Annals of Pure and Applied Logic 56 (1992) 183–220
North-Holland

A game semantics for linear logic*

Andreas Blass
Mathematics Department, University of Michigan, Ann Arbor, MI 48109, United States

Communicated by A. Nerode
Received 1 June 1991

Abstract

Blass, A., A game semantics for linear logic, Annals of Pure and Applied Logic 56 (1992) 183–220.

We present a game (or dialogue) semantics in the style of Lorenzen (1959) for Girard's linear logic (1987). Lorenzen suggested that the (constructive) meaning of a proposition φ should be specified by telling how to conduct a debate between a proponent P who asserts φ and an opponent O who denies φ. Thus propositions are interpreted as games, connectives (almost) as operations on games, and validity as existence of a winning strategy for P. (The qualifier 'almost' will be discussed later when more details have been presented.) We propose that the connectives of linear logic can be naturally interpreted as the operations on games introduced for entirely different purposes by Blass (1972). We show that affine logic, i.e., linear logic plus the rule of weakening, is sound for this interpretation. We also obtain a completeness theorem for the additive fragment of affine logic, but we show that completeness fails for the multiplicative fragment. On the other hand, for the multiplicative fragment, we obtain a simple characterization of game-semantical validity in terms of classical tautologies. An analysis of the failure of completeness for the multiplicative fragment leads to the conclusion that the game interpretation of the connective $\otimes$ is weaker than the interpretation implicit in Girard's proof rules; we discuss the differences between the two interpretations and their relative advantages and disadvantages. Finally, we discuss how Gödel's Dialectica interpretation (1958), which was connected to linear logic by de Paiva (1989), fits with game semantics.

1. Introduction to game semantics

Classical logic is based upon truth values. To understand a sentence is to know under what circumstances it is true. The meaning of a propositional connective is explained by telling how the truth value of a compound formula is obtained from the truth values of its constituents, i.e., by giving a truth table.

Intuitionistic logic is based upon proofs. To understand a sentence is to know what constitutes a proof of it. The meaning of a propositional connective is explained by describing the proofs of a compound formula, assuming that one knows what constitutes a proof of a constituent.

* Partially supported by NSF grant DMR-8801988.

Lorenzen [10] proposed a semantics based upon dialogues. To understand a sentence is to know the rules for attacking and defending it in a debate. ('Dialogue', 'debate' and 'game' are used synonymously.) The meaning of a propositional connective is explained by telling how to debate a compound formula, assuming that one knows how to debate its constituents.

The proof-based semantics of intuitionistic logic can be viewed as a special case of game semantics, namely the case in which the opponent O has nothing to do. The 'debate' consists of the proponent P presenting a proof; the winner of the debate is P or O according to whether the proof is correct or not. The truth-value semantics of classical logic is the even more special case in which P has nothing to do either. The winner is P or O according to whether the sentence being debated is true or false. There is a suggestive analogy between truth-value, proof, and game semantics on the one hand and deterministic, nondeterministic, and alternating computation on the other.

We describe informally Lorenzen's semantics for the propositional connectives. To attack a conjunction, O may select either conjunct, and P must then defend that conjunct. To attack a disjunction, O may demand that P select and defend one of the disjuncts. To attack a negation $\sim\varphi$, O may assert and defend φ, with P now playing the role of opponent of φ. To attack an implication $\varphi \to \psi$, O may assert φ; then P may either attack φ or assert and defend ψ. (Negation can be viewed as the special case of implication where the consequent ψ is an indefensible statement.)

The simplicity of this description of the connectives is somewhat deceptive, for Lorenzen needs supplementary rules to obtain a game semantics for constructive logic. One supplementary rule is that atomic formulas are never attacked or defended, but P may assert them only if they have been previously asserted by O. The idea behind this rule is that the semantics is to describe logical validity, not truth in a particular situation, so a winning strategy for P should succeed independently of any information about atomic facts. Thus P can safely assert such a fact only if O is already committed to it; otherwise, it might turn out to be wrong. A consequence of this rule governing atomic statements is that negating a formula does not fully interchange the roles of the two players, because it is still P who is constrained by the rule.

The remaining supplementary rules govern repeated attacks on or defenses of the same assertion. To see the need for such rules, observe that nothing in the preceding rules distinguishes $\sim(\sim\alpha \wedge \alpha)$, which is constructively valid, from $\alpha \vee \sim\alpha$, which is not. Consider these two games in the simple case that α is atomic. In the debate about $\alpha \vee \sim\alpha$, P must choose and assert one of α and $\sim\alpha$. He cannot assert α, an atomic formula that O has not yet asserted. So he asserts $\sim\alpha$, and O attacks by asserting α. Now that O has asserted α, P would like to go back and revise his defense of $\alpha \vee \sim\alpha$ by choosing α instead of $\sim\alpha$. Unfortunately (for P, but fortunately for constructive logic) Lorenzen's rules forbid such a revision. Now consider the debate about $\sim(\sim\alpha \wedge \alpha)$. O attacks by asserting $\sim\alpha \wedge \alpha$. P can attack this assertion by demanding that O assert the conjunct α. Then, P re-attacks the same assertion by demanding that O assert the other conjunct $\sim\alpha$. After O does so, P wins by attacking $\sim\alpha$ with α, which O has already asserted. The difference between the two debates is that Lorenzen's supplementary rules allow P to re-attack $\sim\alpha \wedge \alpha$ but not to re-defend $\alpha \vee \sim\alpha$.

Supplementary rules governing repeated attacks and defenses were devised by Lorenzen [10] so that the formulas for which P has a winning strategy are exactly the intuitionistically provable ones. Subsequently, Lorenz [9] systematically analyzed many variations of these supplementary rules.

Our game semantics differs from Lorenzen's in two major ways. Together, these two changes will allow us to dispense with supplementary rules and work exclusively with very natural operations on games.

The first change is that in our games a play takes infinitely many moves. In particular, assertions of atomic formulas are not terminal positions in the games but are debatable like any other formula.

A well-known result of [4] says that, when games are allowed to be infinitely long, it is possible that neither player has a winning strategy. We can use such undetermined games as the interpretations of atomic formulas, to model the idea that the players (particularly P) do not know whether an atomic formula is true or not. Then we do not need Lorenzen's supplementary rule that P cannot assert an atomic formula until O has asserted it. Consider, for example, the formula $\alpha \vee \sim\alpha$, where atomic formula α is interpreted as an undetermined game, and where we keep Lorenzen's supplementary rule prohibiting re-defense of a disjunction, but we drop the supplementary rule governing atomic formulas. Does P have a winning strategy for $\alpha \vee \sim\alpha$? If such a strategy begins by asserting α, then it must contain a winning strategy for P in α. If it begins by asserting $\sim\alpha$, then it must contain a winning strategy for P in $\sim\alpha$, i.e., for O in α. As neither P nor O has a winning strategy in α, P has no winning strategy in $\alpha \vee \sim\alpha$. Of course, if P (or O) had a winning strategy in α, then P could win $\alpha \vee \sim\alpha$ by asserting α (or $\sim\alpha$) on his first move. From this point of view, Lorenzen's supplementary rule prevents P from winning α and from winning $\sim\alpha$, at the cost of destroying the symmetry in $\sim$. By allowing infinite games, we achieve the same goal while preserving the symmetry.

Our second change in Lorenzen's rules concerns the issue of re-attacking and re-defending formulas. Rather than choosing supplementary rules to cover this question (rules that tend to introduce new asymmetries between the players), we take the more radical approach of having two sorts of conjunction (respectively, disjunction), one of which can be re-attacked (respectively, re-defended) while the other cannot.

For one sort of conjunction, which we write $\alpha \wedge \beta$, the debate consists of a choice by O of one of the conjuncts, followed by a debate about that conjunct. O cannot go back to the other conjunct.

For the other sort of conjunction, which we write $\alpha \otimes \beta$, the debate consists of two debates, one for each conjunct, interleaved as follows. O chooses which of the two subdebates to begin first. Thereafter, at each of his moves, O may choose to continue the current subdebate or to switch to the other one. When O returns to a previously abandoned debate, it is resumed at the position where it was abandoned. O wins the debate for $\alpha \otimes \beta$ if and only if he wins at least one of the subdebates. (This will be explained more precisely below.)

Analogously, replacing 'O' with 'P' in the preceding two paragraphs, we define two sorts of disjunction, $\alpha \vee \beta$ which cannot be re-defended, and $\alpha \mathbin{\breve{\otimes}} \beta$ which can. As negation consists of switching the roles of the two players, we have $\alpha \vee \beta = \sim(\sim\alpha \wedge \sim\beta)$ and $\alpha \mathbin{\breve{\otimes}} \beta = \sim(\sim\alpha \otimes \sim\beta)$.

Our earlier discussion shows that, if α is an undetermined game, then P has no winning strategy in $\alpha \vee \sim\alpha = \sim(\sim\alpha \wedge \alpha)$. On the other hand, P always has a winning strategy in $\alpha \mathbin{\breve{\otimes}} \sim\alpha = \sim(\sim\alpha \otimes \alpha)$, namely to start with the subdebate where O moves first, to switch subdebates at every move, and to copy in the new subdebate the move that O just made in the other subdebate. This mimicking strategy in effect makes O play α against himself and ensures that P wins (exactly) one of the subdebates.

It is clear that, with two sorts of conjunction and disjunction, we are not dealing with ordinary constructive (or classical) logic. Nevertheless, intuitionistic propositional logic can be embedded in this system by using a unary 'repetition' connective, called R in [2]. A debate for $R(\alpha)$ consists of many debates of α, interleaved like an iterated $\otimes$, i.e., O may, at each of his moves, continue the current debate, start a new one, or resume a previously abandoned one. In any two of the subdebates, as long as O plays the same way, P is required to do the same. Finally, O wins the debate of $R(\alpha)$ if and only if he wins at least one of the subdebates of α. [2, Theorem 7] shows that one obtains an interpretation of intuitionistic propositional logic by interpreting implication as $\sim R(\alpha) \mathbin{\breve{\otimes}} \beta$ (hence interpreting negation as $\sim R(\alpha)$ instead of $\sim\alpha$) and using the nonrepeating versions $\wedge$, $\vee$ of conjunction and disjunction.

Readers familiar with Girard's linear logic [6] will have noticed many similarities between that system and the operations on games described above. Since we have used notation from [2], we indicate the correspondence with Girard's notation and terminology in Table 1.

Table 1

Our notation	Girard's notation	
$\sim$	$\perp$	(linear negation, duality)
$\wedge$	&	(with)
$\vee$	$\oplus$	(plus)
$\otimes$	$\otimes$	(times)
$\breve{\otimes}$	⅋	(par)
R	!	(of course)
$\breve{R}$	?	(why not)

To complete the correspondence with propositional linear logic, we associate to Girard's two versions of 'true', 1 and $\top$, a game which P always wins, and we associate to his two versions of 'false', $\perp$ and 0, a game which O always wins.

2. Games and operations

In this section, we define games and the operations on them that correspond to the connectives of linear logic. We also point out some information about these operations that will be useful later. For analyzing games it will be very convenient to have them in a rather strictly normalized form (e.g., players move alternatively; the set of possible moves is the same at every position), but for describing games it will be very inconvenient to normalize the description. Accordingly, we shall define games with the strict normalization, but we shall also indicate ways in which the normalization can be relaxed and ways in which the resulting 'relaxed games' can be converted to equivalent 'strict games'.

A (strict) *game* (or debate or dialogue) between two players, P and O, consists of the following data: a set M of possible moves, a specification of P or O as the player who moves first, and a set G of infinite sequences of members of M, namely the plays won by P. Thus, a game is an ordered triple (M, s, G), where $s \in \{P, O\}$ and $G \subseteq {}^{\omega}M$, but we often write simply G. We write $\bar{s}$ for the nonstarting player, the member of $\{P, O\}$ other than s.

A *position* p in a game is a finite sequence of moves. It is a position with the starting player $s \in \{P, O\}$ to move if its length is even; otherwise, it is a position with the other player $\bar{s}$ to move. A *strategy* for either player is a function σ into M from the set of all positions where that player is to move. A player *follows* strategy σ in a play $x \in {}^{\omega}M$ if, for every position p in x (i.e., finite initial segment p of x) where that player is to move, the next term after p in x is $\sigma(p)$. A strategy σ for P (respectively, O) is a *winning* strategy if all (respectively, none) of the plays in which P (respectively, O) follows σ are in G. A game is *determined* if there is a winning strategy for one of the players. It is well known [4] that games with plays of infinite length need not be determined; extensions of this result will play an important role in our analysis of game-semantical validity.

It will often be convenient to describe games in a way that does not adhere to all the conventions built into our definition of strict games. In a relaxed game, we do not assume that the players move alternately; rather, the rules specify, for each position, who is to move next. We require, for technical reasons, that a player has only finitely many consecutive moves. Clearly, any game of this sort can be regarded as a strict game by declaring a block of consecutive moves by the same player in the relaxed game to be a single move in the strict game. This will, of course, require that the set M of possible moves in the strict game is a bit more complex than in the relaxed game.

We also allow, in relaxed games, rules of the sort 'in position p, the next move must be in M_p' where M_p is some subset of M. In a strict game, where all moves in M are always legal, we regard such a rule as meaning that any play violating the rule is lost by the player responsible for the earliest violation.

As an example for both sorts of relaxation, we mention the game $\alpha \wedge \beta$, where O's first move, the choice of α or β, may be immediately followed by another move of O if he is the starting player in the chosen component. Furthermore, the first move is required to be a choice of α or β and all subsequent moves are required to be in that component. Writing out the rules for $\alpha \wedge \beta$ in strict form (assuming that α and β are strict) should convince the reader of the utility of relaxed descriptions of games.

We are now ready to define and discuss the operations on games that will be used to interpret the connectives of linear logic.

