Geometry of Interaction and the Dynamics of Proof Reduction: a tutorial

Esfandiar Haghverdi [a], Philip Scott [b,1]

[a] *School of Informatics, Indiana University*
Bloomington, IN 47408, USA

[b] *Department of Mathematics & Statistics, University of Ottawa,*
585 King Edward, Ottawa, Ontario, K1N 6N5, CANADA

Abstract

Girard's Geometry of Interaction (GoI) is a program that aims at giving mathematical models of algorithms independently of any extant languages or computing models, thus making it possible to prove general theorems about algorithms. In the context of proof theory, where one views algorithms as proofs and computation as cut-elimination, this program translates to providing a mathematical modelling of the dynamics of cut-elimination. The kind of logics we deal with, such as Girard's linear logic, are resource sensitive and have their proof-theory intimately related to various monoidal (tensor) categories. The GoI interpretation of dynamics aims to develop an algebraic/geometric theory of invariants for information flow in networks of proofs.

Key words: Linear Logic, Cut Elimination, Monoidal and Traced Monoidal Categories, Geometry of Interaction,, Partial Trace, Abstract Orthogonality, Execution Formula.

Contents

Email addresses: ehaghver@indiana.edu, phil@mathstat.uottawa.ca (Philip Scott).

[1] Research supported by a Discovery Grant from NSERC, Canada.

1 Introduction

In the 1930's, Gerhard Gentzen developed a profound approach to Hilbert's proof theory, in which formal laws for deriving logical entailments $\Gamma \vdash \Delta$ (i.e. premisses Γ entail conclusions Δ) were carefully systematized, breaking the laws of logic into three groups: (i) the Axiom and Cut-Rule, (ii) Structural Rules, and (iii) Logical Rules. Gentzen's work revealed the hidden symmetries in logical syntax, and his remarkable Cut-Elimination Theorem, one of the deepest in logic, has also had considerable significance for theoretical computer science.

In these lectures we shall give the background, both logical and categorical, to a remarkable new approach to Gentzen's work, stemming from J-Y Girard's introduction of Linear Logic in 1987 [Gi87]. Linear Logic, a radical analysis of the Gentzen rules of traditional logic, is based upon studying the use of resources in these rules, e.g. in duplicating and eliminating premisses and conclusions in a logical inference. We may think of proofs as *dynamical systems*, with inputs and outputs being the hypotheses and the conclusions respectively, and we think of the rules involved in transforming, i.e. in rewriting, proof trees (in Gentzen's Cut-Elimination Algorithm) as *interaction* between these dynamical systems. We are looking for mathematical invariants for the dynamics of these systems.

Girard's *Geometry of Interaction* (GoI) project began in the late 1980s [Gi89,Gi89a]. The first paper on GoI was set in an operator algebraic context: proofs were interpreted as operators on the Hilbert space of square summable sequences. The GoI interpretation of cut-elimination was given by a finite sum, which was finite due to nilpotency of the summands. This already pointed to the usefulness of the GoI view of logic: one has a degree of nilpotency that measures the complexity of cut-elimination (=computation). This also inspired a different line of work in GoI research, the so called path-semantics with relationships to lambda calculus, a fundamental model of computation [DR95].

One might ask: why is this important? The answer lies in realizing that one way to model computation is precisely as an instance of Gentzen's algorithm. We search for mathematical models of this dynamical process of cut-elimination, expecting that such an analysis will shed deep light on the very nature of computation and its complexity. Indeed, there are connections of the whole project with complexity, as we mention in Remark 5.2 in these notes.

The early work on understanding the categorical framework of GoI was begun in lectures of Abramsky and of Hyland in the early 90's. This brought the notion of abstract trace (in the sense of Joyal, Street, and Verity [JSV96]) into the picture. Work by Hyland, by Abramsky [Abr96] and later by us [AHS02,HS04a] has emphasized the role of abstract traces in modelling cut-elimination in GoI. Our categorical modelling of GoI has recently led us to the use of $*$-categories (see Section 8), already familiar to theoretical physicists in the work by Doplicher, Roberts and others. This approach to GoI offers a potential connection to the literature in several areas of interest in mathematics and physics, for example to knot theory, where trace appears under the name *braid closure* (cf. [Abr07]). The most recent work by Girard [Gi08], makes use of type II_1 von Neumann algebras to offer a new interpretation of GoI, although the categorical meaning is totally open. It is our strong hope and belief that the categorical and logical structures outlined in these notes will be conducive to non-trivial and productive connections with applications to physics.

2 From Monoidal Categories to *-Autonomy

2.1 Monoidal Categories

Monoidal (tensor) categories are a fundamental mathematical structure arising in many areas of mathematics, theoretical computer science and physics, and increasingly in mathematical logic. The subject is a vast one, so we will just include definitions and examples relevant to these lectures. For general background, the reader is referred to standard category theory texts [Bor93,Mac98]. For general surveys of monoidal categories in relation to categorical and linear logics, see the articles [Sc00,BS04,Mel07] and further references given below.

Definition 2.1 A *monoidal (or tensor) category* $(\mathcal{C}, \otimes, I, \alpha, \ell, r)$ is a category $\mathcal{C}$, with functor $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$, unit object $I \in ob(\mathcal{C})$, and specified isomorphisms (natural in A, B, C): $\alpha_{ABC} : (A \otimes B) \otimes C \xrightarrow{\cong} A \otimes (B \otimes C)$, $\ell_A : I \otimes A \xrightarrow{\cong} A$, $r_A : A \otimes I \xrightarrow{\cong} A$ satisfying the following equations (in diagrammatic form):

$\ell_I = r_I : I \otimes I \to I$, as well as:

$$\begin{tikzcd}
(A\otimes I)\otimes C \arrow[r, "\alpha"] \arrow[d, "r_A\otimes id_C"'] & A\otimes (I\otimes C) \arrow[d, "id_A\otimes \ell_C"] \\
A\otimes C \arrow[r, phantom, "="] & A\otimes C
\end{tikzcd}
\qquad
\begin{tikzcd}
A(B(CD)) & (AB)(CD) \arrow[l, "\alpha"'] & ((AB)C)D \arrow[l, "\alpha"'] \arrow[d, "\alpha\otimes id_D"] \\
A((BC)D) \arrow[u, "id_A\otimes \alpha"] & & (A(BC))D \arrow[ll, "\alpha"']
\end{tikzcd}$$

where we omit $\otimes$'s and subscripts in the second diagram for typographical reasons. This latter diagram is known as the Mac Lane pentagon. It expresses an equality between the two a priori different natural isomorphisms between $((A\otimes B)\otimes C)\otimes D$ and $A\otimes(B\otimes(C\otimes D))$.

Monoidal structure is not generally unique nor canonical: there may be several (nonisomorphic) tensor structures on the same category. An interesting special case is when the isos α, ℓ, r are all identity morphisms. In that case, we say the monoidal category is *strict*.

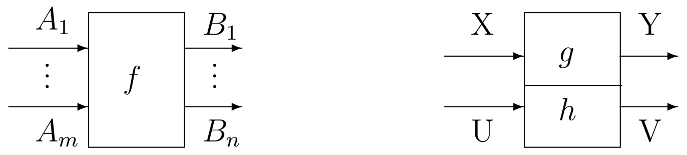
Definition 2.2 A *strict monoidal category* is a category $\mathcal{C}$ with a functor $\otimes : \mathcal{C}\times\mathcal{C}\to\mathcal{C}$ and $I \in ob(\mathcal{C})$ satisfying the following equations:

- $(A\otimes B)\otimes C = A\otimes(B\otimes C)$.
- $A\otimes I = A = I\otimes A$.
- $(f\otimes g)\otimes h = f\otimes(g\otimes h)$ for any arrows f, g, h.
- $f\otimes id_I = f = id_I\otimes f$, for any arrow $f : A\to B$.

Many concrete examples of strict monoidal categories arise in knot theory, quantum groups and related areas (e.g. [KRT97]). More generally, the Mac Lane Coherence Theorem [Mac98] states that every monoidal category is equivalent to a strict one. This essentially says that in an arbitrary monoidal category $\mathcal{C}$, every "formal" diagram of arrows (from a source object to a target object) which is built from instances of the maps α, ℓ, r under the monoidal category operations automatically commutes. Thus, without loss of generality (up to equivalence) we can assume our monoidal categories are strict. Notice in a strict monoidal category, the objects form a monoid (= semigroup with unit) under $\otimes$.

From now on we write $(\mathcal{C}, \otimes, I)$ for monoidal categories, omitting the remaining structure maps α, ℓ, r when it is clear. We introduce some standard graphical notation for arrows in Figure 1.

In any monoidal category $(\mathcal{C}, \otimes, I)$, we can define the *monoid of scalars* to be $\mathcal{C}(I, I)$. For example, in the monoidal category $(\mathbf{Vec}, \otimes, I)$ of $\mathbf{k}$-vector spaces and linear maps, with the usual notion of algebraic tensor product, and $I = \mathbf{k}$ (the base field) observe $\mathbf{Vec}(I, I) \cong I$. The following result is from [KL80] (see also [Abr05]).



$$A_1 \otimes \cdots \otimes A_m \xrightarrow{f} B_1 \otimes \cdots \otimes B_n \qquad X \otimes U \xrightarrow{g\otimes h} Y \otimes V$$

Fig. 1. Pictorial Representation of Morphisms

Proposition 2.3 (Kelly-LaPlaza) *In any monoidal category, the scalars form a commutative monoid.*

There are many additional structures one may add to this basic definition. We shall introduce below such notions as *symmetric* ,*closed*, $*$*-autonomous*, and *traced* structure, which are key to modelling proofs in linear logic.

Suppose first that there is a natural isomorphism $s_{AB} : A \otimes B \to B \otimes A$ (called a *braiding*) making the following three diagrams commutative.

$$\begin{tikzcd}
\text{(1)} & A \otimes B \arrow[r, "s_{A,B}"] \arrow[dr, "id"'] & B \otimes A \arrow[d, "s_{B,A}"] & \text{(2)} & B \otimes I \arrow[r, "s_{B,I}"] \arrow[dr, "\wr"'] & I \otimes B \arrow[d, "\wr"] \\
& & A \otimes B & & & B
\end{tikzcd}$$

$$\begin{tikzcd}
\text{(3)} & A \otimes (B \otimes C) \arrow[r, "\alpha^{-1}"] \arrow[d, "id_A \otimes s"'] & (A \otimes B) \otimes C \arrow[r, "s"] & C \otimes (A \otimes B) \arrow[d, "\alpha^{-1}"] \\
& A \otimes (C \otimes B) \arrow[r, "\alpha^{-1}"] & (A \otimes C) \otimes B \arrow[r, "s \otimes id_B"] & (C \otimes A) \otimes B
\end{tikzcd}$$

where in (3) we have omitted subscripts for typographical reasons. We say $\mathcal{C}$ is *symmetric* if diagrams (1), (2), and (3) commute. Notice in a symmetric monoidal category, $s_{AB} = (s_{BA})^{-1}$.

More generally, a *braided* monoidal category is a monoidal category satisfying the commutativity of diagrams (2), (3), and (3'), where (3') is like (3) but replacing α^{-1} by α, $id \otimes s$ by $s \otimes id$ and appropriately relabelling the nodes. Such categories arise in knot theory and physics [JS91,JS93,KRT97] as well as in recent semantical studies in Quantum Computing [AbCo04,Abr05].

Let us give some examples that will be useful later.

Examples 2.4 (Symmetric Monoidal Categories)

1. Any cartesian category (=finite products), with $\otimes = \times$.
2. Any co-cartesian category (= finite coproducts), with $\otimes = +$
3. $\mathbf{Rel}_\times$. This is the category **Rel** whose objects are sets and whose arrows are binary relations. Recall the composition of two arrows is their relational product: given $A \xrightarrow{R} B \xrightarrow{S} C$, define $A \xrightarrow{S \circ R} C$ to be the relation defined by

$$a(S \circ R)c \text{ iff } \exists_{b \in B}\, aRb \wedge bSc \quad .$$

The identity morphism $A \xrightarrow{id_A} A$ is simply the diagonal relation $\Delta_A = \{(a,a) \mid a \in A\}$. The functor $\otimes : \mathbf{Rel} \times \mathbf{Rel} \to \mathbf{Rel}$ is defined as follows. On objects, $\otimes = \times$, the cartesian product of sets; on arrows, $A \otimes B \xrightarrow{R \otimes S} C \otimes D$ is the relation given by: $(a,b) R \otimes S (c,d)$ iff aRc & bSd. The tensor unit $I = \{*\}$, any one element set.
4. $\mathbf{Rel}_+$. This is again the category **Rel**, except $\otimes = +$ (disjoint union), where disjoint union in **Set** is given by: $X + Y = X \times \{1\} \cup Y \times \{2\}$. On arrows, $A \otimes B \xrightarrow{R \otimes S} C \otimes D$ is the relation given by:

   $(x,i) R \otimes S (y,j)$ iff $[(i = j = 1$ and $xRy)$ or $(i = j = 2$ and $xSy)]$
   Here the tensor unit $I = \emptyset$.
5. Two important monoidal subcategories of $\mathbf{Rel}_+$ are:
   (1) **Pfn**: Sets and partial functions. Here the morphisms between sets are relations which are *functional*, i.e. binary relations $A \xrightarrow{R} B$ satisfying: $\forall_{x \in A} \forall_{y,y' \in B}\, [xRy \wedge xRy' \to y = y']$.
   (2) **PInj**: Sets and partial injective functions. This is the subcategory of **Pfn** consisting of those partial functions which are also injective on their domains: $\forall_{x,x' \in A} \forall_{y \in B}\, [xRy \wedge x'Ry \to x = x']$.
6. $\mathbf{Vec}_{fd}$ and **Vec** : (finite dimensional) vector spaces over $\mathbf{k}$, where $\mathbf{k}$ is a field. Here $V \otimes W$ is taken to be the usual tensor product, and $I = \mathbf{k}$.
7. The categories $\mathbf{Hilb}_\oplus$, $(\mathbf{Hilb}_\oplus)_{fd}$ of Hilbert spaces (resp. finite dimensional Hilbert spaces) and bounded linear maps with the direct sum $\oplus$ as tensor. Similarly, we may consider the categories **Ban** and $\mathbf{Ban}_{fd}$ of Banach spaces (resp. finite dimensional Banach spaces) and bounded linear maps. Important subcategories of the above include **cBan** and **cHilb**, where the maps are (nonexpansive) contractions, i.e. linear maps L satisfying $\|L(x)\| \leq \|x\|$.
8. $\mathbf{Hilb}_\otimes$ is the category of Hilbert spaces and bounded linear maps, with the tensor being the usual Hilbert space tensor product. There are also a variety of tensor products on Banach spaces, but we shall not require that theory.

In order to deal with internal function spaces, we introduce the notion of *closedness*, as an adjoint functor to $\otimes$:

Definition 2.5 A *symmetric monoidal closed category* (smcc) $\mathcal{C}$ is a symmetric monoidal category such that for all $A \in \mathcal{C}$, the functor $- \otimes A : \mathcal{C} \to \mathcal{C}$ has a right adjoint $A \multimap -$, i.e. there is an isomorphism, natural in B, C, satisfying

$$\mathcal{C}(C \otimes A, B) \cong \mathcal{C}(C, A \multimap B) \tag{1}$$

We say $A \multimap B$ is the "linear exponential" or "linear function space". In particular, the isomorphism (1) induces *evaluation* and *coevaluation* maps $(A \multimap B) \otimes A \to B$ and $C \to (A \multimap (C \otimes A))$, satisfying the adjoint equations.

Examples 2.6

1. Any ccc, with $A \otimes B = A \times B$ and $A \multimap B = A \Rightarrow B$.
2. A poset $\mathcal{P} = (P, \leq)$ is an smcc iff there are operations $\otimes, \multimap$: $P^2 \to P$, $1 \in P$ satisfying:
   (1) $(P, \otimes, 1)$ is a commutative monoid.
   (2) $\otimes, \multimap$ are functorial in the posetal sense: i.e. $x \leq x', y \leq y'$ implies $x \otimes y \leq x' \otimes y'$ and $x' \multimap y \leq x \multimap y'$
   (3) (Closedness) $x \otimes y \leq z$ iff $x \leq y \multimap z$.
3. *Girard's Phase Semantics*: This is a posetal smcc, in the sense of Example 2 above. Let $M = (M, \cdot, e)$ be a commutative monoid. Consider the poset $\mathcal{P}(M)$, the powerset of M. We view $\mathcal{P}(M)$ as a poset ordered by inclusion. For $X, Y \in \mathcal{P}(M)$, define

$$\begin{aligned} X \otimes Y &= XY =_{def} \{x \cdot y \mid x \in X, y \in Y\} \\ X \multimap Y &= \{z \in M \mid z \cdot X \subseteq Y\} \quad \text{and} \quad I = \{e\} \end{aligned}$$

4. **Vec**, where $V \otimes W$ is the usual algebraic tensor product and $V \multimap W = Lin(V, W)$. More generally, consider $\mathcal{R}$-Modules over a commutative ring $\mathcal{R}$, with the standard algebraic notions of $V \otimes_{\mathcal{R}} W$ and $V \multimap W = Hom(V, W)$.
5. $\mathbf{MOD}(G)$. This example extends groups acting on sets to groups acting linearly on vector spaces. Let G be a group and V a vector space. A *representation of* G *on* V is a group homomorphism $\rho : G \to Aut(V)$; equivalently, it is a left G-action $G \times V \xrightarrow{\cdot} V$ (satisfying the same equations as a G-set) such that $v \mapsto g \cdot v$ is a linear automorphism, for each $g \in G$. The pair (ρ, V) is called a *G-module* or *G-space*. $\mathbf{MOD}(G)$ has as objects the G-modules and as morphisms the linear maps commuting with the G-actions. Define the smcc structure of $\mathbf{MOD}(G)$ as follows:

$$\begin{aligned} V \otimes W &= \text{the usual tensor product, with action determined by} \\ & \quad g{\cdot}(v \otimes w) = g{\cdot}v \otimes g{\cdot}w \\ V \multimap W &= Lin(V,W),\ \text{with action } (g{\cdot}f)(v) = g{\cdot}f(g^{-1}{\cdot}v)\ , \\ & \quad \textit{the contragredient action.} \end{aligned}$$

2.3 Monoidal Categories with Duality

For the purposes of studying linear logic, as well as general duality theories, we need to consider monoidal categories equipped with a notion of involutive negation (or "duals"). A general categorical theory of such dualities, including many traditional mathematical duality theories, was developed by M. Barr [Barr79] in the mid 1970's, some ten years before linear logic.

Definition 2.7 ([Barr79]) A $*$-*autonomous category* $(\mathcal{C}, \otimes, I, \multimap, \bot)$ is an smcc with a distinguished *dualizing* object $\bot$, such that (letting $A^* = A \multimap \bot$), the canonical map $\mu_A : A \to A^{**}$ is an iso, for all A (i.e. "all objects are reflexive").

Facts about $*$-autonomous categories $\mathcal{C}$:

- The operation $(-)^*$ induces a contravariant dualizing functor $\mathcal{C}^{op} \xrightarrow{(\ )^*} \mathcal{C}$ such that $\mathcal{C}(A,B) \cong \mathcal{C}(B^*, A^*)$ which is a natural iso and which satisfies all natural coherence equations.
- $\mathcal{C}$ is closed under duality of categorical constructions: e.g. $\mathcal{C}$ has products iff it has coproducts, $\mathcal{C}$ is complete iff it is co-complete, etc.
- $(A \multimap B)^* \cong A \otimes B^*$ and $I \cong \bot^*$ Also $A \multimap B \cong B^* \multimap A^*$.
- We may define $A \parr B = (A^* \otimes B^*)^*$, a kind of "de Morgan dual" of $\otimes$. In linear logic, this is the connective "par", a kind of "parallel disjunction". In general, $\otimes \neq \parr$, and (in general) there is not even a $\mathcal{C}$-morphism $A \otimes B \to A \parr B$.
- As we shall see below, categorical models of multiplicative, additive linear logic will be $*$-autonomous categories with products (hence coproducts).

The first two examples are from Example 2.4 above.

Example 2.8 $\mathbf{Rel}_\times$. The category of relations $\mathbf{Rel}_\times$ is probably the simplest $*$-autonomous category. For sets A, B, $A \otimes B = A \multimap B = A \times B$. Let the dualizing object $\bot = \{*\}$, any one-element set. As for the dualizing functor $(-)^*$, on objects define $A^* = A$. On arrows, given a relation $R : A \to B$, we define $R^* = R^{op} : B \to A$ to be the *opposite* relation (so that bR^*a iff aRb, for any $a \in A, b \in B$). Notice: $(A \otimes B)^* = A^* \otimes B^* = A \times B$.

Example 2.9 **Vec** is symmetric monoidal closed, where $I = \mathbf{k}$ (the base field), $V \otimes W$ is the usual algebraic tensor product and $V \multimap W = Lin(V, W)$. Note that in the category **Vec** of vector spaces over the field **k**, V satisfies $V \cong V^{**}$(via the canonical map μ_V) iff V is finite dimensional (see [Ger85], p. 68). Hence, $\mathbf{Vec}_{fd}$ is $*$-autonomous, where $I = \mathbf{k}$ (the base field) and $V^* = V \multimap I$ is the usual dual space.

Unfortunately, the above two examples $\mathbf{Rel}_\times$ and $\mathbf{Vec}_{fd}$ are "degenerate" $*$-autonomous categories (from the viewpoint of linear logic), since $\parr \cong \otimes$. That is, $(A \otimes B)^* \cong A^* \otimes B^*$. We shall mention these below, as examples of *compact* categories. Indeed, from the viewpoint of linear logic, it is quite hard to find nice examples of nondegenerate $*$-autonomous categories. One of the motivations that led to [Barr79] was that such categories arise quite naturally in various topological duality theories. The following discussion is a quick summary, primarily based on work of M. Barr (e.g. [Barr79]) and the treatment in Blute [Bl96], based on a topology originally due to Lefschetz ([Lef]). See also [BS96]. Let **TVec** denote the category 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, topologized with the topology of pointwise convergence. (It is shown in [Barr79] that the forgetful functor **TVec**→**Vec** is tensor-preserving) . Let V^* denote $V \multimap \mathbf{k}$. Lefschetz proved that the canonical embedding $V \rightarrow V^{**}$ is always a bijection, but need not be an isomorphism. Now we just cut down to so-called *reflexive* spaces: those for which the embedding $V \rightarrow V^{**}$ is actually an isomorphism:

Theorem 2.10 (Barr) **RTVec**, *the full subcategory of reflexive objects in* **TVec**, *is a complete, cocomplete $*$-autonomous category, with $I^* = I = \mathbf{k}$ the dualizing object. Moreover, in* **RTVec**, $\otimes$ *and* $\parr$ *are not isomorphic.*

More generally, other classes of $*$-autonomous categories arise by taking a continuous linear analog of G-sets, namely categories of group representations, using the category **RTVec**.

Definition 2.11 Let G be a group. A *continuous G-module* is a linear action of G on a space V in **TVec**, such that for all $g \in G$, the induced map $g\cdot(\ ) : V \rightarrow V$ is continuous. Let $\mathbf{TMOD}(G)$ denote the category of continuous G-modules and continuous equivariant maps. Let $\mathbf{RTMOD}(G)$ denote the full subcategory of reflexive objects.

