Towards Full Completeness of the Linear Logic of Chu Spaces

Vaughan Pratt*
Dept. of Computer Science
Stanford University
Stanford, CA 94305-2140
pratt@cs.stanford.edu

June 27, 1997

Abstract

We investigate the linear logic of Chu spaces as defined by its dinaturality semantics. For those formulas of multiplicative linear logic limited to at most two occurrences of each variable we prove full completeness of Girard's MIX-free axiomatization, namely that the cut-free proof-nets of such formulas are in a natural bijection with the dinatural elements of the corresponding functors.

1 Introduction

Whereas ordinary logic axiomatizes theorems, linear logic axiomatizes proofs. The semantic criterion for theoremhood is *validity*: the truth function denoted by a formula is required to be universally true. Following Lambek and Scott [11] and (as applied to linear logic) Blute and Scott [3, 4], we shall take the semantic criterion for proofhood to be *naturality*: the transformation denoted by a proof is required to commute with all morphisms of the ambient category. Dinaturality is a small but important generalization of naturality accommodating mixed variance, the possibility of a variable occurring both positively (covariantly) and negatively (contravariantly).

The strength of the naturality condition can be appreciated by considering the transformations from the identity functor on **Set** into itself, namely the (large) families $\langle f : A \to A\rangle_{A\in\mathbf{Set}}$ of endofunctions indexed by sets A, which form a proper class. Yet only a single member of that class is natural, namely the identity natural transformation $\langle 1_A : A \to A\rangle_{A\in\mathbf{Set}}$. In the categorical logic of sets this transformation is the denotation of the unique proof of the sequent $A \vdash A$. More generally, naturality has the dramatic effect of reducing a wildly chaotic class to a small set whose structure can often be attractively characterized in logical, more specifically proof-theoretic, terms.

In ordinary logic, an axiom system that generates all valid formulas is called complete. In Girard's linear logic [6], an axiom system that generates all proofs that are valid in some suitable sense has been called *fully complete* by Abramsky and Jagadeesan [1], who interpret multiplicative linear logic (MLL) over a category of games as objects and strategies as morphisms, and show full completeness of MLL with the MIX rule, from A and B infer $A \parr B$.

However, while MIX is validated by many naturally arising models of linear logic, Girard has nevertheless choosen to omit it from his axiomatization of linear logic. In support of this choice,

*This work was supported by ONR under grant number N00014-92-J-1974

Hyland and Ong [9] have modified Abramsky and Jagadeesan's model to refute MIX by disallowing certain plays they characterize as "unfair."

Game semantics imbues linear logic with a computationally appealing procedural quality. From a mathematical standpoint however the definitions appear contrived when contrasted with the more fundamental notion of naturality. Blute and Scott [3, 4] have treated the problem of full completeness of linear logic using Lauchli semantics, invariance under (continuous) group actions, dinaturality, and logical relations. It seems to us that completeness results based on naturality are of deeper significance than those based on game semantics.

In the spirit of Blute and Scott we investigate the linear logic of Chu spaces using dinaturality semantics. For those formulas of MLL having at most two occurrences of each variable we prove full completeness of Girard's MIX-free axiomatization, namely that the cut-free proof-nets of such formulas are in a natural bijection with the dinatural elements of the corresponding functors.

As a static object a Chu space (A, R, X) is merely a binary relation $R \subseteq A \times X$. The interest in Chu spaces is in their dynamics: a Chu space transforms via a pair of functions, one acting covariantly on A, the other contravariantly on X, as though it were a topological space having A as the set of points, X the set of open subsets of A, and R the membership relation between points and open sets. However there is no requirement that the open sets be closed under arbitrary union and finite intersection, making the category **Top** of topological spaces and their continuous functions a full subcategory of the category **Chu** of Chu spaces and their continuous functions. Unlike **Top** however, **Chu** is a self-dual closed category, admitting functors *perp*, $\mathcal{A}^{\perp}$, and *tensor*, $\mathcal{A} \otimes \mathcal{B}$, which interpret the corresponding basic operations of MLL.

Chu spaces were originally defined by M. Barr and studied by P. Chu [2], for the more general case of *enriched* Chu spaces (a, r, x), namely objects a, x and a morphism $r : a \otimes x \to k$ of a symmetric closed category V. The *ordinary* Chu spaces considered here were first considered in detail by Lafont and Streicher [10], who defined them independently and called them *games*. However they can be obtained from the earlier more general Chu construction simply by taking $V = \mathbf{Set}$ and k any set K. (This passage is intimately related to that betwen ordinary categories and categories enriched in V.) For the full completeness results we have been able to obtain thus far, the set $K = 2 = \{0, 1\}$ has proved sufficient. Soundness being independent of K, our results therefore apply equally well to $\mathbf{Chu}(\mathbf{Set}, 2)$ and $\mathbf{Chu}(\mathbf{Set}, K)$ for larger K.

Our interest in Chu spaces as a model of linear logic is two-fold. Our original and motivating application of Chu spaces has been as a model of concurrency, with linear logic serving as a process algebra [8, 7, 13]. More recently we have come to regard Chu spaces as "universal topology" analogous to universal algebra. A Chu space is to a topological space as an algebra (in the sense of one or more carriers and a family of operations) is to a group. The remarkable thing is that every algebra, indeed every relational structure, with or without additional topological structure, is faithfully, fully, and concretely representable as a Chu space, meaning that it has the same underlying set and transforms via the same functions, no more or less.

From this viewpoint of Chu spaces as universal objects, the operations of linear logic constitute "pure" versions of their counterparts in more application-specific categories, e.g. direct sum $U \oplus V$, tensor product $U \otimes V$, and dual U^* of vector spaces [14, 15]. They are pure in the sense that their definition makes no concessions to special features of any category of mathematics such as vector spaces or complete semilattices, whose tensor products are defined to take into account that the result should remain within the category at hand. While this condition also applies to **Chu**, the universality of Chu spaces makes the condition vacuous.

We prove three theorems stating the soundness, completeness, and full completeness respectively of the Girard axioms for the binary case of MLL under the Chu space interpretation. Soundness of

MLL for Chu spaces asserts that every theorem with a syntactic proof (in Girard's axiomatization) has a semantic proof in **Chu**. Completeness asserts that every formula having a semantic proof has a syntactic one. Full completeness asserts that the syntactic and semantic proofs are in a coherent one-to-one correspondence.

An earlier version of this paper presented at MFPS'97 was based on the behavior of the "quad," a self-dual Chu space consisting of the integers mod 4 with the consecutive-pair topology. The original rationale for concentrating on the quad was that its very interesting structure promised to extend the results beyond the binary fragment of MLL, i.e. to more than two occurrences per variable. However we have been unable to realize this promise thus far, and for the binary case treated on its own, Chu spaces limited to either one state or one point suffice and deliver the goods with considerably less fuss.

