Some Semantical Aspects of Linear Logic

Andreas Blass*
Mathematics Department
University of Michigan
Ann Arbor, MI 48109–1109, U.S.A.
ablass@@umich.edu

April 20, 2002

Abstract

We describe and discuss several semantical views of linear logic. Our primary topic is game semantics, including modifications suggested by Abramsky, Jagadeesan, Hyland, Ong, and Japaridze. We also briefly discuss Girard's coherence spaces and de Paiva's Dialectica-like semantics.

1 Introduction

The WoLLIC talk on which this paper is based consisted of some introductory material on linear logic and several observations about its semantics, with particular emphasis on game semantics. In this paper, I shall not repeat all the introductory material, because it is available in Girard's original paper on the subject [12] or in my previous paper [5]. On the other hand, I shall include here some observations for which I didn't have time in the talk.

The two essential ideas at the basis of linear logic are resource-consciousness and linear negation. The former refers to keeping track of how often a hypothesis is used in deriving a conclusion. The latter is a duality which,

*Partially supported by NSF grant DMS–9505118. I thank the WoLLIC organizers both for inviting me to participate and for arranging a very enjoyable conference.

although related to the classical true-false duality, is sufficiently different to require (in my opinion) interaction as an underlying intuition.

This paper is organized as follows. In Section 2, we describe, much as in Girard's original presentation [12], the effect of resource-consciousness on logic, particularly the distinction between additive and multiplicative conjunction and the "of course" modality. We include here some comments about possible variations in the interpretation of this modality and about the intuitions underlying resource-consciousness. In Section 3, we indicate, following [5], how the intuition behind the additive connectives naturally leads to an interactive interpretation of (at least part of) the logic, i.e., a game semantics. With this as a basis, it becomes rather easy to interpret negation (in Section 4). Sections 5 through 10 are concerned with the problem that determinacy of games poses for game semantics: it threatens to reduce the logic to classical logic. In these sections I describe and comment on several approaches to avoiding this problem. The final two sections are about connections between game semantics and two other semantical systems for linear logic, namely de Paiva's Dialectica-like semantics [9] and Girard's coherence spaces [12].

Throughout this paper, we shall consider only *propositional* linear logic. Much of what we say can, however, be applied to first-order linear logic by regarding the quantifiers as analogous to additive connectives.

Words like "intuitively" will frequently refer to my own intuitions. I hope, of course, that the reader's intuition agrees with mine.

2 Resource-Consciousness

In linear logic, a sequent $\Gamma \vdash A$, where Γ is a finite sequence of formulas and A a single formula, is interpreted to mean that A can be deduced from Γ using each of the hypotheses in Γ exactly once. (Linear logic also allows sequents with several formulas on the right of $\vdash$, but an explanation of their meaning depends on linear negation; see Section 4.) Thus, the rules of contraction

$$\frac{\Gamma, A, A \vdash B}{\Gamma, A \vdash B}$$

and weakening (or thinning)

$$\frac{\Gamma \vdash B}{\Gamma, A \vdash B}$$

are not admissible in linear logic. A related system, *affine logic*, interprets $\Gamma \vdash A$ to mean that A can be deduced from Γ using each of the hypotheses in Γ at most once. Thus, the rule of weakening is valid in affine logic, but contraction is not.

It follows from the resource-consciousness of linear logic that there are two possible meanings for conjunction, according to whether "one use" of $A \wedge B$ means one use of each conjunct or one use of only one conjunct (with the user allowed to choose the conjunct — that's why it's a conjunction rather than a disjunction). The former, for which Girard introduced the notation $A \otimes B$, has the introduction rule

$$\frac{\Gamma \vdash A \qquad \Delta \vdash B}{\Gamma, \Delta \vdash A \otimes B}$$

while the latter, $A \& B$ in Girard's notation, has the introduction rule

$$\frac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \& B}.$$

The idea is that, in order to deduce the "both conjuncts" version, $A \otimes B$, one needs enough hypotheses to deduce both of A and B separately, while to deduce the "user's choice of one conjunct" version, $A \& B$, one needs only enough hypotheses to deduce, at the user's demand, either one of the conjuncts. The two introduction rules for conjunction would be equivalent in the presence of weakening and contraction, but in linear logic they are genuinely different.

The part of linear logic described so far is sufficient to suggest why it is difficult to produce a natural semantics for the logic. The concept of resource-consciousness was described in explicitly proof-oriented terms, involving the number of uses of hypotheses in a deduction. It is not clear how one could introduce resource-consciousness into classical semantical notions. What would it mean for a sentence A to be a semantical consequence of a list of sentences Γ "used exactly once each"? Semantical consequence in the usual sense simply doesn't "use" hypotheses as proofs do. This difficulty is reflected in the fact that, although linear logic has an elegant, well-developed proof theory (including, for example, Girard's theory of proof nets [12]) its semantics is still in an experimental stage, with different approaches capturing different aspects of linear (or affine) logic but none (in my opinion) doing justice to the whole system and its underlying intuitions.

The intuitive semantical situation improves somewhat if we apply the Curry-Howard isomorphism [18], also known as the propositions-as-types paradigm. Here, the formulas of a logical system are interpreted not as statements (or truth values thereof) but as types or (equivalently for our purposes) sets. Implication $A \to B$ is interpreted as the set of functions from A to B, conjunction as cartesian product, and disjunction as disjoint union. This is closely related to the Brouwer-Heyting interpretation of the logical connectives in intuitionistic logic [10, 26]; indeed, the two are identified if we identify a proposition with the set of its proofs. In the type-theoretic context, resource-consciousness means keeping track of how often a function uses each of its inputs in producing its output. This may not make much sense for functions in the classical sense, but it seems reasonably intuitive for algorithms.

Linear logic comes even closer to intuition if, following Girard [12], we think of it as being about actions or abilities. Girard presents an example where A represents having a dollar and B represents having a pack of cigarettes (whose price is a dollar); then $A \vdash B$ describes the process of buying a pack of cigarettes. The resource A is used up in obtaining B. $B \otimes B$ represents having two packs of cigarettes, so $A, A \vdash B \otimes B$ represents buying two packs, using up two dollars, but the result of contraction, $A \vdash B \otimes B$ is the (generally) impossible process of getting two packs for the price of one.

