Chapter 6
Rule-Following and the Limits of Formalization: Wittgenstein's Considerations Through the Lens of Logic

Paolo Pistone

6.1 Introduction

The justification of logical rules stumbles upon a celebrated remark by Wittgenstein: it takes rules to justify the application of rules. This inner circularity affecting logic makes it difficult to explain what by following a rule (typically, by endorsing the alleged compulsion of deduction) is inexorably left implicit.

Loosely following Searle's revision (see Searle 1969) of Kant's vocabulary, I'll style "constitutive" those conditions (if any) on which the existence of the activity of (correctly) applying a given rule depends. Then, a series of well-known arguments (that I briefly recall in Sect. 6.2) seem to undermine the possibility to provide logical rules with such constitutive conditions, since the description of the latter would involve the use of the same rules they should be constitutive of; it would be indeed patently circular to explain the existence of a successful practice of correctly applying the rule of, say, *modus ponens* by relying on the existence of the practice of *modus ponens*!

On the other hand, in Sect. 6.3, I'll try to show that by exploiting the advances made by proof theory on the dynamics of logic, a different approach to rules is possible: my thesis is indeed that the focus on the concrete syntactic manipulation which underlies the application of logical rules within formal systems can represent a fertile direction where to look for a way out from the "blind spot" of circular explanations.

The formalization of logical inference, as it is clear from the developments of mathematical logic from Frege's *Begrisschrift* to most recent programming languages, may carry some arbitrariness: for instance, we know that Frege's formalism,

P. Pistone (✉)
Department of Philosophy, Communication and Visual Arts, Università Roma Tre,
via Ostiense 234, 00144, Rome, Italy
e-mail: heighteight@gmail.com

G. Lolli et al. (eds.), *From Logic to Practice*, Boston Studies in the Philosophy
and History of Science 308, DOI 10.1007/978-3-319-10434-8_6

natural deduction and sequent calculus actually code exactly the same notion of classical provability, though in quite different ways. Yet, the apparently superficial differences between these formal systems hide relevant structural information: for instance, Gentzen was able to show that, among those three, only sequent calculus exposes in a clear way the symmetries underlying classical inference, leading to his celebrated *Hauptsatz*.

It seems quite tempting to call those aspects regarding logical formatting the *subjective* side of logic, for at least two reasons: first, since they are generally considered as inessential to the characterization of "objective" logic (in our example, classical provability) and, second, since they are directly related to the way logic is actually written, that is, to the most significative observable form of access to logical inference. In particular, since writing is an activity concretely happening in space and time, it is quite natural to expect this syntactic dimension to be subject to the results and perspectives coming from computational complexity theory. Just to give an idea, taking λ-calculus or Turing machines as our favourite formalism for calculus will produce exactly the same "objective" class of computable functions, but with a very remarkable difference of grain: the single, atomic step of computation in λ-calculus (called β-reduction) requires a polynomial time to be simulated on a Turing machine; so to say, *what appears on one side as the immediate application of a rule takes quite a long time to be executed on the other*.

Finally, in the last section, one will find a technical sketch of the mathematical perspective of Girard's *transcendental syntax*, which reconstructs logic in the algebra and geometry of linear operators, building on proof-theoretic results coming from linear logic: such a reconstruction provides mathematical content to the philosophical reflections motivating this article, so as a concrete connection with computational complexity and its theoretical open questions (concerning the space and time in which formal devices evolve).

6.2 The Blind Spot of Rules

In Wittgenstein's *Tractatus*, it is stated that matters about logical formatting, i.e. about the choice of rules and notations for logic, cannot even be meaningfully formulated: these matters would indeed *say* something on the *logical form*, that is, on the way in which logical languages are involved in referential practices, i.e. somehow attached to the world; since the logical form plays a normative role, the one of assuring the indisputability of deduction, then it cannot be the object of discussion and challenge, since any challenge should presuppose it as a constitutive condition.

Logical conventionalism, with Carnap's distinction between *internal* and *external* questions (see Carnap 1950), was intended to save the theory of the logical form, eliminating at the same time all reference to "the mystic": "what one should be silent of" (Wittgenstein 2001), with Wittgenstein's words, became in Carnap's work the object of purely pragmatical considerations.

On the other hand, the celebrated criticism of conventionalism contained in Quine (1984), where it is stated that

> [...] if logic is to proceed mediately from conventions, logic is needed for inferring logic from the conventions. Quine (1984)

seems to show that the theory contained in the *Tractatus* cannot be reduced to a conventionalist one[1]; that criticism was indeed grounded on the remark that, in the description of logical rules, the explicitation of what is required in order to correctly apply those rules inexorably eludes the description: as Wittgenstein would say, it is *shown* but cannot be *said*; Quine's argument is just a variation on a theme published by Lewis Carroll forty one years before (see Carroll 1895). To make the long story short, Carroll's (and *a fortiori* Quine's) argument is based on the remark that in order to describe how to follow the rule of *modus ponens*

$$\frac{A \quad A \Rightarrow B}{B}\ (MP) \tag{6.1}$$

one is naturally led to produce an inflated argument which is nothing but another form (rather, a sequence of instances) of the rule of *modus ponens*, whose major premise is a formula expressing the content of the schema (6.1):

$$\frac{A \Rightarrow B \qquad \dfrac{A \quad A \Rightarrow ((A \Rightarrow B) \Rightarrow B)}{(A \Rightarrow B) \Rightarrow B}\ (MP)}{B}\ (MP) \tag{6.2}$$

which leads naturally to a more inflated argument and so on...the attempt at justifying our tentative to follow the logical convention expressed by the *modus ponens* schema, in definitive, ends up in an infinite regress.