2 MLL Axiomatization

Girard developed linear logic [6] within the Gentzen tradition of inferences between sequents $\Gamma \vdash \Delta$ where Γ and Δ are lists of formulas, For multiplicative linear logic (MLL), the Gentzen-style sequent $A_1 \ldots, A_m \vdash B_1, \ldots, B_n$ may be identified with the Hilbert-style formula $A_1 \otimes \ldots \otimes A_m \multimap B_1 \parr \ldots \parr B_n$. This formulation eliminates $\vdash$ as a separate connective while admitting attractive axiomatizations that retain the essence of the distinction between proofs with and without cuts.

The three binary operations of MLL appearing in this paper are *tensor* $A \otimes B$, *par* $A \parr B$, and *linear implication* $A \multimap B$. Any two of these can be defined in terms of the third. In particular if we take $A \otimes B$ as primitive then we obtain $A \parr B$ as $(A^\perp \otimes B^\perp)^\perp$ (the De Morgan dual of $\otimes$) and $A \multimap B$ as $(A \otimes B^\perp)^\perp$. If instead we start from $A \multimap B$ then we may take $A \otimes B = (A \multimap B^\perp)^\perp$ and $A \parr B = A^\perp \multimap B$.

As will become apparent, there is no right choice of primitive connective. A natural starting point for the Chu interpretations of these operations is $\multimap$. On the other hand manipulations involving the linear distributivity rule are most efficiently understood in terms of $\otimes$ and $\parr$. But the preservation of dinaturality by linear distributivity is most naturally viewed from the perspective of $\otimes$ and $\multimap$. We will therefore make the right choice for each occasion, with the caveat that the choice will vary from one occasion to the next.

To simplify the axiomatizations and generally reduce overhead, we make the following identifications of formulas: $A \otimes (B \otimes C) = (A \otimes B) \otimes C$ (associativity), and $A^{\perp\perp} = A$ (double negation).

In addition we define an equivalence relation $\equiv$ on formulas satisfying $A \otimes B \equiv B \otimes A$ (commutativity). We would prefer that this equivalence be identity, as for associativity. This is fine for ordinary completeness, namely that existence of a dinatural implies existence of a proof. For full completeness however, namely a bijection between dinaturals and proofs, actual identification is not only confusing but unsound. We therefore settle for equivalence, reflected in our axiom systems by giving commutativity as an explicit rule while incorporating associativity and double negation into the definition of "formula" so that they do not need their own rule.[1]

The formulas of the resulting language can then have one of three forms, variables P, negated formulas $A^\perp$, and products $A_1 \otimes \ldots \otimes A_k$, $k \geq 2$, where the argument of a negated formula is either a variable or a product, and the arguments of a product are either variables or negated formulas. For theoremhood the order of the arguments of a product is immaterial. For proofhood we treat

[1]The semantic justification for treating double negation as identity is clear from the behavior of negation as transposition, which is an involution. The consistency of "on-the-nose" associativity can, Mac Lane [12, VII.1] notwithstanding, be demonstrated by taking **Set** to consist of well-ordered sets closed under lexicographic product, a categorical product that is associative up to identity. This can then be inherited by **Chu**.

formulas differing only in such order as equivalent. Equivalent formulas will be seen to denote isomorphic functors.

These formulas may equivalently be understood in the usual way as being built up from literals (variables or negated variables) with $\otimes$ and $\parr$ *alternating*, meaning that the arguments of $\otimes$ are either literals or $\parr$'s, and the arguments of $\parr$ are either literals or $\otimes$'s. Each $\parr$ corresponds to a negatively occurring $\otimes$, and each negative literal to a negatively occurring variable, where an occurrence is negative just when it is below an odd number of negations along the path from that occurrence up to the root.

We mention in passing the following axiomatization of the theorems of MLL. This amounts to the Hilbert-style formulation of linear logic with cut.

System 1.
Axioms
A1. *(Identity)* $A \multimap A$
A2. *(Composition)* $(B \multimap C) \multimap ((A \multimap B) \multimap (A \multimap C))$
Rules.
R1. *(Commutativity)* From $\varphi(A \otimes B)$ infer $\varphi(B \otimes A)$.
R2. *(Modus Ponens)* From A and $A \multimap B$ infer B.

Here A, B, C denote arbitrary formulas, making A1 and A2 axiom schemas rather than axioms. $\varphi(A \otimes B)$ denotes a formula φ containing $A \otimes B$ as a subformula (which may appear as any two consecutive A_i's of a larger subformula $A_1 \otimes \ldots \otimes A_k$ of φ).

Rule R2 is a restricted form of the cut rule. Whereas cut denotes external composition, R2 denotes external application.

Modus ponens is an inessentially restricted form of the cut rule, which permits arbitrarily large formulas to be built up in the course of proving a given theorem. Hence theorems proved with modus ponens may have arbitrarily long proofs.

We shall not pursue System 1 further. Our attention will focus instead on the following gentler system, which eliminates Modus Ponens, along with its ability to stretch out or sometimes shorten a proof.

System 2.
(Axiom)
A1. *(Iterated identity)* $(P_1 \multimap P_1) \otimes \ldots \otimes (P_n \multimap P_n)$
Rules.
R1. *(Commutativity)* From $\varphi(A \otimes B)$ infer $\varphi(B \otimes A)$.
R2. *(Linear distributivity)* From $\varphi((A \multimap B) \otimes C)$ infer $\varphi(A \multimap (B \otimes C))$.

Each P_i in axiom A1 denotes a simple variable, in contrast to our use of $A, B, \ldots$ which denote arbitrary formulas. The P_i's need not be all distinct; thus $(P \multimap P) \otimes (Q \multimap Q) \otimes (P \multimap P)$ is an instance of A1 in which $n = 3$, $P_1 = P_3 = P$, and $P_2 = Q$.

The rule R1 is as in System 1.

In R2, the subformula $(A \multimap B) \otimes C$ may appear at the top of φ, in which case $\varphi((A \multimap B) \otimes C)$ is simply $(A \multimap B) \otimes C$. Or it may occur as a proper subformula of φ, in which case we may without loss of generality view φ as having the form $D_1 \multimap (E_1 \otimes (D_2 \multimap (E_2 \otimes \ldots (D_k \multimap (E_k \otimes ((A \multimap B) \otimes C))) \ldots)))$ for $k \geq 1$, optionally without the leading $D_1 \multimap$ (to handle the case when the outermost operation is tensor) and without the trailing $E_k \otimes$ (when $A \multimap B$ and C are the only arguments of their parent $\otimes$).