The *negation* of a game (M, s, G) is defined by

$$\sim(M, s, G) = (M, \bar{s}, {}^{\omega}M - G),$$

which simply interchanges the roles of the two players. In particular, a winning strategy for either player in $\sim G$ is the same as a winning strategy for the other player in G. Obviously $\sim\sim G = G$.

The (nonrepeating or additive) *conjunction* of two games $(M_0, s_0, G_0) \wedge (M_1, s_1, G_1)$ has the following relaxed description. The set of moves is $M_0 \cup M_1 \cup \{0, 1\}$. The first move is by O and must be 0 or 1. If it is 0 (respectively, 1), then subsequent play is governed — both as to whose move it is at any finite stage and as to who had won when the play is complete — by the rules of G_0 (respectively, G_1). A winning strategy for O in $G_0 \wedge G_1$ consists (essentially) of a choice of one constituent, G_0 or G_1, and a winning strategy for O in that constituent. (The word 'essentially' covers the technicality that a strategy includes responses to positions that cannot arise when the strategy is followed. If this irrelevant part of a strategy is ignored, then our description of strategies for $G_0 \wedge G_1$ is accurate.) A winning strategy for P in $G_0 \wedge G_1$ consists of winning strategies for P in both constituent games.

The (nonrepeating or additive) disjunction of two games is defined by reversing the roles of P and O in the preceding description of conjunction. Equivalently and more succinctly,

$$G_0 \vee G_1 = \sim(\sim G_0 \wedge \sim G_1).$$

The repeating or multiplicative conjunction, also called the (tensor) *product*, $G_0 \otimes G_1$ of games (M_0, s_0, G_0) and (M_1, s_1, G_1) is somewhat more complex. We assume that G_0 and G_1 are in strict form and that M_0 and M_1 are disjoint (otherwise replace them with disjoint copies). We describe $G_0 \otimes G_1$ in relaxed form. Its set of moves is $M = M_0 \cup M_1$. If $p \in {}^{<\omega}M$ is any position, $(p)_0$ and $(p)_1$ are the subsequences of p consisting of moves in M_0 and M_1, respectively. We use similar notation $(x)_0$ and $(x)_1$ when $x \in {}^{\omega}M$ is an (infinite) play of $G_0 \otimes G_1$.

Note that although x is infinite, one of $(x)_0$ and $(x)_1$ could be finite. In any position p, O is to move if and only if O is to move in *both* $(p)_0$ and $(p)_1$ according to the rules of G_0 and G_1, respectively. (Any move in M is legal for O.) So P is to move at position p in $G_0 \otimes G_1$ if and only if P is to move at $(p)_0$ or $(p)_1$ (in G_0 or G_1, respectively) or both; if he is to move at both $(p)_0$ and $(p)_1$, then any move in M is legal, but if he is to move only at $(p)_i$, then his next move must be in M_i. A play $x \in {}^{\omega}M$ is won by O if at least one of the subsequences $(x)_i$ is in ${}^{\omega}M_i - G_i$, i.e., is infinite and is a win for O in the appropriate constituent game G_i. Thus, P wins the play x of $G_0 \otimes G_1$ if and only if $(x)_i \in {}^{<\omega}M \cup G_i$ for both $i = 0$ and $i = 1$.

This completes the definition of $G_0 \otimes G_1$, but the following observations will clarify what can happen during the game and will be useful in several proofs below.

First consider any move by P or by O in $G_0 \otimes G_1$, and let p be the position after this move. Then O is to move in at least one of the two constituent games, i.e., at least one of the subsequences $(p)_i$ has O to move in G_i. Indeed, if the move in question was by P and was in M_i, then O is to move in G_i afterward. (Recall that G_0 and G_1 are in strict form, so after a move of P in G_i it is O's turn.) On the other hand, if the move in question is by O, then, before this move, O was to move in *both* constituent games; he moved in one, so it is still his move in the other.

Thus the only time player P can be to move in both constituents is at the beginning of the game (if $s_0 = s_1 = P$). In this situation, P makes two consecutive moves, one in each constituent. (Of course, the strict version of $G_0 \otimes G_1$ would combine these two into a single move.) In all other situations, the players move alternately in $G_0 \otimes G_1$ (even in our relaxed version). After O moves in a constituent G_i, P must reply in the same G_i, as it is not his move in the other constituent. Thus, if two consecutive moves are in different constituents, then the first is by P and the second by O, unless the two moves are at the very beginning and are both by P. For any legal position p of length $\geqslant 2$ in $G_0 \otimes G_1$, the subsequence $(p)_0$ or $(p)_1$ that does not contain the last term of p has O to move; in other words, a constituent game that has been abandoned (or has not been started after two moves) has O to move.

The rules governing who is to move at a position p and who has won a play x can be conveniently summarized by the following device. Label the situations 'P has won' (for an infinite play x) and 'O is to move' (for a finite position p) with the truth value True, and label the situations 'O has won' and 'P is to move' by False. Then the truth value associated to any play x or position p in $G_0 \otimes G_1$ is obtained from the truth values associated to $(x)_0$ or $(p)_0$ in G_0 and to $(x)_1$ or $(p)_1$ in G_1 by applying the truth table for conjunction. We refer to this as the truth table description of $\otimes$. The verification that this description agrees with our definition of $G_0 \otimes G_1$ is routine, given the observation above that abandoned constituents have O to move. For example, an infinite play x is labeled False if and only if O has won it, if and only if at least one of $(x)_0$ and $(x)_1$ is infinite and won by O, if and only if at least one of $(x)_0$ and $(x)_1$ is labeled False (because a finite $(x)_i$ would have O to move and would therefore be labeled True).

A winning strategy of P in $G_0 \otimes G_1$ must include winning strategies for P in both G_0 and G_1, for O could choose to always move in the same constituent G_i and then P wins the play of $G_0 \otimes G_1$ if and only if he wins in that G_i. Conversely, if P has winning strategies in both games G_i, then he can use them to win the 'simultaneous exhibition' $G_0 \otimes G_1$ by playing the two constituent games independently. Thus, P has a winning strategy for $G_0 \otimes G_1$ if and only if he has winning strategies for both G_0 and G_1. This is why $\otimes$ can be considered a form of conjunction.

One might expect that O has a winning strategy for $G_0 \otimes G_1$ if and only if he has one in at least one G_i, but only half of this is true. With a winning strategy for one constituent, O can win $G_0 \otimes G_1$ by never playing in the other constituent. But it is possible for O to have a winning strategy $G_0 \otimes G_1$ without having one in either G_0 or G_1. We shall discuss this possibility further after introducing the dual operation $\breve{\otimes}$.

The operation dual to tensor product, the repeating or multiplicative disjunction, also called *par*, is defined by

$$G_0 \breve{\otimes} G_1 = \sim(\sim G_0 \otimes \sim G_1).$$

Its definition can thus be obtained from the definition of $\otimes$ by interchanging P and O. The same interchange, applied to our discussion of $\otimes$, gives information about $\breve{\otimes}$ that we will use without further explanation. For the truth-table description, however, we prefer to interchange not only P and O be also the two truth values. The effect of this is that the labeling rules (True when P wins or O is to move, False when O wins or P is to move) are the same as for $\otimes$, but the truth table for conjunction is replaced by that for disjunction.

Notice that the same labeling conventions also lead to a truth-table description of $\sim$, using (of course) the truth table for negation. It follows that, if a game is built from various subgames by means of $\sim$, $\otimes$ and $\breve{\otimes}$, then the winner of an (infinite) play and the next player to move at a (finite) position can be computed from the corresponding information in all the subgames by means of our standard labeling rules and a truth table as follows. In the expression for the compound game in terms of its subgames, replace each subgame by the truth value describing the play or position there, replace the game operations $\sim$, $\otimes$ and $\breve{\otimes}$ by negation, conjunction and disjunction, respectively, and evaluate the resulting Boolean expression. (Caution: if the same subgame has several occurrences in the compound game, then these occurrences can have different truth values assigned to them, as there may be different positions or plays there. Thus, although $A \breve{\otimes} \sim A$ corresponds to a tautology, it is not the case that, in a game of this form, O is always to move and P wins.)

To finish our discussion of the basic properties of $\otimes$ and $\bar{\otimes}$, we describe a game (from [2] but known earlier) G such that O has a winning strategy in $G \otimes G$ without having one in G; in fact, P will have a winning strategy in $G \bar{\otimes} G$ (which clearly implies O does not have one in G). Fix a nonprincipal ultrafilter U on the set ω of natural numbers. G is played as follows. P moves first by putting into his pocket some finite initial segment $[0, a]$ of ω. (Formally, his move is just $[0, a]$.) Then O puts into his own pocket a finite initial segment $[a+1, b]$ of what remains. Then P takes $[b+1, c]$ for some $c > b$, and so on. After infinitely many moves, ω has been partitioned into two parts — the set in P's pocket and the set in O's pocket. One and only one of these sets is in U and the owner of that set wins the game.

Here is a winning strategy for O in $G \otimes G$. As P moves first in G, he makes two consecutive moves to open $G \otimes G$, one in each component. Suppose these moves are $[0, a_0]$ and $[0, a_1]$, and without loss of generality suppose $a_0 \leq a_1$. Then O should play $[a_1+1, a_1+1]$ (i.e., pocket just one number) in the latter constituent. When P replies, necessarily in the same constituent, with $[a_1+2, b]$, then O should switch to the other constituent and play $[a_0+1, b]$. From now on, at each of his moves, O switches constituents and chooses the same interval that P just chose in the other constituent. This ensures that the set pocketed by O in the first constituent game and the set pocketed by O in the second constituent game are complementary, modulo a finite set (bounded by b). As the ultrafilter U is nonprincipal, it contains (exactly) one of these almost complementary sets, so O wins (exactly) one of the constituent games and therefore wins $G \otimes G$.

A similar argument shows that P has a winning strategy in $G \bar{\otimes} G$, and therefore neither player has a winning strategy in G.

We complete the definition of our operations on games by introducing the (unbounded) *repetition* $R(G)$ of a game G. The definition of $R(G)$ is slightly simpler if O moves first in G, so we begin with this case.

Let (M, O, G) be a strict game where O moves first. The set of moves of $R(G)$ is $M \times \omega$. (Referring to the description of $R(G)$ in Section 1, we regard $(m, i) \in M \times \omega$ as move m in the ith copy of G.) For a position $p \in {}^{<\omega}(M \times \omega)$ or a play $x \in {}^{\omega}(M \times \omega)$, and for any $i \in \omega$, consider the subsequence of p or x consisting of all terms with second component i, and let $(p)_i$ or $(x)_i$ be obtained from this subsequence by deleting these i's and keeping only the first components. Thus $(p)_i \in {}^{<\omega}M$ and $(x)_i \in {}^{\leq\omega}M$. At position p, O is to move in $R(G)$ if and only if O is to move at all the positions $(p)_i$ in G; any move in $M \times \omega$ is legal for O. Then P is to move in $R(G)$ at p if and only if for some i, P is to move in G at $(p)_i$. There are two constraints on P's moves. First, he must move in a constituent game where it is his turn, i.e., the second component of his move must be an i such that P is to move in G at $(p)_i$. Second, P must play consistently in the various constituent games in the sense that as long as O makes the same moves in the ith and jth copies of G, so must P. Formally, this means that, if P makes a move (m, i) and if $(p)_i$ is a proper initial segment of $(p)_j$, then the first term in $(p)_j$ after $(p)_i$ must be m. (This consistency requirement is what distinguishes $R(G)$ from an infinitary tensor product.) O wins a play x of $R(G)$ if and only if at least one $(x)_i$ is infinite and is a win for O in G.

Since O moves first in G, it is easy to verify that the two players move alternately. After a move by P, it is O's move in all copies of G; after a move by O, it is P's move in exactly one copy of G, so P never has a choice about which copy of G to move in. As with tensor products, a constituent game G in $R(G)$ can be abandoned only in positions with O to move. It follows that the rules governing who is to move at p and who has won x in $R(G)$ are given by a truth-table description using infinite conjunction.

Now suppose G is a game (M, P, G) in which P moves first. If $R(G)$ were defined as above, P would make the first move, say (m, j). Thereafter, he would have to keep playing (m, i) with the same m (by the consistency rule) and different i's. O would never get to move, since it would take forever for P to make his opening moves in all of the copies of G. As the consistency rule requires all these opening moves to be the same, we adopt the convention that P's first move, made in any copy of G, is instantly duplicated in all the other copies, so that the play can proceed. Formally, this means that, for any nonempty $p \in {}^{<\omega}M$ and any $x \in {}^{\omega}M$, with first term (m, j), we define $(p)_j$ and $(x)_j$ as before, but for $i \neq j$ we define $(p)_i$ and $(x)_i$ to have an extra term m at the beginning. The rest of the definition of $R(G)$ is then exactly as in the case where O moves first in G.

We write $\tilde{R}$ for the dual operation, $\tilde{R}(G) = {\sim}R({\sim}G)$. Thus, everything we have said about R applies to $\tilde{R}$ if we reverse the roles of P and O (and, in the truth-table description, interchange True with False and conjunction with disjunction).

Finally, we fix the notations $\top$ and $\bot$ for games in which P and O, respectively, have winning strategies. It will not matter how these games are chosen, but for definiteness we let $\top$ be the game with set of moves $M = \{0\}$, with O moving first, and with the unique possible play being a win for P, and we define $\top$ to be ${\sim}\bot$.

3. Linear logic, affine logic and game semantics

In this section, we present the formal systems, introduced by Girard [6], which our game semantics is intended to interpret. To avoid repeated references to Table 1, we present Girard's system using a notation for the connectives that matches our notation in Section 2 (mostly taken from [2]) for the operations on games. This notational change is only for convenience in the present paper and should not be construed as recommending a new notation for linear logic.

