



HAL
open science

Dialogical Argumentation and Textual Entailment

Davide Catta, Richard Moot, Christian Retoré

► **To cite this version:**

Davide Catta, Richard Moot, Christian Retoré. Dialogical Argumentation and Textual Entailment. Natural Language Processing in Artificial Intelligence-NLPinAI 2020, 939, Springer, pp.191-226, 2021, Studies in Computational Intelligence, 978-3030637866. 10.1007/978-3-030-63787-3_7. hal-02915526v2

HAL Id: hal-02915526

<https://hal.science/hal-02915526v2>

Submitted on 9 Sep 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Dialogical Argumentation and Textual Entailment

Davide Catta and Richard Moot and Christian Retoré

Abstract In this chapter, we introduce a new dialogical system for first order classical logic which is close to natural language argumentation, and we prove its completeness with respect to usual classical validity. We combine our dialogical system with the Grail syntactic and semantic parser developed by the second author in order to address automated textual entailment, that is, we use it for deciding whether or not a sentence is a consequence of a short text. This work — which connects natural language semantics and argumentation with dialogical logic — can be viewed as a step towards an inferentialist view of natural language semantics.

1 Presentation: Argumentation, Inference, Semantics

This work takes its inspiration from the observation that logical or natural language inferences should be related to inferentialism, i.e., a view of the semantics of a formula or of a sentence as the inferential possibilities of the statement in reasoning or argumentation. Let us first present inferentialism, which although not new is not that well-known in logic or in natural language semantics.

Davide Catta

LIRMM, Université de Montpellier, CNRS, 860 rue Saint Priest, 34095 Montpellier, France e-mail: davide.catta@lirmm.fr

Richard Moot

LIRMM, Université de Montpellier, CNRS, 860 rue Saint Priest, 34095 Montpellier, France e-mail: richard.moot@lirmm.fr

Christian Retoré

LIRMM, Université de Montpellier, CNRS, 860 rue Saint Priest, 34095 Montpellier, France e-mail: christian.retore@lirmm.fr

Inferentialism

A problem with the standard view of both natural language semantics and of logical interpretations of formulas is that the models or possible worlds in which a sentence is true cannot be computed or even enumerated [33]. As far as pure logic is concerned there is an alternative view of meaning called *inferentialism* [14, 15, 36, 12, 5]. Although initially inferentialism was introduced within a constructivist view of logic [14], there is no necessary conceptual connection between accepting an inferentialist position and rejecting classical logic as explained in [12].

As its name suggests inferentialism replaces *truth* as the primary semantic notion by the *inferential activity* of an agent. According to this paradigm, the meaning of a sentence is viewed as the *knowledge* needed to understand the sentence.

This view is clearly stated by Cozzo [12]

A theory of meaning should be a theory of understanding. The meaning of an expression (word or sentence) or of an utterance is what a speaker-hearer must know (at least implicitly) about that expression, or that utterance, in order to understand it.

This requirement has some deep consequences: as speakers are only able to store a finite amount of data, the knowledge needed to understand the meaning of the language itself should also be finite or at least recursively enumerable from a finite set of data and rules. Consequently, an inferentialist cannot agree with the Montagovian view of the meaning of a proposition as the possible worlds in which the proposition is true [29]. In particular because there is no finite way to enumerate the infinity of possible worlds nor to finitely enumerate the infinity of individuals and of relations in a single of those possible worlds.

Let us present an example of the knowledge needed to understand a word. If it is a referring word like “*eat*”, one should know what it refers to *the action that someone eats something*, possibly some postulates related to this word like *the eater is animated*,¹ and how to compose it with other words *eat is a transitive verb, a binary predicate*; if it is a non referring word like “*which*” one should know that it combines with a sentence without subject (a unary predicate), and a noun (a unary predicate), and makes a conjunction of those two predicates.

Observe that this knowledge is not required to be explicit for human communication. Most speakers would find difficult to explicitly formulate these rules, especially the grammatical ones. However, this does not mean that they do not possess this knowledge.

An important requirement for a theory of meaning is that the speaker’s knowledge can be *observed*, i.e., his knowledge can be observed in the interactions between the speaker(s), the hearer(s) and the environment. This requirement is supported by the famous argument against the *private language* of Wittgenstein [40]. This argument can be presented as follows. Imagine that two speakers have the same use of a sentence *S* in all possible circumstances. Assume that one of the two speakers includes as part of the meaning of *S* some ingredient that cannot be observed. This ingredient has to

¹ Our system is able to deal with metaphoric use, like the *The cash machine ate my credit card*. see e.g., [37].

be ignored when defining the knowledge needed to master the meaning of *S*. Indeed, according to the inferentialist view, a misunderstanding that can neither be isolated nor observed should be precluded.

Another requirement for a theory of meaning is the distinction between sense and force, on which we shall be brief. Since Frege [19], philosophy of language introduced the distinction between the sense of the sentence and its force. The sense of a sentence is the propositional content conveyed by the sentence, while its force is its *mood* — this use of the word “mood” is more general than its linguistic use for verbs. Observe that the same propositional content can be asserted, refuted, hoped etc., as in the three following sentences *Is the window open? Open the window! The window is open..* Here we focus on assertions and questions.

Observe that this draws a connection between inferentialism and study of dialogue in linguistics and philosophy. Indeed, the most common interaction between the speaker(s), the hearer(s) and the environment is a dialogue in natural language. Only argumentative dialogues are relevant from an inferentialist perspective, but they still are dialogues and as such the wide literature on dialogue may be helpful to deepen the connection developed in this paper, in particular [3, 20, 10, 24]. The force or mood of the intervention of the speaker and the hearer in argumentative dialogue are particular cases of functional roles in dialogues (see e.g., [38]), and this connection with different moods of dialogue interaction paves the way to a similar treatment of other dialogues which are not limited to argumentative interaction. As an intermediate step between argumentative dialogues which we now deal with, and general dialogues which we plan to study, could be argumentative dialogues with non standard inferences resulting from enthymemes or from psychological disorders respectively studied in [6] and [4]. But as a first exploration of inferential semantics, let us focus on the applications of inferentialism to argumentative dialogues and in particular to textual entailment.

Inferentialism and Textual Entailment

We will illustrate the application of the inferentialist view to natural language semantics with an very natural task: the recognition of natural language inference, a task also known as textual entailment. In the current context, we use textual entailment in a more limited sense than it is generally used in natural language processing tasks. Textual entailment in natural language processing generally aims to obtain human-like performance on relating a text and a possible conclusion [13]. Natural language processing systems are evaluated on their ability to approach the performance of humans when deciding between entailment, contradiction and unknown (i.e., neither the entailment relation nor the contradiction relation holds between the text and the given candidate conclusion). We consider textual entailment from a purely logical point of view, taking entailment and contradiction in their strictly logical meanings. In our opinion, a minimal requirement for a textual entailment system should be that

it can handle the well-known syllogisms of Aristotle, as well as a number of other patterns [34] with perfect accuracy².

The computational correspondence between natural language sentences and logical formulas is obtained both theoretically and practically using type-logical grammars. In a sense, type-logical grammars are *designed* to produce logical meanings for grammatical sentence. They compute the possible meanings of a sentence viewed as logical formulae. In particular, the Grail platform, a wide-scale categorial parser which maps French sentences to logical formulas [31, 30, 32], will be presented. We then use Grail and dialogical logic to solve some examples of textual entailment from the FraCaS database [11].

Overview

This chapter is structured as follows. Sect. 2 introduces dialogical logic for classical first order logic and dialogical validity. Sect. 3 gives a proof of the fact that the class of formulas that are dialogically valid is (almost) equal to the class of formulas that are valid in the standard meaning of the term, i.e., true in all interpretations. Sect. 4 is an introduction to categorial grammars and Grail. We conclude with the applications of semantics to the problem of textual inference.

We then show applications of this semantics to the problem of textual entailment using examples taken from the FraCaS database.

Hence this paper deepens and extends our first step in this direction “inferential semantics and argumentative dialogues” [9], mainly because of a strengthened link with natural language semantics and textual entailment:

- Technically, we provide a proof that classical validity is equivalent, in a sense that will be specified below, to dialogical validity.
- We do not require the formulas to be in a normal negative form as done in [9]. In this way the strategies correspond to derivations in a two-sided sequent calculus, i.e., with multiple hypothesis and multiple conclusions.
- We use Grail to connect natural language inference (or textual entailment) to argumentative dialogues: indeed Grail turns natural language sentences into logical formulas (delivered as DRS) — while our previous work only assumed that sentences could be turned into logical formulas.
- This lead us closer to inferentialist semantics: a sentence S can be interpreted as all argumentative dialogues (possibly expressed in natural language) whose conclusion is S — under assumptions corresponding to word meaning and to the speaker beliefs.

² The patterns in the well-known corpus for testing Textual Entailment recognition called FraCaS [11] greatly vary in difficulty. We expect only some of them (monotonicity, syllogisms) to be handled easily, whereas we expect others (plurals, temporal inference for aspectual classes) to be much more difficult for systems based on automated theorem provers, or, indeed, for any automated system.

2 Dialogical Logic

The formal developments of the inferentialist view of meaning — as far as logic is concerned — are usually based on natural deduction [18]. We choose a different approach, and give a formal implementation of the ideas of inferentialist meaning theory using the tools of dialogical logic. In our view, the connection with semantics based on the notion of argument is clearer within the latter paradigm: an argument in favor of a statement is often developed when a critical audience, real or imaginary, doubts the truth, or the plausibility of the proposition. In this case, in order to successfully assert the statement, a speaker or proponent of it must be capable of providing all the justifications that the audience is entitled to demand. Taking this idea seriously, an approximation of the meaning of a sentence in a given situation can be obtained by studying the *argumentative dialogues* that arise once the sentence is asserted in front of such a critical audience. This type of situation is captured — with a reasonable degree of approximation — by dialogical logic. In the dialogical logic framework, knowing the meaning of a sentence means being able to provide a justification of the sentence to a critical audience. Note that with this type of methodology the requirement of manifestability required to attribute knowledge of the meaning of a sentence to a locutor is automatically met. The locutor who asserts a certain formula is obliged to make his knowledge of the meaning manifest so that he can answer the questions and objections of his interlocutor. In addition, any concessions made by his interlocutor during the argumentative dialogue will form the linguistic context in which to evaluate the initial assertion. Our approach shares some similarities with the one in [17, 27], which propose a formal implementation of the inferentialist view of meaning in the frame of Ludics [22]: the main objects of Ludics, called *designs*, can be viewed as argumentative strategies.

We now progressively present dialogical logic. Although the study of dialectics — the art of correct debate — and logic — the science of valid reasoning — have been intrinsically linked since their beginnings [8, 35, 7], modern mathematical logic had to wait until the 50s of the last century to ensure that the logical concept of validity was expressed through the use of dialogical concepts and techniques. Inspired by the *Philosophical Investigations* of Wittgenstein [40], the German mathematician and philosopher Lorenzen [28] proposed to analyze the concept of validity of a formula F through the concept of winning strategy in a particular type of two-player game. This type of game is nothing more than an argumentative dialogue between a player, called *proponent*, who affirms the validity of a certain formula F and another player, called *opponent*, who contests its validity. The proponent starts the argumentative dialogue by affirming a certain formula. The opponent takes turns and attacks the claim made by the proponent according to its logical form. The proponent can, depending on his previous assertion and on the form of the attack made by the opponent, either defend his previous claim or counter attack. The debate evolves following this pattern. The proponent wins the debate if he has the last word, i.e., the defence against one of the attacks made by the opponent is a proposition that the opponent can not attack without violating the debate rules.

Dialogical logic was initially conceived by Lorenzen as a foundation for intuitionistic logic (IL). Lorenzen’s idea was, roughly speaking, the following. It is possible to define a “natural” class of dialogue games in which, given a formula F of IL , the proponent can always win a game on F , *no matter how the opponent chooses to attack his assertion in the debate*, if F is IL-valid. This intuition was formalized as the completeness of the dialogical proof system with respect to provability or validity in any model:

Completeness of a dialogical proof system: Given a logical language L and a notion of validity for L (either proof theoretical or model theoretical) and a notion of dialogical game, a formula F is valid in L if and only if, there is a winning strategy for the proponent of F in the class of games under consideration.

A winning strategy can be intuitively understood as an algorithm that takes as input the moves of the game made so far, and outputs moves for the proponent which guarantee he will always win. Unfortunately, almost 40 years of work were needed to get a first correct proof of the completeness theorem [16]. Subsequently, different systems of dialogical logic were developed. We present here a system of dialogical logic that is complete for classical first order logic.