It may seem that this consideration of abilities and actions has taken us quite far afield from the ordinary propositional interpretation of sentential logic. There is, however, an interesting connection in the area of interactive proofs. In this connection a proposition corresponds to the ability to exhibit a proof of it. We describe the idea in a specific, easy example.

Suppose G_0 and G_1 are two (finite) graphs, known (in the form of adjacency tables, say) to two people traditionally called Prover and Verifier. Suppose further that Prover knows an isomorphism between these graphs but Verifier does not. Finally suppose that the size of the graphs is such that it is feasible for Verifier to check any given isomorphism between them but it is not feasible for Verifier to find an isomorphism if none is given. If Prover wants to convince Verifier that the graphs are isomorphic (and that he, Prover, knows an isomorphism), the obvious protocol would be for Prover simply to exhibit the isomorphism that he knows. Things get more interesting if, in addition, Prover does not want Verifier to learn the isomorphism. Here is a (well known) protocol for this situation. First, Prover randomly produces an isomorphic copy G' of G_0 (by relabeling its vertices at random) and shows it (i.e., its adjacency table) to Verifier. Then Verifier randomly selects a number $i \in \{0, 1\}$ and tells it to Prover. Finally, Prover must show Verifier an isomorphism between G_i and G'. If Prover is telling the truth, i.e., if he knows an isomorphism between G_0 and G_1, then he also knows isomorphisms between each of these and G'. (The isomorphism $G_0 \cong G'$ comes from Prover's construction of G', and the isomorphism $G_1 \cong G'$ comes from composition with the known $G_1 \cong G_0$.) So a truthful Prover can always successfully complete the protocol. A dishonest Prover (who doesn't know any isomorphism $G_1 \cong G_0$) can produce G' with a known isomorphism to one G_j but not the other; he can then complete the protocol only if Verifier happens to choose $i = j$; i.e., Prover succeeds with probability $1/2$ (if Verifier picks i randomly as he should). If the protocol is repeated many times (with independent, random choices of G' and i at every step), then the probability that a dishonest Prover can cheat Verifier by successfully completing the protocol every time rapidly approaches zero. Thus, successful completion of the protocol, say, 100 times is a reasonable way for Prover to convince Verifier that he knows an isomorphism $G_1 \cong G_0$.

In terms of the three interpretations of propositional logic mentioned above — using statements, types, and actions (or abilities) — we can make the following commentary on this protocol. From the statement point of view, "$G_0 \cong G_1$" means simply that these graphs are isomorphic. From the type point of view, "$G_0 \cong G_1$" represents the set of isomorphisms, which one might regard as the set of proofs of the statement in the preceding sentence. Finally, from the action point of view, "$G_0 \cong G_1$" represents Prover's exhibiting an isomorphism to Verifier. In the protocol above, Prover performs actions represented by $(G_0 \cong G') \,\&\, (G_1 \cong G')$; that is, he exhibits an isomorphism from G_i to G', where i is not under his own control (being selected by Verifier). On the basis of this evidence, Verifier becomes convinced that Prover also has the ability to perform the action $(G_0 \cong G') \otimes (G_1 \cong G')$, i.e., to exhibit both isomorphisms. From that it follows, by composition of isomorphisms, that Prover could perform the action $G_0 \cong G_1$, which is what Prover wanted to demonstrate.

Verifier's conviction, after successful completion of the protocol, that Prover knows an isomorphism $G_0 \cong G_1$ is thus based on a step from $(G_0 \cong G') \,\&\, (G_1 \cong G')$ to the other kind of conjunction $(G_0 \cong G') \otimes (G_1 \cong G')$. Why is that step legitimate here, when it is clearly illegitimate in the cigarette-buying example? The answer is (I believe) that the abilities demonstrated by Prover or inferred by Verifier are based entirely on *knowledge* — Prover's knowledge of certain isomorphisms. Unlike dollars (in the cigarette example), knowledge is not lost when used, and so & is equivalent in this context to $\otimes$.

On the basis of such examples, I believe that linear logic and in particular the distinction between & and $\otimes$ are relevant directly to the analysis of actions and abilities, not to the analysis of knowledge and truth. Linear logic acquires an indirect relevance to the latter when one considers, as in the protocol above, connections between knowledge and actions.

Linear logic has a mechanism, the modal operator ! (read "of course") for handling resources like knowledge that are not destroyed by being used. For any formula A, the formula $!A$ means a form of A that can be used any number of times (including 0 times) rather than just once. Thus, the rules of contraction and weakening are admitted when the formula being contracted or introduced by weakening has the form $!A$. In the formulas-as-types interpretation, an object of type $!A$ should be viewed as an object of type A stored in such a way that it can be repeatedly accessed by a computation. In the action interpretation, the ability $!A$ means the ability to do A repeatedly.

I believe that the preceding description accurately captures Girard's intention in introducing the "of course" modality, but a somewhat different interpretation of ! (different at least from the type-theoretic point of view) also occurs implicitly in [12]. It is suggested there that $!A$ can be approximated by

$$(1 \mathbin{\&} A) \otimes (1 \mathbin{\&} A) \otimes (1 \mathbin{\&} A) \otimes \dots. \tag{1}$$

Here 1 represents a singleton type, say $\{*\}$; it is present to allow non-use of A. So this approximation (1) means the type of sequences in which each term is either $*$ or an object of type A. Thus, it can, like a stored element of type A, be accessed repeatedly, but there is no guarantee that each access produces the same element. That is, what (1) represents is more like a stream of elements of A than like a single, stored element of A. (The stream description seems more natural for the cigarette-buying example than for computational interpretations involving stored data. If A represents having a dollar, then $!A$ could represent having an unlimited supply of dollars, rather than having one, arbitrarily reusable dollar.)