Theorem 2.12 *The category* $\mathbf{TMOD}(G)$ *is symmetric monoidal closed. The category* $\mathbf{RTMOD}(G)$ *is $*$-autonomous, and a reflective subcategory of* $\mathbf{TMOD}(G)$ *via the functor* $(\quad)^{**}$. *Furthermore the forgetful functor* $|\ | : \mathbf{RTMOD}(G) \rightarrow \mathbf{RTVec}$ *preserves the $*$-autonomous structure.*

Following [Bl96], still more general classes of $*$-autonomous categories arise analogously using the category $\mathbf{RTMOD}(\mathsf{H})$, the reflective subcategory of linearly topologized H -modules, for a cocommutative Hopf algebra H .

The next notion is much more familiar mathematically, although logically it corresponds to a rather degenerate case of linear logic: the case where $\otimes = \parr$:

Definition 2.13 A *compact closed category* [KL80] is a symmetric monoidal category such that for each object A there exists a dual object A^*, and canonical morphisms:

$$\begin{aligned}\nu &\colon I \to A \otimes A^* \\ \psi &\colon A^* \otimes A \to I\end{aligned}$$

such that evident equations hold. In the case of a strict monoidal category, these equations reduce to the usual adjunction triangles.

Remark 2.14 (From Compactness to *-Autonomy) For constructing new models of multiplicative linear logic, there is a general categorical construction *Double Glueing* which can be used to turn compact closed categories into nontrivial *-autonomous ones, essentially by breaking the isomorphism between $\otimes$ and $\parr$. This is described in detail in [HylSc03]. Indeed, double gluing can be iterated, to obtain interesting categories, e.g. see [HamSc07].

We have already remarked above that the categories $\mathbf{Rel}_\times$ and $\mathbf{Vec}_{fd}$ are compact closed. In the study of quantum computing, Abramsky and Coecke [AbCo04] have shown the utility of *strongly compact closed* categories, those with additional structure abstracting the theory of inner product spaces.

Lemma 2.15

- *Compact closed categories are $*$-autonomous, with the tensor unit as dualizing object.*
- *Recall $A \parr B = (A^* \otimes B^*)^*$. In any $*$-autonomous category in which the tensor unit is the dualizing object, there is a canonical morphism $A \otimes B \to A \parr B$ given by: $\mu_A \otimes \mu_B : A \otimes B \to A^{**} \otimes B^{**} \cong (A^* \otimes B^*)^*$. In a compact closed category, this morphism is an isomorphism.*

What are monoidal functors between monoidal categories? Here there can be several notions. Let us pick an important one:

Definition 2.16 A *monoidal functor* between monoidal categories is a 3-tuple (F, m_I, m) where $F : \mathcal{C} \to \mathcal{D}$ is a functor, together with a morphism $m_I : I \longrightarrow F(I)$ and a natural transformation $m_{UV} : F(U) \otimes F(V) \longrightarrow F(U \otimes V)$ satisfying some coherence diagrams (which we omit). F is *strict* if m_I, m_{UV} are identities. A monoidal functor is *symmetric* if m commutes with the symmetries: $m_{B,A}s_{FA,FB} = F(s_{A,B})m_{A,B}$, for all A, B.

Finally, we need an appropriate notion of natural transformation for monoidal functors.

Definition 2.17 A natural transformation between monoidal functors $\alpha : F \to G$ is *monoidal* if it is compatible with both m_I and m_{UV}, for all U, V, in the sense that the following equations hold: (i) $\alpha_I \circ m_I = m_I$ and
(ii) $m_{UV} \circ (\alpha_U \otimes \alpha_V) = \alpha_{U \otimes V} \circ m_{UV}$.

Remark 2.18 (Alternative Treatments of $*$-autonomy) There are alternative definitions of $*$-autonomous categories, some based on attempts to axiomatize a fully faithful dualizing functor $(-)^* : \mathcal{C}^{op} \to \mathcal{C}$. This leads to thorny problems concerning what are the appropriate categorical coherence equations to impose. Recent work of Robin Houston [Hou07] has shown that this is subtle and is inadequately addressed in the literature, so we omit discussing it.

A more radical alternative categorical treatment of the various layers of linear logic (and thus of $*$-autonomous categories) arose in work of Cockett and Seely and coworkers [CS97,BCST96,BCS00]. Their idea is to first consider *linearly (or weakly) distributive categories*: monoidal categories with two monoidal structures tensor ($\otimes$) and cotensor ($\parr$), together with various coherence and (weak) distributive laws relating them. This corresponds to a kind of multiplicative linear logic of just conjunction/disjunction, without any negation or duality relating the two tensors. On top of this structure, one can impose an involutive negation $(-)^*$ (which will satisfy a De Morgan duality between tensor and cotensor); in addition, one may adjoin products $\times$ (and thus, coproducts $+$) for the additive structure. Finally, one may impose the further exponential structure of linear logic (see below). The authors in the above papers also take extra care in handling the logical units (for tensor and cotensor) and the various categorical coherence problems these require.

3 Linear Logic and Categorical Proof Theory

3.1 Gentzen's proof theory

Gentzen's approach to Hilbert's proof theory [GLT,Mel07], especially his *sequent calculi* and his fundamental theorem on Cut-Elimination, have had a profound influence not only in logic, but recently in category theory and computer science as well. The connections of Gentzen proof theory with categorical logic (and linear logic) are discussed in various survey papers

[Sc00,BS04,Mel07]. Let us just introduce some basic terminology.

A *sequent* for a logical language $\mathcal{L}$ is an expression

$$A_1, A_2, \cdots, A_m \vdash B_1, B_2, \cdots, B_n \tag{2}$$

where $A_1, A_2, \cdots, A_m$ and $B_1, B_2, \cdots, B_n$ are finite lists (possibly empty) of formulas of $\mathcal{L}$. Sequents are denoted $\Gamma \vdash \Delta$, where Γ and Δ are lists of formulas. We think of sequent (2) as a formal *entailment* relationship between the premisses Γ and (potential) conclusions Δ.

Traditional logicians would give the semantical meaning (of the truth) of the sequent (2) as: *the conjunction of the A_i entails the disjunction of the B_j*. More generally, following Lambek and Lawvere, category theorists interpret *proofs* of such sequents (modulo equivalence of proofs) as *arrows* in appropriate (freely generated) monoidal categories. For logics $\mathcal{L}$ similar to Girard's linear logic [Gi87], we interpret a proof π of sequent (2) in a *-autonomous category $(\mathcal{C}, \otimes, I, \multimap, \bot)$ with a "cotensor" $\parr$ (see Definition 2.7 above) as an arrow of the following form

$$[\![A_1]\!] \otimes [\![A_2]\!] \otimes \cdots \otimes [\![A_m]\!] \xrightarrow{[\![\pi]\!]} [\![B_1]\!] \parr [\![B_2]\!] \parr \cdots \parr [\![B_n]\!] \tag{3}$$

Here $[\![-]\!] : \mathcal{L} \to \mathcal{C}$ is an interpretation function of formulas and proofs (of the logic $\mathcal{L}$) into the objects and arrows of $\mathcal{C}$. We interpret formulas A_i as objects $[\![A_i]\!] \in \mathcal{C}$ by induction, starting with an arbitrary interpretation of the atoms (as objects of $\mathcal{C}$).

Remark 3.1 (Notation) We abuse notation for $[\![\pi]\!]$ above and omit writing $[\![-]\!]$ on formulas when it is clear; thus we write the arrow (3) above as

$$A_1 \otimes A_2 \otimes \cdots \otimes A_m \xrightarrow{[\![\pi]\!]} B_1 \parr B_2 \parr \cdots \parr B_n \tag{4}$$

as an interpretation of sequent (2) above in category $\mathcal{C}$.

Gentzen's approach to proof theory gives rules for generating *formal proofs* of sequents. These formal proofs are trees generated by certain rules (called *rules of inference*) for building new sequents from old sequents, starting from initially specified given sequents called *axioms*. Thus, a (formal) proof of $\Gamma \vdash \Delta$ is a tree with root labelled by $\Gamma \vdash \Delta$ and in which every node is labelled by a rule of inference and in which the leaves are labelled by instances of axioms.

Lambek [L89] pointed out that Gentzen's sequent calculus was analogous to Bourbaki's method of bilinear maps. For example, given lists $\Gamma = A_1 \cdots A_m$ and $\Delta = B_1 B_2 \cdots B_n$ of $R - R$ bimodules of a given ring R, there is a natural isomorphism

$$Mult(\Gamma A B \Delta, C) \cong Mult(\Gamma A \otimes B \Delta, C) \tag{5}$$

between $m+n+2$-linear and $m+n+1$-linear maps. Bourbaki derived many aspects of tensor products just from this universal property. Such a formal bijection is at the heart of Linear Logic, whose rules we now present briefly.

3.1.1 Gentzen's Rules

Gentzen's rules analyze the deep structure and implicit symmetries hidden in logical syntax. Gentzen broke down the manipulations of logic into two classes of rules applied to sequents: *structural rules* and *logical rules* (including Axiom and Cut rules.) All rules come in pairs (left/right) applying to the left (resp. right) side of a sequent.

Gentzen's Structural Rules (Left/Right)

$$\begin{array}{llll}
\textit{Permutation} & \dfrac{\Gamma \vdash \Delta}{\sigma(\Gamma) \vdash \Delta} & \dfrac{\Gamma \vdash \Delta}{\Gamma \vdash \tau(\Delta)} & \sigma, \tau \text{ permutations.} \\
\textit{Contraction} & \dfrac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta} & \dfrac{\Gamma \vdash \Delta, B, B}{\Gamma \vdash \Delta, B} & \\
\textit{Weakening} & \dfrac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta} & \dfrac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, B} &
\end{array}$$

Permutation says that from $\Gamma \vdash \Delta$, we can permute arbitrarily the order of the lists of premisses and conclusions. *Contraction* says (for the Left-side) that from an inference of Δ from premisses Γ together with two copies of premise A, we can still infer Δ from Γ but using only one copy of A; dually for contraction on the right. *Weakening* (on the left) says if Γ entails Δ, then adding extra premisses to Γ still entails Δ, and dually for the right hand rule (see [GSS,Abr93,L89]).

In linear logic we do not allow such uncontrolled contraction and weakening; rather, formulas which can be contracted or weakened are marked with $!A$ (for left rules) and $?A$ for right rules. We shall mention more on this below.

By controlling (and making explicit) these traditional structural rules, logic takes on a completely different character.

Definition 3.2 *Formulas* of the theory **LL** (linear logic) are generated from atoms and their negations $p, p^\perp, q, q^\perp, \cdots$, constants $I, \perp, \mathbf{1}, \mathbf{0}$ using the binary connectives $\otimes, \parr, \times, +$ and unary operations $!, ?$. Negation is extended by de Morgan duality to all expressions as follows: $p^{\perp\perp} = p, (A \otimes B)^\perp = A^\perp \parr B^\perp$ and dually, as well as $(A \times B)^\perp = A^\perp + B^\perp$ and $(!A)^\perp = ?(A^\perp)$ and dually. Finally, $A \multimap B$ is defined to be $A^\perp \parr B$ (this connective is redundant, but useful for understanding the categorical semantics of linear logic later below).

Structural	*Perm*	$\dfrac{\Gamma \vdash \Delta}{\sigma(\Gamma) \vdash \tau(\Delta)}$		σ, τ permutations.
Axiom & Cut	*Axiom*	$A \vdash A$		
	Cut	$\dfrac{\Gamma \vdash A, \Delta \quad \Gamma', A \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'}$		
Negation		$\dfrac{\Gamma \vdash A, \Delta}{\Gamma, A^{\perp} \vdash \Delta}$		$\dfrac{\Gamma, A \vdash \Delta}{\Gamma \vdash A^{\perp}, \Delta}$
Multiplicatives	*Tensor*	$\dfrac{\Gamma, A, B \vdash \Delta}{\Gamma, A \otimes B \vdash \Delta}$		$\dfrac{\Gamma \vdash A, \Delta \quad \Gamma' \vdash B, \Delta'}{\Gamma, \Gamma' \vdash A \otimes B, \Delta, \Delta'}$
	Par	$\dfrac{\Gamma, A \vdash \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \parr B \vdash \Delta, \Delta'}$		$\dfrac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \parr B, \Delta}$
	Units	$\dfrac{\Gamma \vdash \Delta}{\Gamma, I \vdash \Delta}$		$\vdash I$
		$\perp \ \vdash$		$\dfrac{\Gamma \vdash \Delta}{\Gamma \vdash \perp, \Delta}$
	Implication	$\dfrac{\Gamma \vdash A, \Delta \quad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \multimap B \vdash \Delta, \Delta'}$		$\dfrac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \multimap B, \Delta}$
Additives	*Product*	$\dfrac{\Gamma, A \vdash \Delta}{\Gamma, A \times B \vdash \Delta} \quad \dfrac{\Gamma, B \vdash \Delta}{\Gamma, A \times B \vdash \Delta}$		$\dfrac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \times B, \Delta}$
	Coproduct	$\dfrac{\Gamma, A \vdash \Delta \quad \Gamma, B \vdash \Delta}{\Gamma, A + B \vdash \Delta}$		$\dfrac{\Gamma \vdash A, \Delta}{\Gamma \vdash A + B, \Delta} \quad \dfrac{\Gamma \vdash B, \Delta}{\Gamma \vdash A + B, \Delta}$
	Units	$\Gamma, \mathbf{0} \ \vdash \Delta$		$\Gamma \vdash \mathbf{1}, \Delta$
Exponentials	*Weakening*	$\dfrac{\Gamma \vdash \Delta}{\Gamma, !A \vdash \Delta}$	*Contraction*	$\dfrac{\Gamma, !A, !A \vdash \Delta}{\Gamma, !A \vdash \Delta}$
	Storage	$\dfrac{!\Gamma \ \vdash A}{!\Gamma \ \vdash !A}$	*Dereliction*	$\dfrac{\Gamma, A \vdash \Delta}{\Gamma, !A \vdash \Delta}$

Fig. 2. Rules for Classical Propositional LL

The Logical Rules of Linear Logic are in Figure 2. Previously equivalent notions now split into subtle variants based on resource allocation. For example, the rules for *Multiplicative* connectives simply concatenate their input hypotheses Γ and Γ', whereas the rules for *Additive* connectives merge two input hypotheses Γ into one. The situation is analogous for conclusions Δ and Δ'.

In Figure 2 we illustrate the rules of linear logic. The Exponential rules are the rules of the connective ! (e.g. contraction and weakening on the left side of sequents). Using the rules of negation, one can obtain the dual laws (e.g. contraction and weakening on the right side) by using the dual ? connective.

The logical connectives in linear logic can represent linguistic distinctions related to resource use which are simply impossible to formulate in traditional logic (see [Gi89,Abr93]). For example, we think of a linear entailment $A_1, \cdots, A_m \vdash B$ as an *action*–a kind of process that in a single step consumes the inputs A_i and produces output B. For example, this permits representing in a natural manner the step-by-step behaviour of various abstract machines, certain models of concurrency like Petri Nets, etc. Thus, linear logic permits us to describe the instantaneous state of a system, and its step-wise evolution, intrinsically within the logic itself (e.g. with no need for explicit time parameters, etc.)

We should note that linear logic is not about simply removing Gentzen's structural rules, but rather modulating their use. The particular connective $!A$, which indicates that contraction and weakening may be applied to formula A, yields the *Exponential* connectives in Figure 2. From a resource viewpoint, an hypothesis $!A$ is one which can be reused arbitrarily. It is roughly like an infinite tensor power $\otimes^{\omega} A$, and more generally (for physicists) something like an exterior algebra or Fock-space like construction.

Moreover, this connective permits decomposing intuitionistic implication "$\Rightarrow$" (categorically, the cartesian closed function space) into more basic notions:

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

Remark 3.3 (1-sided Sequents & Theories) Observe that in classical linear logic **LL**, two-sided sequents can be replaced by one-sided sequents, since $\Gamma \vdash \Delta$ is equivalent to $\vdash \Gamma^{\perp}, \Delta$, with $\Gamma^{\perp}$ the list $A_1^{\perp}, \cdots, A_n^{\perp}$, where Γ is $A_1, \cdots, A_n$. This permits halving the number of rules, and we shall use this notation frequently, see Figure 3. Finally, we end with the following standard terminology of subtheories of **LL** in Figure 2. *The literature usually presents the theories below using 1-sided sequents.*

MLL: *multiplicative linear logic* is built from the atoms and multiplicative units $\{I, \perp\}$ using the connectives $\{\otimes, \parr, (\ )^{\perp}\}$. The rules include the structural, axioms, cut, negation and the multiplicative rules. This theory corresponds semantically to $*$-autonomous categories.

MALL: *multiplicative additive linear logic* is built from the atoms and the units $\{I, \perp, \mathbf{0}, \mathbf{1}\}$ using the connectives $\{\otimes, \parr, (\ )^{\perp}, \times, +\}$. The rules include the **MLL** rules together with the additive rules. This theory corresponds semantically to $*$-autonomous categories with products (hence coproducts).

Structural	*Perm*	$\dfrac{\vdash\ \Gamma}{\vdash\ \tau(\Gamma)}$	τ a permutation.	
Axiom & Cut	*Axiom*	$\vdash\ A^{\perp}, A$		
	Cut	$\dfrac{\vdash\ A, \Gamma \quad \vdash\ A^{\perp}, \Gamma'}{\vdash\ \Gamma, \Gamma'}$		
Multiplicatives	*Tensor*	$\dfrac{\vdash\ A, \Gamma \quad \vdash\ B, \Gamma'}{\vdash\ A \otimes B, \Gamma, \Gamma'}$		
	Par	$\dfrac{\vdash\ A, B, \Gamma}{\vdash\ A \parr B, \Gamma}$		
	Units	$\vdash\ I$	$\dfrac{\vdash\ \Gamma}{\vdash \bot, \Gamma}$	
Additives	*Product*	$\dfrac{\vdash\ A, \Gamma \quad \vdash\ B, \Gamma}{\vdash\ A \times B, \Gamma}$		
	Coproduct	$\dfrac{\vdash\ A, \Gamma}{\vdash\ A + B, \Gamma} \quad \dfrac{\vdash\ B, \Gamma}{\vdash\ A + B, \Gamma}$		
	Units	$\vdash\ \mathbf{1}, \Gamma$		
Exponentials	*Weakening*	$\dfrac{\vdash\ \Gamma}{\vdash ?A, \Gamma}$	*Contraction*	$\dfrac{\vdash ?A, ?A, \Gamma}{\vdash ?A, \Gamma}$
	Storage	$\dfrac{\vdash ?\Gamma\ , A}{\vdash ?\Gamma\ ,\ !A}$	*Dereliction*	$\dfrac{\vdash\ A, \Gamma}{\vdash ?A, \Gamma}$

Fig. 3. 1-Sided Rules for Classical Propositional **LL**

MELL: *multiplicative exponential linear logic* is built from those formulas of **LL** that do not use any of the additive structure: that is, formulas built from the atoms and multiplicative units $\{I, \bot\}$ using the connectives $\{\otimes, \parr, (\ )^{\perp}, !, ?\}$. The rules include the structural, axioms, cut, negation and the multiplicative and exponential rules.

3.1.2 Categorical Proof Theory

One of the basic ideas of categorical logic and categorical proof theory is that (the proof theory of) various logics generate interesting classes of free categories: free cartesian, cartesian closed, monoidal, monoidal closed, *-

autonomous, toposes, etc. The intuition is:

- Formulas of a logic should be the objects of a category.
- Proofs (or, rather, equivalence classes of proofs) should be the morphisms.

The subject began in the work of Lawvere and of Lambek in the 1960's and is discussed in detail in [LS86] (cf. also the expository treatment in [BS04]). One of the early applications of Lambek was to apply these methods to solve coherence problems for various monoidal categories.

3.2 Categorical Models of Linear Logic

We are interested in finding the categories appropriate to modelling linear logic proofs (just as cartesian closed categories modelled intuitionistic $\wedge, \Rightarrow, \top$ proofs). The basic equations we certainly *must* postulate arise from the operational semantics–that is *cut-elimination of proofs*. If we have a proof-rewriting $A_1, \cdots, A_m \overset{\overset{\pi}{\vdots}}{\vdash} B_1, \cdots, B_n \rightsquigarrow A_1, \cdots, A_m \overset{\overset{\pi'}{\vdots}}{\vdash} B_1, \cdots, B_n$ then the categorical interpretation $\llbracket - \rrbracket$ of these proofs (as arrows in an appropriate category as in (4) above) should be to give *equal* arrows $\llbracket \pi \rrbracket = \llbracket \pi' \rrbracket :$ $A_1 \otimes \cdots \otimes A_m \to B_1 \parr \cdots \parr B_n$.

In the case of sequent calculi, this rewriting is generated by the rules of Gentzen's Cut-Elimination algorithm [GLT]. However, there are sometimes natural categorical equations (e.g. the universal property of cartesian products) which are not decided by traditional proof theoretic rewriting, and need to be postulated separately (otherwise, conjunction only gives a "weak product" [LS86]). Precisely which equations to add, to make a mathematically natural and beautiful structure, is an important question. The problem is further compounded in linear logic (at the level of the exponentials) where the equations and coherences are more subtle, with more variations possible.

The first attempted categorical semantics of **LL** is in Seely's paper [See89] which is still a good resource (although some fine details have turned out to require modification). Since that time, considerable effort by many researchers has led to major clarifications and quite different axiomatizations. An excellent survey of the current state-of-the-art is in Melliès [Mel07]. In the case of Multiplicative-Additive classical linear logic **MALL**, there is little controversy: the syntax should generate a free $*$-autonomous category with products (and thus coproducts). In more detail, in Figure 4 we present the categorical structure of (free) $*$-autonomous categories considered as symmetric monoidal closed categories (smcc's) with dualizing objects $\perp$, as in the discussion above.

We may think of the arrows $A \xrightarrow{f} B$ as proofs of very simple sequents $A \vdash B$ (where premisses and conclusions are lists of length 1). For example, the identity map $A \xrightarrow{id} A$ corresponds to the axiom $A \vdash A$. The remaining laws of linear logic follow from the arrow-generating rules. The associated equations guarantee that: (i) we get all the axioms of $*$-autonomous categories with products, but also (ii) these are the equations between proofs we must postulate to get a nice categorical structure (this is relevant to our next section on Cut-Elimination).

At this point we could also add coproducts, denoted $+$, and their associated equations, dual to products. But once we have the equations of $*$-autonomous categories (at the bottom of Figure 4) we get coproducts for free, essentially by De Morgan duality. Finally we add any necessary coherence equations, as in Barr's monograph [Barr79].

Arrow-generating Rules	Equations
$A \xrightarrow{id} A \qquad \dfrac{A \xrightarrow{f} B \quad B \xrightarrow{g} C}{A \xrightarrow{gf} C}$	equations of a category
$\dfrac{A \xrightarrow{f} B \quad A' \xrightarrow{g} B'}{A \otimes A' \xrightarrow{f \otimes g} B \otimes B'}$	$\otimes$ is a functor : $ff' \otimes gg' = (f \otimes g)(f' \otimes g')$ $id \otimes id = id$
$(A \otimes B) \otimes C \xrightarrow{\alpha} A \otimes (B \otimes C)$ $A \otimes B \xrightarrow{s} B \otimes A$ $I \otimes A \xrightarrow{\ell} A$	α, s, ℓ are natural isos equations for symmetric monoidal structure
$\dfrac{A \otimes B \xrightarrow{f} C}{A \xrightarrow{f^*} (B \multimap C)} \multimap \mathrm{R}$	equations for monoidal closedness
$(A \multimap B) \otimes A \xrightarrow{ev} B$	(this gives smcc's)
$\dfrac{C \xrightarrow{f} A \quad C \xrightarrow{g} B}{C \xrightarrow{\langle f,g \rangle} A \times B}$	cartesian products
$A \times B \xrightarrow{\pi_1} A \quad A \times B \xrightarrow{\pi_2} B$	(this gives smcc's + products)
$A \xrightarrow{!_A} \top$	
$\dfrac{A \xrightarrow{f} B}{B^* \xrightarrow{f^*} A^*}$	$(-)^*$ is a contravariant functor
$A^* \longrightarrow (A \multimap \bot)$ $(A \multimap \bot) \longrightarrow A^*$	these are natural isos
$(A \multimap B) \rightarrow (B^* \multimap A^*)$	natural strength iso
$A \rightarrow ((A \multimap \bot) \multimap \bot)$	natural iso

Fig. 4. *-Autonomous Categories Equationally

By far the most subtle question is how to model the linear modality !. We begin with seven basic derivation forms, arising from the rules of linear logic and then postulate equations which arise directly from the categorical viewpoint.