2.1 Language, Formulas, Subformulas, Trees

Throughout the paper we assume that a first order language \mathcal{L} has been defined, by a set of constants and of function symbols (each of them with an arity) and a set of predicate symbols (each of them with a specified arity), including a particular 0-ary predicate, i.e., a proposition letter written \perp – beware that, in some of the sequent calculi to be introduced later on, this proposition letter has no particularity while in some others it has a specific rule, *ex falso quod libet sequitur*. The set of terms \mathcal{T} is defined as the smallest set containing constants and variables (their set is \mathcal{V}), and closed under function symbols: if t_1, \dots, t_n are n terms and if f in an n -ary function symbol, the expression $f(t_1, \dots, t_n)$ is a term as well.

An atomic formula is an expression $P(t_1, \dots, t_p)$ where P is an n -ary predicate symbol and t_1, \dots, t_n are n terms. In particular a proposition letter is an atomic proposition.

Formulas and sub-formulas are defined as usual, except that we have no negation connective — in some of the proof systems considered in the paper $G \Rightarrow \perp$ enjoys all of the properties of $\neg G$ while in some others it does not. The set of terms of the language will be denoted by \mathcal{T} , the set of variables of the language will be denoted by \mathcal{V} . The set of set of formulas on the language will be denoted by \mathcal{F} .

We here recall the definitions of formulas and two notions of subformula, the usual one and the Gentzen one. The set of *formulas* as well as the multisets of *positive (resp. negative) occurrences of subformulas* $sub^+(F)$ (resp. $sub^-(F)$) of a formula F are defined as follows, where \uplus denotes multiset union:

- if F is an atomic formula $P(t_1, \dots, t_n)$ then F is a formula and

$$\begin{aligned} \text{sub}^+(F) &= \{F\} \\ \text{sub}^-(F) &= \emptyset \end{aligned}$$
- if F_1 and F_2 are formulas then $F = F_1 * F_2$ with $*$ $\in \{\wedge, \vee\}$ is a formula and

$$\begin{aligned} \text{sub}^+(F) &= \text{sub}^+(F_1) \uplus \text{sub}^+(F_2) \uplus \{F\}. \\ \text{sub}^-(F) &= \text{sub}^-(F_1) \uplus \text{sub}^-(F_2) \end{aligned}$$
- If if F_1 and F_2 are formulas then $F = F_1 \Rightarrow F_2$ is a formula and

$$\begin{aligned} \text{sub}^+(F) &= \text{sub}^-(F_1) \uplus \text{sub}^+(F_2) \uplus \{F\} \\ \text{sub}^-(F) &= \text{sub}^+(F_1) \uplus \text{sub}^-(F_2) \end{aligned}$$
- if F_1 is a formula and x is a variable then $F = Qx. F_1$ with $Q \in \{\exists, \forall\}$ is a formula and

$$\begin{aligned} \text{sub}^+(F) &= \text{sub}^+(F_1) \uplus \{F\} \\ \text{sub}^-(F) &= \text{sub}^-(F_1) \end{aligned}$$
- nothing else is a formula.

The subformula occurrences of a formula F are simply $\text{sub}^+(F) \uplus \text{sub}^-(F)$. Observe that, because we use multisets which distinguish the different occurrences of the same subformula in a formula, a given occurrence of a subformula in F is either positive or negative, but not both, although the same underlying formula may appear both in $\text{sub}^+(F)$ and in $\text{sub}^-(F)$.

Let $F[t/x]$ stand for the formula obtained by replacing the free occurrences of x in F by the term t in \mathcal{T} . The *Gentzen variants* $gv(G)$ of a formula G are obtained by recursively replacing the quantified variables of G by terms in \mathcal{T} , starting from outermost quantifiers:

- $gv(A(t_1, \dots, t_n)) = \{A(t_1, \dots, t_n)\}$ there are no variants but the formula itself when the formula is atomic
- $gv(F_1 * F_2) = \{F_1 * F_2\} \cup gv(F_1) \cup gv(F_2)$ when $*$ is a binary connective among $\wedge, \vee, \Rightarrow$
- $gv(Qx F_1) = \{Qx F_1\} \cup gv(F_1[t/x])$ when Q is a quantifier among \exists, \forall

The positive (resp. negative) *Gentzen subformulas* of a formula F are the Gentzen variants $gv(H)$ with H a positive (resp. negative) occurrence of a subformula in F , i.e., for $H \in \text{sub}^+(F)$ (resp. $H \in \text{sub}^-(F)$). Thus any Gentzen subformula is anchored to an occurrence of a subformula.

The above logical definitions about formulas are standard (see, e.g., [21]), but given that they may differ slightly from one textbook to another, we prefer to present them in full detail.

The paper also deal with trees, because proofs and strategies are trees. We recall that a tree can either be defined as a set of prefix closed sequences (the empty sequence corresponds to the root), or inductively: given a family of trees t_1, \dots, t_n , one may define a new tree $f(t_1, \dots, t_n)$, whose daughters are the t_i s, by adding a new root f . In a tree there is a unique path from a node to another, in particular from any node x to the root: such a path is called a branch, its length is the height of the node x ; the branch is said to be maximal when it is not the prefix of a longer branch, and in this case x is called a leaf. The paper does not need a more formal reminder about trees to be understood.

2.2 Argumentation Forms

Let us consider a set Aux of auxiliary symbols containing the symbols $\wedge_1, \wedge_2, \vee, \exists$ and the expressions $\forall[t/x]$ for all terms in \mathcal{T} and variables x in \mathcal{L} , and nothing else.

Following the terminology of Felscher [16], an argumentation form Arg is a function assigning to each non atomic formula F in \mathcal{F} a set of pairs consisting in one *question* (also called *attacks* in the literature) and one *answer* (also called *defence* in the literature) with questions being either formulas in \mathcal{F} or symbols in Aux and answers being formulas in \mathcal{F} .³

$$\begin{aligned} Arg(F_1 \Rightarrow F_2) &= \{(F_1, F_2)\} \\ Arg(F_1 \wedge F_2) &= \{(\wedge_1, F_1), (\wedge_2, F_2)\} \\ Arg(F_1 \vee F_2) &= \{(\vee, F_1), (\vee, F_2)\} \\ Arg(\forall x F) &= \{(\forall[t/x], F[t/x]) \mid t \in \mathcal{T}\} \\ Arg(\exists x F) &= \{(\exists, F[t/x]) \mid t \in \mathcal{T}\} \end{aligned}$$

In a pair $(\forall[t/x], F[t/x]) \in Arg(\forall x F)$ the term t in the question $\forall[t/x]$ is called the *chosen term*. In each couple $(\exists, F[t/x])$ the term t in the answer $F[t/x]$ is called *chosen term*. Given a formula F , a question q that belongs to a couple $(q, a) \in Arg(F)$ is called a *question on F* . An answer a is called an *answer to q* whenever the couple (q, a) is an element of $Arg(F)$. So, for example, if F is $F_1 \wedge F_2$, both \wedge_1 and \wedge_2 are question on F but only F_1 is an answer to \wedge_1 and only F_2 is an answer to \wedge_2 . If $F = F_1 \vee F_2$, the symbol \vee is a question on F , and both F_1, F_2 are answers to \vee . Consider the case where F is $F_1 \Rightarrow F_2$. In this case F_1 is a question on F and F_2 is an answer to F_1 . The use of the expression “question” to qualify F_1 may sounds odd. The bizarre impression that the expression “question” generates if associated with F_1 disappears if we paraphrase F_1 as a question in the following way: “could you convince me that F_2 holds by assuming the hypothesis that F_1 holds?”.

2.3 Prejustified Sequences

Moves are pairs (i, s) with $i \in \{?, !\}$ and s being either a formula or an auxiliary symbol. Moves are called attacks whenever $i = ?$ and defences whenever $i = !$.

Some moves are called *assertions*. There are two types of assertions: all defences are assertions, and attacks of the form $(?, F)$ where F is a formula are assertions.

A *prejustified* sequence is a sequence of moves $\mathbf{M} = M_0, M_1, \dots, M_j, \dots$, together with a partial function $f : \mathbf{M} \rightarrow \mathbf{M}$ such that for all all M_i in the sequence for which f is defined, $f(M_i)$ is an M_j such that $j < i$. We say that $f(M_i)$ is the enabler of M_i or

³ The words “question” and “answer” are called “attack” and “defence” by Felscher in [16]; we deviate from this terminology because we will rather use the terms “attack” and “defence” exclusively for the moves in a game, avoiding possible confusion.

that M_i is enabled by $f(M_i)$. Given a prejustified sequence $\mathbf{M} = M_0, M_1, \dots, M_j, \dots$, an attack move M_n in \mathbf{M} of the form $(?, s)$ is said to be *justified* whenever $f(M_n)$ is an assertion (i, F) and s is a question on F . The attack move $M_n = (?, s)$ is called an existential attack whenever $s = \exists$, and a universal attack whenever $s = \forall[t/x]$.

A defence M_n of the form $(!, F)$ is *justified* if $f(M_n) = (?, s)$ is a justified attack of M_j , M_j is an assertion (i, F') , s a question on F' , and the couple (s, F) belongs to $\text{Arg}(F')$. An assertion M_n of the form (i, F) is a *reprise* if and only if there exists another move M_j , with $j < i$ and j having opposite parity of i , of the form (i', F) . An assertion M_n is called an *existential repetition*, if its enabler is of the form $(?, \exists)$ and there is another move of the same parity M'_n with $n' < n$ having the same enabler. A pre-justified sequence in which each move is justified is called *justified sequence*

2.4 Games

Definition 1 A *game* for a formula F is a pre-justified sequence

$$M_0, M_1, \dots, M_j, \dots$$

such that

1. M_0 is $(!, F)$ and M_1, \dots, M_j, \dots is a justified sequence in which each odd-indexed move is enabled by its immediate predecessor and in which each even-indexed move is enabled by a preceding odd-index move,
2. if an even-indexed move asserts an atomic formula then it is a reprise,
3. for all even m, n if M_m and M_n are defence moves that assert the subformula F_1 of F and are enabled by the same move M_j , then $m = n$ unless M_j is $(?, \vee)$ and $f(M_j) = (F_1 \vee F_1)$

The item 2 is usually called the formal rule.

Odd-index moves are opponent moves (**O**-moves) while even index moves are called proponent moves (**P**-moves). A move M is *legal* for a game G if the pre-justified sequence G, M is a game.

A game G is *won* by the proponent if, and only if, it is finite and there is no **O**-move which is legal for G . It is won by the opponent otherwise.

Remark 1 All formulas that are asserted in a Game G for a formula F are, by construction of the game, Gentzen subformulas of F

We give three examples of games. Each game will be represented as a table with two columns. The first column, read from top to bottom represents the sequence of moves of the game. The second column shows the value of the function f at each point of the game.

The first is a game won by **P** for the formula $a(x) \vee \neg a(x)$ where $a(x)$ is an atomic formula.

$$\begin{array}{l|l}
M_0 = (!, a(x) \vee \neg a(x)) & \\
M_1 = (?, \vee) & M_0 \\
M_2 = (!, \neg a(x)) & M_1 \\
M_3 = (?, a(x)) & M_2 \\
M_4 = (!, a(x)) & M_1
\end{array}$$

The game starts by the assertion of $a(x) \vee \neg a(x)$. The move M_1 is a justified attack on M_0 . In fact $f(M_1) = M_0$ and \vee is a question on $a(x) \vee \neg a(x)$. The following defence move M_2 is justified. In fact $f(M_2) = M_1$ is a justified attack, and $\neg a(x)$ is an answer to the question \vee on the formula $a(x) \vee \neg a(x)$. The assertion M_3 is itself a justified attack. $f(M_3) = M_2$, M_2 is an assertion of $\neg a(x) = a(x) \Rightarrow \perp$ and $a(x)$ is a question on $a(x) \Rightarrow \perp$. The final move M_4 is a justified defence. In fact $f(M_4) = M_1$, M_1 is a justified attack on M_0 and $a(x)$ is an answer to $?\vee$ which, in turn, is a question on $a(x) \vee \neg a(x)$. Moreover M_4 is a *reprise*: there exists another assertion, the assertion M_3 , that asserts the same formula $a(x)$, with 3 odd and smaller than 4.

