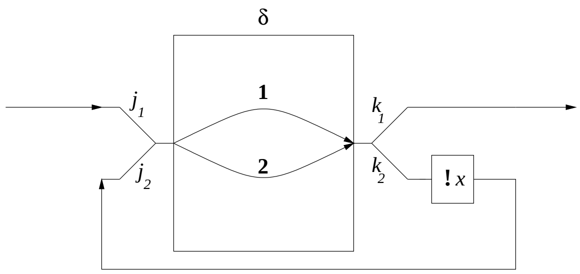
Hence $D = j_2 j_1 k_1^2 + j_1 j_2 f_D k_2^2 + j_2^2 g_D k_2 k_1 + j_1^2 k_1 k_2$. where $f_D = ul_U, g_D = l'_U v$. The verification is similar to the previous cases.

Combinator δ:

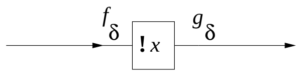
δ must satisfy the equation $\delta \bullet !x = !!x$. As in the previous case we need to explain the way we have found the specification used here. We have replaced $!!x$ by $f_\delta; !x; g_\delta$ because we have to get $!!x$, using $!x$ which is given. Here $f_\delta = ue_U T(v)v$ and $g_\delta = uT(u)e'_U v$. Also note that $!!x = f_\delta; !x; g_\delta$ as morphisms in $\mathbb{C}(U, U)$.

$$\begin{aligned}
g_\delta \mathbf{!}x f_\delta &= uT(u)e'_U vuT(x)vue_U T(v)v \\
&= uT(u)e'_U T(x)e_U T(v)v \\
&= uT(u)e'_U e_U T(T(x))T(v)v \qquad \text{naturality of } e_U \\
&= uT(u)T(T(x))T(v)v \\
&= uT(uT(x)v)v \\
&= uT(\mathbf{!}x)v \\
&= \mathbf{!!}x
\end{aligned}$$

Required Specification			
Path	*Label*	*Design*	*Solution*
$Env \longrightarrow \mathbf{!}x$	f_δ	$j_1; \mathbf{1}; k_2$	$\mathbf{1} = k_1; f_\delta; j_2$
$\mathbf{!}x \longrightarrow Env$	g_δ	$j_2; \mathbf{2}; k_1$	$\mathbf{2} = k_2; g_\delta; j_1$



(a)



(b)

Figure 27: Combinator δ

Hence $\delta = j_2 f_\delta k_1 + j_1 g_\delta k_2$ where $f_\delta = ue_U T(v)v, g_\delta = uT(u)e'_U v$. The verification is similar to the previous cases.

Combinator F:

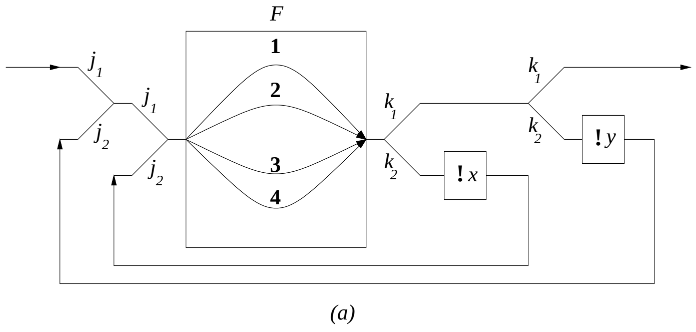
F must satisfy the equation $F{\cdot}\mathbf{!}x{\cdot}\mathbf{!}y = \mathbf{!}(x{\cdot}y)$. Notice that we need to evaluate $\mathbf{!}(x{\cdot}y)$ and express it in terms of $\mathbf{!}x$ and $\mathbf{!}y$ as these are the given morphisms for our design. Hence, we proceed as follows:

$$\begin{array}{rcll}
\mathbf{!}(x{\cdot}y) & = & uT(x{\cdot}y)v & \\
 & = & uT(Tr((1\otimes y)(kxj)))v & \\
 & = & uTr(\psi^{-1}T((1\otimes y)(kxj))\psi)v & T \text{ is traced} \\
 & = & uTr(\psi^{-1}T(1\otimes y)T(kxj)\psi)v & \\
 & = & uTr(\psi^{-1}\psi(T(1)\otimes T(y))\psi^{-1}T(kxj)\psi)v & T \text{ is monoidal} \\
 & = & uTr((1_{TU}\otimes T(y))\psi^{-1}T(kxj)\psi)v & \\
 & = & Tr((u\otimes T(y))\psi^{-1}T(k)T(x)T(j)\psi(v\otimes 1_{TU})) & \text{naturality of } Tr \\
 & = & Tr((u\otimes vuT(y)vu)\psi^{-1}T(k)vuT(x)vuT(j)\psi(v\otimes 1)) & \\
 & = & Tr((u\otimes v\mathbf{!}yu)\psi^{-1}T(k)v\mathbf{!}xuT(j)\psi(v\otimes 1)) & \\
 & = & Tr((1_U\otimes v)(1_U\otimes \mathbf{!}y)(u\otimes u)\psi^{-1}T(k)v\mathbf{!}xuT(j)\psi(v\otimes 1)) & \\
 & = & Tr((1_U\otimes \mathbf{!}y)(u\otimes u)\psi^{-1}T(k)v\mathbf{!}xuT(j)\psi(v\otimes v)) & \text{dinaturality of } Tr \\
 & = & Tr((1\otimes \mathbf{!}y)(g_F\mathbf{!}xf_F)) &
\end{array}$$

where $f_F = uT(j)\psi(v\otimes v)$, $g_F = (u\otimes u)\psi^{-1}T(k)v$ and $T = (T,\psi,\psi_I)$. Also, $f_{F1} = f_F\iota_1$, $f_{F2} = f_F\iota_2$, $g_{F1} = \rho_1 g_F$ and $g_{F2} = \rho_2 g_F$. (see Figure 28(b).)

Proceeding as in the previous cases:

Required Specification			
Path	*Label*	*Design*	*Solution*
$Env \longrightarrow \mathbf{!}x$	f_{F1}	$j_1; j_1; \mathbf{1}; k_2$	$\mathbf{1} = k_1; k_1; f_{F1}; j_2$
$\mathbf{!}x \longrightarrow \mathbf{!}y$	g_{F2}	$j_2; \mathbf{2}; k_1; k_2$	$\mathbf{2} = k_2; g_{F2}; j_2; j_1$
$\mathbf{!}y \longrightarrow \mathbf{!}x$	f_{F2}	$j_2; j_1; \mathbf{3}; k_2$	$\mathbf{3} = k_1; k_2; f_{F2}; j_2$
$\mathbf{!}x \longrightarrow Env$	g_{F1}	$j_2; \mathbf{4}; k_1; k_1$	$\mathbf{4} = k_2; g_{F1}; j_1; j_1$



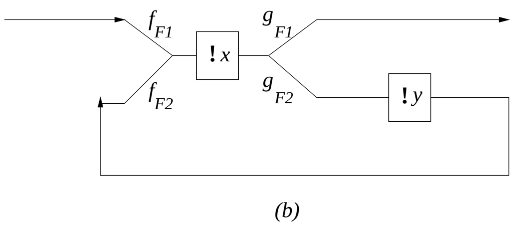


Figure 28: Combinator F

Hence $F = j_2 f_{F1} k_1^2 + j_1 j_2 g_{F2} k_2 + j_2 f_{F2} k_2 k_1 + j_1^2 g_{F1} k_2$.

The verification is similar to the previous cases.

Theorem 7.1.1 *Let $\mathbb{C}$ be a UDC, T an additive endofunctor on $\mathbb{C}$ and U an object in $\mathbb{C}$ such that $(\mathbb{C}, T, U)$ is a GoI Situation. Let the distinguished elements $I, B, C, K, W, \delta, D, F$ of $\mathbb{C}(U, U)$ be as defined above. Then*

1. *$(\mathbb{C}(U, U), \cdot, \mathbf{!})$ is an LCA.*

2. *Combinators respect the dataflow (particle) semantics above.*

Proof. The verification procedure given above constitutes a proof. However, we need to show that the defining terms for combinators (the sums) actually exist. This is achieved using the correspondence of such sums to morphisms in the model category. Below, we give a complete proof for this theorem by showing that the definitions given for combinators in Section 3 of Chapter 6 reduce to the sums that we have derived above.

1. Combinator I:

$$
\begin{aligned}
I &= j\sigma k \\
&= \begin{bmatrix} j_1 & j_2 \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \\
&= j_2 k_1 + j_1 k_2
\end{aligned}
$$

2. Combinator K:

$$
\begin{aligned}
K &= j(j \otimes 1)\pi(1 \otimes f_K \otimes 1)(k \otimes 1)k \\
&= \begin{bmatrix} j_1 & j_2 \end{bmatrix} \begin{bmatrix} j_1 & j_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 \\ 0 & f_K & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} k_1 & 0 \\ k_2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \\
&= j_2 k_1^2 + j_1 j_2 f_K k_2 k_1 + j_1^2 k_2 \\
&= j_2 k_1^2 + j_1^2 k_2.
\end{aligned}
$$

Note that $f_K = u w_U w'_U v$ and $w_U : I \longrightarrow TU = 0_{I,TU}$. This follows from the naturality of w_U and additivity of T: $w_U = w_U 1_I = T(0_{UU})w_U = 0_{TU,TU} w_U = 0_{I,TU}$. Therefore $f_K = 0_{UU}$.

3. Combinator B:

$$
\begin{aligned}
B &= j(j \otimes 1)(j \otimes j \otimes j)\pi(k \otimes k \otimes k)(k \otimes 1)k \\
&= \begin{bmatrix} j_1 & j_2 \end{bmatrix} \begin{bmatrix} j_1 & j_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} j_1 & j_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & j_1 & j_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & j_1 & j_2 \end{bmatrix} \pi(k \otimes k \otimes k)(k \otimes 1)k
\end{aligned}
$$

$$= \left[\begin{array}{cccccc} j_1^3 & j_1^2 j_2 & j_1 j_2 j_1 & j_1 j_2^2 & j_2 j_1 & j_2^2 \end{array}\right] \left[\begin{array}{cccccc} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{array}\right] (k \otimes k \otimes k)(k \otimes 1)k$$

$$= \left[\begin{array}{cccccc} j_1^3 & j_1 j_2^2 & j_2^2 & j_1^2 j_2 & j_1^3 & j_1 j_2 j_1 \end{array}\right] \left[\begin{array}{ccc} k_1 & 0 & 0 \\ k_2 & 0 & 0 \\ 0 & k_1 & 0 \\ 0 & k_2 & 0 \\ 0 & 0 & k_1 \\ 0 & 0 & k_2 \end{array}\right] \left[\begin{array}{cc} k_1 & 0 \\ k_2 & 0 \\ 0 & 1 \end{array}\right] \left[\begin{array}{c} k_1 \\ k_2 \end{array}\right]$$

$$= j_2 j_1 k_1^3 + j_1 j_2^2 k_2 k_1^2 + j_2^2 k_1 k_2 k_1 + j_1^2 j_2 k_2^2 k_1 + j_1^3 k_1 k_2 + j_1 j_2 j_1 k_2^2.$$

4. Combinator C:

$C = j(j \otimes j)(j \otimes 1 \otimes j \otimes 1)\pi(k \otimes 1 \otimes k \otimes 1)(k \otimes k)k$

$$= \left[\begin{array}{cc} j_1 & j_2 \end{array}\right] \left[\begin{array}{cccc} j_1 & j_2 & 0 & 0 \\ 0 & 0 & j_1 & j_2 \end{array}\right] \left[\begin{array}{cccccc} j_1 & j_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & j_1 & j_2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right] \pi(k \otimes 1 \otimes k \otimes 1)(k \otimes k)k$$

$$= \left[\begin{array}{cccccc} j_1^3 & j_1^2 j_2 & j_1 j_2 & j_2 j_1^2 & j_2 j_1 j_2 & j_2^2 \end{array}\right] \left[\begin{array}{cccccc} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{array}\right] (k \otimes 1 \otimes k \otimes 1)(k \otimes k)k$$

$$= \begin{bmatrix} j_2 j_1^2 & j_2^2 & j_2 j_1 j_2 & j_1^3 & j_1 j_2 & j_1^2 j_2 \end{bmatrix} \begin{bmatrix} k_1 & 0 & 0 & 0 \\ k_2 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & k_1 & 0 \\ 0 & 0 & k_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} k_1 & 0 \\ k_2 & 0 \\ 0 & k_1 \\ 0 & k_2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}$$

$$= j_2 j_1 k_1^3 + j_2^2 k_2 k_1^2 + j_2 j_1 j_2 k_2 k_1 + j_1^3 k_1^2 k_2 + j_1 j_2 k_2 k_1 k_2 + j_1^2 j_2 k_2^2.$$

5. Combinator W :

$$W = j(1 \otimes j)(j \otimes j \otimes 1)\pi(1 \otimes g_W \otimes 1 \otimes f_W)(1 \otimes 1 \otimes 1 \otimes \sigma)(k \otimes k \otimes 1)(1 \otimes k)k$$

$$= \begin{bmatrix} j_1 & j_2 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & j_1 & j_2 \end{bmatrix} \begin{bmatrix} j_1 & j_2 & 0 & 0 & 0 \\ 0 & 0 & j_1 & j_2 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \pi(1 \otimes g_W \otimes 1 \otimes f_W)(1 \otimes 1 \otimes 1 \otimes \sigma)$$

$$(k \otimes k \otimes 1)(1 \otimes k)k$$

$$= \begin{bmatrix} j_1^2 & j_2 j_2 & j_2 j_1^2 & j_2 j_1 j_2 & j_2^2 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & f_{W2} & f_{W1} \\ 1 & 0 & 0 & 0 & 0 \\ 0 & g_{W2} & 0 & 0 & 0 \\ 0 & g_{W1} & 0 & 0 & 0 \end{bmatrix} (k \otimes k \otimes 1)(1 \otimes k)k$$

$$= \begin{bmatrix} j_2 j_1^2 & j_2 j_1 j_2 g_{W2} + j_2^2 g_{W1} & j_1^2 & j_1 j_2 f_{W2} & j_1 j_2 f_{W1} \end{bmatrix} \begin{bmatrix} k_1 & 0 & 0 \\ k_2 & 0 & 0 \\ 0 & k_1 & 0 \\ 0 & k_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & k_1 \\ 0 & k_2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}$$

$$= j_2 j_1^2 k_1^2 + j_2 j_1 j_2 g_{W2} k_2 k_1 + j_2^2 g_{W1} k_2 k_1 + j_1^2 k_1^2 k_2 + j_1 j_2 f_{W2} k_2 k_1 k_2 + j_1 j_2 f_{W1} k_2^2.$$

6. Combinator D:

$$D = j(j \otimes j)\pi(1 \otimes g_D \otimes 1 \otimes f_D)(k \otimes k)k$$

$$= \begin{bmatrix} j_1 & j_2 \end{bmatrix} \begin{bmatrix} j_1 & j_2 & 0 & 0 \\ 0 & 0 & j_1 & j_2 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & f_D \\ 1 & 0 & 0 & 0 \\ 0 & g_D & 0 & 0 \end{bmatrix} \begin{bmatrix} k_1 & 0 \\ k_2 & 0 \\ 0 & k_1 \\ 0 & k_2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}$$
$$= j_2 j_1 k_1^2 + j_2^2 g_D k_2 k_1 + j_1^2 k_1 k_2 + j_1 j_2 f_D k_2^2.$$

7. Combinator δ:

$$\delta = j\sigma_{U,U}(f_\delta \otimes g_\delta)k$$
$$= \begin{bmatrix} j_1 & j_2 \end{bmatrix} \begin{bmatrix} 0 & g_\delta \\ f_\delta & 0 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}$$
$$= j_2 f_\delta k_1 + j_1 g_\delta k_2.$$

8. Combinator F:

$$F = j(j \otimes 1)\pi(f_F \otimes g_F)(k \otimes 1)k$$
$$= \begin{bmatrix} j_1 & j_2 \end{bmatrix} \begin{bmatrix} j_1 & j_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & g_{F1} \\ 0 & 0 & g_{F2} \\ f_{F1} & f_{F2} & 0 \end{bmatrix} \begin{bmatrix} k_1 & 0 \\ k_2 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}$$
$$= j_2 f_{F1} k_1^2 + j_2 f_{F2} k_2 k_1 + j_1^2 g_{F1} k_2 + j_1 j_2 g_{F2} k_2.$$

$\square$

Chapter 8

Examples and Computations

In this chapter we consider various examples of GoI Situations formed by traced UDCs, in particular, we give detailed discussion for the categories **PInj**, $\mathbf{Hilb}_2$ and **SRel**. We will also investigate the combinators in these categories. In Section 6 we give two examples of computations with combinators using the algebraic (matricial) machinery that we discussed in Chapter 7. We also discuss Abramsky and Jagadeesan's work on GoI [5] and relate it to a GoI Situation, based on the category of domains. $\mathbb{N}$ and $\mathbb{C}$ denote the set of natural and complex numbers respectively.

8.1 PInj

Consider the category **PInj** of sets and partial injective functions. We have already seen that **PInj** is a traced UDC. We will show that $(\mathbf{PInj}, \mathbb{N}\times -, \mathbb{N})$ is a GoI Situation.

Lemma 8.1.1 *Let $\mathbb{C}$ be a traced UDC and $T : \mathbb{C} \longrightarrow \mathbb{C}$ be an additive functor. Then, T is traced.*

Proof. Let $f : X \otimes U \longrightarrow Y \otimes U$ be a morphism in $\mathbb{C}$, then $Tr^U_{X,Y}(f) = f_{11} +$

$\sum_{n\in\omega} f_{12}f_{22}^n f_{21}$. Let $T = (T, \psi, \psi_I)$ and $g = \psi_{Y,U}^{-1} T(f)\psi_{X,U}$. Then,

$$\begin{aligned} g_{11} &= \rho_1^{TY,TU} g \iota_1^{TX,TU} \\ &= T(\rho_1^{Y,U})\psi_{Y,U} g \psi_{X,U}^{-1} T(\iota_1^{X,U}) \\ &= T(\rho_1^{Y,U})\psi_{Y,U}\psi_{Y,U}^{-1} T(f)\psi_{X,U}\psi_{X,U}^{-1} T(\iota_1^{X,U}) \\ &= T(\rho_1^{Y,U} f \iota_1^{X,U}) \\ &= T(f_{11}). \end{aligned}$$

Similarly, we have $g_{12} = T(f_{12}), g_{21} = T(f_{21})$, and $g_{22} = T(f_{22})$.

$$\begin{aligned} T(Tr_{X,Y}^U(f)) &= T(f_{11} + \textstyle\sum_{n\in\omega} f_{12}f_{22}^n f_{21}) \\ &= T(f_{11}) + \textstyle\sum_{n\in\omega} T(f_{12})T(f_{22})^n T(f_{21}), \ T \text{ is additive} \\ &= g_{11} + \textstyle\sum_{n\in\omega} g_{12}g_{22}^n g_{21} \\ &= Tr_{TX,TY}^{TU}(\psi_{Y,U}^{-1} T(f)\psi_{X,U}). \end{aligned}$$

Therefore, T is traced. □

Proposition 8.1.2 $(\mathbf{PInj}, \mathbb{N}\times -, \mathbb{N})$ *is a GoI Situation.*

Proof. Recall from Chapter 4 that **PInj** is traced symmetric monoidal category. The tensor product is taken to be the disjoint union of sets with $\emptyset$ as the unit. Clearly $T = \mathbb{N} \times -$, with $T = (T, \psi, \psi_I)$, is a symmetric monoidal functor where $\psi_{X,Y} : \mathbb{N}\times X \uplus \mathbb{N}\times Y \longrightarrow \mathbb{N}\times(X\uplus Y)$ is given by $(1,(n,x)) \mapsto (n,(1,x))$ and $(2,(n,y)) \mapsto (n,(2,y))$ and it has an inverse defined by: $(n,(1,x)) \mapsto (1,(n,x))$ and $(n,(2,y)) \mapsto (2,(n,y))$. Also, $\psi_I : \emptyset \longrightarrow \mathbb{N}\times\emptyset$ given by $1_\emptyset$ is clearly an isomorphism. We need to show that T is additive. Let $\{f_i\}_{i\in I}$ be a summable family in $\mathbf{PInj}(X,Y)$, then

$$(1_{\mathbb{N}} \times \textstyle\sum_I f_i)(n,x) = \begin{cases} (n, f_j(x)), & \text{if there exists a } j\in I \text{ such that } x \in Dom(f_j); \\ \text{undefined}, & \text{else.} \end{cases}$$

but this is exactly the definition of $(\sum_I(1_{\mathbb{N}}\times f_i))(n,x)$ for all $(n,x)\in\mathbb{N}\times X$. Therefore, $\mathbb{N}\times -$ is an additive functor and thus, by Lemma 8.1.1 it is also traced. In other words, given $f : X\uplus U \longrightarrow Y\uplus U$ we have $1_{\mathbb{N}}\times Tr_{X,Y}^U(f) = Tr_{\mathbb{N}\times X,\mathbb{N}\times Y}^{\mathbb{N}\times U}(\psi^{-1}(1_{\mathbb{N}}\times f)\psi)$.

We show that $\mathbb{N}$ is a reflexive object.

- $\mathbb{N} \uplus \mathbb{N} \lhd \mathbb{N}(j,k)$ is given as follows: $j : \mathbb{N} \uplus \mathbb{N} \longrightarrow \mathbb{N}, j(1,n) = 2n, j(2,n) = 2n+1$ and $k : \mathbb{N} \longrightarrow \mathbb{N} \uplus \mathbb{N}, k(n) = (1, n/2)$ for n even, and $(2,(n-1)/2)$ for n odd. Clearly $kj = 1_{\mathbb{N} \uplus \mathbb{N}}$. Define $j_1 = j\iota_1, j_2 = j\iota_2$, $k_1 = \rho_1 k$, $k_2 = \rho_2 k$ where ρ_i and ι_i are as defined in Example 4.0.15.

- $\emptyset \lhd \mathbb{N}$ using the empty partial function as the retract morphisms.

- $\mathbb{N} \times \mathbb{N} \lhd \mathbb{N}(u,v)$ is defined as: $u(m,n) = <m,n> = \frac{(m+n+1)(m+n)}{2} + n$ (Cantor surjective pairing) and v as its inverse, $v(n) = (n_1, n_2)$ with $<n_1, n_2> = n$. Clearly, $vu = 1_{\mathbb{N} \times \mathbb{N}}$.

We next define the necessary monoidal natural transformations.

- $\mathbb{N} \times (\mathbb{N} \times X) \xrightarrow{e_X} \mathbb{N} \times X$ and $\mathbb{N} \times X \xrightarrow{e'_X} \mathbb{N} \times (\mathbb{N} \times X)$

  $\mathbb{N} \times (\mathbb{N} \times X) \xrightarrow{e_X} \mathbb{N} \times X$ is defined by, $e_X(n_1,(n_2,x)) = (<n_1,n_2>, x)$. Given $f : X \longrightarrow Y$, $(1_{\mathbb{N}} \times f)e_X((n_1,(n_2,x)) = (<n_1,n_2>, f(x)) = e_Y(1_{\mathbb{N}} \times (1_{\mathbb{N}} \times f)(n_1,(n_2,x))$ for all $n_1, n_2 \in \mathbb{N}$ and $x \in X$ proving the naturality of e_X. $e'_X(n,x) = (n_1,(n_2,x))$ where $<n_1,n_2> = n$. $e'_X e_X(n_1,(n_2,x)) = e'_X(<n_1,n_2>, x) = (n_1,(n_2,x))$ for all $n_1, n_2 \in \mathbb{N}$ and $x \in X$.

- $X \xrightarrow{l_X} \mathbb{N} \times X$ and $\mathbb{N} \times X \xrightarrow{l'_X} X$

  $l_X(x) = (n_0, x)$ for a fixed $n_0 \in \mathbb{N}$. Given $f : X \longrightarrow Y$, $(1_{\mathbb{N}} \times f)l_X(x) = (n_0, f(x)) = l_Y f(x)$ for any $x \in X$, proving the naturality of l_X.

$$l'_X(n,x) = \begin{cases} x, & \text{if } n = n_0; \\ \text{undefined}, & \text{else.} \end{cases}$$

  $l'_X l_X(x) = l'_X(n_0, x) = x$ for all $x \in X$.

- $(\mathbb{N} \times X) \uplus (\mathbb{N} \times X) \xrightarrow{c_X} \mathbb{N} \times X$ and $\mathbb{N} \times X \xrightarrow{c'_X} (\mathbb{N} \times X) \uplus (\mathbb{N} \times X)$.

$$c_X = \begin{cases} (1,(n,x)) \mapsto (2n, x) \\ (2,(n,x)) \mapsto (2n+1, x) \end{cases}$$

  Given $f : X \longrightarrow Y$, $(1_{\mathbb{N}} \times f)c_X(1,(n,x)) = (2n, f(x)) = c_Y(1_{\mathbb{N}} \times f \uplus 1_{\mathbb{N}} \times f)(1,(n,x))$ for all $n \in \mathbb{N}$ and $x \in X$. Similarly $(1_{\mathbb{N}} \times f)c_X(2,(n,x)) = (2n +$

$1, f(x)) = c_Y(1_{\mathbb{N}} \times f \uplus 1_{\mathbb{N}} \times f)(2,(n,x))$ for all $n \in \mathbb{N}$ and $x \in X$, proving the naturality of c_X.

$$c'_X(n,x) = \begin{cases} (1,(n/2,x)), & \text{if } n \text{ is even;} \\ (2,((n-1)/2,x), & \text{if } n \text{ is odd.} \end{cases}$$

Finally, $c'_X c_X(1,(n,x)) = c'_X(2n,x) = (1,(n,x))$ and $c'_X c_X(2,(n,x)) = c'_X(2n+1,x) = (2,(n,x))$.

- $\emptyset \xrightarrow{w_X} \mathbb{N} \times X$ and $\mathbb{N} \times X \xrightarrow{w'_X} \emptyset$.

  Let w_X and w'_X both be the empty partial function. Clearly for any $f : X \longrightarrow Y$, $(1_{\mathbb{N}} \times f)w_X = w'_X 1_\emptyset$, proving the naturality of w_X. $w'_X w_X = 1_\emptyset$.

$\square$

Note that the combinators are elements of $\mathbf{PInj}(\mathbb{N},\mathbb{N})$. We use $n \stackrel{k}{\equiv} n'$ to denote $n \equiv n' (\text{mod } k)$. More explicitly we have :

- Recall that $I = j_2k_1 + j_1k_2$. Hence, $I(n) = n+1$ for n even and $n-1$ for n odd. Note that I is a total function.

- $K = j_2k_1^2 + j_1^2k_2$. Hence, $K(n) = n/2+1$ if $n \stackrel{4}{\equiv} 0$, $2n-2$ if n is odd and undefined else. Therefore K is a partial function and this partiality intuitively corresponds to the information erasing feature of K as a combinator: (since $Kx\mathbf{!}y = x$, $\mathbf{!}y$ is "lost".)

- $B = j_2j_1k_1^3 + j_1^3k_1k_2 + j_2^2k_1k_2k_1 + j_1j_2j_1k_2^2 + j_1^2j_2k_2^2k_1 + j_1j_2^2k_2k_1^2$.

$$B(n) = \begin{cases} n/2+1 & \text{if } n \stackrel{8}{\equiv} 0, \\ 2(n-1) & \text{if } n \stackrel{4}{\equiv} 1, \\ n/2+2 & \text{if } n \stackrel{8}{\equiv} 2, \\ 2n-4 & \text{if } n \stackrel{4}{\equiv} 3, \\ n-2 & \text{if } n \stackrel{8}{\equiv} 6, \\ n+2 & \text{if } n \stackrel{8}{\equiv} 4. \end{cases}$$

  Note that B is a total function.

- $C = j_2 j_1^2 k_1^3 + j_1^3 k_1^2 k_2 + j_1 j_2 k_2 k_1 k_2 + j_2 j_1 j_2 k_2 k_1 + j_1^2 j_2 k_2^2 + j_2^2 k_2 k_1^2$.

$$C(n) = \begin{cases} n+1 & \text{if } n \overset{8}{\equiv} 0, \\ n-1 & \text{if } n \overset{8}{\equiv} 1, \\ (n-1)/2 & \text{if } n \overset{8}{\equiv} 5, \\ 2n+1 & \text{if } n \overset{4}{\equiv} 2, \\ 2(n-1) & \text{if } n \overset{4}{\equiv} 3, \\ n/2+1 & \text{if } n \overset{8}{\equiv} 4. \end{cases}$$

Note that C is a total function.

- $W = j_2 j_1^2 k_1^2 + j_1 j_2 f_{W1} k_2^2 + j_1 j_2 f_{W2} k_2 k_1 k_2 + j_2^2 g_{W1} k_2 k_1 + j_2 j_1 j_2 g_{W2} k_2 k_1 + j_1^2 k_1^2 k_2$, where $f_{W1} = uc_U(v \otimes v)\iota_1$, $f_{W2} = uc_U(v \otimes v)\iota_2$, $g_{W1} = \rho_1(u \otimes u)c'_U v$, and $g_{W2} = \rho_2(u \otimes u)c'_U v$. Therefore, the maps f_{Wi}, g_{Wi} for $i = 1, 2$ are given by:

$f_{W1}(n) = <2n_1, n_2>$,

$f_{W2}(n) = <2n_1 + 1, n_2>$,

$g_{W1}(n) = <n_1/2, n_2>$ if n_1 is even, and undefined else, and

$g_{W2}(n) = <(n_1 - 1)/2, n_2>$ for n_1 odd and undefined else.

Here $n = <n_1, n_2>$.

$$W(n) = \begin{cases} 2n+1 & \text{if } n \overset{4}{\equiv} 0, \\ 4(<m_1/2, m_2>) + 3 & \text{if } n \overset{4}{\equiv} 2,\ m_1 \text{ even},\ (n-2)/4 = <m_1, m_2>, \\ 8(<(m_1-1)/2, m_2>) + 5 & \text{if } n \overset{4}{\equiv} 2,\ m_1 \text{ odd},\ (n-2)/4 = <m_1, m_2>, \\ (n-1)/2 & \text{if } n \overset{8}{\equiv} 1, \\ 4(<2m_1+1, m_2>) + 2 & \text{if } n \overset{8}{\equiv} 5,\ (n-5)/8 = <m_1, m_2>, \\ 4(<2m_1, m_2>) + 2 & \text{if } n \overset{4}{\equiv} 3,\ (n-3)/4 = <m_1, m_2>. \end{cases}$$

Note that W is a total function.

- $D = j_2 j_1 k_1^2 + j_1^2 k_1 k_2 + j_1 j_2 f_D k_2^2 + j_2^2 g_D k_2 k_1$ where $f_D = ul_U, g_D = l'_U v$. Fix a natural number n_0, then $f_D(n) = <n_0, n>$ and

$$g_D(n) = \begin{cases} n_2, & \text{if } n_1 = n_0; \\ \text{undefined}, & \text{else.} \end{cases}$$

where $n = <n_1, n_2>$.

$$D(n) = \begin{cases} n+1 & \text{if } n \overset{4}{\equiv} 0, \\ n-1 & \text{if } n \overset{4}{\equiv} 1, \\ 4(<n_0, (n-3)/4>) + 2 & \text{if } n \overset{4}{\equiv} 3, \\ 4m_2 + 3 & \text{if } n \overset{4}{\equiv} 2,\ m_1 = n_0,\ (n-2)/4 = <m_1, m_2>, \\ \text{undefined} & \text{otherwise.} \end{cases}$$

Note that D is a partial function. Recall that D as a combinator behaves according to $D{\bullet}x{\bullet}!y = x{\bullet}y$, that is intuitively it chooses a single copy of y from the given infinitely many copies of y. We can think of this as erasing all the other copies and hence the partiality in its definition (as a function on natural numbers.)

- $\delta = j_2 f_\delta k_1 + j_1 g_\delta k_2$ where $f_\delta = ue_U T(v)v$, $g_\delta = uT(u)e'_U v$. Therefore $f_\delta(<n_1, <m_1, m_2>>) = \ll n_1, m_1>, m_2>$ and $g_\delta(\ll m_1, m_2>, n_2>) = <m_1, <m_2, n_2\gg$.

$$\delta(n) = \begin{cases} 2(\ll n_1, m_1>, m_2>) + 1 & \text{if } n \text{ even},\ n/2 = <n_1, <m_1, m_2\gg, \\ 2(<m_1, <m_2, n_2\gg) & \text{if } n \text{ odd},\ (n-1)/2 = \ll m_1, m_2>, n_2> . \end{cases}$$

Note that δ is a total function.

- $F = j_2 f_{F1} k_1^2 + j_1^2 g_{F1} k_2 + j_1 j_2 g_{F2} k_2 + j_2 f_{F2} k_2 k_1$ where $f_F = uT(j)\psi(v \otimes v)$, $g_F = (u \otimes u)\psi^{-1} T(k) v$ and $T = (T, \psi, \psi_I)$. Also, $f_{F1} = f_F \iota_1$, $f_{F2} = f_F \iota_2$, $g_{F1} = \rho_1 g_F$ and $g_{F2} = \rho_2 g_F$.

$$f_{F1}(<n_1, n_2>) = <n_1, 2n_2>,$$

$$f_{F2}(<n_1, n_2>) = <n_1, 2n_2 + 1>,$$

$$g_{F1}(<n_1, n_2>) = \begin{cases} <n_1, n_2/2> & \text{if } n_2 \text{ is even,} \\ \text{undefined, else.} \end{cases}$$

$$g_{F2}(<n_1, n_2>) = \begin{cases} <n_1, (n_2 - 1)/2> & \text{if } n_2 \text{ is odd,} \\ \text{undefined} & \text{else.} \end{cases}$$

$$F(n) = \begin{cases} 2(<m_1, 2m_2>) + 1 & \text{if } n \overset{4}{\equiv} 0,\ n/4 = <m_1, m_2>, \\ 2(<m_1, 2m_2 + 1>) + 1 & \text{if } n \overset{4}{\equiv} 0,\ (n-2)/4 = <m_1, m_2>, \\ 4(<m_1, m_2/2>) & \text{if } n \text{ odd, } m_2 \text{ even, } (n-1)/2 = <m_1, m_2> \\ 4(<m_1, (m_2 - 1)/2>) + 2 & \text{if } n \text{ odd, } m_2 \text{ odd, } (n-1)/2 = <m_1, m_2> \end{cases}$$

Note that F is a total function.