The linear distributivity rule replaces the subformula $(A \multimap B) \otimes C$ with $A \multimap (B \otimes C)$, leaving the rest of φ untouched. Structurally the rule behaves like associativity. Thus if a proof begins with $2n$ literal occurrences, at the end it still has the same $2n$ literal occurrences. Furthermore every proof step pushes some $\otimes$ down past a $\invamp$. (It is a nice exercise to show that the longest possible proof has $n(n+1)/2 - 1$ steps where n is the number of identity implications in the instance of A1 with which the proof begins.)

Rule R2 in System 2 is closely related to System 1's A2. The MLL formula corresponding most closely to R2 is $((A \multimap B) \otimes C) \multimap (A \multimap (B \otimes C))$. Substituting $C^\perp$ for C and suitably rearranging yields A2.

To each theorem of System 2 we associate one or more *cut-free proof nets*, each determined by a proof of that theorem. For any given formula, such a proof net is given simply by a matching of opposite literals of that formula.

For the axiom A1, the associated proof net matches up n pairs of literals, one pair from each subformula $P_i \multimap P_i$. Since linear distributivity neither creates nor annihilates literals, and since the identities of the literals can be kept track of from one step to the next, the initial matching in A1 can simply be retained at each step. By this means, any given derivation of a theorem associates a proof net to that theorem. We take that proof net as the abstract syntactic proof of the theorem.

When all n pairs of P_i's are distinct in A1, all consequences of A1 have a unique matching, in which case all derivations of a given theorem determine the same abstract proof net. However repetitions of P_i's, as in $(P \multimap P) \otimes (P \multimap P)$, raise the possibility that a conseqence such as $(P \otimes P) \multimap (P \otimes P)$ may have more than one matching. In this example two matchings are possible, which may be understood as corresponding to the two theorems $(P \otimes Q) \multimap (P \otimes Q)$ and $(P \otimes Q) \multimap (Q \otimes P)$, in each of which Q is then renamed to P. (Before the renaming these two theorems are equivalent but not identical.)

Danos and Regnier [5] give the following characterization of those matchings arising in this way as cut-free proofs of a theorem. Take the alternating-$\otimes$-and-$\invamp$ view of the theorem, with literals at the leaves. Associate to this theorem an undirected graph whose vertices are the operations and literals, and whose edges connect each operation to its arguments, in other words the usual parse tree with the edges understood as undirected. This of course is a tree, in the undirected-graph sense that it is connected (i.e. consists of one connected component) and acyclic.

Now add further edges, one per matched pair of literals, which in general introduces cycles. Then for each $\invamp$ operation, delete all but one of its descendent edges. This deletion may remove some or all cycles, or even separate the graph into two or more components. The Danos-Regnier criterion for a matching to be a correct cut-free proof-net is that, no matter which edges are selected for deletion, the resulting graph is a tree (connected and acyclic). That is, the deletions must remove all cycles yet leave the graph connected.

It is not hard to show that every proof net produced by System 2 as described above meets this criterion. Somewhat less obviously, the converse also holds: every proof-net meeting this criterion is derivable in System 2.

3 Chu spaces

A *Chu space* $\mathcal{A} = (A, R, X)$ *over* the set $2 = \{0, 1\}$ consists of sets A and X and a binary relation $R \subseteq A \times X$. We interpret A and X as the carrier and cocarrier of $\mathcal{A}$, consisting of respectively *points* $a, b, \ldots$ and *states* $x, y, \ldots$.

We write R_a for $\{x \in X \mid R(a, x)\}$ and R^x for $\{a \in A \mid R(a, x)\}$, called respectively *row* a and *column* x. Rows and columns are to be understood as representations of points and states respectively. A Chu space no two states (respectively points) of which have the same representation is called *extensional* (respectively *separated* or T_0); with both properties it is called *biextensional*. (So an extensional Chu space whose columns are closed under arbitrary union and finite intersection constitutes a topological space whose open sets are the columns.) In this paper we shall work mainly with biextensional spaces.

A Chu space is called *discrete* when it is extensional and every subset of its carrier appears as a column, and *coherent* when it is separated and every subset of its cocarrier appears as a row.

The *dual* of $\mathcal{A} = (A, R, X)$, denoted $\mathcal{A}^\perp$, is $(X, R^{\smile}, A)$ where the converse $R^{\smile}$ is defined as $R^{\smile}(x, a) = R(a, x)$. Duality interchanges points and states, rows and columns, separation and extensionality, and discreteness and coherence. Clearly $\mathcal{A}^{\perp\perp} = \mathcal{A}$.

The following spaces play a prominent role in this paper. 0 denotes the empty discrete space, having no points and one state x. $\top$ denotes the inconsistent singleton $0^\perp$, having one point and no states (whence "inconsistent"). 1 denotes the singleton discrete space (A, R, X) where $A = \{a\}$, $X = \{x, y\}$, and $R = \{(a, x)\}$; viewing x and y as open sets and R as the membership relation, x corresponds to the whole space and y to the empty space. $\perp$ denotes the coherent doubleton $1^\perp$, having two points and one state. These provide the Chu space interpretations of the linear logic constants of those names.

There are two Chu spaces with one point and one state, each obtainable by omitting one of the two states of the Chu space 1. We denote by W (for white) the space with the empty open set, and B (black) the other.

Statically a Chu space is just a binary relation. The intrinsic interest in Chu spaces only emerges when its transformational structure, or some adequate reflection thereof, is included in the picture, as follows.

A *Chu transform*, or just *map*, from (A, R, X) to (B, S, Y) is an *adjoint pair* $(f, f^\perp)$ of functions $f{:}A \to B$, $f^\perp{:}Y \to X$, f for short, defined as satisfying $S(f(a), y) = R(a, f^\perp(y))$ for all $a \in A$ and $y \in Y$, called the *adjointness condition*. The composite

$$(A, R, X) \xrightarrow{f} (B, S, Y) \xrightarrow{g} (C, T, Z)$$

defined by $(g, g^\perp)(f, f^\perp) = (gf, f^\perp g^\perp)$ is a Chu transform because

$$T(gf(a), z) = S(f(a), g^\perp(z)) = R(a, f^\perp g^\perp(z)).$$

Hence Chu spaces and their maps so composed form the category **Chu**.

When the domain of a map is extensional, it may be seen that $f^\perp{:}Y \to X$ is determined by $f{:}A \to B$, in which case $(f, f^\perp)$ may be abbreviated to f without ambiguity as to $f^\perp$. Dually when its target is separated, $f^\perp$ determines f. When (A, R, X) is discrete, the maps to (B, S, Y) are exactly the functions $f{:}A \to B$, with the evident dual when (B, S, Y) is coherent.