The second example is a game that is not won by **P** for the formula $a(x) \vee \neg a(y)$

$$\begin{array}{l|l}
M_0 = (!, a(x) \vee \neg a(y)) & \\
M_1 = (?, \vee) & M_0 \\
M_2 = (!, \neg a(y)) & M_1 \\
M_3 = (?, a(y)) & M_2
\end{array}$$

The game starts by the assertion of $a(x) \vee \neg a(y)$. The move M_1 is a justified attack on M_0 , since $f(M_1) = M_0$ and \vee is a question on $a(x) \vee \neg a(y)$. The following defence move M_2 is justified because $f(M_2) = M_1$ is a justified attack, and $\neg a(y)$ is an answer to the question \vee on the formula $a(x) \vee \neg a(y)$. The assertion M_3 in its turn is a justified attack. $f(M_3) = M_2$, M_2 is an assertion of $\neg a(y) = a(y) \Rightarrow \perp$ and $a(y)$ is a question on $a(y) \Rightarrow \perp$. The game ends here and its lost by **P**. Remark that **P** cannot extend the game: he cannot assert \perp as an answer to $a(y)$ in M_3 , because there is no move M_k by **O** that asserts \perp . For the same reason he cannot assert $a(x)$ as an answer to \vee in M_1 . Moreover he cannot assert $\neg a(y)$ a second time because of condition 3 on the definition of dialogical games.

The third and last example is a game won by **P** for the formula $a \vee b \Rightarrow a$. This formula is not a tautology. The fact that **P** can win a game on this formula means that the notion of validity cannot be captured using only the definition of game.

$$\begin{array}{l|l}
M_0 = (!, a \vee b \Rightarrow a) & \\
M_1 = (?, a \vee b) & M_0 \\
M_2 = (?, \vee) & M_1 \\
M_3 = (!, a) & M_2 \\
M_4 = (!, a) & M_1
\end{array}$$

The game starts by the assertion of $a \vee b \Rightarrow a$. The following move M_1 is a justified attack on M_0 , because $F(M_1) = M_0$ and $a \vee b$ is a question on $a \vee b \Rightarrow a$. The following attack move M_2 is justified. In fact $f(M_2) = M_1$ is an assertion of $a \vee b$ and \vee is an attack on this last formula. The defence move M_3 is justified, since a is an answer on \vee which is a question on $a \vee b$. Finally the move M_4 is a justified

defence. $f(M_4) = M_1$ is a justified attack, a is an answer to $a \vee b$ if this last is a question on $a \vee b \Rightarrow a$. Moreover M_4 is a reprise, because there exists an earlier assertion by **O**, namely M_3 , that asserts a .

Here are some properties that will be used in the final section of this chapter.

Proposition 1 *If a finite game $G = M_0, \dots, M_n$ is won by **P** in the sense defined above (**O** has no further legal move left) then M_n is the assertion of some atomic formula $a(t_1, \dots, t_m)$.*

Proposition 2 *For all games G , for all formulas F and for all subformula F' of F . If there is an **P**-move (**O**-move) in G that asserts F' then F' is a positive (negative) subformula of F .*

Proof Let F be any formula and G any game. We show the proposition by induction on the length of G . If the length is 1 then G consist of only one move that is a **P**-move asserting the formula F and so the proposition holds.

Suppose that the proposition holds for all games G having length n and let G' be a game having length $n + 1$. Let M_n be the last move of G' . Suppose that M_n is a **P**-move (the argument for **O**-moves runs in a very similar way) We have three cases.

1. If M_n is not an assertion, the proposition holds automatically by induction hypothesis.
2. If M_n is a defence asserting some formula F' , then, since G' is a game, the sequence $M_1 \dots M_n$ is justified. Thus M_n is enabled by some **O**-move M_k with ($k < n$). If $M_k := (?, F_1)$ (the other cases are easier) then it is an attack against M_{k-1} and M_{k_1} is a **P**-move that asserts $F_1 \Rightarrow F'$. By induction hypothesis $F_1 \Rightarrow F'$ is a positive subformula of F , and F_1 is a negative subformula of F . Thus F' is a positive subformula of F by definition.
3. If M_n is an assertion and an attack, let F' be the asserted formula. As before there must exists an enabler of M_n , call it M_k ($k < n$), M_k is necessarily an **O**-move that asserts the formula $F' \Rightarrow F''$. By induction hypothesis this last formula is a negative subformula of F , thus F' is a positive subformula of F by definition. \square

An easy consequence of the latter proposition is the following

Proposition 3 *Let G be a game for a formula F and let M_n be a **P**-move in G asserting an atomic formula $a(t_1, \dots, t_m)$. Then this latter formula appears both as a negative and positive subformula of F*

2.5 Strategies

Informally speaking, a strategy for a player is an algorithm for playing the game that tells the player what to do for every possible situation throughout the game. We informally describe how a strategy should operate and then formalize this notion.

Imagine being engaged in a game G , that the last move of G was played according to the strategy, and that it is now your opponent's turn to play. Your opponent could extend the game in different ways: for example if you are playing chess, you are white and you just made your first move by moving a pawn to a certain position of the chessboard, black can in turn move a pawn or move a horse. If you are playing according to the strategy, the strategy should tell you how to react against either type of move. If black moves a pawn to $C6$ and you just moved your pawn to $C3$ then move the horse to $H3$. If black moves a horse to $H6$ and you just moved your pawn to $C3$ then move your pawn in $B4$. Therefore, a strategy can be viewed as tree in which each node is a move in the game, the moves of my opponent have at most one daughter, and my moves have as many daughters as there are available moves for my opponent.

In game semantics and in the dialogical logic literature, it is rather standard to represent a strategy as a tree of games [16, 1, 25]. Nevertheless one should keep in mind that a strategy is a function telling one player which move to play in her turn to play whatever the history of the game is, and this is easily represented as a tree. We thus formalize the notion of strategy as follows. Given a game G we say that a variable x appears in the game if, and only if, the variable x appears in some asserted formula or is free in the choice of some universal attack. Let $(v_i)_{i \in I}$ be an enumeration of the variables in L . A *strategy* S is a prefix-closed set of games for the same formula (i.e., a tree of games for the same formula) such that:

1. if G belongs to the strategy and the last move of G is a **P** move that is neither an assertion of a universally quantified formula nor an existential attack, then G, M belongs to the strategy, for all moves M legal for G ,
2. if G, M and G, M' belong to S and M, M' are **P**-moves then $M = M'$,
3. if G, M and G, M' belongs to S and M, M' are **O**-moves and universal attacks then $M = M'$; moreover the chosen variable is the first variable in the enumeration that does not appear in G ,
4. if D, M and D, M' belong to S and M, M' are **O**-moves and existential defences then $M = M'$; moreover the term chosen to defend is the first variable in the enumeration that does not appear in G ,
5. if G belongs to S and the last move of G is an **O**-move that is a question on an existential quantifier, then G, M belongs to S , where M is enabled by the last move of G .

A strategy S is **P**-winning iff each game in S is won by **P**. Given that a strategy is a tree of games, in what follows we will sometimes speak of nodes of a strategy as a shortcut for moves of a game G that belongs to the strategy.

2.6 Validity

Definition 2 Given a first order formula F we say that F is *dialogically valid* if, and only if, there exists a winning strategy S for the the formula.

Fig. 1 shows three examples of winning strategies. The blue dotted arrows represent the function f that points back from **P**-moves to the **O**-move that enables them. Keep in mind that every **O**-move is enabled by the immediately preceding move.

We give an explanation of the strategy for the drinker theorem, i.e., the formula $\exists x(a(x) \Rightarrow \forall y a(y))$. Call the moves in the strategy, from the root to the leaf, $M_0, M_1 \dots M_8$.

The strategy starts by **P**-move asserting the drinker theorem. The subsequent move, M_1 by **O** ($?, \exists$) is an attack move directed toward the move M_0 . A possible paraphrase of M_1 would be: “could you choose a term to instantiate the existential formula you asserted?”. By condition 5 in the definition of strategy, the move M_2 must be a defence move enabled by M_1 . In the picture there is an arrow pointing back from M_2 to M_1 and the formula asserted in M_2 , $a(c) \Rightarrow \forall y a(y)$, is an answer to \exists . Since each **O**-move in a game is enabled by the immediately preceding **P**-move, and since a strategy is a tree of games, **O** has no choice but to attack the move M_2 by asserting $a(c)$. $a(c)$ is a question on the formula asserted by M_2 . The player **P** chooses (move M_4) to answer to $a(c)$ by asserting $\forall y.a(y)$. The player **O** attacks M_4 by choosing, as the definition of strategy prescribes, a variable w that does not appear in the Game, and asking the player **P** to assert $a[w/y]$, (move M_5). The player **P** cannot immediately answer to the question in M_5 : there is no **O**-move in the dialog that ends with M_5 that is an assertion of $a(w)$, thus by condition 2 in the definition of games, the move M_6 cannot be the assertion of $a(w)$. The player **P** decides, instead, to again answer the question \exists played by **O** in M_1 by instantiating the drinker theorem with the variable w , i.e., the move M_6 is $!(, (a(w) \Rightarrow \forall y.a(y)))$ and it is enabled by M_1 (this move is an existential repetition). The player **O** is obliged by the definition of game to assert, as move M_7 , $a(w)$ as an answer on the formula asserted by **P** in M_6 . At this point the player **P** can answer to the question $? \forall [w/x]$ in M_5 by making the move $M_8 = !(, a(w))$ and he wins the game.

3 Dialogical Validity is Equivalent to Classical Validity

In this section we show the equivalence, for a formula F between the dialogical validity of F (the existence of winning strategy for the proponent of F) and classical validity of F (F being true in all interpretations).

To prove this equivalence, we use a particular version of the sequent calculus, GKs (see Table 1). The calculus GKs (For Gentzen Kleene strategy) is equivalent to the sequent calculus GKc (Gentzen Kleene classical, see Table 2); GKc is complete for first order logic (see [39] pp. 84–86) in the sense that a sequent $\Gamma \vdash \Delta$ can be derived if and only if the formula $\bigwedge \Gamma \Rightarrow \bigvee \Delta$ classically valid.

Later on we shall consider a restriction (*strategic* derivations) on the use of the left implication introduction rule and on the use of the right existential introduction rule in the calculus GKs. This restriction does not affect the completeness of the sequent calculus with respect to validity. The restriction on the use of the left implication introduction rule was already studied by Herbelin in his PhD thesis [23] for a sequent

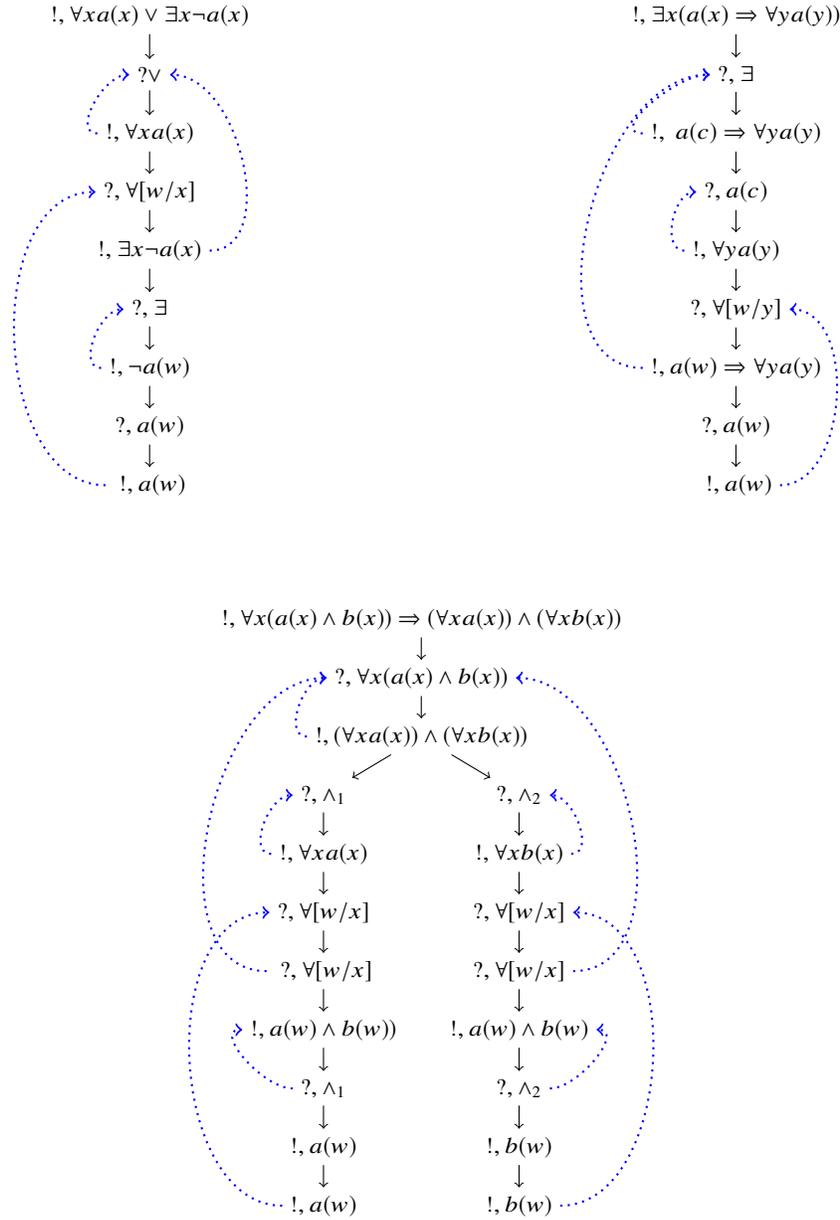


Fig. 1 Three winning strategies.

calculus for propositional classical logic called LKQ. The restriction on the use of the

left implication introduction rule was used by Herbelin to prove the correspondence between winning strategies for propositional classical logic and derivations in LKQ.