Propositional linear logic can be viewed as a standard sequent calculus for propositional logic minus the rules of contraction and weakening. The absence of these two rules means that classically equivalent formulations of the rules for conjunction and disjunction are no longer equivalent. One formulation is used for $\wedge$ and $\vee$, the other for $\otimes$ and $\bar{\otimes}$. We follow the fairly standard convention of using linear negation to write all our sequents in one-sided form, i.e., with all formulas on the right of $\vdash$. A sequent with formulas on the left of $\vdash$ is to be identified with the result of applying $\sim$ to these formulas while transposing them to the right.

The *formulas* of linear logic are built from propositional variables, negated (by $\sim$) propositional variables, and the constants $1, \bot, \top, 0$ by means of the binary operations $\otimes, \bar{\otimes}, \wedge, \vee$ and the unary operations R and $\tilde{R}$. A *sequent* is an expression $\vdash \Gamma$, where Γ is a finite list of (not necessarily distinct) formulas. Negation of formulas is primitive for atomic formulas and is defined for other formulas by

$$
\begin{array}{ll}
\sim\sim A = A, & \\
\sim 1 = \bot, & \sim \bot = 1, \\
\sim \top = 0, & \sim 0 = \bot, \\
\sim(A \otimes B) = (\sim A) \bar{\otimes} (\sim B), & \sim(A \bar{\otimes} B) = (\sim A) \otimes (\sim B), \\
\sim(A \wedge B) = (\sim A) \vee (\sim B), & \sim(A \vee B) = (\sim A) \wedge (\sim B), \\
\sim R(A) = \tilde{R}(\sim A), & \sim \tilde{R}(A) = R(\sim A).
\end{array}
$$

(Strictly speaking, the first of these formulas, $\sim\sim A = A$, is a definition only when A is a propositional variable; it is a trivial theorem for all other A.)

The axioms and rules of linear logic are the following, in which A and B represent arbitrary formulas and Γ and Δ represent arbitrary lists of formulas.

Logical axioms.

$$\vdash A, \sim A.$$

Structural rules.

$$\text{(Exchange)}\ \frac{\vdash \Gamma, A, B, \Delta}{\vdash \Gamma, B, A, \Delta}, \qquad \text{(Cut)}\ \frac{\vdash \Gamma, A \quad \vdash \Delta, \sim A}{\vdash \Gamma, \Delta}.$$

(Notice that the exchange rule lets us ignore the order of formulas in a sequent; we can work with multisets of formulas rather than lists.)

Additive rules.

$$\frac{\vdash \Gamma, A \quad \vdash \Gamma, B}{\vdash \Gamma, A \wedge B}, \tag{$\wedge$}$$

$$\vdash \Gamma, \top, \tag{$\top$}$$

$$\frac{\vdash \Gamma, A}{\vdash \Gamma, A \vee B}, \qquad \frac{\vdash \Gamma, B}{\vdash \Gamma, A \vee B}. \tag{$\vee$}$$

(There is no rule introducing 0.)

Multiplicative rules.

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

$$\vdash 1, \tag{1}$$

$$\frac{\vdash \Gamma, A, B}{\vdash \Gamma, A \mathbin{\tilde{\otimes}} \mathrm{B}}, \tag{$\tilde{\otimes}$}$$

$$\frac{\vdash \Gamma}{\vdash \Gamma, \perp}. \tag{$\perp$}$$

Exponential rules.

$$\frac{\vdash \Gamma, A}{\vdash \Gamma, \tilde{R}(A)}, \tag{Dereliction $\tilde{R}$}$$

$$\frac{\vdash \Gamma}{\vdash \Gamma, \tilde{R}(A)}, \tag{Weakening $\tilde{R}$}$$

$$\frac{\vdash \Gamma, \tilde{R}(A), \tilde{R}(A)}{\vdash \Gamma, \tilde{R}(A)}, \tag{Contraction $\tilde{R}$}$$

$$\frac{\vdash \tilde{R}(\Gamma), A}{\vdash \tilde{R}(\Gamma), R(A)}. \tag{R}$$

(In the (R) rule, $\tilde{R}(\Gamma)$ means the result of applying $\tilde{R}$ to all members of Γ.)

Affine logic (see [7]) is obtained by adding to linear logic the structural rule of weakening:

$$\frac{\vdash \Gamma}{\vdash \Gamma, A}. \tag{Weakening}$$

By including the rule of weakening, affine logic obliterates the distinction between 1 and $\top$ as well as the distinction between 0 and $\perp$. To see this, note that $\vdash \perp, \top$ is provable in linear logic (by $(\top)$) and $\vdash 1, 0$ is provable in affine logic (by (1) and weakening). The former is (up to exchanges) both $\vdash \perp, {\sim}0$ and $\vdash \top, {\sim}1$, while the latter is (up to exchanges) both $\vdash 1, {\sim}\top$ and $\vdash 0, {\sim}\perp$. These sequents and the cut rule allow us to replace 0 by $\perp$, 1 by $\top$, $\top$ by 1 and $\perp$ by 0 in any deduction.

Conversely, if we identify 1 with $\top$ and 0 with $\perp$ in linear logic, the rule of weakening becomes derivable as follows. From $\vdash \Gamma$, we obtain $\vdash \Gamma, \perp$ by rule $(\perp)$, and we have $\vdash A, \top$ by $(\top)$. But $\top$ has been identified with ${\sim}\perp$ (i.e., with 1), so the cut rule gives $\vdash \Gamma, A$.

The preceding observations establish the following simplification of the presentation of affine logic.

Proposition. *Affine logic is equivalent to the system obtained from linear logic by deleting the primitive symbols* 0 *and* 1, *redefining* $\sim\top$ *as* $\perp$ *and* $\sim\perp$ *as* $\top$, *and deleting rule* (1).

By a *game interpretation* of linear or affine logic, we mean a function assigning to each propositional variable a game. Such a function extends to arbitrary formulas by interpreting each of the symbols $\sim$ (applied to atomic formulas), $\wedge$, $\vee$, $\otimes$, $\bar{\otimes}$, R, $\bar{R}$ of linear logic as the operation on games written with the same symbol in Section 2, and by using the games $\top$ and $\perp$ defined at the end of Section 2 as the interpretations of $\top$ and $\perp$, respectively (and of 1 and 0, respectively, in the case of linear logic). Notice that all the clauses in the definition of negation for nonatomic formulas are correct when read as assertions about game operations. This ensures that negation in linear logic is interpreted as the operation $\sim$ on games, even if the formula being negated is not atomic. A sequent is interpreted by applying $\bar{\otimes}$ to the interpretations of the formulas in it. When an interpretation has been specified, we can use the same symbol for a formula or sequent and its interpretation; this convention will cause no confusion and will facilitate reading.

A sequent or formula is *true* in an interpretation if, in the game associated to the sequent by the interpretation, P has a winning strategy. A sequent or formula is (game-semantically) *valid* if it is true in all game interpretations.

Soundness Theorem. *For an arbitrary game interpretation, all axioms of affine logic are true and all rules of affine logic preserve truth.*

Corollary. *All sequents provable in affine logic are valid.*

Proof of the Soundness Theorem. We work with the simplified system of affine logic described in the Proposition. Recall that sequents are interpreted by combining the interpretations of the formulas with $\bar{\otimes}$; this makes soundness of the $\bar{\otimes}$ rule trivial. Recall also that, when games are combined by $\bar{\otimes}$, P has the option of switching from one component game to another at any of his moves, and P wins the compound game if he wins at least one component. For the rest of this proof, we work with a fixed but arbitrary game interpretation, and we do not distinguish notationally between a formula or sequent and its interpretation as a game.

Soundness of the exchange rule is trivial, as $\bar{\otimes}$ is a commutative operation on games. Rules ($\perp$) and (Weakening $\bar{R}$) are special cases of the general weakening rule, which is sound because if P has a winning strategy for Γ, then he can win $\Gamma \bar{\otimes} A$ by using his strategy in Γ and never playing in A. This observation also shows that, since P has a winning strategy in $\top$, he has one in $\Gamma \bar{\otimes} \top$, so ($\top$) is also sound. We now verify the soundness of the remaining axioms and rules one by one.

Logical axioms. P wins $A \bar{\otimes} \sim A$ by a mimicking strategy as follows. Assume, with loss of generality, that P moves first in A (otherwise interchange the roles of A and $\sim A$). So a play of $A \bar{\otimes} \sim A$ starts with a move by O in $\sim A$. Let P immediately switch to the other constituent, A, and make the same move there. Whatever move O replies with in A is to be copied by P in $\sim A$. In general, after any move by O, P should immediately switch to the other constituent and make the same move there. This strategy ensures that the plays in the two constituents, $(x)_0$ and $(x)_1$ in the notation of Section 2, are identical. As P is playing opposite roles in A and $\sim A$, he is sure to win one (and lose the other), thereby winning $A \bar{\otimes} \sim A$.

Cut. Suppose P has winning strategies σ in $\Gamma \bar{\otimes} A$ and τ in $\Delta \bar{\otimes} \sim A$. Here are instructions by which P can win $\Gamma \bar{\otimes} \Delta$. Pretend that, in addition to the components Γ and Δ, a game of A is being played; you will make the moves of both P and O in this imaginary A. Your actual moves in Γ and your moves as P in A will be played according to σ; your actual moves in Δ and your moves as O in A, i.e., P in $\sim A$, will be played according to τ. The situation may be visualized in terms of two assistants of P, of whom one knows σ and the other knows τ. The former plays (as P) in the real Γ and the imaginary A; the latter plays (as P) in the real Δ and (as O) in the imaginary A. Whenever P is to move in $\Gamma \bar{\otimes} \Delta$, hence in both Γ and Δ, one of the assistants, namely the one who is to move in A, is to move in his game, $\Gamma \bar{\otimes} A$ or $\Delta \bar{\otimes} \sim A$. If that assistant's move is in Γ or Δ, then it counts as P's move in the actual game $\Gamma \bar{\otimes} \Delta$. If it is in A, then afterward it is the other assitant's turn; the assistants keep playing according to σ and τ until one makes a move in Γ or Δ. Notice that they cannot keep moving in A forever, for if they did, then whichever of them lost that play of A would also have lost his game $\Gamma \bar{\otimes} A$ or $\Delta \bar{\otimes} \sim A$ (since only finitely many moves were made in Γ and Δ), an absurdity since σ and τ are winning strategies. This observation shows that P (or his assistants) eventually produces a move in $\Gamma \bar{\otimes} \Delta$, so we have described a strategy for P in $\Gamma \bar{\otimes} \Delta$. It remains to verify that it is a winning strategy.

Consider any play of $\Gamma \bar{\otimes} \Delta$ where P used this strategy. Suppose first that, during this play, the assistant responsible for $\Gamma \bar{\otimes} A$ made only finitely many moves (in Γ and in A). Then the other assistant made infinitely many moves (as every move by P is from one of the assistants) and therefore won $\Delta \bar{\otimes} \sim A$. As only finitely many moves were made in A, this assistant must have won Δ, and so P wins $\Gamma \bar{\otimes} \Delta$. An analogous argument applies if the assistant responsible for $\Gamma \bar{\otimes} \sim A$ made only finitely many moves. There remains the case that both assistants made infinitely many moves and therefore won their games $\Gamma \bar{\otimes} A$ and $\Delta \bar{\otimes} \sim A$. If the first assistant won Γ or the second won Δ, then P won $\Gamma \bar{\otimes} \Delta$, as desired. Otherwise, both assistants won A, which is absurd as they played A against each other. This completes the proof that the cut rule is sound.

($\wedge$). Suppose P has winning strategies σ for $\Gamma \bar{\otimes} A$ and τ for $\Gamma \bar{\otimes} B$. Here are instructions whereby P can win $\Gamma \bar{\otimes} (A \wedge B)$. As O moves first in $A \wedge B$ (to pick a component), by the time it is P's move in $\Gamma \bar{\otimes} (A \wedge B)$, O will have chosen A

or B, so the game being played is effectively $\Gamma \tilde{\otimes} A$ or $\Gamma \tilde{\otimes} B$, and P uses σ or τ accordingly.

($\vee$). Suppose P has a winning strategy σ in $\Gamma \tilde{\otimes} A$. Then P wins $\Gamma \tilde{\otimes} (A \vee B)$ by making his first move in the component $A \vee B$, choosing A there, and then following σ. The other ($\vee$) rule is handled the same way.

($\otimes$). Suppose P has winning strategies σ in $\Gamma \tilde{\otimes} A$ and τ in $\Delta \tilde{\otimes} B$. Here are instructions whereby P can win $\Gamma \tilde{\otimes} \Delta \tilde{\otimes} (A \otimes B)$. Make sure that all your moves in Γ and A (respectively, in Δ and B) are played in accordance with σ (respectively, τ). Whenever it is P's move in $\Gamma \tilde{\otimes} \Delta \tilde{\otimes} (A \otimes B)$, it is (by the truth-table descriptions for $\tilde{\otimes}$ and $\otimes$) P's move in Γ, in Δ, and in at least one of A and B. (By our discussion of $\otimes$, 'at least one' can be replaced by 'exactly one' as soon as some moves have been made in $A \tilde{\otimes} B$.) If it is P's move in A, then σ provides a move in Γ or in A; otherwise, τ provides a move in Δ or in B. In either case, P can move in accordance with our instructions, so we have a strategy. To see that it is a winning strategy, consider any play using it. If P wins either the Γ or the Δ component, then he wins $\Gamma \tilde{\otimes} \Delta \tilde{\otimes} (A \otimes B)$. If not, then he wins A (because σ must win at least one of Γ and A) and B (similarly) and therefore $A \otimes B$ and therefore $\Gamma \tilde{\otimes} \Delta \tilde{\otimes} (A \otimes B)$.

Dereliction $\tilde{R}$. If P has a winning strategy in $\Gamma \tilde{\otimes} A$, then he can use it to win $\Gamma \tilde{\otimes} \tilde{R}(A)$ by never exercising his option to start a second copy of A in $\tilde{R}(A)$.