Girard [13] mentions that the proof rules for ! do not uniquely determine this modality. That is, if one added to linear logic a second modal operator

$!'$ subject to the same inference rules as $!$, one could not deduce that the two operators are equivalent in the sense that $!A \vdash !'A$ and vice versa. The preceding comments indicate that in fact this multiplicity of !'s has been realized, at least on an intuitive level, since the beginning of linear logic.

3 Interaction and Game Semantics

The description above of the intended meaning of the additive[1] conjunction & implicitly involves interaction. In the type interpretation, an element of $A \& B$ should be an element of A or an element of B, the choice between these alternatives being made by a user. Again, in the action interpretation, to be able to perform the action $A \& B$, I must be able to perform either of the actions A and B as demanded by someone else.

By contrast, the ability to choose for oneself one of the actions A and B and then perform it is represented by the additive disjunction $A \oplus B$.

In order to formalize this distinction, it is natural to consider two-party protocols, where one party is carrying out the action represented by a formula while the other makes the external choices required for &. There are several ways to think about this and correspondingly several terminologies.

In [4], I described a viewpoint oriented toward proving propositions. The two participants in a protocol (or dialog or game) were a proponent and an opponent; the former seeks to establish the truth of the formula under consideration, the latter to establish its falsity, and the heart of the semantics consists of rules for the debate between these two. The idea of using two-party dialogs to explain the meanings of logical connectives goes back to Lorenzen [22], and the particular operations on games, which interpret the connectives in [4], go back to [3].

In [5], I described a very similar semantics based on a type-theoretic viewpoint. The intuition here is that a formula A represents a server capable of providing, to some user, elements of type A. If A is of the form $A_1 \& A_2$ then before the server can provide an appropriate element, the user must specify which A_i he wants an element from. For $A_1 \oplus A_2$, on the other

[1]Girard's terminology "additive" for the connectives & and $\oplus$ and "multiplicative" for $\otimes$ and $\parr$ is derived from the interpretations of these connectives in coherence spaces; see [12] or Section 12 below. It also describes the behavior of the connectives in the distributive laws provable in linear logic; $A \otimes (B \oplus C)$ is linearly equivalent to $(A \otimes B) \oplus (A \otimes C)$, while $A \parr (B \& C)$ is linearly equivalent to $(A \& B) \parr (A \& C)$.

hand, the server may choose arbitrarily which A_i to provide an element of. Compound formulas like

$$(A \mathbin{\&} B) \oplus (C \mathbin{\&} D)$$

require a slightly longer interaction between server and user. The server must tell the user which side of the $\oplus$ it intends to provide an element of, so that the user can choose one side of the appropriate &, after which the server finally provides the required element (unless of course A, B, C, and D contain more additive connectives, in which case an even longer protocol results). Thus, consideration of the additive connectives automatically leads to a semantics with at least some game-like aspects. (The terminology of "client" and "server" was suggested to me by Dexter Kozen, after a lecture in which I had used "user" and "database.")

The proponent in the first of these pictures corresponds to the server in the second. The opponent in the first corresponds to the client in the second. And the proponent's winning a play of the game corresponds to the server's successfully completing a run of the protocol, i.e., the server's never lacking a response when the protocol requires it to give one.

It should be emphasized that the semantics described in [4] and [5], though similar in many respects, are not identical. For example, the former makes the rule of weakening valid while the latter does not. More importantly, the former avoids the determinacy problems described in Section 5 below (see Section 6), while the latter needs to be modified, perhaps using some form of uniformity (see Section 7).

Nevertheless, most of what we say below is applicable to both versions; it will be convenient to use primarily the terminology of games between a proponent and an opponent.

The inference above, from the intended meaning of additive connectives to a game interpretation of formulas, shows us how to interpret the additive connectives as operations on games. Game $A \mathbin{\&} B$ is played by first letting the opponent choose one of A and B and then playing the chosen game. $A \oplus B$ is the same except that the initial choice of A or B is made by the proponent.

The proper interpretation of multiplicative connectives is less clear. I confine myself here to describing the interpretation of $\otimes$ given in [4, 5] (or more precisely the common part of these two interpretations); an extensive motivational discussion is given in [5]. A play of $A \otimes B$ consists of interleaved runs of the two constituents A and B. Whenever it is the proponent's turn to move in one or both constituents, he must move there. When it is the opponent's turn to move in both constituents, he must choose one and move in it. In versions of game semantics where the criterion for winning must be specified (as in [4]), the opponent wins a play of $A \otimes B$ if and only if he finishes and wins at least one of the two subgames, and otherwise the proponent wins.

The interpretation of the "of course" modality suggested in [4] is similar to a repeated $\otimes$. A play of $!A$ consists of numerous plays of A interleaved, with the opponent having the right to start new plays of A or resume previously abandoned ones (at the position where they were abandoned). To win, the opponent must finish and win at least one of the subgames. Furthermore, it was required in [4] that as long as the opponent makes the same moves in two of these plays of A so does the proponent. This last requirement can be omitted without damaging the soundness of the interpretation. Imposing the requirement corresponds intuitively to the "stored value" interpretation of !, and omitting it corresponds to the "stream" intrpretation, as discussed at the end of Section 2.

4 Linear Negation

Game semantics makes it easy to define linear negation or duality: $A^\perp$ is the game A with the roles of proponent and opponent reversed. This operation is very natural from the point of view of debates about the correctness of propositions. It is considerably less so from the point of view of clients and servers or from the point of view of abilities or actions. Recalling that resource-consciousness was more natural from the latter viewpoints and less natural from the former, I consider it quite remarkable that Girard was able to combine the two notions, resource-consciousness and duality, in a single, coherent logical system. I also believe that much of the difficulty in producing a good semantics for linear logic is caused by the need to switch from one viewpoint to another in order to get a clear picture of the basic intuitions underlying the logic.