Definition 3 The sequent calculus GKs is defined by the rules in Table 1. The sequent calculus GKc is defined by the rules in Table 2. In both calculi greek upper-case letters Γ, Δ, \dots stand for multisets of formulas. In the *Id*-rule A is an atomic formula. In the $\forall R$ and $\exists L$ rules the variable y does not occur in the conclusion sequent. The bold formulas in the conclusion of each rule are called active formulas.

A derivation (or a proof) π of a sequent $\Gamma \vdash \Delta$ in GKs (resp. GKc) is a tree of sequents constructed according to the rules of GKs (resp. GKc) in which leaves are *Id*-rules (resp. *Id*-rules or $\perp L$ -rules) and whose root also called conclusion is $\Gamma \vdash \Delta$.

A sequent $\Gamma \vdash \Delta$ is said to be *derivable* or *provable* in a sequent calculus $GK \dots$ whenever there exists a proof with conclusion $\Gamma \vdash \Delta$. When the sequent is $\vdash F$ the formula F is said to be provable in $GK \dots$.

A binary (resp unary) rule R is said to be *admissible* in a sequent calculus GKX if one may derive from any sequents S_1, S_2 (resp. S_1) that are premises of R and, possibly axioms (also called identity rules) of GKX , the conclusion of R applied to S_1, S_2 (resp. S_1).

Remark 2 The calculi GKs and GKc differ in several aspects.

- First of all in GKc there is a rule for \perp (called *ex falso quod libet sequitur*) while there is no rule for \perp in GKs: this entails that in GKc the negation $\neg G$ of a formula G can be defined as $\neg G = G \Rightarrow \perp$, while in GKs, the formula $G \Rightarrow \perp$ is “almost” a negation, enjoying *tertium non datur* but not some other properties.
- Moreover each premise of a rule in GKc contains the active formula of the conclusion, while this is not the case in GKs. In GKs only the left introduction rules and the right introduction rule for the existential quantifier have this property. Finally the right introduction rule for disjunction, using the terminology of linear logic, is additive in GKc (contexts in premise sequent(s) and conclusion sequent are the same) and multiplicative in GKs (contexts may be different in the premise sequent(s) and are concatenated in the conclusion sequent).
- The sequent $\perp \vdash$ is provable in GKc but not in GKs, while $\perp \vdash \perp$ is provable in both GKc and GKs.

Proposition 4 *Contraction and weakening are admissible in GKs, i.e., for all A, Γ, Δ*

- if $\Gamma, A, A \vdash \Delta$ is provable in GKs then $\Gamma, A \vdash \Delta$ is provable in GKs
- if $\Gamma \vdash \Delta, A, A$ is provable then $\Gamma \vdash \Delta, A$ is provable in GKs
- if $\Gamma \vdash \Delta$ is provable in GKs then $\Gamma, A \vdash \Delta$ is provable in GKs
- if $\Gamma \vdash \Delta$ is provable in GKs then $\Gamma \vdash \Delta, A$ is provable in GKs

Proof (Sketch) The admissibility of contraction and weakening in GKs is an easy adaptation of the proof contained in [39, pp. 78–81]. \square

Proposition 5 *All rules of GKs are admissible in GKc*

Table 1 The GKs sequent calculus

$$\begin{array}{c}
\frac{}{\Gamma, A \vdash A, \Delta} Id \\
\\
\frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \Rightarrow B, \Delta} \Rightarrow R \qquad \frac{\Gamma, A \Rightarrow B \vdash A, \Delta \quad \Gamma, A \Rightarrow B, B \vdash \Delta}{\Gamma, A \Rightarrow B \vdash \Delta} \Rightarrow L \\
\\
\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \wedge R \qquad \frac{\Gamma, A, A \wedge B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \wedge L_1 \qquad \frac{\Gamma, B, A \wedge B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \wedge L_2 \\
\\
\frac{\Gamma \vdash A, B, \Delta}{\Gamma \vdash A \vee B, \Delta} \vee R \qquad \frac{\Gamma, A \vee B, A \vdash \Delta \quad \Gamma, A \vee B, B \vdash \Delta}{\Gamma, A \vee B \vdash \Delta} \vee L \\
\\
\frac{\Gamma \vdash \exists x A, A[t/x], \Delta}{\Gamma \vdash \exists x A, \Delta} \exists R \qquad \frac{\Gamma, A[y/x], \exists x A \vdash \Delta}{\Gamma, \exists x A \vdash \Delta} \exists L \\
\\
\frac{\Gamma \vdash A[y/x], \Delta}{\Gamma \vdash \forall x A, \Delta} \forall R \qquad \frac{\Gamma A(t), \forall x A \vdash \Delta}{\Gamma, \forall x A \vdash \Delta} \forall L
\end{array}$$

Proof One shows that if the premises of a rule R from GKs are derivable in GKc then the conclusion of R is derivable in GKc, using as base case the identity rule which is the same in the two system. The proof uses the admissibility of weakening and contraction for GKc [39]. We show just one case. All other cases are similar. Suppose that, in GKc, there is a derivation of the premise of the right rule introduction for \vee in GKs, i.e., that the sequent $\Gamma \vdash A, B, \Delta$ is derivable. We want to show that the sequent $\Gamma \vdash A \vee B, \Delta$ is derivable using the rules of GKc. By the admissibility of weakening we have a derivation of the sequent $\Gamma \vdash A, B, A \vee B, \Delta$ in GKc. We can then construct the following derivation using the rules of GKc

$$\frac{\frac{\Gamma \vdash A, B, A \vee B, \Delta}{\Gamma \vdash A, A \vee B, \Delta} \vee R_2}{\Gamma \vdash A \vee B, \Delta} \vee R_1$$

We now prove that GKs and GKc prove *almost* the same sequents.

Proposition 6 *Given a pair of multi-sets of formulas Γ, Δ , $\Gamma \vdash \Delta, \perp$ is provable in GKs if, and only if, $\Gamma \vdash \Delta$ is provable in GKc.*

Proof For the left to right direction. if $\Gamma \vdash \Delta, \perp$ is provable in GKc, then we can conclude, by Proposition 5 that $\Gamma \vdash \Delta, \perp$ is provable in GKc. Since GKc is complete for classical first order logic we can conclude that the formula $\bigwedge \Gamma \Rightarrow (\bigvee \Delta \vee \perp)$ is

Table 2 The GKc sequent calculus.

$$\begin{array}{c}
\frac{}{\Gamma, A \vdash \mathbf{A}, \Delta} Id \\
\frac{\Gamma, A \vdash B, A \Rightarrow B, \Delta}{\Gamma \vdash A \Rightarrow \mathbf{B}, \Delta} \Rightarrow R \qquad \frac{\Gamma, A \Rightarrow B \vdash A, \Delta \quad \Gamma, A \Rightarrow B, B \vdash \Delta}{\Gamma, A \Rightarrow \mathbf{B} \vdash \Delta} \Rightarrow L \\
\frac{\Gamma \vdash A, A \wedge B, \Delta \quad \Gamma \vdash B, A \wedge B, \Delta}{\Gamma \vdash \mathbf{A} \wedge \mathbf{B}, \Delta} \wedge R \qquad \frac{\Gamma, A_i, A_1 \wedge A_2 \vdash \Delta}{\Gamma, \mathbf{A}_1 \wedge \mathbf{A}_2 \vdash \Delta} \wedge L_i \\
\frac{\Gamma \vdash A_i, A_1 \vee A_2, \Delta}{\Gamma \vdash \mathbf{A}_1 \vee \mathbf{A}_2, \Delta} \vee R_i \qquad \frac{\Gamma, A \vee B, A \vdash \Delta \quad \Gamma, A \vee B, B \vdash \Delta}{\Gamma, \mathbf{A} \vee \mathbf{B} \vdash \Delta} \vee L \\
\frac{\Gamma \vdash \exists x A, A[t/x], \Delta}{\Gamma \vdash \exists \mathbf{x} \mathbf{A}, \Delta} \exists R \qquad \frac{\Gamma, A[y/x], \exists x A \vdash \Delta}{\Gamma, \exists \mathbf{x} \mathbf{A} \vdash \Delta} \exists L \\
\frac{\Gamma \vdash A[y/x], \forall x A, \Delta}{\Gamma \vdash \forall \mathbf{x} \mathbf{A}, \Delta} \forall R \qquad \frac{\Gamma A(t), \forall x A \vdash \Delta}{\Gamma, \forall \mathbf{x} \mathbf{A} \vdash \Delta} \forall L
\end{array}$$

valid. But this means that the formula $\wedge \Gamma \Rightarrow \vee \Delta$ is also valid. Using completeness again we can conclude that the sequent $\Gamma \vdash \Delta$ is provable in GKc.

For the right to left direction. Remark that all rules of GKc except the rule $\perp L$ are admissible in GKs because of the admissibility of contraction and weakening for GKs. However, if the rule $\perp L$ is used to prove a sequent $\perp, \Gamma \vdash \Delta$ in GKc, the sequent $\perp, \Gamma \vdash \Delta, \perp$ will be provable in GKs using an instance of the identity rule.

Remark 3 GKs is stronger than minimal logic; for example both Peirce's law and the law of excluded middle are provable in GKs. We show a derivation of Peirce's law and a derivation of the law of the excluded middle.

$$\begin{array}{c}
\frac{}{((a \Rightarrow b) \Rightarrow a, a \vdash b, a)} Id \\
\frac{(a \Rightarrow b) \Rightarrow a \vdash a \Rightarrow b, a}{(a \Rightarrow b) \Rightarrow a \vdash a} \Rightarrow R \qquad \frac{}{a, (a \Rightarrow b) \Rightarrow a \vdash a} Id \\
\frac{(a \Rightarrow b) \Rightarrow a \vdash a \quad a, (a \Rightarrow b) \Rightarrow a \vdash a}{(a \Rightarrow b) \Rightarrow a \vdash a} \Rightarrow L \\
\frac{}{\vdash ((a \Rightarrow b) \Rightarrow a) \Rightarrow a} \Rightarrow R
\end{array}$$

$$\frac{\frac{\overline{a \vdash a, \perp} \text{Id}}{\vdash a \Rightarrow \perp, a} \Rightarrow R}{\vdash (a \Rightarrow \perp) \vee a} \vee R$$

3.1 From Strategies to Proofs in GKs

Let us say a few words that justify the introduction of the calculus GKs. GKs is been chosen strategically because it is “easy” to map winning strategies of dialogical logic to its derivations. The fact that all binary rules of GKs are context sharing (or additive) is motivated by the fact that, as it is explained below, we recursively associate sequents to the nodes of a strategy starting from the root of the strategy. Using this methodology it would be hard to split the sequent in the right way as it is requested by a context splitting (or multiplicative) rule. The fact that all the left introduction rule of GKs carry the active formula of the conclusion in the premises of the rule is motivated by the fact that, as we will see below, left introduction “corresponds” to attack move by **P**. The player **P** can attack the *same* formula many times. This corresponds, in a GKs derivation, to a left introduction rule having the same active formula and being used many times in the derivation. The fact that the only right introduction rule in which the premise carries the active formula of the conclusion is the existential rule, is motivated by the following fact: right introduction rules corresponds to defence moves by **P** and **P** can answer to the same question on an existential formula many times. This corresponds, in a derivation in GKs, to an existential right rule having the same active formula and being used at different points of the derivations. Finally, the, admittedly odd, treatment of negation in GKs is motivated by the following fact: without such treatment the correspondence between strategies and derivations would be trickier and more tedious to prove. One should give a dialogical meaning to the constant \perp . This is not impossible but the definitions became less harmonious and the proof longer. We can now proceed with our correspondence proof.