8.2 Pfn

Recall that **Pfn** is a PAC and hence is a traced UDC. The trace formula for **Pfn** is the standard sum form as for **PInj**. Clearly $\mathbb{N} \times -$ is an additive and hence a traced functor. Note that the morphisms used in the previous example are partial injective functions, hence partial functions. In view of this we have:

Proposition 8.2.1 $(\mathbf{Pfn}, \mathbb{N} \times -, \mathbb{N})$ *is a GoI Situation.*

The combinators are defined as in the case of **PInj**.

8.3 $\mathbf{Rel}_+$

Recall that the category $\mathbf{Rel}_+$ of sets and binary relations is a PAC and hence a traced UDC. Trace is given by the standard sum. Also, $\mathbb{N} \times -$ is an additive and hence a traced functor. The morphisms used in the case of **PInj** are partial injective functions and hence in particular they are relations. Thus we have:

Proposition 8.3.1 $(\mathbf{Rel}_+, \mathbb{N}\times -, \mathbb{N})$ *is a GoI Situation.*

The combinators are the same as in the case of **PInj**.

8.4 $\mathbf{Hilb}_2$

Recall from Chapter 4 that $\mathbf{Hilb}_2$ is a traced UDC with trace given by the standard sum formula. The tensor product is the direct sum of Hilbert spaces. However, recall that this operation is not a biproduct (or coproduct) in the category $\mathbf{Hilb}_2$. With this proviso we keep using the notation $\oplus$ for this tensor product. In view of the contravariant functor ℓ_2 we have the following result. Let ℓ^2 denote the space $\ell_2(\mathbb{N})$. Recall that this is the Hilbert space of square summable sequences.

Proposition 8.4.1 $(\mathbf{Hilb}_2, \ell^2\otimes -, \ell^2)$ *is a GoI Situation.*

Proof. Clearly the functor $\ell^2\otimes - : \mathbf{Hilb}_2 \longrightarrow \mathbf{Hilb}_2$ is a symmetric monoidal functor. Also, observe that $\ell^2\otimes \ell_2(X)\cong \ell_2(\mathbb{N}\times X)$ and $1_{\ell^2}\otimes \ell_2(f)\cong \ell_2(1_{\mathbb{N}}\times f)$. Moreover, $\ell^2\otimes -$ is an additive and hence a traced functor. This follows from the fact that $\mathbb{N}\times -$ is an additive symmetric monoidal endofunctor on **PInj**.

Also, we have that $\ell^2\oplus\ell^2\cong \ell_2(\mathbb{N}\uplus\mathbb{N})\lhd \ell_2(\mathbb{N})$ $(\ell_2(k), \ell_2(j))$, $\{0\} = \ell_2(\emptyset)\lhd \ell^2$, and finally $\ell^2\otimes\ell^2\cong \ell_2(\mathbb{N}\times\mathbb{N})\lhd \ell_2(\mathbb{N})$ $(\ell_2(v), \ell_2(u))$ for j, k, u, v as in Section 8.1. This proves that ℓ^2 is a reflexive object in $\mathbf{Hilb}_2$.

As for the monoidal natural transformations:

- $\ell^2\otimes(\ell^2\otimes\ell_2(X))\lhd \ell^2\otimes\ell_2(X)$ $(\ell_2(e'_X), \ell_2(e_X))$ for e_X, e'_X as in Section 8.1.
- $\ell_2(X)\lhd \ell^2\otimes\ell_2(X)$ $(\ell_2(l'_X), \ell_2(l_X))$ for l_X, l'_X as in Section 8.1.
- $(\ell^2\otimes\ell_2(X))\oplus(\ell^2\otimes\ell_2(X))\lhd \ell^2\otimes\ell_2(X)$ $(\ell_2(c'_X), \ell_2(c_X))$ for c_X, c'_X as in Section 8.1.
- $\{0\}\lhd \ell^2\otimes\ell_2(X)$ $(\ell_2(w'_X), \ell_2(w_X))$ for w_X, w'_X as in Section 8.1

The naturality of the morphisms above follows from the underlying structure of **PInj** and functoriality of ℓ_2. □

The combinators are simply the ℓ_2 images of those given in Section 8.1.

Remark 8.4.2 The GoI Situation $(\mathbf{Hilb}_2, \ell^2 \otimes -, \ell^2)$ captures the operator algebraic setting used by Girard in [47] (see Section I in [47].) However, there is an important difference between our execution (trace) formula and Girard's execution formula. Recall that Girard's execution formula

$$EX(u, \sigma) = (1 - \sigma^2) \textstyle\sum_{n=0}^{\infty} u(\sigma u)^n (1 - \sigma^2)$$

is a sum that may or may not converge: it converges whenever (σu) is a nilpotent operator. Also, recall from Chapter 5 that strong normalisation was interpreted by nilpotency. On the other hand, our trace formula

$$Tr(f) = f_{11} + \textstyle\sum_n f_{12} f_{22}^n f_{21}$$

always exists. Therefore, the *nilpotency* criterion should be replaced by a different property that will make sense for the latter formula. The author strongly believes in the conjecture:

$$\text{strong normalisation } = \text{ finite sum}$$

The proof of this conjecture and further research into using this criterion to distinguish the *strongly normalising* terms of untyped lambda calculus will be undertaken in a future work.

8.5 SRel

The category **SRel** of stochastic relations is another example of a PAC and hence it is a traced UDC. The trace formula is given by the standard sum.

We define $T : \mathbf{SRel} \longrightarrow \mathbf{SRel}$ as $T(X, \mathcal{F}_X) = (\mathbb{N} \times X, \mathcal{F}_{\mathbb{N}\times X})$ where $\mathcal{F}_{\mathbb{N}\times X}$ is the σ-field on $X \uplus X \uplus X \cdots$ (ω copies). For a given $f : (X, \mathcal{F}_X) \longrightarrow (Y, \mathcal{F}_Y)$,

$Tf((n,x), \biguplus_{i\in\omega} B_i) = f(x, B_n)$. Let $U = \mathbb{N}^\infty$ with the Baire metric which induces a topology on $\mathbb{N}^\infty$ which corresponds to the product topology obtained from $\mathbb{N}$ with discrete topology. The following proposition is due to P. Panangaden and the author.

Proposition 8.5.1 $(\mathbf{SRel}, T, \mathbb{N}^\infty)$ *is a GoI situation.*

Proof. We show that T as defined above is an additive and hence a traced symmetric monoidal functor. First, observe that $T = (T, \psi, \psi_I)$ is a monoidal functor with
$\psi_{X,Y} : ((\mathbb{N}\times X) \uplus (\mathbb{N}\times Y), \mathcal{F}_{(\mathbb{N}\times X)\uplus(\mathbb{N}\times Y)}) \longrightarrow (\mathbb{N}\times(X\uplus Y), \mathcal{F}_{\mathbb{N}\times(X\uplus Y)})$ given by
$\psi_{X,Y}((1,(n,x)), \biguplus_{i\in\omega}(A_i \uplus B_i)) = \delta(x, A_n)$ and

$\psi_{X,Y}((2,(n,y)), \biguplus_{i\in\omega}(A_i \uplus B_i) = \delta(y, B_n)$.

Also, $\psi^{-1}_{X,Y}((n,(1,x)), (\biguplus_{i\in\omega} A_i)\uplus(\biguplus_{i\in\omega} B_i)) = \delta(x, A_n)$ and $\psi^{-1}_{X,Y}((n,(2,y)), (\biguplus_{i\in\omega} A_i)\uplus (\biguplus_{i\in\omega} B_i)) = \delta(y, B_n)$.

Finally, $\psi_I = 1_I$.

Given a summable family $\{f_i\}_{i\in I} \in \mathbf{SRel}(X,Y)$, $\sum_I Tf_i((n,x), \mathbb{N}\times Y) = \sum_I f_i(x, Y) \leq 1$ since $\{f_i\}$ is summable, and hence $\{Tf_i\}$ is summable.

$$\begin{aligned}(\textstyle\sum_I Tf_i)((n,x), \biguplus_{i\in\omega} B_i) &= \textstyle\sum_I Tf_i((n,x), \biguplus_{i\in\omega} B_i)\\ &= \textstyle\sum_I f_i(x, B_n)\\ &= (\textstyle\sum_I f_i)(x, B_n)\\ &= T(\textstyle\sum_I f_i)((n,x), \biguplus_{i\in\omega} B_i)\end{aligned}$$

Hence, T is an additive and by Lemma 8.1.1, a traced functor. We next show that $\mathbb{N}^\infty$ is a reflexive object. We will denote the objects in **SRel** by their first components only in order not to overload the notation.

- $\mathbb{N}^\infty \uplus \mathbb{N}^\infty \lhd \mathbb{N}^\infty \; (j,k)$

  Consider the maps $j : \mathbb{N}^\infty \uplus \mathbb{N}^\infty \longrightarrow \mathbb{N}^\infty$ defined by $j((1,\vec{n}), A) = \delta(0.\vec{n}, A)$, $j((2,\vec{n}), A) = \delta(1.\vec{n}, A)$ and $k : \mathbb{N}^\infty \longrightarrow \mathbb{N}^\infty \uplus \mathbb{N}^\infty$ defined by

$$k(\vec{n}, A_1 \uplus A_2) = \begin{cases} \delta(\vec{n}', A_1), & \text{if } \vec{n} = 0.\vec{n}';\\ \delta(\vec{n}', A_2), & \text{if } \vec{n} = 1.\vec{n}';\\ 0, & \text{else.}\end{cases}$$

$$\begin{aligned}
kj((1,\vec{n}), A_1 \uplus A_2) &= \int_{\mathbb{N}^\infty} k(\vec{n}', A_1 \uplus A_2) j((1,\vec{n}), d\vec{n}') \\
&= \int_{\mathbb{N}^\infty} k(\vec{n}', A_1 \uplus A_2) \delta(0.\vec{n}, d\vec{n}') \\
&= k(0.\vec{n}, A_1 \uplus A_2) \\
&= \delta(\vec{n}, A_1) \\
&= 1_{\mathbb{N}^\infty \uplus \mathbb{N}^\infty}((1,\vec{n}), A_1 \uplus A_2)
\end{aligned}$$

Similarly, $kj((2,\vec{n}), A_1 \uplus A_2) = \delta(\vec{n}, A_2) = 1_{\mathbb{N}^\infty \uplus \mathbb{N}^\infty}((2,\vec{n}), A_1 \uplus A_2)$.

- $\emptyset \lhd \mathbb{N}^\infty$, as $(\emptyset, \mathcal{F}_\emptyset)$ is the zero object.

- $\mathbb{N} \times \mathbb{N}^\infty \longrightarrow \mathbb{N}^\infty$ (u, v)

  u is defined by $u((i,\vec{n}), A) = \delta(i.\vec{n}, A)$ and $v : \mathbb{N}^\infty \longrightarrow \mathbb{N} \times \mathbb{N}^\infty$ by $v(i.\vec{n}, \biguplus_{i\in\omega} A_i) = \delta(\vec{n}, A_i)$.

$$\begin{aligned}
vu((i,\vec{n}), \biguplus_{i\in\omega} A_i) &= \int_{\mathbb{N}^\infty} v(\vec{n}', \biguplus_{i\in\omega} A_i) u((i,\vec{n}), d\vec{n}') \\
&= \int_{\mathbb{N}^\infty} v(\vec{n}', \biguplus_{i\in\omega} A_i) \delta(i.\vec{n}, d\vec{n}') \\
&= v(i.\vec{n}, \biguplus_{i\in\omega} A_i) \\
&= \delta(\vec{n}, A_i) \\
&= 1_{\mathbb{N}\times\mathbb{N}^\infty}((i,\vec{n}), \biguplus_{i\in\omega} A_i).
\end{aligned}$$

Next we consider the monoidal natural transformations.

- $\mathbb{N} \times (\mathbb{N} \times X) \lhd \mathbb{N} \times X$ (e_X, e'_X)

  $e_X((n_1,(n_2,x)), \biguplus_{i\in\omega} A_i) = \delta(x, A_{<n_1,n_2>})$ and $e'_X((n,x), \biguplus_i(\biguplus_j A_{ij})) = \delta(x, A_{n_1 n_2})$

where $n = <n_1, n_2>$.

$$\begin{aligned}
e'_X e_X((n_1,(n_2,x)), \biguplus_{ij} A_{ij}) &= \int_{\mathbb{N}\times X} e'_X((n,x'), \biguplus_{ij} A_{ij}) e_X((n_1,(n_2,x)), d(n,x')) \\
&= \int_{\mathbb{N}\times X} e'_X((n,x'), \biguplus_{ij} A_{ij}) \delta(x, d(<n_1,n_2>, x')) \\
&= e'_X((<n_1,n_2>, x), \biguplus_{ij} A_{ij}) \\
&= \delta(x, A_{n_1 n_2}) \\
&= 1_{\mathbb{N}\times(\mathbb{N}\times X)}((n_1,(n_2,x)), \biguplus_{ij} A_{ij})
\end{aligned}$$

- $X \lhd \mathbb{N} \times X \ (l_X, l'_X)$

  $l_X(x, \biguplus_i A_i) = \delta(x, A_{n_0})$ for a fixed $n_0 \in \mathbb{N}$ and $l'_X((n,x), A) = \delta(x, A)$, if $n = n_0$ and 0, otherwise.

$$\begin{aligned}
l'_X l_X(x, A) &= \int_{\mathbb{N}\times X} l'_X((n,x'), A) l_X(x, d(n,x')) \\
&= \int_{\mathbb{N}\times X} l'_X((n,x'), A) \delta(x, d(n_0, x')) \\
&= l'_X((n_0, x), A) \\
&= \delta(x, A) \\
&= 1_X(x, A).
\end{aligned}$$

- $\mathbb{N} \times X \uplus \mathbb{N} \times X \lhd \mathbb{N} \times X \ (c_X, c'_X)$

  $c_X((1,(n,x)), \biguplus_{i\in\omega} A_i) = \delta(x, A_{2n}),\ c_X((2,(n,x)), \biguplus_i A_i) = \delta(x, A_{2n+1})$ and

$$c'_X((n,x), (\biguplus_i A_i) \uplus (\biguplus_i B_i)) = \begin{cases} \delta(x, A_{n/2}), & \text{if } n \text{ is even;} \\ \delta(x, B_{(n-1)/2}), & \text{if } n \text{ is odd.} \end{cases}$$

$c'_X c_X((1,(n,x)), (\biguplus_i A_i) \uplus (\biguplus_i B_i))$

$= \int_{\mathbb{N}\times X} c'_X((n',x'), (\biguplus_i A_i) \uplus (\biguplus_i B_i)) c_X((1,(n,x)), d(n',x'))$

$= \int_{\mathbb{N}\times X} c'_X((n',x'), (\biguplus_i A_i) \uplus (\biguplus_i B_i)) \delta(x, d(2n, x'))$

$= c'_X((2n,x), (\biguplus_i A_i) \uplus (\biguplus_i B_i)))$

$= \delta(x, A_n)$

$= 1_{(\mathbb{N}\times X)\uplus(\mathbb{N}\times X)}((1,(n,x)), (\biguplus_i A_i) \uplus (\biguplus_i B_i))$

Similarly, $c'_X c_X((2,(n,x)), (\biguplus_i A_i) \uplus (\biguplus_i B_i)) = \delta(x, B_n)$

Therefore, $c'_X c_X = 1_{(\mathbb{N}\times X)\uplus(\mathbb{N}\times X)}$.

- $\emptyset \lhd \mathbb{N} \times X \ (w_X, w'_X)$

  w_X is the empty function and $w'_X((n,x),\emptyset) = 0$. Clearly $w'_X w_X = 1_{(\emptyset,\mathcal{F}_\emptyset)}$.

We show that e_X is natural in X. Let $f : X \longrightarrow Y$,

$(Tf)e_X((n_1,(n_2,x)), \biguplus_{i\in\omega} B_i)$

$= \int_{\mathbb{N}\times X} Tf((n',x'), \biguplus_i B_i) e_X((n_1,(n_2,x)), d(n',x'))$

$= \int_{\mathbb{N}\times X} Tf((n',x'), \biguplus_i B_i) \delta(x, d(<n_1,n_2>, x'))$

$= Tf((<n_1,n_2>, x), \biguplus_i B_i)$

$= f(x, B_{<n_1,n_2>})$

$e_Y(T(Tf))((n_1,(n_2,x)), \biguplus_i B_i)$

$= \int_{\mathbb{N}\times(\mathbb{N}\times Y)} e_Y((n'_1,(n'_2,y)), \biguplus_i B_i) T(Tf)((n_1,(n_2,x)), d(n'_1,(n'_2,y))$

$= \int_{\mathbb{N}\times(\mathbb{N}\times Y)} e_Y((n'_1,(n'_2,y)), \biguplus_i B_i) f(x, d(n_1,(n_2,y)))$

$= \int_{\mathbb{N}\times(\mathbb{N}\times Y)} \delta(y, B_{<n'_1,n'_2>}) f(x, d(n_1,(n_2,y)))$

$= f(x, B_{<n_1,n_2>})$

Therefore, $(Tf)e_X = e_Y(T(Tf))$ for all $f : X \longrightarrow Y$. The naturality of l_X, c_X and w_X follow in a similar way. □

8.6 Some Computations

We consider two examples of computations involving the combinators C and K.

Example 8.6.1

Suppose we would like to evaluate $K{\bullet}x{\bullet}!y$. We first consider
$kKj = \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \begin{bmatrix} j_2k_1^2 + j_1^2k_2 \end{bmatrix} \begin{bmatrix} j_1 & j_2 \end{bmatrix} = \begin{bmatrix} 0 & j_1 \\ k_1 & 0 \end{bmatrix}.$

$K{\bullet}x = Tr\left(\begin{bmatrix} 1 & 0 \\ 0 & x \end{bmatrix} \begin{bmatrix} 0 & j_1 \\ k_1 & 0 \end{bmatrix}\right) = Tr\left(\begin{bmatrix} 0 & j_1 \\ xk_1 & 0 \end{bmatrix}\right)$
$= 0 + \sum_{n=0}^{\infty} j_1 0^n (xk_1) = j_1xk_1.$

Next consider $k(K{\bullet}x)j = \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \begin{bmatrix} j_1xk_1 \end{bmatrix} \begin{bmatrix} j_1 & j_2 \end{bmatrix} = \begin{bmatrix} x & 0 \\ 0 & 0 \end{bmatrix}.$

$K{\bullet}x{\bullet}!y = (K{\bullet}x){\bullet}!y = Tr\left(\begin{bmatrix} 1 & 0 \\ 0 & !y \end{bmatrix} \begin{bmatrix} x & 0 \\ 0 & 0 \end{bmatrix}\right) = Tr\left(\begin{bmatrix} x & 0 \\ 0 & 0 \end{bmatrix}\right) = x + 0 = x.$ □

Example 8.6.2

We next evaluate $C{\bullet}K{\bullet}K$.

$kCj = \begin{bmatrix} 0 & j_1^2k_1^2 + j_2k_2k_1 + j_1j_2k_2 \\ j_1^2k_1^2 + j_1j_2k_2 + j_2k_2k_1 & 0 \end{bmatrix}.$

$C{\bullet}K = Tr\left(\begin{bmatrix} 1 & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} 0 & j_1^2k_1^2 + j_2k_2k_1 + j_1j_2k_2 \\ j_1^2k_1^2 + j_1j_2k_2 + j_2k_2k_1 & 0 \end{bmatrix}\right)$
$= j_1^2k_2k_1 + j_1j_2k_1^2.$

Therefore, $k(C{\bullet}K)j = \begin{bmatrix} j_1k_2 + j_2k_1 & 0 \\ 0 & 0 \end{bmatrix}.$

$(C{\bullet}K){\bullet}K = Tr\left(\begin{bmatrix} 1 & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} j_1k_2 + j_2k_1 & 0 \\ 0 & 0 \end{bmatrix}\right) = j_1k_2 + j_2k_1 = I.$

□

We close this section by a discussion of Abramsky & Jagadeesan's construction in [5]. The construction in [5] uses subcategories of the category **Predom** of ω-complete posets and continuous maps, called $\mathcal{GI}$-categories, to construct a model of linear logic, that is a *-autonomous category with products. Also one defines a functor T_D :**Dom**$\longrightarrow$**Dom**, by $T_D(E) = (D + 1 + (E \times E))_\perp$. Here **Dom** is the category of domains and continuous maps, $((-)_\perp, up, \mu, t)$ is Moggi's [94] strong monad of lifting and $1 = \{\perp\}$ is the terminal object in **Dom**. For every D, the functor T_D has an initial algebra (hence a fixed-object) which is denoted by $(\mathcal{T}D, fold_D)$ where $fold_D : (D + 1 + (\mathcal{T}D \times \mathcal{T}D))_\perp \longrightarrow \mathcal{T}D$ is an isomorphism whose inverse, we call $unfold_D$.

Next, a type-free $\mathcal{GI}$-model is used to give an interpretation of geometry of interaction. A type-free model consists of an object D in a $\mathcal{GI}$-category such that it is a domain and satisfies some retractions analogues to those in Definition 6.3.1 (3).

This construction does not fit into a traced UDC based model. Notice also, that although the category **Dom** is a traced symmetric monoidal category, the trace is not of the canonical form used in traced UDCs; rather, it is induced by the fixed-point operator in **Dom**. However, we can show that (**Dom**$, T, D$) forms a GoI Situation, where D is a type-free $\mathcal{GI}$-model and T :**Dom**$\longrightarrow$**Dom** is defined by $T(A) = \mathcal{T}A$, that is it sends a domain A to the fixed-object $\mathcal{T}A$ of the functor T_A. Given $f : A \longrightarrow B$, $T(f) = fold_B \circ (f + 1 + (\varphi \times \varphi))_\perp \circ unfold_A$ where φ is the unique map from $(\mathcal{T}A, fold_A)$ to $(\mathcal{T}B, \delta)$ (recall that $(\mathcal{T}A, fold_A)$ is an initial algebra.)

The necessary retractions are given in [5] (see pages 66 and 68.) We just need to prove $TT \lhd T\,(e, e')$ (Comultiplication).

Define $e_A : TT(A) \longrightarrow T(A) = \alpha \circ [1, \perp, \perp]_\perp \circ unfold$ and $e'_A : T(A) \longrightarrow TT(A) = fold \circ up \circ in_1$ where $((-)_\perp, up, \mu, t)$ is Moggi's strong monad of lifting and given a domain A, $\alpha_A : A_\perp \longrightarrow A$ is its structure map. It follows easily that e_A is natural in A and $e'_A e_A = 1$.

Chapter 9

Dynamic Algebras

In this chapter we present a categorical model for dynamic algebras and show that Girard's operator algebraic model in [47] and the *small model* introduced by Danos in [33] can be obtained as instances of this general categorical model.

Dynamic algebras were introduced by Danos in his PhD thesis [33]. These are algebraic structures which form the basis for a *path-semantics* of linear logic. The idea in path-semantics is to associate weights from a dynamic algebra, with every path on a proof net. Thus one can view a proof net as a set of paths. The computation of a proof net is represented by the execution formula which takes the form of a sum over all "good" paths. The idea is that the execution formula is an "invariant" of computation and hence the "good" paths are those that survive the cut-elimination process. For exact definitions and details see [36] and [8].

The equational theory of dynamic algebras gives a notion of *normal form* for paths. The cut elimination procedure can then be represented by rewriting rules for the paths that are dictated by the equations of the dynamic algebra. The interesting aspect of this new view is that each path can be reduced to its normal form independently of all the other paths: an *asynchronous* reduction. The reduction procedure also has the property of being *local*: the rewrite rules of the algebra only affect the part of the path that is being reduced. This leads to a local and asynchronous device that can be implemented [84]. An application of this methodology to β-reduction is given in [35]. Essentially, then β-reduction is decomposed into parallel, local and asynchronous microscopic reduction steps. There has been ample amount of work on this issue by Danos and Regnier, for example see [99, 35, 36] and [88] and the references therein. Mackie, in his thesis [84] applies these ideas to develop a basis for efficient, correct implementation of functional programming languages.

We would also like to mention the connections with optimal reduction strategies in λ-calculus. In 1980, Jean-Jacques Lévy [81] introduced a notion of optimal reduction for the λ-calculus. The idea was to use the *least* number of β-reductions in normalising a λ-term. The algorithm to implement such an optimal reduction came only 10 years later in the work of Lamping [79] and Kathail [72]. The work of Gonthier, Abadi and Lévy [53] establishes an interesting connection between optimal reduction and the Geometry of Interaction by observing the strong relation between Lamping's sharing combinators and the notion of path computation in the GoI.

As mentioned earlier the first model of a dynamic algebra, called Λ^* was given by Girard [47]. [1] In fact, Girard's original implementation of GoI programme was based on the C^*-algebra $\mathbb{B}(\ell^2)$ of bounded linear operators on the Hilbert space ℓ^2. He also defined an abstract C^*-algebra, Λ^* using continuous step functions on the Cantor continuum. More specifically, these are functions from the Cantor space $\{0,1\}^{\mathbb{N}}$ with the product topology to a finite set with the discrete topology. However, it is shown in [47] that one can represent this C^*-algebra as a subalgebra of $\mathbb{B}(\ell^2)$, namely those partial isometries that are induced by partial injective functions, i.e. of the form $\ell_2(f)$ with a partial injective f in our terminology. This follows from the Gelfand-Naimark representation theorem [40] which states that any C^*-algebra can be realised as a C^*-subalgebra of $\mathbb{B}(H)$ for some Hilbert space H. We will consider the original $\mathbb{B}(\ell^2)$-based model and show that it follows from our categorical model.

The definition of dynamic algebra given below follows [84, 33].

Definition 9.0.3 A *Dynamic Algebra*, denoted by Λ^*, is a single sorted Σ-algebra (L, Σ). We write x and y for the variables, and $0, 1, p, q, r, s, t, d \in \Sigma$ are the constants of the theory. There is an associative multiplication operator $\cdot : L \times L \longrightarrow L$, (we write xy for $x{\cdot}y$), which has unit 1 and absorbing element 0. The theory is equipped

[1] We will use this notation for a dynamic algebra in general.

with an involution $* : L \longrightarrow L$ and an exponential operator $! : L \longrightarrow L$. The following equations define the properties that we require.

$$\begin{array}{cc} 0^* = !(0) = 0 & 1^* = !(1) = 1 \\ 0x = x0 = 0 & 1x = x1 = x \\ !(x)^* = !(x^*) & (xy)^* = y^*x^* \\ (x^*)^* = x & !(x)!(y) = !(xy) \end{array}$$

Annihilation There are six constants to consider: p and q are the *multiplicative* coefficients, r and s are the *contraction* coefficients, d is the *dereliction* coefficient, and t is the *side door* coefficient. The following are the annihilation equations:

$$\begin{array}{cc} p^*p = q^*q = 1 & q^*p = p^*q = 0 \\ r^*r = s^*s = 1 & s^*r = r^*s = 0 \\ d^*d = 1 & t^*t = 1 \end{array}$$

Communication The exponential coefficients (r, s, t, d) interact with the exponential operator in the following manner.

$$\begin{array}{cc} !(x)r = r!(x) & !(x)s = s!(x) \\ !(x)t = t!!(x) & !(x)d = dx \end{array}$$

The terms built out of this theory are called *monomials*. To write polynomials we need the notion of a sum of monomials. We denote this formal sum by $+$ and give the equations for it.

$$\begin{array}{cc} x + y = y + x & x + 0 = x \\ (x + y)z = (xz + yz) & z(x + y) = (zx + zy) \\ (x + y)^* = x^* + y^* & !(x + y) = !x + !y \end{array}$$

Definition 9.0.4 A *dynamic situation* $\mathcal{S}$ is a tuple $(\mathbb{C}, T, U, (-)^*)$ where

1. $\mathbb{C}$ is a traced UDC,
2. $T : \mathbb{C} \longrightarrow \mathbb{C}$ is an additive functor,

3. $(-)^* : \mathbb{C}^{op} \longrightarrow \mathbb{C}$ is an additive functor which is strictly involutive and the identity on objects,[2]

4. U is an object of $\mathbb{C}$,

such that the following conditions hold:

- $(\mathbb{C}, T, U)$ forms a GoI Situation, with all the retractions of the form (f, f^*),
- $TU \cong U$,
- $T(-)^* = (-)^*T$ and $\iota_i^* = \rho_i$ for all $i \in I$ and finite set I.[3]

Theorem 9.0.5 *A dynamic situation $\mathcal{S}$ is a (sound) model of Λ^*.*

Proof. We let the carrier of the model be $\mathbb{C}(U,U)$. The constants of Λ^* are interpreted as follows: (see Definition 6.3.1, Chapter 6 for the morphisms used below)
$[\![0]\!] = 0_{UU}$ where 0_{UU} is the zero morphism of the Σ-Monoid $\mathbb{C}(U,U)$, $[\![1]\!] = 1_U$, $[\![p]\!] = j_1$, $[\![q]\!] = j_2$, $[\![r]\!] = uc_U\iota_1 v$, $[\![s]\!] = uc_U\iota_2 v$, $[\![t]\!] = ue_U T(v)v$, $[\![d]\!] = ul_U$. $[\![x{\cdot}y]\!] = [\![x]\!][\![y]\!]$, $[\![x^*]\!] = [\![x]\!]^*$, $[\![!x]\!] = uT([\![x]\!])v$ and the formal sum is interpreted by the sum on the homset $\mathbb{C}(U,U)$.

We next verify the equations. Let $u : TU \xrightarrow{\cong} U$ and v denote its inverse. In the following, to simplify the notation we will use the convention of denoting $[\![x]\!]$ by x.
$0^* = 0$, additivity of $(-)^*$ functor.
$!(0) = uT(0)v = 0$, additivity of T.
$1^* = 1$, functoriality of $(-)^*$.
$!(1) = uT(1)v = uv = 1$, functoriality of T and $TU \cong U$.
$0x = x0 = 0$, 0 is the zero morphism.
$1x = x1 = x$, 1 is the identity morphism.
$!(x)^* = (uT(x)v)^* = v^*T(x)^*u^* = uT(x^*)v = !(x^*)$, note that T is a *-functor and $uv = 1_U$ and therefore $u^* = v$ and $v^* = u$.
$(xy)^* = y^*x^*$, $(-)^*$ is a contravariant functor.

[2]A category $\mathbb{C}$ with such a functor is known as a **-category* [42, 3].
[3]Such a functor T is called a **-functor* [42].

$(x^*)^* = x$, $(-)^*$ is an involutive functor.
$!x!y = uT(x)vuT(y)v = uT(xy)v = !(xy)$, $TU \cong U$ and T is a functor.

The annihilation equations:
$p^*p = j_1^*j_1 = k_1j_1 = 1$, $kj = 1_{U\otimes U}$ and hence $k_1j_1 = 1_U$ and $j_1^* = k_1$.
$q^*q = j_2^*j_2 = k_2j_2 = 1$, $kj = 1_{U\otimes U}$ and hence $k_2j_2 = 1_U$ and $j_2^* = k_2$.
$q^*p = j_2^*j_1 = k_2j_1 = 0$, $kj = 1_{U\otimes U}$ and hence $k_2j_1 = 0$.
$p^*q = j_1^*j_2 = k_1j_2 = 0$, $kj = 1_{U\otimes U}$ and hence $k_1j_2 = 0$.
$r^*r = (uc_U\iota_1v)^*(uc_U\iota_1v) = (v^*\iota_1^*c_U^*u^*)(uc_U\iota_1v) = u\rho_1c'_Uvuc_U\iota_1v = 1$, $\iota_1^* = \rho_1$, $c'_Uc_U = 1$, and hence $c_U^* = c'_U$.
$s^*s = (uc_U\iota_2v)^*(uc_U\iota_2v) = (v^*\iota_2^*c_U^*u^*)(uc_U\iota_2v) = u\rho_2c'_Uvuc_U\iota_2v = 1$, $\iota_2^* = \rho_2$.
$s^*r = (uc_U\iota_2v)^*(uc_U\iota_1v) = u\rho_2c'_Uvuc_U\iota_1v = 0$, note that $\rho_2\iota_1 = 0$ by definition.
$r^*s = (uc_U\iota_1v)^*(uc_U\iota_2v) = u\rho_1c'_Uvuc_U\iota_2v = 0$, note that $\rho_1\iota_2 = 0$ by definition.
$d^*d = (ul_U)^*(ul_U) = l'_Uvul_U = 1$, $l'_Ul_U = 1$ and hence $l_U^* = l'_U$.
$t^*t = (ue_UT(v)v)^*(ue_UT(v)v) = uT(u)e'_Uvue_UT(v)v = 1$, $e'_Ue_U = 1$ and hence $e_U^* = e'_U$.