Contraction $\tilde{R}$. Suppose P has a winning strategy σ in $\Gamma \tilde{\otimes} \tilde{R}(A) \tilde{\otimes} \tilde{R}(A)$. To win $\Gamma \tilde{\otimes} \tilde{R}(A)$, he should pretend that he is playing $\Gamma \tilde{\otimes} \tilde{R}(A) \tilde{\otimes} \tilde{R}(A)$, the even (respectively, odd) numbered constituents of the one actual $\tilde{R}(A)$ being identified with all the constituents of the first (respectively, second) imaginary $\tilde{R}(A)$. Using σ in the imaginary game gives, via this identification, a win for P in the real game.

(In the imaginary game, the consistency rule constrains O only for pairs of A's within the same $\tilde{R}(A)$; in the real game, all pairs of A's are constrained. So the real game is actually a bit easier for P. In other words, σ contains information that P will never need for the real game.)

(R). Suppose P has a winning strategy σ in $\tilde{R}(\Gamma) \tilde{\otimes} A$, i.e., $\tilde{R}(C_1) \tilde{\otimes} \cdots \tilde{\otimes} \tilde{R}(C_r) \tilde{\otimes} A$. To win $\tilde{R}(C_1) \tilde{\otimes} \cdots \tilde{\otimes} \tilde{R}(C_r) \tilde{\otimes} R(A)$, he should proceed as follows. Within each $\tilde{R}(C_i)$ there are countably infinitely many copies of C_i, indexed (according to the definition of $\tilde{R}$) by ω. Use a pairing function to re-index them by $\omega \times \omega$; the copies indexed by (k, l) for a fixed k and varying l will be called the kth block of copies of C_i. The idea is that the kth blocks of $C_1, \ldots, C_r$ and the kth copy of A in $R(A)$ will be treated as a copy of $\tilde{R}(C_1) \tilde{\otimes} \cdots \tilde{\otimes} \tilde{R}(C_r) \tilde{\otimes} A$, and σ will be applied to it. More precisely, when P is to move in $\tilde{R}(C_1) \tilde{\otimes} \cdots \tilde{\otimes} \tilde{R}(C_r) \tilde{\otimes} R(A)$, it is his move in all copies of all the C_i's and in some copy, say the kth, of A. Then P should make the move prescribed by σ for the current position in this kth copy of A and the kth blocks of all the C_i's; this makes sense, as this is a position with P to move in $\tilde{R}(C_1) \tilde{\otimes} \cdots \tilde{\otimes} \tilde{R}(C_r) \tilde{\otimes} A$. To see that this strategy is a winning one, consider any play where P uses it. If P wins any component of any $\tilde{R}(C_i)$, then he wins

$\tilde{R}(C_1) \bar{\otimes} \cdots \bar{\otimes} \tilde{R}(C_r) \bar{\otimes} R(A)$, as desired. If not, then, for each k, as P has not won any copy of any C_i in the kth block, he must have won the kth copy of A, because σ is a winning strategy. But then P has won $R(A)$ and therefore $\tilde{R}(C_1) \bar{\otimes} \cdots \bar{\otimes} \tilde{R}(C_n) \bar{\otimes} R(A)$, as desired. □

4. Additive completeness

The Soundness Theorem proved in Section 3 makes it natural to ask about completeness. Are all valid sequents provable? We shall show in this section that the answer is affirmative when all the formulas in the sequent belong to the additive fragment of the language, defined as follows. *Additive formulas* are formulas built from propositional variables, their negations, $\top$, and $\perp$ by means of the *additive connectives* $\wedge$ and $\vee$. This definition is adapted to affine logic; in linear logic $\top$ and 0 are additive while 1 and $\perp$ are not, but our definition incorporates the identification of 1 with $\top$ and 0 with $\perp$ in affine logic. A sequent is additive if all the formulas in it are additive. Notice that, when a sequent has two or more formulas, they are interpreted as if joined by $\bar{\otimes}$, so this nonadditive connective is implicit in additive sequents.

Additive Completeness Theorem. *An additive sequent is provable in affine logic if and only if it is game-semantically valid.*

Proof. 'Only if' is given by the Corollary of the Soundness Theorem, and does not require additivity. To prove 'if', we suppose that $\vdash \Gamma$ is an unprovable additive sequent, and we construct strict games to interpret the propositional variables in Γ so that P has no winning strategy in Γ. The game associated to a propositional variable p will have as its set of possible moves $\{0, 1\}$, and P will move first. We have yet to specify the set $G_p \subseteq {}^{\omega}\{0, 1\}$ of plays won by P in the game associated to p, but what we have already specified is enough to determine the set of positions in the game Γ and the set of strategies for P. As there are only countably many positions, the number of strategies is the cardinality c of the continuum. Fix a well-ordering of the set of all strategies for P, having order-type the initial ordinal of cardinality c: thus, each strategy σ has fewer than c predecessors in this well-ordering.

We shall define the sets $G_p \subseteq {}^{\omega}\{0, 1\}$ by transfinite recursion over this well-ordering. At the recursion step associated to a strategy σ, we shall decide, for finitely many $x \in {}^{\omega}\{0, 1\}$, which G_p's x will be in. We say that these x's are decided at stage σ. These decisions will be made in a way that ensures that σ is not a winning strategy for P in Γ. As every possible strategy for P in Γ occurs in our well-ordering, the whole construction will ensure that P has no winning strategy for Γ. The rest of the proof consists of showing how to carry out one step in the induction, say the step associated to σ.

There have been fewer than c previous steps, each deciding only finitely many $x \in \{0, 1\}$. We split each of these x's into the two subsequences of moves attributable to the two players; thus, for each decided x, one subsequence consists of the even-numbered terms in x, the other of the odd-numbered terms, because we are dealing with strict games. There are fewer than c subsequences so obtained — two from each of fewer than c decided x's — so we can fix a $z \in {}^{\omega}\{0, 1\}$ that is not such a subsequence.

Let Γ be $C_1, C_2, \ldots, C_r$ where the C_i are additive formulas. Consider an arbitrary play of the game Γ, i.e., of $C_1 \bar{\otimes} \cdots \bar{\otimes} C_r$. The moves in any component game C_i come in two phases. In phase 1, the players are choosing conjuncts or disjuncts in subformulas for C_i. For example, if C_i is $(p \wedge \sim q) \vee r$, where p, q, r are propositional variables, then phase 1 contains P's opening move, choosing $p \wedge \sim q$ or r, and, if he chooses the former, then phase 1 also contains O's reply, choosing p or $\sim q$. Each phase 1 move replaces the ith component of Γ by one of its conjuncts or disjuncts, and phase 1 continues in the ith component until it is reduced to a *literal*, i.e., to a propositional variable or the negation of one or $\top$ or $\bot$. Then comes phase 2, in which the players play (the game associated to) that literal. In any component, the phase 1 moves precede the phase 2 moves, but it is possible for phase 2 to begin in one component before phase 1 is finished in another component. It is also possible for a play of Γ to have only finitely many moves in some component, and then phase 1 may not be finished there.

At any stage of the play, we write Γ' for the current list of component games. Initially, Γ' is Γ, but every phase 1 move replaces some formula in Γ' with one of its conjuncts or disjuncts.

The preceding discussion concerned an arbitrary play of Γ. We now focus our attention on the particular play of Γ in which P follows strategy σ (recall that we are describing step σ in the inductive construction of the sets G_p) while O

(1) plays phase 1 so that $\vdash \Gamma'$ is never provable;

(2) plays phase 2 moves in (the games corresponding to) literals of the form $\top$ or $\bot$ in the only legal way; and

(3) plays phase 2 moves in literals of the form p or $\sim p$ by making the fixed sequence z of moves in each such literal.

Recall that the set of moves for $\top$ and $\bot$ is simply $\{0\}$, so 'the only legal way' in (2) means the moves are all 0's. Recall also that z was chosen to be distinct from the subsequence of either player's moves in every decided x. Thus, (3) ensures that the plays x in literals of the forms p and $\sim p$ are not yet decided.

Instructions (2) and (3) are unproblematic, but we must verify that O can play as required by instruction (1). Initially, $\vdash \Gamma'$ is $\vdash \Gamma$, which is unprovable, by assumption. If $\vdash \Gamma'$ is unprovable at some point during the play, and if P then makes a phase 1 move, then $\vdash \Gamma'$ will still be unprovable after this move. Indeed a phase 1 move of P replaces a component of the form $A \vee B$ with A or with B, so, up to order of components, $\vdash \Gamma'$ before the move was $\vdash \Delta, A \vee B$ and $\vdash \Gamma'$ after the move is either $\vdash\Delta, A$ or $\vdash\Delta, B$. But if either $\vdash\Delta, A$ or $\vdash\Delta, B$ were provable, then, by rule ($\vee$), $\vdash\Delta, A \vee B$ would also be provable, a contradiction. Thus, phase 1 moves of P cannot make $\vdash\Gamma'$ provable. A phase 1 move of O changes $\vdash\Gamma'$ from $\vdash\Delta, A \wedge B$ to $\vdash\Delta, A$ or to $\vdash\Delta, B$. By rule ($\wedge$), if $\vdash\Delta, A \wedge B$ is unprovable, then so is at least one of $\vdash\Delta, A$ and $\vdash\Delta, B$. So O can make his phase 1 moves in accordance with instruction (1).

Consider a particular play of Γ where P follows strategy σ while O obeys instructions (1)–(3) above. By the preceding discussion of (1), $\vdash\Gamma'$ never becomes provable. In particular, by rule ($\top$), the literal $\top$ never occurs in $\vdash\Gamma'$. Also, by the logical axioms and weakening, the literals in Γ' never include both a propositional variable p and its negation $\sim p$.

For each occurrence of a literal p (respectively, $\sim p$) that eventually appears in Γ', if infinitely many moves are made in that component of Γ', let $x \in {}^{\omega}\{0, 1\}$ be the sequence of phase 2 moves in that component, and declare x to be out of (respectively, in) G_p. As we noticed earlier, instruction (3) ensures that the x's involved here are different from the previously decided x's, so the decisions just made (for stage σ) do not conflict with earlier decisions. Nor do they conflict with each other, even if the same x arises in several components, for p and $\sim p$ cannot both occur in Γ'. If (as is likely) some x has just been put into or out of certain G_p's but remains undecided for other G_q's, then make these other decisions arbitrarily.

This completes stage σ in the construction of the sets G_p. Notice that the particular play of Γ used in the construction, with P following σ while O follows (1)–(3), is won by O. Indeed, O won components of the form p (respectively, $\sim p$) where infinitely many moves were made, because we put the corresponding x's out of (respectively, into) G_p; O wins components of the form $\perp$ automatically (if infinitely many moves are made there); and there are no componnts of the form $\top$. So every component where infinitely many moves are made is won by O. Thus, σ is not a winning strategy for P in Γ.

After the inductive construction of the G_p's in complete (with arbitrary conventions for any x's not decided at any stage), we have a game interpretation for the propositional variables in Γ such that no strategy for P wins Γ. Thus $\vdash\Gamma$ is not game-semantically valid. $\square$

Remark. In the special case where Γ consists of two formulas, one containing no negated variables, the other containing no unnegated variables, and netiher containing $\top$ or $\perp$ (so Γ amounts to $A \vdash B$, where A and B are built from variables by means of $\wedge$ and $\vee$), the Additive Completeness Theorem is a special case of [2, Theorem 4]. The proof there is essentially the same as the proof just given. Notice that, in this special case, the deduction rules for $\wedge$ and $\vee$ are equivalent to Whitman's description [11] of the inequalities that hold in free lattices.

5. The multiplicative fragment

A *multiplicative formula* is one built from propositional variables, negated propositional variables, $\top$, and $\perp$ by means of the *multiplicative connectives* $\otimes$ and $\breve{\otimes}$. A sequent is *multiplicative* if every formula in it is. In this section, we shall characterize, in terms of classical propositional logic, the valid multiplicative sequents.

We observe first that validity of a sequent $\vdash C_1, \ldots, C_r$ is equivalent to validity of the formula $C_1 \breve{\otimes} \cdots \breve{\otimes} C_r$. Since $\breve{\otimes}$ is a multiplicative connective, we need only characterize validity of multiplicative formulas.

Multiplicative formulas can be read as formulas in classical propositional logic, with $\otimes$ and $\breve{\otimes}$ read as conjunction and disjunction, respectively (and $\sim$, $\top$ and $\perp$ read as a negation, truth and falsity, respectively). We do not speak of translating multiplicative formulas into the standard symbolism of classical logic (replacing $\otimes$ with $\wedge$, etc.), as this might lead to confusion with the additive connectives. Instead, we pretend that classical logic is formulated with $\sim$, $\otimes$ and $\breve{\otimes}$ as its connectives, so that multiplicative formulas of affine logic *are* also formulas of classical logic. So it makes sense to speak of a multiplicative formula being a tautology, or of a positive or negative occurrence of a variable in a multiplicative formula, or of any other concept familiar from classical propositional logic. In particular, by an *instance* or (substitution instance) of a multiplicative formula A, we mean a formula obtained by replacing the propositional variables in A uniformly by some multiplicative formulas; if negations of nonatomic formulas result, these are to be understood according to the definition of such formulas in linear logic, so that the result is a multiplicative formula.

We call a multiplicative formula *binary* if each propositional variable has at most one positive and one negative occurrence.

Multiplicative Validity Theorem. *A multiplicative formula is game-semantically valid if and only if it is an instance of a binary tautology.*

Proof. 'If'. Since any instance of a valid formula is clearly also valid, it suffices to prove that all binary tautologies are valid. In fact, it will suffice to consider binary tautologies in which every variable occurs exactly twice (once positively and once negatively) and $\top$ and $\perp$ do not occur. For brevity, we call such tautologies *special*.

To see that we may confine attention to special tautologies, suppose these were known to be valid, and consider an arbitrary binary tautology C. Starting with C, we repeatedly replace subformulas according to the following rules as long as any of the rules apply.

(1) Any propositional variable having only a positive occurrence (respectively, only a negative occurrence) is replaced by $\perp$ (respectively, $\top$).