Because sequent calculus GKs is sound for classical logic, the following proposition shows that a formula F with a winning strategy is true in any interpretation.

Proposition 7 *Given a formula F , if there is a winning strategy for F then there is a proof π of F in sequent calculus GKs.*

Proof This proposition results from the results below:

- The function $\pi(S)$ defined below associates to each **O**-move of a strategy S a sequent thus yielding a tree of sequents $\pi(S)$.
- The tree of sequents $\pi(S)$ enjoys the eigen variable condition. (Prop. 8))
- The tree of sequents $\pi(S)$ can easily be turned in a proof of GKs without losing the eigen variable property. (Prop. 9) □

Let us consider $D(S)$, the **O** restriction of a strategy S , the tree of **O**-moves only obtained from S (which is a tree of **P** and **O** moves) by forgetting all the **P**-moves and adding a root.

Given a strategy S for a formula F we associate, by induction of the length n of a sequence of **O**-moves $\mathbf{M} = M_1, M_3, \dots, M_{2n-1}$ of $D(S)$, a sequent $\Gamma_{\mathbf{M}} \vdash \Delta_{\mathbf{M}}$ to each sequence of **O** moves \mathbf{M} in $D(S)$ as follows:

1. If \mathbf{M} is the empty sequence, then $\pi(\mathbf{M}) = F$
2. if the sequence ends with an **O**-move which is an assertion:
 - a. if the assertion is a defence move $(!, F)$ then we associate the sequent $\Gamma, F \vdash \Delta$ where $\Gamma \vdash \Delta$ is the sequent associated to the prefix of the sequence
 - b. if the assertion is an attack move $(?, F)$ against an assertion (in the game) of $F \Rightarrow C$ then we associate the sequent $\Gamma, F \vdash C, \Delta$ where $\Gamma \vdash \Delta$ is the sequent associated to the prefix of the sequence from which we have erased the formula $F \Rightarrow C$ on the right of \vdash
3. if the sequence ends in a move that is not an assertion then it should be an attack, $(?, s)$ where s is an auxiliary symbol. Two cases may occur:
 - a. If s is either $?\forall, \wedge_1, \wedge_2$ or $\forall[w/x]$ then we associate the sequent $\Gamma \vdash \Delta'$ where Γ is equal to the Γ associated to the prefix of the sequence. Δ' is the sequent obtained from Δ , the sequent associated to the prefix of the sequence, in the following way: $\Delta' = \Delta - C \cup \{\Pi\}$ where C is the formula such that s is a question on C , and Π is the multiset of answers to s (remark that the formulas that are answers to s do not necessarily occurs in some move of the strategy).
 - b. if s is \exists then we associate the sequent $\Gamma \vdash F(t), \Delta$ where $F(t)$ is the formula asserted by **P** in its defence against $(?, s)$ and $\Gamma \vdash \Delta$ is the sequent associated to the prefix of the sequence. Remark that the **P**-defence must exist in the strategy by the definition of strategy 5.

The following proposition shows that the tree of sequents $\pi(S)$ satisfies the variable restriction on $\forall R \exists L$, known as the eigen variable condition:

Proposition 8 *Let S be a winning strategy and let $\pi(S)$ be the tree of sequent associated with S by the above procedure.*

Let \mathbf{M} be a sequence in $D(S)$ ending with a move that is

- *either an attack against a universal quantifier $(?, \forall[w/x])$*
- *or a defence against an existential attack $(!, A[w/x])$.*

Then the variable w does not appear free in the sequent associated to the proper prefix of \mathbf{M}

In order to prove Proposition 7 we just need some little syntactic manipulations on $\pi(S)$ to obtain a sequent calculus proof of $\vdash F$:

Proposition 9 *To each sequence of **O**-moves \mathbf{M} in $D(S)$ we can associate a derivation $\pi_{\mathbf{M}}$ of $\Gamma_{\mathbf{M}} \vdash \Delta_{\mathbf{M}}$ — the sequent associated to \mathbf{M} .*

Proof By well-founded induction on $(D(S), <)$. Suppose that for each suffix \mathbf{M} of \mathbf{M}' the proposition holds. We associate a derivation to \mathbf{M}' by considering the last move of M_{2n} $\Phi(\mathbf{M})$. Where $\Phi(\mathbf{M})$ is the unique game G in S ending in a \mathbf{P} -move such that \mathbf{M} is obtained from G by erasing \mathbf{P} -moves.

We only prove some cases which are not straightforward:

1. if M_{2n} is an attack $(?, A)$ on the assertion $A \Rightarrow C$ depending on the form of A
 - if A is atomic then the immediate suffix of \mathbf{M} is $\mathbf{M}(!, C)$ for which the proposition hold by hypothesis. We associate it with the following derivation.

$$\frac{\frac{\Gamma, A \Rightarrow C, A \vdash A}{\Gamma, A \Rightarrow C \vdash F} \text{Id} \quad \frac{\vdots \pi_{\mathbf{M}(!, C)}}{\Gamma, A \Rightarrow C, C \vdash F} \Rightarrow L}{\Gamma, A \Rightarrow C \vdash F} \Rightarrow L$$

- if $A = (A_1 \Rightarrow A_2)$ then \mathbf{M} has two immediate suffixes namely $\mathbf{M}, (? , A_1)$ and $\mathbf{M}, (!, C)$, for which the proposition holds by hypothesis. We associate the following derivation to \mathbf{M} .

$$\frac{\frac{\frac{\vdots \pi_{\mathbf{M}'(? , A_1)}}{\Gamma, (A_1 \Rightarrow A_2) \Rightarrow C, A_1 \vdash A_2} \Rightarrow R \quad \frac{\vdots \pi_{\mathbf{M}(!, C)}}{\Gamma, (A_1 \Rightarrow A_2) \Rightarrow C, C \vdash F} \Rightarrow L}{\Gamma, (A_1 \Rightarrow A_2) \Rightarrow C \vdash A_1 \Rightarrow A_2} \Rightarrow R}{\Gamma, (A_1 \Rightarrow A_2) \Rightarrow C \vdash F} \Rightarrow L$$

- If M_{2n} is an existential repetition asserting a formula $F[t/x]$ we proceed as follows: we only consider the case where $F[t/x] = (F_1 \Rightarrow F_2)[t/x]$. By induction hypothesis there is derivation of the sequent $\Gamma, F_1 \vdash \exists x F_1[t/x] \Rightarrow F_2[t/x], \Delta$ associated to the direct suffix $\mathbf{M}(?, F_1[t/x])$ of \mathbf{M} . We associate the following derivation to \mathbf{M} .

$$\frac{\frac{\frac{\vdots \pi_{\mathbf{M}(?, F_1)}}{\Gamma, F_1[t/x] \vdash \exists x(F_1 \Rightarrow F_2), F_2[t/x], \Delta} \Rightarrow R}{\Gamma \vdash \exists x(F_1 \Rightarrow F_2), (F_1 \Rightarrow F_2)[t/x], \Delta} \exists R}{\Gamma \vdash \exists x(F_1 \Rightarrow F_2), \Delta} \Rightarrow R$$

This proves Proposition 7 and thus assures us that that if a formula is dialogically valid then it is provable in sequent calculus, hence true in any interpretation.

3.2 From Proofs in GKs to Strategies

We have just shown that if a formula is dialogically valid then it is provable in sequent calculus GKs. We now show the converse, by turning a GKs sequent calculus deriva-

tion into a winning strategy, but we shall impose a restriction on the derivations of GKs — a restriction which derives exactly the same sequents.

Indeed not all derivations in GKs are the image of some winning strategy. For instance, the two derivations below where $c(x), a, b$ are atomic formulas are not the image of any winning strategy S although there are winning strategies for the two formulas (bold formulas are active occurrences of formulas in the sequent):

$$\frac{\frac{c(x) \vdash \mathbf{c(x)}}{\forall \mathbf{x c(x)} \vdash c(x)} \forall L}{\forall x c(x) \vdash \exists \mathbf{x c(x)}} \exists R \quad \frac{\frac{a \Rightarrow b, a \vdash \mathbf{a}, b}{a \Rightarrow b \vdash a, \mathbf{a} \Rightarrow \mathbf{b}} \Rightarrow R}{\vdash (a \Rightarrow b) \Rightarrow (a \Rightarrow b)} \Rightarrow R \quad \frac{\frac{b, a \Rightarrow b, a \vdash \mathbf{b}}{b, a \Rightarrow b \vdash \mathbf{a} \Rightarrow \mathbf{b}} \Rightarrow R}{\vdash (a \Rightarrow b) \Rightarrow (a \Rightarrow b)} \Rightarrow L$$

This leads us to restrict proofs of GKs to strategic proofs which derive the same sequents but always correspond to winning strategies, and to proceed as follows:

- We describe a procedure that turns a proof into strategy, by tree traversal from root to higher nodes — the order of traversal of daughters is irrelevant.
- By looking at how the derivation should be made in order for the procedure to be successful, we define a subclass of derivations of GKs called *strategic derivations*.
- We show that the subclass is complete, in the sense that if the sequent $\Gamma \vdash \Delta$ is provable then it corresponds to a strategic derivation.

We describe a procedure, that we call *p2s* (from a Proof in GKs to a strategy), that converts a proof π of a formula F into a winning strategy S for F . The procedure *p2s* explore the proofs π starting from the root and proceeding by level order traversal. The procedure associate to π a prefix closed set of games for the formula F .

Assume that for each node x of the proof of the formula F having depth n , the branch of the derivation from the root to x is already associated with a prefix closed set S_x of games for the formula F and also assume that each game maximal game G in S_x ends with an **O**-move.

The prefix closed set of games S_{a_1} associated with a_1 where a_1 is any daughter of a is defined as follows:

1. if a_1 is obtained by an identity rule $\Gamma, A \vdash A, \Delta$ then $S_{a_1} = S_a \cup \{G, (!, A)\}$ where A is the active formula of the identity rule and G is a maximal game in S_a such that $(!, A)$ is legal for G .
2. If a_1 is labelled with a sequent obtained from a right introduction rule with active formula A .

If A is not a conjunction nor a universal formula then $S_{a_1} = S_a \cup \{G, (!, A), (? , s)\}$ where G is a maximal game in S_a such that $(!, A)$ is legal for G and $(? , s)$ is an attack move such that s is a question on A

if A is $\forall x A'$ then $S_{a_1} = S_a \cup \{G, (!, \forall x A), (? , \forall [w/x])\}$ where G is a maximal game in S_a such that $(!, A)$ is legal for G and the variable w in $(? , \forall [w/x])$ is the variable that appears in the premise of a_1 but not in a_1 .

if A is $B \wedge C$ then $S_{a_1} = S_a \cup \{G, (!, A \wedge B), (? , \wedge_1)\} \cup \{G, (!, A \wedge C), (? , \wedge_2)\}$ where G is a maximal game in S_a such that the **P**-move $(!, A \wedge B)$ is legal for G

3. If a_1 is labelled with a sequent obtained from a left introduction rule with active formula A . If A is neither a disjunction nor an implication formula then $S_{a_1} = S_a \cup \{G, (? , q), (! , a)\}$ where G is a maximal game in S_a such that $(? , q)$ is legal for G , $(? , q)$ is a **P**-move where q is a question on A and $(! , a)$ is **O**-move such that the couple $(q, a) \in \text{Arg}(A)$.
- If A is $B \vee C$ then $S_{a_1} = S_a \cup \{G, (? , \vee)(! , B)\} \cup \{G, (? , \vee)(! , C)\}$ where G is a maximal game in S_a such that the **P**-move $(? , \vee)$ is legal for G .
 - if A is $B \Rightarrow C$ then $S_{a_1} = S_a \cup \{G, (? , B), (? , q_1)\} \cup \dots \cup \{G, (? , B), (? , q_n)\} \cup \{(G, (? , B), (! , C))\}$. Where G is a maximal game in S_a such that the **P**-move $(? , B)$ is legal for G each q_i is a question on B .