The communication equations:

$$\begin{aligned}
!(x)r &= uT(x)vuc_U\iota_1v \\
&= uc_U(T(x)\otimes T(x))\iota_1v, \text{ naturality of } c_U \\
&= uc_U\iota_1T(x)v, \text{ using the equation } (T(x)\otimes T(x))\iota_1 = \iota_1T(x) \\
&= (uc_U\iota_1v)(uT(x)v), \ vu = 1 \\
&= r!(x).
\end{aligned}$$

$$
\begin{aligned}
!(x)s &= uT(x)vuc_U\iota_2 v \\
&= uc_U(T(x)\otimes T(x))\iota_2 v, \text{ naturality of } c_U \\
&= uc_U\iota_2 T(x)v, \text{ using the equation } (T(x)\otimes T(x))\iota_2 = \iota_2 T(x) \\
&= (uc_U\iota_2 v)(uT(x)v), \ vu = 1 \\
&= s!(x).
\end{aligned}
$$

$$
\begin{aligned}
!(x)t &= uT(x)vue_U T(v)v \\
&= ue_U T(T(x))T(v)v, \text{ naturality of } e_U \\
&= (ue_U T(v)v)(uT(u)T(T(x))T(v)v), \ vu = 1 \\
&= tuT(uT(x)v)v \\
&= tuT(!x)v \\
&= t!!x.
\end{aligned}
$$

$$
\begin{aligned}
!(x)d &= uT(x)vul_U \\
&= uT(x)l_U \\
&= ul_U x, \text{ naturality of } l_U, \\
&= dx.
\end{aligned}
$$

$x+y=y+x$, sum is commutative.
$x+0=x$, zero morphism is the neutral element of sum.
$(x+y)z=(xz+yz)$ and $z(x+y)=(zx+zy)$, composition distributes over sum from left and right.
$(x+y)^* = x^* + y^*$, $(-)^*$ is an additive functor.
$!(x+y) = uT(x+y)v = (uT(x)v)(uT(y)v) =!x!y$, T is an additive functor. □

Note that the formal sum in a dynamic algebra gets interpreted by a partial sum. However, this is enough in all the applications as demonstrated in the case of the *small model* in [88, 36].

The model discussed in the following proposition is known as the *small model*. See [33, 99] for a discussion of the small model.

Proposition 9.0.6 $(\mathbf{PInj}, \mathbb{N} \times -, \mathbb{N}, (-)^{-1})$ *is a dynamic situation.*

Proof. We have already seen that **PInj** is a traced UDC and $\mathbb{N} \times -$ is an additive functor. Also in the previous chapter we showed that $(\mathbf{PInj}, \mathbb{N} \times -, \mathbb{N})$ is a GoI Situation. The retraction morphisms u, v used in $\mathbb{N} \times \mathbb{N} \lhd \mathbb{N}$ are actually isomorphisms. Note that $(-)^{-1}$ is identity on objects and f^{-1} is defined by

$$f^{-1}(y) = x \text{ iff } f(x) = y$$

for $f : X \longrightarrow Y$. Clearly it is involutive. Let $\{f_i\}_I$ be a summable family in $\mathbf{PInj}(X, Y)$, note that the f_i have pairwise disjoint domain and codomain and hence f_i^{-1} will have pairwise disjoint domain and codomain and hence $\{f_i^{-1}\}$ is a summable family in $\mathbf{PInj}(Y, X)$. Next, $(\sum_I f_i)^{-1}(y) = x$ iff $(\sum_I f_i)(x) = y$ iff there exists a $j \in I$ such that $f_j(x) = y$ iff there exists a $j \in I$ such that $f_j^{-1}(y) = x$ iff $(\sum_I f_i^{-1})(y) = x$, for all $y \in Y$ and hence $(-)^{-1}$ is an additive functor.

Also, given $f : X \longrightarrow Y$, $(1_{\mathbb{N}} \times f)^{-1}(n, y) = (n', x)$ iff $(1_{\mathbb{N}} \times f)(n', x) = (n, y)$ iff $n = n'$ and $f(x) = y$ iff $n = n'$ and $f^{-1}(y) = x$ iff $(1_{\mathbb{N}} \times f^{-1})(n, y) = (n', x)$ for all $n \in \mathbb{N}$ and $y \in Y$. Hence $\mathbb{N} \times -$ is a *-functor.

Finally, let $X \lhd Y$ (f, g), then $f^{-1}(y) = x$ iff $f(x) = y$ iff $g(y) = g(f(x)) = x$, for all $y \in Y$, therefore $f^{-1} = g$. We have thus proved something stronger than we need, namely that all retractions are of the form (f, f^{-1}). □

By using $\mathbf{Hilb}_2$ as the underlying category, the functor ℓ_2 and taking the $(-)^*$ to be the adjoint operation we get the following result.

Proposition 9.0.7 $(\mathbf{Hilb}_2, \ell_2(\mathbb{N}) \otimes -, \ell_2(\mathbb{N}), (-)^*)$ *is a dynamic situation.*

Proof. We have already seen that $\mathbf{Hilb}_2$ is a traced UDC and $\ell_2(\mathbb{N}) \otimes -$ is an additive functor. Also in the previous chapter we showed that $(\mathbf{Hilb}_2, \ell_2(\mathbb{N}) \otimes -, \ell_2(\mathbb{N}))$ is a GoI Situation. The retraction morphisms u, v used in $\ell_2(\mathbb{N}) \otimes \ell_2(\mathbb{N}) \lhd \ell_2(\mathbb{N})$ are actually isomorphisms. Recall that given $f : H \longrightarrow K$ a bounded linear map between Hilbert spaces H and K, $f^* : K \longrightarrow H$ is the unique map defined by:

$$<f^*(x), y>=<x, f(y)>$$

for $x \in K$ and $y \in H$.

Note that $(-)^*$ is identity on objects and strictly involutive. We will show that given $\ell_2(f) : \ell_2(Y) \longrightarrow \ell_2(X)$, $\ell_2(f)^* = \ell_2(f^{-1})$ for $f : X \longrightarrow Y$ a partial injective function.

$$\begin{aligned}
<\ell_2(f^{-1})a, b> &= \sum_{y \in Y} \ell_2(f^{-1})(a)(y).\overline{b(y)} \\
&= \sum_{y \in Y} a(f^{-1}(y)).\overline{b(y)} \\
&= \sum_{x \in X} a(x).\overline{b(f(x))} \\
&= \sum_{x \in X} a(x).\overline{\ell_2(f)(b)(x)} \\
&= <a, \ell_2(f)(b)>
\end{aligned}$$

for $a \in \ell_2(X)$ and $b \in \ell_2(Y)$.

Additivity of $(-)^*$ follows from the additivity of ℓ_2 which was established in Chapter 4.

It is a standard result in Hilbert space theory that $1_H^* = 1_H$ and $(f \otimes g)^* = f^* \otimes g^*$ for $f : H \longrightarrow K$ and $g : H' \longrightarrow K'$ and H, K, H', K' Hilbert spaces. Therefore, given $\ell_2(f) : \ell_2(X) \longrightarrow \ell_2(Y)$, $T(\ell_2(f)^*) = 1_{\ell_2(\mathbb{N})} \otimes \ell_2(f)^* = (1_{\ell_2(\mathbb{N})} \otimes \ell_2(f))^* = T(\ell_2(f))^*$. Hence $\ell_2(\mathbb{N}) \otimes -$ is a *-functor.

Finally, let $\ell_2(X) \lhd \ell_2(Y)$ $(\ell_2(f), \ell_2(g))$, then $\ell_2(g)\ell_2(f) = 1$ and hence $fg = 1_X$, since ℓ_2 is a faithful functor. By the result in the previous example, $f^{-1} = g$, and therefore $\ell_2(f)^* = \ell_2(f^{-1}) = \ell_2(g)$.

Note that the latter result also holds in **Hilb**, that is if $gf = 1_H$ for $f : H \longrightarrow K$ and $g : K \longrightarrow H$, then f is an isometric embedding and $g = f^*$. See [16] for a proof. □

This model corresponds to Girard's original operator algebraic model, Λ^*, for a dynamic algebra presented in [47].

Proposition 9.0.8 $(\mathbf{Rel}_+, \mathbb{N} \times -, \mathbb{N}, (-)^{-1})$ *is a dynamic situation.*

Proof. We have already seen that $\mathbf{Rel}_+$ is a traced UDC and $\mathbb{N}\times -$ is an additive functor. Also in the previous chapter we showed that $(\mathbf{Rel}_+, \mathbb{N}\times -, \mathbb{N})$ is a GoI Situation. The retraction morphisms u, v used in $\mathbb{N}\times\mathbb{N}\lhd\mathbb{N}$ are actually isomorphisms. Note that $(-)^{-1}$ is identity on objects and R^{-1} is defined by

$$(y,x)\in R^{-1} \text{ iff } (x,y)\in R$$

for $R : X \longrightarrow Y$. Clearly it is involutive. Let $\{R_i\}_I$ be a summable family in $\mathbf{Rel}_+(X,Y)$, all families in $\mathbf{Rel}_+$ are summable hence, $\{R_i^{-1}\}_I$ is a summable family. Also, $(y,x)\in(\cup_I R_i)^{-1}$ iff $(x,y)\in\cup_I R_i$ iff there exists $j\in I$ such that $(x,y)\in R_j$ iff $(y,x)\in R_j^{-1}$ iff $(y,x)\in\cup_I R_i^{-1}$ Hence $(-)^{-1}$ is an additive functor.

Also given $R : X \longrightarrow Y$, $(n,y,n',x)\in(1_{\mathbb{N}}\times R)^{-1}$ iff $(n',x,n,y)\in 1_{\mathbb{N}}\times R$ iff $n=n'$ and $(x,y)\in R$ iff $n=n'$ and $(y,x)\in R^{-1}$ iff $(n,y,n',x)\in 1_{\mathbb{N}}\times R^{-1}$ for all $n,n'\in\mathbb{N}$, $x\in X$ and $y\in Y$. Hence $(-)^{-1}$ is a *-functor.

Note that it is *not* true in general that given a retraction $X\lhd Y$ (R,S), $R^{-1}=S$. In fact $S\subseteq R^{-1}$. Nevertheless, it will be the case whenever R and S are partial injective functions. Hence we can use the morphisms as in the case of **PInj** for the retractions in the GoI Situation. In fact, we used these morphisms in showing that $(\mathbf{Rel}_+, \mathbb{N}\times -, \mathbb{N})$ is a GoI Situation in the previous chapter. Also, note that quasi injections and projections in $\mathbf{Rel}_+$ are partial injective functions too.

□

We encounter examples (see below) which have the structure of a dynamic situation, save for the existence of a contravariant functor $(-)^*$. It turns out that such examples are also models of a dynamic algebra. However, in these examples we explicitly define an operation $(-)^\bullet$, a *partial* operation on the set of morphisms in the category $\mathbb{C}$ which will be used to interpret $(-)^*$. In the rest of this section we define such structures that we call *weak dynamic situations* and give two examples.

Definition 9.0.9 A *weak dynamic situation* $\mathcal{S}$ is a tuple $(\mathbb{C}, T, U)$ where

1. $\mathbb{C}$ is a traced UDC,

2. $T : \mathbb{C} \longrightarrow \mathbb{C}$ is an additive functor,

3. U is an object of $\mathbb{C}$,

such that the following conditions hold:

- $(\mathbb{C}, T, U)$ forms a GoI Situation,
- $TU \cong U$.

Proposition 9.0.10 *A weak dynamic situation $\mathcal{S}$ is a model of Λ^*.*

Proof. Consider the set $S = \{0_{UU}, 1_U, \iota_1, \iota_2, \rho_1, \rho_2, j_1, j_2, k_1, k_2, u, v, e_U, e'_U, l_U, l'_U, c_U, c'_U\}$. $\iota_i : TU \longrightarrow TU \otimes TU$ $(i = 1, 2)$, $\rho_i : TU \otimes TU \longrightarrow TU$ $(i = 1, 2)$ and the rest of the elements in S are the retraction morphisms of the given GoI Situation. We define an operation $(-)^\bullet$ on the elements of S, as follows: $0^\bullet_{UU} = 0_{UU}, 1^\bullet_U = 1_U, \iota^\bullet_1 = \rho_1, \iota^\bullet_2 = \rho_2, \rho^\bullet_1 = \iota_1, \rho^\bullet_2 = \iota_2, j^\bullet_1 = k_1$, $j^\bullet_2 = k_2, k^\bullet_1 = j_1, k^\bullet_2 = j_2, u^\bullet = v, v^\bullet = u, e^\bullet_U = e'_U, (e'_U)^\bullet = e_U$, $l^\bullet_U = l'_U, (l'_U)^\bullet = l_U, c^\bullet_U = c'_U$ and $(c'_U)^\bullet = c_U$. Clearly $(f^\bullet)^\bullet = f$ for every $f \in S$. The operation $(-)^\bullet$ can be extended to terms (finite words) generated on S using the composition operation, by an inductive definition: $(\alpha f)^\bullet = f^\bullet \alpha^\bullet$ for any term α and any $f \in S$. We also define $T(f)^\bullet = T(f^\bullet)$ for all $f \in S$ and $(\alpha + \beta)^\bullet = \alpha^\bullet + \beta^\bullet$ for all terms α and β provided all the sums exist.

The function $[\![-]\!]$ is defined as in the proof of Proposition 9.0.5, for all the constants and the binary operation $\cdot$, and !. We define $[\![x^*]\!] = [\![x]\!]^\bullet$. It follows that $[\![0^*]\!] = 0_{UU}$, $[\![1^*]\!] = 1_U$, etc. as in the proof of Proposition 9.0.5. Note that $[\![x]\!]$ is a morphism in $\mathbb{C}$ built from the elements in the set S.

To complete the proof, we need to show that the relevant identities hold.

- $[\![0^*]\!] = [\![0]\!]^\bullet = 0^\bullet_{UU} = 0_{UU} = [\![0]\!]$,
- $[\![1^*]\!] = [\![1]\!]^\bullet = 1^\bullet_U = 1_U = [\![1]\!]$,
- $\left[\!\left[(x^*)^* \right]\!\right] = [\![x]\!]$ and $\left[\!\left[(xy)^* \right]\!\right] = \left[\!\left[y^* x^* \right]\!\right]$ follow from the definition of $(-)^\bullet$.
- $\left[\!\left[!(x)^* \right]\!\right] = (uT([\![x]\!])v)^\bullet = v^\bullet T([\![x]\!]^\bullet)u^\bullet = uT([\![x]\!]^\bullet)v = \left[\!\left[!(x^*) \right]\!\right]$,
- The annihilation equations hold, the proof is the same as in Proposition 9.0.5.
- Finally, $\left[\!\left[(x+y)^* \right]\!\right] = ([\![x + y]\!])^\bullet = ([\![x]\!] + [\![y]\!])^\bullet = [\![x]\!]^\bullet + [\![y]\!]^\bullet = [\![x^*]\!] + [\![y^*]\!]$.

$\square$

Proposition 9.0.11 $(\mathbf{Pfn}, \mathbb{N} \times -, \mathbb{N})$ *is a weak dynamic situation.*

Proof. We know that **Pfn** is a PAC and hence a traced UDC and that $T = \mathbb{N} \times -$ is an additive functor.

We have also seen that $(\mathbf{Pfn}, \mathbb{N} \times -, \mathbb{N})$ is a GoI Situation, in Chapter 8. We use the same retraction morphisms as those in Chapter 8. We also have $\mathbb{N} \times \mathbb{N} \cong \mathbb{N}$, Cantor surjective pairing.

□

Let $T : \mathbf{SRel} \longrightarrow \mathbf{SRel}$ be as in Section 8.5.

Proposition 9.0.12 $(\mathbf{SRel}, T, \mathbb{N}^\infty)$ *is a weak dynamic situation.*

Proof. We know that **SRel** is a PAC and hence a traced UDC and that T is an additive functor.

We have also seen that $(\mathbf{SRel}, T, \mathbb{N}^\infty)$ is a GoI Situation. We use the same retraction morphisms as in Section 8.5. We need to show that $T\mathbb{N}^\infty \cong \mathbb{N}^\infty$.

Recall that $u : \mathbb{N} \times \mathbb{N}^\infty \longrightarrow \mathbb{N}^\infty$ was defined by $u((i, \vec{n}), A) = \delta(i.\vec{n}, A)$ and $v(i.\vec{n}, \biguplus_{i\in\omega} A_i) = \delta(\vec{n}, A_i)$. We have already seen in Chapter 8 that $vu = 1_{\mathbb{N}\times\mathbb{N}^\infty}$.

$$
\begin{aligned}
uv(i.\vec{n}, A) &= \int_{\mathbb{N}\times\mathbb{N}^\infty} u((l, \vec{n}'), A)v(i.\vec{n}, d(l, \vec{n}')) \\
&= \int_{\mathbb{N}\times\mathbb{N}^\infty} u((l, \vec{n}'), A)\delta(\vec{n}, d(i, \vec{n}')) \\
&= u((i, \vec{n}), A) \\
&= \delta(i.\vec{n}, A) \\
&= 1_{\mathbb{N}^\infty}(i.\vec{n}, A)
\end{aligned}
$$

Therefore $T\mathbb{N}^\infty \cong \mathbb{N}^\infty$. □

Chapter 10

Full Completeness

In this chapter we construct a new class of models for the multiplicative fragment of linear logic. These models are constructed on partially additive categories using the *Int* construction of Joyal, Street and Verity and the double glueing construction of Hyland and Tan. We prove full completeness for these models. We first discuss briefly the full completeness problem and give a short historical account on the relevant work in the literature.

10.1 Introduction

Completeness theorems establish connections between syntax and semantics of a logic. Traditional completeness theorems are with respect to provability, whereas full completeness is with respect to proofs. This can be best explained in a categorical model [78]. Let $\mathbb{M}$ be a categorical model of the formulas and proofs of a logic $\mathcal{L}$. This means that $\mathbb{M}$ is a category with an appropriate structure such that formulas of $\mathcal{L}$ are interpreted as objects in $\mathbb{M}$ and proofs Π in $\mathcal{L}$ of entailments $A \vdash B$ are interpreted by morphisms $[\![\Pi]\!] : [\![A]\!] \longrightarrow [\![B]\!]$. Finally convertibility of proofs in $\mathcal{L}$ with respect to cut-elimination is soundly modeled by the equations between morphisms holding in $\mathbb{M}$. Traditional completeness theorems assert that $\mathbb{M}([\![A]\!], [\![B]\!]) \neq \emptyset$ implies $A \vdash B$ is provable in the logic $\mathcal{L}$ (= truth implies provability.) Let's see this point in more detail. As is well known, all traditional semantics (Tarski, Kripke, etc.) fit into the categorical logic framework [87]. Formulas are interpreted in the subobject lattices in a model category $\mathcal{M}$: we assume that subobject lattices have sufficient structure to interpret the formulas of the logic $\mathcal{L}$ in the sense of [87]. By the soundness theorem, provability gets interpreted as "$\leq$", i.e. if $A \vdash B$ is provable is $\mathcal{L}$ then $[\![A]\!]_{\mathcal{M}} \leq [\![B]\!]_{\mathcal{M}}$ in the appropriate subobject lattice. Conversely, completeness says that if for all models $\mathcal{M}$, $[\![A]\!]_{\mathcal{M}} \leq [\![B]\!]_{\mathcal{M}}$ then $A \vdash B$ is provable in $\mathcal{L}$. Viewing the subobject lattices as trivial (posetal) categories $\mathbb{M}$, $[\![A]\!]_{\mathcal{M}} \leq [\![B]\!]_{\mathcal{M}}$ means $\mathbb{M}([\![A]\!], [\![B]\!]) \neq \emptyset$. Thus, the completeness theorem reads as: if for all models $\mathbb{M}$, $\mathbb{M}([\![A]\!], [\![B]\!]) \neq \emptyset$, then $A \vdash B$ is provable in $\mathcal{L}$.

Recall from [78] that the formulas of $\mathcal{L}$ are objects in the free category $\mathbb{F}$ on $\mathcal{L}$ and morphisms are equivalence classes of proofs. Also, let's confine our attention to a single model $\mathbb{M}$. We say that $\mathbb{M}$ *is complete for* $\mathcal{L}$ if the unique free functor (with respect to interpretation of the generators) $[\![-]\!] : \mathbb{F} \longrightarrow \mathbb{M}$ is *weakly full.* Recall from [57] that $[\![-]\!]$ is weakly full if $\mathbb{M}([\![A]\!], [\![B]\!]) \neq \emptyset$, then $\mathbb{F}(A, B) \neq \emptyset$ (i.e. $A \vdash B$ is provable in $\mathcal{L}$.) This version of completeness arose in Läuchli's study of Kripke semantics. Note that there is no functorial relation established at this level. The notion of *full completeness* establishes such a functorial connection [103, 26]. We say that $\mathbb{M}$ is *fully complete* for $\mathcal{L}$ if for all formulas A, B of $\mathcal{L}$, every morphism f in $\mathbb{M}([\![A]\!], [\![B]\!])$ is the denotation of some proof Π of $A \vdash B$ in $\mathcal{L}$: $f = [\![\Pi]\!]$. This amounts to asking that the unique free functor (with respect to any interpretation of the generators) $[\![-]\!] : \mathbb{F} \longrightarrow \mathbb{M}$ be full. Here $\mathbb{F}$ is the free category generated by the logic $\mathcal{L}$. Thus, full completeness establishes a tight connection between syntax and semantics compared to completeness. This connection can be made even stronger by requiring that the functor $[\![-]\!]$ be faithful too. In other word, a full faithful completeness theorem asserts that every morphism in $\mathbb{M}([\![A]\!], [\![B]\!])$ is the denotation of a unique proof of $A \vdash B$. It is pointed out in [4], that the idea is related to representation theorems in category theory [39], to full abstraction theorems in programming language semantics [92, 97] and to studies of parametric polymorphism [12, 65].

The term "full completeness" was coined by Abramsky and Jagadeesan in [4] where they also proved full completeness for a game semantics of Multiplicative Linear

Logic with the MIX rule (MLL + MIX). This was followed by a series of papers which established full completeness results for a variety of models with respect to various versions of MLL [64, 26, 27, 82, 83]. Recently Abramsky and Mellies [6] introduced a new concurrent form of game semantics for linear logic and proved a full completeness theorem for Multiplicative-Additive Linear Logic for this semantics. In this thesis we will be mainly concerned with MLL and hence we will not further discuss this latter work. However, we will give more detailed discussion of the work on MLL full completeness theorems.

The idea that dinatural transformations could provide a semantics for proofs of a logical system was first introduced in [12] in the programme called "functorial polymorphism" (see below). In this setting a formula is interpreted by a multivariant functor: this is because atomic propositions can occur both positively and negatively (e.g $A \parr A^{\perp}$.) A dinatural transformation between multivariant functors provides the interpretation for proofs. The problem with dinatural transformations is that they do not compose in general to give a dinatural transformation (more on this below). Girard, Scedrov and P. Scott [51] showed that a dinatural interpretation in the framework of cartesian closed categories is sound with respect to intuitionistic logic without the cut rule. Based on these ideas and results R. Blute and P. Scott [26] proved a full completeness theorem for MLL + MIX using the *-autonomous category **RTVec** of reflexive topological vector spaces. Moreover, as a natural extension of this work they proved a full completeness theorem for the multiplicative fragment of a variant of noncommutative linear logic, Yetter's *cyclic linear logic* (CyLL) with the MIX rule [27]. Hamano [54] has also proven a full completeness proof for MLL+MIX in the *-autonomous category of topological vector spaces. He has been able to eliminate the use of dinatural transformations and instead use a denotational semantics based on the notion of " Z-invariance". See also [55].

There has also been a considerable body of work on full completeness theorems for MLL in *-autonomous categories constructed from compact closed or symmetric monoidal closed categories. Devarajan, Hughes, Plotkin and Pratt [37] prove a full completeness theorem for MLL without MIX interpreted over binary logical transformations of Chu spaces over a two-letter alphabet $\mathbf{Chu}(\mathbf{Set}, \mathbf{2})$. The Chu construction

[14], yields a *-autonomous category from a given symmetric monoidal closed category.

Another important approach to full completeness theorems was introduced in Tan's PhD thesis [104] (summarised in Section 10.4 below.) Hyland and Tan introduced the double glueing construction which given a compact closed category $\mathbb{C}$ constructs a *-autonomous category $\boldsymbol{G}\mathbb{C}$. The setting is the proofs as dinatural transformations paradigm. This work is especially important as it has initiated a systematic approach to the full completeness theorems for such categorical models. Explicitly, Tan defines a *compact closed full completeness* and tries to reduce the full completeness problem for $\boldsymbol{G}\mathbb{C}$ to compact closed full completeness for $\mathbb{C}$. The lifting of compact closed full completeness to $\boldsymbol{G}\mathbb{C}$ establishes the desired full completeness result. Tan studies several examples: $\mathbf{Rel}_\times$, **FDVec**, a category of Conway games and topological vector spaces. She proves full completeness for MLL + MIX in these categories (with the exception of $\mathbf{Rel}_\times$ where she has the result for MLL without MIX). However, the passage from compact closed full completeness to full completeness of $\boldsymbol{G}\mathbb{C}$ is not completely algorithmic, that is each case requires a different treatment.

In this chapter, we will construct categorical models for MLL+MIX based on partially additive categories and prove full completeness theorems for such models. The semantic setting we will be using is the functorial polymorphism of [12]. We approach the full completeness problem in the framework provided by Hyland and Tan that we explained briefly above.

More explicitly, we start with a partially additive category $\mathbb{C}$ and use the Int construction of Joyal, Street and Verity (equivalently the Geometry of Interaction construction $\mathcal{G}$ of Abramsky) that we discussed in Chapter 2 to get a compact closed category $Int\,\mathbb{C}$. We next prove compact closed full completeness for this category. Finally, applying the double glueing construction of Hyland and Tan, we construct a *-autonomous category $\boldsymbol{G}(Int\,\mathbb{C})$, which is a model of MLL + MIX. Finally, we prove full completeness for MLL + MIX in $\boldsymbol{G}(Int\,\mathbb{C})$ by lifting compact closed full completeness in $Int\,\mathbb{C}$. This approach works for all partially additive categories in a uniform way. Also, we show that we get full completeness in traced unique decomposition categories. In this thesis we will only be concerned with full completeness for *unit-free* formulas.

We continue by discussing the functorial polymorphism setting which we will use as our semantic setting.

10.2 Functorial Polymorphism

Functorial polymorphism will be the semantic setting we use in our categorical models. *Functorial polymorphism* introduced in [12], provides a general categorical framework for parametric polymorphic lambda calculus. In this setting, types are represented by multivariant functors and terms by certain multivariant, i.e. *dinatural* transformations. This permits handling such fundamental issues as : (i) simultaneous co- and contravariant (i.e, positive and negative) occurrences of type variables in type expressions, and (ii) uniformity of algorithms intrinsic to polymorphic terms. The theory was extended to linear logic in [23] and examined further in [51, 83]. Applications of this framework for proving full completeness theorems for fragments of linear logic can be found in [26, 27, 55, 37, 6, 104]. The fundamental problem in this framework is that dinatural transformations do not compose in general [26, 6], however the dinats corresponding to proofs do compose.

10.2.1 Dinatural Interpretation for MLL

Definition 10.2.1 Let $\mathbb{C}$ be a category and $F, G : \mathbb{C}^n \times (\mathbb{C}^{op})^n \longrightarrow \mathbb{C}$ be multivariant functors. We write $\underline{X}$ for the list $X_1, X_2, \cdots, X_n$. A *dinatural transformation* $\rho : F \stackrel{..}{\longrightarrow} G$ is a family of $\mathbb{C}$-morphisms $\rho = \{\rho_{\underline{X}} : F(\underline{X}, \underline{X}) \longrightarrow$

$G(\underline{X},\underline{X}) | \underline{X}$ a list of objects in $\mathbb{C}\}$ satisfying (for all $f_i : X_i \longrightarrow Y_i$):

$$\begin{tikzcd}
 & F(\underline{X},\underline{X}) \arrow[r, "\rho_{\underline{X}}"] & G(\underline{X},\underline{X}) \arrow[dr, "F(\underline{f},1_{\underline{X}})"] & \\
F(\underline{X},\underline{Y}) \arrow[ur, "F(1_{\underline{X}},\underline{f})"] \arrow[dr, "F(\underline{f},1_{\underline{Y}})"'] & & & G(\underline{Y},\underline{X}) \\
 & F(\underline{Y},\underline{Y}) \arrow[r, "\rho_{\underline{Y}}"] & G(\underline{Y},\underline{Y}) \arrow[ur, "G(1_{\underline{Y}},\underline{f})"'] &
\end{tikzcd}$$

As was discussed in Chapter 2, a model of MLL consists of a *-autonomous category $\mathbb{C}$. Following the methods of functorial polymorphism, we interpret formulas of MLL as multivariant functors over such a category $\mathbb{C}$, using the operations

$$(F \otimes G)(\underline{A},\underline{B}) = F(\underline{A},\underline{B}) \otimes G(\underline{A},\underline{B})$$

$$F^{\perp}(\underline{A},\underline{B}) = (F(\underline{B},\underline{A}))^{\perp}$$

on n-ary multivariant functors $F, G : \mathbb{C}^n \times (\mathbb{C}^{op})^n \longrightarrow \mathbb{C}$. Here $\underline{A}$ and $\underline{B}$ are lists of objects in $\mathbb{C}$ that occur co- and contravariantly respectively.

Next we give the interpretation for formulas of MLL.

Let $\varphi(\alpha_1, \cdots, \alpha_n)$ be an MLL formula built from the literals $\alpha_1, \cdots, \alpha_n$ and $\alpha_1^{\perp}, \cdots, \alpha_n^{\perp}$. To each such formula we associate its *interpretation* $\llbracket \varphi(\alpha_1, \cdots, \alpha_n) \rrbracket : \mathbb{C}^n \times (\mathbb{C}^{op})^n \longrightarrow \mathbb{C}$ as follows:

1. If $\varphi(\alpha_1, \cdots, \alpha_n) \equiv \alpha_i$, then $\llbracket \varphi \rrbracket(\underline{A},\underline{B}) = A_i$, the covariant projection functor onto the ith component of $\underline{A}$. We denote this functor by Π_i.

2. If $\varphi(\alpha_1, \cdots, \alpha_n) \equiv \alpha_i^{\perp}$, then $\llbracket \varphi \rrbracket(\underline{A},\underline{B}) = B_i^{\perp}$, the linear negation of the contravariant projection onto the ith component of $\underline{B}$, denoted $\Pi_i^{\perp}$.

3. If $\varphi = \varphi_1 \otimes \varphi_2$, then $\llbracket \varphi \rrbracket = \llbracket \varphi_1 \rrbracket \otimes \llbracket \varphi_2 \rrbracket$

4. If $\varphi = \varphi_1^\perp$, then $[\![\varphi]\!] = [\![\varphi_1]\!]^\perp$.

The connective $⅋$ is defined by De Morgan duality. For example,

$$\begin{aligned}
[\![\alpha_1 ⅋ \alpha_2]\!](A_1A_2, B_1B_2) &= \left[\!\!\left[(\alpha_1^\perp \otimes \alpha_2^\perp)^\perp\right]\!\!\right](A_1A_2, B_1B_2) \\
&= \left(\left[\!\!\left[\alpha_1^\perp \otimes \alpha_2^\perp\right]\!\!\right](B_1B_2, A_1A_2)\right)^\perp \\
&= \left(\left[\!\!\left[\alpha_1^\perp\right]\!\!\right](B_1B_2, A_1A_2) \otimes \left[\!\!\left[\alpha_2^\perp\right]\!\!\right](B_1B_2, A_1A_2)\right)^\perp \\
&= \left(\Pi_1^\perp(B_1B_2, A_1A_2) \otimes \Pi_2^\perp(B_1B_2, A_1A_2)\right)^\perp \\
&= (A_1^\perp \otimes A_2^\perp)^\perp \\
&= A_1 ⅋ A_2 \\
&= [\![\alpha_1]\!](A_1A_2, B_1B_2) ⅋ [\![\alpha_2]\!](A_1A_2, B_1B_2)
\end{aligned}$$

We say that a functor is *definable* if it is the interpretation of a formula in the logic or equivalently it is an interpretation of an object in the free category representing the logic. A proof Π of $\vdash \Gamma$ is interpreted as a dinatural transformation from the constant $\mathbf{1}$ functor, $\mathcal{K}_{\mathbf{1}}$, to the multivariant functor $[\![\Gamma]\!]$ similarly to the interpretation in Chapter 2.

In accordance with our definition of interpretation as a functor from the free category (free *-autonomous category in our case) to the model category we note the following remarks.

Remark 10.2.2 A formula $\varphi(\alpha_1, \cdots, \alpha_n)$ in MLL is an object $F(\underline{X}) = F(X_1, \cdots, X_n)$ in the free *-autonomous category $\mathcal{F}^*(\{X_1, \cdots, X_n\})$ generated on n objects (see Section 3 below). $F(\underline{X})$ is built from $X_1, \cdots, X_n$ and $X_1^\perp, \cdots, X_n^\perp$ using tensor and par products.[1] A proof Π of $\vdash \varphi(\alpha_1, \cdots, \alpha_n)$ in MLL is a morphism from the unit of tensor, $\mathbf{1}$, to the object $F(\underline{X})$ in $\mathcal{F}^*(\{X_1, \cdots, X_n\})$ (see Section 3 below.)

Remark 10.2.3 As explained above we interpret formulas of MLL as multivariant functors and the proofs as dinatural transformations. However, multivariant functors and dinatural transformations do not form a category, simply because dinatural

[1] We sometimes use $F(\underline{X}, \underline{X})$ to denote $F(\underline{X})$, in particular when we want to emphasize the functoriality of F.

transformations do not in general compose to give dinatural transformations. However, those dinaturals that are denotations of cut-free proofs in MLL do compose, as first shown in Girard, Scedrov, Scott [51], see also [23].

Ironically, this is the full completeness theorem that we are trying to prove. Moreover, full completeness also establishes the fact that the definable multivariant functors and dinatural transformations form a *-autonomous category [23]. Therefore, nothing in lost by using one sided sequents instead of two sided ones that are used, for example in [83].