A similar argument is notoriously part of Wittgenstein's discussion on rule-following in the *Philosophical Investigations*, where he observes that in order to act in accordance with what a given rule, say *modus ponens*, requires him to do (i.e. in order to apply the rule), one has to *interpret* a certain schema (e.g. the schema (6.1)); on the other hand, if a person were asked to justify, or simply to explain, his course of actions as an interpretation of the schema or, as one would say, as somehow *compelled* by the associated rule, then he would end up picking up another schema (think of (6.2)) to be interpreted, giving rise to an infinite regress. Wittgenstein claims, however, that in practice mutual comprehension takes place without any regress of that kind:

> §201 It can be seen that there is a misunderstanding here from the mere fact that in the course of our argument we give one interpretation after another; as if each one contented us at least for a moment, until we thought yet of another standing behind it. [...]

[1] This way of reading the *Tractatus* presupposes the thesis that there is a theoretical unity tying together the theory of the logical form and what Wittgenstein writes later on the following rules: in both cases, one indeed finds that constitutive conditions of "saying" cannot be expressed linguistically without falling into circularity or regress.

> §217 If I have exhausted the justifications I have reached bedrock, and my spade is turned. Then I am inclined to say: 'This is simply what I do'. Wittgenstein (2009)

Again, in the infinite regress, one would find himself trying to *say*, continuously missing the point, what is implicit in the adoption of the rule, that is, what he should content himself to *show* in his linguistic practice.

As we have already remarked, the appeal to the distinction between internal and external questions (so as to the more familiar distinction between *object language* and *metalanguage*) does not constitute a viable way out of the presented arguments for the regress: to assert that a rule is justified at the meta-level by an argument which employs the alter ego of the rule itself cannot indeed constitute a proper justification of the rule, since practically every rule could be justified in that way: choose a rule R, and then write a semantics for R, by exploiting a rule *meta-*R; with few technical work, soundness and completeness (for instance, if R enjoys a kind of subformula property, enabling thus an inductive argument) will be proved. There would be thus no difference in principle between the rule of *modus ponens*, the most unassailable of deductive rules, and the worst rule one can imagine!

On one side, as a consequence, rules, meta-rules and meta-languages dramatically fail to provide constitutive conditions for deduction; on the other, it seems quite natural to expect that logically correct rules must be somehow explained, since otherwise they would simply be *arbitrary* ones.[2]

The alternative conclusion that logical deduction is fundamentally ungrounded, since we cannot discursively express what the validity of its rules consists in, is discussed in epistemological terms in *On certainty*: Wittgenstein maintains that grammatical rules are in charge of establishing what is constitutively exempt from doubt, in the sense that one could not make sense of what a doubt whatsoever about it could mean, since he could not display nor imagine any use of it:

> §314. Imagine that the schoolboy really did ask: "And is there a table there even when I turn around, and even when no one is there to see it?"
>
> §315 [. . .] The teacher would feel that this was only holding them up, that this way the pupil would only get stuck and make no progress.[. . .] this pupil has not learned how to ask questions. He has not learned the game that we are trying to teach him. Wittgenstein (1969)

The idea conveyed by these examples is that the activity of rule-following, and thus, in the case of logic, the alleged normativity of deduction, is one whose constitutive conditions, when we try to grasp them, inexorably slip out of our hands, hiding themselves behind the mirror effects of interpretations (semantics and meta-languages) obsessively reproducing the same questions at higher levels: we cannot *say* what underlies any form of saying, so as we cannot see the blind spot which enables any form of seeing.

The point at the heart of the work of the late Wittgenstein is that if we cannot say anything about conditions of possibility of rules, then we cannot even raise

[2] This expectation clearly goes well beyond Wittgenstein's view on rules and logic, since he believed that logical rules are truly arbitrary ones, though enjoying a special position in language.

meaningful doubts concerning them: in the course of our infinite interpretative regress, "doubt gradually loses its sense" Wittgenstein (1969).

> §307. And here the strange thing is that when I am quite certain of how the words are used, have no doubt about it, I can still give no grounds for my way of going on. If I tried I could give a thousand, but none as certain as the very thing they were supposed to be grounds for. Wittgenstein (1969)

This is to say that whereas the adoption of a given system of rules (a linguistic game, so to say) commits to a structured arrangement of referential claims, its "semantic ascent", with Quine's words, nothing can be said about the legitimacy to impose such a structure on experience. This remark revives the "mystical" question: why just these rules? To make an example, the adoption of a certain system of rules for modal logic (say $S5$) naturally leads to the acceptance of truth claims regarding propositions on possible worlds ("Is water the same as H_2O in every possible world?" "Is gold the same as goodness only knows what in some possible world?"), but what is to assure us, not about the truth-value of those sentences but rather about the right by which we are authorized to take such sentences as demanding for a definite truth-value? In a word, to quote a celebrated passage from Kant, *quid iuris*?

The moral to be drawn from the problem of the logical form would be thus that no serious questioning of logical formatting can be raised: entitlement to consider "logic" this or that system of rules would thus reduce to a matter of mere convention or to Wittgenstein's Promethean attitude (the "mystic"). In definitive, logic would be thus simply incapable to answer to a matter of right.

6.3 Proofs as Programs, Rules as Behaviours

The aim of this section is to discuss the significance of logical formatting ("why just these rules?") from a more specifically logical (loosely technical) viewpoint: this will be achieved by privileging an approach directed to the *dynamics* of logic, heritage of the so-called Curry-Howard correspondence, which establishes a connection (indeed an isomorphism) between logical proofs and programs written in a functional language like λ-calculus: a proof, as isomorphic with a program, is considered not only as a mathematical object to be *constructed* but also as a program to be *executed*; for instance, a proof of $\forall x \in AB(x)$ corresponds to a program which, when taken as input an element $t \in A$, produces, once executed, a proof of $B(t)$.