Functoriality	$\dfrac{A \xrightarrow{f} B}{!A \xrightarrow{!f} !B}$	
Monoidalness	$I \xrightarrow{m_I} !I$	$!A \otimes !B \xrightarrow{m_{AB}} !(A \otimes B)$
Products	$I \xrightarrow{n_I} !\top$	$!A \otimes !B \xrightarrow{n_{AB}} !(A \times B)$
Dereliction	$!A \xrightarrow{\varepsilon_A} A$	
Weakening	$!A \xrightarrow{\varepsilon'_A} I$	
Contraction	$!A \xrightarrow{\delta'_A} !A \otimes !A$	
Digging (Storage)	$!A \xrightarrow{\delta_A} !!A$	

Fig. 5. Basic Exponential Laws

Exercise 3.4 Prove the laws in Figure 5, using the 2-sided rules in Figure 2. Let us give two examples. As mentioned above, we think of $A \xrightarrow{f} B$ as a proof f of the sequent $A \vdash B$.

- Functoriality: $\dfrac{\dfrac{\begin{array}{c} f \\ \vdots \\ A \vdash B \end{array}}{!A \vdash B}\,\text{Derel.}}{!A \vdash !B}\,\text{Storage}$, where f is the given proof of $A \vdash B$.
- Contraction: Applying functoriality to the axiom $A \vdash A$, we get a proof $\pi : !A \vdash !A$. Now use π twice in the following proof tree:

$$\dfrac{\dfrac{\begin{array}{c} \pi \\ \vdots \\ !A \vdash !A \end{array} \quad \begin{array}{c} \pi \\ \vdots \\ !A \vdash !A \end{array}}{!A, !A \vdash !A \otimes !A}\,\otimes\text{R}}{!A \vdash !A \otimes !A}\,\text{Contr}$$

So, what is a model of full linear logic? The state-of-the-art is described in work of Hyland-Schalk [HylSc03] and especially Melliès [Mel07]. Here is one class of structure that is popular to impose: let $\mathcal{C}$ be a model of **MALL** proofs, i.e. a $*$-autonomous category with products (and hence coproducts). We add:

- $(!, m_I, m_{AB}) : \mathcal{C} \to \mathcal{C}$ is a monoidal endofunctor
- $!A \xrightarrow{\varepsilon_A} A$ and $!A \xrightarrow{\delta_A} !!A$ are monoidal natural transformations.
- $(!, \delta, \varepsilon)$ is a monoidal comonad.
- n_I, n_{AB} are isomorphisms, natural in A, B.
- The associated adjunction structure $\langle F, U, \eta, \varepsilon\rangle$ between the co-Kleisli category of ! and $\mathcal{C}$ is monoidal.
- Various coherence equations [BCS96,Mel07].

However, for the purposes of Geometry of Interaction, we shall not need all this elaborate structure of the exponentials and the associated properties of cocommutative comonoids, etc. Indeed, beyond the basic derivations in Figure 5, one merely needs the exponential structure associated to a *Linear Combinatory Algebra* [AHS02], as we shall see.

3.4 Cut Elimination: Gentzen's Operational Semantics of Proofs

Let us briefly discuss the Cut-Elimination theorem in proof theory. For more details, the reader may examine the works of Girard (e.g. [GLT,Gi87]) or the survey of Melliès [Mel07] or the textbook [TrSchw]. Recall the Cut-Rule, which is a kind of generalized composition law:

$$\frac{\Gamma \vdash \Delta, A \quad \Gamma', A \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'}\,\mathrm{Cut}$$

A fundamental theorem of logic is the following result of Gentzen:

Cut-elimination (Gentzen's Haupsatz, 1934): If π is a proof of $\Gamma \vdash \Delta$, then there is a proof π' of $\Gamma \vdash \Delta$ which does not use the cut rule.

It is the basis of Proof Theory, at the very foundations of Hilbert's approach to logic, and has applications in a wide range of areas of both logic and theoretical computer science.

For usual sequent calculus, Gentzen gave a Non-Deterministic algorithm $\pi \rightsquigarrow \pi'$ (the cut-elimination procedure) for transforming proofs π into proofs π'. The details of the rewriting steps (for each proof rule, Left and Right) become rather intricate. Here is an example of a rewriting step, with respect to the Contraction Rule on the Right:

$$\dfrac{\dfrac{\overset{\vdots}{\Gamma \vdash B, B}}{\Gamma \vdash B}\,\mathit{Contr} \qquad \overset{\vdots}{B \vdash \Delta}}{\Gamma \vdash \Delta}\,\text{Cut} \quad \rightsquigarrow \quad \begin{array}{c} \dfrac{\dfrac{\overset{\vdots}{\Gamma \vdash B, B} \quad \overset{\vdots}{B \vdash \Delta}}{\Gamma \vdash B, \Delta}\,\text{Cut} \qquad \overset{\vdots}{B \vdash \Delta}}{\Gamma \vdash \Delta, \Delta}\,\text{Cut} \\ \vdots\ \mathit{Contrs\&Perms} \\ \Gamma \vdash \Delta \end{array}$$

Notice that this is slightly strange: starting from the root of the tree and going upwards, the subproof of $B \vdash \Delta$ in the original left proof is now duplicated in the right proof *higher up in the tree.* So moving from the LHS proof to the RHS proof, we have replaced a single Cut (on B) by two cuts on B higher up in the tree (beyond the contraction); at the same time we have postponed the contractions until later, lower down in the proof. But since the duplicated proof of $B \vdash \Delta$ may be arbitrarily complex (millions of lines long) it is not obvious that the rewriting above has "simplified" anything. The point is that Gentzen, with respect to subtle complexity measures, is able to show that there is a measure which decreases, thus the process terminates. This is explained in more details in [GLT,GSS,TrSchw]. Thus, to every proof we obtain a "cut-free" proof, i.e. its *normal form.* One sometimes calls the process *proof normalization.*

For the systems of linear logic we deal with in this paper, the rewriting/cut-elimination process yields *unique* normal forms, that is the cut-free form of a proof is independent of the order of applying the rewriting steps. This is proved by a Church-Rosser (or Diamond Lemma) type of argument [LS86,GLT]. In GoI, we shall obtain analogs of this property (e.g., see Lemma 6.8).

4 Traced monoidal categories

The theory of traces has had a fundamental impact within diverse areas of mathematics, from functional analysis and noncommutative geometry to topology and knot theory. More recently, abstract traces have arisen in logic and theoretical computer science. For example, in the 80's and 90's it was realized there was a need for algebraic structures modelling cyclic operations. parametrized fixedpoints and feedback in such areas as: flowchart schemes, dataflow, network algebra, and more recently in quantum computing and biological modelling.

Traced monoidal categories were introduced by Joyal, Street, and Verity [JSV96]. These categories and their variants have turned out to be key ingredients in discussing the above phenomena. As quoted by [JSV96],

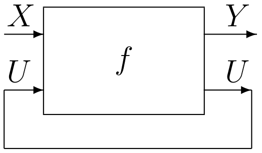


Fig. 6. The trace $\mathtt{Tr}^U_{X,Y}(f)$

> *This paper introduces axioms for an abstract trace on a monoidal category. This trace can be interpreted in various contexts where it could alternatively be called contraction, feedback, Markov trace or braid closure...*

There have been various extensions of traces and partial traces: we discuss more of this in Section 7 as well as in Remark 4.4 below.

Definition 4.1 A *traced symmetric monoidal category* is a symmetric monoidal category $(\mathcal{C}, \otimes, I, s)$ with a family of functions $\mathtt{Tr}^U_{X,Y} : \mathcal{C}(X \otimes U, Y \otimes U) \longrightarrow \mathcal{C}(X, Y)$ pictured in Figure 6, called a *trace*, subject to the following axioms:

(1) **Natural** in X, $\mathtt{Tr}^U_{X,Y}(f)g = \mathtt{Tr}^U_{X',Y}(f(g \otimes 1_U))$, where $f : X \otimes U \longrightarrow Y \otimes U$, $g : X' \longrightarrow X$,

(2) **Natural** in Y, $g\mathtt{Tr}^U_{X,Y}(f) = \mathtt{Tr}^U_{X,Y'}((g \otimes 1_U)f)$, where $f : X \otimes U \longrightarrow Y \otimes U$, $g : Y \longrightarrow Y'$,

(3) **Dinatural** in U, $\mathtt{Tr}^U_{X,Y}((1_Y \otimes g)f) = \mathtt{Tr}^{U'}_{X,Y}(f(1_X \otimes g))$, where $f : X \otimes U \longrightarrow Y \otimes U'$, $g : U' \longrightarrow U$,

(4) **Vanishing (I, II)**, $\mathtt{Tr}^I_{X,Y}(f) = f$ and $\mathtt{Tr}^{U \otimes V}_{X,Y}(g) = \mathtt{Tr}^U_{X,Y}(\mathtt{Tr}^V_{X \otimes U, Y \otimes U}(g))$, for $f : X \otimes I \longrightarrow Y \otimes I$ and $g : X \otimes U \otimes V \longrightarrow Y \otimes U \otimes V$.

(5) **Superposing**,

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

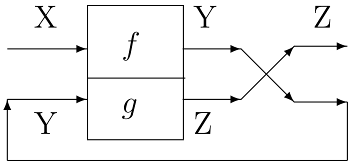
for $f : X \otimes U \longrightarrow Y \otimes U$ and $g : W \longrightarrow Z$.

(6) **Yanking**, $\mathtt{Tr}^U_{U,U}(s_{U,U}) = 1_U$.

Given $f : X \otimes U \to Y \otimes U$, we think of $\mathtt{Tr}^U_{X,Y}(f)$ as "feedback along U", as in Figure 6. Similarly, the axioms of traced monoidal categories have suitable geometrical representation, given in Appendix A (cf. also [JSV96,AHS02,Has08]).

Observe that if $X = Y = I$, up to isomorphism we have $\mathtt{Tr}^U_{X,Y}(f) : \mathcal{C}(U, U) \to \mathcal{C}(I, I)$ is a *scalar-valued* trace (cf. Proposition 2.3).

Exercise 4.2 (Generalized Yanking) Let $\mathcal{C}$ be a traced symmetric monoidal category, with arrows $f : X {\to} Y$ and $g : Y {\to} Z$. Then $g{\circ}f = \mathtt{Tr}^Y_{X,Z}(s_{Y,Z}{\circ}(f{\otimes}g))$. Geometrically, stare at the diagram in Figure 7 , and do a "string-pulling" argument (For an algebraic proof, see Proposition 2.4 in [AHS02])



$$= \quad X \xrightarrow{f} Y \xrightarrow{g} Z$$

Fig. 7. Generalized Yanking

Note that this exercise actually says that composition $g{\circ}f$ in a traced monoidal category is definable from tensor and symmetries. More generally, in [AHS02] we have the following normal-form theorem for arrows in traced symmetric monoidal categories:

Theorem 4.3 *Let $\mathcal{C}$ be a traced symmetric monoidal category, and T a collection of arrows in $\mathcal{C}$. Then any expression E built from arrows in T using tensor product, composition, and trace can be represented as* $\mathtt{Tr}(\pi F \tau)$ *where F consists of tensor products of arrows in T and π, τ are permutations (built from symmetry and identity maps).*

Let us remark that for logicians, the discussion above prefigures the Execution Formula (see Equation (10) and Figure 9(b) below), since it illustrates the reduction of general composition ("cut") to a global trace applied to primitive compositions of permutations and tensoring.

Generalized yanking is also often used in some axiomatizations for partial traces [ABP99,Pl03] although for our purposes it is equivalent to yanking [AHS02].

Remark 4.4 (Some traces literature) In computer science, there has been a long tradition of studying theories related to traces and partial traces in the analysis of feedback, fixed points, iteration theories, and related notions in network algebra and flowcharts. Detailed and fundamental categorical work by Manes and Arbib[MA86], Bloom and Esik [BE93], and Stefanescu [Ste00] have greatly influenced our development here. We should mention very interesting work on circuits and feedback categories in a series of papers by Katis, Sabadini, and Walters (e.g. [KSW02]). They also introduce an interesting notion of partial trace, an important topic we introduce (for purposes of GoI) in Section 7 below. We should also mention work of P. Hines [Hi97,Hi03] both on analyzing GoI and studies of abstract machines. Finally, a survey of recent results on traced monoidal categories is in Hasegawa[Has08].

4.1 Wave vs. Particle style traces

Many examples of traces can be divided into two styles[Abr96,AHS02]: *Product Style* and *Sum Style*, or more evocatively (following Abramsky) "wave style" and “particle style”. These refer, respectively, to whether the monoidal tensor $\otimes$ is given by a cartesian product versus whether it is given by a disjoint union. As explained in [Abr96,AHS02], product-style traces may be thought of as passing information in a “global information wave” while sum-style traces can be modelled by streams of particles or tokens flowing around a network (cf. [AHS02,Hag00,Hi97]). We shall now illustrate both styles of trace.

Examples 4.5 (Product Style Traces)

1. The category $\mathbf{Rel}_\times$ is traced. Let $R : X \times U \longrightarrow Y \times U$ be a morphism in $\mathbf{Rel}_\times$. Then $\mathtt{Tr}^U_{X,Y}(R) : X \longrightarrow Y$ is defined by: $\mathtt{Tr}^U_{X,Y}(R)(x, y) = \exists u.R(x, u, y, u)$.
2. The category $\mathbf{Vec}_{fd}$ is traced. Let $f : V \otimes U \longrightarrow W \otimes U$ be a linear map, where U, V, W are finite dimensional vector spaces with bases $\{u_i\}, \{v_j\}, \{w_k\}$. We define $\mathtt{Tr}^U_{V,W}(f) : V \longrightarrow W$ by:

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

   This reduces to the usual trace of $f : U \longrightarrow U$ when V and W are one dimensional.
3. Note that both $\mathbf{Rel}_\times$ and $\mathbf{Vec}_{fd}$ are compact closed categories. More generally [JSV96], every compact closed category has a *unique* canonical trace given by:

   $$\mathtt{Tr}^U_{A,B}(f) = A \cong A \otimes I \xrightarrow{id\otimes\nu} A \otimes U \otimes U^* \xrightarrow{f\otimes id} B \otimes U \otimes U^* \xrightarrow{id\otimes\psi\circ s} B \otimes I \cong B.$$

   Uniqueness of this trace is shown in [Has08].
4. *Coherent Logic and $\exists$-Doctrines.* A slight generalization of Example (1) is to consider any theory in multisorted coherent logic, that is the fragment $\{\exists, \wedge\}$ of ordinary logic (here it doesn’t matter if one picks intuitionist or classical logic) [KR77]. The objects are Sorts (assumed closed under $\times$), denoted X, Y, Z, etc. Morphisms are (equivalence classes of) formulas, thought-of as relations between sorts: $R(x, y) : X \to Y$, modulo provable equivalence. Composition is defined like relational composition: $X \xrightarrow{R(x,y)} Y \xrightarrow{S(y,z)} Z = X \xrightarrow{T(x,z)} Z$, where

   $$T(x, z) = \exists y.R(x, y) \wedge S(y, z)$$

   This is a well-defined operation, using laws of coherent logic. Omitting pairing symbols, given $R(x, u, y, u') : X \times U \longrightarrow Y \times U$, define

   $$\mathtt{Tr}^U_{X,Y}(x, y) : X \to Y = \exists u R(x, u, y, u)$$

   The same calculations used in Example (1) can be mimicked in Coherent Logic to show that this yields a trace. The close connections of Coherent Logic with Regular Categories provides a large stock of examples of these styles of trace. Indeed, still more generally, the calculations are true for Lawvere's Existential Doctrines with $\exists_\pi$-quantifiers along projections, Frobenius Reciprocity, Beck-Chevalley, and in which equality is definable by $\exists_\Delta$, existential quantification along a diagonal [Law69,Law70].

5. The category ω-$\mathbf{CPO}_\perp$ consists of objects of ω-$\mathbf{CPO}$ with a least element $\perp$, and maps of ω-$\mathbf{CPO}$ that do not necessarily preserve $\perp$. Here $\otimes = \times$, $I = \{\perp\}$. The (dinatural) family of least-fixed-point combinators $\mathsf{Y}_U : U^U \to U$ induces a trace, given as follows (using informal lambda calculus notation): for any $f : X \times U \to Y \times U$, $\mathtt{Tr}^U_{X,Y}(f)(x) = f_1(x, \mathsf{Y}_U(\lambda u.f_2(x,u)))$, where $f_1 = \pi_1 \circ f : X \times U \to Y$, $f_2 = \pi_2 \circ f : X \times U \to U$ and $\mathsf{Y}_U(\lambda u.f_2(x,u))$ = the least element u' of U such that $f_2(x,u') = u'$.
6. (cf. Katis, Sabadini,Walters [KSW02]) Take any (Lawvere) equational theory, for example the theory of rings. Define a category whose objects are of the form R^n for a fixed ring R, where $n \in \mathbb{N}$. Define $Hom(R^n, R^m)$ = m-tuples of polynomials in n indeterminates, with composition being substitution. For example, the identity map $R^n \xrightarrow{id} R^n$ is given by the list of n polynomials $p_1, \cdots, p_n$, where $p_i(x_1, \cdots, x_n) = x_i$. Here $\otimes$ is cartesian product.

   A morphism $\vec{f}, \vec{g} \in Hom(R^n \times R^p, R^m \times R^p)$ is a list of $m + p$ polynomials in $n + p$ unknowns. We can write it as a system of polynomial equations:

$$\begin{aligned} y_1 &= f_1(\vec{x}, \vec{u}) \\ &\vdots \\ y_m &= f_m(\vec{x}, \vec{u}) \\ u'_1 &= g_1(\vec{x}, \vec{u}) \\ &\vdots \\ u'_p &= g_p(\vec{x}, \vec{u}) \end{aligned}$$

The operation of trace or feedback is the formal identification of the variables u'_i on the LHS of the equations with the u_i on the RHS. Of course, to know this setting is consistent (yielding a nontrivial category) we should provide models in which there exist nontrivial solutions of such simultaneous feedback equations. These are discussed, for example, in [KSW02] above. This example admits many generalizations: for example, to general Lawvere theories, in which morphisms are represented by (equivalence classes of) terms with free variables, modulo provable equality in the theory.

Unfortunately, the above examples do not really illustrate the notion of feedback as data flow: the movement of tokens through a network. This latter view, emphasized in work of Abramsky and later Haghverdi and Hines (cf. [Abr96,AHS02,Hag00,Hi97]), is illustrated by examples based on sum-style monoidal structure. They are related to dataflow interpretations of graphical networks. We illustrate this view with categories connected to **Rel**.

Examples 4.6 (Sum-style Traces)

1. **Rel**$_+$, the category **Rel** with $\otimes = +$, disjoint union. Suppose $X+U \xrightarrow{R} Y+U$ is a relation. The coproduct injections induce four restricted relations : $R_{UU}, R_{UY}, R_{XY}, R_{XU}$ (for example, $R_{XY} \subseteq X \times Y$ is such that $R_{XY}(x,y) = R(in_1^{X,U}(x), in_1^{Y,U}(y))$. Let R^* be the reflexive, transitive closure of the relation R. A trace can be defined as follows:
$$\begin{aligned}\mathtt{Tr}^U_{X,Y}(R) &= R_{XY} \cup \bigcup_{n\geq 0} R_{UY} \circ R^n_{UU} \circ R_{XU}\\ &= R_{XY} \cup R_{UY} \circ R^*_{UU} \circ R_{XU}. \end{aligned} \tag{6}$$

2. The categories **Pfn** and **PInj** of sets and partial functions (resp. sets and partial injective functions), as monoidal subcategories of **Rel**$_+$. The tensor product is given by the disjoint union of sets, where we identify $A+B = A\times\{1\} \cup B \times \{2\}$ (note that this is not a coproduct in **PInj**, although it is a coproduct in **Pfn**). There are the obvious injections $in_1^{A,B} : A \to A+B$ and $in_2^{A,B} : B \to A+B$ as well as "quasiprojections" $\rho_1 : A+B \longrightarrow A$ given by $\rho_1((a,1)) = a$ (where $\rho_1((b,2))$ is undefined) and similarly for $\rho_2 : A+B \longrightarrow B$.

   Given a morphism $f : X+U \longrightarrow Y+U$, we may consider its four "components" $f_{XY} : X \to Y$, $f_{XU} : X \to U$, $f_{UX} : U \to X$, and $f_{UU} : U \to U$ obtained by pre- and post-composing with injections and quasiprojections: for example, $f_{XY} = X \xrightarrow{in_1} X+U \xrightarrow{f} Y+U \xrightarrow{\rho_1} Y$, (See Figure 8).

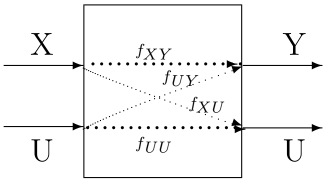


Fig. 8. Components of $f : X+U \to Y+U$

   Both **Pfn** and **Pinj** are traced, the trace being given by the following iterative formula
$$\mathtt{Tr}^U_{X,Y}(f) = f_{XY} + \sum_{n\in\omega} f_{UY} f^n_{UU} f_{XU}, \tag{7}$$

   which we interpret as follows:

For the category **Pfn** (respectively **PInj**), a family $\{h_i\}_{i\in I} : X \longrightarrow Y$ is said to be *summable* if the h_i's have pairwise disjoint domains (respectively, have pairwise disjoint domains and codomains). In either case, we define the sum of the family to be:

$$(\sum_{i\in I} h_i)(x) = \begin{cases} h_j(x), & \text{if } x \in \mathtt{Dom}(h_j) \text{ for some } j \in I; \\ \text{undefined, else.} \end{cases}$$

From a dataflow view, particles enter through X, travel around a loop on U some number n of times, then exit through Y. Numerous other examples of such "particle-style" traces are studied in [AHS02,Hag00]. We shall now introduce a general theory of such traces, based upon Haghverdi's *Unique Decomposition Categories.*

4.2 Unique Decomposition Categories and Particle-Style Traces

How do we make sense of sums such as in equation (7) above? Haghverdi [Hag00,Hag00a] introduced symmetric monoidal categories whose homsets come equipped with (technically, are *enriched in*) an abstract summability structure, called a Σ-monoid. Sigma monoids, and their variants, permit forming certain infinite sums of maps, in a manner compatible with the monoidal category structure. Haghverdi's work is a generalization of the work of Manes and Arbib [MA86] who introduced *partially additive categories* in programming language semantics. These categories form a useful general framework for speaking of while-loops, and axiomatizing Elgot's work on feedback and iteration, as well as fixed-point semantics.

Recently, Hines and Scott [HiSc07] extended the the work of Haghverdi and Manes-Arbib to more general Σ-structures with certain partially defined traces (cf Section 7 below), so as to account for a general theory of "quantum while-loops" in quantum computing.

In what follows, we give a basic framework for Σ structures sufficient for our purposes.