10.3 Coherence and Free Monoidal Categories

Recall that a full completeness theorem asserts that a given morphism in the model category is the denotation of a proof of the corresponding formula. In other words, that such a morphism is the denotation of a certain morphism in the free category representing the logic. We therefore need a description of the morphisms in a specified free category. This is known as "the coherence problem" [25]. More precisely, the logical approach to coherence initiated by Lambek in [76] via the equivalence between the deductions in a deductive system and morphisms in a free category, can be used to describe morphisms in a free category. For example, a morphism in the free *-autonomous category (without units), can be interpreted as a proof-net [23]. As proof nets are graphs satisfying a correctness criterion, they may be used to determine the existence of morphisms in various free monoidal categories [25]. We will be dealing with compact closed and *-autonomous categories. Hence, we next discuss the free compact closed and *-autonomous categories generated on a set of objects. The presentation in this section follows [104].

10.3.1 The free compact closed category

The description of the free compact closed category on a given category was given by Kelly and Laplaza [73]. In this section we will look at the free compact closed category $\mathcal{F}(\mathcal{A})$ on a given discrete category: a set of objects $\mathcal{A}$.

The objects of $\mathcal{F}(\mathcal{A})$ are defined as follows:

- All objects in $\mathcal{A}$ are objects in $\mathcal{F}(\mathcal{A})$,
- I is an object in $\mathcal{F}(\mathcal{A})$,
- If A is an object in $\mathcal{F}(\mathcal{A})$, then there is an object A^* in $\mathcal{F}(\mathcal{A})$,
- If A and B are objects in $\mathcal{F}(\mathcal{A})$, then there is an object $A \otimes B$ in $\mathcal{F}(\mathcal{A})$.

In addition to categorical structure, we require associativity morphisms,

$$\alpha_{ABC} : A \otimes (B \otimes C) \longrightarrow (A \otimes B) \otimes C; \quad \alpha^{-1}_{ABC} : (A \otimes B) \otimes C \longrightarrow A \otimes (B \otimes C),$$

morphisms for the unit I,

$$\rho_A : A \otimes I \longrightarrow A; \quad \rho^{-1}_A : A \longrightarrow A \otimes I,$$

a symmetry morphism

$$\sigma_{AB} : A \otimes B \longrightarrow B \otimes A,$$

and unit and counit morphisms

$$\eta_A : I \longrightarrow A \otimes A^*; \quad \epsilon_A : A^* \otimes A \longrightarrow I,$$

for all objects A, B and C in $\mathcal{F}(\mathcal{A})$. We also require that given $f : A \longrightarrow B$ and $g : C \longrightarrow D$ in $\mathcal{F}(\mathcal{A})$, there is a morphism $f \otimes g : A \otimes C \longrightarrow B \otimes D$. Finally, we assert all naturality and coherence equations necessary for compact closed structure.

We will need the following definition.

Definition 10.3.1 For any object A in a traced symmetric monoidal category $\mathbb{C}$ we define the *dimension* of A to be the endomorphism

$$dim(A) = Tr^A_{I,I}(1_I \otimes 1_A) : I \longrightarrow I$$

Tan gives a definition for the dimension of an object in a compact closed category. Our definition is more general and includes Tan's definition: a compact closed category is a traced symmetric monoidal category. For a compact closed category we have:

$$dim(A) : I \xrightarrow{\eta_A} A \otimes A^* \xrightarrow{\sigma_{A,A^*}} A^* \otimes A \xrightarrow{\epsilon_A} I.$$

In $\mathbf{Rel}_\times$, $dim(\emptyset) = 0_I$, the empty relation, and $dim(A) = 1_I$ for all $A \neq \emptyset$. In **FDVec**, $dim(V)$ is the dimension of the vector space V. In a partially additive category, $dim(A) = 1_I = 0_I$ for all A, since I is the zero object. An object A with $dim(A) = 1_I$ is said to have *trivial* dimension. So in a PAC all objects have trivial dimensions. The Kelly-Mac Lane graph [74] of $dim(A)$, for A in a compact closed category is a "loop" passing through A and A^*.

We now describe the morphisms in the free compact closed category $\mathcal{F}(\mathcal{A})$. For this purpose let $\mathcal{F}_1(\mathcal{A})$ be the free compact closed category generated on a set of objects $\mathcal{A} = \{A_1, \cdots, A_n\}$ with trivial dimension. Now suppose $F(A_1, \cdots, A_n)$ and $G(A_1, \cdots, A_n)$ are objects in $\mathcal{F}_1(\mathcal{A})$ built from $A_1, \cdots A_n$, $A_1^*, \cdots, A_n^*$, called *literals* using $\otimes$.

A morphism in $\mathcal{F}_1(\mathcal{A})$ from $F(A_1, \cdots, A_n)$ to $G(A_1, \cdots, A_n)$ is described by pairing the occurrences of literals in the objects (formulas) F and G as follows:

- Each literal occurrence is paired with precisely one other literal occurrence,
- An occurrence of A_i (in F, say) may be paired with either an occurrence of A_i^* in the same formula (F), or with another occurrence of A_i in the other formula (G),
- An occurrence of A_i^* may be paired with either an occurrence of A_i in the same formula, or with another occurrence of A_i^* in the other formula.

Such identifications are just pre or post compositions with unit and counit morphisms. For more details see [104]. Also notice that there are no "loops" in such morphisms because all the generating objects have trivial dimension.

Now a morphism $F(A_1, \cdots, A_n) \longrightarrow G(A_1, \cdots, A_n)$ in $\mathcal{F}(\mathcal{A})$ is a morphisms in $\mathcal{F}_1(\mathcal{A})$ tensored with finitely many maps of the form $dim(A_i) : I \longrightarrow I$.

The idea is that to construct a morphism in a free compact closed category, we first ignore all the loops (by working with objects with trivial dimension) and form all possible basic morphisms and next we add the loops to them (tensoring with $dim(A_i)$.)

10.3.2 The free *-autonomous category

The description of the free *-autonomous category $\mathcal{F}^*(\mathcal{A})$ on a set of objects $\mathcal{A}$ is formulated in a tensor-par setting.

Given a set of objects $\mathcal{A}$, the objects of $\mathcal{F}^*(\mathcal{A})$ are defined as follows:

- All objects in $\mathcal{A}$ are objects in $\mathcal{F}^*(\mathcal{A})$,
- $\mathbf{1}$ is an object in $\mathcal{F}^*(\mathcal{A})$,
- If A is an object in $\mathcal{F}^*(\mathcal{A})$, then there exists an object $A^\perp$ in $\mathcal{F}^*(\mathcal{A})$,
- If A and B are objects in $\mathcal{F}^*(\mathcal{A})$, then there exist objects $A \otimes B$ and $A \parr B$ in $\mathcal{F}^*(\mathcal{A})$.

We also require the associativity, left and right unit and symmetry morphisms, the two weak distribution natural transformations w^L and w^R and the negation families η and ϵ (see Chapter 2). We assert the functoriality of tensor and par, and all naturality and coherence conditions necessary for a *-autonomous structure.

Let $F(\underline{X}) = F(X_1, \cdots, X_n)$ be a unit-free object built from $X_1, \cdots, X_n$ and $X_1^\perp, \cdots, X_n^\perp$ by tensor and par connectives. $F(\underline{X})$ corresponds to a formula in MLL and a proof $\vdash F(\underline{X})$ in MLL has a categorical interpretation in $\mathcal{F}^*(\underline{X})$ as a morphism $\mathbf{1} \longrightarrow F(\underline{X})$ and conversely a morphism $\mathbf{1} \longrightarrow F(\underline{X})$ in $\mathcal{F}^*(\underline{X})$ is the categorical representation of a proof of $\vdash F(\underline{X})$ in MLL. Therefore, MLL proof nets can be regarded as a graphical description of morphisms in the free *-autonomous category [23].

Finally, note that the free *-autonomous category supporting the MIX rule merely requires the addition of the unary MIX morphism $m : \perp \longrightarrow \mathbf{1}$ and the necessary coherence equations (see Chapter 2.)

10.4 A Double Glueing Construction

The double glueing construction we recall here is due to Tan and Hyland. Given a compact closed category, this construction produces a *-autonomous category which makes distinction between tensor and par products. The motivation for this construction lies in the work of Loader [82] on Linear Logical Predicates (LLP). See also Hasegawa [60] for a more abstract treatment and generalisations of glueing construction.

The presentation here follows [104].

Let $\mathbb{C} = (\mathbb{C}, \otimes, I, (-)^*)$ be a compact closed category. Let H denote the covariant hom functor $\mathbb{C}(I, -) : \mathbb{C} \longrightarrow \mathbf{Set}$ and K denote the contravariant functor $\mathbb{C}(-, I) \cong \mathbb{C}(I, (-)^*) : \mathbb{C}^{op} \longrightarrow \mathbf{Set}$.

Define a new category, $\boldsymbol{G}\mathbb{C}$ the glueing category of $\mathbb{C}$, whose objects are triples $\mathcal{A} = (|\mathcal{A}|, \mathcal{A}_s, \mathcal{A}_t)$ where

- $|\mathcal{A}|$ is an object of $\mathbb{C}$
- $\mathcal{A}_s \subseteq H(|\mathcal{A}|) = \mathbb{C}(I, A)$, is a set of *points* of A,
- $\mathcal{A}_t \subseteq K(|\mathcal{A}|) = \mathbb{C}(A, I) \cong \mathbb{C}(I, A^*)$ is a set of *copoints* of A.

A morphism $f : \mathcal{A} \longrightarrow \mathcal{B}$ in $\boldsymbol{G}\mathbb{C}$ is a morphism $f : |\mathcal{A}| \longrightarrow |\mathcal{B}|$ in $\mathbb{C}$ such that $Hf : \mathcal{A}_s \longrightarrow \mathcal{B}_s$ and $Kf : \mathcal{B}_t \longrightarrow \mathcal{A}_t$. Given $f : \mathcal{A} \longrightarrow \mathcal{B}$ and $g : \mathcal{B} \longrightarrow \mathcal{C}$ in $\boldsymbol{G}\mathbb{C}$, the composite $gf : |\mathcal{A}| \longrightarrow |\mathcal{C}|$ is induced by the morphism gf in $\mathbb{C}$. The identity morphism on $\mathcal{A}$ is given by the identity morphism on $|\mathcal{A}|$ in $\mathbb{C}$.

We will denote the underlying object of $\mathcal{A}$ by A, etc. Given objects $\mathcal{A}$ and $\mathcal{B}$ we define the tensor product as follows:

- $| \mathcal{A} \otimes \mathcal{B} | = A \otimes B$

- $(\mathcal{A} \otimes \mathcal{B})_s = \{\sigma \otimes \tau \mid \sigma \in \mathcal{A}_s, \tau \in \mathcal{B}_s\}$,
- $(\mathcal{A} \otimes \mathcal{B})_t = \boldsymbol{G}\mathbb{C}(\mathcal{A}, \mathcal{B}^\perp)$.

where given $\mathcal{A}$, $\mathcal{A}^\perp = (A^*, \mathcal{A}_t, \mathcal{A}_s)$. We define $\mathcal{A} \multimap \mathcal{B} = (\mathcal{A} \otimes \mathcal{B}^\perp)^\perp$ and $\mathcal{A} \parr \mathcal{B} = (\mathcal{A}^\perp \otimes \mathcal{B}^\perp)^\perp$.

Proposition 10.4.1 (Tan) *For any compact closed category $\mathbb{C}$, $\boldsymbol{G}\mathbb{C}$ is a *-autonomous category with tensor $\otimes$ as above and unit $\mathbf{1} = (I, \{id_I\}, \mathbb{C}(I, I))$.*

Remark 10.4.2 Note that $\boldsymbol{G}\mathbb{C}$ is a nontrivial categorical model of MLL. That is, the tensor and par products are always distinct. For example, $(I, \emptyset, \emptyset) \otimes (I, \emptyset, \emptyset) = (I, \emptyset, \mathbb{C}(I, I))$ while $(I, \emptyset, \emptyset) \parr (I, \emptyset, \emptyset) = (I, \mathbb{C}(I, I), \emptyset)$.

Proposition 10.4.3 (Tan) *$\boldsymbol{G}\mathbb{C}$ supports the MIX rule iff $\mathbb{C}(I, I) = \{1_I\}$.*

In a logical setting one can think of an object $\mathcal{A}$, as an object A in $\mathbb{C}$ together with a collection of proofs of A (the collection $\mathcal{A}_s$) and a collection of disproofs or refutations of A (the collection $\mathcal{A}_t$.)

Proposition 10.4.4 (Tan) *The forgetful functor $U : \boldsymbol{G}\mathbb{C} \longrightarrow \mathbb{C}$ preserves the *-autonomous structure of $\boldsymbol{G}\mathbb{C}$. Furthermore, it has a right adjoint $R : \mathbb{C} \longrightarrow \boldsymbol{G}\mathbb{C}$, specified by $RA = (A, \mathbb{C}(I, A), \emptyset)$ and a left adjoint $L : \mathbb{C} \longrightarrow \boldsymbol{G}\mathbb{C}$, specified by $LA = (A, \emptyset, \mathbb{C}(I, A^*))$.*

10.5 Approaching Full Completeness

Based on our earlier discussion, the full completeness problem for MLL in our setting amounts to the following: Given a *-autonomous category $\mathbb{C}$ and a dinatural transformation $\rho : \mathcal{K}_\mathbf{1} \longrightarrow [\![F]\!]$ where $\mathbf{1}$ is the unit of tensor and $[\![F]\!]$ is a definable multivariant functor, we would like to prove that ρ is induced by (is a denotation of) a morphism $\mathbf{1} \longrightarrow F(\underline{X}, \underline{X})$ in the free *-autonomous category on n objects $\{X_1, \cdots, X_n\}$. We will be working with unit-free formulas and thus such a morphism is described by the proof net of the formula F.

The novelties in Tan's thesis included the approach to this problem using model *-autonomous categories which are the glueing of compact closed categories. That is, $\boldsymbol{G}\mathbb{C}$ with $\mathbb{C}$ a compact closed category. Now, there is a forgetful functor $U : \boldsymbol{G}\mathbb{C} \longrightarrow \mathbb{C}$ as we saw in the previous section. The idea is that a dinatural transformation $\rho : \mathcal{K}_{\mathbf{1}} \longrightarrow F$ in $\boldsymbol{G}\mathbb{C}$ induces a dinatural transformation $U\rho : \mathcal{K}_I \longrightarrow UF$ in the underlying compact closed category $\mathbb{C}$ and is completely determined by it. Note that UF simply consists of tensor products. Then, Tan defines full completeness for a compact closed category, in the same way, that is a dinatural $\mathcal{K}_I \longrightarrow F$ must be the denotation of a morphism $I \longrightarrow F(\underline{X}, \underline{X})$ in the free compact closed category on n objects (formal definition to be given later.) Now suppose one can prove compact closed full completeness of $\mathbb{C}$. Then proving full completeness of $\boldsymbol{G}\mathbb{C}$ will be just the lifting of this result to the level of $\boldsymbol{G}\mathbb{C}$. Therefore, the full completeness problem for a certain class of *-autonomous categories (those that are glueings of compact closed categories) is reduced to:

1. Proving full completeness for the underlying compact closed category,
2. Lifting the result to the *-autonomous category.

We proceed by reviewing the necessary formal definitions and theorems from [104].

Definition 10.5.1 Let $\mathbb{C}$ be a compact closed category. Then $\mathbb{C}$ satisfies *compact closed full completeness* if every dinatural transformation $\rho : \mathcal{K}_I \longrightarrow [\![F]\!]$ (with $[\![F]\!] : \mathbb{C}^n \times (\mathbb{C}^{op})^n \longrightarrow \mathbb{C}$), is induced by a morphism $I \longrightarrow F(\underline{X}, \underline{X})$ in the free compact closed category on n objects $X_1, \cdots, X_n$.

Here is a useful observation about dinatural transformations in compact closed categories.

Proposition 10.5.2 (Tan) *Let $\mathbb{C}$ be a compact closed category, let $F : \mathbb{C}^n \times (\mathbb{C}^{op})^n \longrightarrow \mathbb{C}$ be a multivariant functor such that*

$$F(\underline{A}, \underline{A}) \cong A_{\mu_1} \otimes \cdots \otimes A_{\mu_l} \otimes A^*_{\lambda_1} \otimes \cdots \otimes A^*_{\lambda_m},$$

(where $\mu_i, \lambda_i \in \{1, \ldots, n\}$ for all i) and let σ be a collection of morphisms $\sigma_{\underline{A}} : I \longrightarrow F(\underline{A}, \underline{A})$ in $\mathbb{C}$. Define $F^-(\underline{A}) = A_{\lambda_1} \otimes \cdots \otimes A_{\lambda_m}$ and $F^+(\underline{A}) = A_{\mu_1} \otimes \cdots \otimes A_{\mu_l}$, so that each $\sigma_{\underline{A}}$ is canonically equivalent to a morphism $\tilde{\sigma}_{\underline{A}} : F^-(\underline{A}) \longrightarrow F^+(\underline{A})$.

Then, σ is a dinatural transformation in $\mathbb{C}$ iff $\tilde{\sigma}$ is a natural transformation in $\mathbb{C}$.

In view of this observation we can redefine compact closed full completeness as:

> A compact closed category $\mathbb{C}$ satisfies compact closed full completeness if every natural transformation $[\![F^-]\!] \longrightarrow [\![F^+]\!]$ (with $[\![F^-]\!], [\![F^+]\!] : \mathbb{C}^n \longrightarrow \mathbb{C}$), is induced by a morphism $F^-(\underline{X}) \longrightarrow F^+(\underline{X})$ in the free compact closed category on n objects.

Finally we need the essential result that will be used in lifting the full completeness of $\mathbb{C}$ to $\boldsymbol{G}\mathbb{C}$.

Theorem 10.5.3 (Tan) *Suppose that we have a multivariant functor $F : (\boldsymbol{G}\mathbb{C})^n \otimes (\boldsymbol{G}\mathbb{C}^{op})^n \longrightarrow \boldsymbol{G}\mathbb{C}$, such that $\rho : \mathcal{K}_{\mathbf{1}} \longrightarrow F$ is a dinatural transformation in $\boldsymbol{G}\mathbb{C}$. If $\mathcal{A}_1, \ldots, \mathcal{A}_n, \mathcal{B}_1, \ldots, \mathcal{B}_n$, objects in $\boldsymbol{G}\mathbb{C}$ are such that $U\mathcal{A}_i = U\mathcal{B}_i$ for all i, then $U\rho_{\underline{\mathcal{A}}} = U\rho_{\underline{\mathcal{B}}}$.*

This theorem tells us that if ρ is a dinatural transformation then all morphisms $\rho_{\underline{\mathcal{A}}}$ with identical objects $U\mathcal{A}_i$ are completely determined by the same underlying arrow in $\mathbb{C}$, independent of the choice of the sets $(\mathcal{A}_i)_s$ and $(\mathcal{A}_i)_t$. So, the dinatural transformation ρ in $\boldsymbol{G}\mathbb{C}$ is completely determined by the dinatural transformation $U\rho$ in $\mathbb{C}$.

10.6 Full Completeness in GoI Models

We have used traced UDCs to provide a categorical model for the GoI interpretation. In this section we use PACs to construct models of MLL. Recall that PACs are traced UDCs, in particular they are traced symmetric monoidal categories. For any PAC $\mathbb{D}$, $Int\,\mathbb{D}$ is a compact closed category and hence $\boldsymbol{G}(Int\,\mathbb{D})$ is a *-autonomous category. In this way we get a class of models for MLL+MIX. We will show that these models are fully complete for MLL+MIX. Our models support the MIX rule: $Int\,\mathbb{D}(I,I) = Int\,\mathbb{D}((I,I),(I,I)) = \mathbb{D}(I,I) = \{1_I\}$, then use Proposition 10.4.3.

Hereafter, $\mathbb{D}$ denotes a PAC and $\mathbb{C}$ denotes $Int\,\mathbb{D}$.

10.6.1 Compact Closed Full Completeness

Definition 10.6.1 A sequent Γ is *balanced* if each propositional atom α occurs the same number of times as does its linear negation $\alpha^\perp$.

The *length* of a sequent Γ is the number of occurrences of literals in Γ.

If Γ has length p, then we can speak of the position where each literal occurs, numbered 1 to p. If Γ is balanced, and hence p is even, then we can specify the axiom links of a cut-free proof structure associated with Γ by a map $\varphi : \{1, \cdots, p\} \longrightarrow \{1, \cdots, p\}$ such that

- $\varphi(i) \neq i$ for all i, φ is *fixed-point-free*,
- if a propositional atom α occurs in position i, then there is an occurrence of $\alpha^\perp$ in position $\varphi(i)$,
- $\varphi^2(i) = i$ for all i, φ is an *involution.*

Thus a cut-free proof structure can be specified as (Γ, φ), where Γ is a balanced sequent of length p and φ is a fixed-point-free involution on $\{1, \cdots, p\}$ specifying the axiom links.

Let $F(\underline{X}, \underline{X})$ be a formula of length p generated by $X_1, \ldots X_n$, $X_1^* \ldots X_n^*$ using $\otimes$. As we saw earlier in this chapter, $F(\underline{X}, \underline{X})$ induces a multivariant functor $[\![F]\!] : \mathbb{C}^n \times (\mathbb{C}^{op})^n \longrightarrow \mathbb{C}$, which we will refer to as F. Also let $\sigma : \mathcal{K}_I \longrightarrow F$ be a dinatural transformation from constant I functor to F. We can canonically transform σ into a natural transformation $\tilde{\sigma} : F^- \longrightarrow F^+$ (see Section 4). Suppose that the component of σ at $\underline{A}$ is given by $\sigma_{\underline{A}} : I \longrightarrow F(\underline{A}, \underline{A})$ where $F(\underline{A}, \underline{A}) = A_{\xi_1}^{\zeta_1} \otimes \cdots \otimes A_{\xi_p}^{\zeta_p}$ with $\xi_i \in \{1, \cdots, n\}$ and $\zeta_i \in \{1, *\}$, (A_i^1 is read as A_i). Also let $N = \{i | \zeta_i = *\}$ and $P = \{i | \zeta_i = 1\}$. The component of $\tilde{\sigma}$ at $\underline{A}$ is of the form $\tilde{\sigma}_{\underline{A}} : F^-(\underline{A}) \longrightarrow F^+(\underline{A})$

where

$$F^-(\underline{A}) = A_{\lambda_1}\otimes\cdots\otimes A_{\lambda_m} \qquad F^+(\underline{A}) = A_{\mu_1}\otimes\cdots\otimes A_{\mu_l}$$

with $\lambda_i, \mu_i \in \{1, 2, \cdots, n\}$. Therefore $|N| = m$ and $|P| = l$.

Lemma 10.6.2 *Let $\tilde\sigma : F^- \longrightarrow F^+$ be a natural transformation as above. Then each type variable that occurs in F^- must also occur in F^+.*

Proof. Suppose A_i occurs $k > 0$ times in $F^-(\underline{A})$ and does not occur in $F^+(\underline{A})$. That is, there exist $i_1, i_2, \ldots, i_k$ for some $k \le m$ such that $\lambda_{i_1} = \lambda_{i_2} = \ldots = \lambda_{i_k} = i$ for some $i \in \{1, \cdots, n\}$. By instantiating $\tilde\sigma$ at $I\cdots IA_iI\cdots I$ and $I\cdots IB_iI\cdots I$, we get that the following diagram must commute for all $f : A_i \longrightarrow B_i$.

$$\begin{tikzcd}
\overbrace{A_i\otimes\cdots\otimes A_i}^{k} \arrow[r, "\tilde\sigma_{\underline{A}}"] \arrow[d, "f\otimes\cdots\otimes f"'] & I \arrow[d, "1_I"] \\
\underbrace{B_i\otimes\cdots\otimes B_i}_{k} \arrow[r, "\tilde\sigma_{\underline{B}}"] & I
\end{tikzcd}$$

Hence, $\tilde\sigma_{\underline{A}} = \tilde\sigma_{\underline{B}}(f\otimes\cdots\otimes f)$. Here $\tilde\sigma_{\underline{B}}(f\otimes\cdots\otimes f) = Tr^{\otimes_k B_i^-}M$, ($\otimes_k A$ denotes the tensor product of k copies of A), where M is given below and $A_i = (A_i^+, A_i^-)$ and $B_i = (B_i^+, B_i^-)$.

$$\begin{array}{cccccc|c}
A_i^+ & \cdots & A_i^+ & B_i^- & \cdots & B_i^- & \\
\hline
 & & f_{21} & f_{22} & & & A_i^- \\
\mathbf{0} & \iddots & \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \vdots \\
f_{21} & & & & & f_{22} & A_i^- \\
\tilde\sigma_{1,1}f_{11} & \cdots & \tilde\sigma_{1,k}f_{11} & \tilde\sigma_{1,k}f_{12} & \cdots & \tilde\sigma_{1,1}f_{12} & B_i^- \\
\vdots & & \vdots & \vdots & & \vdots & \vdots \\
\tilde\sigma_{k,1}f_{11} & \cdots & \tilde\sigma_{k,k}f_{11} & \tilde\sigma_{k,k}f_{12} & \cdots & \tilde\sigma_{k,1}f_{12} & B_i^-
\end{array}$$

Let f be the morphism with all components zero, i.e. $f_{ij} = 0$ for $i, j = 1, 2$. Then we conclude that $\tilde\sigma_{\underline{A}} = \mathbf{0}$ where $\mathbf{0}$ is the zero matrix. Now, since $\tilde\sigma_{\underline{A}} = \tilde\sigma_{\underline{B}}(f\otimes\cdots\otimes f)$

for all $f : A_i \longrightarrow B_i$, we conclude that $f_{21} = 0$, contradicting the fact that $\tilde{\sigma}$ is a natural transformation. The case of k occurrences of A_i in $F^+(\underline{A})$ and zero occurrence in $F^-(\underline{A})$ is completely symmetric. $\square$

Lemma 10.6.3 *Let $\tilde{\sigma} : F^- \longrightarrow F^+$ be a natural transformation as above. Then each type variable that occurs in F^- must also occur in F^+ with the same multiplicity.*

Proof. Suppose A_i occurs k times in $F^-(\underline{A})$ and $0 < s < k$ times in $F^+(\underline{A})$. That is, there exist $i_1, i_2, \ldots, i_k$ for some $k \leq m$ such that $\lambda_{i_1} = \lambda_{i_2} = \ldots = \lambda_{i_k} = i$. Also suppose that there exist $j_1, j_2, \ldots, j_s$ such that $\mu_{j_1} = \mu_{j_2} = \ldots = \mu_{j_s} = i$. By instantiating $\tilde{\sigma}$ at $I \cdots I A_i I \cdots I$ and $I \cdots I B_i I \cdots I$ we get that the following diagram must commute for all $f : A_i \longrightarrow B_i$.

$$\begin{tikzcd}
\overbrace{A_i \otimes \cdots \otimes A_i}^{k} \arrow[r, "\tilde{\sigma}_{\underline{A}}"] \arrow[d, "f \otimes \cdots \otimes f"'] & \overbrace{A_i \otimes \cdots \otimes A_i}^{s} \arrow[d, "f \otimes \cdots \otimes f"] \\
\underbrace{B_i \otimes \cdots \otimes B_i}_{k} \arrow[r, "\tilde{\sigma}_{\underline{B}}"'] & \underbrace{B_i \otimes \cdots \otimes B_i}_{s}
\end{tikzcd}$$

Here $(f \otimes \cdots \otimes f)\tilde{\sigma}_{\underline{A}} = Tr^{\otimes_s A_i^-} M$ and $\tilde{\sigma}_{\underline{B}}(f \otimes \cdots \otimes f) = Tr^{\otimes_k B_i^-} M'$ with matrices M and M' given below. Note that $\tilde{\sigma}_{\underline{A}}$ and $\tilde{\sigma}_{\underline{B}}$ have the same components as $\tilde{\sigma}$ is a natural transformation.

M:

$$
\begin{array}{ccccccccc|c}
A_i^+ & \cdots & A_i^+ & B_i^- & \cdots & B_i^- & A_i^- & \cdots & A_i^- & \\
\hline
f_{11}\tilde{\sigma}_{1,1} & \cdots & f_{11}\tilde{\sigma}_{1,k} & & & f_{12} & f_{11}\tilde{\sigma}_{1,k+1} & \cdots & f_{11}\tilde{\sigma}_{1,k+s} & B_i^+ \\
\vdots & & \vdots & \mathbf{0} & \iddots & \mathbf{0} & \vdots & & \vdots & \vdots \\
f_{11}\tilde{\sigma}_{s,1} & \cdots & f_{11}\tilde{\sigma}_{s,k} & f_{12} & & & f_{11}\tilde{\sigma}_{s,k+1} & \cdots & f_{11}\tilde{\sigma}_{s,k+s} & B_i^+ \\
\tilde{\sigma}_{s+1,1} & \cdots & \tilde{\sigma}_{s+1,k} & & & & \tilde{\sigma}_{s+1,k+1} & \cdots & \tilde{\sigma}_{s+1,k+s} & A_i^- \\
\vdots & & \vdots & & \mathbf{0} & & \vdots & & \vdots & \vdots \\
\tilde{\sigma}_{s+k,1} & \cdots & \tilde{\sigma}_{s+k,k} & & & & \tilde{\sigma}_{s+k,k+1} & \cdots & \tilde{\sigma}_{s+k,k+s} & A_i^- \\
f_{21}\tilde{\sigma}_{s,1} & \cdots & f_{21}\tilde{\sigma}_{s,k} & f_{22} & & & f_{21}\tilde{\sigma}_{s,k+1} & \cdots & f_{21}\tilde{\sigma}_{s,k+s} & A_i^- \\
\vdots & & \vdots & \mathbf{0} & \ddots & \mathbf{0} & \vdots & & \vdots & \vdots \\
f_{21}\tilde{\sigma}_{1,1} & \cdots & f_{21}\tilde{\sigma}_{1,k} & & & f_{22} & f_{21}\tilde{\sigma}_{1,k+1} & \cdots & f_{21}\tilde{\sigma}_{1,k+s} & A_i^-
\end{array}
$$

M':

$$
\begin{array}{ccccccccc|c}
A_i^+ & \cdots & A_i^+ & B_i^- & \cdots & B_i^- & B_i^- & \cdots & B_i^- & \\
\hline
\tilde{\sigma}_{1,1}f_{11} & \cdots & \tilde{\sigma}_{1,k}f_{11} & \tilde{\sigma}_{1,k+1} & \cdots & \tilde{\sigma}_{1,k+s} & \tilde{\sigma}_{1,k}f_{12} & \cdots & \tilde{\sigma}_{1,1}f_{12} & B_i^+ \\
\vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots \\
\tilde{\sigma}_{s,1}f_{11} & \cdots & \tilde{\sigma}_{s,k}f_{11} & \tilde{\sigma}_{s,k+1} & \cdots & \tilde{\sigma}_{s,k+s} & \tilde{\sigma}_{s,k}f_{12} & \cdots & \tilde{\sigma}_{s,1}f_{12} & B_i^+ \\
 & & f_{21} & & & & f_{22} & & & A_i^- \\
\mathbf{0} & \iddots & \mathbf{0} & & \mathbf{0} & & \mathbf{0} & \ddots & \mathbf{0} & \vdots \\
f_{21} & & & & & & & & f_{22} & A_i^- \\
\tilde{\sigma}_{s+1,1}f_{11} & \cdots & \tilde{\sigma}_{s+1,k}f_{11} & \tilde{\sigma}_{s+1,k+1} & \cdots & \tilde{\sigma}_{s+1,k+s} & \tilde{\sigma}_{s+1,k}f_{12} & \cdots & \tilde{\sigma}_{s+1,1}f_{12} & B_i^- \\
\vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots \\
\tilde{\sigma}_{s+k,1}f_{11} & \cdots & \tilde{\sigma}_{s+k,k}f_{11} & \tilde{\sigma}_{s+k,k+1} & \cdots & \tilde{\sigma}_{s+k,k+s} & \tilde{\sigma}_{s+k,k}f_{12} & \cdots & \tilde{\sigma}_{s+k,1}f_{12} & B_i^-
\end{array}
$$

Let f be the morphism with all components zero, then from the equation $TrM = TrM'$ above with

$$
M = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \boldsymbol{B}_4 & \mathbf{0} & \boldsymbol{B}_6 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \qquad M' = \begin{bmatrix} \mathbf{0} & \boldsymbol{B}'_2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{B}'_8 & \mathbf{0} \end{bmatrix}
$$

we get $\boldsymbol{B}_4 = \mathbf{0}$ and $\boldsymbol{B}'_2 = \mathbf{0}$ and hence $\tilde{\sigma}_{\underline{A}}$ is of the form $\begin{bmatrix} \boldsymbol{C}_1 & \mathbf{0}_{ss} \\ \mathbf{0}_{kk} & \boldsymbol{C}_4 \end{bmatrix}$ where

$$\boldsymbol{C}_1 = \begin{bmatrix} \tilde{\sigma}_{1,1} & \tilde{\sigma}_{1,2} & \cdots & \tilde{\sigma}_{1,k} \\ \vdots & \vdots & \cdots & \vdots \\ \tilde{\sigma}_{s-1,1} & \tilde{\sigma}_{s-1,2} & \cdots & \tilde{\sigma}_{s-1,k} \\ \tilde{\sigma}_{s,1} & \tilde{\sigma}_{s,2} & \cdots & \tilde{\sigma}_{s,k} \end{bmatrix} \text{ and } \boldsymbol{C}_4 = \begin{bmatrix} \tilde{\sigma}_{s+1,k+1} & \tilde{\sigma}_{s+1,k+2} & \cdots & \tilde{\sigma}_{s+1,k+s} \\ \tilde{\sigma}_{s+2,k+1} & \tilde{\sigma}_{s+2,k+2} & \cdots & \tilde{\sigma}_{s+2,k+s} \\ \vdots & \vdots & \cdots & \vdots \\ \tilde{\sigma}_{s+k,k+1} & \tilde{\sigma}_{s+k,k+2} & \cdots & \tilde{\sigma}_{s+k,k+s} \end{bmatrix}$$

Note that $A_i = (A_i^+, A_i^-)$ and $B_i = (B_i^+, B_i^-)$ for A_i^+, A_i^-, B_i^+ and B_i^- objects in $\mathbb{D}$. Now let $A_i^+ = A_i^-$ and $B_i^+ = B_i^-$ and f be the twist, that is $f_{11} = f_{22} = 0$ and $f_{12} = f_{21} = 1$. We get the following equations from $Tr(M) = Tr(M')$.

$$\begin{array}{lcl} \tilde{\sigma}_{s+1,k+1}\tilde{\sigma}_{s,1} + \cdots + \tilde{\sigma}_{s+1,k+s}\tilde{\sigma}_{1,1} & = & 0 \\ \vdots & & \vdots \\ \tilde{\sigma}_{s+1,k+1}\tilde{\sigma}_{s,k-1} + \cdots + \tilde{\sigma}_{s+1,k+s}\tilde{\sigma}_{1,k-1} & = & 0 \\ \tilde{\sigma}_{s+1,k+1}\tilde{\sigma}_{s,k} + \cdots + \tilde{\sigma}_{s+1,k+s}\tilde{\sigma}_{1,k} & = & 1 \\ \vdots & & \vdots \\ \tilde{\sigma}_{s+k,k+1}\tilde{\sigma}_{s,1} + \cdots + \tilde{\sigma}_{s+k,k+s}\tilde{\sigma}_{1,1} & = & 1 \\ \vdots & & \vdots \\ \tilde{\sigma}_{s+k,k+1}\tilde{\sigma}_{s,k} + \cdots + \tilde{\sigma}_{s+k,k+s}\tilde{\sigma}_{1,k} & = & 0 \end{array}$$

We can express these equations in a matricial form which we will call a (k, s)-*system*:

$$N_{k\times s}N'_{s\times k} = \begin{bmatrix} \tilde{\sigma}_{s+1,k+1} & \tilde{\sigma}_{s+1,k+2} & \cdots & \tilde{\sigma}_{s+1,k+s} \\ \tilde{\sigma}_{s+2,k+1} & \tilde{\sigma}_{s+2,k+2} & \cdots & \tilde{\sigma}_{s+2,k+s} \\ \vdots & \vdots & \cdots & \vdots \\ \tilde{\sigma}_{s+k,k+1} & \tilde{\sigma}_{s+k,k+2} & \cdots & \tilde{\sigma}_{s+k,k+s} \end{bmatrix} \begin{bmatrix} \tilde{\sigma}_{s,1} & \tilde{\sigma}_{s,2} & \cdots & \tilde{\sigma}_{s,k} \\ \tilde{\sigma}_{s-1,1} & \tilde{\sigma}_{s-1,2} & \cdots & \tilde{\sigma}_{s-1,k} \\ \vdots & \vdots & \cdots & \vdots \\ \tilde{\sigma}_{1,1} & \tilde{\sigma}_{1,2} & \cdots & \tilde{\sigma}_{1,k} \end{bmatrix}$$

$$= antidiag(1, \cdots, 1)$$

In other words $R_i^N C_j^{N'} = 1$ for $j = k - i + 1$ and 0 else, for $i = 1, \cdots, k$. We use R_i^N and C_j^N to denote the ith row and the jth column of N respectively.