Through the mathematical analysis of cut-elimination algorithms, directly corresponding to the normalization of terms in λ-calculus, the rules of logic, as we'll sketch, are characterized by the symmetries which discipline the use (i.e. the dynamics) of proofs/programs. Particular attention is given to the problem of the termination of executions, i.e. of the computations that represent the interaction between proofs and programs: a program can indeed be of some use only when its execution achieves a certain goal within a finite amount of time and, on the other side, that logicalness constitutes an answer to the question “does this program actually do what it is designed for?” is well known by computer scientists.

That logical syntax is in charge to tame the dynamics of programs by forcing termination,[3] is best seen from the viewpoint of the so-called *formulas-as-types* correspondence: by this, formulas become sets of proofs/programs satisfying the norms associated by logical rules to formulas. For instance, a program π is typed $A \Rightarrow B$ (and it is indeed a proof of $A \Rightarrow B$) when, for every proof/program σ-typed A, the execution of the application $(\pi)\sigma$ of π to σ terminates producing a proof-/program-typed B.

In this way, the *static* viewpoint which individuates rules by their *schemas*, for instance, the *modus ponens* schema, is replaced by a *dynamic* one, in which proofs manifesting the correct application of a rule are those whose execution satisfies opportune *behavioural* (i.e. observable through interaction) norms: rules are then not imposed *a priori* on proofs, but rather different proofs manifesting the same behaviour can be “typed” in the same way. The input of computer science is crucial here: a program is a source code written in a given language (λ-calculus, for instance); it is thus a “subjective” (in the sense explained) device; yet, one is generally only interested in what the program does when it is run. The behaviour of the program can in fact be largely independent from its code, and programs written in different languages can behave exactly in the same way. These considerations prompt the search for a behavioural explanation of rules, that is, one in which the “objective” content of a rule is expressed as a problem of *interface* between proofs (more detailed, though informal, descriptions of this bridge between logic and computer science can be found in Joinet (2009, 2011)).

From the viewpoint of sequent calculus, resolving a problem of interface corresponds to eliminate cuts: let’s see, through an easy example, how properties of rule-schemas can be derived from the behaviour of proofs through cut-elimination. Let’s suppose to cut an arbitrary proof Π[4] of conclusion $A \wedge B$ with a *cut-free* proof Λ:

$$\dfrac{\begin{array}{c}\vdots\ \Pi\\ \vdash A \wedge B\end{array}\qquad \dfrac{\begin{array}{c}\vdots\ \Lambda\\ \Delta, A, B \vdash \Gamma\end{array}}{\Delta, A \wedge B \vdash \Gamma}\,(L\wedge)}{\Delta \vdash \Gamma}\,\mathit{cut} \tag{6.3}$$

[3] And by that logical correctness, since the eliminability of cuts (which implies the termination of all reductions) has as a corollary the logical coherence of the calculus: since no introduction rule is given for the absurdity $\bot$, if a proof of $\bot$ existed, then it would reduce to a proof whose last rule is an introduction for $\bot$, thus no such proof can exist.

[4] In this section, I’ll adopt the convention to name sequent calculus proofs with capital Greek letters $\Pi, \Lambda, \ldots$ and with small letters $\pi, \lambda, \ldots$ the programs, respectively, associated.

In order not to let interaction fail, Π must be reduced to a proof that we can split into a proof Π_1 of conclusion A and a proof Π_2 of conclusion B:

$$\dfrac{\dfrac{\overset{\vdots\ \Pi_1}{\vdash A} \qquad \overset{\vdots\ \Lambda}{\Delta, A, B \vdash \Gamma}}{\Delta, B \vdash \Gamma}\,cut \qquad \overset{\vdots\ \Pi_2}{\vdash B}}{\Delta \vdash \Gamma}\,cut \tag{6.4}$$

that is, by cut-elimination, the request that Π interfaces with Λ has two consequences: first, the rule-schema for ∧ must have the form one might expect[5]

$$\frac{\vdash A \quad \vdash B}{\vdash A \wedge B}\ (L\wedge) \tag{6.5}$$

Second, it must be possible to transform a proof of conclusion $A \wedge B$ into one whose *last rule* is expressed by the schema (6.5). Behavioural norms impose thus, through cut-elimination, syntactical constraints on the form of rules and proofs.

A crucial remark in the argument above is that the rule-schema for ∧ has been derived by taking into consideration an arbitrary cut-free proof *dual* to an arbitrary proof of $A \wedge B$; more generally, let's say that two proof-programs are *polar* each other when they can be cut together giving rise to a terminating execution (i.e. cut-elimination converges); given a set T of proofs-programs, we can define the set $T^{\perp}$ as the set of all proof-programs polar to every element of T. A set of this form will be called an "abstract type" (remark that these definitions make no use of rule-schemas). A simple argument shows that an abstract type T satisfies the equation

$$T = T^{\perp\perp} \tag{6.6}$$

this equation asserts that T is *complete* in the sense that an arbitrary proof-program polar to an arbitrary proof-program in $T^{\perp}$ is already in T.

Let now $\mathcal{S}$ be the set of proofs/programs of a given formula A built following the rules of a certain logical syntax. By embedding the programs in $\mathcal{S}$ in a more general functional language, we can ask the nontrivial question: is $\mathcal{S}$ an abstract type? If we recognize the abstract type $\mathcal{S}^{\perp}$ as a set of tests for $\mathcal{S}$ and thus $\mathcal{S}^{\perp\perp}$ as the smallest abstract type containing $\mathcal{S}$, our question can be rephrased as

$$\mathcal{S}^{\perp\perp} \subseteq \mathcal{S}? \tag{6.7}$$

This way of embedding syntactic proofs into abstract types defined by their bipolar is reminiscent of the notion of *completion* in topology and functional analysis. It can easily be acknowledged that what (6.7) expresses is a form of completeness

[5] We neglect for simplicity matters regarding the discipline of contexts, which would require a more sophisticated argument to reach exactly the same conclusion.