(2) A subformula of the form $\top \otimes A$ or $\bot \bar{\otimes} A$ is replaced by A.

(3) A subformula of the form $\top \bar{\otimes} A$ (respectively, $\bot \otimes A$) is replaced by $\top$ (respectively $\bot$).

It is clear that all the formulas produced are binary tautologies, that the process terminates (because each replacement reduces the total number of occurrences of propositional variables, $\otimes$, and $\bar{\otimes}$), and that the final result C' is either a special tuatology or simply $\top$. So by our assumption (and the obvious validity of $\top$), C' is valid. We intend to infer from this that C is valid, as desired. It suffices to consider the replacement process one step at a time and show that, if a valid formula B' is obtained from a formula B by a single replacement of the form (1), (2) or (3), then B is also valid. In fact, we shall show that $\vdash B, \sim B'$ is provable; since this and $\vdash B'$ yield $\vdash B$ by the cut rule, the desired conclusion will follow by the Soundness Theorem. To show that $\vdash B, \sim B'$ is provable, we use the following elementary lemma.

Lemma 1. *Let B and B' be multiplicative formulas. Suppose B' is obtained from B by replacing positive occurrences of a subformula X by X'. Then $\vdash B, \sim B'$ is deducible from $\vdash X, \sim X'$.*

Proof. The result is obvious if B is X or if no occurrences of X are replaced. If B is $B_1 \otimes B_2 \neq X$, then we may assume inductively that both $\vdash B_1, \sim B_1'$ and $\vdash B_2, \sim B_2'$ are deducible from $\vdash X, \sim X'$. But from these we obtain by the $\otimes$ rule $\vdash B_1 \otimes B_2, \sim B_1', \sim B_2'$ and then by the $\bar{\otimes}$ rule $\vdash B_1 \otimes B_2, \sim B_1' \bar{\otimes} \sim B_2'$, which is $\vdash B, \sim B'$. The case of $\bar{\otimes}$ is handled the same way. □ (Lemma 1).

The lemma applies to the situation at hand, namely replacements according to (1)–(3), provided we remember that, in a multiplicative formula, compound subformulas as in (2) and (3) can occur only positively, and provided we regard the replacement of a negative p by $\top$ in (1) as replacement of the positively occurring $\sim p$ by $\bot$. (A negative occurrence of p in a multiplicative formula can only be in the context $\sim p$.) We must check that $\vdash X, \sim X'$ is provable whenever X is replaced by X', i.e., we must check provability of

$$\vdash A, \sim\bot, \qquad \vdash \top \otimes A, \sim A, \qquad \vdash \bot \bar{\otimes} A, \sim A,$$

$$\vdash \top \bar{\otimes} A, \sim\top, \qquad \vdash \bot \otimes A, \sim\bot,$$

but all of these are easy. This completes the verification that we can safely confine our attention to special tautologies.

Fix a special tautology C and fix a game interpretation. We assume that the games assigned to variables are strict games. (As usual, we shall use the same symbol for a formula and for the game assigned to it by this interpretation.) We shall complete the proof of the 'if' half of the theorem by describing a winning strategy for P in the game C.

By the literals in C, we shall mean the occurrences of variables and negated variables from which C is built by $\otimes$ and $\bar{\otimes}$. Thus, a negative occurrence of a variable p does not count as a literal; rather its context $\sim p$ is a literal. As C is special, the literals come in pairs, each containing p and $\sim p$ for some variable p. The game C consists of subgames, one for each literal, and the rules governing who is to move in a given position and who has won a given play are summarized by the truth-table description in Section 2. It will be convenient to think in terms of the parse tree of C, with C at the root, literals at the leaves, and $\otimes$ or $\bar{\otimes}$ labeling the internal nodes. A position or play gives a labeling of the leaves as in Section 2 (True if P has won or O is to move, False if O has won or P is to move), and the truth values propagate from the leaves to the other nodes according to the truth tables for conjunction ($\otimes$) and disjunction ($\bar{\otimes}$). Since C is a tautology, the root will have label True *provided* each pair of literals, p and $\sim p$, have opposite truth values, so that the labeling is really a truth assignment (in the sense of ordinary logic). In fact, since only positive connectives are used in the tree, the root will also have label True if some p and $\sim p$ are both labeled True. But it is, of course, entirely possible that p and $\sim p$ are both labeled False in a particular position or play (e.g., if P is to move in both of these subgames), and then the root C may well be labeled False.

Whenever P is to move at a certain position in game C, i.e., when C is labeled False, his move consists of choosing a path through the parse tree from the root to a leaf l, such that all the nodes along the path are labeled False, and then making a move in l. (The choice of path will involve a real choice at $\bar{\otimes}$ nodes, where a False label means that both successors are also labeled False. At $\otimes$ nodes, usually (i.e., expect at the first visit to this node) only one successor will be labeled False; see the discussion of $\otimes$ in Section 2.)

The essential idea for P's winning strategy is to make sure that the plays in paired subgames p and $\sim p$ are identical. Since he plays opposite roles in these two subgames, he will (if infinitely many moves are made in each of them) win one and lose the other, so the final labeling of the tree will be a real truth assignment, C will be labeled True, and so P will win C. We must still show that P can carry out the proposed strategy and that he will win even if some of the subgames are unfinished (i.e., have only finitely many moves made in them). For this purpose, we must describe the strategy in somewhat more detail.

P is to ensure that, at each moment during the play of the game, for each pair p, $\sim p$ of literals, either the positions (=sequences of moves already played) in p and $\sim p$ are identical or else one of them equals the other plus one subsequent move made by O. This condition is certainly satisfied initially, as all positions are initially the empty sequence.

Furthermore, this condition cannot be destroyed by a move of O. To see this, suppose the condition is satisfied at a certain moment, which we call 'before', and that O then moves, say in subgame p. If the positions in p and $\sim p$ were identical before, then afterward the position in p is that in $\sim p$ plus the single move just made by O, so the condition remains satisfied. If the position in p before were that in $\sim p$ plus a move by O, then O could not have moved in p as it would be P's turn there.

Finally, if the position in $\sim p$ before were that in p plus a move of O, then the presence of that move of O in $\sim p$ means that the position without the move, the 'before' position in p, is a position with O to move in $\sim P$, hence is a position with P to move in p; so again O could not move in p. This shows that a move of O in p (or, for symmetrical reasons, in $\sim p$) cannot destroy the condition that P is trying to maintain.

The preceding discussion shows furthermore that, as long as the condition is satisfied, whenever the positions in p and $\sim p$ are different, it is P's turn to move in both of them, whereas of course if the positions in p and $\sim p$ are equal, then P is to move in one and O in the other.

We show next that if the condition holds at a certain position and if P is to move, then he can move so as to maintain the condition. More precisely, we show that there is a literal l such that (1) the path from l to the root in the parse tree of C is labeled entirely with False, so that P can legally move in the subgame l, and (2) the position in $\sim l$ is one move longer than that in l. Here (2) means that O has made a move in $\sim l$ which P can simply copy in l, thereby maintaining the desired condition. We call a literal l *good* at a given position if (1) and (2) hold.

Lemma 2. *Let the parse tree of C be labeled, using the appropriate truth tables at the interior nodes but arbitrary labels at the leaves. If C is labeled False, then there is a pair p, $\sim p$ of literals such that the paths joining them to the root are both labeled entirely with False.*

Proof. Suppose we had a labeling that is a counterexample to the lemma. We saw earlier that, because C is a tautology yet labeled False and because the connectives at interior nodes are monotone, there must be a pair of literals p, $\sim p$ both labeled False. As the labeling is a counter-example to the lemma, the path from one of p, $\sim p$, say p, to the root contains a label True. Alter the labeling of the leaves by changing p from False to True, and consider the resulting new labeling of the parse tree (in accordance with the connectives, as always). The change at the leaf p can affect only the labels along the path from p to the root and indeed can only increase these labels (i.e., change False to True) because the connectives are monotone. There was already a True label somewhere on this path. That label will therefore be unchanged. But then all labels between that True and the root are also unaffected by our change at p. In particular, the root C retains its previous label, False. This fact, and the fact that no True has been changed to False, means that our modified labeling is still a counterexample to the lemma. It has strictly fewer leaves labeled False than the original counterexample. So, by repeating the process, we have a contradiction. □(Lemma 2)

The lemma shows that whenever P is to move, thre is a pair of subgames p, $\sim p$ such that P can legally move in either of them. In particular, the positions in p and $\sim p$ cannot be identical, for then it would be P's move in only one of them. If the condition that P wants to maintain holds, then the position in one of p, $\sim p$ is one move longer than in the other. But then the latter is a good subgame.

We have seen that if P is to move and the condition holds, then there is a good subgame and P can move in any good subgame so as to maintain the condition. Once the good subgame is chosen, the appropriate move for P is unique; it consists of copying O's last move in the paired subgame. We specify P's strategy more completely by requiring that if there are several good subgames, then he should move in one where the sequence of previous moves is as short as possible. If several are equally short, choose arbitrarily between them.

Having described P's strategy and verified its feasibility, we show that it is a winning strategy. Suppose x were a play in which P used this strategy but lost. For each literal l, we let $(x)_l$ be the subsequence of moves in x in subgame l. We indicated a proof earlier that P wins if each $(x)_l$ is infinite; the possibility of some $(x)_l$'s being finite necessitates a subtler argument.

Label the nodes of the parse tree of C in the usual way for the play x. As P lost, C is labeled False. Apply Lemma 2 to obtain p and $\sim p$ such that the paths joining these literals to the root are both labeled entirely with False. Then $(x)_p$ and $(x)_{\sim p}$ cannot both be infinite, for then they would be identical, thanks to P's strategy, and would have opposite labels (by definition of $\sim$). Nor can one be finite and the other infinite, for P's strategy ensures that their lengths never differ by more than one. So both are finite, and one, say $(x)_p$ without loss of generality, is one move shorter than the other. Fix such a p.

While playing the game C, leading to the play x, the players arrive after finitely many moves at a position with the following properties for every literal l.

(1) If $(x)_l$ is finite, then all moves that will ever be made in subgame l have already been made. (Subgames that will remain unfinished have been permanently abandoned.)

(2) If $(x)_l$ is infinite, then the number of moves already made in subgame l exceeds the number that have been (or ever will be) made in any finite $(x)_{l'}$.

Of course, once (1) and (2) hold, they continue to hold at all later positions. Call a position *1,2-late* if conditions (1) and (2) hold.

Consider any 1,2-late position with P to move. By (1), P's move is in some l such that $(x)_l$ is infinite. But, by his strategy, P moves in a good literal where the current move sequence is as short as possible. By (2), the current move sequence in p is shorter than in l, since $(x)_p$ is finite and $(x)_l$ infinite. So if p were good, P would not have moved in l. Therefore, p is not good. But the position in $\sim p$ (which is $(x)_{\sim p}$ by (1)) is one move longer than the position in p (which is $(x)_p$), because of our choice of p. So the only way for p not to be good is that, on the path from p to the root, there is a node labeled True.

We claim that, from some moment on, the positions leading to the play x satisfy

(3) Some node between the root and p is labeled True.

We have just shown this for positions where P is to move. When P moves, however, labels only increase. (One leaf goes from False (with P to move) to True (with O to move), the other leaves are unchanged, and internal nodes are given by monotone connectives.) So P's move cannot destroy (3). Thus, all 1,2-late positions, except possibly the first, are in fact 1,2,3-late (in the obvious sense).

At any 1,2,3-late position, consider the location of the True label nearest p on the path from p to the root. Consider how this location changes as the play proceeds. A move of P is always at a good literal, is therefore never at a leaf beyond this (or any) True label, and therefore never affects either this True label or the False labels between it and p. So the location is unchanged when P moves. A move of O can only decrease labels (from True to False), by the dual of the argument in the preceding paragraph. So the False labels between p and the location being studied are not changed; the True label at this location may change to False, and in this case the new location of the True nearest p (which still exists by (3)) is nearer the root. In summary, the location of the True nearest p moves, at 1,2,3-late stages of the play, only toward the root. As the path on which it moves is finite, it must eventually stop moving. Let X be its final location. Thus, at all sufficiently late stages of the play, we have

(4) X is labeled True.

At such stages, P will never move in literals that are beyond X in the parse tree (i.e., are subformulas of X), because he only moves in good literals and these have only False labels between them and the root. Therefore, at 1,2,3,4-late stages, O moves at most once in any literal beyond X, for once he moves in such a literal, it is P's turn there, and it remains P's turn there forever since P does not move there any more. Therefore, from some stage on, we have

(5) No moves are made in literals beyond X.

But this means that the labeling of leaves beyond X does not change any more. This labeling is therefore the same for any 1,2,3,4,5-late stage as for the final (infinite) play x. The same therefore holds for the label of X. But X is labeled True at a 1,2,3,4,5-late stage, by (4), and is labeled False for x, by our choice of p. This contradiction shows that, when he uses the strategy we described, P cannot lose C. This completes the proof of the 'if' half of the theorem.

'Only if'. Let the multiplicative formula C not be an instance of a binary tautology. We shall construct, for each variable p occurring in C, a strict game, with set of moves $\{0, 1\}$ and with P moving first, such that C is not true in this game interpretation. As in the proof of the Additive Completeness Theorem, there are c (the cardinality of the continuum) strategies for P in C, and we well-order them so that each has fewer than c predecessors. We define the sets $G_p \subseteq {}^{\omega}\{0, 1\}$ of plays won by P in the games associated to the variables p, by transfinite induction along this well-ordering. At the stage corresponding to strategy σ, we decide membership in the G_p's for finitely many $x \in {}^{\omega}\{0, 1\}$, so as to ensure that σ is not a winning strategy for P in C.