This gives us $k \times k$ equations each having s summands. Solutions to such equations if they exist are 0 or 1 because the instantiation of $\tilde{\sigma}$ at $I \cdots I$ yields matrices with entries in $\mathbb{D}(I, I) = \{1_I\} = \{0_I\}$. Instantiation at other objects will distinguish between 0's and 1's. We will prove by induction that this system of equations is inconsistent. First, consider a $(2,1)$-system, that is $k = 2$ and $s = 1$. Then we will have the following set of equations:

$$\begin{aligned}\tilde{\sigma}_{2,3}\tilde{\sigma}_{1,1} &= 0\\ \tilde{\sigma}_{3,3}\tilde{\sigma}_{1,2} &= 0\\ \tilde{\sigma}_{2,3}\tilde{\sigma}_{1,2} &= 1\\ \tilde{\sigma}_{3,3}\tilde{\sigma}_{1,1} &= 1\end{aligned}$$

Clearly this set of equations is inconsistent since from the third and the fourth equations, we conclude that $\tilde{\sigma}_{2,3} = \tilde{\sigma}_{1,2} = \tilde{\sigma}_{1,1} = \tilde{\sigma}_{3,3} = 1$ contradicting the first and the second equations. We refer to these equations as the *Base System* and we say that the tuple $(\tilde{\sigma}_{1,1}, \tilde{\sigma}_{1,2}, \tilde{\sigma}_{2,3}, \tilde{\sigma}_{3,3})$ satisfies the Base System.

Consider a $(k,1)$-system with $k > 1$. These systems form the basis step of the induction. The equations are given by:

$$\begin{bmatrix} \tilde{\sigma}_{s+1,k+1} \\ \tilde{\sigma}_{s+2,k+1} \\ \vdots \\ \tilde{\sigma}_{s+k,k+1} \end{bmatrix} \begin{bmatrix} \tilde{\sigma}_{s,1} & \tilde{\sigma}_{s,2} & \cdots & \tilde{\sigma}_{s,k} \end{bmatrix} = antidiag(1, 1, \cdots, 1)$$

Therefore, we get $\tilde{\sigma}_{s+1,k+1}\tilde{\sigma}_{s,k} = 1, \tilde{\sigma}_{s+2,k+1}\tilde{\sigma}_{s,k-1} = 1, \tilde{\sigma}_{s+1,k+1}\tilde{\sigma}_{s,k-1} = 0$ and $\tilde{\sigma}_{s+2,k+1}\tilde{\sigma}_{s,k} = 0$ and hence $(\tilde{\sigma}_{s,k-1}, \tilde{\sigma}_{s,k}, \tilde{\sigma}_{s+1,k+1}, \tilde{\sigma}_{s+2,k+1})$ satisfies the base system and hence the $(k,1)$-system is inconsistent for all $k > 1$.

Now consider a general (k,s)-system with $k > s$ and $s > 1$. There are k equations with RHS equal to 1. Consider the equation $R_1^N C_k^{N'} = 1$, more explicitly

$$\tilde{\sigma}_{s+1,k+1}\tilde{\sigma}_{s,k} + \tilde{\sigma}_{s+1,k+2}\tilde{\sigma}_{s-1,k} + \cdots + \tilde{\sigma}_{s+1,k+s}\tilde{\sigma}_{1,k} = 1$$

we will prove that none of the s summands in this equation can be 1 and hence they are all zero but then we get $0 + 0 + \cdots + 0 = 0 = 1$ a contradiction and hence the (k,s)-system will be inconsistent. Suppose $\tilde{\sigma}_{s+1,k+t}\tilde{\sigma}_{s-t+1,k} = 1$ for an arbitrary $t \in \{1,\cdots,s\}$, we show that $\tilde{\sigma}_{s+j,k+t}\tilde{\sigma}_{s-t+1,k-j+1} = 0$ for all $j = 2,\cdots,k$. Let $\tilde{\sigma}_{s+j,k+t}\tilde{\sigma}_{s-t+1,k-j+1} = 1$ for some $j \in \{2,\cdots,k\}$, then using equations $R_1^N C_{k-j+1}^{N'} = 0$ and $R_j^N C_k^{N'} = 0$ we have $\tilde{\sigma}_{s+1,k+t}\tilde{\sigma}_{s-t+1,k-j+1} = 0$ and $\tilde{\sigma}_{s+j,k+t}\tilde{\sigma}_{s-t+1,k} = 0$ and therefore the tuple $(\tilde{\sigma}_{s-t+1,k-j+1}, \tilde{\sigma}_{s-t+1,k}, \tilde{\sigma}_{s+1,k+t}, \tilde{\sigma}_{s+j,k+t})$ satisfies the base system. Removing R_1^N and C_t^N from matrix N and $R_t^{N'}$ and $C_k^{N'}$ from matrix N' we get two new matrices $L_{k'\times s'}$ and $L'_{s'\times k'}$ where $k' = k-1$ and $s' = s-1$. Note that $R_i^L C_j^{L'} = 1$ for $j = k'-i+1$ and 0 else, for $i = 1,\cdots,k'$, because $R_{i+1}^N C_j^{N'} = 1$ for $j = k-i$ and 0 else, for $i = 1,\cdots,k-1$, thus L and L' form a $(k-1,s-1)$-system which by induction hypothesis is inconsistent and hence we conclude that $\tilde{\sigma}_{s+1,k+t}\tilde{\sigma}_{s-t+1,k} = 0$. As t was chosen arbitrarily we conclude that all the summands in the equation $R_1^N C_k^{N'}$ are zero. The case of $0 < k < s$ is completely symmetric. □

Proposition 10.6.4 *Let $F(\underline{X},\underline{X})$ be an MLL formula of length p generated by $X_1,\ldots X_n$, $X_1^* \ldots X_n^*$ using $\otimes$. Let $[\![F]\!] : \mathbb{C}^n \times (\mathbb{C}^{op})^n \longrightarrow \mathbb{C}$ be the induced multivariant functor on $\mathbb{C}$. If $\sigma : \mathcal{K}_I \longrightarrow [\![F]\!]$ is a dinatural transformation, then F is balanced.*

Proof. In Lemma 10.6.2 above we have proven that every propositional atom occurring on one side also occurs on the other side. Also in Lemma 10.6.3 we have shown that the multiplicity of occurrences of any propositional atom must be the same on both sides. Hence it follows that F is balanced.

□

Theorem 10.6.5 *$\tilde{\sigma} : F^- \longrightarrow F^+$ is a permutation on the tensor factors, i.e. $\tilde{\sigma}_{\underline{A}} : F^-(\underline{A}) \longrightarrow F^+(\underline{A})$ is of the form $\begin{bmatrix} \boldsymbol{B}_1 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{B}_4 \end{bmatrix}$ where $\boldsymbol{B}_1$ and $\boldsymbol{B}_4$ are permutation matrices and $\boldsymbol{B}_4 = \boldsymbol{B}_1^t$ and the permutation $\delta \in S_m$ induced by $\boldsymbol{B}_1$ satisfies $\mu_i = \lambda_{\delta(i)}$ for $i = 1,\cdots,m$. Here $(-)^t$ denotes matrix transposition obtained by reflection across* antidiagonal *elements.*

Proof. Let $\tilde{\sigma} : F^- \longrightarrow F^+$ be a natural transformation in $\mathbb{C}$, that is the following diagram commutes:

$$
\begin{tikzcd}
A_{\lambda_1} \otimes \cdots \otimes A_{\lambda_m} \arrow[r, "\tilde{\sigma}_{\underline{A}}"] \arrow[d, "f_{\lambda_1} \otimes \cdots \otimes f_{\lambda_m}"'] & A_{\mu_1} \otimes \cdots \otimes A_{\mu_m} \arrow[d, "f_{\mu_1} \otimes \cdots \otimes f_{\mu_m}"] \\
B_{\lambda_1} \otimes \cdots \otimes B_{\lambda_m} \arrow[r, "\tilde{\sigma}_{\underline{B}}"] & B_{\mu_1} \otimes \cdots \otimes B_{\mu_m}
\end{tikzcd}
$$

Hence, we have $Tr^{A^-_{\mu_m} \otimes \cdots \otimes A^-_{\mu_1}}(M) = (f_{\mu_1} \otimes \cdots \otimes f_{\mu_m})\tilde{\sigma}_{\underline{A}} = \tilde{\sigma}_{\underline{B}}(f_{\mu_1} \otimes \cdots \otimes f_{\mu_m}) = Tr^{B^-_{\lambda_m} \otimes \cdots \otimes B^-_{\lambda_1}}(M')$ where M and M' are given below.

M:

$$
\begin{array}{ccccccccc|c}
A^+_{\lambda_1} & \cdots & A^+_{\lambda_m} & B^-_{\mu_m} & \cdots & B^-_{\mu_1} & A^-_{\mu_m} & \cdots & A^-_{\mu_1} & \\
\hline
f^{\mu_1}_{11}\tilde{\sigma}_{1,1} & \cdots & f^{\mu_1}_{11}\tilde{\sigma}_{1,m} & & & f^{\mu_1}_{12} & f^{\mu_1}_{11}\tilde{\sigma}_{1,m+1} & \cdots & f^{\mu_1}_{11}\tilde{\sigma}_{1,2m} & B^+_{\mu_1} \\
\vdots & & \vdots & \mathbf{0} & \iddots & \mathbf{0} & \vdots & & \vdots & \vdots \\
f^{\mu_m}_{11}\tilde{\sigma}_{m,1} & \cdots & f^{\mu_m}_{11}\tilde{\sigma}_{m,m} & f^{\mu_m}_{12} & & & f^{\mu_m}_{11}\tilde{\sigma}_{m,m+1} & \cdots & f^{\mu_m}_{11}\tilde{\sigma}_{m,2m} & B^+_{\mu_m} \\
\tilde{\sigma}_{m+1,1} & \cdots & \tilde{\sigma}_{m+1,m} & & & & \tilde{\sigma}_{m+1,m+1} & \cdots & \tilde{\sigma}_{m+1,2m} & A^-_{\lambda_m} \\
\vdots & & \vdots & & \mathbf{0} & & \vdots & & \vdots & \vdots \\
\tilde{\sigma}_{2m,1} & \cdots & \tilde{\sigma}_{2m,m} & & & & \tilde{\sigma}_{2m,m+1} & \cdots & \tilde{\sigma}_{2m,2m} & A^-_{\lambda_1} \\
f^{\mu_m}_{21}\tilde{\sigma}_{m,1} & \cdots & f^{\mu_m}_{21}\tilde{\sigma}_{m,m} & f^{\mu_m}_{22} & & & f^{\mu_m}_{21}\tilde{\sigma}_{m,m+1} & \cdots & f^{\mu_m}_{21}\tilde{\sigma}_{m,2m} & A^-_{\mu_m} \\
\vdots & & \vdots & \mathbf{0} & \ddots & \mathbf{0} & \vdots & & \vdots & \vdots \\
f^{\mu_1}_{21}\tilde{\sigma}_{1,1} & \cdots & f^{\mu_1}_{21}\tilde{\sigma}_{1,m} & & & f^{\mu_1}_{22} & f^{\mu_1}_{21}\tilde{\sigma}_{1,m+1} & \cdots & f^{\mu_1}_{21}\tilde{\sigma}_{1,2m} & A^-_{\mu_1}
\end{array}
$$

M':

$$
\begin{array}{ccccccccc|c}
A^+_{\lambda_1} & \cdots & A^+_{\lambda_m} & B^-_{\mu_m} & \cdots & B^-_{\mu_1} & B^-_{\lambda_m} & \cdots & B^-_{\lambda_1} & \\
\hline
\tilde{\sigma}_{1,1}f^{\lambda_1}_{11} & \cdots & \tilde{\sigma}_{1,m}f^{\lambda_m}_{11} & \tilde{\sigma}_{1,m+1} & \cdots & \tilde{\sigma}_{1,2m} & \tilde{\sigma}_{1,m}f^{\lambda_m}_{12} & \cdots & \tilde{\sigma}_{1,1}f^{\lambda_1}_{12} & B^+_{\mu_1} \\
\vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots \\
\tilde{\sigma}_{m,1}f^{\lambda_1}_{11} & \cdots & \tilde{\sigma}_{m,m}f^{\lambda_m}_{11} & \tilde{\sigma}_{m,m+1} & \cdots & \tilde{\sigma}_{m,2m} & \tilde{\sigma}_{m,m}f^{\lambda_m}_{12} & \cdots & \tilde{\sigma}_{m,1}f^{\lambda_1}_{12} & B^+_{\mu_m} \\
 & & f^{\lambda_m}_{21} & & & & f^{\lambda_m}_{22} & & & A^-_{\lambda_m} \\
\mathbf{0} & \iddots & \mathbf{0} & & \mathbf{0} & & \mathbf{0} & \ddots & \mathbf{0} & \vdots \\
f^{\lambda_1}_{21} & & & & & & & & f^{\lambda_1}_{22} & A^-_{\lambda_1} \\
\tilde{\sigma}_{m+1,1}f^{\lambda_1}_{11} & \cdots & \tilde{\sigma}_{m+1,m}f^{\lambda_m}_{11} & \tilde{\sigma}_{m+1,m+1} & \cdots & \tilde{\sigma}_{m+1,2m} & \tilde{\sigma}_{m+1,m}f^{\lambda_m}_{12} & \cdots & \tilde{\sigma}_{m+1,1}f^{\lambda_1}_{12} & B^-_{\lambda_m} \\
\vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots \\
\tilde{\sigma}_{2m,1}f^{\lambda_1}_{11} & \cdots & \tilde{\sigma}_{2m,m}f^{\lambda_m}_{11} & \tilde{\sigma}_{2m,m+1} & \cdots & \tilde{\sigma}_{2m,2m} & \tilde{\sigma}_{2m,m}f^{\lambda_m}_{12} & \cdots & \tilde{\sigma}_{2m,1}f^{\lambda_1}_{12} & B^-_{\lambda_1}
\end{array}
$$

By instantiating at $f_{kl}^{\lambda_i} = f_{kl}^{\mu_i} = 0$ for $k, l = 1, 2$ for all $i = 1, \cdots, m$ and using $Tr(M) = Tr(M')$, we conclude that $\tilde{\sigma}_{\underline{A}}$ is of the form: $\begin{bmatrix} \boldsymbol{B}_1 & \mathbf{0} \\ \mathbf{0} & \boldsymbol{B}_4 \end{bmatrix}$ where

$$\boldsymbol{B}_1 = \begin{array}{cccc|c} A^+_{\lambda_1} & A^+_{\lambda_2} & \cdots & A^+_{\lambda_m} & \\ \hline \tilde{\sigma}_{1,1} & \tilde{\sigma}_{1,2} & \cdots & \tilde{\sigma}_{1,m} & A^+_{\mu_1} \\ \tilde{\sigma}_{2,1} & \tilde{\sigma}_{2,2} & \cdots & \tilde{\sigma}_{2,m} & A^+_{\mu_2} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \tilde{\sigma}_{m,1} & \tilde{\sigma}_{m,2} & \cdots & \tilde{\sigma}_{m,m} & A^+_{\mu_m} \end{array}$$

$$\boldsymbol{B}_4 = \begin{array}{cccc|c} A^-_{\mu_m} & A^-_{\mu_{m-1}} & \cdots & A^-_{\mu_1} & \\ \hline \tilde{\sigma}_{m+1,m+1} & \tilde{\sigma}_{m+1,m+2} & \cdots & \tilde{\sigma}_{m+1,2m} & A^-_{\lambda_m} \\ \tilde{\sigma}_{m+2,m+1} & \tilde{\sigma}_{m+2,m+2} & \cdots & \tilde{\sigma}_{m+2,2m} & A^-_{\lambda_{m-1}} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \tilde{\sigma}_{2m,m+1} & \tilde{\sigma}_{2m,m+2} & \cdots & \tilde{\sigma}_{2m,2m} & A^-_{\lambda_1} \end{array}$$

Next, let $A^+_{\lambda_i} = A^-_{\lambda_i}$ and $B^+_{\lambda_i} = B^-_{\lambda_i}$ and f^{λ_i} be the twist, i.e. $f_{11}^{\lambda_i} = f_{22}^{\lambda_i} = 0$ and $f_{12}^{\lambda_i} = f_{21}^{\lambda_i} = 1$ for all $i = 1, \cdots, m$. Similarly for A_{μ_i}, B_{μ_i} and f^{μ_i} for $i = 1, \cdots, m$. We get the following system of equations:

System I:

$$NN' = \begin{bmatrix} \tilde{\sigma}_{1,m} & \cdots & \tilde{\sigma}_{1,1} \\ \tilde{\sigma}_{2,m} & \cdots & \tilde{\sigma}_{2,1} \\ \vdots & \cdots & \vdots \\ \tilde{\sigma}_{m,m} & \cdots & \tilde{\sigma}_{m,1} \end{bmatrix} \begin{bmatrix} \tilde{\sigma}_{m+1,m+1} & \cdots & \tilde{\sigma}_{m+1,2m} \\ \tilde{\sigma}_{m+2,m+1} & \cdots & \tilde{\sigma}_{m+2,2m} \\ \vdots & \cdots & \vdots \\ \tilde{\sigma}_{2m,m+1} & \cdots & \tilde{\sigma}_{2m,2m} \end{bmatrix} = antidiag(1, 1, \cdots, 1)$$

that is, we have $R_i^N C_j^{N'} = 1$ for $j = m - i + 1$ and 0 else, for $i = 1, \cdots, m$.

System II:

$$PP' = \begin{bmatrix} \tilde{\sigma}_{m+1,m+1} & \cdots & \tilde{\sigma}_{m+1,2m} \\ \tilde{\sigma}_{m+2,m+1} & \cdots & \tilde{\sigma}_{m+2,2m} \\ \vdots & \cdots & \vdots \\ \tilde{\sigma}_{2m,m+1} & \cdots & \tilde{\sigma}_{2m,2m} \end{bmatrix} \begin{bmatrix} \tilde{\sigma}_{m,1} & \cdots & \tilde{\sigma}_{m,m} \\ \tilde{\sigma}_{m-1,1} & \cdots & \tilde{\sigma}_{m-1,m} \\ \vdots & \cdots & \vdots \\ \tilde{\sigma}_{1,1} & \cdots & \tilde{\sigma}_{1,m} \end{bmatrix} = antidiag(1,1,\cdots,1)$$

that is $R_i^P C_j^{P'} = 1$ for $j = m - i + 1$ and 0 else, for $i = 1, \cdots, m$.

The rest of the proof consists of several steps:

Step 1: We show that $\boldsymbol{B}_4 = \boldsymbol{B}_1^t$, that is, $\tilde{\sigma}_{i,j} = \tilde{\sigma}_{2m+1-j,2m+1-i}$ for $i, j = m+1, m+2, \cdots, 2m$. Note that $(-)^t$ denotes the matrix transposition which is obtained by reflection across the *antidiagonal* entries.

• *case 1*: $\tilde{\sigma}_{i,j} = 1, \tilde{\sigma}_{2m+1-j,2m+1-i} = 0$. Note that $R_{j-m}^P C_k^{P'} = 0$ for all $k = 1, \cdots, m$, $k \neq 2m - j + 1$. $\tilde{\sigma}_{i,j} = 1$ implies that $\tilde{\sigma}_{2m-j+1,k} = 0$ for $k \neq 2m - i + 1$, however $\tilde{\sigma}_{2m-j+1,2m-i+1} = 0$ is given. Therefore, $R_{j-m}^{P'} = 0$ and $R_{2m-j+1}^N = R_{j-m}^{P'} = 0$ giving $R_{2m-j+1}^N C_{j-m}^{N'} = 0$, a contradiction.

• *case 2*: $\tilde{\sigma}_{i,j} = 0, \tilde{\sigma}_{2m+1-j,2m+1-i} = 1$. Note that $R_{2m-i+1}^N C_k^{N'} = 0$ for all $k = 1, \cdots, m$, $k \neq i - m$. $\tilde{\sigma}_{2m-j+1,2m-i+1} = 1$ implies $\tilde{\sigma}_{i,k} = 0$ for all $k \neq j$, but $\tilde{\sigma}_{i,j} = 0$ is given, hence $R_{i-m}^{N'} = 0$, and $R_{i-m}^P = R_{i-m}^{N'} = 0$ giving $R_{i-m}^P C_{2m-i+1}^{P'} = 0$, a contradiction.

Hence $\boldsymbol{B}_4 = \boldsymbol{B}_1^t$.

Step 2: There are no all-zero rows or columns in $\boldsymbol{B}_1$ or $\boldsymbol{B}_4$.

The ith row of $\boldsymbol{B}_1$ is equal to R_i^N in reverse order, hence it cannot be all zero since $R_i^N C_{m-i+1}^{N'} = 1$. Also the jth column of $\boldsymbol{B}_1$ is equal to $C_j^{P'}$ in reverse order and hence it cannot be all zero since $R_{m+1-j}^P C_j^{P'} = 1$.

The statement is trivially true for $\boldsymbol{B}_4$ as $\boldsymbol{B}_4 = \boldsymbol{B}_1^t$.

Step 3: In $\boldsymbol{B}_1$ and $\boldsymbol{B}_4$ every row and column has exactly one 1. Suppose any two elements on the ith row of $\boldsymbol{B}_1$ are 1; $\tilde{\sigma}_{i,j} = \tilde{\sigma}_{i,k} = 1$ for $k \neq j$ with $i, j, k \in \{1, \cdots, m\}$. The ith row of $\boldsymbol{B}_1 = R_{m-i+1}^{P'}$ and hence $C_{m-i+1}^P = 0$. For example, suppose $\tilde{\sigma}_{1,1} = \tilde{\sigma}_{1,m} = 1$, then using system II we see that all the elements on the last column of P are zero by just using the fact that $\tilde{\sigma}_{1,1} = 1$. Also $\tilde{\sigma}_{2m,2m} = 0$ because $\tilde{\sigma}_{1,m} = 1$. Note that $C_{m-i+1}^P = C_{m-i+1}^{\boldsymbol{B}_4} = 0$, a contradiction.

Also let any two elements on the jth column of $\boldsymbol{B}_1$ be both 1; $\tilde{\sigma}_{i,j} = \tilde{\sigma}_{k,j} = 1$ for $i \neq k$ with $i, j, k \in \{1, \cdots, m\}$. The jth column of $\boldsymbol{B}_1 = C_{m-j+1}^N$ and hence

$R^{N'}_{m-j+1} = 0$, and as $R^{\boldsymbol{B}_4}_{m-j+1} = R^{N'}_{m-j+1}$, we get a contradiction.

As $\boldsymbol{B}_4 = \boldsymbol{B}_1^t$, the statement follows for $\boldsymbol{B}_4$.

Therefore, $\boldsymbol{B}_1$ and $\boldsymbol{B}_4$ are permutation matrices. Let $\delta \in S_m$ be the permutation induced by $\boldsymbol{B}_1$, that is $\delta(i) = j$ iff $\tilde{\sigma}_{i,j} = 1$. Then we have $\tilde{\sigma}_{i,j} : A^+_{\lambda_j} \longrightarrow A^+_{\mu_i} = 1$ and $\tilde{\sigma}_{2m+1-j,2m+1-i} : A^-_{\mu_i} \longrightarrow A^-_{\lambda_j} = 1$ and thus $A_{\lambda_{\delta(i)}} = A_{\lambda_j} = (A^+_{\lambda_j}, A^-_{\lambda_j}) = (A^+_{\mu_i}, A^-_{\mu_i}) = A_{\mu_i}$ for all $i = 1, \cdots, m$.

□

We can view the natural transformation $\tilde{\sigma}_{\underline{A}} : F^-(\underline{A}) \longrightarrow F^+(\underline{A})$ as matching $A_{\lambda_{\delta(i)}}$ to A_{μ_i} for all $i = 1, \cdots, m$. Hence we have:

Corollary 10.6.6 (Full Completeness in $\mathbb{C}$) *Every natural transformation $\tilde{\sigma}$: $F^- \longrightarrow F^+$ in $\mathbb{C}$ is induced by a unique morphism $F^-(\underline{X}) \longrightarrow F^+(\underline{X})$ in the free compact closed category on n objects $X_1, \cdots, X_n$ with trivial dimension.*

Note that all objects in $\mathbb{C}$ have trivial dimension, i.e. $dim(A) = 1_I$ for all objects A in $\mathbb{C}$, since $\mathbb{C}(I, I) = \mathbb{D}(I, I) = \{1_I\}$. Therefore, the restriction on dimension can be removed: it is tensoring with finitely many 1_I maps, which have no effect. Thus

Corollary 10.6.7 (Full Completeness in $\mathbb{C}$) *Every natural transformation $\tilde{\sigma}$: $F^- \longrightarrow F^+$ in $\mathbb{C}$ is induced by a morphism $F^-(\underline{X}) \longrightarrow F^+(\underline{X})$ in the free compact closed category on n objects $X_1, \cdots, X_n$.*

Note that this is a weaker result than Corollary 10.6.6, because we lose uniqueness.

Theorem 10.6.8 *Suppose that σ is a dinatural transformation in $\mathbb{C}$ from $\mathcal{K}_I$ to the multivariant functor F. Then there exists a fixed-point-free involution φ on $\{1, \cdots, p\}$ such that $\xi_{\varphi(i)} = \xi_i, \zeta_{\varphi(i)} \neq \zeta_i$.*

Proof. The permutation δ associated with the natural transformation $\tilde{\sigma}$ identifies occurrences of A_{ξ_i} in $F^-(\underline{A})$ with occurrences of A_{ξ_i} in $F^+(\underline{A})$. We will use δ to define the permutation φ on $\{1, \cdots, p\}$ which will be associated with the dinatural transformation σ. By a previous theorem we know that F is balanced. Write $N =$

$\{i_1, i_2, \cdots, i_m\}$ and $P = \{j_1, \cdots, j_m\}$ where $N = \{i|\zeta_i = *\}$ and $P = \{i|\zeta_i = 1\}$, that is N and P contain the indices of the negative and positive occurrences of propositional atoms respectively. Note that N and P have the same cardinality because F is balanced. Also assume that $\xi_{i_k} = \lambda_k$ and $\xi_{j_k} = \mu_k$ for $k = 1, \cdots, m$. Recall that $\delta \in S_m$, the permutation induced by $\tilde{\sigma}$, matches $A_{\lambda_{\delta(k)}}$ to A_{μ_k}, thus $\xi_{i_{\delta(k)}} = \xi_{j_k}$ and similarly, $\xi_{j_{\delta^{-1}(k)}} = \xi_{i_k}$, for all $k = 1, \cdots, m$.

Define $\varphi(i_k) = j_{\delta^{-1}(k)}$ and $\varphi(j_k) = i_{\delta(k)}$ for all $k = 1, \cdots, m$. Then

$$\varphi^2(i_k) = \varphi(j_{\delta^{-1}(k)}) = i_k, \;\; \varphi^2(j_k) = \varphi(i_{\delta(k)}) = j_k$$

for all k, hence φ is an involution. Also, φ maps N to P and vice versa, and so $\zeta_{\varphi(i)} \neq \zeta_i$. Finally, $\xi_{\varphi(i_k)} = \xi_{j_{\delta^{-1}(k)}} = \xi_{i_k}$ and $\xi_{\varphi(j_k)} = \xi_{i_{\delta(k)}} = \xi_{j_k}$ for all $k = 1, \cdots, m$.

$\square$

In view of this theorem we see that σ determines a *unique* set of axiom links and hence a unique MLL proof structure for the formula F. In the next section we will show that this proof structure is indeed a proof net. That is, we need to check the Danos-Regnier correctness criterion. However, as $\mathbb{C}(I, I) = \mathbb{D}(I, I) = \{1_I\}$, $\boldsymbol{G}\mathbb{C}$ satisfies the MIX rule and hence we need only check the acyclicity condition.

10.6.2 Full Completeness in $\boldsymbol{G}\mathbb{C}$

Given a dinatural transformation $\rho : \mathcal{K}_{\mathbf{1}} \longrightarrow F$ in $\boldsymbol{G}\mathbb{C}$, we have the specification of a unique proof structure because we have the formula F and the axiom links are given by the fixed-point free involution φ induced by the dinatural transformation $U\rho$ in the underlying compact closed category $\mathbb{C} = Int(\mathbb{D})$. We show that this proof structure is indeed a proof net. For this purpose we only need to prove acyclicity as our category $\boldsymbol{G}\mathbb{C}$ satisfies the MIX rule.

Lemma 10.6.9 *Let $F(\underline{X}, \underline{X}) = F_1(\underline{X}, \underline{X}) \otimes F_2(\underline{X}, \underline{X})$ be an object in the free compact closed category on n objects $X_1, \cdots, X_n$ with trivial dimension and $\Gamma : I \longrightarrow F(\underline{X}, \underline{X})$ be a morphism. Suppose also that the induced fixed-point free involution φ does not make a matching between formulas in F_1 and those in F_2, then $\Gamma = \Gamma_1 \otimes \Gamma_2$ where $\Gamma_1 : I \longrightarrow F_1(\underline{X}, \underline{X})$ and $\Gamma_2 : I \longrightarrow F_2(\underline{X}, \underline{X})$.*

Proof. Γ induces a fixed-point-free involution φ on $F(\underline{X}, \underline{X})$ which consists of a pair (φ_1, φ_2) of fixed-point-free involutions on $F_1(\underline{X}, \underline{X})$ and $F_2(\underline{X}, \underline{X})$ respectively and hence there are unique morphisms $\Gamma_1 : I \longrightarrow F_1(\underline{X}, \underline{X})$ and $\Gamma_2 : I \longrightarrow F_2(\underline{X}, \underline{X})$ which correspond to φ_1 and φ_2 respectively. Therefore, $\Gamma_1 \otimes \Gamma_2$ is a morphism in the free category corresponding to φ, however such a morphism is unique and hence $\Gamma = \Gamma_1 \otimes \Gamma_2$. $\Box$

Theorem 10.6.10 (Acyclicity) *Suppose that ρ is a dinatural transformation in $\boldsymbol{G}\mathbb{C}$ from the constant functor $\mathcal{K}_{\mathbf{1}}$ to F. Consider the unique proof structure associated with ρ. Then for any DR-switching, the associated DR-graph is acyclic.*

Proof. Suppose that for a certain DR-switching, the associated DR-graph contains a cycle. Express the shortest cycle as lower connected pairs $(a_1, b_1), \cdots, (a_r, b_r)$ where $\varphi(b_i) = a_{i+1}$ for all $i \in \mathbb{Z}_r$. Recall that a lower connected pair in a proof structure is a pair of formulas that are connected with paths not traversing any axiom links [82, 83]. Using the weak distributivity natural transformations, binary MIX morphisms and associativity and commutativity natural transformations for par and tensor we transform the given dinatural transformation ρ into $\tilde{\rho} : \mathcal{K}_{\mathbf{1}} \longrightarrow \tilde{F}$ such that the cycle is preserved, where $\tilde{\rho}_{\underline{\mathcal{A}}} : \mathbf{1} \longrightarrow \tilde{F}(\underline{\mathcal{A}}, \underline{\mathcal{A}})$ and

$$\tilde{F}(\underline{\mathcal{A}}, \underline{\mathcal{A}}) = \Gamma_{\underline{\mathcal{A}}} \parr (\mathcal{A}_{\xi_{a_1}}^{\zeta_{a_1}} \otimes \mathcal{A}_{\xi_{b_1}}^{\zeta_{b_1}}) \parr \cdots \parr (\mathcal{A}_{\xi_{a_r}}^{\zeta_{a_r}} \otimes \mathcal{A}_{\xi_{b_r}}^{\zeta_{b_r}})$$

The procedure is as follows (see also [104] and [4]):

- If a fragment of F has the form $A \otimes (B \parr C)$, and A and B are lower connected, then the switching must have assigned *left* to the par-link in question. In this case, we compose ρ with a natural transformation built from w^L_{ABC},
- If A and C are lower connected, then the switching must have assigned *right* to the par-link. In this case, we compose with a natural transformation built from w^R_{ABC},
- We apply binary MIX, commutativity and associativity, whenever necessary to separate out each lower connected pair.