The above lines inductively define the mapping of a proof in GKs to a prefix closed set of games, but are all the obtained prefix closed sets of games strategies? Not always:

- If in (2) the active formula is an existentially quantified formula $\exists x B$ then **P** asserts the formula and next it is attacked by **O** with $(? , \exists)$. By the definition of a strategy, **P** has to assert $B[t/x]$. This means that a_1 should have just one daughter a_2 in which the formula $B[t/x]$ is active.
- A similar situation occurs in (3) when the active formula is a conditional $A \Rightarrow B$: **P** has to assert A , so A must be the active formula of the left premise of the \Rightarrow L rule.

In order to overcome this problem, we introduce the following definition:

Definition 4 (Strategic derivations) A derivation π in GKs is said to be *strategic* whenever it satisfies the two following conditions:

- for each application of a left implication introduction rule, the formula occurrence A in the left-hand premise is active.

$$\frac{\Gamma, A \Rightarrow B \vdash A, \Delta \quad \Gamma, A \Rightarrow B, B \vdash \Delta}{\Gamma, A \Rightarrow B \vdash \Delta} \Rightarrow L$$

- for each application of right existential introduction rule, the formula occurrence $A[t/x]$ is active in the premise:

$$\frac{\Gamma \vdash \exists x A, A[t/x], \Delta}{\Gamma \vdash \exists x A, \Delta} \exists R$$

Proposition 10 *If π is a strategic derivation of F then the procedure above outputs a winning strategy for F*

Given these last proposition we can conclude our proof by the following

Lemma 1 *For any multiset of formulas Γ, Δ there is a derivation of the sequent $\Gamma \vdash \Delta$ iff and only iff there is a strategic derivation of $\Gamma \vdash \Delta$*

Proof The direction from right to left is straightforward: each strategic derivation is a derivation in GKs.

The other direction results from a structural induction on the derivation π in GKs. All cases are straightforward except when π is obtained by π' by the application of a $\Rightarrow L$ rule or an $\exists R$ rule. Let us discuss the $\exists R$ rule, which together with a similar result in [23] dealing with all the propositional cases entail our proposition.

If π ends in a $\exists R$ rule application then, by induction hypothesis there is a strategic derivation π_1 of its premise $\Gamma \vdash A[t/x], \exists xA, \Delta$. if $A[t/x]$ is active we are done. If not we can suppose, without loss of generality that the rule application in which $A[t/x]$ is active is just above the last rule R of π_1 . The "hard" case is when R is a $\forall R$ rules, $A[t/x] = \exists yB(t, y)$, i.e., π_1 has the following shape:

$$\frac{\begin{array}{c} \vdots \\ \Gamma \vdash B(t, t'), A(y) \end{array}}{\Gamma \vdash \exists yB(t, y), D(y), \Delta'} \exists R \quad \frac{\Gamma \vdash \exists yB(t, y), D(y), \Delta'}{\Gamma \vdash \exists B(t, y), \forall wD, \Delta'} \forall R$$

The problem is that the term t' can contain a free occurrence of y . In this case we let permute the $\exists R$ upwards in this way:

$$\frac{\begin{array}{c} \vdots \\ \Gamma \vdash B(t, t'), A(y), \Delta' \end{array}}{\Gamma \vdash \exists yB(t, y), D(y), \Delta} \exists R \quad \frac{\Gamma \vdash \exists yB(t, y), D(y), \Delta}{\Gamma \vdash \exists xA, D(y), \Delta'} \exists R \quad \frac{\Gamma \vdash \exists xA, D(y), \Delta'}{\Gamma \vdash \exists xA, \forall wD, \Delta'} \forall R$$

This way we obtain the strategic proof we wanted. \square

This concludes our proof of the equivalence between winning strategies for our dialogical games and the existence of a proof in classical logic (here viewed, without loss of generality, as a strategic GKs proof).

Our result, the equivalence of proofs in GKs and winning strategies can be proved for a sequent calculus which is complete for classical logic. We did not do so, because the proof is much trickier and the intuitive meaning for \perp in games is obscure, so we prefer to present it for GKs which is equivalent to complete sequent calculus GKc in the sense of proposition 6.

4 Categorical Grammars and Automated Theorem Proving

Type-logical grammars are a family of frameworks for the analysis of natural language based on logic and type theory. Type-logical grammars are generally fragments of intuitionistic linear logic, with the Curry-Howard isomorphism of intuitionistic

logic serving as the syntax-semantics interface. Fig. 2 shows the standard architecture of type-logical grammars.

1. given some input text, a *lexicon* translates words into formulas, resulting in a judgment in some logical calculus, such as the Lambek calculus or some variant/extension of it,
2. the grammaticality of a sentence corresponds to the provability of this statement in the given logic (where different proofs can correspond to different interpretations/readings of a sentence),
3. there is a forgetful mapping from the grammaticality proof into a proof of multiplicative, intuitionistic linear logic,
4. by the Curry-Howard isomorphism, this produces a linear lambda-term representing the derivational meaning of the sentence (that is, it provides instructions for how to compose the meanings of the individual words),
5. we then substitute complex lexical meanings for the free variables corresponding to the lexical entries to obtain a representation of the logical meaning of the sentence,
6. finally, we use standard theorem proving tools (in first- or higher-order logic) to compute entailment relations between (readings of) sentences.

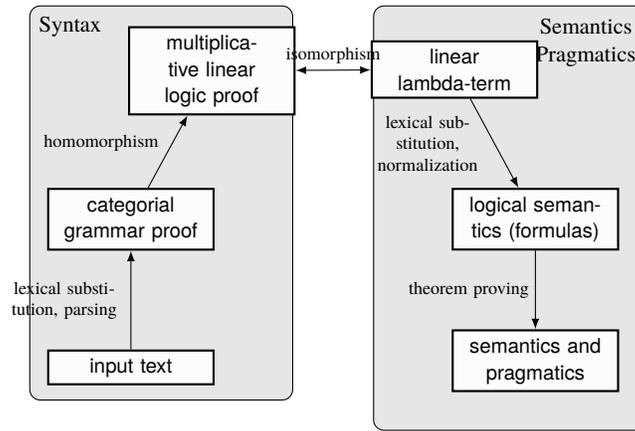


Fig. 2 The standard architecture of type-logical grammars

Table 3 The Lambek calculus

$$\frac{\Gamma \vdash A \quad \Delta \vdash A \setminus B}{\Gamma, \Delta \vdash B} [\setminus E] \quad \frac{\Gamma \vdash B/A \quad \Delta \vdash A}{\Gamma, \Delta \vdash B} [/E]$$

$$\frac{A, \Gamma \vdash B}{\Gamma \vdash A \setminus B} [\setminus I] \quad \frac{\Gamma, A \vdash B}{\Gamma, B/A} [/I]$$

Table 4 The multiplicative intuitionistic linear logic with linear lambda-term labeling

$$\frac{\Gamma \vdash N : A \quad \Delta \vdash M : A \multimap B}{\Gamma, \Delta \vdash (M N) : B} [-\multimap E] \quad \frac{x : A, \Gamma \vdash M : B}{\Gamma \vdash \lambda x.M : A \setminus B} [-\multimap I]$$

To make this more concrete, we present a very simple example, using the Lambek calculus. The Lambek calculus has two connectives⁴, A/B , pronounced *A over B*, representing a formula looking for a B constituent to its right to form an A , and $B \setminus A$, pronounce *B under A*, representing a formula looking for a B constituent to its left to form an A . Table 3 shows the logical rules of the calculus. We'll look at the French sentence 'Un Suédois a gagné un prix Nobel' (*A Swede won a Nobel prize*). Fig. 3 shows a Lambek calculus proof of this sentence. It shows that when we assign the formula n , for (common) noun, to 'prix' and $n \setminus n$ to 'Nobel', we can derive 'prix Nobel' as an n . Similarly, when we assign np/n to 'un' we can combine this with 'prix Nobel' of type n to produce 'un prix Nobel' as a noun phrase np . We can continue the proof as shown in Fig. 3 to show that 'Un Suédois a gagné un prix Nobel' is a main, declarative sentence s .

$$\frac{\frac{\frac{\text{Un}}{np/n} [Lex] \quad \frac{\text{suédois}}{n} [Lex]}{\text{Un, suédois} \vdash np} [/\!E] \quad \frac{\frac{\frac{\frac{\text{a}}{(np \setminus s_{main}) / (np \setminus s_{ppart})} [Lex]}{\text{a, gagné, un, prix, Nobel} \vdash np \setminus s_{main}} [/\!E] \quad \frac{\frac{\frac{\frac{\text{gagné}}{(np \setminus s_{ppart}) / np} [Lex]}{\text{gagné, un, (prix, Nobel} \vdash np \setminus s_{ppart}} [/\!E] \quad \frac{\frac{\frac{\frac{\text{un}}{np/n} [Lex]}{\text{un, prix, Nobel} \vdash np} [/\!E] \quad \frac{\frac{\frac{\frac{\text{prix}}{n} [Lex]}{\text{prix, Nobel} \vdash n} [/\!E] \quad \frac{\frac{\text{Nobel}}{n \setminus n} [Lex]}{\text{prix, Nobel} \vdash n} [/\!E]}}{\text{un, prix, Nobel} \vdash np} [/\!E]}}{\text{a, gagné, un, prix, Nobel} \vdash np \setminus s_{main}} [/\!E]}}{\text{Un, suédois, a, gagné, un, prix, Nobel} \vdash s_{main}} [/\!E]$$

Fig. 3 Lambek calculus proof of 'Un Suédois a gagné un prix Nobel' (*A Swede won a Nobel prize*).

The lambda-term of the corresponding linear logic proof is $(g(u p))(u s)$ (we have simplified a bit here, treating 'a gagné' and 'prix Nobel' as units). We then substitute the lexical semantics to obtain the logical representation of the meaning of this sentence. The simple substitutions are *suédois* for s and *prix_Nobel* for p . The two complicated substitutions are the two occurrences of u which are translated as follows.

$$\lambda P^{e \rightarrow t} \lambda Q^{e \rightarrow t} \exists x. [(P x) \wedge (Q, x)]$$

This is the standard Montague-style analysis of a generalised quantifier. It abstracts over two properties P and Q and states that there is an x which satisfies these two properties. Because of our choice of np for the quantifier (instead of a more standard higher-order type like $s/(np \setminus s)$), the type for the transitive verb has to take care of the quantifier scope. The lexical entry for the transitive verb below chooses the subject wide scope reading.

$$\lambda N^{(e \rightarrow t) \rightarrow t} \lambda M^{(e \rightarrow t) \rightarrow t} (M \lambda x. (N \lambda y. \text{gagner}(x, y)))$$

⁴ We ignore the product connectives ' \bullet ' here, since it has somewhat more complicated natural deduction rules and it is not used in the examples.

Substituting these terms into the lambda-term for the derivation and normalising the resulting term produces the following.

$$\exists x \exists y. [suédois(x) \wedge prix_Nobel(y) \wedge gagner(x, y)]$$

Even though this is an admittedly simple example, it is important to note that, although slightly simplified for presentation here, the output for this example and other examples in this paper are automatically produced by the wide-coverage French parser which is part of the Grail family of theorem provers [31]: Grail uses a deep learning model to predict the correct formulas for each word, finds the best way to combine these lexical entries and finally produces a representation of a logical formula. The full Grail output for the meaning of the example sentence is shown in Fig. 4. Grail uses discourse representation structures [26] for its meaning representation, which is essentially a graphical way to represent formulas in first-order logic. Besides providing a readable presentation of formulas, discourse representation structures also provide a dynamic way of binding, with applications to the treatment of anaphora in natural language.

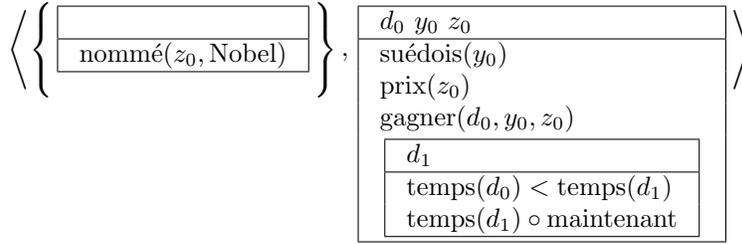


Fig. 4 Grail output for the semantics of ‘Un Suédois a gagné un prix Nobel’

The variables d_0 , y_0 and z_0 in the top part of the rightmost box represent existentially quantified variables, y_0 is a swede, z_0 is a prize (named after Nobel) and d_0 is a variable for an eventuality — essentially denoting a slice of space-time — the inner box indicates that this ‘winning’ event must have occurred at a time before ‘maintenant’ (*now*), i.e., in the past.