Once linear negation is available, it can be used to dualize the other connectives. Following the notation of [12], we have

$$A \parr B = (A^\perp \otimes B^\perp)^\perp, \quad A \oplus B = (A^\perp \& B^\perp)^\perp, \quad ?A = (!(A^\perp))^\perp.$$

Although $\oplus$ seems quite a reasonable disjunction, both $\invamp$ and ? are rather mysterious. In particular, it is argued in [5] that one should resist the temptation to say that $\invamp$, being dual to a version of conjunction, is simply a version of disjunction; in the interpretation presented in [5], it also has a definite conjunctive character.

Linear implication, $A \multimap B$, is defined as $A^{\perp} \invamp B$, i.e., $(A \otimes B^{\perp})^{\perp}$ (where I've used that $A^{\perp\perp} = A$). Untangling the definitions, we find the following description of the game $A \multimap B$. The players play A interleaved with B, with the players' roles being reversed in A. The opponent must move in one or both components whenever it is his turn in at least one component, but when it is the proponent's turn to move in both components he must move in one of them. This sort of compound game was introduced in [3] (with no thought of linear logic, which had yet to be invented) as one natural way for the proponent to defend the claim "I could win B if I were shown how to win A." In effect, the players are playing B but the proponent may, whenever he wishes, temporarily suspend the play of B to consult an expert oracle about how to play A. The consultation consists of the proponent acting as the opponent in a play of A while the oracle, acting as proponent in that play, shows him how to win A. This description in terms of oracles is essentially the same as the previous description of $A \multimap B$, except that in the latter it is the opponent, not an oracle, that acts as proponent in the play of A. That difference has no effect as far as winning strategies for the proponent are concerned.

To complete a comment left hanging earlier, I should mention that, if several formulas occur on the right side of a sequent, the commas between then have the semantic value of $\invamp$. Commas on the left amount to $\otimes$, and the sequent symbol $\vdash$ amounts to $\multimap$. It follows that $\Gamma \vdash \Delta$ is semantically equivalent to the right-sided sequent $\vdash \Gamma^{\perp}, \Delta$, where $\Gamma^{\perp}$ is the result of replacing every formula A in Γ with $A^{\perp}$. Because of this observation, one can present linear (or affine) logic as a calculus of right-sided sequents only; this presentation is often convenient because it requires fewer rules.

For future reference, we indicate the reason for the game validity of the axiom $\vdash A^{\perp}, A$ and the cut rule

$$\frac{\vdash \Gamma, A \qquad \vdash \Delta, A^{\perp}}{\vdash \Gamma, \Delta}.$$

For the former, consider any game A and the same game with the players' roles reversed, $A^{\perp}$. The proponent can win the compound game $A \invamp A^{\perp}$ by a copying strategy. One of the two constituents A and $A^\perp$ begins with the opponent to move, so the opponent must move there in the $⅋$ combination. The proponent then copies this move into the other constituent, and the opponent must reply there. The proponent copies that reply back into the first constituent, etc.

For the cut rule, suppose we have winning strategies σ and τ for the proponent in games $\Gamma ⅋ A$ and $\Delta ⅋ A^\perp$, respectively. (We've written Γ for the $⅋$ combination of the games corresponding to the formulas in the sequence Γ, and similarly for Δ.) Then here is a winning strategy for the proponent in $\Gamma ⅋ \Delta$. Pretend that in addition to the actual plays of Γ and Δ, you are also playing an imaginary game of A. Use σ to produce moves in the actual Γ and moves of the proponent in the imaginary A. Use τ to produce moves in the actual Δ and moves of the opponent in the imaginary A (i.e., moves of the proponent in $A^\perp$). Thus σ and τ play against the actual opponent in Γ and Δ and against each other in the imaginary A. For reasons that vary from one version of game semantics to another, this process cannot get stuck in the imaginary A (i.e., we don't get infinite internal chatter between σ and τ) and the result is a play of $\Gamma ⅋ \Delta$ won by the proponent.

5 Determinacy

A classical theorem of Zermelo [27] (see also [11]) asserts that if a game (of perfect information, between two players, in which every play is a win for exactly one player) is such that it always ends in a finite number of moves, then the game is *determined* in the sense that one of the players has a winning strategy. This theorem poses a serious problem for game interpretations of linear logic, namely, it tends to make them interpretations of classical logic instead.

Specifically, define a map v from finite games (i.e., games where each play has finite length) to classical truth values by letting $v(G)$ be true if the proponent has a winning strategy in G and false if the opponent does. Zermelo's theorem ensures that v is well defined. It is easy to check that the operations on games that interpret the linear connectives $\&$, $\oplus$, $\otimes$, $⅋$, ${}^\perp$, and $\multimap$ correspond via v to the classical connectives $\wedge$, $\vee$, $\wedge$, $\vee$, $\neg$, and $\Rightarrow$, respectively, in the sense that $v(G \,\&\, H) = v(G) \wedge v(H)$, etc. It follows that, if A is a formula of linear logic and if its "translation" into classical logic (using the correspondence of connectives just listed) is a tautology, then A is finite-game valid in the sense that, if we interpret atomic formulas arbitrarily as finite games then the resulting interpretation of A is a win for the proponent. In particular, this notion of validity obliterates the distinction between & and $\otimes$. In effect, it has departed from linear logic and merely describes (in an unnecessarily complicated way) validity in classical propositional logic.

The next few sections are devoted to describing and commenting on some ways to avoid this determinacy problem.

6 Infinite Games

A straightforward way to avoid the difficulty resulting from Zermelo's theorem is to violate the hypothesis of that theorem by allowing games to have infinitely long plays. This is the approach used in [4]. Its feasibility rests on the existence of undetermined games when infinite plays are allowed; such games were constructed (using the axiom of choice) in [11]. Here the criterion for winning can no longer be merely successful completion of the protocol (lest the game be determined again, by another result from [11]) but must be given as part of the data of the game. In fact, to get an undetermined game, the criterion for winning must be quite complicated. (For games with only countably many options at each move, some form of the axiom of choice is needed to obtain undetermined games; with $\aleph_1$ options per move, the axiom of choice is not required but considerable complexity is.)