Consider the test object $\mathcal{A} = (A, \{0_{A^-A^+}\}, \{0_{A^+A^-}\})$ where $A = (A^+, A^-) \neq (I, I)$ and $A^+ = A^-$. Hence $\mathcal{A}^\perp = \mathcal{A}$. Put $\mathcal{A}_i = \mathcal{A}$ for $i = 1, \cdots, n$. In what follows there is no need to put ζ_- superscripts as $\mathcal{A} = \mathcal{A}^\perp$, however we have included these for clarity.

Notice that $U(\tilde{\rho_{\underline{\mathcal{A}}}}) = f_1 \otimes f_2$,

$$f_1 : I \longrightarrow \Gamma_{\underline{A}} \quad \text{and} \quad f_2 : I \longrightarrow (A^{\zeta_{a_1}} \otimes A^{\zeta_{b_1}}) \otimes \cdots \otimes (A^{\zeta_{a_r}} \otimes A^{\zeta_{b_r}})$$

because the part in $\tilde{F}(\underline{\mathcal{A}}, \underline{\mathcal{A}})$ consisting of par product of tensored pairs is closed under the axiom link matchings induced by any dinatural transformation. Therefore, we have that f_2 must lift to a morphism in $\boldsymbol{G}\mathbb{C}$ from $\mathbf{1}$ to $(\mathcal{A}^{\zeta_{a_1}} \otimes \mathcal{A}^{\zeta_{b_1}}) \mathbin{⅋} \cdots \mathbin{⅋} (\mathcal{A}^{\zeta_{a_r}} \otimes \mathcal{A}^{\zeta_{b_r}})$.

Hence $f_2 \in ((\mathcal{A}^{\zeta_{a_1}} \otimes \mathcal{A}^{\zeta_{b_1}}) \mathbin{⅋} \cdots \mathbin{⅋} (\mathcal{A}^{\zeta_{a_r}} \otimes \mathcal{A}^{\zeta_{b_r}}))_s$.

$((\mathcal{A}^{\zeta_{a_1}} \otimes \mathcal{A}^{\zeta_{b_1}}) \mathbin{⅋} \cdots \mathbin{⅋} (\mathcal{A}^{\zeta_{a_r}} \otimes \mathcal{A}^{\zeta_{b_r}}))_s =$

$((\mathcal{A}^{\zeta_{a_1}} \otimes \mathcal{A}^{\zeta_{b_1}})^\perp \otimes \cdots \otimes (\mathcal{A}^{\zeta_{a_r}} \otimes \mathcal{A}^{\zeta_{b_r}})^\perp)_t =$

$\boldsymbol{G}\mathbb{C}((\mathcal{A}^{\zeta_{a_1}} \otimes \mathcal{A}^{\zeta_{b_1}})^\perp \otimes \cdots \otimes (\mathcal{A}^{\zeta_{a_{r-1}}} \otimes \mathcal{A}^{\zeta_{b_{r-1}}})^\perp, (\mathcal{A}^{\zeta_{a_r}} \otimes \mathcal{A}^{\zeta_{b_r}}))$

Now consider $((\mathcal{A}^{\zeta_{a_1}} \otimes \mathcal{A}^{\zeta_{b_1}})^\perp \otimes \cdots \otimes (\mathcal{A}^{\zeta_{a_{r-1}}} \otimes \mathcal{A}^{\zeta_{b_{r-1}}})^\perp)_s = \{\sigma_1 \otimes \cdots \otimes \sigma_{r-1} \mid \sigma_i \in (\mathcal{A}^{\zeta_{a_i}} \otimes \mathcal{A}^{\zeta_{a_i}})^\perp_s\}$.

Notice that $(\mathcal{A}^{\zeta_{a_i}} \otimes \mathcal{A}^{\zeta_{b_i}})^\perp_s = (\mathcal{A}^{\zeta_{a_i}} \otimes \mathcal{A}^{\zeta_{b_i}})_t = \boldsymbol{G}\mathbb{C}(\mathcal{A}, \mathcal{A})$ and hence $1_A \in (\mathcal{A}^{\zeta_{a_i}} \otimes \mathcal{A}^{\zeta_{b_i}})^\perp_s$ and therefore $\underbrace{1_A \otimes \cdots \otimes 1_A}_{r-1\ times} \in ((\mathcal{A}^{\zeta_{a_1}} \otimes \mathcal{A}^{\zeta_{b_1}})^\perp \otimes \cdots \otimes (\mathcal{A}^{\zeta_{a_{r-1}}} \otimes \mathcal{A}^{\zeta_{b_{r-1}}})^\perp)_s$

On the other hand, $(\mathcal{A}^{\zeta_{a_r}} \otimes \mathcal{A}^{\zeta_{b_r}})_s = \{0\}$.

Now by definition, $f_2 \circ \alpha \in (\mathcal{A}^{\zeta_{a_r}} \otimes \mathcal{A}^{\zeta_{b_r}})_s = \{0\}$ for all $\alpha \in ((\mathcal{A}^{\zeta_{a_1}} \otimes \mathcal{A}^{\zeta_{b_1}})^\perp \otimes \cdots \otimes (\mathcal{A}^{\zeta_{a_{r-1}}} \otimes \mathcal{A}^{\zeta_{b_{r-1}}})^\perp)_s$ and hence $f_2 = 0$ which yields a contradiction because such a morphism cannot induce any axiom links.

Thus assuming the existence of a cycle in the proof structure and hence $\tilde{\rho}$ we have obtained a contradiction and hence a dinatural transformation like $\tilde{\rho}$ cannot exist and therefore the proof structure contains no cycles.

$\square$

Theorem 10.6.11 (Full completeness in $\boldsymbol{G}\mathbb{C}$) *Every dinatural transformation in $\boldsymbol{G}\mathbb{C}$ from the constant functor $\mathcal{K}_\mathbf{1}$ to the multivariant functor F is the denotation of a unique cut-free proof in MLL of the formula F, and is therefore induced by a* *unique morphism* $\mathbf{1} \longrightarrow F(\underline{X}, \underline{X})$ *in the free *-autonomous category on* n *objects* $X_1, X_2, \cdots, X_n$.

The techniques that we have used in the proof of the last theorem are general enough to cover the case where $\mathbb{D}$ is a traced UDC. Observe that $\mathbb{D}(I, I) = \{1_I\}$ in any UDC $\mathbb{D}$. Note that by coherence axioms we have $\rho_I = \lambda_I$ and hence $[1_I, 0_I] = [0_I, 1_I]$ and thus $0_I = 1_I$. This also implies that I is a zero object. Thus we have:

Theorem 10.6.12 *Let* $\mathbb{D}$ *be a traced UDC. Then, every dinatural transformation in* $\boldsymbol{G}Int(\mathbb{D})$ *from the constant functor* $\mathcal{K}_{\mathbf{1}}$ *to the multivariant functor* F *is the denotation of a unique cut-free proof in MLL of the formula* F*, and is therefore induced by a unique morphism* $\mathbf{1} \longrightarrow F(\underline{X}, \underline{X})$ *in the free *-autonomous category on* n *objects* $X_1, X_2, \cdots, X_n$.

Hence, in particular, we have full completeness for MLL+MIX in the *-autonomous categories constructed from **PInj** and $\mathbf{Hilb}_2$.

We conclude this chapter by stating a negative result for the class of categories $\boldsymbol{G}\mathbb{C}$ with $\mathbb{C} = Int(\mathbb{D})$ and $\mathbb{D}$ a PAC. Suppose we choose to use the traditional categorical semantics framework [78]. That is, formulas of MLL are objects in $\boldsymbol{G}\mathbb{C}$ and proofs are morphisms. Then we show that $\boldsymbol{G}\mathbb{C}$ fails to be fully complete for MLL.

Theorem 10.6.13 *Let* $\mathbb{D}$ *be a PAC and* $\mathbb{C} = Int(\mathbb{D})$*, interpret the formulas of MLL as objects in* $\boldsymbol{G}\mathbb{C}$ *and the proofs as morphisms. Then,* $\boldsymbol{G}\mathbb{C}$ *is not fully complete for MLL.*

Proof. Let $\mathcal{A} = (A, \mathcal{A}_s, \mathcal{A}_t)$ be an object in $\boldsymbol{G}\mathbb{C}$ with $A = (A^+, A^-)$ and $A^+ = A^-$. Also, let $\mathcal{A}_s = \mathcal{A}_t = \{1_{A^+}\}$. Note that $\mathcal{A} \otimes \mathcal{A}^\perp$ is not an MLL provable formula. We show that there exists a map $f : \mathbf{1} \longrightarrow \mathcal{A} \otimes \mathcal{A}^\perp$. Let $f : I \longrightarrow A \otimes A^*$ be $1_{A^+ \otimes A^+}$. Recall that $\mathbf{1}_s = \{1_I\}$ and therefore $f\alpha = f \in (\mathcal{A} \otimes \mathcal{A}^\perp)_s$ for all $\alpha \in \mathbf{1}_s$. Recall that $(\mathcal{A} \otimes \mathcal{A}^\perp)_t = \boldsymbol{G}\mathbb{C}(\mathcal{A}, \mathcal{A})$ and hence $(\mathcal{A} \otimes \mathcal{A}^\perp)_t \neq \emptyset$. To conclude the proof, we need to show that $\beta f = 1_I$ for all $\beta \in (\mathcal{A} \otimes \mathcal{A}^\perp)_t$, but $\beta f : I \longrightarrow I$ in $\mathbb{C}$ and $\mathbb{C}(I, I) = \{1_I\}$, therefore $\beta f = 1_I$ and thus $\beta f \in \mathbf{1}_t$ for all $\beta \in (\mathcal{A} \otimes \mathcal{A}^\perp)_t$. $\square$

Chapter 11

Conclusions and Further Work

In this thesis we studied Girard's GoI programme à la Abramsky and showed how to construct models of computation (combinatory algebras) based on a GoI Situation. We also showed that the construction can be generalised to obtain combinatory algebras from a weak linear category.

The combinatory algebras we obtain in this way are highly intensional and further research is required in the direction of extensional reflections of these models by considering the type structure (of modest sets or partial equivalence relations) they induce. This line of research is important for theoretical computer science in that it provides models of fine-structural features of computation and optimal reduction strategies for λ-calculus.

In particular, we studied the dataflow computational aspects of Abramsky's programme. We showed that UDCs are a suitable categorical framework for such a computational analysis. Furthermore, such categories were used to construct categorical models of dynamic algebras. This way we have the means of analysing the dynamics of information flow in proof nets in a categorical setting. We also showed that Danos' small model and Girard's operator algebraic models are instances of dynamic situations.

The geometry of interaction provides a suitable semantic basis for explaining and improving Lamping's optimal graph-reduction implementation of the λ-calculus. The connections have been explained in [53]. The investigation of connections and relations between optimal reduction strategies and geometry of interaction offers ideas for efficient and correct implementations. An exciting research direction is the application of the ideas in this thesis to the analysis of optimal reduction strategies. In particular recent work of Asperti and Mairson [10] offers interesting examples amenable to the token pushing analysis developed in this thesis. We intend to examine this further in future research work.

An important issue is to relate the nilpotency criterion in Girard's formulation to properties of sums that we obtain in our UDC-based models. We expect to get a characterisation for strongly normalising terms of untyped lambda calculus.

We intend to further study the models of linear logic and implementations of GoI based on categories that appear in functional analysis, for example categories of Banach spaces such as $\mathbf{Ban}_1$ of real Banach spaces and linear maps with norm less than or equal to 1 and $\mathbf{Ban}_\infty$ of real Banach spaces and linear bounded maps. This would help to provide a functional analytic analysis of combinators in particular and β-reduction in general and to deepen our understanding of cut elimination/proof normalisation processes by making use of techniques and structures in the theory of operator algebras. In particular the ideas used in the construction of $\mathbf{Hilb}_2$ can be generalised using the theory of fields of Hilbert spaces.

More interestingly, we encounter a partial trace operator [3] for **Hilb**. The existence conditions for trace when related to the properties of λ-terms is expected to shed new light on the mathematical analysis of β-reduction.

Another one of the models that I study in my thesis, namely **SRel**, establishes a connection with probabilistic networks. The study of feedback in probabilistic networks has many potential applications in queueing and network theory. Also such models provide a continuous semantics for programming in contrast to many extant discrete models.

Another important direction is iteration theories, more specifically we have used partially additive categories as the main categorical models. We also showed that PACs possess an iteration operation which is used to define a trace operator. There are therefore several connections to iteration theories that can yield new applications for iteration theories on the one hand and on the other hand the extensive and well developed literature on iteration theories can enrich our understanding of geometry of interaction, Kleene *-algebras, etc.

Finally, we showed how to construct models of MLL, i.e. *-autonomous categories based on PACs. We made use of the Int and double glueing constructions to get such models. We also proved that such models are fully complete for MLL+MIX.

A major problem is to extend full completeness to different fragments of linear logic (e.g. additives, exponentials.) Game-theoretical models have recently become a major tool in this area. We intend to unify these models with the ones studied in this thesis. Current results by several research groups around the world appear amenable to an abstract and axiomatic approach which is yet to be developed. First steps towards such an approach were taken by Hyland, Abramsky and their colleagues and students. Recently Abramsky and Mellies announced a novel game-theoretic full completeness result for the multiplicative and additive fragment of linear logic. Current work aims to give new non-game-theoretic fully complete models.

Appendix A

Diagrams for Trace Axioms

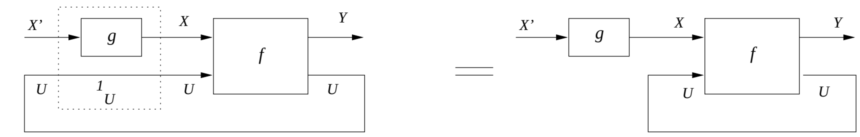


Figure 29: Naturality in X

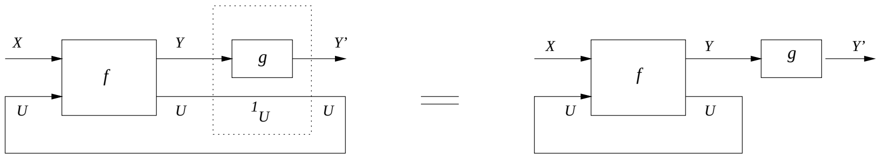


Figure 30: Naturality in Y

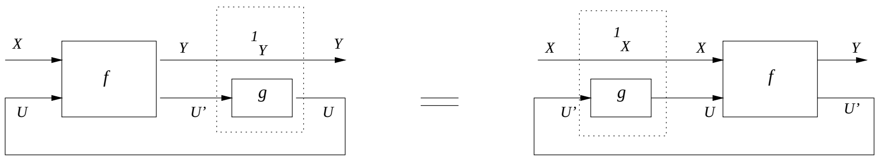


Figure 31: Dinaturality in U

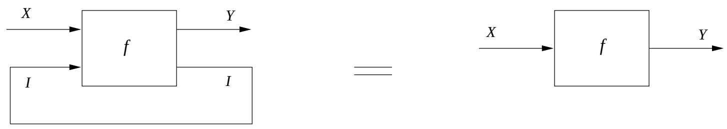


Figure 32: Vanishing I

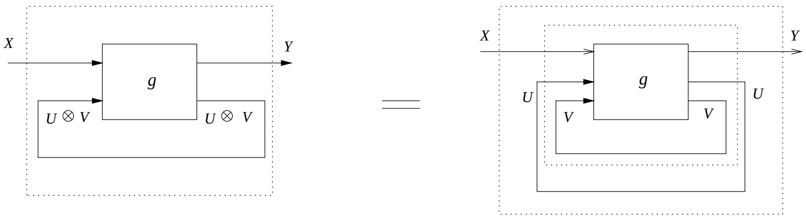


Figure 33: Vanishing II

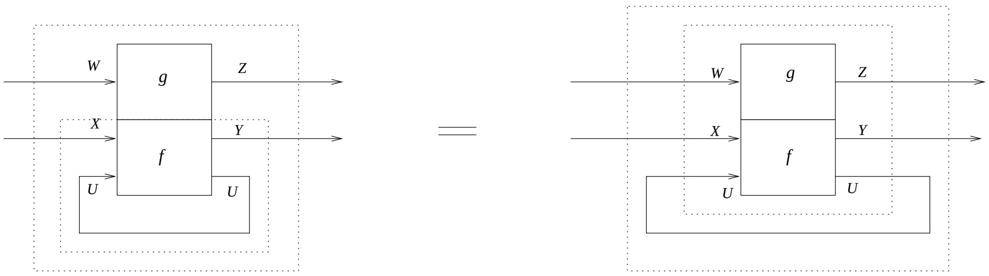


Figure 34: Superposing (see Proposition 2.1.18)

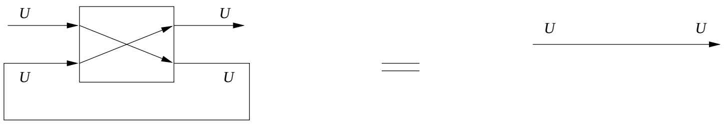


Figure 35: Yanking

Appendix B

Derivation of GoI Combinators

We give a brief summary of our graphical notation in Figure 36. In this figure:

(i) identity morphism, 1_X

(ii) symmetry morphism, σ_{XY}

(iii) tensor product, $f \otimes g : X \otimes X' \longrightarrow Y \otimes Y'$

(iv) trace, $Tr^U_{X,Y}(f) : X \longrightarrow Y$

(v) composition, $gf : (X^+, X^-) \longrightarrow (Z^+, Z^-)$, where $f : (X^+, X^-) \longrightarrow (Y^+, Y^-)$ and $g : (Y^+, Y^-) \longrightarrow (Z^+, Z^-)$

(vi) transposition, $\Lambda(f) : (X^+, X^-) \longrightarrow (Y^+, Y^-) \multimap (Z^+, Z^-)$, where $f : (X^+, X^-) \otimes (Y^+, Y^-) \longrightarrow (Z^+, Z^-)$

(v) evaluation, $ev : ((Y^+, Y^-) \multimap (Z^+, Z^-)) \otimes (Y^+, Y^-) \longrightarrow (Z^+, Z^-)$

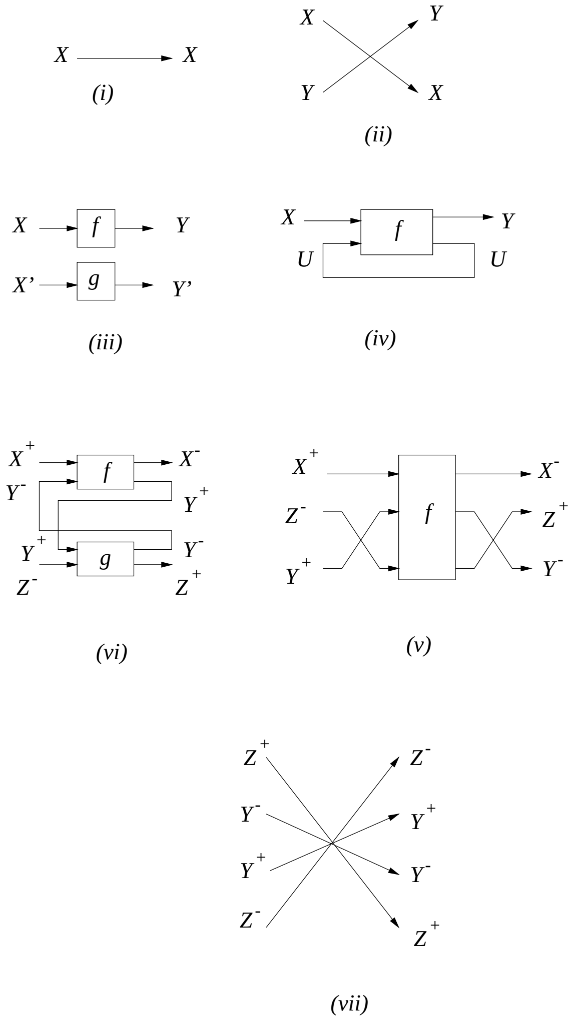


Figure 36: Summary of the graphical notation

In the figures for the combinators, an arrow between two figures means that the figure on the left is reduced to the one on the right using the rules of the graphical calculus. The captions for all diagrams are given below.

- Combinator I, $r\Lambda(1_V)$
- Combinator B
  - (i) $b = ev(1 \otimes ev)$
  - (ii) $\Lambda^3(b)$
  - (iii) $r(1 \multimap r)(s \multimap (s \multimap r))\Lambda^3(b)$

- Combinator C
  (i) $(ev \otimes 1)(1 \otimes \sigma)$
  (ii) $c = ev(ev \otimes 1)(1 \otimes \sigma)$
  (iii) $\Lambda^3(c)$
  (iv) $r(s \multimap r)((1 \multimap s) \multimap (1 \multimap r))\Lambda^3(c)$
- Combinator K
  (i) $k = 1_V \otimes weak_V$
  (ii) $\Lambda^2(k)$
  (iii) $(1 \multimap (q \multimap 1))\Lambda^2(k)$, $f_K = uw_U w'_U v$
  (iv) $r(1 \multimap r)(1 \multimap (q \multimap 1))\Lambda^2(k)$
- Combinator W
  (i) $w = ev(ev \otimes 1)(1 \otimes con_V)$
  (ii) $\Lambda^2(w)$
  (iii) $((p \multimap (p \multimap 1)) \multimap 1)(1 \multimap (q \multimap 1))\Lambda^2(w)$,
  $f_W = uc_U(v \otimes v)$, $g_W = (u \otimes u)c'_U v$
  (iv) $((1 \multimap s) \multimap r)((p \multimap (p \multimap 1)) \multimap 1)(1 \multimap (q \multimap 1))\Lambda^2(w)$
  (v) $r(s \multimap 1)((1 \multimap s) \multimap r)((p \multimap (p \multimap 1)) \multimap 1)(1 \multimap (q \multimap 1))\Lambda^2(w)$
- Combinator D
  (i) $d = ev(1 \otimes der_V)$
  (ii) $\Lambda^2(d)$
  (iii) $r(1 \multimap r)(s \multimap (q \multimap 1))\Lambda^2(d)$, $f_D = ul_U$, $g_D = l'_U v$
- Combinator δ
  (i) $(1 \multimap !p)\Lambda(\delta_V)$

(ii) $(q \multimap p)(1 \multimap !p)\Lambda(\delta_V)$, $f_\delta = ue_U T(v)v$, $g_\delta = uT(u)e'_U v$

(iii) $r(q \multimap p)(1 \multimap !p)\Lambda(\delta_V)$

- Combinator F

(i) $f = !ev\varphi$

(ii) $\Lambda^2(f)$

(iii) $(!s \multimap (q \multimap p))\Lambda^2(f)$

(iv) $(q \multimap 1)(!s \multimap (q \multimap p))\Lambda^2(f)$,
$f_F = uT(j)\psi_{U,U}(v \otimes v)$, $g_F = (u \otimes u)\psi^{-1}_{U,U}T(k)v$

(v) $r(1 \multimap r)(q \multimap 1)(!s \multimap (q \multimap p))\Lambda^2(f)$

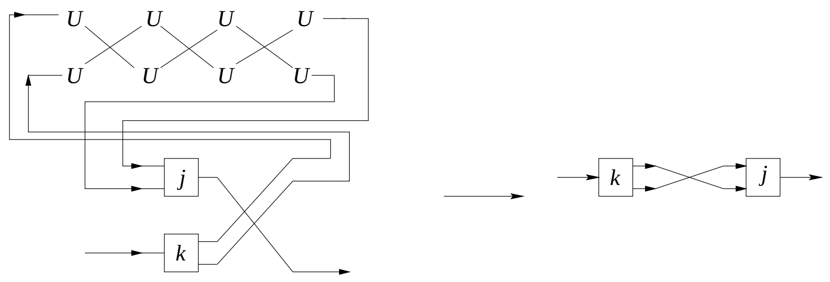
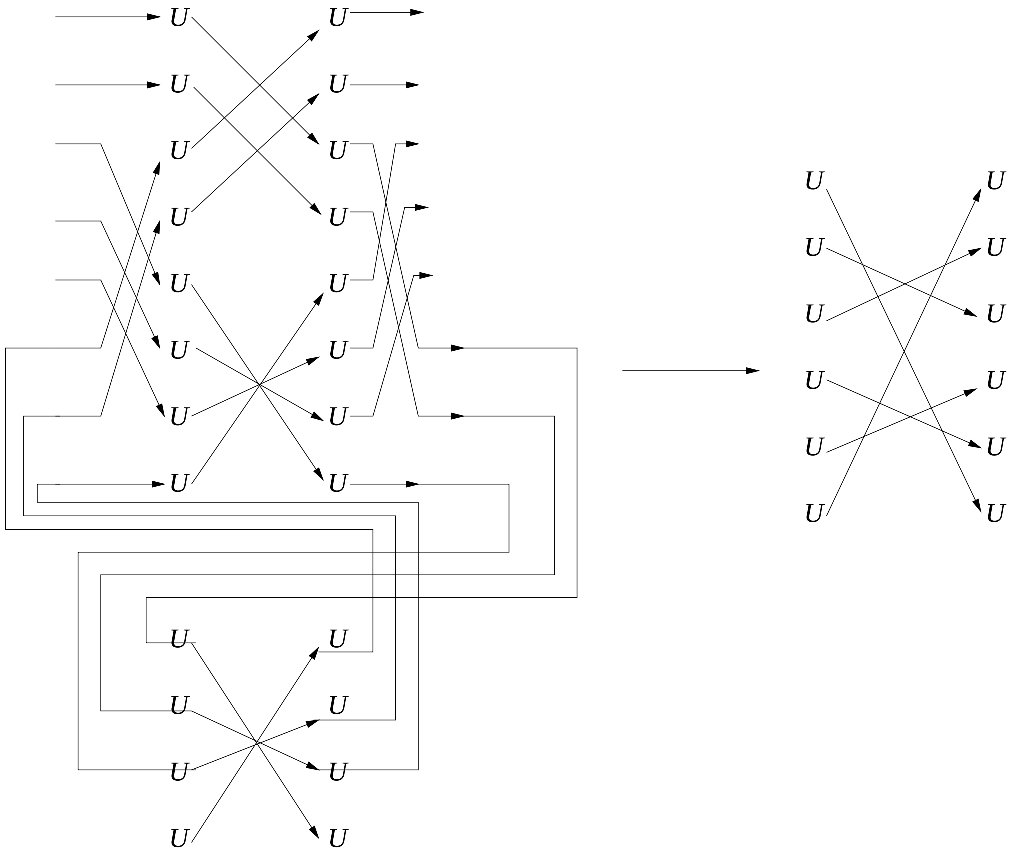
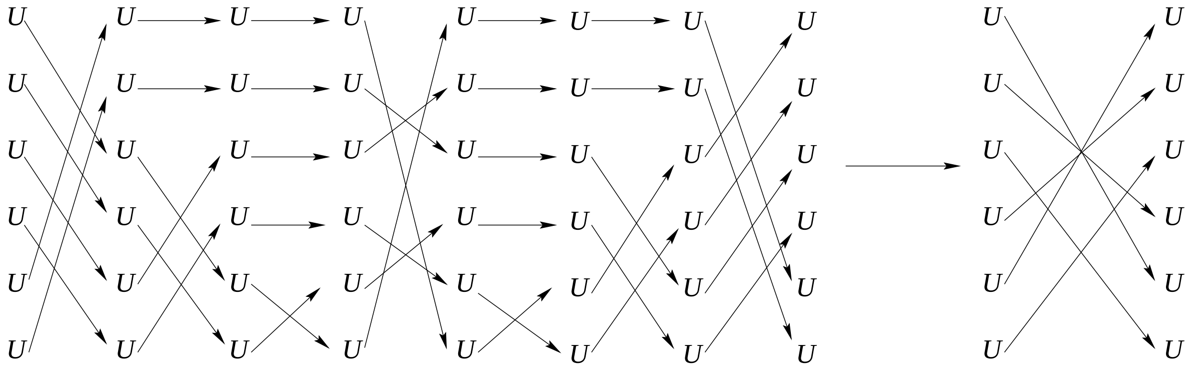


Figure 37: Combinator I

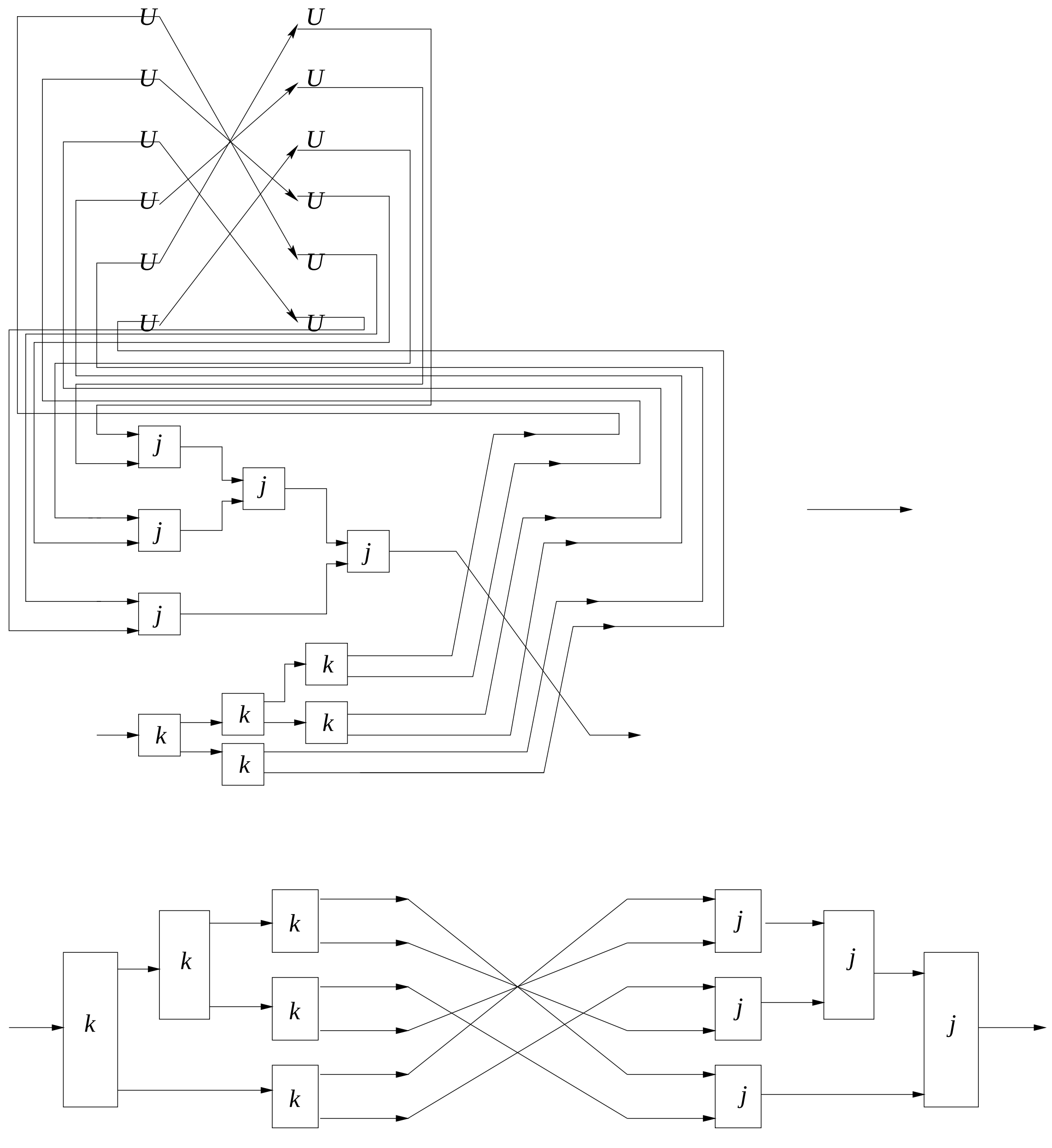


(i)