For the case of topological spaces represented as above, the adjointness condition for $(f, f^\perp)$ is readily seen to be equivalent to continuity of f (and f determines $f^\perp$ by extensionality). In this sense a Chu space is a generalized topological space.

4 Interpreting the Linear Logic Connectives

For MLL it suffices to give the interpretations of just two connectives; we shall define $\mathcal{A}^\perp$ and $\mathcal{A} \multimap \mathcal{B}$. This interpretation of MLL being categorical, we must specify these not merely as operations but as functors, giving their behavior not just on objects but also on morphisms. From these we then obtain $\mathcal{A} \otimes \mathcal{B}$ as $(\mathcal{A} \multimap \mathcal{B}^\perp)^\perp$ and $\mathcal{A} \parr \mathcal{B}$ as $\mathcal{A}^\perp \multimap \mathcal{B}$, noting that this defines $\otimes$ as $\parr$ as functors.

We have already described $\mathcal{A}^{\perp}$ for objects. For morphisms, namely Chu transforms, the evident behavior of $(f, f^{\perp})^{\perp}$ is $(f^{\perp}, f)$, that is, simply exchange the two maps to give $(f^{\perp}, f) : \mathcal{B}^{\perp} \to \mathcal{A}^{\perp}$.

Linear Implication. Given $\mathcal{A} = (A, R, X)$ and $\mathcal{B} = (B, S, Y)$, we define $\mathcal{A} \multimap \mathcal{B}$ to be $(G, T, A \times Y)$ where G is the set of Chu transforms from $\mathcal{A}$ to $\mathcal{B}$ and $T(g, (a, y)) = S(g(a), y)$. In other words each row of $\mathcal{A} \to \mathcal{B}$ represents a Chu transform $h : \mathcal{A} \to \mathcal{B}$ as an A-tuple of rows of $\mathcal{B}$, with the a-th component of that tuple being the row $h(a)$ in $\mathcal{B}$.

This is the most natural representation of h one could imagine. Yet from this representation of the Chu transforms from $\mathcal{A}$ to $\mathcal{B}$ as the rows of $\mathcal{A} \multimap \mathcal{B}$ we can read off the "topology" of $\mathcal{A} \multimap \mathcal{B}$ as its columns, one open set per element of $A \times Y$.

The action of $\multimap$ on morphisms is defined so as to give this operation the characteristics of an internal homfunctor. Given Chu transforms $f : \mathcal{A}' \to \mathcal{A}$ and $h : \mathcal{B} \to \mathcal{B}'$, we take $f \multimap h$ to be the Chu transform from $\mathcal{A} \multimap \mathcal{B}$ to $\mathcal{A}' \multimap \mathcal{B}'$ whose covariant half is $\lambda g.hgf$, sending the Chu transform $g \in G$ (where $\mathcal{A} \multimap \mathcal{B} = (G, T, A \times Y)$) to the composite $hgf : \mathcal{A}' \multimap \mathcal{B}'$. Since Chu transforms are closed under composition hgf is a Chu transform, showing that the covariant half is well-defined.

It remains to produce the contravariant half of $f \multimap h$ and verify the adjointness condition. Evidently it should map $(a', y') \in A' \times Y'$ to $(f(a'), h^{\perp}(y')) \in A \times Y$. Instantiating the adjointness condition calls for the equality of $T'(\lambda g.hgf(g), (a', y'))$ (namely $T(hgf, (a', y'))$) and $T(g, (f(a'), h^{\perp}(y')))$. The definition of T gives the left hand side as $S'(hgf(a'), y')$ and the right hand side as $S(g(f(a')), h^{\perp}(y'))$, but these are equal because h is a Chu transform, and we are done.

One caveat: when $\mathcal{A}$ is extensional and $\mathcal{B}$ is separable, $\mathcal{A} \multimap \mathcal{B}$ will be separable by construction (distinct h's must be distinctly represented). However even if $\mathcal{A}$ and $\mathcal{B}$ are biextensional, $\mathcal{A} \multimap \mathcal{B}$ need not be extensional. When biextensionality is to be preserved, as we shall assume for this paper, this is achieved by identifying repeated columns.

When our definition of $\mathcal{A} \otimes \mathcal{B}$ as $(\mathcal{A} \multimap \mathcal{B}^{\perp})^{\perp}$ is expanded using the interpretation of $\multimap$, the resulting interpretation of $\mathcal{A} \otimes \mathcal{B}$ can be described as $(A \times B, T, G)$ where G consists of the Chu transforms from $\mathcal{A}$ to $\mathcal{B}^{\perp}$ and T satisfies $T((a, b), g) = S(b, g(a)) = S^{\smile}(g(a), b)$, which by adjointness of g equals $R(a, g^{\perp}(b))$. The columns of $\mathcal{A} \otimes \mathcal{B}$ can therefore be seen to be all $A \times B$ matrices whose columns are columns of $\mathcal{A}$ and whose rows are columns of $\mathcal{B}$. This perspective on $\mathcal{A} \otimes \mathcal{B}$ is as almost as simple as the view of $\mathcal{A} \multimap \mathcal{B}$ as consisting of rows of functions naturally represented.

5 Dinaturality

5.1 Denotation of Terms: Functors

The Chu functors $\mathcal{A}^{\perp}$ and $\mathcal{A} \multimap \mathcal{B}$ (or $\mathcal{A} \otimes \mathcal{B}$ depending on which we take as primitive), as interpretations of the *connectives* of MLL, compose to yield n-ary functors interpreting the *formulas* of MLL having n variables. The following defines the denotation of terms in any category C having such functors.

Functoriality being the categorical counterpart of monotonicity, in order to cater for nonmonotone operations such as negation and implication it is necessary to extend the notion of functor to accommodate functors of mixed variance. This notion is can be described with ordinary functors by typing it as $F : (C^{\mathrm{op}})^n \times C^n \to C$. We shall refer to such a functor as an n-ary *sesquifunctor*. These are the entities between which dinatural transformations run.

An MLL formula A in n variables denotes an n-ary sesquifunctor F defined by induction on the structure of A, understood as being built from literals (positive and negative occurrences of variables) using $\otimes$ and $⅋$. In giving this definition, things generally read more naturally when one adopts the polymorphic point of view in which F is treated as a polymorphic or variable object residing in F's codomain, with F's domain constituting the parameter space or domain of variation of that object, over which the variables $P_1, \ldots, P_n$ of the formula range. We specify F by giving its value at the generic point $(p_1, \ldots, p_n; q_1, \ldots, q_n)$ of parameter space. As usual we shall understand p_i and q_i as ranging over morphisms as well as objects, otherwise $(p_1, \ldots, p_n)$ could just as well have come from C^n as $(C^{\mathrm{op}})^n$.)