Definition 4.7 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, $\sum_{i\in I} x_i = \sum_{j\in J}(\sum_{i\in I_j} x_i)$.

(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 $I = \{j\}$.

A morphism of Σ monoids is a function that preserves sums of countably-indexed summable families: i.e. if $\{x_i\}_{i\in I}$ is summable, then so is $\{f(x_i)\}_{i\in I}$ and $f(\sum_{i\in I} x_i) = \sum_{i\in I} f(x_i)$. Σ-monoids form a symmetric monoidal closed category Σ**Mon** (see the discussion in [HiSc07]).

A Σ**Mon**-category $\mathcal{C}$ is a category enriched in Σ**Mon**; i.e. its homsets are enriched with a partial infinitary sum, compatible with composition. Such categories have non-empty homsets, e.g. they have zero morphisms $0_{XY} : X \longrightarrow Y = \sum_{i\in\emptyset} f_i$ for $f_i \in \mathcal{C}(X, Y)$. For details see [MA86,Hag00].

Definition 4.8 A *unique decomposition category* (UDC) $\mathcal{C}$ is a symmetric monoidal Σ**Mon**-category which satisfies the following axiom:

(A) For all $j \in I$ there are morphisms called *quasi injections*: $\iota_j : X_j \longrightarrow \otimes_I X_i$, and *quasi projections*: $\rho_j : \otimes_I X_i \longrightarrow X_j$, 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 4.9 (Finite Matrix 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, $f_{ij} = \rho_i f \iota_j$.*

Thus every morphism $f : \otimes_J X_j \longrightarrow \otimes_I Y_i$ in a UDC can be represented by a matrix; for example f above (with $|I| = m$ and $|J| = n$) is represented by the $m \times n$ matrix $[f_{ij}]$. Composition of morphisms in a UDC then corresponds to matrix multiplication.

Proposition 4.10 (Standard Trace Formula) *Let $\mathcal{C}$ be a unique decomposition category such that for every X, Y, 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*[2] *of f. Then, $\mathcal{C}$ is traced and $\mathtt{Tr}^U_{X,Y}(f) = f_{11} + \sum_{n=0}^{\infty} f_{12} f_{22}^n f_{21}$.*

The trace formula above is called the standard trace, and a UDC with such a trace is called a *traced UDC with standard trace*. Note that a UDC can be traced with a trace different from the standard one. In this paper all traced UDCs are the ones with the standard trace.

We now present some more examples. For further details, see [AHS02,Hag00].

Examples 4.11 (Traced UDC's)

[2] Here $f_{11} : X \to Y, f_{12} : U \to X, f_{21} : X \to U, f_{22} : U \to U$

(1) All the categories in Example 4.6 above. In $\mathbf{Rel}_+$, *all* countable families are summable, and $\sum_{i\in I} R_i = \cup_i R_i$. In the case of **Pfn** and **PInj**, summability of a family of morphisms $\{f_i\}_{i\in I}$ is as given above in the Examples. In this case, the two trace formulas (6) and (7) exactly correspond to the standard trace formula in Proposition 4.10 above.

(2) **SRel**, the category of stochastic relations. Here the objects are measurable spaces $(X, \mathcal{F}_X)$ and maps $f : (X, \mathcal{F}_X) \to (Y, \mathcal{F}_Y)$ are stochastic kernels, i.e. $f : X \times \mathcal{F}_Y \to [0, 1]$ which are bounded measurable in the first variable and subprobability measures in the second. Composition $g{\circ}f(x, C) = \int_Y g(-, C)df(x, -)$ where $f(x, \_)$ is the measure for integration. This category has finite and countable coproducts (which form the tensor). A family $\{f_i\}_{i\in I}$ is summable iff $\sum_{i\in I} f_i(x, Y) \leq 1$ for all $x \in X$.

(3) $\mathbf{Hilb}_2$. Consider the category **cHilb** of Hilbert spaces and linear contractions (norm ≤ 1). Barr [Barr92] defined a contravariant faithful functor ℓ_2 :$\mathbf{PInj}^{op} \longrightarrow \mathbf{cHilb}$ by: for a set X, $\ell_2(X)$ is 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 with norm given by $||a|| = (\sum_{x\in X} |a(x)|^2)^{1/2}$ and inner product given by $< a, b >= \sum_{x\in X} a(x)\overline{b(x)}$ for $a, b \in \ell_2(X)$. Given a partial injection $f : X \to Y$ in **PInj**, then $l_2(f) : \ell_2(Y) \to \ell_2(X)$ is defined by

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

This gives a correspondence between partial injective functions and partial isometries on Hilbert spaces (see also [Gi95a,Abr96].) Let $\mathbf{Hilb_2} = \ell_2[\mathbf{PInj}]$. Its objects are $\ell_2(X)$ for a set X and morphisms $u : \ell_2(X) \longrightarrow \ell_2(Y)$ are of the form $\ell_2(f)$ for some partial injective function $Y \xrightarrow{f} X$. Hence, $\mathbf{Hilb_2}$ is a nonfull subcategory of **Hilb**. It forms a traced UDC with respect to the induced ℓ_2 structure, as follows:

- $\ell_2(X) \oplus \ell_2(Y) \cong \ell_2(X \uplus Y)$ is a tensor product in $\mathbf{Hilb}_2$ (but is a biproduct in **Hilb**) with unit $\ell_2(\emptyset)$.
- Quasi injections and projections = their ℓ_2 images from **PInj**.
- Define: A $\mathbf{Hilb}_2$ family $\{\ell_2(f_i)\}$ is *summable* if
  - · $\{f_i\}$ is summable in **PInj**
  - · In that case, $\sum_i \ell_2(f_i) =_{def} \ell_2(\sum_i f_i)$.
- $\mathbf{Hilb_2}$ is traced. Given

$$u : \ell_2(X) \oplus \ell_2(U) \longrightarrow \ell_2(Y) \oplus \ell_2(U)$$

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

  where $u = \ell_2(f)$ with $f : Y \uplus U \longrightarrow X \uplus U \in$ **PInj**.
- Since **PInj** is self-dual, $\ell_2 : \mathbf{PInj} \to \mathbf{Hilb}_2$ is an equivalence of categories. Here is a chart giving some explicit equivalences:

$\mathbf{PInj}(X,Y)$	$\mathbf{Hilb}(\ell_2(Y),\ell_2(X))$
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

- Many (although not all) of the above examples of traced UDC's are special cases of the Partially Additive Categories of Manes and Arbib [MA86]. Those also form traced UDC's with standard trace formula.

4.3 The Int Construction

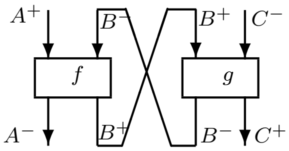
Starting with a symmetric traced monoidal category $\mathcal{C}$, we now describe a compact closed category $\mathbf{Int}(\mathcal{C})$ given in [JSV96] (which is isomorphic to the category $\mathcal{G}(\mathcal{C})$ in [Abr96]). We follow the treatment in [Abr96], and actually give the construction for $\mathcal{G}(\mathcal{C})$; for simplicity, we call both these categories the **Int** construction. The reason for the name is in Exercise 4.13 below.

Definition 4.12 (The Int Construction) Given a traced monoidal category $\mathcal{C}$ we define a compact closed category $\mathbf{Int}(\mathcal{C}) \cong \mathcal{G}(\mathcal{C})$ as follows:

- Objects: Pairs of objects (A^+, A^-) where A^+ and A^- are objects of $\mathcal{C}$.
- Arrows: An arrow $f : (A^+, A^-) \longrightarrow (B^+, B^-)$ in $\mathbf{Int}(\mathcal{C})$ is an arrow $f : A^+ \otimes B^- \longrightarrow A^- \otimes B^+$ in $\mathcal{C}$.
- Identity: $1_{(A^+,A^-)} = s_{A^+,A^-}$, the symmetry or "twist" map.
- Composition: Arrows $f : (A^+, A^-) \longrightarrow (B^+, B^-)$ and $g : (B^+, B^-) \longrightarrow (C^+, C^-)$ have composite $g{\circ}f : (A^+, A^-) \longrightarrow (C^+, C^-)$ given by:

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

  where $\alpha = (1_{A^+} \otimes 1_{B^-} \otimes s_{C^-,B^+})(1_{A^+} \otimes s_{C^-,B^-} \otimes 1_{B^+})$ and $\beta = (1_{A^-} \otimes 1_{C^+} \otimes s_{B^+,B^-})(1_{A^-} \otimes s_{B^+,C^+} \otimes 1_{B^-})(1_{A^-} \otimes 1_{B^+} \otimes s_{B^-,C^+})$. Pictorially, $g{\circ}f$ is given by symmetric feedback:



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

(B^+, B^-) and $g : (C^+, C^-) \longrightarrow (D^+, D^-)$, $f \otimes g = (1_{A^-} \otimes s_{B^+,C^-} \otimes 1_{D^+})(f \otimes g)(1_{A^+} \otimes s_{C^+,B^-} \otimes 1_{D^-})$

- Unit: (I, I).
- Duality: The dual of (A^+, A^-) is given by $(A^+, A^-)^* = (A^-, A^+)$ where the unit $\eta : (I, I) \longrightarrow (A^+, A^-) \otimes (A^+, A^-)^* =_{def} s_{A^-,A^+}$ and the counit map $\epsilon : (A^+, A^-)^* \otimes (A^+, A^-) \longrightarrow (I, I) =_{def} s_{A^-,A^+}$.
- Internal Homs: As usual, $(A^+, A^-) \multimap (B^+, B^-) = (A^+, A^-)^* \otimes (B^+, B^-) = (A^- \otimes B^+, A^+ \otimes B^-)$.

Following Abramsky [Abr96], we interpret the objects of $\mathbf{Int}(\mathcal{C})$ in a game-theoretic manner: A^+ is the type of "moves by Player (the System)" and A^- is the type of "moves by Opponent (the Environment)". The composition of morphisms in $\mathbf{Int}(\mathcal{C})$ is connected to Girard's execution formula (see below) . In [Abr96] it is pointed out that $\mathcal{G}(\mathbf{PInj})$ captures the essence of the original Girard GoI interpretation in [Gi89a] (we discuss this in more detail below), while $\mathcal{G}(\omega\text{-CPO}_\perp)$ is the model of GoI in [AJ94a].

Exercise 4.13 (Why Int?) The **Int** construction above is analogous to (in fact, it yields) the construction of the integers $\mathbb{Z}$ from the natural numbers $\mathbb{N}$. Indeed (using the notation above): put an equivalence relation on $\mathbb{N} \times \mathbb{N}$ by defining: $(A^+, A^-) \sim (B^+, B^-)$ iff $A^+ + B^- = A^- + B^+$ in $\mathbb{N}$. Prove this yields $\mathbb{Z}$. Harder question: show how this is a special case of the **Int** construction.

Translating the work of [JSV96] in our setting we obtain that $\mathbf{Int}(\mathcal{C})$ is a kind of "free compact closure" of $\mathcal{C}$ at the bicategorical level (for which the reader is referred to [JSV96]):

Proposition 4.14 *Let $\mathcal{C}$ be a traced symmetric monoidal category*

(1) $\mathbf{Int}(\mathcal{C})$ *defined above is a compact closed category. Moreover, $F_\mathcal{C} : \mathcal{C} \longrightarrow$* $\mathbf{Int}(\mathcal{C})$ *defined by $F_\mathcal{C}(A) = (A, I)$ and $F_\mathcal{C}(f) = f$ is a full and faithful embedding.*

(2) The inclusion of 2-categories $\mathbf{CompCl} \hookrightarrow \mathbf{TraMon}$ *of compact closed categories into traced monoidal ones has a left biadjoint with unit having component at $\mathcal{C}$ given by $F_\mathcal{C}$.*

We remark that [Has08] shows (in the general setting of [JSV96]) that a traced monoidal category $\mathcal{C}$ is closed iff the canonical inclusion $\mathcal{C} \hookrightarrow \mathbf{Int}(\mathcal{C})$ has a right adjoint. Finally, we should remark that the **Int** construction has seen other applications in recent categorical studies of the semantics of quantum computing, arising from the fundamental paper [AbCo04].

5 What is the Geometry of Interaction?

5.1 Dynamical Invariants for Cut-Elimination

Recall the earlier discussion of Cut-Elimination and the rewriting theory of proofs. We begin with some general questions:

- How do we mathematically model the dynamics of cut-elimination (i.e. the movement of information in the rewriting of the proof trees)?
- Are there dynamical (mathematical) invariants φ for proof normalization, that is: if π rewrites to π', then $\varphi(\pi) = \varphi(\pi')$?
- In what sense is cut-elimination related to recent theories of abstract algorithms?

Recall that in categorical proof theory, for any logic $\mathcal{L}$, we may interpret proofs of sequents $\Gamma \vdash \Delta$ as arrows (in an appropriate structured category $\mathcal{C}$) as in (4) above. This gives an interpretation function (call a *denotation*) $[\![-]\!]_d : \mathcal{L} \to \mathcal{C}$ which satisfies: for any rewriting step $\rightsquigarrow$ in the cut-elimination process, if $\pi \rightsquigarrow \pi'$ then $[\![\pi]\!]_d = [\![\pi']\!]_d$. Such functions $[\![\pi]\!]_d$ lead to a rather bland notion of "invariant" for cut-elimination. Indeed, "$\rightsquigarrow$" implies simply denotational equality, the equations one must impose to give the appropriate algebraic structure of the category of proofs (depending on the logic): e.g. cartesian, cartesian closed, monoidal closed, etc. We search for more meaningful invariants, with deeper connections to the dynamics.

Girard's Geometry of Interaction (GoI) program was the first attempt to model, in a mathematically sophisticated way, the dynamics of cut-elimination, and in particular to find an invariant (the Execution Formula) with more subtle features. The first proposal appeared in [Gi89], followed by an important series of papers [Gi89a,Gi88,Gi95a] written in the language of operator algebras. His recent work [Gi07,Gi08] has moved towards the framework of von Neumann algebras. However, it became clear early on, from lectures of Abramsky [AJ94a,Abr96] and also Hyland in the early 1990's that more simple conceptual machinery, based on traced monoidal categories, suffices to understand many of the fundamental algebraic and geometric ideas underlying early GoI. This was explored by us in a series of papers [AHS02,HS04a,HS04b,HS05a,Hag06]. In what follows we shall explore some algebraic aspects of Girard's early GoI 1, and the notion of information flow. We leave it an open question how to connect this up with Girard's more recent ideas based on von Neumann algebras [Gi07,Gi08].

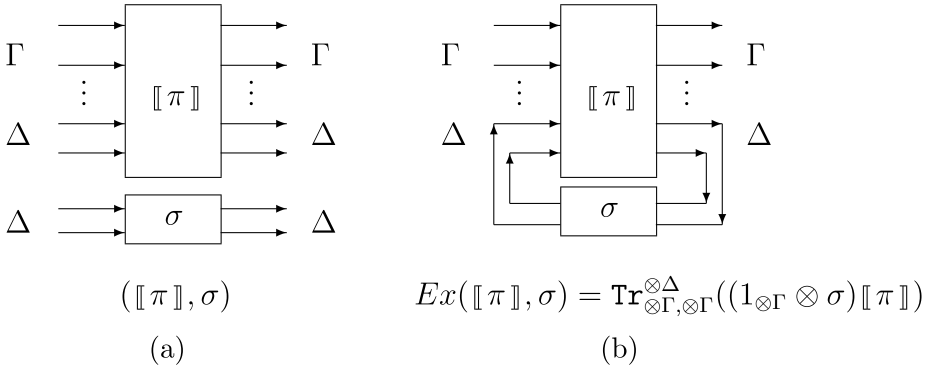
The basic idea of [Gi89a] is to consider proofs as certain matrix operators on a C^*-algebra $\mathcal{B}(\mathsf{H})$ of bounded linear operators on a Hilbert space H. We shall look at proofs of 1-sided sequents in **LL**, say $\pi : \vdash \Gamma$, where Γ is a list of formulas. A key notion in Girard's work was to keep track of all the cut formulas used in a proof. These *general proofs* have the form $\pi :\vdash [\Delta], \Gamma$ where Δ is a list of all the Cut formulas generated from applying the Cut Rule, as follows:

$$\frac{\vdash [\Delta], \Gamma, A \quad \vdash [\Delta'], A^{\perp}, \Gamma'}{\vdash [\Delta, \Delta', A, A^{\perp}] \ \ \Gamma, \ \Gamma'} \text{Cut}$$

We think of this Cut Rule as taking the cut formulas $A, A^{\perp}$ (in that order) and putting them on a stack: the (ordered) list $[\Delta, \Delta', A, A^{\perp}]$.

Thus, in a general proof $\pi :\vdash [\Delta], \Gamma$, we have that Δ is an even length list of cut formulas, say $\Delta = A_1, A_1^{\perp}, \cdots, A_m, A_m^{\perp}$. In general suppose $|\Delta| = 2m$ and $|\Gamma| = n$, so that $\vdash [\Delta], \Gamma$ has $n + 2m$ formulas. Let us informally describe the GoI ingredients.

A key aspect of Girard's interpretation is to consider a *Dynamic Interpretation* $[\![-]\!]$ of proofs. A proof $\pi : \vdash [\Delta], \Gamma$ will be modelled by a pair of I/O (input-output) boxes (Figure 9(a).) Cut-elimination will be modelled by a diagram involving the feedback on σ (Figure 9(b).)



$([\![\pi]\!], \sigma)$

(a)

$Ex([\![\pi]\!], \sigma) = \mathtt{Tr}^{\otimes\Delta}_{\otimes\Gamma,\otimes\Gamma}((1_{\otimes\Gamma} \otimes \sigma)[\![\pi]\!])$

(b)

Fig. 9. Proofs of $\vdash [\Delta], \Gamma$ as I/O Boxes

Formulas in sequents are interpreted (uniformly) by a special object U in the category $\mathcal{C}$. In Girard's GoI 1, $\mathcal{C} = \mathbf{Hilb}$, the category of Hilbert spaces and bounded linear maps and $U = \ell_2(\mathbb{N}) = \ell^2$, the Hilbert space of square summable sequences. Indeed, the interpretation actually occurs in $\mathbf{Hilb}_2$ (Example 4.11 (3)). We know that $\mathbf{Hilb}_2$ is equivalent to **PInj** under the ℓ_2 functor; it follows that the GoI 1 interpretation below may equally well be thought-of as occurring in **PInj**, with $U = \mathbb{N}$.

In the GoI interpretation of logic, formulas are interpreted as *types* via a notion of *orthogonality*, $(\ )^{\perp}$, on certain hom-sets. Such notions of orthogonality are needed both to define types (as sets equal to their biorthogonal) as well as to give convergence-like properties of the Execution Formula. Below we introduce such notions concretely in Definition 6.3, and more abstractly (following [HylSc03]) in Definition 7.9.

Proofs on the other hand are interpreted as morphisms in $\mathbf{Int}(\mathcal{C})$. Suppose we have a proof π of a sequent $\vdash [\Delta], \Gamma$, with $|\Gamma| = n$ and $|\Delta| = 2m$. This is interpreted as a morphism $[\![\pi]\!]$ in $\mathbf{Int}(\mathcal{C})$ from (U^n, U^{2m}) to itself, where U^k is a shorthand for the k-fold tensor product of U with itself: equivalently, as a map $U^{n+2m} \xrightarrow{[\![\pi]\!]} U^{n+2m}$ in $\mathcal{C}$. Notice that all formulas Γ and Δ occur twice (i.e. as both inputs and outputs to $[\![\pi]\!]$) in Figure 9(a).

Remark 5.1 (GoI Notation) For ease of computing the GoI interpretation of proofs π (using their graphical representations as in Figure 9 above) , we often label the inputs and outputs by the I/O formulas themselves (e.g. Γ, Δ in Figure 9), rather than the object U (which uniformly interprets all formulas).

The interpretation of proofs is completed by defining the morphism $\sigma := s^{\otimes m}$ representing Δ, where s is the symmetry (i.e. the identity map in $\mathbf{Int}(\mathcal{C})$). The precise sense in which we interpret formulas and proofs will be described in Section 6 below.

To recap, we will interpret proofs-with-cuts $\pi : \vdash [\Delta], \Gamma$ as pairs $([\![\pi]\!], \sigma)$ such that:

- $[\![\pi]\!] : U^{n+2m} \longrightarrow U^{n+2m}$ is defined inductively on proofs, and
- $\sigma : U^{2m} \longrightarrow U^{2m} = s^{\otimes m}$ (the m-fold tensor product of the symmetry morphism $s_{U,U}$ with itself) represents the cuts Δ.

Here, $|\Delta| = 2m$ and $|\Gamma| = n$. If $\Delta = \emptyset$, π is cut-free and $\sigma = 0$ will be a zero morphism. (This will always exist, since our categories will be Σ-monoid enriched). We note that in Girard's model $\mathbf{Hilb}_2$ and our $*$-category approach in Proposition 8.7 below, $([\![\pi]\!], \sigma)$ are partial symmetries).

As we are working in a traced UDC, we can use the matricial representation of arrows (see Proposition 4.9) to write $[\![\pi]\!]$ as a block matrix:

$$[\![\pi]\!] = \left(\begin{array}{c|c} \pi_{11} & \pi_{12} \\ \hline \pi_{21} & \pi_{22} \end{array}\right)$$

The dynamics of proofs (cut-elimination) will be interpreted using the Execution Formula defined in formula (8) below. This is illustrated in Figure 9(b). In any traced UDC, this can be represented as a sum, as in (9) below.

Execution/Trace Formula

$$\begin{aligned} EX(\llbracket \pi \rrbracket, \sigma) &=_{def} \mathtt{Tr}^{\otimes\Delta}_{\otimes\Gamma,\otimes\Gamma}((1_{\otimes\Gamma} \otimes \sigma) \llbracket \pi \rrbracket) && (8) \\ &= \pi_{11} + \sum_{n\geq 0} \pi_{12}(\sigma\pi_{22})^n(\sigma\pi_{21}) && (9) \end{aligned}$$

Note that the underlying category $\mathcal{C}$ is a traced UDC and more generally in Section 8, a traced category , where $\llbracket \otimes\Delta \rrbracket = U^{2m}$, $\llbracket \otimes\Gamma \rrbracket = U^n$. Thus $EX(\llbracket \pi \rrbracket, \sigma) : U^n \to U^n$ exists as a $\mathcal{C}$-morphism.

The essential mathematical ingredients at work in GoI were understood to consist of a traced symmetric monoidal category, a traced endofunctor and the special object U we alluded to above, called a *reflexive* object. Such structures are called *GoI Situations*, see below for detailed definitions. In [HS04a], we showed that Girard's GoI 1 can be modeled categorically, using GoI Situations where the underlying category $\mathcal{C}$ is a traced UDC. In particular, we proved that the original operator algebraic framework in [Gi89a] is captured by the GoI Situation on the category $\mathbf{Hilb}_2$, see Proposition 6.13 below.

Remark 5.2 (GoI, Path-Based Computing, Complexity) Important approaches to GoI arose in work of V. Danos and L. Regnier and coworkers [Dan90,DR95,Lau01,MR91]. This work analyzes information flow in β-reduction of untyped lambda calculus, using paths in proof-nets. The GoI execution formula may be analyzed as a certain kind of sum-of-paths formula, breaking down β-reduction to local reversible asynchronous steps. The authors give detailed and profound analyses of the kinds of paths and information flow this viewpoint represents, together with fundamental algebraic models for computation. This leads to important connections with previous work in the geometry of β-reduction; in particular, to relations of GoI with optimal reduction [GAL92].