For a fixed σ, here is how stage σ of the construction is to proceed. For each occurrence l of a literal (i.e., a positive occurrence of a variable or negated variable) in C, choose a different sequence $z_l \in {}^{\omega}\{0, 1\}$ that does not occur as the sequence of moves of either player in any previously decided x. Note that the same variable or negated variable may have several occurrences, corresponding to several subgames of C; these count as different literals l and have different z_l's assigned to them.

Construct a play of C as follows. P uses σ. O chooses, at each of his moves, a subgame (=occurrence of literal) l in which (1) he can legally move (i.e., the current labels between l and the root of the parse tree are all True), and (2) the current position in l contains as few moves as possible, subject to (1). In l, O uses z_l as his sequence of moves.

If l and l' are two occurrences of the same literal, and if the play x that we have just produced has infinite subsequences $(x)_l$ and $(x)_{l'}$ of moves in these two subgames, then $(x)_l \neq (x)_{l'}$ because O's moves in these two plays are $z_l \neq z_{l'}$.

On the other hand, it is possible that l and l' are occurrences of p and $\sim p$, respectively, and that $(x)_l = (x)_{l'}$ and these subsequences are infinite. Indeed, O's moves z_l in $(x)_l$ might match P's moves in $(x)_{l'}$ (since l' is the negation of l, the players have reversed roles) and vice versa; for example, P's strategy σ might involve copying O's moves between l and l'. If this occurs, we say that l and l' are *matched*. Notice that, by the preceding paragraph, any l is matched with at most one l'.

Consider the formula C^* obtained from C by changing all occurrences of variables to distinct variables except that matched occurrences of literals p and $\sim p$ retain the same variable. Clearly, C is an instance of C^* and C^* is binary. But we assumed that C is not an instance of a binary tautology. So C^* is not a tautology. Fix a truth assignment making C^* false.

We regard this truth assignment as assigning truth values as labels to the leaves of the parse tree of C. This labeling, which we extend in the usual way to the whole parse tree and call the *preferred* labeling, need not be a real truth assignment for C, since different occurrences of the same variable in C became different variables in C^*, and may thus have received different truth values. However, if l and l' are matched literals in C, then one remained the negation of the other in C^*, so they received opposite truth values. Summarizing the properties of the preferred labeling that we shall need later, we have

(1) matched literals have opposite truth values, and

(2) the root C is labeled False.

For each literal occurrence l such that $(x)_l$ is infinite in the play x described above, we note that $(x)_l$ is a member of ${}^{\omega}\{0, 1\}$ that was not decided at any previous stage of the definition of the G_p's. Indeed, z_l, the subsequence of O's moves in $(x)_l$, was chosen to differ from the subsequence of either player's moves in any previously decided sequence. We can therefore freely choose which G_p's to put any such $(x)_l$ into. We use this freedom to try to make the labeling of the parse tree associated to x match the preferred labeling. Thus, if l is an occurrence of p (respectively, $\sim p$) and is labeled True (respectively, False) in the preferred labeling, then we put $(x)_l$ into G_p. On the other hand, if l is an occurrence of p (respectively, $\sim p$) and is labeled False (respectively, True) in the preferred labeling, then we decide that $(x)_l$ is not to be in G_p. (Other decisions, about membership of $(x)_l$ in G_p when l is neither p nor $\sim p$, can be made arbitrarily.) The decisions just described do not conflict with one another. Indeed, the only possibility for conflict would be if $(x)_l = (x)_{l'}$ for two distinct occurrences of literals, l and l'. But then l and l' are matched and therefore get opposite truth values in the preferred labeling. Since one of l and l' is an occurrence of some p and the other of $\sim p$, opposite labels ensure the same decision about $(x)_l \in G_p$ and $(x)_{l'} \in G_p$.

This completes the description of stage σ of the construction of the G_p's. It remains to verify that this stage ensures that σ is not a winning strategy for P in the game C. For this purpose, we consider the play x used for stage σ. It was defined as a play where P uses strategy σ, so we need only check that O wins this play.

If the sequence $(x)_l$ of moves in (the game corresponding to) l were infinite for every occurrence l of a literal in C, then our task would be trivial. The decisions made at stage σ would ensure that the labeling of the parse tree of C associated to the play x agrees, at all leaves and therefore at all other nodes as well, with the preferred labeling. Since the latter makes the root false, it follows (by the truth-table descriptions of $\otimes$ and $\tilde{\otimes}$ games) that O wins the play x. Unfortunately, there is no reason to expect each $(x)_l$ to be infinite, and a finite $(x)_l$ may give l a label (as always, True if O is to move, False if P is to move) different from the preferred label. So a subtler argument is needed. This argument is quite similar to one already used in the 'if' part of this proof, so we omit some details.

In the play of the game C, at all sufficiently late stages, we have, for each literal occurrence l,

(1) if $(x)_l$ is finite, then all moves that will ever be made in subgame l have already been made, and

(2) if $(x)_l$ is infinite, then the number of moves already made in subgame l exceeds the length of every finite $(x)_{l'}$.

At moves of O this late in the game, he does not move in any subgame l for which $(x)_l$ is finite (by (1)), but he would move in such a subgame if he legally could (by the second clause in the description of how O chooses his moves in x, and by (2)). So, when O is to move this late in the game, the path from each such

l to the root must contain a label False. A move of O only decreases labels, so such a False is still present afterward, when P is to move. So, at all sufficiently late stages

(3) if $(x)_l$ is finite, then the path from l to the root contains at least one label False.

If we temporarily fix an l such that $(x)_l$ is finite and if we consider, on the path from l to the root, the False nearest l, we see that its location is unaffected by moves of O and can move only toward the root at moves of P. So this False is always at the same location X from some stage on. At such late stages, O will never move in subgames l beyond X in the parse tree (i.e., occurrences of literals within the subformula X), and therefore P will move there only finitely often. Waiting until all these moves have been made, we see that, at all sufficiently late stages in the play, nothing happens beyond X, so the labeling of the subtree with root X remains unchanged. In particular, as X was chosen to have label False at all sufficiently late stages, it also has label False in the final labeling associated to the play x.

We have shown that every l for which $(x)_l$ is finite is within a subformula $X(l)$ whose final label is False. We complete the proof by considering the following three labelings of the parse tree of C.

(a) The preferred labeling.

(b) The final labeling associated to the play x.

(c) The labeling that agrees with (a) and (b) at all l for which $(x)_l$ is infinite but assigns False to all l for which $(x)_l$ is finite.

Notice that (c) makes sense, because we already know that (a) and (b) agree at l when $(x)_l$ is infinite. We also know that (a) labels the root C with False; by monotonicity of $\otimes$ and $\bar{\otimes}$, (c) also labels the root with False. Now consider what happens if we change labeling (b) to (c). The only changes at leaves of the parse tree are decreases (from True to False) at some l's for which $(x)_l$ is finite. The only changes at interior nodes are decreases (by monotonicity again) along the paths from such l's to the root. But every such path contains a node $X(l)$ that was already labeled False in (b) and that is therefore unaffected by the decreases in going from (b) to (c). But if the change at l does not affect the label at $X(l)$, it cannot affect labels nearer the root. In particular, C has the same label in (b) as in (c), and we already know that the latter is False. So C is False in the labeling associated to x, i.e., O wins the play x.

This shows the stage σ of our construction prevents σ from being a winning strategy for P in C. The whole construction therefore ensures that P has no winning strategy in (this interpretation of) C, and so C is not valid. □

We close this section by pointing out a curious consequence of the theorem just proved and the soundness of the cut rule. If $A \bar{\otimes} C$ and $B \bar{\otimes} {\sim}C$ are instances of binary tautologies, then so is $A \bar{\otimes} B$. Notice that this consequence has (except for the notation $\bar{\otimes}$ for disjunction) nothing to do with linear logic or with games;

it is a result about classical propositional logic. Notice also that it is not entirely trivial, for the binary tautologies of which $A \breve{\otimes} C$ and $B \breve{\otimes} {\sim}C$ are instances need not have the form $A' \breve{\otimes} C'$ and $B' \breve{\otimes} {\sim}C'$; the C parts need not match. Nevertheless, the result has (as might be expected) a direct proof, which we leave as an exercise for the reader with the hint that it is related to the involution principle of [5].

6. Incompleteness and the meaning of tensor products

The Multiplicative Validity Theorem proved in the preceding section enables us to show that the completeness theorem fails for the multiplicative fragment. Specifically, the formula

$$[({\sim}A \breve{\otimes} {\sim}A) \otimes ({\sim}A \breve{\otimes} {\sim}A)] \breve{\otimes} [(A \breve{\otimes} A) \otimes (A \breve{\otimes} A)] \tag{1}$$

is an instance of (in customary notation)

$$[(A \wedge B) \vee (C \wedge D)] \rightarrow [(A \vee C) \wedge (B \vee D)],$$

a binary tautology. Hence, (1) is game-semantically valid. But the sequent $\vdash$(1) is not provable in affine logic. Probably the easiest way to see this is to use the Cut-Elimination Theorem and observe that, in a cut-free proof of $\vdash$(1), the last step must be a $\breve{\otimes}$ inference from

$$\vdash ({\sim}A \breve{\otimes} {\sim}A) \otimes ({\sim}A \breve{\otimes} {\sim}A), (A \breve{\otimes} A) \otimes (A \breve{\otimes} A),$$

and the step leading to this must be a $\otimes$ inference. But a $\otimes$ inference leading to this sequent has as one of its premises $\vdash A \breve{\otimes} A$ or $\vdash {\sim}A \breve{\otimes} {\sim}A$, neither of which is provable because neither is a tautology.

The unprovability of $\vdash$(1) or equivalently of

$$(A \otimes A) \breve{\otimes} (A \otimes A) \vdash (A \breve{\otimes} A) \otimes (A \breve{\otimes} A) \tag{2}$$

can also be verified using the phase semantics of [6]. Specifically, consider the six-element commutative monoid $\{1, a, a^2, b, b^2, 0\}$ with multiplication defined in the obvious way (e.g., $1 \cdot x = x$, $0 \cdot x = 0$, $a \cdot a = a^2$) subject to $ab = a^3 = b^3 = 0$, and let $\perp = \{0\}$. This is a model for affine logic because $\perp$ is an ideal. If A is interpreted as the ideal (a) generated by a, then the left and right sides of (2) are interpreted as (a, b) and (a^2, b^2), respectively. As the former is not a subset of the latter, (2) is not phase-semantically valid, and hence not provable.

The failure of completeness for the multiplicative fragment suggests that the 'meanings' given to the multiplicative connectives by the inference rules of linear logic do not match the 'meanings' given to the same connectives by game semantics. To analyze the difference, it is useful to regard inference rules as a sort of game (see [1, Section 1.5]) which can then be directly compared to game semantics.

The game associated to a sequent $\vdash \Gamma$ is played as follows. P claims that $\vdash \Gamma$ is provable, and as justification for this claim he specifies, as his first move, a particular inference having $\vdash \Gamma$ as its conclusion; he thereby implicitly claims that all the premises of this inference are provable. O, on the other hand, claims $\vdash \Gamma$ is unprovable, so he cannot accept the provability of all the premises of P's first move. As his first move, O chooses one of these premises $\vdash \Delta$ and disputes P's claim that it is provable. From this point on, the players proceed as above, but with $\vdash \Delta$ in place of $\vdash \Gamma$. If a player is unable to move at some stage (in P's case because no inference has the desired conclusion; in O's case because P's last move was an axiom, i.e., a zero-premise inference), then that player loses. If the game continues forever, O wins. Since the game is open for P (i.e., if P wins, he does so at some finite stage), it is determined [4]. Player P has a winning strategy if and only if $\vdash \Gamma$ is provable.

We propose to compare the 'proof-theoretic' game, using a cut-free axiomatization for affine logic, for the sequent $\vdash C_1, \ldots, C_r$ with the game $\tilde{C}_1 \otimes \cdots \otimes \tilde{C}_r$ associated to this sequent by a game interpretation as defined in Section 2. And, in view of the purpose of this comparison, we intend to concentrate on the handling of connectives, particularly multiplicative connectives. Let us consider P's opening move and O's response in the proof-theoretic game for a sequent $\vdash C_1, \ldots, C_r$. For example, P's opening move might be to cite a logical axiom (in which case O has no response, so P wins) or weakening (in which case O must respond with $\vdash C_2, \ldots, C_{r-1}$) or exchange (in which case O must respond with the permuted version of $\vdash C_1, \ldots, C_r$ specified by P). These possibilities have no direct counterpart in the game semantics of $C_1 \bar{\otimes} \cdots \bar{\otimes} C_r$, nor do they shed light on the meanings of connectives (except for $\sim$). More interesting situations arise when P invokes one of the rules associated to connectives. In such a case, P can arbitrarily choose any compound formula among the C_i as the principal formula of an inference; P's move and O's reply replace this formula with another (or two others in the case of Contraction $\tilde{R}$) and may remove some of the side formulas in the case of $\otimes$. P's freedom to choose C_i and, if he wishes, to choose different C_j's on subsequent moves corresponds to his freedom to choose which subgame of $C_1 \bar{\otimes} \cdots \bar{\otimes} C_r$ to move in at each of his moves. In other words, the commas in a sequent are treated by the proof-theoretic semantics similarly to the game interpretation of $\bar{\otimes}$.

If P chooses a C_i of the form $A \wedge B$, then his move provides the two premises $\vdash \Delta, A$ and $\vdash \Delta, B$ (where Δ consists of the C_j's other than C_i and where we have ignored exchanges); O must then choose one of these, i.e., he must decide whether to replace $A \wedge B$ with A or with B. This corresponds precisely to the opening move in the game interpretation of $\wedge$.

If P chooses a C_i of the form $A \vee B$, then he has the option of providing $\vdash \Delta, A$ or $\vdash \Delta, B$ as the (only) premise, and O must accept this choice. So in this case, P chooses whether to replace $A \vee B$ with A or with B, just as in the game interpretation of $\vee$.