(usually styled *internal completeness*; see Girard 2001) which makes no reference to semantics, that is, radically internal to syntax: it is a query on the internal properties of the logical format, that is, on the symmetries between rules: is anything missing? Is there anything syntax cannot see?

It is to be observed that a test for A isn't but a proof/program of $A \vdash \Gamma$, what, in the case in which $\Gamma = \emptyset$, means a possibly *incorrect* proof of $A \vdash$, that is, a possibly incorrect proof of $\neg A$: the interaction between a proof and a test can thus be interpreted as a *dispute* between two players (this perspective is made explicit in *ludics*; see Girard 2001). The focus on the dynamics of logic, so as the interest in wrong proofs, after all, is just a way to bring *dialectics* back at the heart of logic.

In definitive, what the view briefly sketched tries to achieve is an exposition of logic in which rules and their schemata are not primitive notions, but are derived from the recognition of the normative requirements necessary to avoid divergence, i.e. lack of mutual communication: if you want to sustain $A \wedge B$ and you want your proof to be accepted by others (i.e. your program to interface successfully), then (6.3) tells you that the principal rule of your proof *ought* to be a splitting between A and B.

A similar deconstruction of logical rules can be found in the tradition of proof-theoretical semantics based on Gentzen's *Hauptsatz* (see Prawitz 1973; Dummett 1991). For instance, Dummett describes the dynamics between proofs as the interaction between the proponent of an introduction rule (that he calls the "verificationist") and the proponent of an elimination rule (that he calls the "pragmatist"); the search for an equilibrium between the expectations of both leads to his notion of "logical harmony", corresponding to the requirement of eliminability of cuts.

In Dummett's and Prawitz's theories, though, the building blocks of logical proofs are rule-schemata (for instance, introduction rules); as a consequence, the recognition of something as being obtained through the correct application of a rule (be it harmonious or not) remains a precondition for their interactive explanation of logic, so that Wittgenstein's rule-following infinite regress still applies (see Cozzo (2004) for a brief discussion): Dummett himself, in Dummett (1978), declares that a satisfying justification of logical rules imposes to reject Wittgenstein's point about rule-following (e.g. by endorsing his proposal of a molecularist theory of meaning).

In the approach here sketched (which builds on Girard's *transcendental syntax* program, as it will be made more precise in the next section), on the contrary, rule-schemata do not constitute primitive notions, since proofs are represented by "pure" programs: this means that proofs, like programs in λ-calculus, can be written independently of logical rules; behavioural constraints are then used to retrieve rule-application through the notion of "last rule" (remark that the example (6.3) discussed above clearly recalls Dummett's "fundamental assumption" (Dummett 1991)). Explaining rules through the notion of polarity with respect to a given set of tests, i.e., again, pure programs, yields then a completely behavioural description of logic.

This strategy appears *prima facie* as providing a way out of the mirror effects, i.e. the infinite regress arguments; let's see how, for instance, in the case of Carroll's argument on the *modus ponens*, the step leading to the regress is due to the possibility to pass from the rule-schema of *modus ponens*

$$\frac{A \quad A \Rightarrow B}{B}\ (MP) \tag{6.8}$$

to the logical formula expressing it:

$$A \Rightarrow ((A \Rightarrow B) \Rightarrow B) \tag{6.9}$$

which is used as premise of the inflated argument. Notably, the conclusion that Carroll himself and many others (Dummett included) have drawn from the argument is that this passage is somehow to be forbidden: a rule should not be expressed by a logical sentence. The typical argument (a good recognition can be found in Engel (2005)) is that a rule represents a *dynamic* pattern of linguistic practice, whereas a logical sentence expresses a *static* content. Dummett (1973) makes appeal to Frege's notion of "assertoric force" to distinguish the assertion (a piece of concrete linguistic practice) of the sentence A as premise of a rule of inference from its mere occurrence in the formula $A \Rightarrow ((A \Rightarrow B) \Rightarrow B)$:

> [...] it cannot be the same, because then "*Peter is a Jew; if Peter is a Jew, Andrew is a Jew; therefore Andrew is a Jew*" would be the same as "*If both Peter is a Jew and if Peter is a Jew, then Andrew is a Jew, then Andrew is a Jew*" and it was precisely Lewis Carroll's discovery (in 'What the Tortoise said to Achilles') that it was not. Dummett (1973)

In contrast with such a diagnosis, in what follows I'll accept to pass from the rule-schema (6.8) to the inflated schema, but with the remark that, in the latter, the formula expressing the schema must occur as conclusion of a proof/program μ; here is a sequent calculus proof that could be associated to μ:

$$\dfrac{\dfrac{\dfrac{A \vdash A \quad B \vdash B}{A, A \Rightarrow B \vdash B}}{A \vdash (A \Rightarrow B) \Rightarrow B}}{\vdash A \Rightarrow ((A \Rightarrow B) \Rightarrow B)} \tag{6.10}$$

My thesis is that one is not compelled to go on inflating the argument: an explanation of *modus ponens* is given by showing that, given arbitrary *correct* proofs/programs σ, τ of conclusions, respectively, A and $A \Rightarrow B$, the interaction $(\mu)\sigma\tau$ is convergent, what amounts, essentially, to a (strong) normalization argument. Remark that, by representing a rule by a proof-program of a logical sentence, its dynamic content (so as the assertoric force of its premises and conclusions) is not lost, since the *application* of the rule (6.8) is translated into the *execution* of the program μ. Questioning (6.8) amounts then to ask for an argument which asserts the compatibility of μ with the interface induced by the introduction rules for A and $A \Rightarrow B$: in other words, Dummett's argument for logical harmony is retrieved without presupposing that proofs are built following rules-schemata.