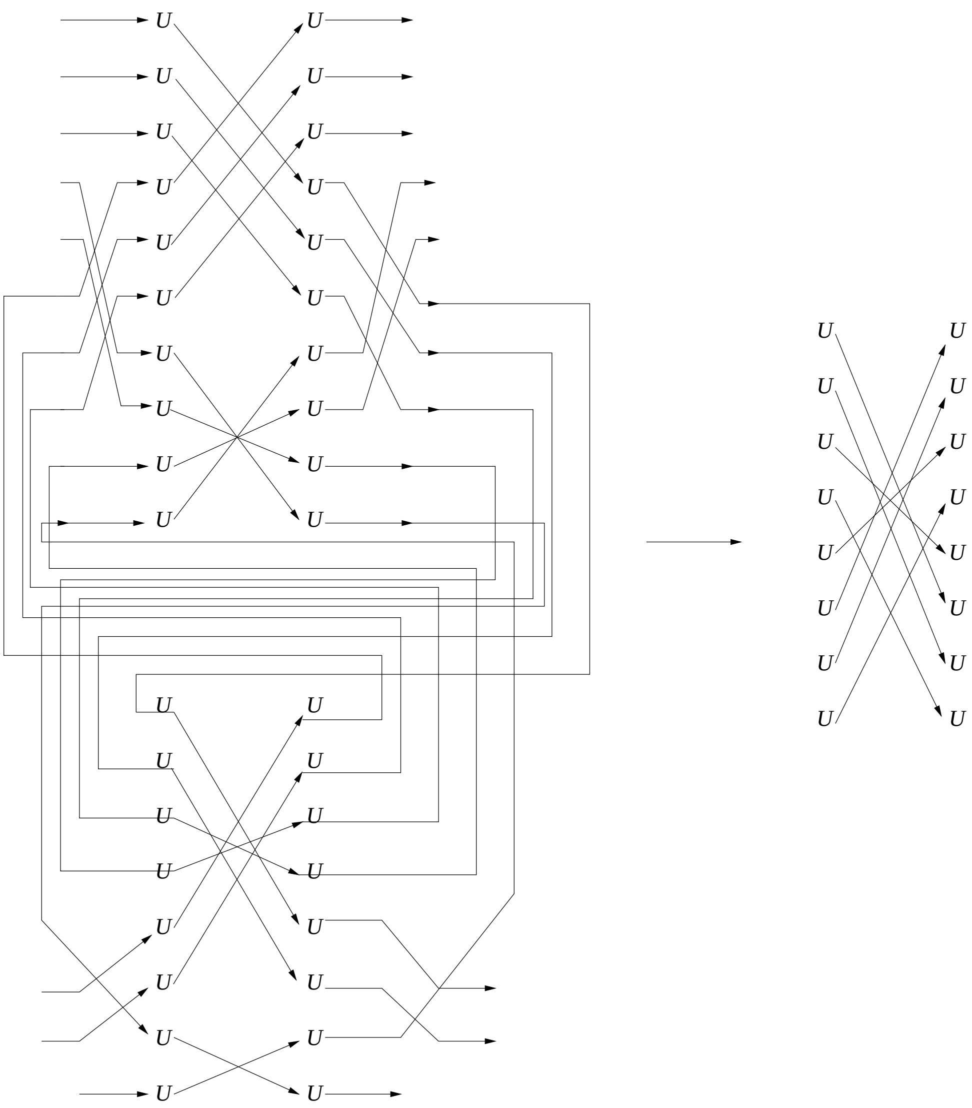
(ii)

Figure 38: Combinator B



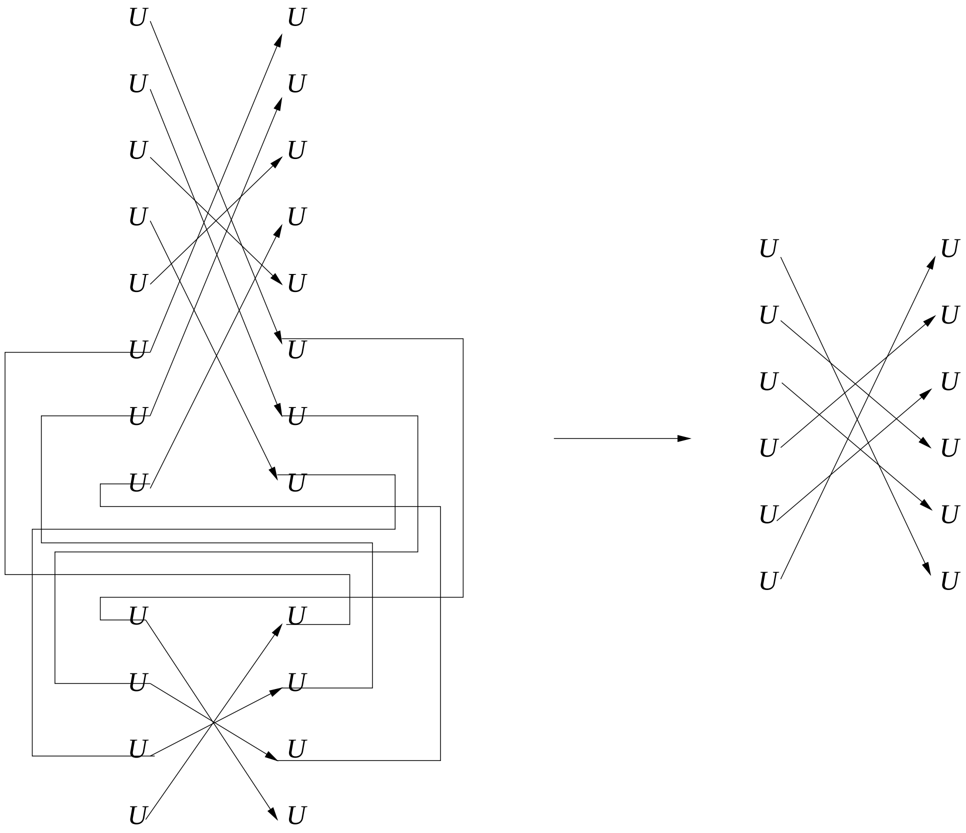
(iii)

Figure 39: Combinator B

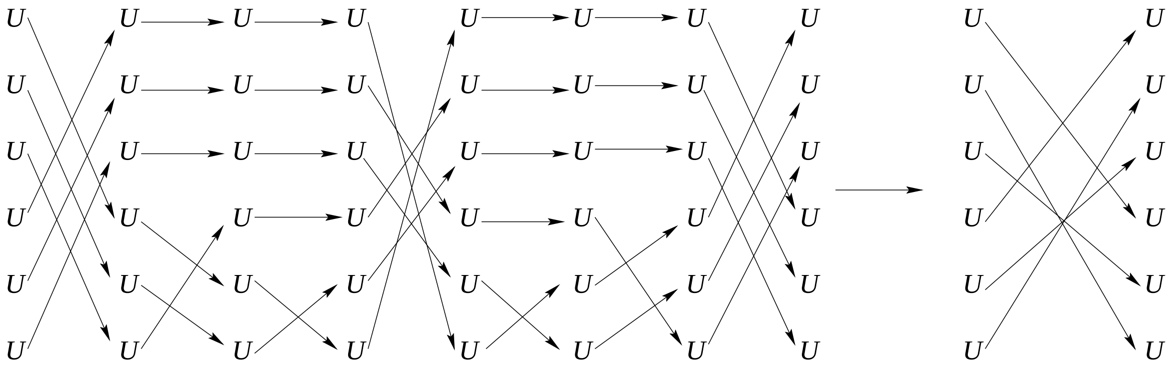


(i)

Figure 40: Combinator C

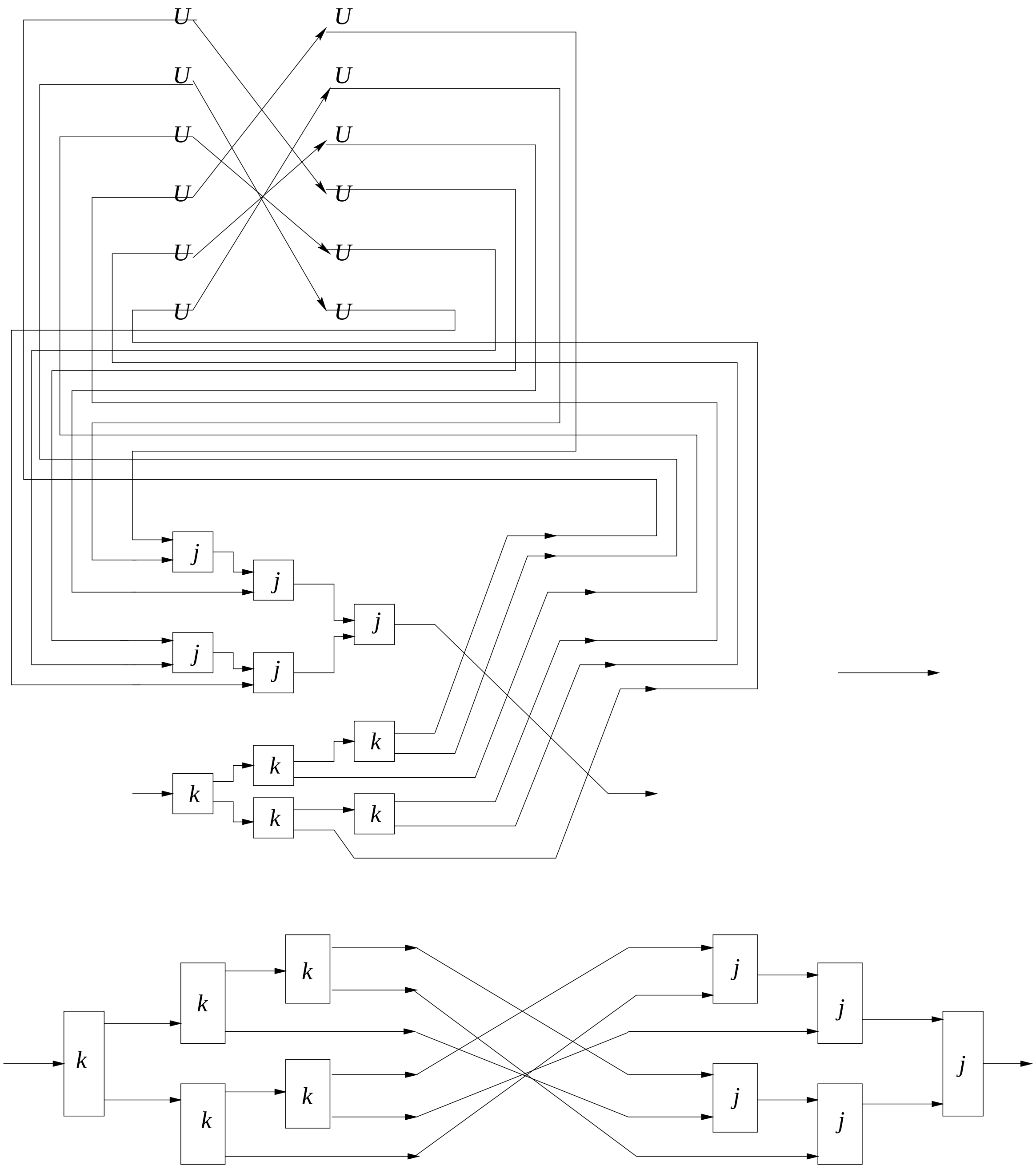


(ii)



(iii)

Figure 41: Combinator C



(iv)

Figure 42: Combinator C

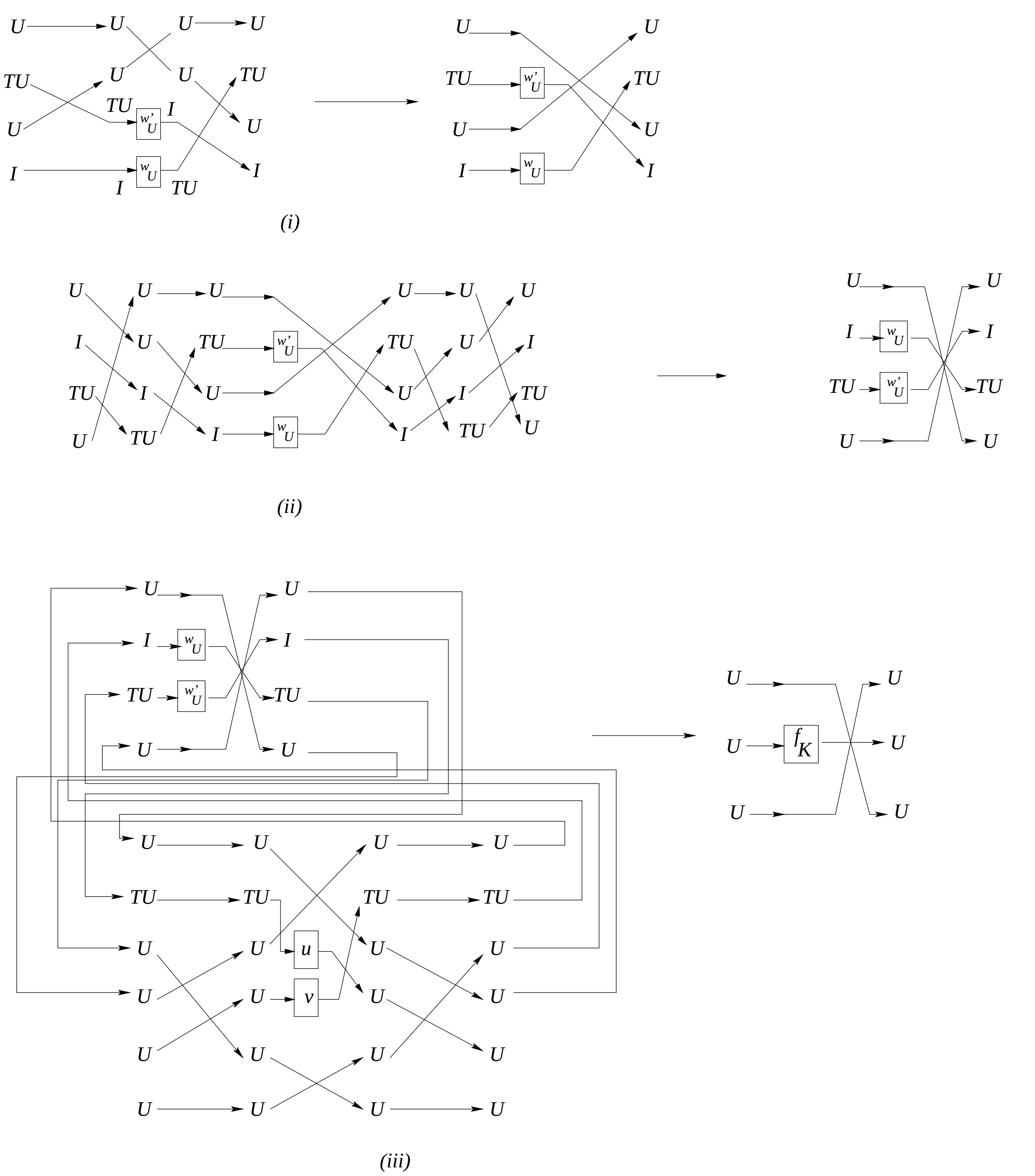
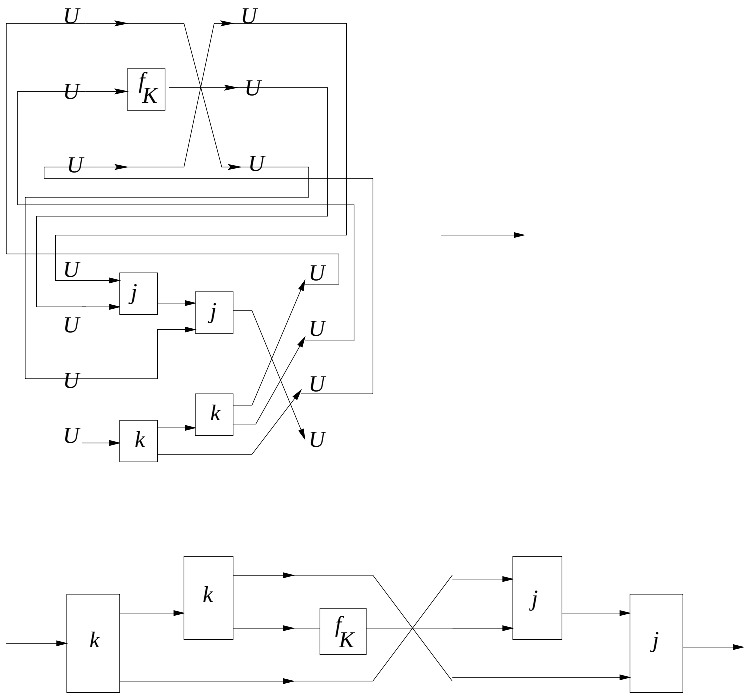
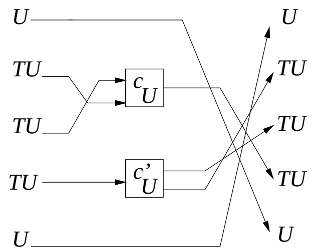


Figure 43: Combinator K

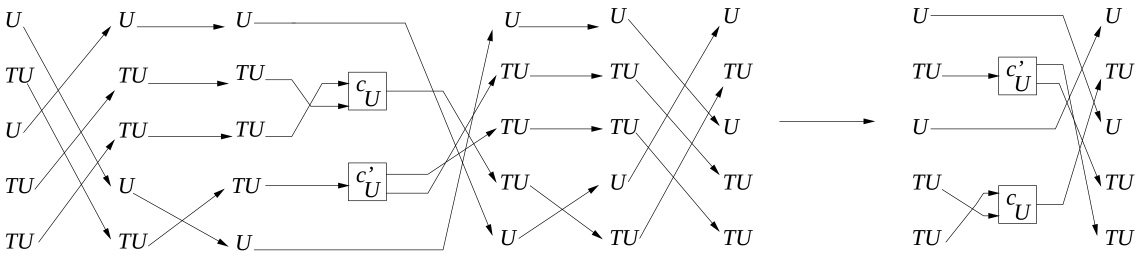


(iv)

Figure 44: Combinator K



(i)



(ii)

Figure 45: Combinator W

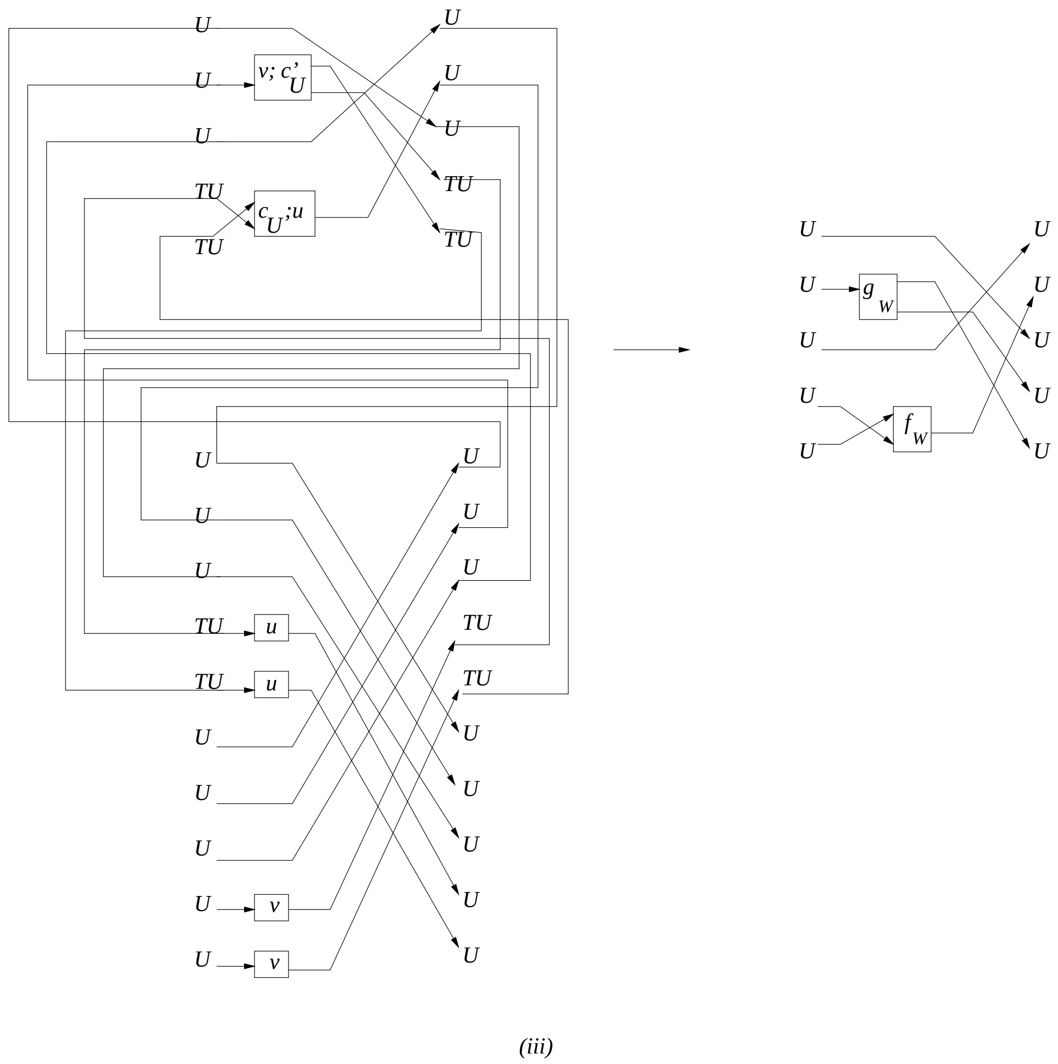
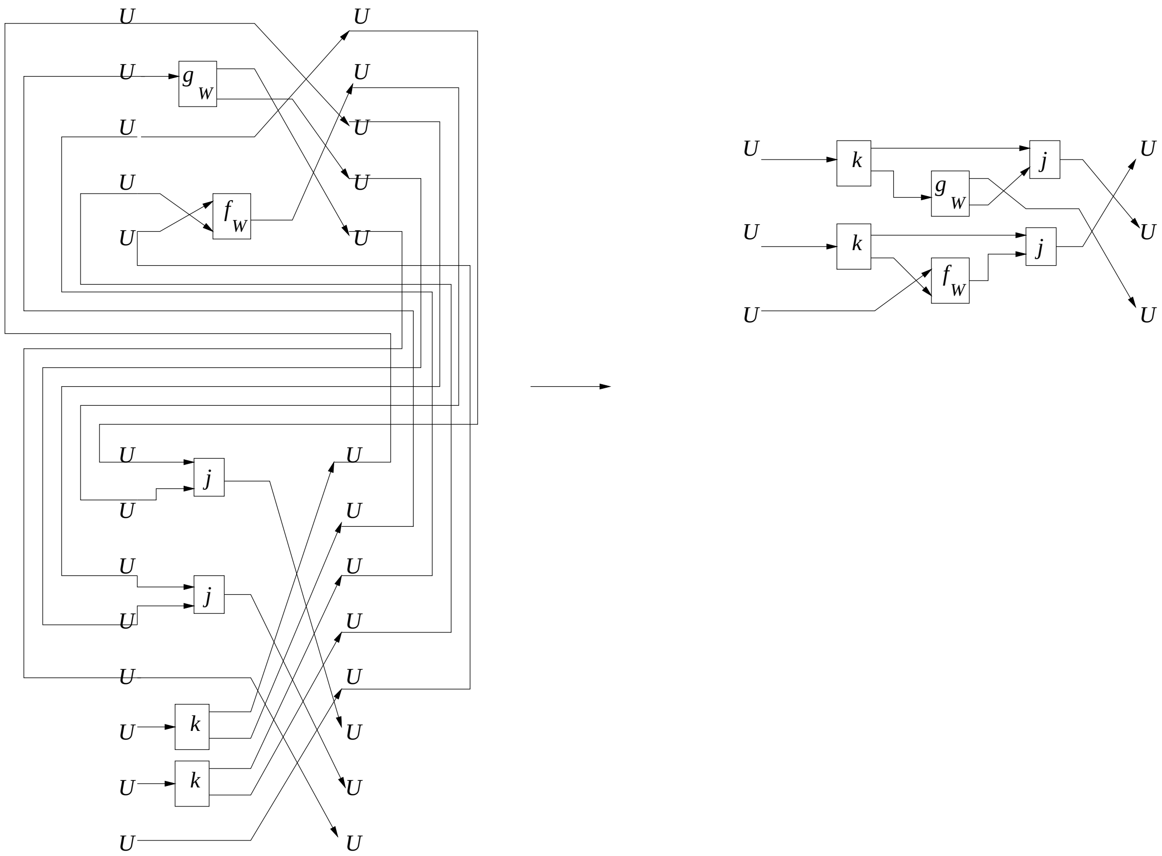
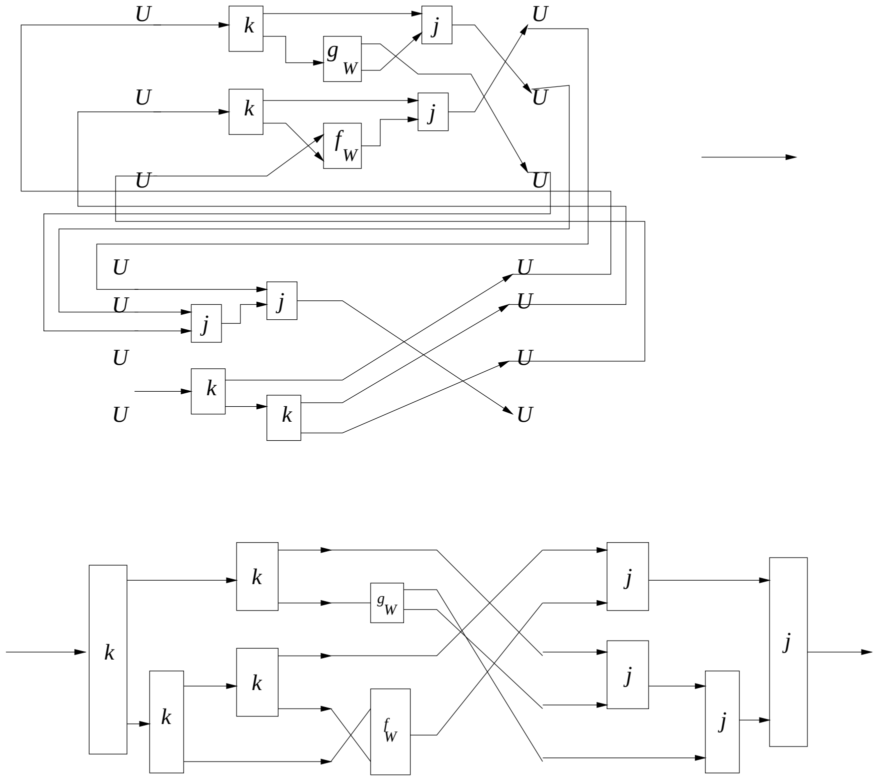


Figure 46: Combinator W



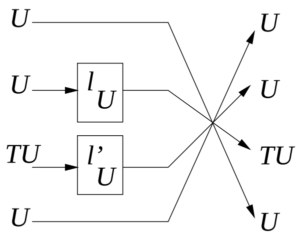
(iv)

Figure 47: Combinator W

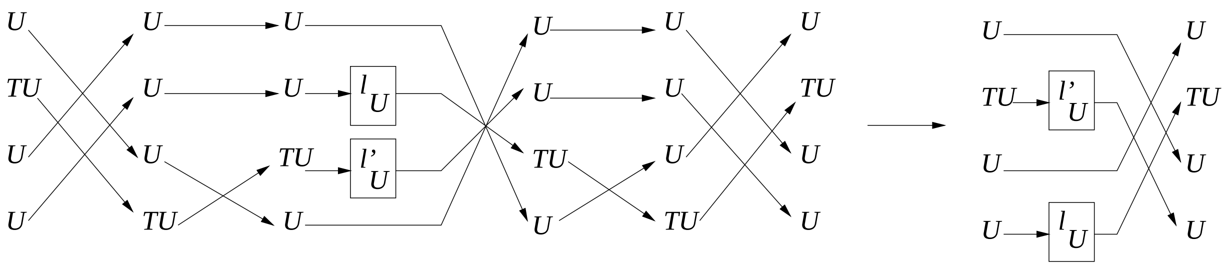


(v)

Figure 48: Combinator W

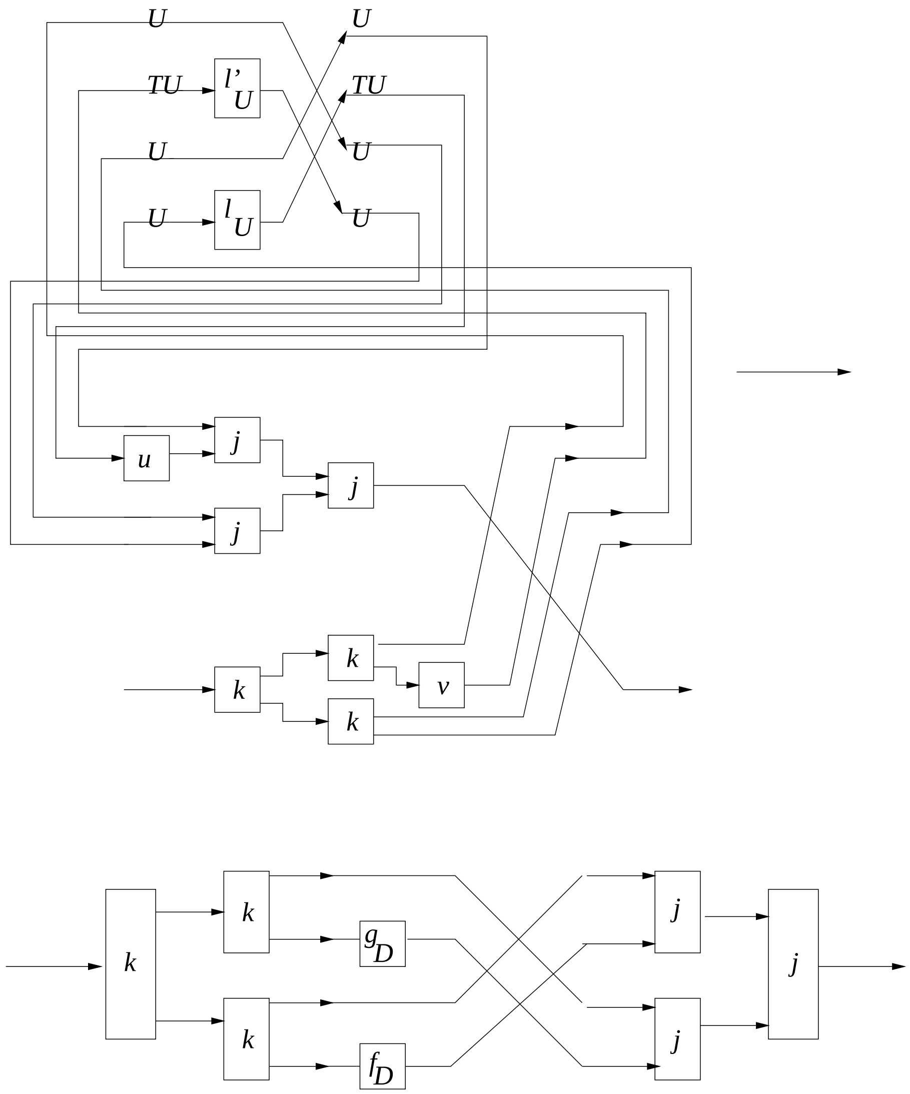


(i)



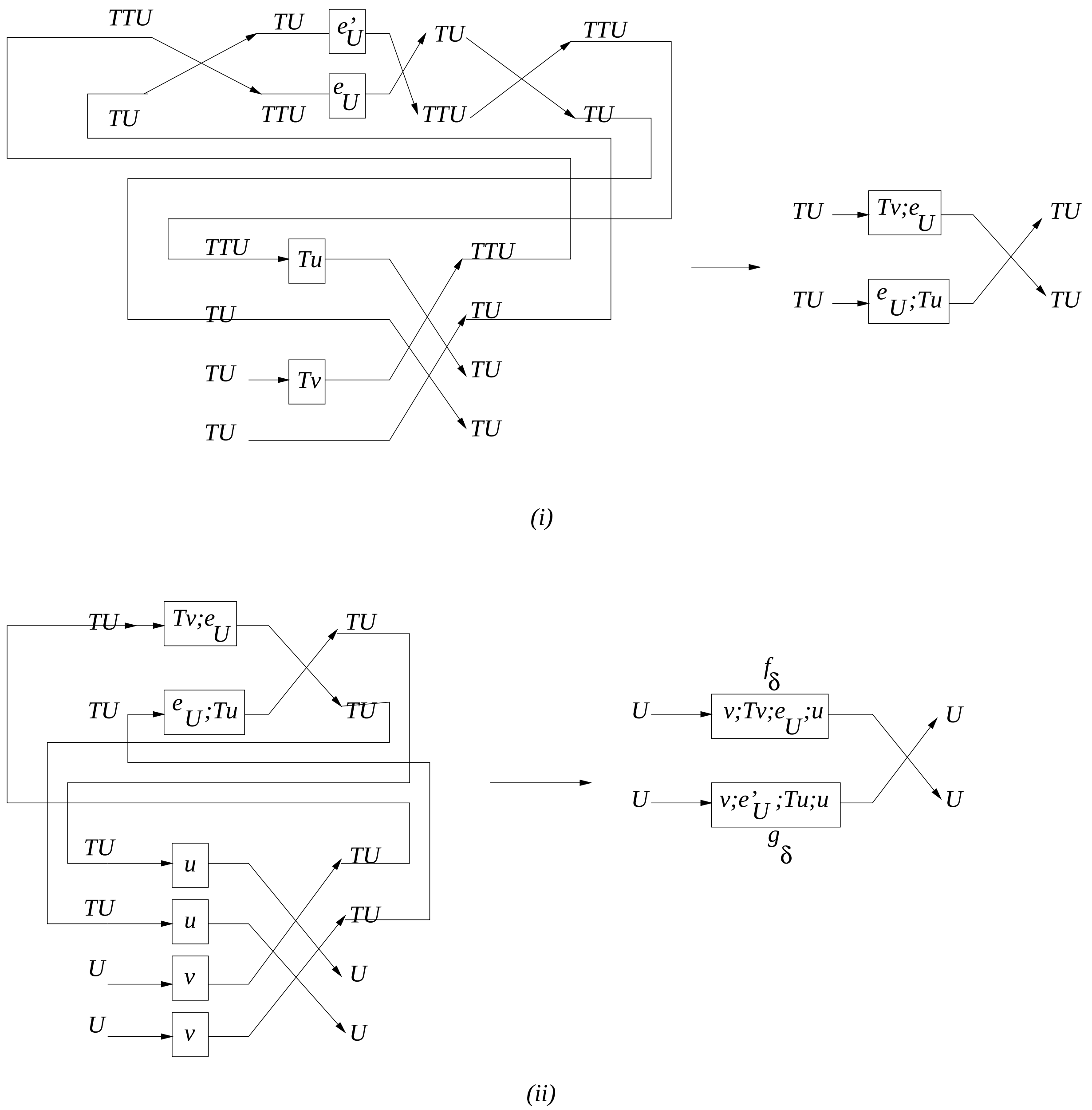
(ii)