Such an infinitary game semantics is rather removed from the ideas of client-server protocols which I regard as the most natural justification for game semantics. Certainly the protocols arising, as described in Section 3, from the additive connectives give no hint of infinity. Infinity can arise from interpreting the "of course" modality (see [5]) but in a very limited way; that is, one gets infinite games with very simple criteria for winning, entirely inadequate to avoid determinacy.

Infinitary games may seem a bit more reasonable from the dialog point of view. Some debates do seem interminable. But it must be admitted that the motivation for introducing infinity in [4] was to have an ample supply of undetermined games.

The ampleness of the supply is crucial for the main result of [4], namely the completeness of this semantics for the additive fragment of affine logic. ("Additive fragment" means that formulas involve only the connectives &, $\oplus$, and $^\perp$; note, however, that $\parr$ is also involved implicitly as the interpretation of the commas in a right-sided sequent.) It is also crucial for the characterization in [4] of the valid sequents of the multiplicative fragment. In both cases, one needs to interpret the atomic formulas with games that are not only undetermined but so independent of each other that there is no strategy for winning one with the help of oracles for others; that is, the proponent should not have a winning strategy for, say, $(A \otimes B) \multimap C$, where A, B, and C are distinct atomic formulas. More complicated independence properties are needed to make more complicated formulas unprovable, and much of the technical work in [4] is directed toward constructing suitably independent games.

7 Uniform Strategies

Abramsky and Jagadeesan [1] introduced several modifications of game semantics in order to obtain a multplicative completeness theorem. They show that a multiplicative sequent is valid in their game semantics if and only if it is provable in linear logic augmented with the MIX rule, which consists of the binary rule

$$\frac{\vdash \Gamma \qquad \vdash \Delta}{\vdash \Gamma, \Delta}$$

and the nullary rule (or axiom) $\vdash$. The binary MIX rule is a trivial consequence of weakening, so its inclusion produces a system intermediate between linear and affine logic. The nullary MIX axiom is less intuitive from the propositional point of view (since an empty disjunction ought to be false), but it makes more sense from a computational point of view, especially if one believes (as in [5]) that $\parr$ has some conjunctive aspects.

This section and the two following ones are devoted discussing the three modifications involved in obtaining the completeness result. It turns out that each of the three is sufficient by itself to avoid the determinacy problem.

It should be mentioned here that Abramsky and Jagadeesan got a stronger result than mere completeness. Not only does the existence of winning strategies (in their sense) imply provability in multiplicative linear logic plus MIX, but every strategy comes from a proof. This property is called *full completeness.* It should also be mentioned that Hyland and Ong [19] subsequently found a further modification that eliminates the need for adding MIX to the logic.

The first of Abramsky's and Jagadeesan's modifications of game semantics is a uniformity requirement on strategies. Consider a sequent Γ and its various game interpretations, obtained by interpreting the atomic formulas occurring in Γ as games. In [4], game-validity of Γ was defined as the existence of winning strategies for all these game interpretations of Γ. In [1], the definition is made more restrictive by requiring that these strategies fit together in a coherent way. Specifically, if two interpretations v and w are such that, for each atomic formula, $v(A)$ is obtained from $w(A)$ by removing some moves, then the strategy assigned to $v(\Gamma)$ is to be the restriction of the strategy assigned to $w(\Gamma)$.

This uniformity requirement solves the determinacy problem because, although each finite game has a winning strategy for one of the players, those strategies are by no means uniform. For example, the proponent's winning strategy in $A \oplus A^{\perp}$ is to completely analyze the finite game A and then, as his first move, choose A or $A^{\perp}$ according to whether A is a win for the proponent or the opponent. This is certainly not uniform, for a change in the interpretation of A by adding or deleting moves can easily change who has the winning strategy.

If one thinks of strategies for ordinary games, like chess, this uniformity requirement seems strange. If one changed chess by adding new sorts of moves, then one would not expect good strategies for standard chess to be restrictions of good strategies for the extension. An old move that was good in the old version of the game may become very bad in the extension if one of the new moves is a brilliant reply to it. But of course Abramsky and Jagadeesan were working in a different context, not seeking to describe good strategies for typical games but strategies corresponding to proofs in linear logic. And for this purpose uniformity works well, largely because (as a bit of reflection on the chess example will show) it prevents the strategy from taking into account the particular games used to interpret atomic formulas. Instead, such uniform strategies must rely on the logical structure of the sequent $\vdash \Gamma$. Indeed, the combination of uniformity and history-freeness (discussed in Section 8) forces strategies to rely simply on copying the opponent's moves from one subgame to another. This limitation on the possible strategies is a crucial ingredient in the completeness proof of [1]. Indeed, such a limitation is necessary in order to get a full completeness theorem, since only such copying strategies arise from proofs in linear logic plus MIX.

There is, however, one piece of information that is lost from completeness theorems when uniformity is assumed, namely uniformity itself. What

I have in mind is perhaps clearest if expressed in terms of Gödel's familiar completeness theorem for classical predicate calculus. Part of the intuitive content of this theorem is that a sentence cannot achieve validity by "just happening" to hold in all structures, perhaps for totally different reasons in different structures. The completeness theorem implies that, if a sentence holds in all structures, then there is a single reason for it, expressed by its proof and applicable to all structures at once. Thus, a sort of uniformity of reasons is part of the conclusion of the theorem. Something similar happens in completeness proofs for linear logic that don't impose a uniformity requirement on the strategies, for example, the completeness theorem in [4] for the additive fragment of affine logic. That proof, along with the easier soundness proof, shows that, if all the games associated to an additive sequent $\vdash \Gamma$ (by various interpretations of atomic formulas as games) have winning strategies for the proponent (possibly very different strategies for different games, strategies that depend sensitively on the nature of the games), then they all have a uniform winning strategy for the proponent, involving essentially only copying. The reason is that, in this situation, the completeness theorem says that $\vdash \Gamma$ is provable, and the proof of the soundness theorem associates to every proof of a sequent a uniform, copying strategy by which the proponent can win all its instances. This sort of additional information is, of course, not available if uniformity is assumed in advance as a restriction on the admissible strategies. That appears to be the price one must pay for full completeness.