The positive literal P_i denotes q_i, the negative literal $P_i^\perp$ denotes p_i. If A, B denote a, b respectively then $A \otimes B$ and $A ⅋ B$ denote $a \otimes b$ and $a ⅋ b$ respectively. (In this polymorphic form the definitions of the binary connectives appear vacuous.)

For example $(P_1 \multimap P_2) \otimes P_1$ denotes $(p_1 \multimap q_2) \otimes q_1$ because the first occurrence of P_1 is negative while the other two literal occurrences are positive.

The above specializes to $C = \mathbf{Chu}$ in the evident way.

5.2 Denotation of Proof-Nets: Dinatural elements

The following diagrams illustrate the evolution of the notion of dinatural element. The first diagram shows the usual naturality condition on a transformation τ between two functors F, G, expressing independence of τ from all perturbing morphisms $f : p \to q$ of C^n (so p, q, f denote n-tuples, f being $\langle f_i : p_i \to q_i \rangle_{i=1..n}$). The second diagram extends this to sesquifunctors, calling for the exhibited split of the vertical arrows.

$$
\begin{tikzcd}
F(p) \arrow[r, "\tau_p"] \arrow[d, "F(f)"'] & G(p) \arrow[d, "G(f)"] \\
F(q) \arrow[r, "\tau_q"] & G(q)
\end{tikzcd}
\qquad
\begin{tikzcd}
F(p;p) \arrow[r, "\tau_p"] & G(p;p) \arrow[d, "G(p;f)"] \\
F(q;p) \arrow[u, "F(f;p)"] \arrow[d, "F(q;f)"'] & G(p;q) \\
F(q;q) \arrow[r, "\tau_q"] & G(q;q) \arrow[u, "G(f;q)"']
\end{tikzcd}
\qquad
\begin{tikzcd}
1 \arrow[r, "\tau_p"] \arrow[d, equal] & H(p;p) \arrow[d, "H(p;f)"] \\
1 & H(p;q) \\
1 \arrow[r, "\tau_q"] \arrow[u, equal] & H(q;q) \arrow[u, "H(f;q)"']
\end{tikzcd}
$$

The third diagram shows the simplification possible when C is a $*$-autonomous category. We can transpose each morphism $\tau_p : F(p;p) \to G(p;p)$ to an element τ_p of $H(p;p)$, where $H(p;q)$ is defined as $(F \multimap G)(p;q)$. We refer to the resulting dinatural transformation as a *dinatural element.* Here 1 is the discrete singleton Chu space defined earlier, consisting of one point and two states.

The same polymorphism we used in specifying the denotations of MLL terms, with the same domain of variation, lets us think of dinatural transformations as variable morphisms, and of dinatural elements as variable elements or points. But unlike sesquifunctors, which are defined on the whole of $(C^{\mathrm{op}})^n \times C^n$, dinaturals are defined only on its diagonal, namely at pairs of n-tuples $(p_1, \ldots, p_n; p_1, \ldots, p_n)$, with the further requirement that the p_i's be objects. This restriction makes the $(C^{\mathrm{op}})^n$ portion of parameter space redundant for dinaturals, and the generic parameter can therefore be simplified to $(p_1, \ldots, p_n)$.

6 Soundness

No completeness proof is complete without a sound soundness proof to complement it. The following enunciation of soundness holds more generally for any $*$-autonomous category [2], of which **Chu** is just an instance. This is a well-understood result whose proof we include for the benefit of newcomers to categorical logic. Stating and proving it for just **Chu** avoids the considerable work of defining a $*$-autonomous category, of which **Chu** is a reasonably representative example.

Theorem 1 *(Soundness) Every n-variable theorem of MLL, when interpreted as an n-ary sesqui-functor on* **Chu**, *contains a dinatural element.*

Proof:

We prove this by induction on the length of proofs in System 2. The obvious choice of dinatural element of the functor interpreting $(P_1 \multimap P_1) \otimes \ldots (P_n \multimap P_n)$ is the n-tuple $(1_{p_1}, \ldots, 1_{p_n})$, where the generic point parametrizing the polymorphism is $(p_1, \ldots, p_n)$ as before, now ranging over $\mathbf{Chu}^n$.

To establish dinaturality of this element, observe that for any n-tuple $\langle f_i : p_i \to q_i \rangle$ of Chu transforms, $F(p_1, \ldots, p_n; f_1, \ldots, f_n) \circ (1_{p_1}, \ldots, 1_{p_n})$ and $F(f_1, \ldots, f_n; q_1, \ldots, q_n) \circ (1_{q_1}, \ldots, 1_{q_n})$ both simplify to $(f_1, \ldots, f_n)$.

Rule R1 is an easy exercise once R2 is understood. Evidently it is realized as an isomorphism, constituting the denotation of the equivalence (in fact congruence) $A \otimes B \equiv B \otimes A$ of formulas and more generally of any formula containing these as subformulas.

For R2, suppose $\varphi((A \multimap B) \otimes C)$ has a dinatural element. We exhibit the corresponding dinatural element of $\varphi(A \multimap (B \otimes C))$. We prove this first for the case where $(A \multimap B) \otimes C$ is the whole formula, then generalize.

Let the formulas A, B, C denote the (variable) Chu spaces $\mathcal{A}, \mathcal{B}, \mathcal{C}$ respectively. A dinatural element of $(\mathcal{A} \multimap \mathcal{B}) \otimes \mathcal{C}$ is a pair (g, c) where $g : \mathcal{A} \to \mathcal{B}$ is a dinatural element of $\mathcal{A} \multimap \mathcal{B}$, equivalently a dinatural transformation from $\mathcal{A}$ to $\mathcal{B}$, and c is a point of $\mathcal{C}$. There is an evident choice of corresponding dinatural element of $\mathcal{A} \multimap (\mathcal{B} \otimes \mathcal{C})$, which we can write as $\lambda a.(g(a), c)$, viewable as a transformation from $\mathcal{A}$ to $\mathcal{B} \otimes \mathcal{C}$.

Claim: The following diagram commutes. (Here $f : p \to q$ is as before the generic test of (di)naturality while g, g' are the values of the given dinatural transformation from $\mathcal{A}$ to $\mathcal{B}$ at p, q respectively.)

$$\begin{array}{ccc}
\mathcal{A}(p;p) & \xrightarrow{\lambda a.(g(a),c)} & \mathcal{B}(p;p) \otimes \mathcal{C}(p;p) \\
\mathcal{A}(f;p)\big\uparrow & & \big\downarrow \mathcal{B}(p;f) \otimes \mathcal{C}(p;f) \\
\mathcal{A}(q;p) & & \mathcal{B}(p;q) \otimes \mathcal{C}(p;q) \\
\mathcal{A}(q;f)\big\downarrow & & \big\uparrow \mathcal{B}(f;q) \otimes \mathcal{C}(f;q) \\
\mathcal{A}(q;q) & \xrightarrow{\lambda a.(g'(a),c)} & \mathcal{B}(q;q) \otimes \mathcal{C}(q;q)
\end{array}$$

This follows from the fact that the maps associated with B and C commute separately, by dinaturality of g and c, whence so does their tensor product. We conclude that $\lambda a.(g(a), c)$ is dinatural.