GoI has also had some connections with complexity [GSS,BP01] in particular with evaluation strategies and rates of growth of numerical measures assigned to proofs in bounded logics. In the case of Traced UDC-style models of GoI and GoI Situations as studied here, more recently Schöpp [Sch07] used this machinery to study fragments of bounded (affine) linear logic suitable for studying logarithmic space.

In a different direction, in [AHS02], a general analysis of algebraic models of GoI is carried out. There it is shown how to use GoI Situations to obtain models of the $\{!, \multimap\}$ fragment of linear logic, presented in terms of *linear combinatary algebras.* These are certain combinatory algebras $(\mathcal{A}, \cdot)$ equipped with a map $\mathbf{!} : \mathcal{A} \to \mathcal{A}$ and constants $B, C, I, K, W, D, \delta, F$ satisfying the combinatory identities for a Hilbert-style axiomatization of $\{!, \multimap\}$. The method is sketched as follows.

Let $\mathcal{C}$ be a traced smc, with an endofunctor $T : \mathcal{C} \to \mathcal{C}$ and an object (called a *reflexive object*) $U \in \mathcal{C}$ with retractions $U \otimes U \lhd U$, $I \lhd U$, and $TU \lhd U$. Then if T satisfies some reasonable axioms and setting $V = (U, U)$ and $I = (I, I)$, it is shown in [AHS02] how the homset $\mathbf{Int}(\mathcal{C})(I, V) = \mathcal{C}(U, U)$ naturally inherits the structure of a linear combinatory algebra. For example, in the case of $\mathcal{C} = \mathbf{PInj}$, $\mathbb{N}$ is such a reflexive object, with endofunctor $T(-) = \mathbb{N} \times (-)$. This example underlies the original Girard GoI constructions. The model in [AJ94a] likewise arises from $\mathbf{Int}(\mathbf{CPO}_\perp)$. Moreover, Girard's original operator-theoretic models (in the category of Hilbert spaces), as well as Danos-Regnier's *small model* [DR95] are also captured in the above framework using some additional functorial structure (see [Hag00]).

We should mention Abramsky's paper [Abr07] which, while discussing Temperley-Lieb Algebra in knot theory, develops a version of *planar* GoI. In a different vein, Fuhrman and Pym [FP07] develop a categorical framework for obtaining models for classical logic using a GoI/Int construction applied to certain extensions of symmetric linearly distributive categories, along the lines of the work of Blue-Cockett-Seely.

6 GoI Interpretation of MELL

Geometry of Interaction interprets an underlying logical system at three levels: *formulas, proofs* and *cut-elimination.* We shall carry out this interpretation for **MELL** without units in the following sections. There are two fundamental ingredients in a GoI interpretation: (i) A GoI Situation containing the underlying traced UDC, and (ii) A notion of orthogonality. We begin by defining these ingredients. We shall discuss generalizations and extensions of these notions in later sections.

Definition 6.1 A *GoI Situation* is a triple $(\mathcal{C}, T, U)$ where:

(1) $\mathcal{C}$ is a traced symmetric monoidal category

(2) $T : \mathcal{C} \longrightarrow \mathcal{C}$ is a traced symmetric monoidal functor with the following *monoidal* retractions (i.e. the retraction pairs are monoidal natural transformations):
   (a) $TT \lhd T \ (e, e')$ (Comultiplication)
   (b) $Id \lhd T \ (d, d')$ (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.

(3) U is an object of $\mathcal{C}$, called *a reflexive object*, with retractions:
   (a) $U \otimes U \lhd U \ (j, k)$, (b) $I \lhd U$, and (c) $TU \lhd U \ (u, v)$.

Here $TT \lhd T\ (e, e')$ means that there are monoidal natural transformations $e_X : TTX \longrightarrow TX$ and $e'_X : TX \longrightarrow TTX$ such that $e'e = 1_{TT}$. We say that TT *is a retract of* T. Similarly for the other items.

Before we proceed, let's consider some examples of GoI Situations $(\mathcal{C}, T, U)$. For comparison of our notation with the notation of Girard and his students, see Appendix B.

Examples 6.2

(1) $(\mathbf{PInj}, \mathbb{N} \times -, \mathbb{N})$. Here $\mathbb{N}$ is the set of natural numbers. The functor $T = \mathbb{N} \times -$, is defined as $TX = \mathbb{N} \times X$ and for a morphism $f : X \longrightarrow Y$, $Tf = 1_{\mathbb{N}} \times f$. We shall refer the reader to [AHS02] for details on this and the following examples. However, we include a few definitions for illustration. For example, consider the cases for $U \otimes U \lhd U\ (j, k)$, Comultiplication and Contraction:

• $\mathbb{N} \uplus \mathbb{N} \lhd \mathbb{N}\ (j, k)$ is defined by $j : \mathbb{N} \uplus \mathbb{N} \longrightarrow \mathbb{N}, j(1, n) = 2n, j(2, n) = 2n + 1$ and $k : \mathbb{N} \longrightarrow \mathbb{N} \uplus \mathbb{N}$,

$$k(n) = \begin{cases} (1, n/2), & \text{if } n \text{ even;} \\ (2, (n-1)/2), & \text{if } n \text{ odd .} \end{cases}$$

Clearly $kj = 1_{\mathbb{N} \uplus \mathbb{N}}$.

• (Comultiplication) $\mathbb{N} \times (\mathbb{N} \times X) \xrightarrow{e_X} \mathbb{N} \times X$ 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, $e_X(n_1, (n_2, x)) = (\langle n_1, n_2 \rangle, x)$. Given $f : X \longrightarrow Y$, $(1_{\mathbb{N}} \times f)e_X((n_1, (n_2, x)) = (\langle n_1, n_2 \rangle, f(x)) = e_Y(1_{\mathbb{N}} \times (1_{\mathbb{N}} \times f)(n_1, (n_2, x))$ 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 $\langle n_1, n_2 \rangle = n$.
$e'_X e_X(n_1, (n_2, x)) = e'_X(\langle n_1, n_2 \rangle, x) = (n_1, (n_2, x))$ for all $n_1, n_2 \in \mathbb{N}$ and $x \in X$.

• (Contraction) $(\mathbb{N} \times X) \uplus (\mathbb{N} \times X) \xrightarrow{c_X} \mathbb{N} \times X$ 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 $c'_X c_X(2,(n,x)) = c'_X(2n+1,x) = (2,(n,x))$.

(2) $(\mathbf{Pfn}, \mathbb{N} \times -, \mathbb{N})$.

(3) $(\mathbf{Rel}_+, \mathbb{N} \times -, \mathbb{N})$.

(4) $(\mathbf{SRel}, T, \mathbb{N}^\infty)$. Here $T : \mathbf{SRel} \longrightarrow \mathbf{SRel}$ is defined 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 $f : (X, \mathcal{F}_X) \longrightarrow (Y, \mathcal{F}_Y)$, $Tf((n,x), \biguplus_{i\in\omega} B_i) = f(x, B_n)$.

Note that throughout this section we shall be working with GoI Situations where the underlying category is a traced UDC.

Definition 6.3 (Orthogonality and Types) Let f, g be morphisms in $\mathcal{C}(U,U)$. We say that f is *nilpotent* if $f^k = 0$ for some $k \geq 1$. We say that f is *orthogonal* to g, denoted $f \perp g$ if gf is nilpotent. Orthogonality is a symmetric relation and it makes sense because 0_{UU} exists. Also, $0 \perp f$ for all $f \in \mathcal{C}(U,U)$.

Given a subset X of $\mathcal{C}(U,U)$, we define

$$X^\perp = \{f \in \mathcal{C}(U,U) | \forall g(g \in X \Rightarrow f \perp g)\}$$

A *type* is any subset X of $\mathcal{C}(U,U)$ such that $X = X^{\perp\perp}$. Note that types are inhabited, since 0_{UU} belongs to every type.

6.1 GoI Interpretation of formulas

Formulas are interpreted by *types* as defined above.

Definition 6.4 Consider a GoI situation $(\mathcal{C}, T, U)$ as above with j_1, j_2, k_1, k_2 components of j and k respectively. Let A be an **MELL** formula. We define *the GoI interpretation of* A, denoted θA, inductively as follows:

(1) If $A \equiv \alpha$ that is A is an atom, then $\theta A = X$ an arbitrary type.

(2) If $A \equiv \alpha^\perp$, $\theta A = X^\perp$, where $\theta\alpha = X$ is given by assumption.

(3) If $A \equiv B \otimes C$, $\theta A = Y^{\perp\perp}$, where $Y = \{j_1 a k_1 + j_2 b k_2 | a \in \theta B, b \in \theta C\}$.

(4) If $A \equiv B \invamp C$, $\theta A = Y^\perp$, where $Y = \{j_1 a k_1 + j_2 b k_2 | a \in (\theta B)^\perp, b \in (\theta C)^\perp\}$.

(5) If $A \equiv \mathbf{!}B$, $\theta A = Y^{\perp\perp}$, where $Y = \{uT(a)v | a \in \theta B\}$.

(6) If $A \equiv ?B$, $\theta A = Y^\perp$, where $Y = \{uT(a)v | a \in (\theta B)^\perp\}$.

An easy consequence of the definition that $(\theta A)^\perp = \theta A^\perp$ for any formula A.

In this section we define the GoI interpretation for proofs of **MELL** without the units. Proofs are interpreted in the homset $\mathcal{C}(U,U)$ of endomorphisms of U.

Convention: All identity morphisms are on tensor copies of U; however we adopt the convention of writing 1_Γ instead of 1_{U^n} with $|\Gamma| = n$, where U^n denotes the n-fold tensor product of U with itself. The retraction pairs are fixed once and for all.

Every **MELL** sequent will be of the form $\vdash [\Delta], \Gamma$ where Γ is a sequence of formulas and Δ is a sequence of cut formulas that have already been made in the proof of $\vdash \Gamma$ (e.g. $A, A^\perp, B, B^\perp$). This is used to keep track of the cuts that are already made in the proof of $\vdash \Gamma$. Suppose that Γ consists of n and Δ consists of $2m$ formulas. Then a proof π of $\vdash [\Delta], \Gamma$ is represented by a morphism $[\![\pi]\!] \in \mathcal{C}(U^{n+2m}, U^{n+2m})$. Recall that this corresponds to a morphism from U to itself, using the retraction morphisms $U \otimes U \lhd U\,(j,k)$. However, it is much more convenient to work in $\mathcal{C}(U^{n+2m}, U^{n+2m})$ (matrices on $\mathcal{C}(U,U)$). Define the morphism $\sigma : U^{2m} \longrightarrow U^{2m}$, as $\sigma = s \otimes \cdots \otimes s$ (m-copies) where s is the symmetry morphism, the 2×2 antidiagonal matrix $[a_{ij}]$, where $a_{12} = a_{21} = 1; a_{11} = a_{22} = 0$. Here σ represents the cuts in the proof of $\vdash \Gamma$, i.e. it models Δ. If Δ is empty (that is for a cut-free proof), we define $\sigma : I \longrightarrow I$ to be the zero morphism 0_{II}. Note that $U^0 = I$ where I is the unit of the tensor in the category $\mathcal{C}$.

Given block matrices A, B, by $A \otimes B$ we mean the block matrix with A and B on the main diagonal (the rest zeros). Thus σ above is the $2m \times 2m$ block matrix with the 2×2 matrix s along the main diagonal.

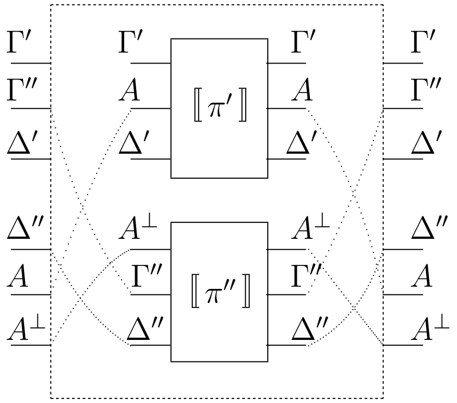
Definition 6.5 (The GoI Interpretation) Let π be a proof of $\vdash [\Delta], \Gamma$. We define *the GoI interpretation of* π, denoted by $[\![\pi]\!]$, by induction on the length of the proof as follows. We illustrate two cases (Cut and Contraction) geometrically below.

(1) π is an *axiom* $\vdash A, A^\perp$, then $m = 0$, $n = 2$ and $[\![\pi]\!] = s = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$.

(2) π is obtained using the *cut* rule on π' and π'' that is

$$\dfrac{\begin{array}{c}\pi' \\ \vdots \\ \vdash [\Delta'], \Gamma', A\end{array} \qquad \begin{array}{c}\pi'' \\ \vdots \\ \vdash [\Delta''], A^\perp, \Gamma''\end{array}}{\vdash [\Delta', \Delta'', A, A^\perp], \Gamma', \Gamma''}\ \text{cut}$$

Then we define $[\![\,\pi\,]\!]$ as follows: $[\![\,\pi\,]\!] = \tau^{-1}([\![\,\pi'\,]\!] \otimes [\![\,\pi''\,]\!])\tau$ where τ and τ^{-1} are the indicated permutations of the interface (the identity on Γ', Δ'):



(3) π is obtained using the *exchange* rule on the formulas A_i and A_{i+1} in Γ'. That is π is of the form

$$\begin{array}{c} \pi' \\ \vdots \\ \dfrac{\vdash [\Delta], \Gamma'}{\vdash [\Delta], \Gamma}\ \text{exchange} \end{array}$$

where in Γ' we have A_i, A_{i+1}. Then, $[\![\,\pi\,]\!]$ is obtained from $[\![\,\pi'\,]\!]$ by interchanging the rows i and $i+1$. So suppose that $\Gamma' = \Gamma'_1, A_i, A_{i+1}, \Gamma'_2$, then $\Gamma = \Gamma'_1, A_{i+1}, A_i, \Gamma'_2$ and $[\![\,\pi\,]\!] = \tau^{-1}[\![\,\pi'\,]\!]\tau$, where $\tau = 1_{\Gamma'_1} \otimes s \otimes 1_{\Gamma'_2 \otimes \Delta}$.

(4) π is obtained using an application of the *par* rule, that is π is of the form:

$$\begin{array}{c} \pi' \\ \vdots \\ \dfrac{\vdash [\Delta], \Gamma', A, B}{\vdash [\Delta], \Gamma', A \parr B}\ \parr \end{array}$$

Then $[\![\,\pi\,]\!] = g[\![\,\pi'\,]\!]f$, where $f = 1_{\Gamma'} \otimes k \otimes 1_\Delta$ and $g = 1_{\Gamma'} \otimes j \otimes 1_\Delta$, recalling that $U \otimes U \lhd U\ (j, k)$.

(5) π is obtained using an application of the *times* rule, that is π has the form

$$\dfrac{\begin{array}{c}\pi' \\ \vdots \\ \vdash [\Delta'], \Gamma', A\end{array} \qquad \begin{array}{c}\pi'' \\ \vdots \\ \vdash [\Delta''], \Gamma'', B\end{array}}{\vdash [\Delta', \Delta''], \Gamma', \Gamma'', A \otimes B}\ \text{times}$$

Then $[\![\, \pi \,]\!] = g\tau^{-1}([\![\, \pi' \,]\!] \otimes [\![\, \pi'' \,]\!])\tau f$, where τ is a permutation and $f = 1_{\Gamma' \otimes \Gamma''} \otimes k \otimes 1_{\Delta' \otimes \Delta''}$ and $g = 1_{\Gamma' \otimes \Gamma''} \otimes j \otimes 1_{\Delta' \otimes \Delta''}$.

(6) π is obtained from π' by an *of course* rule; that is π has the form :

$$\dfrac{\begin{array}{c}\pi' \\ \vdots \\ \vdash [\Delta], ?\Gamma', A\end{array}}{\vdash [\Delta], ?\Gamma', !A}\ \text{of course}$$

Then $[\![\, \pi \,]\!] = ((ue_U)^{\otimes n} \otimes u \otimes u^{\otimes 2m})\varphi^{-1}T((v^{\otimes n} \otimes 1_A \otimes 1_\Delta)[\![\, \pi' \,]\!](u^{\otimes n} \otimes 1_A \otimes 1_\Delta))\varphi((e'_U v)^{\otimes n} \otimes v \otimes v^{\otimes 2m})$, where $TT \lhd T\,(e, e')$, $|\Gamma'| = n, |\Delta| = 2m$, and $\varphi : (T^2U)^{\otimes n} \otimes TU \otimes (TU)^{\otimes 2m} \longrightarrow T((TU)^{\otimes n} \otimes U \otimes U^{\otimes 2m})$ is the canonical isomorphism.

(7) π is obtained from π' by the *dereliction* rule, that is π is of the form :

$$\dfrac{\begin{array}{c}\pi' \\ \vdots \\ \vdash [\Delta], \Gamma', A\end{array}}{\vdash [\Delta], \Gamma', ?A}\ \text{dereliction}$$

Then $[\![\, \pi \,]\!] = (1_{\Gamma'} \otimes ud_U \otimes 1_\Delta)[\![\, \pi' \,]\!](1_{\Gamma'} \otimes d'_U v \otimes 1_\Delta)$ where $Id \lhd T\,(d, d')$.

(8) π is obtained from π' by the *weakening* rule, that is π is of the form:

$$\dfrac{\begin{array}{c}\pi' \\ \vdots \\ \vdash [\Delta], \Gamma'\end{array}}{\vdash [\Delta], \Gamma', ?A}\ \text{weakening}$$

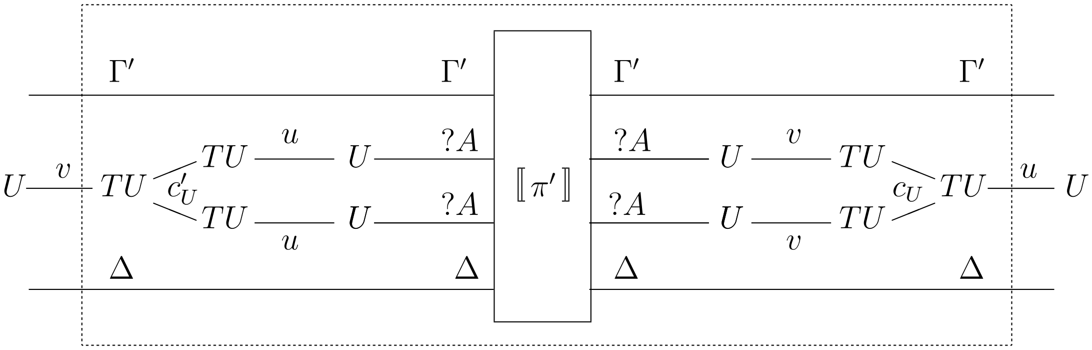
Then $[\![\, \pi \,]\!] = (1_{\Gamma'} \otimes uw_U \otimes 1_\Delta)[\![\, \pi' \,]\!](1_{\Gamma'} \otimes w'_U v \otimes 1_\Delta)$, where $\mathcal{K}_I \lhd T\,(w, w')$.

(9) π is obtained using the *contraction* rule on π', that is

$$\dfrac{\begin{array}{c}\pi' \\ \vdots \\ \vdash [\Delta], \Gamma', ?A, ?A\end{array}}{\vdash [\Delta], \Gamma', ?A}\ \text{contraction}$$

Then we define $[\![\,\pi\,]\!]$ as follows, , where $T\otimes T \lhd T\,(c, c')$:

$$[\![\,\pi\,]\!] = (1_{\Gamma'} \otimes (u(c_U v \otimes v)) \otimes 1_{\Delta})[\![\,\pi'\,]\!](1_{\Gamma'} \otimes (u \otimes u)c'_U v \otimes 1_{\Delta})$$



Example 6.6 Let π be the following proof (of cut applied to the axioms). Categorically it corresponds to *id*∘*id*.

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

Then the GoI semantics of this proof (see the *Cut* rule above) is given by conjugation with a permutation matrix τ:

$$[\![\,\pi\,]\!] = \begin{bmatrix} 1\,0\,0\,0 \\ 0\,0\,0\,1 \\ 0\,1\,0\,0 \\ 0\,0\,1\,0 \end{bmatrix} \left[\begin{array}{cc|cc} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{array}\right] \begin{bmatrix} 1\,0\,0\,0 \\ 0\,0\,1\,0 \\ 0\,0\,0\,1 \\ 0\,1\,0\,0 \end{bmatrix} = \begin{bmatrix} \mathbf{0} & Id_2 \\ Id_2 & \mathbf{0} \end{bmatrix}$$

where Id_2 is the 2×2 identity matrix, $\mathbf{0}$ is the 2×2 zero matrix and the middle matrix is $s \otimes s$.

6.3 GoI Interpretation of cut-elimination

Dynamics is at the heart of the GoI interpretation as compared to denotational semantics and it is hidden in the cut-elimination process. The mathematical model of cut-elimination is given by the *execution formula* defined as follows:

$$EX([\![\,\pi\,]\!], \sigma) = \mathtt{Tr}^{U^{2m}}_{U^n,U^n}((1_{U^n} \otimes \sigma)[\![\,\pi\,]\!]) \tag{10}$$

where π is a proof of the sequent $\vdash [\Delta], \Gamma$. Pictorially this can be represented as in Figure 9(b) in Section 5.2 above.

Note that $EX(\llbracket \pi \rrbracket, \sigma)$ is a morphism from $U^n \longrightarrow U^n$ and it always makes sense since the trace of any morphism in $\mathcal{C}(U^{2m+n}, U^{2m+n})$ is defined. Since we are working with a traced UDC with the standard trace, we can rewrite the execution formula (10) in a more familiar form:

$$EX(\llbracket \pi \rrbracket, \sigma) = \pi_{11} + \sum_{n \geq 0} \pi_{12}(\sigma\pi_{22})^n(\sigma\pi_{21})$$

where $\llbracket \pi \rrbracket = \begin{bmatrix} \pi_{11} & \pi_{12} \\ \pi_{21} & \pi_{22} \end{bmatrix}$. Note that the execution formula defined in this categorical framework *always* makes sense; that is, we do not need a convergence criterion.

The intention here is to prove that the result of this execution formula is what corresponds to the cut-free proof obtained from π using Gentzen's cut-elimination procedure. We will also show that for any proof π of **MELL** the execution formula is a finite sum, which corresponds to termination of computation as opposed to divergence.

Example 6.7 Consider the proof π in Example 6.6 above. Recall also that $\sigma = s$ in this case ($m = 1$). Then

$$EX(\llbracket \pi \rrbracket, \sigma) = \mathtt{Tr}^{U^2}_{U^2,U^2}\left(\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}\right)$$

$$= \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} + \sum_{n \geq 0} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}^n \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} = \left\llbracket \vdash A^{\perp}, A \right\rrbracket .$$

Note that in this case we have obtained the GoI interpretation of the cut-free proof obtained by applying Gentzen's Hauptsatz to the proof π. (Categorically, this just says $id \circ id = id$ in $\mathbf{Int}(\mathcal{C})$, where the composition is obtained dynamically by running the Execution formula). This is generalized in Theorem 6.12 below.

6.4 Soundness of the GoI Interpretation: running the execution formula

In order to ensure that the definition above yields a semantics, we need to prove the soundness of the GoI interpretation. In other words, we have to show that if a proof π is reduced (via cut-elimination) to its cut-free form π', then $EX([\![\pi]\!], \sigma)$ is a finite sum and $EX([\![\pi]\!], \sigma) = [\![\pi']\!]$. Intuitively this says that if one thinks of cut-elimination as computation then $[\![\pi]\!]$ can be thought of as an algorithm. The computation takes place as follows: if we run $EX([\![\pi]\!], \sigma)$, it terminates after finitely many steps (cf. finite sum) and yields a *datum* (cf. cut-free proof). This intuition will be made precise in this section through the definition of *type* and the main theorems (see Theorems 8.12, 8.14). The next result is the analog of the Church-Rosser (or Diamond) property in our setting.