If P chooses a C_i of the form $A \mathbin{\bar{\otimes}} B$, then his and O's moves together replace it with the two formulas A, B. We have already seen that the proof-theoretic game treats commas like the game interpretation of $\bar{\otimes}$. So it treats $\bar{\otimes}$ the same way.

If P chooses a C_i of the form $A \otimes B$, however, the situation is more complicated. According to the rule of inference $\otimes, P$ must partition the set of side formulas into two sets, Γ and Δ, and then O is allowed to choose between $\vdash \Gamma, A$ and $\vdash \Delta, B$. No such partitioning occurs in the game interpretation of $\otimes$, so it appears that the meanings of $\otimes$ in the proof rules and in game semantics do not quite agree. (It is ironic that the discrepancy should concern the one connective for which [2] and [6] use the same symbol.)

What happens in the game interpretation for formulas of the form $A \otimes B$ in contexts like $\Gamma \mathbin{\bar{\otimes}} \Delta \mathbin{\bar{\otimes}} (A \otimes B)$? Since O may switch freely between A and B at his moves in the component $A \otimes B$, P must, if he is to win, be prepared to answer O's moves in both A and B. A splitting into $\Gamma \mathbin{\bar{\otimes}} A$ and $\Delta \mathbin{\bar{\otimes}} B$ is a rather simple way for P to do this; he decides that O's moves in A should be answered in A or in Γ, while O's moves in B should be answered in B or in Δ. (See the proof of the $\otimes$ case of the Soundness Theorem for details.)

But there are more complicated ways for P to handle $A \otimes B$, and our counterexample to multiplicative completeness rests on just such a possibility. It will be convenient to consider the counterexample in its binary form

$$\vdash (\sim A \mathbin{\bar{\otimes}} \sim B) \otimes (\sim C \mathbin{\bar{\otimes}} \sim D),\ (A \mathbin{\bar{\otimes}} C) \otimes (B \mathbin{\bar{\otimes}} D).$$

P can handle the first $\otimes$ here as follows. When O plays in $(\sim A \mathbin{\bar{\otimes}} \sim B)$, P responds using either the A component of $A \mathbin{\bar{\otimes}} C$ (along with the $\sim A$ component in $\sim A \mathbin{\bar{\otimes}} \sim B$) or the B component of $B \mathbin{\bar{\otimes}} D$ (along with the $\sim B$ component in $\sim A \mathbin{\bar{\otimes}} \sim B$). Which of the two options he takes depends on which component of the second $\otimes$ has been chosen by O. (The other case, where O plays in $\sim C \mathbin{\bar{\otimes}} \sim D$, is handled similarly.) In a sense, P is partitioning the side formula $(A \mathbin{\bar{\otimes}} C) \otimes (B \mathbin{\bar{\otimes}} D)$ into a part (Γ) to be used with $\sim A \mathbin{\bar{\otimes}} \sim B$ and a part (Δ) to be used with $\sim C \mathbin{\bar{\otimes}} \sim D$, but (in contrast to what is required in the proof-theoretic game) Γ and Δ are not simple constituents, the two sides of a $\bar{\otimes}$ or rather of a comma, but noncontiguous fragments.

The $\otimes$ of game semantics allows P more (and subtler) ways to defend a formula than the $\otimes$ of the proof system. In other words, my $A \otimes B$ is a weaker (=easier to defend) assertion than Girard's $A \otimes B$. But it is not *too* weak. Specificially, my $A \otimes B$ is strong enough to serve as the negation of $\sim A \mathbin{\bar{\otimes}} \sim B$ for the purposes of the cut rule; the inference

$$\frac{\vdash \Gamma, A \otimes B \quad \vdash \Delta, \sim A \mathbin{\bar{\otimes}} \sim B}{\vdash \Gamma, \Delta}$$

is, as we saw in Section 3, game-semantically correct. This situation seems peculiar, since Girard's rule of inference for $\otimes$ is exactly what is needed for cut-elimination with the $\tilde{\otimes}$ rule. So how can cut still work for my weaker version of $\otimes$? The answer seems to be that there is no rule of inference for my version of $\otimes$; more precisely, no rules of inference for $\otimes$ will make the proof-theoretic game behave like the game-semantical $\otimes$. Less precisely (and less truthfully, perhaps), my $\otimes$ is entirely foreign to proof theory. Yet, from the game-theoretic point of view, my $\otimes$ is very natural, being obtained by reversing the roles of the players in the construction $\tilde{\otimes}$ (which interprets the commas in sequents), while the proof-theoretic $\otimes$ seems to put an artificial restriction on P.

7. Quantifiers

Universal and existential quantification of a formula amount to (possibly infinite) conjunction and disjunction of instances of that formula. Girard's inference rules for quantifiers [6],

$$\frac{\vdash \Gamma, A}{\vdash \Gamma, \bigwedge x A}, \tag{$\bigwedge$}$$

where x is not free in Γ, and

$$\frac{\vdash \Gamma, A[t/x]}{\vdash \Gamma, \bigvee A}, \tag{$\bigvee$}$$

are analogous to the rules for the additive connectives,

$$\frac{\vdash \Gamma, A \quad \vdash \Gamma, B}{\vdash \Gamma, A \wedge B}, \tag{$\wedge$}$$

$$\frac{\vdash \Gamma, A}{\vdash \Gamma, A \vee B} \quad \text{and} \quad \frac{\vdash \Gamma, B}{\vdash \Gamma, A \vee B}. \tag{$\vee$}$$

In ($\bigwedge$), the premise can be viewed as giving each instance of A (each 'conjunct' in $\bigwedge x A$), always with the same side formulas Γ, which is similar to what happens in ($\wedge$) and quite different from ($\otimes$). Again, in ($\bigvee$), the premise has a single instance of A, analogously to ($\vee$) and not to ($\tilde{\otimes}$).

We digress for a moment to speculate about the possibility of 'multiplicative quantifiers' $\forall$ and $\exists$ related to $\bigwedge$ and $\bigvee$ as $\otimes$ and $\tilde{\otimes}$ are to $\wedge$ and $\vee$. There is a plausible analog of the $\otimes$ rule, namely

$$\frac{\vdash \Gamma, A}{\vdash \exists x \Gamma, \forall x A}, \tag{$\forall$}$$

where $\exists x \Gamma$ means the result of quantifying by $\exists x$ every formula in Γ. But I see no plausible analog of the $\tilde{\otimes}$ rule with the current notion of sequent. What would be needed is a way for all the instances of a formula to be 'joined by commas', like the various formulas in a sequent.

Such generalized sequents would also allow us to avoid using $\exists$ in the conclusion of the $\forall$ rule, which seems most desirable since a rule 'should not' introduce new connectives or quantifiers on the side formulas.

Returning to the 'sensible' additive quantifiers $\bigwedge$ and $\bigvee$, we introduce game semantics for them in analogy to the game semantics for $\wedge$ and $\vee$. Thus, a game-structure $\mathfrak{M}$ consists of (1) a nonempty universe of discourse M (or, for many-sorted logic, a universe for each sort), (2) functions to serve as interpretations of function symbols, and (3) for each predicate symbol R and each appropriate list $\boldsymbol{a}$ of elements of the universe, a game to serve as the interpretation of $R\boldsymbol{a}$. Given such a structure $\mathfrak{M}$ and given an assignment s of values (in the universe) to all variables, we interpret terms t as elements $t_{\mathfrak{M}}[s]$ of M as in the usual semantics of first-order logic, and we interpret formulas A as games $A_{\mathfrak{M}}[s]$ as follows. An atomic formula $R\boldsymbol{t}$ is interpreted as the game associated (by part (3) of the structure $\mathfrak{M}$) to $R(\boldsymbol{t}_{\mathfrak{M}}[s])$. The propositional connectives are interpreted as in Section 3 above.

$(\bigwedge xA)_{\mathfrak{M}}[s]$ and $(\bigvee xA)_{\mathfrak{M}}[s]$ are interpreted by first forming the family of instances $A_{\mathfrak{M}}[s \text{ but } x \mapsto m]$ for $m \in M$ (where 's but $x \mapsto m$' is the function that agrees with s except that it sends x to m) and then applying the game-theoretic operations of infinitary $\bigwedge$ and $\bigvee$ (cf. [2]) to obtain $\bigwedge_{m\in M}(A_{\mathfrak{M}}[s \text{ but } x \mapsto m])$ and $\bigvee_{m\in M}(A_{\mathfrak{M}}[s \text{ but } x \mapsto m])$. These infinitary operations are defined in general as follows. In $\bigwedge_{i\in I} A_i$ (respectively, $\bigvee_{i\in I} A_i$), the first move is by O (respectively, P) and consists of choosing some $i \in I$; subsequent moves constitute a play of A_i, and the winner is determined by the rules of A_i. Thus, P has a winning strategy in $\bigwedge_{i\in I} A_i$ (respectively $\bigvee_{i\in I} A_i$) if and only if he has one in every (respectively, some) A_i.

As in propositional logic, we interpret a sequent $\vdash C_1, \ldots, C_r$ as the game $C_1 \bar{\otimes} \cdots \bar{\otimes} C_r$ (literally, as $C_{1\mathfrak{M}}[s] \bar{\otimes} \cdots \bar{\otimes} C_{r\mathfrak{M}}[s]$, but we omit $\mathfrak{M}$ and s when no confusion results). A sequent $\vdash \Gamma$ is *true* in a model $\mathfrak{M}$ if, for every assignment s of values to variables, P has a winning strategy in the game $\Gamma_{\mathfrak{M}}[s]$. *Validity* means truth in all structures.

The Soundness Theorem for proportional game semantics carries over to the case of quantificational logic. The new steps in the proof are as follows. For the $\bigwedge$ rule, suppose $\vdash \Gamma, A$ is true in $\mathfrak{M}$; we must show that $\vdash \Gamma, \bigwedge xA$ is also true in $\mathfrak{M}$. So let any assignment s be given; we exhibit a winning strategy for P in

$$\Gamma_{\mathfrak{M}}[s] \bar{\otimes} \bigwedge_{m\in M} A_{\mathfrak{M}}[s \text{ but } x \mapsto m].$$

By definition of $\bar{\otimes}$, whenever P is to move in this game, O has already made his first move,the choice of m, in the component $\bigwedge_{m\in M} A_{\mathfrak{M}}[s \text{ but } x \mapsto m]$. So P can simply use his winning strategy (given by hypothesis) in $(\Gamma \bar{\otimes} A)_{\mathfrak{M}}[s$ but

$x \mapsto m]$. Of course we are using here that $\Gamma[s$ but $x \mapsto m] = \Gamma[s]$ because x is not free in Γ in the $\bigwedge$ rule.

For the $\bigvee$ rule, suppose $\vdash \Gamma, A[t/x]$ is true in $\mathfrak{M}$, and let s be given; we need a winning strategy for P in

$$\Gamma_{\mathfrak{M}}[s] \mathbin{\bar{\otimes}} \bigvee_{m \in M} A_{\mathfrak{M}}[s \text{ but } x \mapsto m].$$

Let $m_0 = t_{\mathfrak{M}}[s]$ and notice that (just as in classical logic)

$$A[t/x]_{\mathfrak{M}}[s] = A_{\mathfrak{M}}[s \text{ but } x \mapsto m_0].$$

Thus, by hypothesis, P has a winning strategy σ in $\Gamma_{\mathfrak{M}}[s] \mathbin{\bar{\otimes}} A_{\mathfrak{M}}$ $[s$ but $x \mapsto m_0]$, and he can therefore win

$$\Gamma_{\mathfrak{M}}[s] \mathbin{\bar{\otimes}} \bigvee_{m \in M} A_{\mathfrak{M}}[s \text{ but } x \mapsto m]$$

by choosing m_0 in the $\bigvee$ component and thereafter using σ. This completes the verification that quantified affine logic is game-semantically sound.

The Completeness Theorem for the additive fragment also carries over from propositional logic to quantificational logic. Given an unprovable sequent $\vdash \Gamma$, we construct a model $\mathfrak{M}$ and an assignment s such that P has no winning strategy in $\Gamma_{\mathfrak{M}}[s]$. The universe M of $\mathfrak{M}$ consists of all the terms in the language, and function symbols are interpreted in the obvious way. The assignment s sends each variable to itself (viewed as a member of M). The games associated to $\boldsymbol{Ra}$ are strict games with $\{0, 1\}$ as the set of moves and with P moving first; they are constructed by a transfinite recursion over all strategies σ. The construction and the proof that it succeeds are exactly as in Section 4, with the following additional steps in the verification that, at phase 1 moves, P must play and O can play so that $\vdash \Gamma'$ remains unprovable. There are two new cases to consider. First, a phase 1 move of P may select a particular $m_0 \in M$ in a component $\bigvee_{m \in M} A[s$but $x \mapsto m]$. But this has the effect of replacing $\bigvee x A$ in Γ' with $A[m_0/x]$ (since m_0 is, like every element of M, a term and $A[s$ but $x \mapsto m_0] = A[m_0/x][s])$, a replacement that preserves unprovability, thanks to the $\bigvee$ rule. Second, O may need to choose an $m_0 \in M$ in a component $\bigwedge_{m \in M} A[s$ but $x \mapsto m]$. In this situation, O should choose as m_0 a variable not occurring in Γ'. The effect is to replace $\bigwedge x A$ in Γ' with $A[m_0/x]$, which preserves unprovability, thanks to the $\bigwedge$ rule and the easily verified fact that renaming bound variables (to get from $\bigwedge m_0 A[m_0/x]$ to $\bigwedge x A$) is a derived rule of Girard's system. With these additions to the proof in Section 4, we obtain the Additive Completeness Theorem for quantificational logic.

8. Dialectica interpretation

Gödel's Dialectica interpretation [8] transforms arithmetical formulas into formulas with rather simple quantifier structure — existential quantifiers followed by universal ones — but with quantifiers ranging over objects of higher type. (Unlike Skolemization, which achieves the same quantifier form and which raises types by only one, the Dialectica interpretation behaves well with respect to proof-theoretic structure and allows the higher-type variables to range only over relatively simple functionals.) In developing a category-theoretic approach to the Dialectica interpretation, de Paiva [3] found a connection with linear logic. This connection suggests looking at the Dialectica interpretation, in de Paiva's category-theoretic version, from the point of view of game semantics, and this is the purpose of the present section.