When $(A \multimap B) \otimes C$ appears as a proper subformula of $\varphi((A \multimap B) \otimes C)$, we can view the latter as having the form $D_1 \multimap (E_1 \otimes (D_2 \multimap (E_2 \otimes \ldots (D_k \multimap (E_k \otimes ((A \otimes B) \multimap C))) \ldots)))$. This is the same situation as before, except that f now becomes a function of the Chu space $\mathcal{D}_1 \otimes \mathcal{D}_2 \otimes \ldots \otimes \mathcal{D}_k$, where each D_i denotes $\mathcal{D}_i$. ∎

Like f and c, the $\mathcal{D}_i$'s are polymorphically dependent on the parameter space $\mathbf{Chu}^n$; without the polymorphism convention this picture would be much messier.

7 Completeness

In this section we prove completeness of BMLL under the Chu interpretation, namely that every BMLL formula with a semantic proof has a syntactic proof.

Theorem 2 *Every BMLL formula A containing a dinatural is a theorem of MLL.*

Proof: We follow the line of argument initiated by Abramsky and Jagadeesan applicable in the presence of MIX, and generalized by Hyland and Ong in the absence of MIX. Call an MLL formula *semisimple* when it is a *par* of tensor products of literals. (With MIX one uses the stronger notion of simple in which the tensor products contain at most two literals.) A semisimple formula can equivalently be described as being in disjunctive normal form. Hyland and Ong show that if all the semisimple consequents of an MLL formula are theorems then so is that formula. Hence to show that a formula is not a theorem it suffices to show that none of its semisimple consequents are.

We first treat the case of balanced binary, each variable occurring twice, once positively and once negatively. This case uniquely determines a proof-net.

We use the Danos-Regnier criterion to show that every semisimple nontheorem A of BMLL contains no dinatural. We do this by giving a particular assignment of Chu spaces to variables for which A evaluates to the 0×1 Chu space 0.

A semisimple BMLL formula determines an undirected graph whose vertices are the tensor products and whose edges are complementary pairs of literals each in one of the tensor products. Since there is only one $\parr$, at the root of A, its switchings are immaterial. Incorrect proof-nets as such graphs are therefore either disconnected or contain a cycle.

Case 1. The graph of A is disconnected.

We use the fact that all occurrences of a variable must occur within a connected component, since the formulas are binary. Set the variables of one component to the 1×1 Chu space whose one entry is 0, call this W for white. Set the remaining variables to the 1×1 Chu space whose one entry is 1, call this B for black.

Easy calculations show $W^\perp = W \otimes W = W \parr W$, $B^\perp = B \otimes B = B \parr B$, $B \parr W = 0$ where 0 is the 0×1 (empty discrete) Chu space. It follows that each clause in the W component evaluates to W, as does the *par* of those clauses. Similarly each of the remaining clauses evaluates to B, as does their *par*. The *par* of the B clauses and the W clauses is then 0.

We illustrate this case with the formula $P \parr P^\perp \parr Q \parr Q^\perp$, all clauses of which are singletons. The P link connects the first two clauses while the Q link connects the other two, but this leaves two connected components. Setting P to white and Q to black makes $P \parr P^\perp$ white, $Q \parr Q^\perp$ black, and their par 0.

Case 2. The graph of A is connected and contains a cycle.

Select a cycle and orient it, and hence its links. Orient the remaining links to point towards the cycle, in the sense that the target end of the link is closer to the cycle than the source end, with ties broken arbitrarily. Now assign 0 or $\top = 0^\perp$ to the variables in such a way that every link is oriented from 0 to $\top$. It is immediate that all clauses on the cycle contain a 0 literal, and an argument by induction on distance from the cycle extends this property of clauses to the remainder of the graph. Since $0 \otimes \mathcal{A} = 0$ for all Chu spaces $\mathcal{A}$, all clauses evaluate to 0 and hence so does their *par*.

We illustrate case 2 with the formula $(A \otimes B) \parr (A^\perp \otimes B^\perp \otimes C) \parr C^\perp$. The A and B links between the first two clauses create a cycle with two edges. To orient it properly set A to 0 and B to $\top$ (or vice versa). This makes the first two clauses 0. Setting C to $\top$ zeros both $C^\perp$ and the whole formula.

This disposes of the balanced binary case. In the unbalanced case, the two-occurrence limit means that at least one literal has no oppositely signed counterpart. Assign that literal 0, and all other literals W (the 1×1 "white" Chu space, but B would do as well). Then all clauses evaluate to either 0 or W, and there is at least one 0 clause. Since $W \parr W = W$ and $W \parr 0 = 0 \parr 0 = 0$, it follows that the formula evaluates to 0. Hence no dinatural is possible for the unbalanced binary case, or for that matter any formula in which all occurrences of some variable have the same sign.

$\blacksquare$

8 Full Completeness

In this section we prove that every dinatural element of the functor denoted by a binary MLL theorem A corresponds to a cut-free proof net of A. In the binary case this is the rather unexciting result that a theorem has only one dinatural element. We shall first show this for semisimple binary theorems and then extend to general binary formulas.

Theorem 3 *(Full completeness) Every semisimple theorem of BMLL has at most one dinatural.*

(The soundness result showed that it has at least one dinatural.)

Proof: Let A be such a theorem. Being binary, it uniquely determines a graph whose edges are links $P - P^\perp$ as in the previous section. Being a theorem, its graph is an undirected tree. Make it a rooted oriented tree by choosing any singleton clause for the root and orienting the edges to point towards the root.

We shall show by induction on the number n of variables that a dinatural element of A is uniquely determined at every assignment (point of parameter space). We shall start from the assignment of all 1's and proceed to an arbitrary assignment in n steps, showing that at each intermediate assignment all dinaturals have the same value at that assignment.

For the basis of the induction we take the assignment assigning the Chu space $\perp$ to the distal end of each link (that is, to the literal associated with that end of that link) and hence 1 to its near end. Since there is one edge leaving each clause except the root, which has no edges leaving it, it follows that the root clause evaluates to 1 while the remaining clauses evaluate to $1 \otimes 1 \otimes \ldots \otimes \perp \otimes \ldots \otimes 1 = \perp$. Hence A evaluates to 1. But 1 has only one element, so every dinatural element must be the same here.