8 History-Free Strategies

The second of Abramsky's and Jagadeesan's modifications of game semantics is to require the proponent's winning strategies to be history-free. That is, each move prescribed by the strategy should depend only on the opponent's immediately preceding move, not on the entire previous history of the play. As mentioned above, it is shown in [1] that this requirement and uniformity together force the proponent's winning strategies to consist of just copying the opponent's moves from other components of the game.

History-freeness solves the determinacy problem simply because finite games do not in general have history-free winning strategies. For an easy example, consider a game consisting of four moves where the opponent chooses natural numbers on moves 1 and 3, the proponent chooses natural numbers on moves 2 and 4, and the proponent wins if and only if his choice at move 4 agrees with the opponent's choice at move 1. Here the proponent has a trivial winning strategy, but none that is history-free.

It is mentioned in [1] that history-freeness does not work well with the additive connectives. Here is an example of what goes wrong. Consider the introduction rule for &,

$$\frac{\vdash \Gamma, A \qquad \vdash \Gamma, B}{\vdash \Gamma, A\&B},$$

suppose we have interpreted atomic formulas as games, and suppose we have winning, history-free strategies, σ and τ, for the proponent in $\Gamma ⅋ A$ and $\Gamma ⅋ B$, respectively. How should the proponent win $\Gamma ⅋ (A\&B)$? By the game interpretation of ⅋, the opponent, who has the first move in $A\&B$ namely to choose A or B, must make this choice before the proponent needs to do anything. So by the time the proponent is expected to move, he is essentially playing $\Gamma ⅋ A$ or $\Gamma ⅋ B$, so he can use σ or τ. The difficulty is that he must remember which of σ and τ he is using. If either strategy involves some moves in the Γ components, then the proponent cannot, in general, tell from the opponent's immediately preceding move (which might be in Γ), whether he is playing $\Gamma ⅋ A$ or $\Gamma ⅋ B$.

9 Pairs of Games

The last of Abramsky's and Jagadeesan's modifications of game semantics is that their games do not prescribe which player is to move first; in general either player can start a game. Furthermore, there is no necessary connection between the rules of a game when the proponent starts and the rules of the same game when the opponent starts. In effect, therefore, a game in this sense amounts to two games in the older sense, one in which the proponent starts and one in which the opponent starts.

This arrangement defeats determinacy because the two constituent games may have winning strategies for different players. Like history-freeness, it does not fit naturally with additive connectives, since & and $\oplus$ naturally call for a first move by the opponent and proponent, respectively. This leads to the introduction of polar games in Section 5 of [1], games where only one player can start; depending on which player starts, one gets a nice interpretation of one of the additive connectives and a not quite so nice interpretation of the other (involving a "dummy" move by the starting player before the other player chooses a constituent subgame).

Although it was not, as far as I know, intended to be used in this way by Abramsky and Jagadeesan, it may be informative to consider the effect of making only this third modification of game semantics, without uniformity or history-freeness. If we use only finite games, then, instead of a reduction to two truth values as described in Section 5, we get a reduction to four truth values

- T: The proponent has a winning strategy, no matter who starts.
- F: The opponent has a winning strategy, no matter who starts.
- H: The player who starts has a winning strategy.
- C: The player who doesn't start has a winning strategy.

T and F stand for true and false as with the v in Section 5, and H and C stand for "hot" and "cold," suggested by Conway's [8] temperature terminology for closely related (though not identical) concepts. A hot game is one where the players are eager to move; a cold game is one where each player would rather let the other move.

We use the Abramsky-Jagadeesan interpretations in [1] for the multiplicative connectives as well as their convention that truth corresponds to the existence of a winning strategy for the proponent when the opponent starts the game. Thus, the distinguished truth values in this system are T and C. In $A \otimes B$, the first player decides which component, A or B, the play begins in, thereafter only the opponent is allowed to switch from one component to the other; $A \parr B$ is the same except that only the proponent has the right to switch components after the opening move. Negation, of course, just interchanges the roles of the two players, and $A \multimap B$ is identified with

$A^\perp \invamp B$. These definitions lead to the following truth tables.

A	$A^\perp$
T	F
F	T
H	H
C	C

$\otimes$	T	F	H	C
T	T	F	H	T
F	F	F	F	F
H	H	F	F	H
C	T	F	H	C

$\invamp$	T	F	H	C
T	T	T	T	T
F	T	F	H	F
H	T	H	T	H
C	T	F	H	C

$\multimap$	T	F	H	C
T	T	F	H	F
F	T	T	T	T
H	T	H	T	H
C	T	F	H	C

Notice that negation interchanges T with F (as one would surely expect) but fixes the other two truth values. In partuclar, a formula and its negation can both have the distinguished truth value C. Notice also that $\otimes$ is just the minimum operation with respect to the ordering $F < H < T < C$ except that $H \otimes H = F$. Dually, $\invamp$ is the maximum operation with respect to the order $C < F < H < T$ (obtained from the previous order by reversal and negation) except that $H \invamp H = T$.

It is straightforward to verify that the standard axioms and rules of multiplicative linear logic are sound for this four-valued semantics. As in the game semantics of [1], weakening fails: $(H \otimes C) \multimap C$ has the non-distinguished truth value H. On the other hand, all instances of $(A \otimes A) \multimap A$ have distinguished truth values, namely C if A has value C and T in the other three cases. Thus this special case $(A \otimes A) \multimap A$ of weakening is valid. Finally, we note that the MIX rule (binary and nullary) is valid because the units for $\otimes$ and $\invamp$ are the same, namely C.

10 Computable Strategies

Japaridze [21] studied a version of game semantics in which truth is taken to mean existence of an *effective* winning strategy for the proponent. In more detail, games are required to be such that all plays are finitely long (though arbitrarily large finite lengths can occur in a game), the possible moves at each stage are (or can be coded as) natural numbers, and it is effectively decidable at each position who is to move next and whether any particular move is legal. By convention, a player loses just when he has no legal move. Games of this sort were considered by Rabin [24], who showed that they do not in general have effectively computable winning strategies. (In fact [2], one can construct games of this sort whose winning strategies all have complexity at least any prescribed level of the hyperarithmetic hierarchy.) Thus, Japaridze's requirement of effectivity for winning strategies circumvents the determinacy problem.