Lemma 6.8 (Associativity of cut) *Let π be a proof of $\vdash [\Gamma, \Delta], \Lambda$ and σ and τ be the morphisms representing the cut-formulas in Γ and Δ respectively. Then*

$$EX([\![\pi]\!], \sigma \otimes \tau) = EX(EX([\![\pi]\!], \tau), \sigma)$$

Proof. Follows from naturality and vanishing II properties of trace. □

Definition 6.9 Let $\Gamma = A_1, \cdots, A_n$. A *datum* of type $\theta\Gamma$ is a morphism $M : U^n \longrightarrow U^n$ such that for any $\beta_1 \in \theta(A_1^{\perp}), \cdots, \beta_n \in \theta(A_n^{\perp})$, $(\beta_1 \otimes \cdots \otimes \beta_n)M$ is nilpotent. An *algorithm* of type $\theta\Gamma$ is a morphism $M : U^{n+2m} \longrightarrow U^{n+2m}$ for some integer m such that for $\sigma : U^{2m} \longrightarrow U^{2m}$ defined in the usual way, $EX(M, \sigma) = \mathtt{Tr}^{U^{2m}}_{U^n, U^n}((1 \otimes \sigma)M)$ is a finite sum and a datum of type $\theta\Gamma$.

Lemma 6.10 *Let $M : U^n \longrightarrow U^n$ and $a : U \longrightarrow U$. Define $CUT(a, M) = (a \otimes 1_{U^{n-1}})M : U^n \longrightarrow U^n$. Note that the matrix representation of $CUT(a, M)$ is the matrix obtained from M by multiplying its first row by a. Then $M = [m_{ij}]$ is a datum of type $\theta(A, \Gamma)$ iff for any $a \in \theta A^{\perp}$, am_{11} is nilpotent and the morphism $ex(CUT(a, M)) = \mathtt{Tr}^A(s^{-1}_{\Gamma,A} CUT(a, M) s_{\Gamma,A})$ is in $\theta(\Gamma)$. Here $s_{\Gamma,A}$ is the symmetry morphism from $\Gamma \otimes A$ to $A \otimes \Gamma$.*

Theorem 6.11 (Proofs as Algorithms) *Let Γ be a sequent, and π be a proof of Γ. Then $[\![\pi]\!]$ is an algorithm of type $\theta\Gamma$.*

Theorem 6.12 (Ex is an invariant) *Let π be a proof of a sequent $\vdash [\Delta], \Gamma$ in* **MELL**. *Then*

(i) *$EX([\![\pi]\!], \sigma)$ is a finite sum.*
(ii) *If π reduces to π' by any sequence of cut-eliminations and $?A$ does not occur in Γ for any formula A, then $EX([\![\pi]\!], \sigma) = EX([\![\pi']\!], \tau)$. So $EX([\![\pi]\!], \sigma)$ is an invariant of reduction.*
(iii) *In particular, if π' is any cut-free proof obtained from π by cut-elimination, then $EX([\![\pi]\!], \sigma) = [\![\pi']\!]$.*

In [HS04a] we show that we obtain the same execution formula as Girard. Note that in Girard's original execution formula $[\![\pi]\!]$ and σ are both $2m + n$ by $2m+n$ matrices. To connect up with our previous notation, let $\tilde{\sigma} = s \otimes \cdots \otimes s$

(m-times.)

Proposition 6.13 (Original Execution Formula) *Let π be a proof of $\vdash [\Delta], \Gamma$. Then in Girard's model* $\mathbf{Hilb}_2$*,*

$$((1-\sigma^2)\sum_{n=0}^{\infty}[\![\pi]\!](\sigma[\![\pi]\!])^n(1-\sigma^2))_{n\times n} = \mathtt{Tr}^{U^{2m}}_{U^n,U^n}((1\otimes\tilde{\sigma})[\![\pi]\!])$$

where $(A)_{n\times n}$ is the submatrix of A consisting of the first n rows and the first n columns.

In the next two sections we discuss further generalizations of the notions of trace and orthogonality. These notions play crucial roles in GoI interpretations.

7 Partial Trace and Abstract Orthogonality

In this section we look at partial traces. The idea of generalizing the abstract trace of [JSV96] to the partial setting is not new. For example, partial traces were already studied in work of Abramsky, Blute, and Panangaden [ABP99], in unpublished lecture notes of Gordon Plotkin [Pl03], work of Blute, Cockett, and Seely [BCS00] (see Remark 7.2), and others. The guiding example in [ABP99] is the relationship between trace class operators on a Hilbert space and Hilbert-Schmidt operators. This allows the authors to establish a close correspondence between trace and nuclear ideals in a tensor $*$-category. Plotkin's work develops a theory of Conway ideals on biproduct categories, and an associated categorical trace theory. Unfortunately none of these extant theories is appropriate for Girard's GoI. So we present an axiomatization for partial traces suitable for our purposes.

Recall, following Joyal, Street, and Verity [JSV96], a (parametric) trace in a symmetric monoidal category $(\mathcal{C}, \otimes, I, s)$ is a family of maps

$$\mathtt{Tr}^U_{X,Y} : \mathcal{C}(X\otimes U, Y\otimes U) \longrightarrow \mathcal{C}(X,Y),$$

satisfying various well-known naturality equations. A *partial* (parametric) trace requires instead that each $\mathtt{Tr}^U_{X,Y}$ be a partial map (with domain denoted $\mathbb{T}^U_{X,Y}$) and satisfy various closure conditions.

Definition 7.1 (Trace Class) Let $(\mathcal{C}, \otimes, I, s)$ be a symmetric monoidal category. A *(parametric) trace class* in $\mathcal{C}$ is a choice of a family of subsets, for each object U of $\mathcal{C}$, of the form

$$\mathbb{T}^U_{X,Y} \subseteq \mathcal{C}(X\otimes U, Y\otimes U) \text{ for all objects } X,\ Y \text{ of } \mathcal{C}$$

together with a family of functions, called a *(parametric) partial trace*, of the form

$$\mathtt{Tr}^U_{X,Y} : \mathbb{T}^U_{X,Y} \longrightarrow \mathcal{C}(X,Y)$$

subject to the following axioms. Here the parameters are X and Y and a morphism $f \in \mathbb{T}^U_{X,Y}$, by abuse of terminology, is said to be *trace class*.

- **Naturality** in X and Y: For any $f \in \mathbb{T}^U_{X,Y}$ and $g : X' \longrightarrow X$ and $h : Y \longrightarrow Y'$,
$$(h \otimes 1_U) f (g \otimes 1_U) \in \mathbb{T}^U_{X',Y'},$$
$$\text{and} \qquad \mathtt{Tr}^U_{X',Y'}((h \otimes 1_U) f (g \otimes 1_U)) = h\, \mathtt{Tr}^U_{X,Y}(f)\, g.$$
- **Dinaturality** in U: For any $f : X \otimes U \longrightarrow Y \otimes U'$, $g : U' \longrightarrow U$,
$$(1_Y \otimes g) f \in \mathbb{T}^U_{X,Y} \text{ iff } f(1_X \otimes g) \in \mathbb{T}^{U'}_{X,Y},$$
$$\text{and} \qquad \mathtt{Tr}^U_{X,Y}((1_Y \otimes g) f) = \mathtt{Tr}^{U'}_{X,Y}(f(1_X \otimes g)).$$
- **Vanishing I**: $\mathbb{T}^I_{X,Y} = \mathcal{C}(X \otimes I, Y \otimes I)$, and for $f \in \mathbb{T}^I_{X,Y}$
$$\mathtt{Tr}^I_{X,Y}(f) = \rho_Y f \rho_X^{-1}.$$
Here $\rho_A : A \otimes I \longrightarrow A$ is the right unit isomorphism of the monoidal category.
- **Vanishing II**: For any $g : X \otimes U \otimes V \longrightarrow Y \otimes U \otimes V$, if $g \in \mathbb{T}^V_{X \otimes U, Y \otimes U}$, then
$$g \in \mathbb{T}^{U \otimes V}_{X,Y} \text{ iff } \mathtt{Tr}^V_{X \otimes U, Y \otimes U}(g) \in \mathbb{T}^U_{X,Y},$$
$$\text{and} \qquad \mathtt{Tr}^{U \otimes V}_{X,Y}(g) = \mathtt{Tr}^U_{X,Y}(\mathtt{Tr}^V_{X \otimes U, Y \otimes U}(g)).$$
- **Superposing**: For any $f \in \mathbb{T}^U_{X,Y}$ and $g : W \longrightarrow Z$,
$$g \otimes f \in \mathbb{T}^U_{W \otimes X, Z \otimes Y},$$
$$\text{and} \qquad \mathtt{Tr}^U_{W \otimes X, Z \otimes Y}(g \otimes f) = g \otimes \mathtt{Tr}^U_{X,Y}(f).$$
- **Yanking**: $\quad s_{UU} \in \mathbb{T}^U_{U,U}, \quad$ and $\quad \mathtt{Tr}^U_{U,U}(s_{U,U}) = 1_U.$

A symmetric monoidal category $(\mathcal{C}, \otimes, I, s)$ with such a trace class is called a *partially traced category*, or a *category with a trace class*. If we let X and Y be I (the unit of the tensor), we get a family of operations $\mathtt{Tr}^U_{I,I} : \mathbb{T}^U_{I,I} \longrightarrow \mathcal{C}(I,I)$ defining what we call a *non-parametric (scalar-valued) trace*.

Remark 7.2 An early definition of a partial parametric trace is due to Abramsky, Blute and Panangaden in [ABP99]. Our definition is different but related to theirs. First, we have used the Yanking axiom in Joyal, Street and Verity [JSV96], whereas in [ABP99] they use a conditional version of the so-called "generalized yanking"; that is, for $f : X \longrightarrow U$ and $g : U \longrightarrow Y$, $\mathtt{Tr}^U_{X,Y}(s_{U,Y}(f \otimes g)) = gf$ *whenever* $s_{U,Y}(f \otimes g)$ is trace class. In our theory s_{UU} is traceable for all U; on the other hand, many examples in [ABP99] do not have this property. More importantly, we do not require one of the ideal axioms in [ABP99]. Namely, we do **not** ask that for $f \in \mathbb{T}^U_{X,Y}$ and any $h : U \longrightarrow U$, $(1_Y \otimes h)f$ and $f(1_X \otimes h)$ be in $\mathbb{T}^U_{X,Y}$. Indeed in the next section we prove that the categories $(\mathbf{Vec}_{fd}, \oplus)$ of finite dimensional vector spaces, and $(\mathbf{CMet}, \times)$ of complete metric spaces are partially traced. It can be shown that in both categories the above ideal axiom and Vanishing II of [ABP99] fail and hence they are not traced in the sense of [ABP99].

In [BCS00], Blute, Cockett, and Seely develop an interesting and detailed theory of trace (and fixpoint) combinators in a linearly distributive category, including an appropriate version of the *Int* construction of [JSV96] in that setting. The authors take a local view of the trace combinator: rather than assuming that a trace is available at every object, they consider the effect of particular objects having a trace (partiality of trace), as well as restricting to "compatible classes" of trace operators (which guarantees that an object may have at most one trace structure.)

One is obliged to say that there are many different approaches to partial categorical traces and ideals; ours is geared to the details of Girard's GoI. We believe our traceability conditions are most naturally formulated as we did above, as properties of morphisms rather than objects, but this may be a matter of taste.

7.1 Examples of Partial Traces

(a) Finite Dimensional Vector Spaces

The category $\mathbf{Vec}_{fd}$ of finite dimensional vector spaces and linear transformations is a symmetric monoidal, indeed an additive, category (see [Mac98]), with monoidal product taken to be $\oplus$, the direct sum (biproduct). Hence, given $f : \oplus_I X_i \longrightarrow \oplus_J Y_j$ with $|I| = n$ and $|J| = m$, we can write f as an $m \times n$ matrix $f = [f_{ij}]$ of its components, where $f_{ij} : X_j \longrightarrow Y_i$ (notice the switch in the indices i and j).

We give a trace class structure on the category $(\mathbf{Vec}_{fd}, \oplus, \mathbf{0})$ as follows. We shall say an $f : X \oplus U \longrightarrow Y \oplus U$ is *trace class* iff $(I - f_{22})$ is invertible, where I is the identity matrix, and I and f_{22} have size $dim(U)$. In that case, we write

$$\mathtt{Tr}^U_{X,Y}(f) = f_{11} + f_{12}(I - f_{22})^{-1} f_{21} \tag{11}$$

This definition is motivated by a generalization of the fact that for a matrix A, $(I - A)^{-1} = \sum_i A^i$, whenever the infinite sum converges. Clearly this sum converges when the matrix norm of A is strictly less than 1, or when A is nilpotent, but in both cases the general idea is the desire to have $(I - A)$ invertible. If the infinite sum for $(I - f_{22})^{-1}$ exists, the above formula for

$\mathtt{Tr}^U_{X,Y}(f)$ becomes the usual "particle-style" trace in [Abr96,AHS02,HS04a]. One advantage of formula (11) is that it does not *a priori* assume the convergence of the sum, nor even that $(I - f_{22})^{-1}$ be computable by iterative methods.

Proposition 7.3 $(\mathbf{Vec}_{fd}, \oplus, \mathbf{0})$ *is partially traced, with trace class as above.*

The proof is sketched in [HS05a]. Proposition 7.3 uses the following standard facts from linear algebra:

Lemma 7.4 *Let* $M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}$ *be a partitioned matrix with blocks* A $(m \times m)$, B $(m \times n), C$ $(n \times m)$ *and* D $(n \times n)$. *If* D *is invertible, then* M *is invertible iff* $A - BD^{-1}C$ *(the* Schur Complement of D*) is invertible.*

Lemma 7.5 *Given* A $(m \times n)$ *and* B $(n \times m)$, $(I_m - AB)$ *is invertible iff* $(I_n - BA)$ *is invertible. Moreover* $(I_m - AB)^{-1}A = A(I_n - BA)^{-1}$.

Proof (Proposition 7.3). We shall verify a couple of axioms.

- **Naturality** in X and Y: Suppose $f \in \mathbb{T}^U_{X,Y}$ and $g : X' \longrightarrow X$ and $h : Y \longrightarrow Y'$, $(h \oplus 1_U)f(g \oplus 1_U)$ can be represented by its matrix $\begin{bmatrix} hf_{11}g & hf_{12} \\ f_{21}g & f_{22} \end{bmatrix}$ whose component from U to itself is f_{22} and hence $(h \oplus 1_U)f(g \oplus 1_U) \in \mathbb{T}^U_{X',Y'}$ and it is easy to see that $h\mathtt{Tr}^U_{X,Y}(f)g = \mathtt{Tr}^U_{X',Y'}((h \oplus 1_U)f(g \oplus 1_U))$.
- **Dinaturality** in U: Let $f : X \oplus U \longrightarrow Y \oplus U'$, $g : U' \longrightarrow U$. $(1_Y \oplus g)f \in \mathbb{T}^U_{X,Y}$ iff $I - gf_{22}$ is invertible iff $I - f_{22}g$ is invertible by Lemma 7.5 and thus iff $f(1_X \oplus g) \in \mathbb{T}^{U'}_{X,Y}$.

$$\begin{aligned} \mathtt{Tr}^U_{X,Y}((1_Y \oplus g)f) &= f_{11} + f_{12}(I - gf_{22})^{-1}gf_{21} \\ &= f_{11} + f_{12}g(I - f_{22}g)^{-1}f_{21} \text{ by Lemma 7.5.} \\ &= \mathtt{Tr}^{U'}_{X,Y}(f(1_X \oplus g)). \end{aligned}$$

□

As discussed in Remark 7.2, the category $(\mathbf{Vec}_{fd}, \oplus)$ is *not* partially traced in the sense of ABP.

(b) Metric Spaces

Consider the category **CMet** of complete metric spaces with non-expansive maps, that is $f : (M, d_M) \longrightarrow (N, d_N)$ such that $d_N(f(x), f(y)) \leq d_M(x, y)$, for all $x, y \in M$. Note that the tempting collection of complete metric spaces and contractions ($d_N(f(x), f(y)) < d_M(x, y)$) is not a category: there are no identity morphisms! **CMet** has products, namely given (M, d_M) and (N, d_N) we define $(M \times N, d_{M \times N})$ with $d_{M\times N}((m, n), (m', n')) = max\{d_M(m, m'), d_N(n, n')\}$.

We define the trace class structure on **CMet**(where $\otimes = \times$) as follows. We say that a morphism $f : X \times U \longrightarrow Y \times U$ is in $\mathbb{T}^U_{X,Y}$ iff for every $x \in X$ the induced map $\pi_2 \lambda u.f(x, u) : U \longrightarrow U$ has a unique fixed point; in other words, iff for every $x \in X$, there is a unique u, and a y, such that $f(x, u) = (y, u)$. Note that in this case y is necessarily unique. Also, note that contractions have unique fixed points, by the Banach fixed point theorem.

Suppose $f \in \mathbb{T}^U_{X,Y}$. We define $\mathtt{Tr}^U_{X,Y}(f) : X \longrightarrow Y$ by $\mathtt{Tr}^U_{X,Y}(f)(x) = y$, where $f(x, u) = (y, u)$ for the unique u. Equivalently, $\mathtt{Tr}^U_{X,Y}(f)(x) = \pi_1 f(x, u)$ where u is the unique fixed point of $\pi_2\lambda t.f(x, t)$.

Proposition 7.6 $(\mathbf{CMet}, \times, \{*\})$ *is a partially traced category with trace class as above.*

Lemma 7.7 *Let A and B be sets, $f : A \longrightarrow B$ and $g : B \longrightarrow A$. Then, gf has a unique fixed point if and only if fg does. Moreover, let $a \in A$ be the unique fixed point of $gf : A \longrightarrow A$ and $b \in B$ be the unique fixed point of $fg : B \longrightarrow B$. Then $f(a) = b$ and $g(b) = a$.*

Proof (Proposition 7.6). We shall verify the dinaturality axiom. For $f : X \times U \longrightarrow Y \times U$ and $x \in X$, we will use f_x to denote the map $\lambda u.f(x, u) : U \longrightarrow Y \times U$.

Dinaturality in U: Let $f : X \times U \longrightarrow Y \times U'$, $g : U' \longrightarrow U$. Note that for any $x \in X$, $\pi_2((1_Y \times g)f)_x = g(\pi_2 f_x)$ and $\pi_2(f(1_X \times g))_x = (\pi_2 f_x)g$ and $g(\pi_2 f_x)$ has a unique fixed point iff $(\pi_2 f_x)g$ has a unique fixed point, by Lemma 7.7. Thus $(1_Y \times g)f \in \mathbb{T}^U_{X,Y}$ iff $f(1_X \times g) \in \mathbb{T}^{U'}_{X,Y}$.

$$\begin{aligned}
\mathtt{Tr}^U_{X,Y}((1_Y \times g)f)(x) &= \pi_1(1 \times g)f(x, u) \;\; u \text{ is the unique} \\
&\quad\;\text{fixed point of } g(\pi_2 f_x) \\
&= \pi_1 f(x, u) \\
&= \pi_1 f(x, g(u')) \text{ by Lemma 7.7} \\
&\quad\;\text{where } u' \text{ is the unqiue fixed point of } (\pi_2 f_x)g \\
&= \mathtt{Tr}^{U'}_{X,Y}(f(1_X \times g))(x).
\end{aligned}$$

□

Proposition 7.6 remains valid for the category $(\mathbf{Set}, \times)$ of sets and mappings.

The latter then becomes a partially traced category with the same definition for trace class morphisms as in **CMet**. However, this fails for the category $(\mathbf{Rel}, \times)$, of sets and relations, as Lemma 7.7 is no longer valid: consider the sets $A = \{a\}$, $B = \{b, b'\}$, and let $f = \{(a, b), (a, b')\}$ and $g = \{(b, a), (b', a)\}$.

(c) Total Traces

Of course, all (totally-defined) traces in the usual definition of a traced monoidal category yield a trace class, namely the entire homset is the domain of $\mathtt{Tr}$.

Remark 7.8 (A Non-Example) Consider the structure $(\mathbf{CMet}, \times)$. Defining the trace class morphisms as those f such that $\pi_2 \lambda u.f(x, u) : U \longrightarrow U$ is a contraction, for every $x \in X$, does not yield a partially traced category: all axioms are true except for dinaturality and Vanishing II.

For details and motivation on the orthogonality relation we refer the interested reader to [HS05a]. See also the important work by Hyland and Schalk in [HylSc03] for the general definition of orthogonality relations in a symmetric monoidal closed category and its connections to models of linear logic.

Definition 7.9 Let $\mathcal{C}$ be a traced symmetric monoidal category. A *(strong) orthogonality relation* on $\mathcal{C}$ is a family of relations $\perp_{UV}$ between maps $u : V \longrightarrow U$ and $x : U \longrightarrow V$, denoted $V \xrightarrow{u} U \perp_{UV} U \xrightarrow{x} V$, subject to the following axioms:

(i) *Isomorphism* : Let $f : U \otimes V' \longrightarrow V \otimes U'$ and $\hat{f} : U' \otimes V \longrightarrow V' \otimes U$ be such that $\mathtt{Tr}^{V'}(\mathtt{Tr}^{U'}((1 \otimes 1 \otimes s_{U',V'})\alpha^{-1}(f \otimes \hat{f})\alpha)) = s_{U,V}$ and $\mathtt{Tr}^{V}(\mathtt{Tr}^{U}((1 \otimes 1 \otimes s_{U,V})\alpha^{-1}(\hat{f} \otimes f)\alpha)) = s_{U',V'}$. Here $\alpha = (1 \otimes 1 \otimes s)(1 \otimes s \otimes 1)$ with s at appropriate types. Note that this simply means that $f : (U, V) \longrightarrow (U', V')$ and $\hat{f} : (U', V') \longrightarrow (U, V)$ are inverses of each other in $\mathcal{G}(\mathcal{C})$ (the compact closure of $\mathcal{C}$, [Hag00,AHS02].)

Then, for all $u : V \longrightarrow U$ and $x : U \longrightarrow V$,

$$u \perp_{UV} x \;\text{ iff }\; \mathtt{Tr}^{U}_{V',U'}(s_{U,U'}(u \otimes 1_{U'})fs_{V',U}) \perp_{U'V'} \mathtt{Tr}^{V}_{U',V'}((1_{V'} \otimes x)\hat{f});$$

that is, orthogonality is invariant under isomorphism. This is so because the expressions above correspond to composition of u and f, and x and $\hat{f}$ in the compact closed category $\mathcal{G}(\mathcal{C})$.

(ii) *Precise Tensor:* For all $u : V \longrightarrow U$, $v : V' \longrightarrow U'$ and $h : U \otimes U' \longrightarrow V \otimes V'$,

$$(u \otimes v) \perp_{U \otimes U', V \otimes V'} h,$$

$$\text{iff}$$

$$v \perp_{U'V'} \mathtt{Tr}^{U}_{U',V'}(s_{U,V'}(u \otimes 1_{V'})hs_{U',U}) \quad \text{and} \quad u \perp_{UV} \mathtt{Tr}^{U'}_{U,V}((1 \otimes v)h)$$

(iii) *Identity* : For all $u : V \longrightarrow U$ and $x : U \longrightarrow V$, $u \perp_{UV} x$ implies $1_I \perp_{II} \mathtt{Tr}^{V}_{I,I}(xu)$.

(iv) *Symmetry* : For all $u : V \longrightarrow U$ and $x : U \longrightarrow V$, $u \perp_{UV} x$ iff $x \perp_{VU} u$.