Furthermore, one could well imagine to invert the process, namely, to accept μ as representing a correct rule and to adopt it as a test for proofs of A and $A \Rightarrow B$: one will say that σ and τ are correct when $(\mu)\sigma\tau$ is convergent. This isn't but a way to pose the issue of internal completeness: do all correct σ, τ actually come from your favourite syntax?

Summing up, the strategies just presented have the following form:

1. Start by μ and prove that correct proofs of A and $A \Rightarrow B$ are *polar* to μ: σ and τ are thus tests for μ, i.e. the rule is tested.
2. Take arbitrary σ, τ polar to μ and prove that they come, respectively, from proofs of A and $A \Rightarrow B$: μ becomes here a test for σ and τ, i.e. the rule is the tester.

> If a blind man were to ask me "Have you got two hands?" I should not make sure by looking. If I were to have any doubt of it, then I don't know why I should trust my eyes. For why shouldn't I test my *eyes* by looking to find out whether I see my two hands? *What* is to be tested by *what*? Wittgenstein (1969)

Regress is then translated into the balance between introduction and elimination rules, that is, in the acknowledgement of the inner symmetries of (good) syntax.

A fundamental objection to the argument just presented is that it is at least implausible that an argument for normalization can be carried over without appeal to some instance of *modus ponens*, so that circularity would not be eliminated. This is tantamount to saying that it is at least implausible that a serious argument for justifying logic be carried over without using logic somewhere (this remark was indeed the starting point of our discussion). It must be conceded then that such a form of circularity impedes any form of definitive foundation for logic.

By the way, this indirect form of circularity is not the same as the plain circularity exposed by Carroll's argument, since, as we are going to see, it puts on the foreground the syntactical complexity of normalization arguments: these are indeed usually very complex logical arguments and may heavily depend on the formalism chosen. Thus, if one were somehow sceptic about the normalization argument (for defendable reasons, as we'll point out), then he would find himself in the situation hypothesized by Carroll in his article, namely, that of accepting the premises of the *modus ponens*, accepting the schema of the rule and yet remaining perplexed about the conclusion.

> If A and B and C are true, Z *must* be true,“ the Tortoise thoughtfully repeated. That's another *Hypothetical*, isn't? And, if I failed to see its truth, I might accept A and B and C, and *still* not accept Z, mightn't I?" Carroll (1895)

Suppose indeed to have laid down a normalization proof for $(\mu)\sigma\tau$, where Σ, T vary among all the proofs of A and $A \Rightarrow B$ in a given formalism, which establishes a hyper-exponential run time for normalization (this is actually the case for classical and intuitionistic propositional logic): to accept such a proof would mean to accept that by composing σ and τ, a program representing a proof of B can be obtained by execution. By the way, can we reasonably expect to *see* the reduction of $(\mu)\sigma\tau$ terminate? What the normalization proof asserts is indeed the existence of a normal form, without providing a viable procedure to produce one (in a reasonable time). In the case of (6.3), to make another example, the theorem asserts the existence of a tower of exponentials bounding the time necessary to split the proof Π of a conjunction into a conjunction of proofs: our universe could not plausibly assist to such an event!

One could thus well imagine a "strict finitistic" tortoise accepting the schema expressed by μ but questioning the application of the rule, since the latter depends on an argument which provides no concrete mean to show that whose existence it asserts. Again, why not to imagine a "Turing-tortoise" refusing β-reduction since it takes too much time to be implemented on the tape of a Turing machine?

These examples are not meant to motivate the endorsement of a more or less strictly finitistic view on mathematics, but only to show how a (more or less) legitimate doubt could be raised on *formats* disciplining the use of deductive rules, a doubt that could not be raised from a purely rule-theoretic standpoint. A somehow Kantian issue on legitimation is recovered by observing that by adopting sequent calculus or λ-calculus, one is led to accept as valid something he cannot actually *see* as valid. Again: by what right?

Gödel's incompleteness enables us to push the matter even farther: it is well known that since cut-elimination implies coherence, normalization proofs for systems enabling inductive/recursive constructions (and more generally, impredicative second-order constructions) cannot be performed within those systems themselves. This can be read in terms of ordinal run times for normalization: the so-called proof-theoretic ordinal number is associated to every proof as a measure of the computational cost of its normalization. A worst-case run time, which embodies the general normalization proof, is then obtained as the least not proof-theoretic ordinal (usually, a limit ordinal): such a proof cannot be a proof in the system.

What this argument shows is that by incompleteness, in no way one can be certain of the logical correctness of a given format: in no way, one will satisfy once for all the demands by a sceptic tortoise, since normalization arguments must employ principles whose logical correctness is even harder to justify.

Here we stumble again on the already discussed circle of logic presupposing logic, but with a significant quality/price advantage: in his remarks, Wittgenstein clearly stated that all questioning must stop somewhere, namely, when the "bedrock" is reached, i.e. the practice of a given "grammar" and all that is implicit in it. Now, the proof-theoretical and/or computational complexity of strong normalization proofs fills the gap between what can be directly seen and what must be left implicit in the use of formal proofs.

In this sense, the reformulation of Kant's *quid iuris* question in the preceding section reduces to a question on the implicit complexity of syntactic manipulation: can we actually write this? Can we do that in a reasonable time and space?

Summing up, if the argument presented is right (or at least, not drastically wrong, given its quite informal and sketchy description), then the infinite regress of logical rules interpreting themselves must stop at the level of formats, namely, at the level of syntactical prescriptions. As a consequence, a rigorous mathematical exposition of these preconditions of logic (which demands for more sophisticated techniques than the one sketched here; see the next section) might replace the Wittgensteinian "thereof we must be silent" and open the way for an exploration of what lies behind the apparently unquestionable evidence of the practice of logic and its rules.