Among Japaridze's results are the soundness of affine logic (without the modalities ! and ?, which he does not interpret) for his semantics and completeness for the additive fragment. He also shows that completeness fails for the multiplicative fragment. In these respects, his semantics shares the general properties of the semantics in [4]. It would be interesting to understand the connections, if any, between Japaridze's recursion-theoretic arguments and the set-theoretic arguments for the analogous results in [4]. Because of its connection with recursion theory, Japaridze's approach seems the most likely version of game semantics to mediate applications of linear (or affine) logic to actual computing.

11 Dialectica-Like Semantics

In this section and the next, we briefly point out connections between game semantics and two other approaches to modeling linear logic.

The Dialectica-like model of de Paiva [9] is based on quite a general category, but we concentrate here on the case where that category is the category of sets. With this specialization, de Paiva's model interprets formulas as pairs of sets equipped with a binary relation between them, i.e., (A_-, A_+, A) where $A \subseteq A_- \times A_+$. These are the objects of a category, where a morphism from (A_-, A_+, A) to (B_-, B_+, B) is a pair of functions, $f_- : B_- \to A_-$ and $f_+ : A_+ \to B_+$, such that for all $b \in B_-$ and all $a \in A_+$,

$$A(f_-(b), a) \Longrightarrow B(b, f_+(a)).$$

Additive connectives $\oplus$ and & are interpreted as the sum and product in this category:

$$(A_-, A_+, A) \,\&\, (B_-, B_+, B) = (A_- \sqcup B_-, A_+ \times B_+, P)$$

and

$$(A_-, A_+, A) \oplus (B_-, B_+, B) = (A_- \times B_-, A_+ \sqcup B_+, S),$$

where $\sqcup$ represents disjoint union and where $xP(a,b)$ means xAa or xBb according to whether $x \in A_-$ or $x \in B_-$, and similarly for S. Negation replaces a relation with the complement of its converse, $(A_-, A_+, A)^\perp = (A_+, A_-, \neg\breve{A})$. (This is a contravariant involution of the category.) The multiplicative connectives are somewhat more complicated; we define here only $\otimes$, since $\parr$ and $\multimap$ can be expressed in terms of $\otimes$ and negation.

$$(A_-, A_+, A) \otimes (B_-, B_+, B) = (A_-^{B+} \times B_-^{A+}, A_+ \times B_+, T),$$

where the relation T holds between $(f,g) \in A_-^{B+} \times B_-^{A+}, A_+$ and $(a,b) \in A_+ \times B_+$ if and only if both $A(f(b),a)$ and $B(g(a),b)$ hold. Motivations for this definition can be found in [9] and in [6]. Notice (or, rather, calculate) that a morphism from A to B is the same as a member of $(A \multimap B)_+$ related to all members of $(A \multimap B)_-$ by the relation constituent of $A \multimap B$.

One way to relate objects of the form (A_-, A_+, A) to games is to regard them as two-move games, where A_- is the set of possible moves for the opponent, A_+ is the set of possible moves for the proponent, and $A(x,y)$ means that if the opponent chooses x and the proponent y then the proponent wins. Then de Paiva's negation corresponds to interchanging the players, but the rest of the correspondence is not so nice. In particular, in the definition of $\otimes$, the effect of the function sets is that the opponent's move in either component can depend on the proponent's move in the *other* component, which seems rather counterintuitive. There is also the problem that (A_-, A_+, A) contains no information about who is to move first in the associated game; if one attempts to fix a convention, say that the opponent always moves first, then negation no longer works nicely.

A better way to relate de Paiva's interpretation to games is to regard the elements of A_- and A_+ as strategies for the opponent and the proponent, respectively. $A(x,y)$ should mean that the proponent wins if he uses strategy y against the opponent's strategy x. Negation still corresponds to interchanging the two players, but now the additive connectives also correspond properly. Remembering that the game $A \mathbin{\&} B$ consists of a choice of A or B by the opponent followed by a play of the chosen game, we see that a strategy for the proponent in $A \mathbin{\&} B$ is essentially a pair of strategies, one for A and one for B. A strategy for the opponent in $A \mathbin{\&} B$ is a strategy in one of A and B (along with a specification of which game it's a strategy for). This matches precisely the facts that $(A\&B)_+ = A_+ \times B_+$ and $(A\&B)_- = A_- \sqcup B_-$, and it is easy to check that the relation parts also match properly.

The multiplicative connectives do not match so nicely, but this strategy interpretation does make de Paiva's interpretation of $\otimes$ somewhat more intuitive as follows. If we think of a play of $A \otimes B$ as plays of A and B in parallel, then a strategy for the opponent provides moves for him in each component, depending on the proponent's previous moves in both components. A strategy in A or B alone would provide moves in that component, depending on previous moves only in that component. The curious cross-dependence given by the function spaces in $A_{-}^{B_+} \times B_{-}^{A_+}$ may be viewed as a way of permitting moves in one component to depend on previous moves in the other. It goes, from the game point of view, a bit too far, by allowing what happens in one component to depend on the whole strategy used by the other player in the other component rather than only on the moves already played in the other component. And it allows such dependence only for the opponent in $\otimes$ (the proponent in $\parr$) — in game semantics each player is allowed to refer to both components, and the advantage of the opponent in $\otimes$ (the proponent in $\parr$) is rather the right to switch from one component to the other (and, in some versions, the fact that he need only win one component in order to win the compound game).