Example 7.10 (Orthogonality as trace class) Let $(\mathcal{C}, \otimes, I, \mathtt{Tr})$ be a partially traced category where $\otimes$ is the monoidal product with unit I, and $\mathtt{Tr}$ is the partial trace operator as in above. Let A and B be objects of $\mathcal{C}$. For $f : A \longrightarrow B$ and $g : B \longrightarrow A$, we can define an orthogonality relation by declaring $f \perp_{BA} g$ iff $gf \in \mathbb{T}^{A}_{I,I}$. The axioms can be checked easily and we shall not include the verification here. It turns out that this is a variation of the notion of *Focussed orthogonality* of Hyland and Schalk [HylSc03].

Hence, from our previous discussion on traces, we obtain the following examples:

- $\mathbf{Vec}_{fd}$. For $A \in \vec{fd}$, $f, g \in End(A)$, define $f \perp g$ iff $I - gf$ is invertible. Here I is the identity matrix of size $dim(A)$.
- **CMet**. Let $M \in \mathbf{CMet}$. For $f, g \in End(M)$, define $f \perp g$ iff gf has a unique fixed point.

8 The Typed GoI Interpretation for MELL in *-Categories

The GoI interpretation we presented in Section 6 was carried out using a GoI Situation with the underlying category a traced UDC, and using an orthogonality relation, defined based on nilpotency. Moreover, formulas and proofs were interpreted based on a single reflexive object U. It is possible to extend this framework vastly beyond these limits, indeed it is possible to give a multi-object (typed) GoI (MGOI) interpretation for **MELL** using a GoI Situation with additional structure and a compatible abstract orthogonality relation. We shall briefly highlight what is involved without getting into details. Interested readers can refer to [Hag06].

For the purposes of this general version we shall need an additional structure on a monoidal category, namely that of contravariant functor $(\,)^*$. In the following we shall recall the definition of monoidal $*$-categories from [ABP99]. Nevertheless, note that our definition is different from that in [ABP99], as we do not require a conjugation functor, and we demand stronger conditions on the functor $(\,)^*$. Categories such as these with further structure on the homsets (W^*-categories) were first introduced in [GLR85]. The idea there was to generalize the notions and machinery of von Neumann algebras to a categorical setting. Later, similar categories (C^*-categories) were defined in [DopR89] and studied in depth. The motivation in this work was to present a new duality theory for compact groups, itself motivated by the work in early seventies on superselection structure in quantum field theory. Both [GLR85] and [DopR89] are excellent sources for examples of $*$-categories we define here.

Definition 8.1 A *symmetric monoidal $*$-category* $\mathcal{C}$ is a symmetric monoidal category with a strict symmetric monoidal functor $(\ )^* : \mathcal{C}^{op} \longrightarrow \mathcal{C}$ which is strictly involutive and the identity on objects. Note that this in particular implies that $(f \otimes g)^* = f^* \otimes g^*$, and $s^*_{A,B} = s_{B,A}$ where $s_{A,B}$ is the symmetry morphism.

We say that a morphism $f : A \longrightarrow A$ is *Hermitian* if $f^* = f$. A morphism $f : A \longrightarrow B$ is called a *partial isometry* if $f^*ff^* = f^*$ or equivalently, if $ff^*f = f$. A morphism $f : A \longrightarrow A$ is called a *partial symmetry* if it is Hermitian and a partial isometry. That is, if $f^* = f$ and $f^3 = f$. Note that there is no underlying Hilbert space structure on the homsets of $\mathcal{C}$; the terminology here is borrowed from operator algebras to account for the similar properties of such morphisms, which can be expressed in the more general setting of $*$-categories.

An obvious example is the category $\mathbf{Hilb}_\otimes$ of Hilbert spaces and bounded linear maps with tensor product of Hilbert spaces as the monoidal product. Given $f : H \longrightarrow K$, $f^* : K \longrightarrow H$ is given by the adjoint of f, defined uniquely by $\langle f(x), y\rangle = \langle x, f^*(y)\rangle$. It is not hard to see that all the required properties are satisfied. Note that the category $\mathbf{Hilb}_\oplus$ of Hilbert spaces and bounded linear maps but with direct sum as the monoidal product is a $*$-category too, with the same definition for the $(\ )^*$ functor.

Another example is the category $\mathbf{Rel}_\times$ of sets and relations with the cartesian product of sets as the monoidal product. Given $f : X \longrightarrow Y$, $f^* = \overline{f}$ where $\overline{f}$ is the converse relation. Again, note that the category $\mathbf{Rel}_\oplus$ of sets and relations with monoidal product, the disjoint union (categorical biproduct) is a monoidal $*$-category too, with the same definition for the $(\,)^*$ functor.

Yet another example that shows up frequently in the context of GoI is the category $\mathbf{PInj}_\uplus$ of sets and partial injective maps, with disjoint union as the monoidal product. Given $f : X \longrightarrow Y$, $f^* = f^{-1}$.

Other examples include $\mathbf{Hilb}_{fd}$ of finite dimensional Hilbert spaces and bounded linear maps, $\mathbf{URep}(G)$, finite representations of a compact group G, etc. For more details, examples and the ways that such categories show up in logic and computer science, see [ABP99].

Definition 8.2 A *GoI category* is a triple $(\mathcal{C}, T, \perp)$ where $\mathcal{C}$ is a partially traced $*$-category as in Section 7, $T = (T, \psi, \psi_I) : \mathcal{C} \longrightarrow \mathcal{C}$ is a traced symmetric monoidal functor, that is if $f \in \mathbb{T}^U_{X,Y}$, then $\psi^{-1}_{Y,U}T(f)\psi_{X,U} \in \mathbb{T}^{TU}_{TX,TY}$ and $\mathtt{Tr}^{TU}_{TX,TY}(\psi^{-1}_{Y,U}T(f)\psi_{X,U}) = T(\mathtt{Tr}^U_{X,Y}(f))$. Here $\perp$ is an orthogonality relation on $\mathcal{C}$ as in the above. Furthermore, we require that,

- The following natural retractions exist:

(i) $\mathcal{K}_I \lhd T \ (w, w^*)$, $\quad \mathcal{K}_I$ denotes the constant I functor.
(ii) $Id \lhd T \ (d, d^*)$
(iii) $T^2 \lhd T \ (e, e^*)$
(iv) $T \otimes T \lhd T \ (c, c^*)$

• The orthogonality relation must be *GoI compatible*, that is, it must satisfy the following additional axioms:

(c1) For all $f : V \longrightarrow U$, $g : U \longrightarrow V$,

$$f \perp_{U,V} g \ \text{ implies } \ d_U f d_V^* \perp_{TU,TV} Tg.$$

(c2) For all $f : U \longrightarrow U$ and $g : I \longrightarrow I$,

$$w_U g w_U^* \perp_{TU,TU} Tf.$$

(c3) For all $f : TV \otimes TV \longrightarrow TU \otimes TU$ and $g : U \longrightarrow V$,

$$f \perp_{TU \otimes TU, TV \otimes TV} \ \ Tg \otimes Tg \ \text{ implies } \ c_U f c_V^* \perp_{TU,TV} Tg.$$

• The functor T commutes with $(\ )^*$, that is $(T(f))^* = T(f^*)$. Moreover, $\psi^* = \psi^{-1}$ and $\psi_I^* = \psi_I^{-1}$.

Proposition 8.3 *Suppose $\mathcal{C}$ is a partially traced $*$-category that is in addition equipped with an endofunctor T and monoidal retractions as in Definition 8.2. Then, the orthogonality relation $\perp$ defined as in Example 7.10 is GoI compatible.*

Proof. We shall verify the compatibility axioms of Definition 8.2.

(c1) $\mathtt{Tr}^{TV}(T(g) d_U f d_V^*) = \mathtt{Tr}^{TV}(d_V g f d_V^*) = \mathtt{Tr}^{V}(gf)$.

(c2) $\mathtt{Tr}^{TU}(T(f) w_U g w_U^*) = \mathtt{Tr}^{TU}(w_U g w_U^*) = \mathtt{Tr}^{I}(g)$.
Recall that $\mathbb{T}^I_{I,I} = \mathcal{C}(I, I)$.

(c3) $\mathtt{Tr}^{TV}(T(g) c_U f c_V^*) = \mathtt{Tr}^{TV}(c_V (Tg \otimes Tg) f c_V^*) = \mathtt{Tr}^{TV \otimes TV}((Tg \otimes Tg) f)$. □

GoI categories are the main mathematical structures in our semantic interpretation in the following section. Here are a few examples of GoI categories.

Examples 8.4 (a) $(\mathbf{PInj}_{\uplus}, T, \perp)$

We define, $f \perp g$ iff gf is nilpotent. It can be easily checked that this definition satisfies the axioms for an orthogonality relation.

Let us verify the compatibility axioms:

- For $f : V \longrightarrow U$ and $g : U \longrightarrow V$, suppose gf is nilpotent, say $(gf)^n = 0$, then $(T(g)d_U f d_V^*)^n = (d_V gf d_V^*)^n$ by naturality of d_U, but as $d_V^* d_V = 1_V$ we have $(d_V gf d_V^*)^n = d_V (gf)^n d_V^* = 0$.
- As $I = \emptyset$ and $w_I = 0$, we have that $T(f)w_U g w_U^*$ is nilpotent.
- For $f : TV \otimes TV \longrightarrow TU \otimes TU$ and $g : U \longrightarrow V$, suppose $(Tg \otimes Tg)f$ is nilpotent, say $((Tg \otimes Tg)f)^n = 0$, Then $(T(g)c_U f c_V^*)^n = (c_V (Tg \otimes Tg) f c_V^*)^n$, by naturality of c_V, but as $c_V^* c_V = 1_{TV \otimes TV}$ we have $(c_V (Tg \otimes Tg) f c_V^*)^n = c_V ((Tg \otimes Tg)f)^n c_V^* = 0$.

Finally, for any $f : X \longrightarrow Y$, $(Tf)^* = T(f^*)$.

(b) $(\mathbf{Hilb}_\oplus, T, \perp)$, where **Hilb** is the category of Hilbert spaces and bounded linear maps. The monoidal product is the direct sum of Hilbert spaces. We have seen above that $\mathbf{Hilb}_\oplus$ is a partially traced $*$-category.

$T(H) = \ell^2 \otimes H$ where ℓ^2 is the space of square summable sequences.

We define, $f \perp g$ iff $(1 - gf)$ is an invertible linear transformation. Compatibility follows from Proposition 8.3, because for $f : H \longrightarrow K$, $g : K \longrightarrow H$, $f \perp g$ iff $gf \in \mathbb{T}^H$.

Finally, as $\mathbf{Hilb}_\otimes$ is also a $*$-category with f^* the adjoint of f, we have that for any $f : H \longrightarrow K$, $(Tf)^* = T(f^*)$.

(c) $(\mathbf{Rel}_\oplus, T, \perp)$ is a GoI-category with the same definitions for T and $\perp$ as in the case of **PInj**. Note that disjoint union, denoted $\oplus$, is in fact the categorical biproduct in **Rel**.

Multiobject Geometry of Interaction (MGoI) was introduced in [HS05a] and was used to interpret **MLL** without units. It was later extended to exponentials in [Hag06]. The main idea in [HS05a] was to keep the types of the formulas that were defined by a denotational semantics map during the GoI interpretation. For the multiplicative case this also implied that, in contrast to the usual GoI, there was no need for a reflexive object U and this made the interpretation possible in categories like finite dimensional vector spaces. On the other hand, in the case of exponentials, we soon observe that infinity forces itself into the framework: it is no longer possible to carry out the MGoI interpretation in finite dimensions. This transition to infinity occurs, for example when we are forced to admit a retraction $TTA \lhd TA$ for any object A in the relevant category. Note that, although in this way reflexive objects reappear, they are not used to collapse types as in the GoI interpretation using a single object U.

8.1 MGoI Interpretation of formulas

Given a GoI category $(\mathcal{C}, T, \perp)$, let A be an object of $\mathcal{C}$ and let $f, g \in End(A)$. We say that f is *orthogonal to* g, denoted $f \perp g$, if $(f, g) \in \perp$. Also given $X \subseteq End(A)$ we define

$$X^{\perp} = \{f \in End(A) \mid \forall g \in X, f \perp g\}.$$

We now define an operator on the objects of $\mathcal{C}$ as follows: Given an object A, $\mathcal{T}(A) = \{X \subseteq End(A) \mid X^{\perp\perp} = X\}$. We shall also need the notion of a denotational interpretation for formulas. We define an interpretation map $[\![-]\!]$ on the formulas of **MELL** as follows. Given the value of $[\![-]\!]$ on the atomic propositions as objects of $\mathcal{C}$, we extend it to all formulas by:

- $[\![A^{\perp}]\!] = [\![A]\!]$
- $[\![A \parr B]\!] = [\![A \otimes B]\!] = [\![A]\!] \otimes [\![B]\!]$.
- $[\![!A]\!] = [\![?A]\!] = T[\![A]\!]$.

The MGoI-interpretation for formulas is defined as follows.

- $\theta(\alpha) \in \mathcal{T}([\![\alpha]\!])$, where α is an atomic formula.
- $\theta(\alpha^{\perp}) = \theta(\alpha)^{\perp}$, where α is an atomic formula.
- $\theta(A \otimes B) = \{a \otimes b \mid a \in \theta(A), b \in \theta(B)\}^{\perp\perp}$
- $\theta(A \parr B) = \{a \otimes b \mid a \in \theta(A)^{\perp}, b \in \theta(B)^{\perp}\}^{\perp}$
- $\theta(!A) = \{Ta \mid a \in \theta(A)\}^{\perp\perp}$
- $\theta(?A) = \{Ta \mid a \in \theta(A^{\perp})\}^{\perp}$

Easy consequences of the definition are: (i) for any formula A, $(\theta A)^{\perp} = \theta A^{\perp}$, (ii) $\theta(A) \subseteq End([\![A]\!])$, and (iii) $\theta(A)^{\perp\perp} = \theta(A)$.

8.2 MGoI Interpretation of proofs

In this section we define the MGoI interpretation for proofs of **MELL** without units. All references from now on refer to this MGoI interpretation unless stated otherwise.

As before, every **MELL** sequent will be of the form $\vdash [\Delta], \Gamma$ where Γ is a sequence of formulas and Δ is a sequence of cut formulas that have already been made in the proof of $\vdash \Gamma$. This device is used to keep track of the cuts in a proof of $\vdash \Gamma$. A proof π of $\vdash [\Delta], \Gamma$ is represented by a morphism $\llbracket \pi \rrbracket \in End(\otimes \llbracket \Gamma \rrbracket \otimes \llbracket \overline{\Delta} \rrbracket)$. With $\Gamma = A_1, \cdots, A_n$, $\otimes \llbracket \Gamma \rrbracket$ stands for $\llbracket A_1 \rrbracket \otimes \cdots \otimes \llbracket A_n \rrbracket$, and with $\Delta = B_1, B_1^\perp, \cdots B_m, B_m^\perp$, $\llbracket \overline{\Delta} \rrbracket = T^k(\llbracket B_1 \rrbracket \otimes \cdots \otimes \llbracket B_m \rrbracket^\perp)$, for some non-negative integer k, with T^0 being the identity functor. We drop the double brackets wherever there is no danger of confusion. We also define $\sigma = s \otimes \cdots \otimes s$ (m-copies) where s is the symmetry map at different types (omitted for convenience), and $|\Delta| = 2m$. The morphism σ represents the cuts in the proof of $\vdash \Gamma$, i.e. it models Δ. In the case where Δ is empty (that is for a cut-free proof), we define $\sigma : I \longrightarrow I$ to be 1_I where I is the unit of the monoidal product in $\mathcal{C}$.

Definition 8.5 (The MGoI Interpretation) Let π be a proof of $\vdash [\Delta], \Gamma$. We define *the MGoI interpretation of* π, denoted by $\llbracket \pi \rrbracket$, by induction on the length of the proof as follows.

(1) π is an *axiom* $\vdash A, A^\perp$, $\llbracket \pi \rrbracket := s_{V,V}$ where $\llbracket A \rrbracket = \llbracket A^\perp \rrbracket = V$.

(2) π is obtained using the *cut* rule on π' and π'' that is,

$$\cfrac{\begin{array}{cc}\pi' & \pi'' \\ \vdots & \vdots \\ \vdash [\Delta'], \Gamma', A & \vdash [\Delta''], A^\perp, \Gamma''\end{array}}{\vdash [\Delta', \Delta'', A, A^\perp], \Gamma', \Gamma''}\ cut$$

Define $\llbracket \pi \rrbracket = \tau^{-1}(\llbracket \pi' \rrbracket \otimes \llbracket \pi'' \rrbracket)\tau$, where τ is the permutation $\Gamma' \otimes \Gamma'' \otimes \overline{\Delta'} \otimes \overline{\Delta''} \otimes A \otimes A^\perp \xrightarrow{\tau} \Gamma' \otimes A \otimes \overline{\Delta'} \otimes A^\perp \otimes \Gamma'' \otimes \overline{\Delta''}$.
(double brackets and $\otimes$ are dropped for the sake of readability).

(3) π is obtained using the *exchange* rule on the formulas A_i and A_{i+1} in Γ'. That is π is of the form

$$\cfrac{\begin{array}{c}\pi' \\ \vdots \\ \vdash [\Delta], \Gamma'\end{array}}{\vdash [\Delta], \Gamma}\ exchange$$

where $\Gamma' = \Gamma'_1, A_i, A_{i+1}, \Gamma'_2$ and $\Gamma = \Gamma'_1, A_{i+1}, A_i, \Gamma'_2$. Then, $\llbracket \pi \rrbracket$ is obtained from $\llbracket \pi' \rrbracket$ by interchanging the rows i and $i+1$. So, $\llbracket \pi \rrbracket = \tau^{-1} \llbracket \pi' \rrbracket \tau$, where $\tau = 1_{\Gamma'_1} \otimes s \otimes 1_{\Gamma'_2 \otimes \overline{\Delta}}$.

(4) π is obtained using an application of the *par* rule, that is π is of the form:

$$\cfrac{\begin{array}{c}\pi' \\ \vdots \\ \vdash [\Delta], \Gamma', A, B\end{array}}{\vdash [\Delta], \Gamma', A \parr B}\ \parr \qquad . \text{ Then } \llbracket \pi \rrbracket = \llbracket \pi' \rrbracket.$$

(5) π is obtained using an application of the *times* rule, that is π is of the form:

$$\begin{array}{c}
\begin{array}{cc} \pi' & \pi'' \\ \vdots & \vdots \end{array} \\
\dfrac{\vdash [\Delta'], \Gamma', A \quad \vdash [\Delta''], \Gamma'', B}{\vdash [\Delta', \Delta''], \Gamma', \Gamma'', A \otimes B}\ \otimes
\end{array}$$

Then $[\![\, \pi \,]\!] = \tau^{-1}([\![\, \pi' \,]\!] \otimes [\![\, \pi'' \,]\!])\tau$, where τ is the permutation $\Gamma' \otimes \Gamma'' \otimes A \otimes B \otimes \overline{\Delta'} \otimes \overline{\Delta''} \xrightarrow{\tau} \Gamma' \otimes A \otimes \overline{\Delta'} \otimes \Gamma'' \otimes B \otimes \overline{\Delta''}$.

(6) π is obtained from π' by an *of course* rule, that is π has the form :

$$\begin{array}{c}
\pi' \\ \vdots \\
\dfrac{\vdash [\Delta], ?\Gamma', A}{\vdash [\Delta], ?\Gamma', !A}\ \text{of course}
\end{array}$$

Then $[\![\, \pi \,]\!] = (e_{\Gamma'} \otimes 1_{TA} \otimes 1_{\overline{\Delta}})\varphi^{-1}T([\![\, \pi' \,]\!])\varphi(e^*_{\Gamma'} \otimes 1_{TA} \otimes 1_{\overline{\Delta}})$, where $TT \lhd T\,(e, e^*)$, with $\Gamma' = A_1, \cdots, A_n$, $e_{\Gamma'} = e_{A_1} \otimes \cdots \otimes e_{A_n}$, similarly for e^*, and φ is the canonical isomorphism: The isomorphism $\varphi : T^2(\Gamma') \otimes TA \otimes T(\overline{\Delta}) \longrightarrow T(T(\Gamma') \otimes A \otimes \overline{\Delta})$ is defined using the isomorphism $\psi_{X,Y} : TX \times TY \longrightarrow T(X \otimes Y)$. With $\Gamma' = A_1, \cdots, A_n$, $T(\Gamma')$ is a short hand for $TA_1 \otimes \cdots \otimes TA_n$, similarly for $T(\overline{\Delta})$.

(7) π is obtained from π' by the *dereliction* rule, that is, π is of the form :

$$\begin{array}{c}
\pi' \\ \vdots \\
\dfrac{\vdash [\Delta], \Gamma', A}{\vdash [\Delta], \Gamma', ?A}\ \text{dereliction}
\end{array}$$

Then $[\![\, \pi \,]\!] = (1_{\Gamma'} \otimes d_A \otimes 1_{\overline{\Delta}})[\![\, \pi' \,]\!](1_{\Gamma'} \otimes d^*_A \otimes 1_{\overline{\Delta}})$ where $Id \lhd T\,(d, d^*)$.

(8) π is obtained from π' by the *weakening* rule, that is, π is of the form:

$$\begin{array}{c}
\pi' \\ \vdots \\
\dfrac{\vdash [\Delta], \Gamma'}{\vdash [\Delta], \Gamma', ?A}\ \text{weakening}
\end{array}$$

Then $[\![\, \pi \,]\!] = (1_{\Gamma'} \otimes w_A \otimes 1_{\overline{\Delta}})[\![\, \pi' \,]\!](1_{\Gamma'} \otimes w^*_A \otimes 1_{\overline{\Delta}})$, where $\mathcal{K}_I \lhd T\,(w, w^*)$.

(9) π is obtained from π' by the *contraction* rule, that is, π is of the form :

$$\begin{array}{c} \pi' \\ \vdots \\ \dfrac{\vdash [\Delta], \Gamma', ?A, ?A}{\vdash [\Delta], \Gamma', ?A}\ \text{contraction} \end{array}$$

Then $\llbracket \pi \rrbracket = (1_{\Gamma'} \otimes c_A \otimes 1_{\overline{\Delta}}) \llbracket \pi' \rrbracket (1_{\Gamma'} \otimes c^*_A \otimes 1_{\overline{\Delta}})$, where $T \otimes T \lhd T\, (c, c^*)$.

Examples 8.6 (a) Let π be the following proof:

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

Then the MGoI interpretation of this proof is given by $\llbracket \pi \rrbracket = \tau^{-1}(s \otimes s)\tau = s_{V \otimes V, V \otimes V}$ where $\tau = (1 \otimes 1 \otimes s)(1 \otimes s \otimes 1)$ and $\llbracket A \rrbracket = \llbracket A^{\perp} \rrbracket = V$.

(b) Now consider the following proof

$$\dfrac{\dfrac{\dfrac{\vdash A, A^{\perp}}{\vdash A, ?A^{\perp}}}{\vdash !A, ?A^{\perp}} \quad \vdash B, B^{\perp}}{\vdash !A \otimes B, ?A^{\perp} \parr B^{\perp}}$$

Given $\llbracket A \rrbracket = V$ and $\llbracket B \rrbracket = W$, we have $\llbracket \pi \rrbracket = (1 \otimes s \otimes 1)(1 \otimes e \otimes 1 \otimes 1)(\psi^{-1}T(h)\psi \otimes s)(1 \otimes e^* \otimes 1 \otimes 1)(1 \otimes s \otimes 1)$ where $h = (1 \otimes d_V)s(1 \otimes d^*_V)$.