6.4 Linearity and Logical Syntax

The search for an internal explanation of logical rules, through a profound mathematical investigation of formal systems, represents the core of the *transcendental syntax* program (see Girard 2011b), launched in 2011 by Jean-Yves Girard as the foundational counterpart to the refinement of proof-theoretical tools obtained through linear logic and its developments (for instance, *ludics* and *geometry of interaction*). In what follows, I'll try to give a hint at how, through the mathematical treatment of linearity in logic, it is possible to give ground to the discussions of the previous sections and to develop a promising framework to reconstruct logic.

From the viewpoint of syntax, linearity is obtained when the structural rules of *weakening* and *contraction*

$$\frac{\vdash \Gamma}{\vdash \Gamma, A}\,(W) \qquad\qquad \frac{\vdash \Gamma, A, A}{\vdash \Gamma, A}\,(C) \tag{6.11}$$

are eliminated; logically speaking, this amounts to reject the logical principles $\bot \Rightarrow A$ and $A \Rightarrow A \wedge A$. This choice, which could seem at first rather arbitrary, is explained by observing that, in sequent calculus, the structural rules (W) and (C)[6] form an independent group with respect to identity rules (axiom and cut) and logical rules (introduction and elimination of connectives).

Historically, the first reason for styling "linear" the fragment of logic without weakening and contraction[7] was a semantical one: linear logic proofs could indeed be interpreted as *linear maps* between coherent spaces, that is, stable maps on cliques (see Girard 1987a) satisfying

$$f\left(\sum_i^k a_i\right) = \sum_i^k f(a_i) \tag{6.12}$$

where the sum is intended as a disjoint union of sets. However, it was only with the development of *proofnet* theory and *geometry of interaction* (*GoI* in the following) that it was realized that linear proofs could be represented as linear operators in the traditional, matricial, sense. These operators act on the (usually complex) linear space generated by the occurrences of atomic formulas in the formulas to be proved: more precisely, to every occurrence of an atomic formula A, a finite linear space

[6] Along with the rule (Ex) of *exchange* which allows to permute the order of appearance of occurrences of formulas in a sequent and whose rejection leads to *non-commutative logic* NL, see Abrusci and Ruet (2000).

[7] Which happens to be very well organized, with classical connectives splitting into a *multiplicative* and an *additive* version, and with modalities (known as *exponentials*) reintroducing, in a linear setting, controlled versions of weakening and contraction.

$|A|$ is associated; to the general formula $F(A_1, \dots, A_n)$, where the A_i denote the occurrences of atomic formulas in F, the linear space $|F| = \bigoplus_i^n |A_i|$ is associated. Remark the *locativity* of this approach: every occurrence of formula is assigned a distinct *location* (i.e. an independent linear space); for instance, to construct the occurrence space of $(A \Rightarrow A) \Rightarrow (A \Rightarrow A)$, one first has to distinguish the four distinct occurrences of A as $(A_1 \Rightarrow A_2) \Rightarrow (A_3 \Rightarrow A_4)$ and then to associate to the formula the linear space $|A_1| \oplus |A_2| \oplus |A_3| \oplus |A_4|$.

Now, if we have a look at how structural rules are reflected in cut-elimination, we notice that they cause the dimension of the "occurrence space" not to be preserved during the computation: for instance, in the case of contraction, this dimension is, in the worst case, multiplied by the number of contracted occurrences:

$$\dfrac{\dfrac{\begin{array}{c}\vdots\,\Pi\\ \vdash \Gamma, A, A\end{array}}{\vdash \Gamma, A}\,(C) \qquad \begin{array}{c}\vdots\,\Lambda\\ \vdash \Delta, \neg A\end{array}}{\vdash \Gamma, \Delta}\,cut \quad \rightsquigarrow \quad \begin{array}{c}\dfrac{\dfrac{\begin{array}{c}\vdots\,\Pi\\ \vdash \Gamma, A, A\end{array} \quad \begin{array}{c}\vdots\,\Lambda\\ \vdash \Delta, \neg A\end{array}}{\vdash \Gamma, \Delta, A}\,cut \qquad \begin{array}{c}\vdots\,\Lambda\\ \vdash \Delta, \neg A\end{array}}{\vdash \Gamma, \Delta, \Delta}\,cut\\ \hline\hline \vdash \Gamma, \Delta \end{array}\,(C) \tag{6.13}$$

In case of nested contractions, this phenomenon is responsible for the hyper-exponential growth rates of reductions in classical logic.

On the other hand, in the absence of those principles, the dimension of the occurrence space is preserved (the space of all the occurrences of formulas strictly shrinking during reduction), as shown in the following example: selectfont

$$\dfrac{\dfrac{\begin{array}{c}\vdots\,\Pi_1\\ \vdash \Gamma_1, A\end{array} \quad \begin{array}{c}\vdots\,\Pi_2\\ \vdash \Gamma_2, B\end{array}}{\vdash \Gamma, A \wedge B} \qquad \dfrac{\begin{array}{c}\vdots\,\Lambda\\ \vdash \neg A, \neg B, \Delta\end{array}}{\vdash \neg A \vee \neg B, \Delta}}{\vdash \Gamma, \Delta}\,cut \quad \rightsquigarrow \quad \dfrac{\begin{array}{c}\vdots\,\Pi_2\\ \vdash \Gamma_2, B\end{array} \quad \dfrac{\begin{array}{c}\vdots\,\Pi_1\\ \vdash \Gamma_1, A\end{array} \quad \begin{array}{c}\vdots\,\Lambda\\ \vdash \neg A, \neg B, \Delta\end{array}}{\vdash \neg B, \Gamma_1, \Delta}\,cut}{\vdash \Gamma, \Delta}\,cut \tag{6.14}$$