Figure 49: Combinator D



(iii)

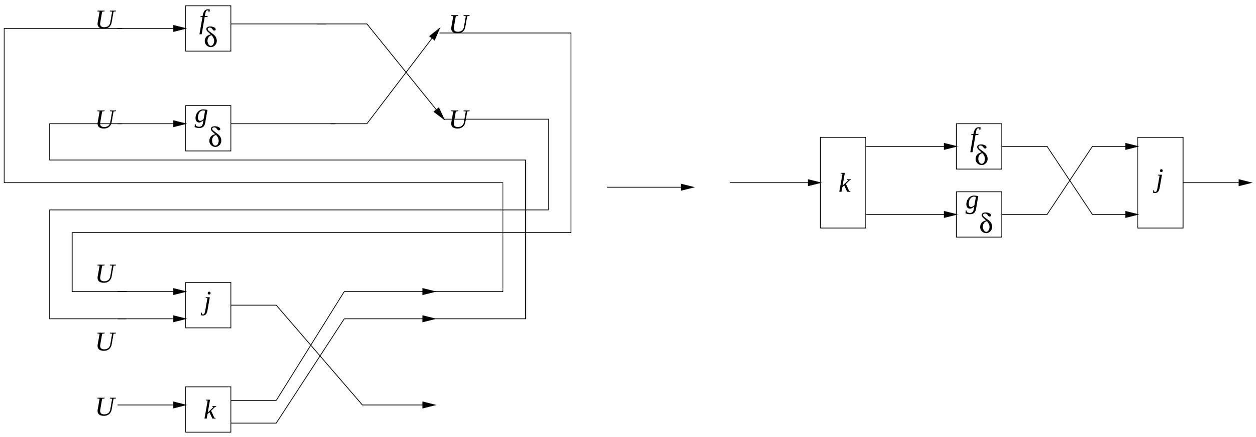
Figure 50: Combinator D



(i)

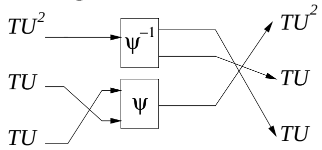
(ii)

Figure 51: Combinator δ

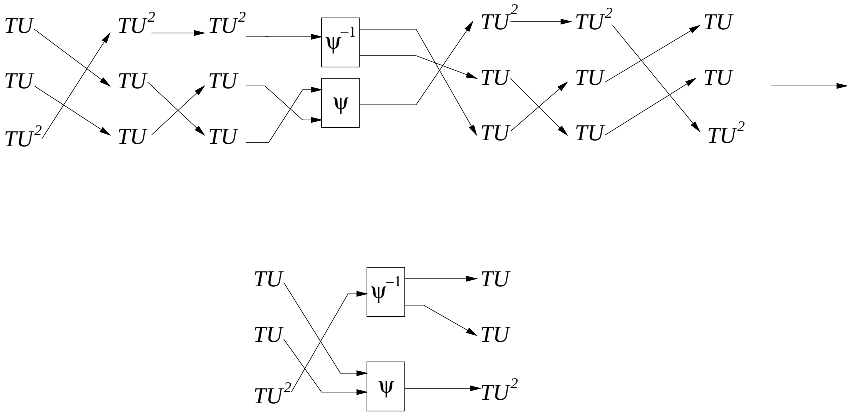


(iii)

Figure 52: Combinator δ

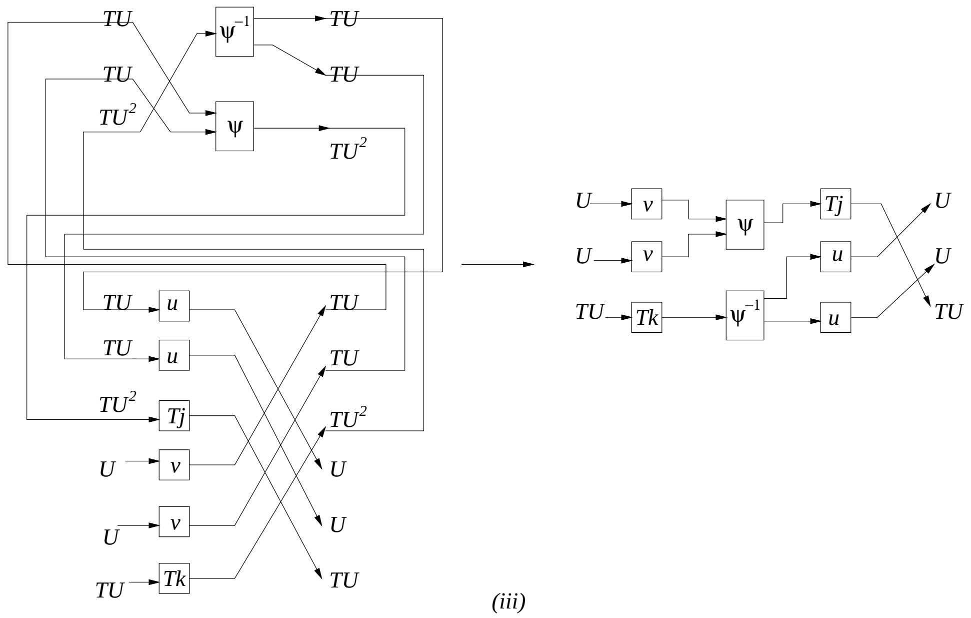


(i)

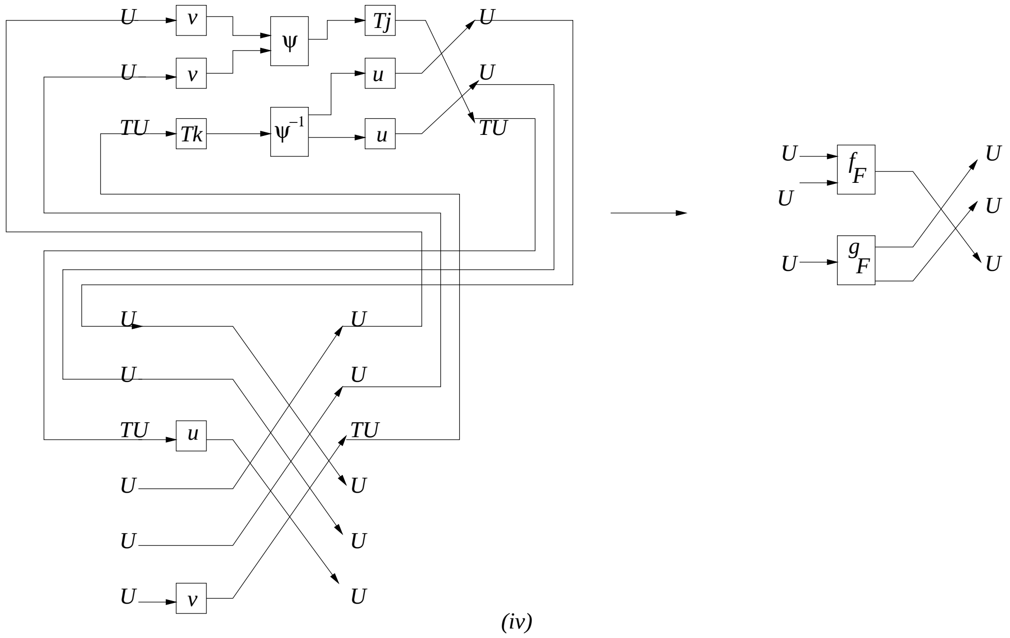


(ii)

Figure 53: Combinator F

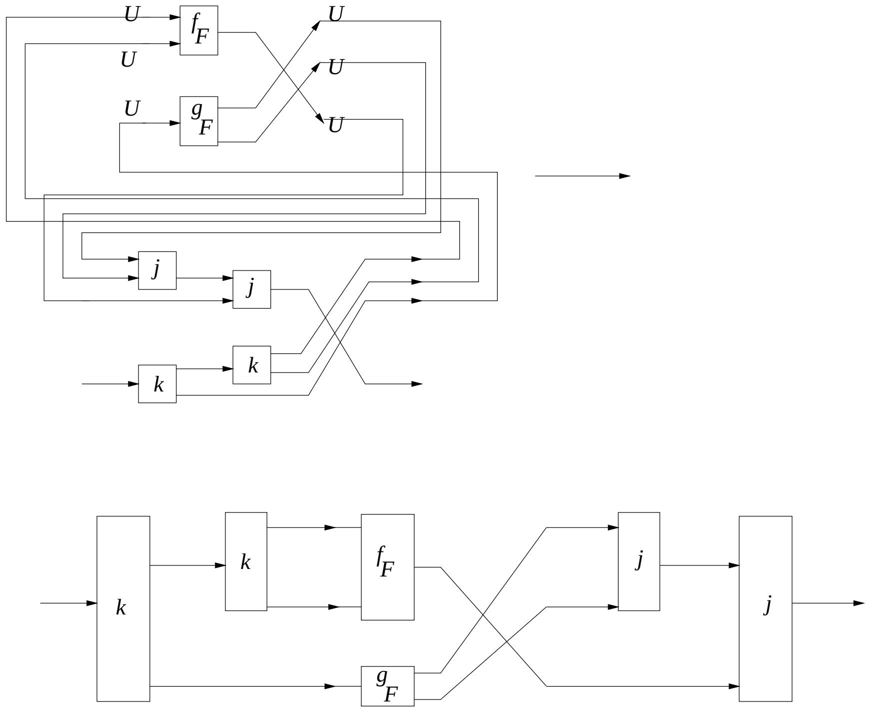


(iii)



(iv)

Figure 54: Combinator F



(v)

Figure 55: Combinator F

Appendix C

Reduction to Normal Forms

Recall the equations for the combinators:

$I{\bullet}x = x.$

$B{\bullet}x{\bullet}y{\bullet}z = x{\bullet}(y{\bullet}z).$

$C{\bullet}x{\bullet}y{\bullet}z = x{\bullet}z{\bullet}y.$

$W{\bullet}x{\bullet}\mathbf{!}y = x{\bullet}\mathbf{!}y{\bullet}\mathbf{!}y.$

$D{\bullet}x{\bullet}\mathbf{!}y = x{\bullet}y.$

$\delta{\bullet}\mathbf{!}x = \mathbf{!!}x.$

$F{\bullet}\mathbf{!}x{\bullet}\mathbf{!}y = \mathbf{!}(x{\bullet}y).$

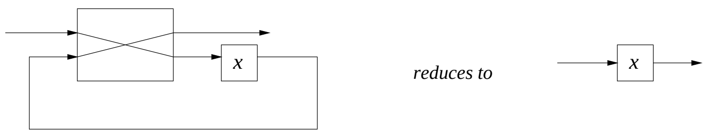


Figure 56: Identity Combinator I

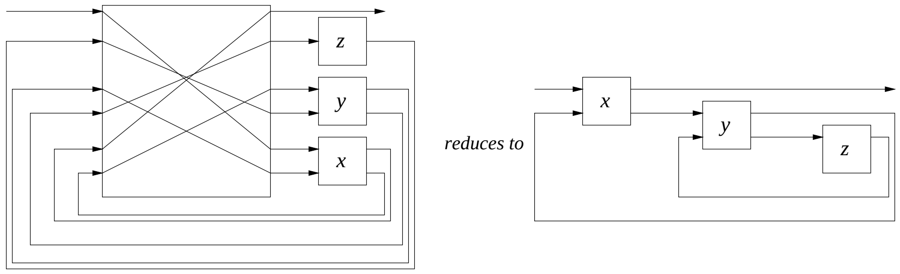


Figure 57: Composition Combinator B

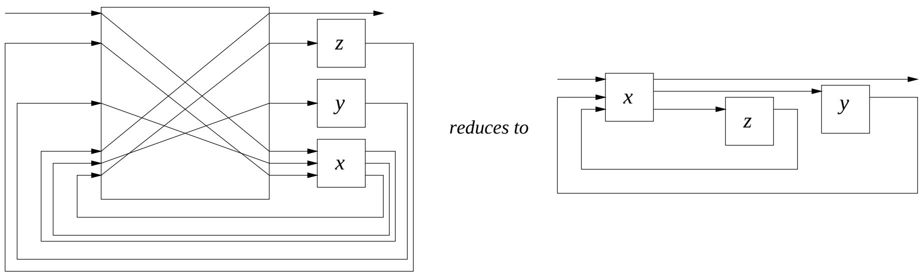


Figure 58: Exchange Combinator C

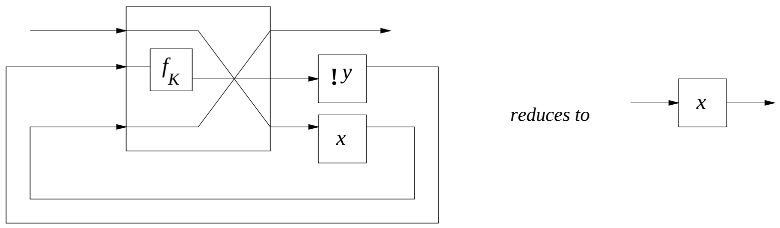


Figure 59: Weakening Combinator K

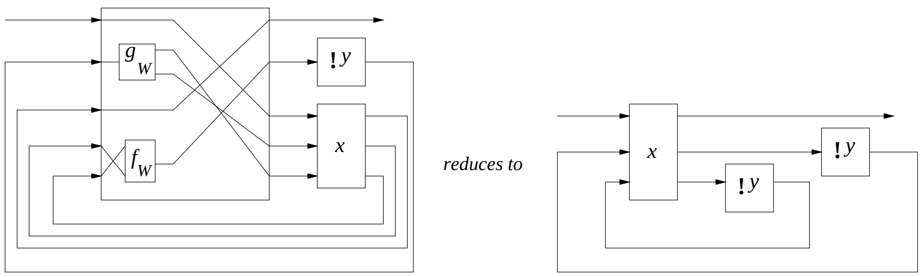


Figure 60: Contraction Combinator W

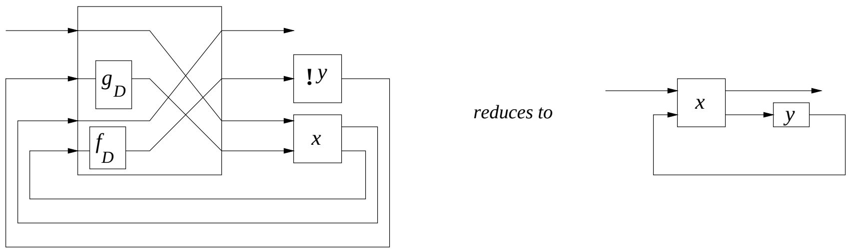


Figure 61: Dereliction Combinator D

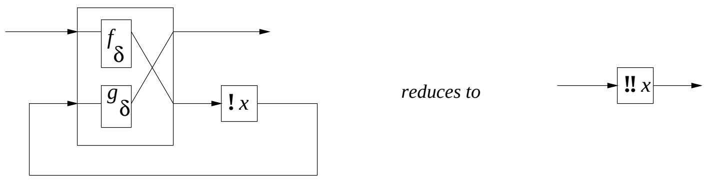


Figure 62: Comultiplication Combinator δ

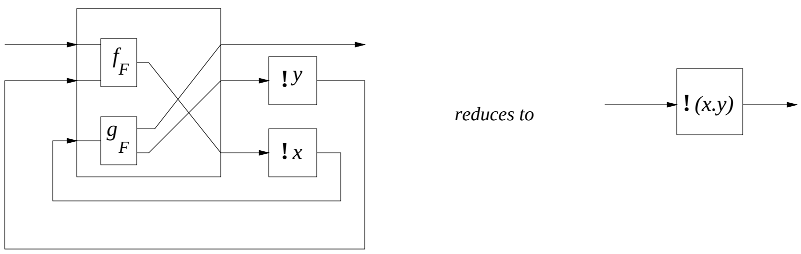


Figure 63: Functoriality Combinator F

Bibliography

[1] Abramsky, S. (1996), Retracing Some Paths in Process Algebra. In *CONCUR 96, Lecture Notes in Computer Science* **1119**, pp. 1-17. Springer-Verlag.

[2] Abramsky, S. (1997), *Interaction, Combinators and Complexity.* Lecture Notes, Siena, Italy.

[3] Abramsky, S., Blute, R. and Panangaden, P. (1997), Nuclear and Traced Ideals in Tensored *-categories. To appear in *J. Pure and Applied Algebra* .

[4] Abramsky, S. and Jagadeesan, R. (1994), Games and full completeness for Multiplicative Linear Logic. *Journal of Symbolic Logic* **59**, pp. 543-574.

[5] Abramsky, S. and Jagadeesan, R. (1994), New Foundations for the Geometry of Interaction. *Information and Computation* **111** (1), pp. 53-119.

[6] Abramsky, S. and Mellies, P. (1999), Concurrent Games and Full Completeness. In *Proc. of 14th LICS*, pp. 431-442. IEEE press.

[7] Arbib, M.A. and Manes E.G. (1980), Partially Additive Categories and Flow-diagram Semantics. *Journal of Algebra* **62**, pp. 203-227.

[8] Asperti, A., Danos, V., Laneve, C. and Regnier, L. (1994), Paths in the λ-calculus: Three years of communication without understanding. In *Proc. of Ninth LICS*, pp. 426-436, IEEE Press.

[9] Asperti, A. and Longo, G. (1991), *Categories, Types, and Structures.* MIT Press.

[10] Asperti, A. and Mairson, H.G. (1998), Parallel beta reduction is not elementary recursive. In *Proc. ACM Symposium on the Principles of Programming Languages.*

[11] Avron, A. (1988), The semantics and proof theory of linear logic. *Theoretical Comp. Science* **57** , pp. 161-184.

[12] Bainbridge, E.S., Freyd, P., Scedrov, A. and Scott, P.J. (1990), Functorial Polymorphism. *Theoretical Comp. Science* **70** , pp. 35-64.

[13] Barendregt, H.P. (1984), *The Lambda Calculus.* Studies in Logic, volume 103, North-Holland.

[14] Barr, M. (1979), **-autonomous Categories.* Springer Lecture Notes in Mathematics **752**.

[15] Barr, M. (1991), *-autonomous categories and linear logic. *Math. Structures in Computer Science* **1**, pp. 159-178.

[16] Barr, M. (1992), Algebraically Compact Functors. *J. Pure and Applied Algebra* **82**, pp. 211-231.

[17] Barr, M. (1999), *-autonomous categories: once more around the track. *Theory and Application of Categories* **6**, pp. 5-24.

[18] Benton, N., Bierman, G., de Paiva, V. and Hyland, M. (1992), Linear λ-calculus and categorical models revisited. In *Proc. of CSL '92, Lecture Notes in Computer Science* **702**, Springer-Verlag.

[19] Benton, N., Bierman, G., de Paiva, V. and Hyland, M. (1992), Term assignment for intuitionistic linear logic. *Technical Report 262*, Computer Laboratory, University of Cambridge.

[20] Bierman, G. (1994), *On Intuitionistic Linear Logic.* PhD thesis, Computer Laboratory, University of Cambridge.

[21] Bloom, S.L., Ésik, Z. (1993), *Iteration theories, the equational logic of iterative processes.* Springer-Verlag.

[22] Bloom, S.L., Ésik, Z. (1996), Fixed-point operations on ccc's. Part I. *Theoretical Comp. Science* **155** , pp. 1-38. Elsevier Science.

[23] Blute, R.F. (1993), Linear Logic, Coherence and Dinaturality. *Theoretical Computer Science* **115**, pp. 3-41.

[24] Blute, R.F. (1996), Hopf algebras and linear logic. *Math. Struct. in Comp. Science* **6**, pp. 189-217.

[25] Blute, R.F., Cockett, J.R.B., Seely, R.A.G. and Trimble, T.H. (1996), Natural deduction and coherence for weakly distributive categories. *J. Pure and Applied Algebra* **113**, pp. 229-296.

[26] Blute, R.F. and Scott, P.J. (1996), Linear Läuchli Semantics. *Annals of Pure and Applied Logic* **77**, pp. 101-142.

[27] Blute, R.F. and Scott, P.J. (1998), The Shuffle Hopf Algebra and Noncommutative Full Completeness. *Journal of Symbolic Logic* **63** (4), pp. 1413-1436.

[28] Borceux, F. (1994), *Handbook of Categorical Algebra.* Encyclopedia of Mathematics and its Applications, volume 50, Cambridge University Press.

[29] Bourbaki, N. (1996), *General Topology, Part 1.* Addison-Wesley.

[30] Braüner, T. (1996), *An Axiomatic Approach to Adequacy.* PhD thesis, University of Aarhus, Denmark.

[31] Cockett, J.R.B. and Seely, R.A.G. (1997), Weakly Distributive Categories. *Journal of Pure and Applied Algebra,* **114**, pp. 133-173.

[32] Cockett, J.R.B. and Seely, R.A.G. (1997), Proof theory for full intuitionistic linear logic, bilinear logic, and mix categories. *Theory and Applications of Categories,* **3** (5), pp. 85-131.

[33] Danos, V. (1990), *La logique linéaire appliquée à l'étude de divers processus de normalisation et principalement du λ-calcul.* PhD thesis, Université Paris VII.

[34] Danos, V. and Regnier, L. (1989), The Structure of Multiplicatives. *Arch. Math. Logic* **28**, pp. 181-203.

[35] Danos, V. and Regnier, L. (1993), Local and Asynchronous beta-reduction (An Analysis of Girard's Execution Formula). In *Proc. of Eighth LICS*, pp. 296-306, IEEE Press.

[36] Danos, V. and Regnier, L. (1995), Proof-nets and Hilbert Spaces. In J-Y. Girard et al. Eds. *Advances in Linear Logic*, London Math. Soc. LNS, **222**, pp. 307–328.

[37] Devarajan, P., Hughes, D., Plotkin, G. and Pratt, V. (1999), Full completeness of the multiplicative linear logic of Chu spaces. In *Proc. of the 14th LICS*, pp. 234-243, IEEE Press.

[38] Fleury, A. and Rétoré, C. (1994), The MIX Rule, *Math. Structures in Computer Science* **4**, pp. 273-285.

[39] Freyd, P. and Scedrov, A. (1991), *Categories and Allegories.* North-Holland Mathematical Library, vol. 39, Elsevier Science B.V.

[40] Gelfand, I.M. and Naimark, M. (1943), On the embedding of normed rings into the ring of operators in Hilbert space. *Mat. Sb.* **12**, pp. 197-213.

[41] Geroch, R. (1985), *Mathematical Physics.* Chicago Lectures in Physics, The University of Chicago Press.

[42] Ghez, P., Lima, R. and Roberts, J.E. (1985), W^*-Categories. *Pacific Journal of Mathematics* **120** (1).

[43] Girard, J.-Y. (1971), Une Extension de l'Interprétation de Gödel à l'Analyse, et son Application à l'Elimination des Coupures dans l'Analyse et la Théorie

des Types. In *Proceedings of the Second Scandinavian Logic Symposium,* J.E. Fenstad, Ed., pp. 63-92, North-Holland, Amsterdam.

[44] Girard, J.-Y. (1987), Linear Logic. *Theoretical Comp. Science* **50** (1), pp. 1-102.

[45] Girard, J.-Y. (1988), Geometry of Interaction II: Deadlock-free Algorithms. In *Proc. of COLOG'88, Lecture Notes in Computer Science* **417**, pp. 76–93. Springer-Verlag.

[46] Girard, J.-Y. (1989), Towards a Geometry of Interaction. *Contemp. Math.,* AMS, **92**, pp. 69-108.

[47] Girard, J.-Y. (1989a) Geometry of Interaction I: Interpretation of System F. In *Proc. Logic Colloquium 88,* pp. 221–260. North Holland.

[48] Girard, J.-Y. (1995), Geometry of Interaction III: Accommodating the Additives. *Advances in Linear Logic,* J.Y. Girard et al. Eds., London Math. Soc. LNS **222**, pp. 329–389, Cambridge University Press.

[49] Girard, J.-Y. (1995), Linear Logic: its syntax and semantics, in *Advances in Linear Logic*, J.Y. Girard et al. Eds., London Math. Soc. Series **222**, Cambridge University Press, pp. 1-42.

[50] Girard, J.-Y., Lafont, Y. and Taylor, P. (1990), *Proofs and Types.* Cambridge tracts in theoretical computer science, Volume 7, Cambridge University Press.

[51] Girard, J.Y., Scedrov, A. and Scott, P.J. (1991), Normal forms and cut-free proofs as natural transformations. In *Logic from Computer Science*, Mathematical Science Research Institute Publications **21**, pp. 217-241.

[52] Giry, M. (1981), A categorical approach to probability theory. In *Springer Lecture Notes in Mathematics* **915**, pp. 68-85.

[53] Gonthier, G., Abadi, M. and Lévy, J.-J. (1992), The geometry of optimal lambda reduction. In *Proc. Ninth ACM Symposium on Principles of Programming Languages,* pp. 15-26.

[54] Hamano, M. (1998), *Z-modules and full completeness of multiplicative linear logic*. Manuscript.

[55] Hamano, M. (1998), Pontrajagin Duality and Full Completeness for MLL. To appear.

[56] Hankin, C. (1994), *Lambda Calculi, A Guide for Computer Scientists*, Gradate Texts in Computer Science, volume 3, Clarendon Press, Oxford.

[57] Harnik, V. and Makkai, M. (1992), Lambek's categorical proof theory and Läuchli's abstract realizability. *Journal of Symbolic Logic* **57**, pp. 200-230.

[58] Hasegawa, M. (1997), *Models of Sharing Graphs, A Categorical Semantics of* `let` *and* `letrec`. PhD thesis, University of Edinburgh.

[59] Hasegawa, M. (1997), Recursion from Cyclic Sharing : Traced Monoidal Categories and Models of Cyclic Lambda Calculus, TLCA'97, *Springer Lecture Notes in Computer Science* **1210**, pp. 196-213.

[60] Hasegawa, M. (1998), *Categorical glueing and logical predicates for models of linear logic*, Manuscript.

[61] Higgs, D. (1980), Axiomatic Infinite Sums-An Algebraic Approach to Integration Theory. *Contemp. Math.* **2**, pp. 205-213.

[62] Hindley R.J. and Seldin J.P. (1986), *Introduction to Combinators and λ-calculus*. London Mathematical Society Student Texts **1**, Cambridge University Press.

[63] Howard, W.A. (1980), The formulae-as-types notion of construction. In *To H.B. Curry: Essays on combinatory logic, lambda calculus and formalism*, J.P. Seldin and J.R. Hindley Eds., pp. 479-490, Academic Press.

[64] Hyland, M and Ong, L. (1993), *Full completeness for multiplicative linear logic without the mix-rule*, electronically distributed manuscript.

[65] Hyland, M., Robinson, E.P. and Rosolini G. (1989), Algebraic types in per models. In *Fifth conference on Mathematical Foundations in Programming Semantics*, *Lecture Notes in Computer Science* **442**, pp. 333-350, Springer-Verlag.

[66] Joyal, A. and Street, R. (1991), The geometry of tensor calculus I. *Advances in Math.* **88**, pp. 55-113.

[67] Joyal, A., Street, R. and Verity, D. (1996), Traced Monoidal Categories. *Math. Proc. Camb. Phil. Soc.* **119**, pp. 447-468.

[68] Kadison, R.V. and Ringrose, J.R. (1983), *Fundamentals of the Theory of Operator Algebras*, Volume 1. Academic Press.

[69] Kahn, G. (1974), The semantics of a simple language for parallel programming. In *Information Processing 74*, J.L. Rosenfeld Ed., pp. 471-475, North-Holland, Amsterdam.

[70] Kahn, G. and MacQueen D. (1977), Coroutines and networks of parallel processes. In *Proceeding of IFIP Congress*, North-Holland, Amsterdam.

[71] Kassel, C. (1995), *Quantum Groups.* Graduate Texts in Mathematics **155**, Springer-Verlag.

[72] Kathail, V.K. (1990), *Optimal Interpreters for Lambda-calculus Based Functional Programming Languages.* PhD thesis, MIT.

[73] Kelly, G.M. and Laplaza, M.L. (1980), Coherence for compact closed categories. *J. Pure and Applied Algebra* **19**, pp. 193-213.

[74] Kelly, G.M. and Mac Lane S. (1971), Coherence in closed categories. *J. Pure and Applied Algebra* **1** (1), pp. 97-140.

[75] Kuroš, A.G. (1963), Direct Decomposition in Algebraic Categories. *Amer. Math. Soc. Transl. Ser.* 2, **27**, pp. 231-255.

[76] Lambek, J. (1969), Deductive systems and categories II. *Springer Lecture Notes in Mathematics* **87**, Springer-Verlag.

[77] Lambek, J. (1980), From λ-calculus to cartesian closed categories. In *To H.B. Curry: Essays on combinatory logic, lambda calculus and formalism*, J.P. Seldin and J.R. Hindley Eds., pp. 375-402, Academic Press.

[78] Lambek, J. and Scott, P.J. (1986), *Introduction to higher order categorical logic*. Cambridge studies in advanced mathematics 7, Cambridge University Press.

[79] Lamping, J. (1990), An algorithm for optimal lambda-calculus reductions. In *Proc. of the Seventeenth ACM Symposium on Principles of Programming Languages*, pp. 16-30, ACM Press.

[80] Lawvere, F.W. (1963), *The category of probabilistic mappings*. Preprint.

[81] Lévy, J.-J. (1980), Optimal reductions in the lambda calculus. In J.P. Hindley and J.R. Seldin, Eds., *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, pp. 159-191. Academic Press.

[82] Loader, R. (1994), Linear Logic, totality and full completeness. In *Proc. Logic in Computer Science (LICS) 1994*. IEEE Press.

[83] Loader, R. (1994), *Models of Lambda Calculi and Linear Logic : Structural, equational and proof-theoretic characterisations*. PhD Thesis, St. Hugh's College, Oxford.

[84] Mackie, I. (1994), *The Geometry of Implementation*. PhD Thesis, University of London, Imperial College of Science, Technology and Medicine.

[85] Mac Lane, S. (1998), *Categories for the Working Mathematician*. Graduate Texts in Mathematics 5, 2nd edition, Springer-Verlag.

[86] Mac Lane, S. and Moerdijk, I. (1992), *Sheaves in Geometry and Logic, A first introduction to topos theory*. Universitext, Springer-Verlag.

[87] Makkai, M. and Reyes, G.E. (1976), Model theoretic methods in the theory of topoi and related categories, I,II. *Bull. Acad. Polon. Sci.* **24**, pp. 379-392.

[88] Malacaria, P. and Regnier. L. (1991), Some Results on the Interpretation of λ-calculus in Operator Algebras. In *Proc. of Sixth LICS,* pp. 63-72, IEEE Computer Society Press.

[89] Manes, E.G. (1992), *Predicate Transformer Semantics.* Cambridge Tracts in Theoretical Computer Science 33, Cambridge University Press.

[90] Manes, E.G. and Arbib, M.A. (1986), *Algebraic Approaches to Program Semantics.* Springer-Verlag.

[91] Manes E.G. and Benson, D.B. (1985), The Inverse Semigroup of a Sum-Ordered Semiring. *Semigroup Forum* **31**, pp. 129-152.

[92] Milner, R. (1975), Processes, a mathematical model of computing agents. *Logic colloquium, Bristol 1973*, pp. 157-174, North-Holland, Amsterdam.

[93] Mitchell, J.C. (1996), *Foundations for Programming Languages.* MIT Press.

[94] Moggi, E. (1991), Notions of Computation and Monads. *Infor. and Comput.* **93**, pp. 55-93.

[95] Panangaden, P. (1997), *Stochastic Techniques in Concurrency.* Lecture Notes, BRICS, Aarhus, Denmark.

[96] Pareigis, B. (1973), *Categories and Functors.* Pure and Applied Mathematics, volume 39, Academic Press.

[97] Plotkin, G.D. (1977), LCF considered as a programming language. *Theoretical Comp. Science* **5** , pp. 223-255.

[98] Plotkin, G.D. and Simpson, A.K. (1998), *Properties of Fixed Points in Axiomatic Domain Theory.* Manuscript.

[99] Regnier, L. (1992), *Lambda-calcul et Réseaux*, PhD Thesis, Université Paris VII.

[100] Reynolds, J.C. (1974), Towards a theory of type structure. In *Paris Colloquium on Programming, Lecture Notes in Computer Science* **19**, Springer-Verlag.

[101] Seely, R.A.G. (1989), Linear logic, *-autonomous categories and cofree coalgebras. in J.W. Gray and A. Scedrov, Eds., *Categories in Computer Science and Logic*, Contemp. Math. **92**. American Math. Soc.

[102] Selinger, P. (1999), Categorical Structure of Asynchrony. *Electronic Notes in Theoretical Computer Science*, **20**. Elsevier Science B.V.

[103] Scott, P.J. (1999), Some Aspects of Categories in Computer Science. In *Handbook of Algebra*, Volume 2, M. Hazewinkel Ed., Elsevier.

[104] Tan, A.M. (1997), *Full Completeness for Models of Linear Logic*. PhD thesis, Cambridge.

[105] Troelstra, A.S. (1992), *Lectures on Linear Logic*. Center for the Study of Language and Information Lecture Notes No. 29.