We now move from this assignment to an arbitrary assignment. For simplicity of exposition assume that the distal literal occurrence of each P is negative, substituting $P^\perp$ for P if necessary, so that all variables have value 1 before the move. Assign Chu spaces arbitrarily to variables. We shall transform the all-ones assignment into this arbitrary assignment one variable at a time, at each stage showing that there is still at most one dinatural element.

Enumerate the variables in order of increasing distance from the root, with ties broken arbitrarily. For each variable P we first change the value of its positive occurrence to the Chu space chosen above for P, call it $\mathcal{A}$, and then change the value of its negative occurrence $P^\perp$ from $\perp$ to $\mathcal{A}^\perp$.

The value of A during the processing of one such variable P can be expressed as $(\mathcal{B} \parr P^\perp) \parr (P \otimes \mathcal{C})$ where $\mathcal{B}$ and $\mathcal{C}$ remain fixed while P varies.

$\mathcal{C}$ is the value, at the start of this stage, of the clause containing the positive occurrence of P. $P^\perp$ comes from the clause containing $P^\perp$, all of whose literals other than $P^\perp$ are positive and have value 1 throughout this stage, being further from the root than P. $\mathcal{B}$ denotes the combined value of all the remaining clauses (as combined by $\parr$). It will be convenient to rename $\mathcal{B}$ to $\mathcal{B}^\perp$ allowing us to rewrite this expression as $(\mathcal{B} \otimes P) \multimap (\mathcal{C} \otimes P)$.

At the start of this stage $P = 1$, so $A = \mathcal{B} \multimap \mathcal{C}$. By the induction hypothesis all dinatural elements have the same value here, call it $f : \mathcal{B} \to \mathcal{C}$. Transform the positive (righthand) occurrence of P from 1 to $\mathcal{A}$ via an arbitrarily chosen map $a : 1 \to \mathcal{A}$ (an element of $\mathcal{A}$). This determines a map $\mathcal{B} \multimap (\mathcal{C} \otimes a) : (\mathcal{B} \multimap (\mathcal{C} \otimes 1)) \to (\mathcal{B} \multimap (\mathcal{C} \otimes \mathcal{A}))$. This map sends the dinatural f to $\lambda c.\langle f(c), a\rangle$, a function from $\mathcal{B}$ to $\mathcal{C} \otimes \mathcal{A}$.

Now consider $(\mathcal{B}\otimes\mathcal{A})\multimap(\mathcal{C}\otimes\mathcal{A})$. The above map $a : 1 \rightarrow \mathcal{A}$ determines a map $(\mathcal{B}\otimes a)\multimap(\mathcal{C}\otimes\mathcal{A}) : ((\mathcal{B}\otimes\mathcal{A})\multimap(\mathcal{C}\otimes\mathcal{A})) \rightarrow ((\mathcal{B}\otimes 1)\multimap(\mathcal{C}\otimes\mathcal{A})$. Any dinatural element g at $(\mathcal{B}\otimes\mathcal{A}) \rightarrow (\mathcal{C}\otimes\mathcal{A})$ is sent by this map to $\lambda c.g(c, a)$, which by dinaturality must equal $\lambda c.\langle f(c), a\rangle$ as defined above. This must hold for all elements $a \in \mathcal{A}$, whence g must satisfy $g(c, a) = \langle f(c), a\rangle$ for all $c \in \mathcal{B}$ and $a \in \mathcal{A}$. But this uniquely determines g, whence $(\mathcal{B}\otimes\mathcal{A})\multimap(\mathcal{C}\otimes\mathcal{A})$ can contain at most one dinatural.

Proceeding in this way for all variables of A, we arrive at the arbitrarily chosen assignment to all variables, at which A still has at most one dinatural element. ∎

We now extend the above theorem from the semisimple case to general formulas of BMLL.

Theorem 4 *(Full completeness) Every theorem of BMLL has at most one dinatural.*

Proof: It suffices to show that linear distributivity maps dinaturals injectively. For if our claim were false and some BMLL theorem had two or more dinaturals, each of its semisimple consequents would have two or more as well, contradicting the previous theorem.

Consider first the top-level case where $(A\multimap B)\otimes C$ derives $A\multimap(B\otimes C)$. Here (f, c) is mapped to $\lambda a.(f(a), c)$. If $(f, c) \neq (f', c')$ then either $f \neq f'$ or $c \neq c'$. In either case $\lambda a.(f(a), c) \neq \lambda a.(f'(a), c')$.

In the general case the dinatural element being so transformed becomes a function from $D_1 \otimes D_2 \otimes \ldots \otimes D_k$ to $A\multimap(B\otimes C)$. Having two distinct dinaturals means that for some tuple $(d_1, \ldots, d_k)$ we have $(f, c) \neq (f', c')$ as above. But then at the same tuple we have the corresponding $\lambda a.(f(a), c) \neq \lambda a.(f'(a), c')$. ∎

9 Future Work

The long-term goal is the enumeration of all dinatural transformations representable in **Chu** by the full language of linear logic. A more immediate goal is to extend the results of this paper beyond the binary case to all of MLL.

We now discuss some subcases and side attractions.

9.1 Tensor only

One extension of the binary case that turns out to have an easy solution is as follows.

Theorem 5 *The dinatural transformations between the functors denoted by two terms A, B each built from positive literals only with $\otimes$ correspond to MLL proofs of $A\multimap B$.*

Proof: We outline the argument for the case $A \otimes A \vdash A \otimes A$, leaving as an exercise its generalization to arbitrary products of variables.

The argument in the preceding section on full completeness, when adapted to this situation, shows only that any dinatural must fix the *diagonal* of $A\otimes A$. (Consider the unique map from $1\otimes 1$ to $1 \otimes 1$, then take any element $a : 1 \rightarrow \mathcal{A}$ of any Chu space $\mathcal{A}$. The point (a, a) of $\mathcal{A}\otimes\mathcal{A}$ must be fixed by any dinatural transformation.)

To extend off the diagonal, set A to the discrete doubleton 2, a Chu space with two points and four states, and let $(a, b) : 2 \rightarrow \mathcal{A}$ be an arbitrary ordered pair of points of an arbitrary Chu space $\mathcal{A}$, sending the "upper right corner" 01 of $2\otimes 2$ to the arbitrary element (a, b) of $A\otimes\mathcal{A}$. The value of any dinatural at $A = 2$ is a Chu transform from $2 \otimes 2$ to itself which in conjunction with the above map (a, b) determines by dinaturality where the dinatural at $A = \mathcal{A}$ sends the element (a, b)

of $\mathcal{A} \otimes \mathcal{A}$. Hence the dinaturals from $A \otimes A$ to itself are uniquely determined by their values at $A = 2$.