The matricial representation of a linear proof in *GoI* is given by the symmetry (called *wire*) on the occurrence space representing the exchange between the dual occurrences of formulas in the axioms of the proof; for instance, to the following proof Π[8]

$$\dfrac{\dfrac{\vdash A, \sim A \quad \vdash B, \sim B}{\vdash A \otimes B, \sim A, \sim B}}{\vdash A \otimes B, \sim A \parr \sim B} \tag{6.15}$$

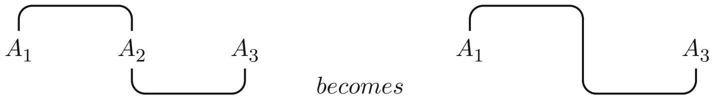
[8] Where $\sim, \otimes, \parr$ denote, respectively, linear negation, multiplicative conjunction (*tensor*) and multiplicative disjunction (*par*).

one straightforwardly associates the symmetry below, acting on the space generated by $A_1, B_1, \sim A_2, \sim B_2$ (isomorphic to $\mathbb{C}^4$),[9] which exchanges, respectively, $|A_1|$ with $|\sim A_2|$ and $|B_1|$ with $|\sim B_2|$:

$$\pi = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \tag{6.16}$$

Not every symmetry on the occurrence space comes from a proof of the formula associated; this is because the representation of proofs as wires radically forgets about rules (except axiom rules): it erases all sequentiality (i.e. rule-application) from the proofs.

In order to characterize the wires which represent correct sequent calculus proofs, the elegant solution, coming from proofnet theory, is to represent rules as *tests* to be performed on the wires: every logical rule is interpreted as a set of permutation matrices on the occurrence space, so that inductive composition of rules produces a set of general permutations τ_i (which can still be seen as wires) on the occurrence space, that can interact with the wire. In *GoI*, interaction (cut-elimination) can be directly implemented on wires: the intuition behind that is that the interaction of two wires is obtained by "plugging" them together to produce another wire:



(6.17)

In geometrical terms, this means to go through all the paths produced alternating the two wires. Let's see a little more in detail how to use this idea to interpret cut-elimination in sequent calculus: the cut rule (in locative form)

$$\cfrac{\begin{array}{c}\vdots\ \Pi \\ \vdash \Gamma, A_1\end{array} \quad \begin{array}{c}\vdots\ \Lambda \\ \vdash\sim A_2, \Delta\end{array}}{\vdash \Gamma, \Delta}\ cut \tag{6.18}$$

is represented (as every rule) as a test given by the following symmetry on the occurrence space which is the direct sum of the occurrence spaces of π and λ:

$$\sigma = \begin{pmatrix} I & 0 & 0 & 0 \\ 0 & 0 & I & 0 \\ 0 & I & 0 & 0 \\ 0 & 0 & 0 & I \end{pmatrix} \tag{6.19}$$

[9] In this section, I'll adopt the convention to name sequent calculus proofs with capital greek letters $\Pi, \Lambda, \ldots$ and with small letters $\pi, \lambda, \ldots$ the wires, respectively, associated.

σ acts as the identity on the subspace $|\Gamma| \oplus |\Delta|$ and as a symmetry on the space $|A_1| \oplus | \sim A_2|$ exchanging every atomic occurrence in A_1 with its dual one in $\sim A_2$.

We interpret the cut-elimination algorithm as follows: suppose to start in a point in $|\Gamma|$ and to reach, by the symmetry π, a point in $|A_1|$; then, by the symmetry σ, you are sent to a point in $|A_2|$, and two distinct possibilities arise:

- The symmetry λ sends you into $|\Delta|$, and you're done, since you've exited the "combat zone".
- λ sends you to another point in $|A_2|$; by σ, you are sent back into $|A_1|$. Again, two distinct possibilities arise....

Strong normalization (i.e. logical correctness) warrants that you will always exit $|A_1| \oplus |A_2|$ (towards $|\Gamma| \oplus |\Delta|$) in a finite amount of time, that is, that you won't get into a cycle in $|A_1| \oplus |A_2|$.

To simplify things, let's suppose $\Delta = \emptyset$; then, given the block decomposition,

$$\pi = \begin{pmatrix} \pi_{\Gamma\Gamma} & \pi_{\Gamma A} \\ \pi_{A\Gamma} & \pi_{AA} \end{pmatrix} \tag{6.20}$$

the general algorithm can be translated into the expression

$$\pi_{\Gamma\Gamma} + \pi_{\Gamma A}\big(\sigma + \sigma\pi_{AA}\sigma + \sigma\pi_{AA}\sigma\pi_{AA}\sigma + \cdots + \sigma(\pi_{AA}\sigma)^n + \ldots\big)\pi_{A\Gamma} \tag{6.21}$$

which, if the series $\sum_n^\infty (\pi_{AA}\sigma)^n$ converges, corresponds to the algebraic expression[10]

$$\pi_{\Gamma\Gamma} + \pi_{\Gamma A}\sigma\big(I - \pi_{AA}\sigma\big)^{-1}\pi_{A\Gamma} \tag{6.22}$$

Strong normalization is then translated into the *nilpotency* (corresponding to acyclicity) of $\pi_{AA}\sigma$, namely, the existence of an integer N, the *order of nilpotency*, such that $(\pi_{AA}\sigma)^N = 0$, what trivially implies convergence: nilpotency reduces indeed the expression (6.22) to the finite one below

$$\pi_{\Gamma\Gamma} + \pi_{\Gamma A}\big(\sigma + \sigma\pi_{AA}\sigma + \sigma\pi_{AA}\sigma\pi_{AA}\sigma + \cdots + \sigma(\pi_{AA}\sigma)^N\big)\pi_{A\Gamma} \tag{6.23}$$

corresponding to a finite computation.