Proposition 8.7 *Let π be an* **MELL** *proof of $\vdash [\Delta], \Gamma$. Then $\llbracket \pi \rrbracket$ is a partial symmetry.*

Proof. Proof follows by induction on the length of the proofs, noting that the functor $(\ )^*$ is a strict symmetric monoidal functor, $T(f)^* = T(f^*)$, $\psi^* = \psi^{-1}$, and $\psi^*_I = \psi^{-1}_I$. □

8.3 Interpretation of cut-elimination

As we saw previously, the mathematical model of cut-elimination is given by the *execution formula* as in (8), defined as follows:

$$EX(\llbracket \pi \rrbracket, \sigma) = \mathtt{Tr}^{\otimes\Delta}_{\otimes\Gamma, \otimes\Gamma}((1 \otimes \sigma)\llbracket \pi \rrbracket)$$

where π is a proof of the sequent $\vdash [\Delta], \Gamma$, and $\sigma = s^{\otimes m}$ models Δ, where $|\Delta| = 2m$. Note that $EX(\llbracket \pi \rrbracket, \sigma)$ is a morphism from $\otimes\Gamma \longrightarrow \otimes\Gamma$, when it exists. We shall prove below (see Theorem 8.12) that the execution formula always exists for any **MELL** proof π.

Example 8.8 Consider the proof π in Example 6.6 above. Recall also that $\sigma = s$ in this case ($m = 1$). Then $EX(\llbracket \pi \rrbracket, \sigma) = \mathtt{Tr}((1 \otimes s_{V,V}) s_{V \otimes V, V \otimes V}) = s_{V,V}$.

8.4 Soundness of the Interpretation

In this section we discuss the soundness of the MGoI interpretation. We show that if a proof π is reduced (via cut-elimination) to another proof π', then $EX(\llbracket \pi \rrbracket, \sigma) = EX(\llbracket \pi' \rrbracket, \tau)$; that is, $EX(\llbracket \pi \rrbracket, \sigma)$ is an invariant of reduction. In particular, if π' is cut-free (i.e. a normal form) we have $EX(\llbracket \pi \rrbracket, \sigma) = EX(\llbracket \pi' \rrbracket, 1_I) = \llbracket \pi' \rrbracket$.

We shall not give the proof of the soundness here, but will mention the main lemmas used in this proof.

Lemma 8.9 (Associativity of cut) *Let π be a proof of $\vdash [\Gamma, \Delta], \Lambda$ and σ and τ be the morphisms representing the cut-formulas in Γ and Δ respectively. Then*

$$EX(\llbracket \pi \rrbracket, \sigma \otimes \tau) = EX(EX(\llbracket \pi \rrbracket, \tau), \sigma) = EX(EX((1 \otimes s) \llbracket \pi \rrbracket (1 \otimes s), \sigma), \tau),$$

whenever all traces exist.

Definition 8.10 Let $\Gamma = A_1, \cdots, A_n$ and $V_i = \llbracket A_i \rrbracket$.

- A *datum of type* $\theta\Gamma$ is a morphism $M : \otimes_i V_i \longrightarrow \otimes_i V_i$ such that for any $a_i \in \theta(A_i^{\perp})$, $\otimes_i a_i \perp M$ and
$$M{\bullet}a_1 := \mathtt{Tr}^{V_1}(s^{-1}_{\otimes_{i \neq 1} V_i, V_1}(a_1 \otimes 1_{V_2} \otimes \cdots \otimes 1_{V_n}) M s_{\otimes_{i \neq 1} V_i, V_1})$$
and
$$M\hat{\bullet}(a_2 \otimes \cdots \otimes a_n) := \mathtt{Tr}^{V_2 \otimes \cdots \otimes V_n}((1 \otimes a_2 \otimes \cdots \otimes a_n) M)$$
both exist.
- An *algorithm of type* $\theta\Gamma$ is a morphism $M : \otimes_i V_i \otimes \llbracket \Delta \rrbracket \longrightarrow \otimes_i V_i \otimes \llbracket \Delta \rrbracket$ for some $\Delta = B_1, B_2, \cdots, B_{2m}$ with m a nonnegative integer and $B_{i+1} = B_i^{\perp}$ for $i = 1, 3, \cdots, 2m-1$, such that if $\sigma : \otimes_{i=1}^{2m} \llbracket B_i \rrbracket \longrightarrow \otimes_{i=1}^{2m} \llbracket B_i \rrbracket$ is $\otimes_{i=1,odd}^{2m-1} s_{\llbracket B_i \rrbracket, \llbracket B_{i+1} \rrbracket}$, $EX(M, \sigma)$ exists and is a datum of type $\theta\Gamma$.

Lemma 8.11 *Let $\widetilde{\Gamma} = A_2, \cdots, A_n$ and $\Gamma = A_1, \widetilde{\Gamma}$. Let $V_i = \llbracket A_i \rrbracket$, and $M : \otimes_i V_i \longrightarrow \otimes_i V_i$, for $i = 1, \cdots, n$. Then, M is a datum of type $\theta(\Gamma)$ iff for all $a_i \in \theta(A_i^{\perp})$, $M{\bullet}a_1$ and $M\hat{\bullet}(a_2 \otimes \cdots \otimes a_n)$ (defined as above) exist and are in $\theta(\widetilde{\Gamma})$, and $\theta(A_1)$, respectively.*

Theorem 8.12 (Proofs as algorithms) *Let π be an* **MELL** *proof of a sequent $\vdash [\Delta], \Gamma$. Then $\llbracket \pi \rrbracket$ is an algorithm of type $\theta\Gamma$.*

Corollary 8.13 (Existence of Dynamics) *Let π be an* **MELL** *proof of a sequent $\vdash [\Delta], \Gamma$. Then $EX(\llbracket \pi \rrbracket, \sigma)$ exists.*

Theorem 8.14 (EX is an invariant) *Let π be an MELL proof of a sequent $\vdash [\Delta], \Gamma$ such that $?A$ does not occur in Γ for any formula A. Then,*

- *If π reduces to π' by any sequence of cut-elimination steps, then $EX(\llbracket \pi \rrbracket, \sigma) = EX(\llbracket \pi' \rrbracket, \tau)$. So $EX(\llbracket \pi \rrbracket, \sigma)$ is an invariant of reduction.*
- *In particular, if π' is any cut-free proof obtained from π by cut-elimination, then $EX(\llbracket \pi \rrbracket, \sigma) = EX(\llbracket \pi' \rrbracket, 1_I) = \llbracket \pi' \rrbracket$.*

9 Concluding Remarks

We have mentioned several open questions in the tutorial, and the reader will be able to find many interesting questions in following up the literature in the Bibliography. Still, a few questions seem particularly apt.

(i) The GoI interpretation does not seem to deal well with units in **LL**. Thus, one should formulate GoI taking into account **-autonomous categories without units*. One such study is in R. Houston's thesis [Hou07].

(ii) The question of how to take into account the additives of **LL** in GoI and the associated categorical analysis of [Gi95a], both along the style here, as well as in the style of [AHS02], is still open.

(iii) Finding examples of our GoI situations in von Neumann algebras, and categorically analyzing Girard's recent notions of GoI [Gi07,Gi08] is a challenge, and presumably would need to accommodate categorical versions of Polarized Linear Logics, as in [HamSc07].

References

[Abr93] S. Abramsky, Computational Interpretations of Linear Logic, *Theoretical Computer Science***111**, 1993, 3-57.

[Abr96] S. Abramsky (1996), Retracing Some Paths in Process Algebra. In *CONCUR 96, Springer LNCS* **1119**, 1-17.

[Abr05] S. Abramsky, Abstract Scalars, Loops, and Free Traced and Strongly Compact Closed Categories, in: CALCO 2005, Vol. 3629, *Springer Lecture Notes in Computer Science*, 2005, 1-31.

[Abr07] S. Abramsky, Temperley-Lieb algebra: from knot theory to logic and computation via quantum mechanics In *Mathematics of Quantum Computing and Technology*, Goong Chen, Louis Kauffman and Sam Lomonaco, eds. Taylor and Francis, 2007, 515–558 .

[ABP99] S. Abramsky, R. Blute, and P. Panangaden (1999), Nuclear and trace ideals in tensored *-categories, *J. Pure and Applied Algebra* vol. 143, 3–47.

[AbCo04] S. Abramsky and B. Coecke, A categorical semantics of quantum protocols, in *Proc. 19th Annual IEEE Symposium on Logic in Computer Science (LICS)* , IEEE Computer Science Press, 415-425, 2004.

[AHS02] S. Abramsky, E. Haghverdi, and P. J. Scott (2002), Geometry of Interaction and Linear Combinatory Algebras. *MSCS*, vol. 12(5), 2002, 625-665, CUP.

[AJ94a] S. Abramsky and R. Jagadeesan, New Foundations for the Geometry of Interaction. *Information and Computation* **111** (1), 53-119 (1994).

[Bail95] P. Baillot (1995), Abramsky-Jagadeesan-Malacaria strategies and the geometry of interaction, mémoire de DEA, Universite Paris 7, 1995.

[BP01] P. Baillot and M. Pedicini, Elementary complexity and geometry of interaction, *Fundamenta Informaticae,* vol. 45, no. 1-2, 2001

[Barr79] M. Barr, $*$-*Autonomous Categories*, *Springer Lecture Notes in Mathematics* **752** , 1979.

[Barr92] Barr, M. (1992), Algebraically Compact Functors. *JPAA*, Vol. **82**, 211-231.

[BE93] Stephen L. Bloom and Zoltan Esik. *Iteration theories: equational logic of iterative processes.* EATCS monographs on theoretical computer science, Springer-Verlag, 1993.

[Bl96] R. Blute, Hopf algebras and linear logic, Mathematical Structures in Computer Science 6, pp. 189-212, (1996).

[BCST96] R. Blute, J. R. B. Cockett, R. A. G. Seely and T. Trimble. Natural deduction and coherence for weakly distributive categories. Journal of Pure and Applied Algebra 13, pp. 229–296, (1996)

[BCS96] R. Blute, J. R. B. Cockett, R. A. G. Seely. ! and ?: Storage as tensorial strength. Mathematical structures in Computer Science 6, pp. 313-351, (1996).

[BCS00] R. Blute, J. R. B. Cockett, R. A. G. Seely (2000), Feedback for linearly distributive categories: traces and fixpoints, Bill (Lawvere) Fest, *Journal of Pure and Applied Algebra*, vol. 154, pp 27-69.

[BS96] R. Blute, P. Scott. Linear Lauchli semantics, Annals of Pure and Applied Logic 77, pp. 101-142 (1996).

[BS04] R. Blute, P. Scott. Category Theory for Linear Logicians, in *Linear Logic in Computer Science*, Camb. U. Press, 2004, 3–64

[Bor93] F. Borceux. *Handbook of Categorical Algebra*, Cambridge University Press, (1993).

[CS97] J. R. B. Cockett, R. A. G. Seely. Weakly distributive categories. Journal of Pure and Applied Algebra 114, pp. 133-173, (1997).

[Dan90] V. Danos (1990), *La logique linéaire appliquée à l'étude de divers processus de normalisation et principalement du λ-calcul.* PhD thesis, Université Paris VII.

[DR95] V. Danos and L. Regnier (1995), Proof-nets and the Hilbert Space. In: *Advances in Linear Logic*, London Math. Soc. Notes, **222**, CUP, 307–328.

[DopR89] Doplicher, S. and Roberts, J.E. (1989), A New Duality for Compact Groups. *Invent. Math.* **98**, pp. 157-218.

[FP07] C. Fuhrman and D. Pym, On categorical Models of classical logic and the Geometry of Interaction, *Math. Structures in Comp. Science* (2007), Cambridge, 957–1027

[Ger85] R. Geroch. *Mathematical Physics*, University of Chicago Press. 1985

[GLR85] Ghez, P., Lima, R. and Roberts, J.E. (1985), W^*-categories. *Pacific Journal of Math.* **120**, pp. 79-109.

[Gi87] J.-Y. Girard Linear Logic. *Theoretical Computer Science* **50** (1) (1987), 1-102.

[Gi88] J.-Y. Girard (1988), Geometry of Interaction II: Deadlock-free Algorithms. In *Proc. of COLOG'88, LNCS* **417**, Springer, 76–93.

[Gi89] J-Y. Girard, Towards a Geometry of Interaction, in: *Categories in Computer Science and Logic*, ed. by J.W. Gray and A. Scedrov, *Contemp. Math*, **92**, AMS , 1989, pp. 69-108.

[Gi89a] J.-Y. Girard (1989a), Geometry of Interaction I: Interpretation of System F. In *Proc. Logic Colloquium 88,* North Holland, 221–260.

[Gi95a] J.-Y. Girard (1995), Geometry of Interaction III: Accommodating the Additives. In: *Advances in Linear Logic,* LNS **222**,CUP, 329–389,

[Gi07] J.-Y. Girard (2007). Le Point Aveugle I, II, Hermann Éditeurs, Paris , 567 + pp.

[Gi08] J.-Y. Girard. Geometry of Interaction V: logic in the hyperfinite factor, manuscript, 2008.

[GLT] J.-Y. Girard, Y. Lafont, P.Taylor. *Proofs and Types*, Cambridge Tracts in Theoretical Computer Science 7, 1989.

[GSS] Girard, J.-Y., A. Scedrov, and P.J. Scott Bounded linear logic, *Theoretical Comp. Science***97**, 1992, pp. 1–66.

[GAL92] G. Gonthier, M. Abadi, and J.-J. Lévy (1992), The geometry of optimal lambda reduction. In Proceedings of Logic in Computer Science, vol. 9 pp. 15-26.

[Hag00] E. Haghverdi, *A Categorical Approach to Linear Logic, Geometry of Proofs and Full Completeness*, PhD Thesis, University of Ottawa, Canada 2000.

[Hag00a] E. Haghverdi, Unique Decomposition Categories, Geometry of Interaction and combinatory logic, *Math. Struct. in Comp. Science*, vol. **10**, 2000, 205-231.

[HS04a] E. Haghverdi and P.J.Scott, A categorical model for the Geometry of Interaction, *Theoretical Computer Science* Volume 350, Issues 2-3 , Feb 2006, pp. 252-274. (Preliminary Version in: in *Automata , Languages, Programming*(ICALP 2004), Springer LNCS 3142, pp. 708-720).

[HS04b] E. Haghverdi and P.J.Scott, From Geometry of Interaction to Denotational Semantics. Proceedings of CTCS2004. In ENTCS, vol. 122, pp. 67-87. Elsevier.

[HS05a] E. Haghverdi and P.J.Scott, Towards a Typed Geometry of Interaction, CSL2005 (Computer Science Logic), Luke Ong, Ed. SLNCS 3634, pp. 216-231.

[Hag06] E. Haghverdi, Typed GoI for Exponentials. in: M. Bugliesi et al. (Eds.): Proc. of ICALP 2006, Part II, LNCS 4052, pp. 384-395, 2006. Springer Verlag.

[HamSc07] M. Hamano, P. Scott, A categorical semantics for polarized MALL, *Ann. Pure & Applied Logic*, 145 (2007),276-313

[Has97] Hasegawa, M. (1997), Recursion from Cyclic Sharing : Traced Monoidal Categories and Models of Cyclic Lambda Calculus, *Springer LNCS* **1210**, 196-213.

[Has08] Hasegawa, M. On Traced Monoidal Closed Categories, MSCS, to appear (2008).

[Hi97] P. Hines. The Algebra of Self-Similarity and its Applications. Thesis. University of Wales, (1997).

[Hi03] P. Hines (2003), A categorical framework for finite state machines, *Math. Struct. in Comp. Science*, vol. **13**, 451-480.

[HiSc07] P. Hines and P. Scott (2007) Conditional Quantum Iteration from Categorical Traces, manuscript.

[Hou07] R. Houston (2007) Modelling Linear Logic without Units, PhD Thesis, Dept. of Computer Science, Manchester University.

[HylSc03] M. Hyland and A. Schalk (2003), Glueing and Orthogonality for Models of Linear Logic. *Theoretical Computer Science* vol. 294, pp. 183–231.

[JS91] A. Joyal and R. Street, The geometry of tensor calculus I, *Advances in Mathematics* **88**, 1991, pp. 55–112.

[JS93] A. Joyal and R. Street, Braided tensor categories , *Advances in Mathematics* **102**, no. 1 1993, pp. 20-79.

[JSV96] A. Joyal, R. Street, and D. Verity (1996), Traced Monoidal Categories. *Math. Proc. Camb. Phil. Soc.* **119**, 447-468.

[KRT97] C. Kassel, M. Rosso, V. Turaev. *Quantum Groups and Knot Invariants.* Soc. Mathématique de France, 1997.

[KSW02] P. Katis, N. Sabadini, R.F.C. Walters, Feedback, trace and fixed-point semantics, *Theoret. Informatics Appl* 36, (2002), 181-194.

[KL80] G. M. Kelly and M. Laplaza. Coherence for compact closed categories. *Journal of Pure and Applied Algebra* 19 (1980) 193–213 .

[KR77] A. Kock and G. Reyes, Doctrines in categorical logic, in *Handbook of Mathematical Logic*, J. Barwise, ed. North-Holland, 1977.

[LS86] J. Lambek and P. J. Scott. *Introduction to Higher Order Categorical Logic*, Cambridge Studies in Advanced Mathematics **7**, Cambridge University Press, 1986.

[L89] J. Lambek, Multicategories Revisited. *Contemp. Math.***92**, pp. 217-239.

[Lau01] O. Laurent, (2001), A Token Machine for Full Geometry of Interaction. In TLCA '01, SLNCS 2044, pp. 283-297.

[Law69] F.W. Lawvere, Adjointness in Foundations, *Dialectica*, **23**, pp. 281-296.

[Law70] F. W. Lawvere, Equality in hyperdoctrines and comprehension schema as an adjoint functor, *Applications of Category Theory*, Proc. of A.M.S. Symposia on Pure Math XVII, AMS, Providence, RI. 1970.

[Lef] S. Lefschetz, *Algebraic Topology*, Am. Math. Soc. Colloquium Publications, 1942.

[Mac98] S. Mac Lane (1998), *Categories for the Working Mathematician*, 2nd Ed. Springer.

[MR91] P. Malacaria and L. Regnier (1991), Some Results on the Interpretation of λ-calculus in Operator Algebras. *Proc. Logic in Computer Science (LICS)* pp. 63-72, IEEE Press.

[MA86] E. Manes and M. Arbib. *Algebraic Approaches to Program Semantics*, Springer-Verlag, 1986.

[Mel07] P.-A. Melliès, Categorical semantics of linear logic: a survey, 132pp. (in preparation). See website `http://www.pps.jussieu.fr/ mellies/`

[Pl03] G. Plotkin, Trace Ideals, MFPS 2003 invited lecture, Montreal (unpublished).

[Reg92] L. Regnier (1992), *Lambda-calcul et Réseaux*, PhD Thesis, Université Paris VII.

[Sch07] U. Schöpp, Stratified Bounded Affine Logic for Logarithmic Space, *Proc. Logic in Computer Science (LICS)*, IEEE, 2007, 411-420.

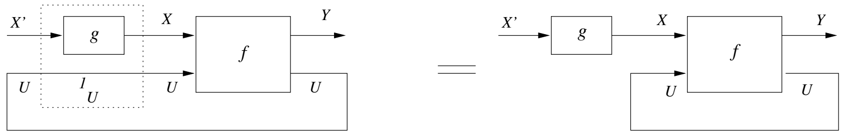
[Sc00] P. Scott, Some Aspects of Categories in Computer Science, in *Handbook of Algebra*, Vol. 2 , M. Hazewinkel, ed., 2000, Elsevier,pp. 3–77.

[See89] R.A.G. Seely. Linear logic, *-autonomous categories and cofree coalgebras. *Contemporary Mathematics*, Volume 92. American Mathematical Society, (1989).

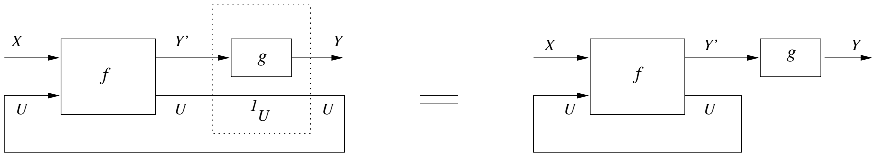
[Ste00] G. Stefanescu. *Network Algebra*, Springer-Verlag, 2000.

[TrSchw] A. S. Troelstra and H. Schwichtenberg. *Basic Proof Theory*, Camb. U. Press, 1996.

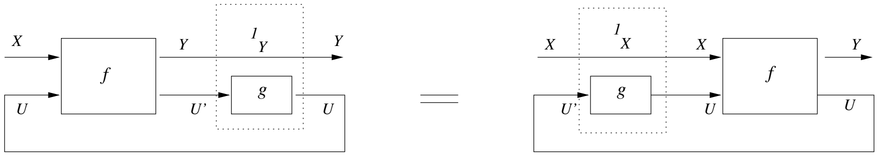
A Graphical Representation of The Trace Axioms



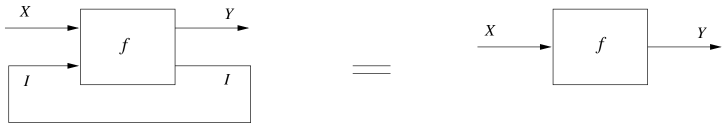
Naturality in X



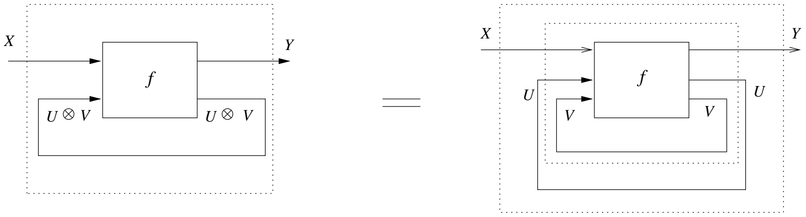
Naturality in Y



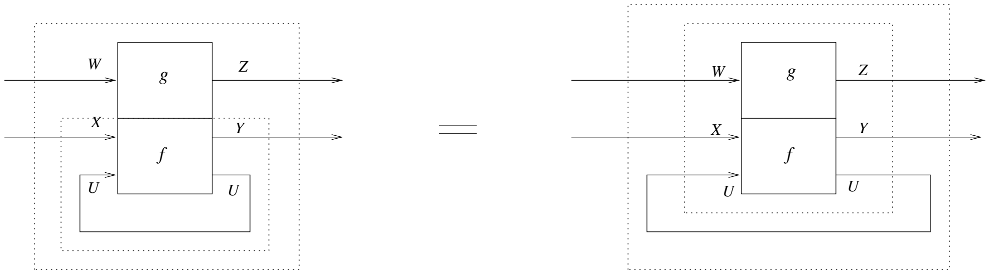
Dinaturality in U



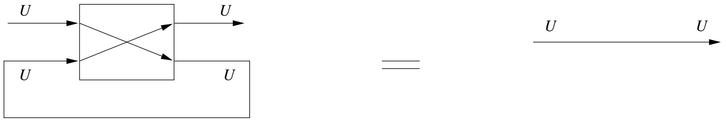
Vanishing I



Vanishing II



Superposing



Yanking

B Comparing GoI Notation

Girard	*This Paper*
$1 \otimes a$	$uT(a)v$
p, p^*	j_1, k_1
q, q^*	j_2, k_2
$(1 \otimes r), (1 \otimes r^*)$	uc_1v, uc'_1v
$(1 \otimes s), (1 \otimes s^*)$	uc_2v, uc'_2v
t, t^*	$ue_U(Tv)v, u(Tu)e'_Uv$
d, d^*	ud_U, d'_Uv