Even though the meaning assigned is in some ways simplistic, the advantage is that it can be automatically obtained and that it is of exactly the right form for logic-based entailment tasks.

5 Textual Entailment

What can dialogical argumentation contribute to the study of textual entailment? In natural language processing, *textual entailment* is usually defined as a relation between text fragments that holds whenever the truth of one text fragments follows from another text. *Textual entailment recognition* is the task of determining, given text fragments, whether the relation of textual entailment holds between these texts.

Our examples below are taken from the FraCaS benchmark, but translated into French. This is due to the fact that our methodology involves the use of Grail and the latter is developed mainly for the French language. Recently a French version of the FraCaS data set has been developed [2]. In this work we do not use this version and the examples are translated by us. In future works, however, we will evaluate examples taken from [2].

The FraCaS benchmark was built in the mid 1990s; the aim was developing a general framework from computational semantics. The data set consists of problems each containing one or more statements and one yes/no-question. An example taken from the data set is the following

- (1) A Swede won a Nobel prize.
- (2) Every Swede is a Scandinavian.
- (3) Did a Scandinavian win a Nobel prize? [Yes]

5.1 First Example

We illustrate our methodology to solve inference problems using examples. First of all we turn the question (3) into an assertion, i.e.,

- (4) Some Scandinavian won a Nobel prize.

We then translate each sentence in French and use Grail on each sentence in order to get a logical formula. In the enumeration below we report, in order: the sentence in English. A word-for-word translation, then a more natural paraphrase which takes into account French grammar and idioms, and, finally, the logical formula that Grail outputs from the input of the latter

- (5) A Swede won a Nobel prize
Un suédois a gagné un Nobel prix
Un suédois a gagné le prix Nobel
 $\exists x \exists y. [suédois(x) \wedge prix_Nobel(y) \wedge gagner(x, y)]$
- (6) Every Swede is a Scandinavian
Tout suédois est un scandinave
Tout suédois est scandinave
 $\forall u. (suédois(u) \Rightarrow scandinave(u))$

- (7) Some Scandinavian won a Nobel prize
 Un scandinave a gagné un Nobel prix
 Un scandinave a gagné un prix Nobel
 $\exists w.\exists z.[(scandinave(w) \wedge (prix_Nobel(z) \wedge gagner(w, z)))]$

We then construct a *winning strategy* for the formula $H_1 \wedge H_2 \dots \wedge H_n \Rightarrow C$ where each H_i is the logical formula that Grail associates to each statement from the data set, and C is the formula that Grail associates to the assertion obtained from the pair question-answer in the data-set.

$$F = \left(\exists x \exists y. [su(x) \wedge p_N(y) \wedge g(x, y)] \wedge \forall u. [su(u) \Rightarrow sc(u)] \right) \\ \Rightarrow \exists w \exists z. [(sc(w) \wedge (p_N(z) \wedge g(w, z)))]$$

In the above formula su stands for *suédois*, p_N for *prix_Nobel*, g pour *gagner* and sc for *scandinave*. A winning strategy for the formula F is shown in Fig. 5 in two steps.

5.2 Second Example

- (8) Some Irish delegates finished the survey on time.
 (9) Did any delegates finish the survey on time? [Yes]

The answer to the question is affirmative. This means that if (8) is true then the sentence “*some delegate finished the survey on time*” must also be true.

- (10) Some Irish delegates finished the survey on time
 Certains irlandais délégués ont terminé l' enquête à temps
 Certain délégués irlandais ont terminé l'enquête à temps
 $\exists x \exists y ((délégué(x) \wedge irlandais(x)) \wedge (enquête(y) \wedge terminé-à-temps(x, y)))$
- (11) Some delegates finished the survey on time
 Certains délégués ont terminé l' enquête à temps
 Certains délégués ont terminé l'enquête à temps
 $\exists x \exists y (délégué(x) \wedge (enquête(y) \wedge terminé-à-temps(x, y)))$

We have that $F_1 \Rightarrow F_2$. Where

$$F_1 = \exists x \exists y ((délégué(x) \wedge irlandais(x)) \wedge (enquête(y) \wedge terminé-à-temps(x, y)))$$

$$F_2 = \exists x \exists y (délégué(x) \wedge (enquête(y) \wedge terminé-à-temps(x, y)))$$

Fig. 6 shows the winning strategy for this formula.

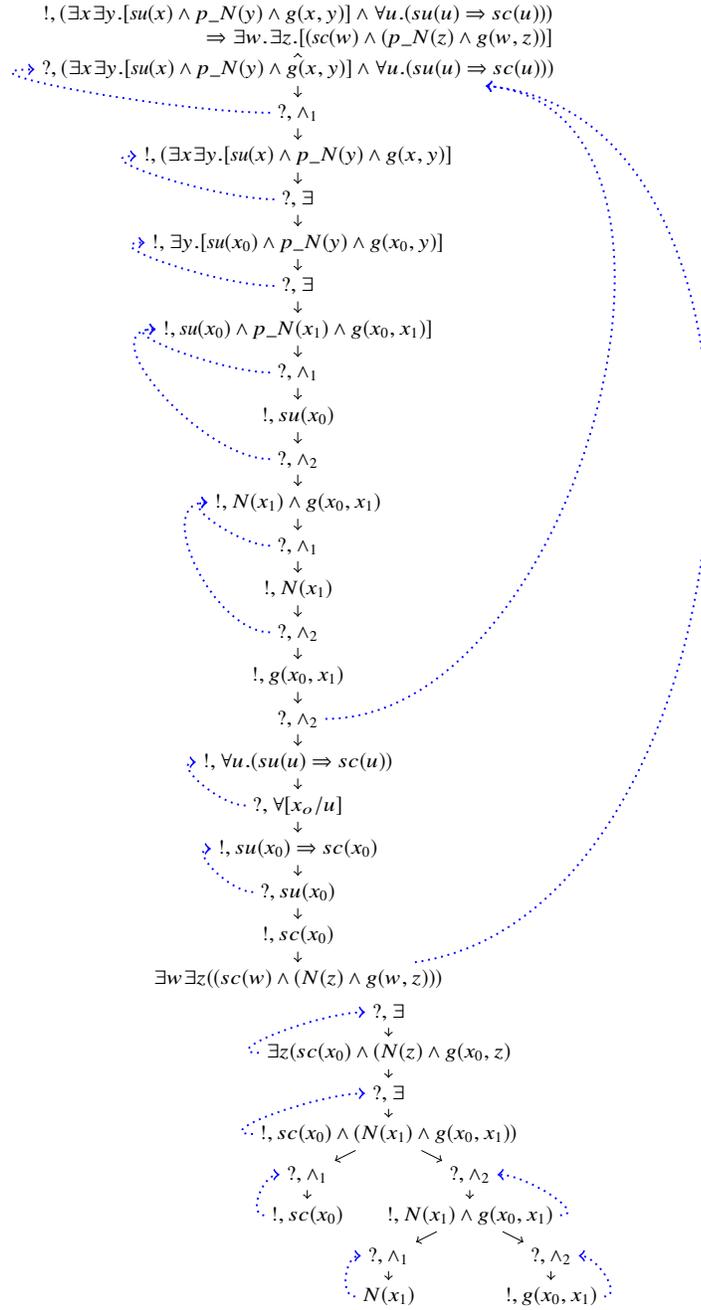


Fig. 5 Winning strategy showing entailment for the first example

- (13) Did any Scandinavian delegate finished the report on time? [No]

In this example, the question should get a negative reply. A positive answer would be implied by the existence of a Scandinavian delegate who finished the report in the time allotted. Thus the sentence (12) plus the sentence *Some Scandinavian delegate finished the report on time* should imply a contradiction. We first translate the two sentences in French and use Grail to get the corresponding logical formulas.

- (14) No delegate finished the report on time
 Aucun délégué n'a terminé le rapport à temps
 Aucan délégué n'a terminé le rapport à temps
 $\forall x(\text{délégué}(x) \Rightarrow \neg \text{terminé-le rapport-à-temps}(x))$
- (15) Some Scandinavian delegate finished the report on time
 Un scandinave délégué a terminé le rapport à temps
 un délégué scandinave a terminé le rapport à temps
 $\exists x((\text{délégué}(x) \wedge \text{scandinave}(x)) \wedge \text{terminé-le-rapport-a-temps}(x))$

The two formulas

$$F_1 = \forall x(\text{délégué}(x) \Rightarrow \neg \text{terminé-le rapport-à-temps}(x))$$

$$F_2 = \exists x((\text{délégué}(x) \wedge \text{scandinave}(x)) \wedge \text{termine-le-rapport-a-temps}(x))$$

Are contradictory. So there exists a winning strategy for the formula $\neg(F_1 \wedge F_2)$ as shown in Fig. 7. We recall that the expression $\neg F$ is just a shortcut for $F \Rightarrow \perp$, that enjoys most of the properties of the negations but not all of them. For reason of space in Fig. 7 two moves are omitted at the end of the strategies. Both of them are assertion move of the form $(!, \perp)$. The move closer to the root of the strategy is an **O**-move. The second, which is the leaf of the strategy, is a **P**-move enabled by the **O**-move that just below the root of the strategy.

5.4 Fourth Example

In the last example we focus on a series of sentences that our system should not solve, because the question asked neither has a positive nor a negative answer.

- (16) A Scandinavian won a Nobel prize.
 (17) Every Swede is a Scandinavian
 (18) Did a Swede win a Nobel prize? [Don't know]

This means that, on the basis of the information provided, we can neither say that a Swede has won a Nobel Prize nor that there are no Swedes who have won a Nobel Prize.

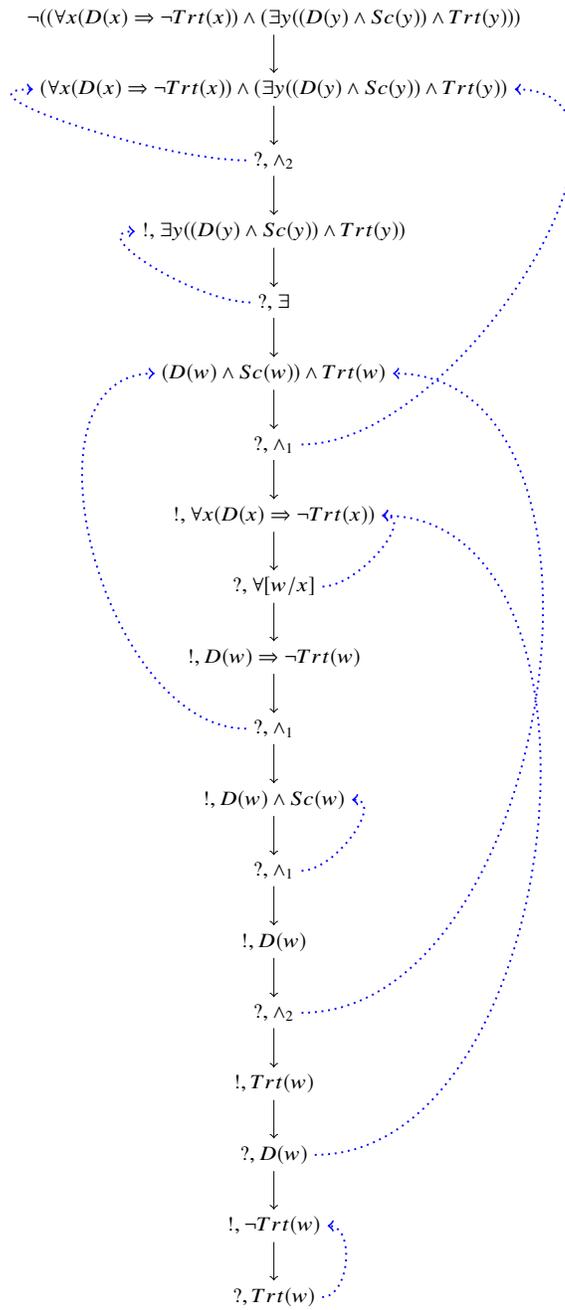


Fig. 7 Winning strategy showing contradiction for the third example.