Under the interpretations given to the quantifiers in Section 6, the quantifier prefix $\bigvee\bigwedge$ attained by the Dialectica interpretation represents a move of P followed by a move of O. Subsequent moves, in the game corresponding to the rest of the formula, behave like (interpretations of) additional quantifiers (over the set of moves). To genuinely capture, in game semantics, the idea that there are no quantifiers beyond the initial $\bigvee\bigwedge$, we should interpret the rest of the formula, the quantifier-free part, by a game in which the players do not move.

We did not allow such degenerate games in previous sections, insisting instead that plays of games must be infinite sequences. But there is no difficulty in extending the definitions so that some (finite) positions in a game are allowed to be terminal; the game would then include specifications of which positions are terminal and which player has won in each of these situations. In particular, the truth-table descriptions of $\otimes$ and $\check{\otimes}$ still work, and they indicate that to lose a game at a finite stage is just like being to move and having no (legal) move. (For more details, see [2], where terminal positions were systematically permitted.) This modification of the notion of game allows simpler definitions of $\top$ and $\bot$; they are the games with no moves that are won by P and O, respectively, and they are the only games with no moves. So these are the games that should interpret the quantifier-free parts of our $\bigvee\bigwedge$ formulas.

There is a straightforward way to convert any game G into a game $S(G)$ in $\bigvee\bigwedge$ form, i.e., consisting of only two moves, the first by P and the second by O. The opening move of P in $S(G)$ is a strategy for P in G; O's reply is a play of G in which P uses this strategy. (An alternative view of $S(G)$ is that P begins by divulging his strategy σ for G, and then the players play G with P required to follow σ. The previous description recognizes that P has no real choices after divulging σ and therefore the whole play, after the choice of σ, amounts to a move by O.) In terms of quantifier structure, $S(G)$ amounts to Skolemization of G.

We shall compare de Paiva's operations on (what amount to) $\bigvee\bigwedge$ formulas, in her version of the Dialectica interpretation, with our standard operations on games followed by an application of S to restore $\bigvee\bigwedge$ form.

De Paiva constructs, for an arbitrary category $\boldsymbol{C}$ with finite limits, a new category $\boldsymbol{DC}$ whose objects are relations $\alpha : A \rightarrowtail U \times X$. (We shall describe the morphisms later.) For comparison to game semantics, we take $\boldsymbol{C}$ to be the category of sets; otherwise we would have to deal with games having an object in C (rather than a set) of moves. A relation $A \subseteq U \times X$, an object of DC, corresponds to the formula $\bigvee u \bigwedge x\, A(u, x)$, where u and x range over U and X, respectively. It thus corresponds to the two-move game where P chooses a member u of U, then O chooses a member x of X, and finally P or O is declared the winner according to whether or not $A(u, x)$ holds. We now consider the result of applying the (game interpretation of the) connectives of linear logic to such games and then restoring the two-move form by means of the 'divulge P's strategy' operation S described above.

For this purpose, one of the simplest connectives is $\otimes$, because S is not needed. The game

$$\bigvee_u \bigwedge_x A(u, x) \otimes \bigvee_v \bigwedge_y B(v, y)$$

is, when viewed as a strict game, already in $\bigvee\bigwedge$ form. Indeed, in this game, since it is initially P's move (label False) in both components, P must begin by choosing both u and v. (Technically, he may choose them in either order, but this will not matter for the rest of the play.) Then (with both labels now true) it is O's turn, and he must choose x or y. If he chooses x so that $\sim A(u, x)$ or chooses y so that $\sim B(v, y)$, then he has won (as it is P's move and P has no move); otherwise, it is still O's move (as both labels are still True), and he must move in the other subgame, winning if and only if his choice makes $\sim A(u, x)$ or $\sim B(v, y)$ hold. Thus, all of P's moves are made before any of O's, which means that this $\otimes$ game, viewed as a strict game, is in $\bigvee\bigwedge$ form. If we simplify the game by (1) insisting that O choose both an x and a y, even if he could have stopped after the first choice (which will not affect play if X and Y are nonempty), and (2) ignoring the relative order in which P chooses u and v as well as the relative order in which O chooses x and y, then the $\otimes$ game becomes

$$\bigvee_{(u,v)} \bigwedge_{(x,y)} (A(u, x) \text{ and } B(v, y)),$$

which agrees with the tensor product in DC.

For contrast with $\otimes$, let us next consider $\wedge$. In the game

$$\bigvee_u \bigwedge_x A(u, x) \wedge \bigvee_v \bigwedge_y (B(v, y),$$

O moves first, choosing one of the components, and then P and O make the two moves constituting a play of that component. This is a three-move game, so we apply S to convert it to $\bigvee\bigwedge$ form. In the resulting game, P's opening move is a strategy σ, which in this case amounts to a choice of a reply (u or v) for each of O's two possible opening moves in the $\wedge$ game. So σ amounts to a pair (u, v). O's reply to this is a choice of subgame and a choice of x or y in the chosen subgame; thus, O chooses an element of the disjoint union $X + Y$ (which we can view as $(\{0\} \times X) \cup (\{1\} \times Y)$, so O's move codes in its first component his choice of subgame). Thus, we obtain the $\bigvee\bigwedge$ game

$$\bigvee_{(u,v)\in U\times V}\bigwedge_{z\in X+Y}[(z\in X \text{ and } A(u,z)) \text{ or } (z\in Y \text{ and } B(v,z))],$$

which agrees with the product in *DC*.

Instead of considering $\check{\otimes}$ next, we consider the closely related connective of linear implication defined by

$$C \multimap D = (\sim C)\check{\otimes} D,$$

since this corresponds to a construction treated in [3]. In the game

$$\bigvee_y \bigwedge_x A(u,z) \multimap \bigvee_v \bigwedge_y B(v,y),$$

i.e.,

$$\Big(\bigwedge_u \bigvee_x \sim A(u,x)\Big)\check{\otimes}\Big(\bigvee_v \bigwedge_y B(v,y)\Big),$$

play proceeds as follows. Initially (labels True on the left of $\check{\otimes}$ and False on the right) it is O's move, and he must choose u. Then it is P's move, and he can move in either component.

Case 1. P chooses x. If $\sim A(u,x)$, P wins. On the other hand, if $A(u,x)$, then P must choose v and O must reply by choosing y. Then P wins if and only if $B(v,y)$. As long as the sets V and Y are nonempty, play is essentially unchanged if we require P and O to choose v and y even if P has already won by virtue of $\sim A(u,x)$. With this modification, Case 1 is: P chooses x and v, O chooses y and P wins if and only if $A(u,x)$ implies $B(v,y)$.

Case 2. P chooses v. O must reply by choosing y, and if $B(v,y)$, then P has won. On the other hand, if $\sim B(v,y)$, then P must choose x, and P wins if and only if $\sim A(u,x)$. Again, we modify this by making P choose x even if he has already won because $B(v,y)$. So now Case 2 is: P chooses v, O chooses y, P chooses x and P wins if and only if $A(u,x)$ implies $B(v,y)$.

The only difference between the two cases is in the order in which x and y are chosen. From the point of view of P, Case 2 is preferable, because P can see y before having to choose x. More precisely, any strategy for P that uses Case 1 yields an equivalent strategy using Case 2. If we make a (third) inessential modification by insisting that P use his preferred approach, Case 2, then a strategy for P amounts to two functions, one giving v as a function of u, and one giving x as a function of u and y. So applying S to the (modified) game yields

$$\bigvee_{(v,\xi)V^U\times X^{(U\times Y)}}\bigwedge_{(u,y)\in U\times Y}[A(u,\xi(u,y)) \text{ implies } B(v(u),y)],$$

which is the internal Hom construction in [3].

A winning strategy (v,ξ) for P in this modified game is, by definition, a morphism from $A\subseteq U\times X$ to $B\subseteq V\times Y$ in *DC*.

In our discussions of $\otimes$ and $\multimap$ above, after applying the connective to two $\bigvee\bigwedge$ games and before applying S, we applied some inessential simplifications to the games in order to match the constructions in [3]. I have not checked what happens if one abstains from such simplifications, so that a morphism from $A \subseteq U \times X$ to $B \subseteq V \times Y$ would be a strategy for P in the unmodified $\multimap$ game.

Continuing with the list of connectives, we consider the $\vee$ game

$$\Big(\bigvee_u \bigwedge_x A(u,x)\Big) \vee \Big(\bigvee_v \bigwedge_y B(v,y)\Big),$$

which needs no S as it is already in $\bigvee\bigwedge$ form: P chooses a subgame and u or v, i.e., he chooses from $U+V$. Then O chooses from $X+Y$ (losing if he chooses from the wrong component), P wins if $A(u,x)$ or $B(v,y)$ (or O chose from the wrong component). This corresponds to the relation

$$\begin{aligned} A+B+(U\times Y)+(V\times X) &\subseteq (U\times X)+(V\times Y)+(U\times Y)+(V\times X) \\ &\cong (U+V)\times(X+Y). \end{aligned}$$

In contrast to what happened for the other connectives, this does not correspond to any of the operations in [3]. In particular, de Paiva's weak coproduct $\oplus$ (corresponding to $\vee$) is quite different.

Although de Paiva did not present an interpretation of $\tilde{\otimes}$ — she worked with intuitionistic linear logic where $\tilde{\otimes}$ occurs only in the context of linear implication — we record that, in

$$\bigvee_u \bigwedge_x A(u,x) \,\tilde{\otimes}\, \bigvee_v \bigwedge_y B(v,y),$$

modified as before by requiring the players to continue choosing even if P has already won, a strategy for P is an element of $(U\times V^X)+(V\times U^Y)$, so S produces the game

$$\begin{aligned} \bigvee_{z\in(U\times V^X)+(V\times U^Y)} \ \bigwedge_{(x,y)\in X\times Y} &[(z=(u,v)\in U\times V^X \text{ and } A(u,x) \text{ or } B(v(x),y)) \\ &\text{or } (z=(v,\mu)\in V\times U^Y \text{ and } A(\mu(y),x) \text{ or } B(v,y))]. \end{aligned}$$

Finally, we turn to the exponential operators R and $\tilde{R}$ (or ! and ? in standard notation), of which only the former occurs in [3]. Recalling the definition of R for games where P moves first, we find that in a play of

$$R\Big(\bigvee_u \bigwedge_x A(u,x)\Big),$$

P begins by choosing a value of u, which serves as his opening move in all of the (infinitely many) copies of $\bigvee u \bigwedge x\, A(u,x)$. Then O chooses (possibly different) x's in all these copies, and O wins if and only if he wins in at least one copy.

Thus, this game is already in $\bigvee\bigwedge$ form, and S does nothing to it. Formally, it is

$$\bigvee_u \bigwedge_{(x_0,x_1,\ldots)\in{}^\omega X} [A(u, x_n) \text{ for all } n].$$

This resembles de Paiva's interpretation of ! in that there is a single u but many x's and in that the original A is to hold for the one u and every x; it differs, however, in that we have infinitely many x's where de Paiva has finitely many.

We could modify the game semantics of R by adding the rule that, if O moves in infinitely many of the subgames, then he loses. This would bring our construction into agreement with de Paiva's, and it would not damage the soundness proof for game semantics.

The game interpretation of

$$\tilde{R}\Big(\bigvee_u \bigwedge_x A(u, x)\Big)$$

is a game where P and O alternately choose u_i's and x_i's, each x_i being chosen by O just after P chooses u_i and just before P chooses u_{i+1}. P wins if and only if $A(u_i, x_i)$ holds for some i. Applying S, we obtain the game

$$\bigvee_{(\mu_0,\mu_1,\ldots)\in\prod_i U^{X^i}} \bigwedge_{(x_0,x_1,\ldots)\in{}^\omega X} [\text{for at least one } i,\ A(\mu_i(x_0, \ldots, x_{i-1}), x_i)].$$

References

[1] P. Aczel, An introduction to inductive definitions, in: J. Barwise, ed., Handbook of Mathmetical Logic Stud. Logic Found. Math. 90 (North-Holland, Amsterdam, 1977) 739–782.

[2] A. Blass, Degrees of indeterminacy of games, Fund. Math. 77 (1972) 151–166.

[3] V.C.V. de Paiva, The Dialectica categories, in: J.W. Gray and A. Scedrov, eds, Categories in Computer Science and Logic, Contemp. Math. 92 (Amer. Math. Soc., Providence, RI, 1989) 47–62.

[4] D. Gale and F.M. Stewart, Infinite games with perfect information, in: Contributions to the Theory of Games, Ann. of Math. Stud. 28 (Princeton Univ. Press, Princeton, NJ. 1953) 245–266.

[5] A. Garsia and S. Milne, Method for constructing bijections for classical partition identities, Proc. Nat. Acad. Sci. U.S.A. 78 (1981) 2026–2028.

[6] J.Y. Girard, Linear Logic, Theoret. Comp. Sci. 50(1) (1987) 1–102.

[7] J.Y. Girard, Towards a geometry of interaction, in: J.W. Gray and A. Scedrov, eds., Categories in Computer Science and Logic, Contemp. Math. 92 (Amer. Math. Soc., Providence, RI, 1989) 69–108.

[8] K. Gödel, Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes, Dialectica 12 (1958) 280–287.

[9] K. Lorenz, Dialogspiele als semantische Grundlage von Logikkalkülen, Arch. Math. Logik Grundlag. 11 (1968) 32–55; 73–100.

[10] P. Lorenzen, Ein dialogisches Konstruktivitätskriterium, in: Infinitistic Methods (PWN, Warsaw, 1959) 193–200.

[11] P. Whitman, Free lattices, Ann. of Math. 42 (1941) 325–330.