12 Coherence Spaces

One of the oldest semantical interpretations of linear logic, in fact part of the motivation for the original development of linear logic, uses coherence spaces [12]. A *coherence space*, also known to combinatorialists and group theorists as a *flag complex* [25], is a family $\mathcal{A}$ of sets closed under subsets and under "coherent unions," where the latter means that $\bigcup \mathcal{X} \in \mathcal{A}$ whenever $\mathcal{X} \subseteq \mathcal{A}$ and the union of every two members of $\mathcal{X}$ is in $\mathcal{A}$. Such a space can equivalently be described as the family of cliques (i.e., pairwise adjacent sets of nodes) in an undirected graph. The nodes of the graph are the members of the members of $\mathcal{A}$, and nodes x and y are adjacent if $\{x, y\} \in \mathcal{A}$.

Girard interpreted formulas of linear logic as coherence spaces, with the connectives being given by the following operations on the associated graphs. Negation leaves the set of nodes unchanged, but two distinct nodes are adjacent in $\mathcal{A}^{\perp}$ if and only if they are not adjacent in $\mathcal{A}$. For the additive connectives, one takes the disjoint union of the two sets of nodes and leaves the adjacency relation unchanged within each piece; two nodes from different pieces are adjacent in $\mathcal{A} \& \mathcal{B}$ but not in $\mathcal{A} \oplus \mathcal{B}$. The multiplicative conjunction

$\otimes$ is interpreted by taking the direct product of the two sets of nodes and declaring two distinct ordered pairs to be adjacent if they are componentwise adjacent-or-equal. (The interpretations of $\parr$ and $\multimap$ follow via negation, and we omit the modalities from the present discussion.)

We can connect games to coherence spaces as follows. Given a game, consider the graph whose vertices are all the possible plays of the game, two distinct plays being adjacent in the graph if the first difference between them is a move of the opponent. Any strategy σ for the proponent gives rise to a clique C_σ in this coherence space, namely the set of all plays in which the proponent follows the strategy σ. In fact, the C_σ's for all strategies σ are exactly the maximal cliques of this coherence space.

This transformation, from games to coherence spaces, commutes with negation, and it commutes with the additive connectives up to a canonical isomorphism. The correspondence is (as in the preceding section) not so nice for the multiplicative connectives; if two coherence spaces arise from games, their $\otimes$ product need not so arise. Intuitively, this product corresponds to playing two games in parallel with no interconnection between them, an activity that is not quite a (single) game.

The main discrepancy, however, in this correspondence between games and coherence spaces is that a game has a distinguished collection of strategies, namely the winning ones, while a coherence space does not have a similarly distinguished collection of (maximal) cliques. It is conceivable that adding such a collection, as additional structure, to coherence spaces may produce a useful semantics intermediate between coherence spaces and games.

References

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

[2] Andreas Blass. Complexity of winning strategies. *Discrete Math.*, 3:295–300, 1972.

[3] Andreas Blass. Degrees of indeterminacy of games. *Fund. Math.*, 77:151–166, 1972.

[4] Andreas Blass. A game semantics for linear logic. *Ann. Pure Appl. Logic*, 56:183–220, 1992.

[5] Andreas Blass. Is game semantics necessary? In Börger et al. [7], pages 66–77.

[6] Andreas Blass. Questions and answers — a category arising in linear logic, complexity theory, and set theory. In Girard et al. [14], pages 61–81.

[7] Egon Börger, Yuri Gurevich, and Karl Meinke, editors. *Computer Science Logic: 7th Workshop, CSL '93*, volume 832 of *Lecture Notes in Computer Science*. Springer-Verlag, 1994.

[8] John H. Conway. *On Numbers and Games*, volume 6 of *London Math. Soc. Monographs*. Academic Press, 1976.

[9] Valeria de Paiva. A Dialectica-like model of linear logic. In Pitt et al. [23], pages 341–356.

[10] Michael Dummett. *Elements of Intuitionism*. Oxford University Press, 1977.

[11] David Gale and F. M. Stewart. Infinite games with perfect information. *Ann. Math. Studies*, 28:245–266, 1953.

[12] Jean-Yves Girard. Linear logic. *Theoret. Comp. Sci.*, 50:1–102, 1987.

[13] Jean-Yves Girard. Toward a geometry of interaction. In Gray and Scedrov [15], pages 69–108.

[14] Jean-Yves Girard, Yves Lafont, and Laurent Regnier, editors. *Advances in Linear Logic*, volume 222 of *London Math. Soc. Lecture Notes*. Cambridge Univ. Press, 1995.

[15] John W. Gray and Andre Scedrov, editors. *Categories in Computer Science and Logic*, volume 92 of *Contemp. Math.* Amer. Math. Soc., 1989.

[16] J. Roger Hindley and Jonathan P. Seldin, editors. *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*. Academic Press, 1980.

[17] E. W. Hobson and A. E. H. Love, editors. *Proceedings of the 5th International Congress of Mathematicians*. Cambridge Univ. Press, 1913.

[18] William A. Howard. The formulae-as-types notion of construction. In Hindley and Seldin [16], pages 479–490.

[19] J. Martin E. Hyland and C.-H. Luke Ong. Fair games and full completeness for multiplicative linear logic without the MIX-rule. preprint, 1993.

[20] *Infinitistic Methods*. PWN, 1961.

[21] Giorgi Japaridze. A constructive game semantics for the language of linear logic. Preprint, 1995.

[22] P. Lorenzen. Ein dialogisches Konstruktivitätskriterium. In Infinitistic Methods [20], pages 193–200.

[23] David H. Pitt, David E. Rydeheard, Peter Dybjer, Andrew M. Pitts, and Axel Poigné, editors. *Category Theory and Computer Science*, volume 389 of *Lecture Notes in Computer Science*. Springer-Verlag, 1989.

[24] Michael O. Rabin. Effective computability of winning strategies. *Ann. Math. Studies*, 39:147–157, 1957.

[25] J. Tits. *Buildings of Spherical Type and Finite BN-Pairs*, volume 386 of *Lecture Notes in Mathematics*. Springer-Verlag, 1974.

[26] Anne S. Troelstra and Dirk van Dalen. *Constructivism in Mathematics — An Introduction*, volume 1. North-Holland, 1988.

[27] Ernst Zermelo. Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels. In Hobson and Love [17], pages 501–504.