- (19) A Scandinavian won a Nobel prize
 Un scandinave a gagné un Nobel prix
 Un scandinave à gagne un prix Nobel
 $\exists x \exists y (scandinave(x) \wedge (prix_Nobel(y) \wedge gagne(x, y)))$
- (20) Every Swede is a Scandinavian
 Tout suédois est un scandinave
 Tout suédois est scandinave
 $\forall u. (suédois(u) \Rightarrow scandinave(u))$

Call the formula in (19) F_1 and the formula in (20) F_2 . In dialogical logic terms, the fact that we do not have enough information to answer the question (18), either in a positive fashion or in a negative way, means that there is no winning strategy for the formula $F_1 \wedge F_2 \Rightarrow F_3$ nor for the formula $F_1 \wedge F_2 \Rightarrow \neg F_3$ where the formula F_3 is

$$F_3 = \exists w \exists z (suédois(w) \wedge (prix_Nobel(z) \wedge gagne(w, z)))$$

In general, given a sentence F of first order logic there it is not *decidable* whether F is valid. However *in some cases* we can manage this problem. Luckily the present case is one of those we can manage. We consider what a winning strategy for the formula $F_1 \wedge F_2 \Rightarrow F_3$ must look like. A winning strategy S for this formula will necessarily contain a dialog whose last move M_n is a **P**-move that asserts $suédois(t)$ for some term t in the language. Since $suédois(t)$ is an atomic formula by Proposition 3 above $suédois(t)$ must occur both as a positive and negative Gentzen sub-formula of $F_1 \wedge F_2 \Rightarrow F_3$ but this is not the case. Thus there is no winning strategy for the latter formula.

Let us now discuss why there is no winning strategy for the formula $F_1 \wedge F_2 \Rightarrow \neg F_3$. First of all, Proposition 1 guarantees that each game won by **P** ends with the assertion of some atomic formula, and that this assertion is a **P**-move. By Proposition 3 above, the only candidate for this is again $suédois(t)$ for all terms t in the language. If the **P**-move M_k is an assertion of $suédois(t)$, then it must be an attack. If it were a defence instead, this would mean that there must be a formula F of the form $\forall w. suédois(w)$ or $\exists w (suédois(w))$ or $F' \vee su'édoids(w)$ or $F' \wedge suédois(w)$ or $F' \Rightarrow suédois(t)$ such that **P** asserts F . This implies that this formula F must be a positive Gentzen subformula of $F_1 \wedge F_2 \Rightarrow \neg F_3$. But not such formula exists. Thus the move M_n asserting $suédois(t)$ must be an attack. Since the only formula that can be attacked by this means is the formula $suédois(t) \Rightarrow scandinave(t)$, **O** can answer by asserting $scandinave(t)$, and **P** cannot win the game. Thus there is no winning strategy for the formula $F_1 \wedge F_2 \Rightarrow \neg F_3$.

6 Conclusion

In this paper, we adapted our simple version of argumentative dialogues and strategies of [9] to two-sided sequents (hypotheses and conclusions): this point of view better

matches natural language statements, because the assumption sentences of a textual entailment task can be viewed as sequent calculus hypotheses, while the conclusion text can be viewed as the conclusion of the sequent.

In the present paper, we successfully use the syntactic and semantic platform Grail to “translate” natural language sentences into DRS that can be viewed as logical formulas.

This brings us closer to inferentialist semantics: a sentence S can be interpreted as all argumentative dialogues in natural language whose conclusion is S — under assumptions corresponding to word meaning and to the speaker beliefs.

We are presently working to extend our work with semantics modelled in classical first order logic to a broader setting which models semantics in first-order modal logic. Indeed, modal reasoning (temporal, deontic, alethic, etc.) is rather common in natural language argumentation.

Regarding the architecture of our model of natural language argumentation we would like to encompass lexical meaning as axioms along the lines of [9] and to use hypotheses to model the way the two speakers differ in their expectations, beliefs and knowledge, using insights from existing work on functional roles in dialogue modelling [38].

We plan to explore the connection between our restricted view of dialogue, which only concerns argumentative dialogues (i.e., games), and well-developed theories of discourse and dialogue such as the one presented in the books [3, 20] or the more innovative approach of [27] whose viewpoint is closer to ours.

References

1. Abramsky, S., McCusker, G.: Game semantics. In: U. Berger, H. Schwichtenberg (eds.) *Computational Logic*, pp. 1–55. Springer Berlin Heidelberg, Berlin, Heidelberg (1999)
2. Amblard, M., Beysson, C., de Groote, P., Guillaume, B., Pogodalla, S.: A French version of the FraCaS test suite. In: *LREC 2020 - Language Resources and Evaluation Conference* (2020)
3. Asher, N., Lascarides, A.: *Logics of conversation*. Cambridge University Press (2003)
4. Boritchev, M., Amblard, M.: Picturing questions and answers - a formal approach to slam. In: M. Amblard, M. Musiol, M. Rebuschi (eds.) *(In)coherence of discourse - Formal and Conceptual issues of Language, Language, Cognition and Mind*. Springer (2019). To appear.
5. Brandom, R.: *Articulating Reasons: An Introduction to Inferentialism*. Harvard University Press (2000)
6. Breitholtz, E.: *Enthymemes in dialogue: A micro-rhetorical approach*. Ph.D. thesis, Humanistiska fakulteten, Göteborgs universitet (2014)
7. Castelnérac, B., Marion, M.: Arguing for inconsistency: Dialectical games in the academy. In: G. Primiero (ed.) *Acts of Knowledge: History, Philosophy and Logic*. College Publications (2009)
8. Castelnérac, B., Marion, M.: Antilogic. *The Baltic International Yearbook of Cognition, Logic and Communication* **8**(1) (2013). DOI 10.4148/1944-3676.1079
9. Catta, D., Pellissier, L., Retoré, C.: Inferential semantics as argumentative dialogues. In: S. González, A. González-Briones, A. Gola, G. Katranas, M. Ricca, R. Loukanova, J. Prieto (eds.) *Distributed Computing and Artificial Intelligence, Special Sessions, 17th International Conference, Advances in Intelligent Systems and Computing*, pp. 72–81 (2020). URL https://doi.org/10.1007/978-3-030-53829-3_7

10. Cooper, R.: Update conditions and intensionality in a type-theoretic approach to dialogue semantics. In: R. Fernández, A. Isard (eds.) *Proceedings of the 17th Workshop on the Semantics and Pragmatics of Dialogue* (2013)
11. Cooper, R., Crouch, D., Eijck, J.V., Fox, C., Genabith, J.V., Jaspars, J., Kamp, H., Milward, D., Pinkal, M., Poesio, M., Pulman, S., Briscoe, T., Maier, H., Konrad, K.: *Using the framework* (1996). FraCaS deliverable D16
12. Cozzo, C.: *Meaning and Argument: A Theory of Meaning Centred on Immediate Argumental Role*. Stockholm Studies in Philosophy. Almqvist & Wiksell International (1994)
13. Dagan, I., Roth, D., Sammons, M., Zanzotto, F.M.: Recognizing textual entailment: Models and applications, *Synthesis Lectures on Human Language Technologies*, vol. 6(4). Morgan & Claypool Publishers (2013). URL <https://doi.org/10.2200/S00509ED1V01Y201305HLT023>
14. Dummett, M.A.E.: What is a theory of meaning? In: S. Guttenplan (ed.) *Mind and Language*. Oxford University Press (1975)
15. Dummett, M.A.E.: *The Logical Basis of Metaphysics*. Harvard University Press (1991)
16. Felscher, W.: Dialogues as a foundation for intuitionistic logic. In: D.M. Gabbay, F. Guenther (eds.) *Handbook of Philosophical Logic*, pp. 115–145. Springer Netherlands, Dordrecht (2002). DOI 10.1007/978-94-017-0458-8_2. URL http://dx.doi.org/10.1007/978-94-017-0458-8_2
17. Fouqueré, C., Quatrini, M.: Argumentation and inference a unified approach. In: *The Baltic International Yearbook of Cognition, Logic and Communication Volume 8: Games, Game Theory and Game Semantics*, pp. 1–41. New Paririe Press (2013)
18. Francez, N.: Proof Theoretical Semantics, *Studies in Logic*, vol. 57. College Publication (2015)
19. Frege, G.: The thought: A logical inquiry. *Mind* **65**(259), 289–311 (1956). DOI 10.1093/mind/65.1.289
20. Ginzburg, J.: *The Interactive Stance*. Oxford University Press (2012)
21. Girard, J.Y.: *Proof-Theory and Logical Complexity — vol. I*. Studies in Proof Theory. Bibliopolis, Napoli (1987)
22. Girard, J.Y.: Locus solum. *Mathematical Structures in Computer Science* **11**(3), 301–506 (2001)
23. Herbelin, H.: *Séquents qu'on calcule : de l'interprétation du calcul des séquents comme calcul de λ -termes et comme calcul de stratégies gagnantes*. Thèse d'université, Université Paris 7 (1995)
24. Hunter, J., Asher, N., Lascarides, A.: A formal semantics for situated conversation. *Semantics and Pragmatics* **11**, 1–52 (2018). URL <https://doi.org/10.3765/sp.11.10>
25. Hyland, M.: Game semantics. In: A. Pitts, P. Dybjer (eds.) *Semantics and Logics of Computation*, pp. 131–182. Cambridge University Press (1997)
26. Kamp, H., Reyle, U.: *From Discourse to Logic*. Kluwer Academic Publishers, Dordrecht (1993)
27. Lecomte, A.: *Meaning, Logic and Ludics*. Imperial College Press (2011)
28. Lorenzen, P., Lorenz, K.: *Dialogische Logik*. Wissenschaftliche Buchgesellschaft (1978). URL <https://books.google.fr/books?id=pQ5sQgAACAAJ>
29. Montague, R.: English as a formal language. In: B. Visentini (ed.) *Linguaggi nella Società e nella Tecnica*, pp. 189–224. Edizioni di Comunità, Milan, Italy (1970). (Reprinted in R. Thomason (ed) *The collected papers of Richard Montague* Yale University Press, 1974.)
30. Moot, R.: A type-logical treebank for french. *Journal of Language Modelling* **3**(1), 229–264 (2015). DOI <http://dx.doi.org/10.15398/jlm.v3i1.92>. URL <http://dx.doi.org/10.15398/jlm.v3i1.92>
31. Moot, R.: The Grail theorem prover: Type theory for syntax and semantics. In: S. Chatzikiriakidis, Z. Luo (eds.) *Modern Perspectives in Type Theoretical Semantics*, pp. 247–277. Springer (2017). DOI https://doi.org/10.1007/978-3-319-50422-3_10. URL https://doi.org/10.1007/978-3-319-50422-3_10
32. Moot, R.: The Grail family of theorem provers (syntactic and semantic parser) (2018). URL <https://richardmoot.github.io>. <https://richardmoot.github.io>

33. Moot, R., Retoré, C.: Natural language semantics and computability. *Journal of Logic, Language and Information* **28**, 287–307 (2019). URL <https://doi.org/10.1007/s10849-019-09290-7>
34. Moss, L.: Natural logic. In: S. Lappin, C. Fox (eds.) *The Handbook of Contemporary Semantic Theory*, 2 edn., pp. 559–592. Blackwell (2015)
35. Novaes, C.D.: Medieval "obligationes" as logical games of consistency maintenance. *Synthese* **145**(3), 371–395 (2005). URL <http://www.jstor.org/stable/20118602>
36. Prawitz, D.: The epistemic significance of valid inference. *Synthese* **187**(3), 887–898 (2012). DOI 10.1007/s11229-011-9907-7. URL <https://doi.org/10.1007/s11229-011-9907-7>
37. Retoré, C.: The Montagovian Generative Lexicon ΛT_{y_n} : a Type Theoretical Framework for Natural Language Semantics. In: R. Matthes, A. Schubert (eds.) 19th International Conference on Types for Proofs and Programs (TYPES 2013), *Leibniz International Proceedings in Informatics (LIPIcs)*, vol. 26, pp. 202–229. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2014). DOI 10.4230/LIPIcs.TYPES.2013.202. URL <https://doi.org/10.4230/LIPIcs.TYPES.2013.202>
38. Sabah, G., Prince, V., Vilnat, A., Ferret, O., Vosniadiou, S., Dimitracopoulou, A., Papademetriou, E., Tsivgouli, M.: What dialogue analysis can tell about teacher strategies related to representational changes. In: D. Kayser, S. Vosniadou (eds.) *Modelling Changes in Understanding: Case Studies in Physical Reasoning*, *Advances in Learning and Instruction*, pp. 223–279. Pergamon press (2000)
39. Troelstra, A.S., Schwichtenberg, H.: *Basic Proof Theory*. Cambridge University Press, USA (1996)
40. Wittgenstein, L.: *Philosophische Untersuchungen / Philosophical Investigations*. Oxford University Press (1953). Translated by G. E. M. Anscombe / Bilingual edition.