Now there can only be 4^4 functions from $2 \otimes 2$ to itself, of which only $2^4 = 16$ fix the diagonal of $2 \otimes 2$, corresponding to 4 places to send the upper right element 01 of $2 \otimes 2$ and 4 for the lower left, 10.

We now claim that where 01 goes determines where 10 goes, namely they remain diametrically opposite in $2 \otimes 2$. Consider the twist map $\neg : 2 \to 2$. $\neg \otimes \neg$ swaps the upper right (01) and lower left (10) elements of $2 \otimes 2$, and the claim then follows by inspection of the dinaturality condition at the twist map.

We are now down to four transformations from $A \otimes A$ to itself. But so far we have considered only discrete Chu spaces, the Chu counterpart of sets. In **Set** all four of these transformations are dinatural, corresponding to the identity, $\neg \otimes \neg$, and the two projections. MLL has only two proofs of $A \otimes A$ from itself, corresponding to the first two of these. To get rid of the two projections we must consider other than discrete spaces.

All we shall need is the Chu space $\mathcal{D} = (\{0,1\}, \{(1,x)\}, \{x,y\})$, which represents the doubleton pointed set where the point 0 is the constant. Calculation reveals $\mathcal{D} \otimes \mathcal{D}$ to be smash product, identifying the pairs 00, 01, and 10. Now consider the map from 2 to $\mathcal{D}$ sending 0 to 0 and 1 to 1. The projection transformation moves 01 and 10 parallel to each other to reach 00 and 11 respectively (if projecting onto the first coordinate). Smash product however moves them at right angles to each other, both moving to 00. This difference is enough to break dinaturality of projection. ∎

A natural direction in which to extend the above is to throw in $\parr$ along with $\otimes$. This seems considerably harder.

Another direction is to generalize the limited results above about unbalanced formulas to arbitrary unbalanced formulas, the goal being to show that these contain dinaturals. This seems easier but we have not had time to investigate it. We strongly conjecture the stronger result that, with all variables assigned the above-mentioned "quad," namely the integers mod 4 with the consecutive-pair topology (open sets $\{0,1\}$, $\{1,2\}$, $\{2,3\}$, $\{3,0\}$), no element of the denotation of an unbalanced formula can be invariant under all eight automorphisms of the quad (which form the evident dihedral group on the square). This should not be hard to prove.

9.2 The only natural Church numeral

In **Set** the Church numeral for n is defined as the transformation from $A \multimap A$ to itself that takes $f : A \to A$ to $f^n : A \to A$, the n-fold composition of f with itself. It is readily shown that every Church numeral is dinatural. Furthermore that argument shows that every Church numeral in any closed category, if it exists, is dinatural. What remains to determine is which Church numerals exist in that category. The difficulty is that elements of $A \to A$, assuming they are even defined, might not compose with themselves. And indeed this can be shown, illustrated here for just the Church numeral 2.

Take p to be the Chu space $(\{0,1\}, \{(0,x),(1,y)\}, \{x,y\})$. This has only two endomorphisms, identity and twist, both of which when squared are the identity. $p \multimap p$ is isomorphic to itself, whence the Church numeral 2 on it is a constant function, which is not among the endomorphisms of p. Hence the Church numeral 2 does not exist at this p and *a fortiori* is not a dinatural transformation in **Chu**.

From the standpoint of **Chu** as a universal category, representative of the whole of "mathematical nature," those Church numerals that do exist in **Chu** can be regarded as existing "in nature." In this sense mathematics contains only one natural Church numeral, namely 1.

The linear λ-calculus is a restriction of the simply-typed λ-calculus requiring every λ-bound variable λa to match up with exactly one bound occurrence of a. The one linear Church numeral is 1; as noted above it is also the only Church numeral in **Chu**.

This leads to the question, which terms of the simply-typed λ-calculus denote transformations of the functors in **Chu** corresponding to those types? An appealing conjecture is that exactly the linear λ-terms so denote.

Acknowledgments. This work grew out of a summer collaboration on this problem with Gordon Plotkin and Richard Blute.

References

[1] Samson Abramsky and Radha Jagadeesan. Games and full completeness for multiplicative linear logic. *Journal of Symbolic Logic*, 59(2):543–574, 1994.

[2] M. Barr. *$*$-Autonomous categories*, volume 752 of *Lecture Notes in Mathematics*. Springer-Verlag, 1979.

[3] R.F. Blute and P.J. Scott. Linear Läuchli semantics. *Annals of Pure and Applied Logic*, 77:101–142, 1996.

[4] R.F. Blute and P.J. Scott. The shuffle Hopf algebra and noncommutative full completeness. *Submitted to J. Symbolic Logic*, 1996.

[5] V. Danos and L. Regnier. The structure of multiplicatives. *Archive for Mathematical Logic*, 28:181–203, 1989.

[6] J.-Y. Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.

[7] V. Gupta. *Chu Spaces: A Model of Concurrency.* PhD thesis, Stanford University, September 1994. Tech. Report, available as ftp://boole.stanford.edu/pub/gupthes.ps.Z.

[8] V. Gupta and V.R. Pratt. Gates accept concurrent behavior. In *Proc. 34th Ann. IEEE Symp. on Foundations of Comp. Sci.*, pages 62–71, November 1993.

[9] J.M.E. Hyland and C.-H.L. Ong. Fair games and full completeness for multiplicative linear logic without the mix-rule. Available by ftp from ftp.comlab.ox.ac.uk as fcomplete.ps.gz in /pub/Documents/techpapers/Luke.Ong, 1993.

[10] Y. Lafont and T. Streicher. Games semantics for linear logic. In *Proc. 6th Annual IEEE Symp. on Logic in Computer Science*, pages 43–49, Amsterdam, July 1991.

[11] J. Lambek and P. Scott. *Introduction to Higher-Order Categorical Logic.* Cambridge University Press, 1986.

[12] S. Mac Lane. *Categories for the Working Mathematician.* Springer-Verlag, 1971.

[13] V.R. Pratt. Chu spaces and their interpretation as concurrent objects. In J. van Leeuwen, editor, *Computer Science Today: Recent Trends and Developments*, volume 1000 of *Lecture Notes in Computer Science*, pages 392–405. Springer-Verlag, 1995.

[14] V.R. Pratt. The Stone gamut: A coordinatization of mathematics. In *Logic in Computer Science*, pages 444–454. IEEE Computer Society, June 1995.

[15] V.R. Pratt. Broadening the denotational semantics of linear logic. In *Special Issue on the Linear Logic 96*, volume 3 of *ENTCS (Electronic Notes of Theoretical Computer Science)*, Tokyo, 1996.