It is now possible to implement typing through tests: let's say that two wires σ, τ on the same occurrence space are *polar* (notation, $\sigma \perp \tau$) when $\sigma\tau$ is nilpotent, and let $S^\perp := \{\tau \in \mathcal{M}(\mathcal{X}) | \sigma \perp \tau, \sigma \in S\}$ for every set S of wires on a given occurrence space $\mathcal{X}$. We recover thus logical types as the "abstract types" of the previous section: they are polar to sets of tests induced by logical rules, that is, as

[10] A simple calculation shows that, for u a linear operator, $(I - u)^{-1} = \sum_n^\infty u^n$, given the convergence of the series on the right or, equivalently, the invertibility of $I - u$.

the sets of *all* the wires following a given sequence of rules. The concrete translation of the rules of logic into tests follows the ideas in Girard (1987b), but I won't enter here into the details.

Notice that this framework enables the issue of *internal completeness*: do all wires in a given logical type come from actual sequent calculus proofs? A fundamental result, giving substance to the whole theory at the end of the eighties, was the theorem establishing completeness in the case of multiplicative linear logic (see Girard 1987b).

The justification for *modus ponens* we gave in the previous section can now be translated in the vocabulary of *GoI* as follows: given a wire σ on the occurrence space $|A|$ representing a proof of A and a wire ϕ on the occurrence space $|A| + |B|$ representing a proof of $A \Rightarrow B$, we can obtain a wire $[\phi]\sigma$ (intuitively, the application of ϕ to σ) on the occurrence space $|B|$ representing a proof of B; in fact ϕ can be pictured as the block matrix

$$\phi = \begin{pmatrix} \phi_{AA} & \phi_{AB} \\ \phi_{BA} & \phi_{BB} \end{pmatrix} \tag{6.24}$$

where ϕ_{XY} represents a linear operator from $|X|$ to $|Y|$. Following (6.22), $[\phi]\sigma$ is defined as follows:

$$[\phi]\sigma = \phi_{BB} + \phi_{BA}\sigma(I - \phi_{AA}\sigma)^{-1}\phi_{AB} \tag{6.25}$$

what can be done only in case the series $\sum_i^{\infty}(\phi_{AA}\sigma)^n$ converges (i.e. $I - \phi_{AA}\sigma$ is invertible). As expected, for wires coming from logical syntax, convergence is assured by the proof that $\phi_{AA}\sigma$ is nilpotent (this constituted in fact the main theorem of *GoI* – see Girard (1989)).

Remark that the nilpotency order N can be arbitrarily big, and it can be read as a measure of the complexity involved in the computation: it provides indeed a bound on the number of iterations necessary to reach convergence. Consider the following (linear) proof

$$\dfrac{\dfrac{A_3 \vdash A_1 \quad A_2 \vdash A_4}{A_1 \Rightarrow A_2, A_3 \vdash A_4}}{A_1 \Rightarrow A_2 \vdash A_3 \Rightarrow A_4} \tag{6.26}$$

represented by the wire

$$\phi = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \tag{6.27}$$

Let's call σ the simple wire $\left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$ on the occurrence space corresponding to the first two occurrences of A in the previous proof, representing the axiom $A \Rightarrow A$; we can easily compute $[\phi]\sigma$ and find that the nilpotency order is bounded by 2, the dimension of the occurrence space in which ϕ and σ actually "meet". Again, no room for doubt, no room for the Tortoise.

The wire ϕ corresponds to the λ-calculus term $\lambda f.\lambda x.(f)x$, usually taken as the *Church numeral* $\underline{1}$. On the other hand, the general Church numeral $\underline{n} = \lambda f.\lambda x.(f)^n x$ requires the encoding in *GoI* of n contractions. This means that we must transform, in a linear way, a space of dimension $k + n$ into a space of dimension n. This is done by means of a *linear isomorphism* $\Phi : \mathcal{X}^2 \to \mathcal{X}$. The existence of these isomorphisms requires the adoption of infinitary operator algebras, like C^*-algebras. To give an example, if we consider an operator algebra acting over the Hilbert space $\ell^2(\mathbb{N})$, we can take

$$\Phi\left(\sum_n (\alpha_n e_n \oplus \beta_n e_n)\right) = \sum_{2n} \alpha_{2n} e_{2n} + \sum_{2n+1} \beta_{2n+1} e_{2n+1} \tag{6.28}$$

which follows the coding of pairs of integers into integers known as "the Hilbert Hotel". Concretely, the Hilbert Hotel is what is required in order to double a wire τ into two distinct isomorphic copies τ', τ'': here is a typical feature of *GoI*, namely, the exposition of the procedures implicit in the manipulation of formal systems (introducing superscripts may indeed require to "make space on the blackboard").

The subjectivity of notations is thus eventually recovered: the Hilbert Hotel by no means constitutes a canonical choice for the isomorphism Φ, and different choices are not warranted to agree, i.e. to interact without producing interferences; Φ can be written as $\Phi = P + Q$ with the property that for every $u, v \in \mathcal{X}$,

$$\begin{aligned} P^* u P Q^* v Q &= Q^* v Q P^* u P \\ P^* Q &= Q^* P = 0 \end{aligned} \tag{6.29}$$

it can thus be used to produce commutations between arbitrary operators, by sending them into disjoint copies of the same universe. Remark that, when interpreting logic, this doubling of the universe has the effect to enlarge the dimension of the occurrence space (without violating linearity). On the other hand, any P and Q satisfying (6.29) will do the job, so that we can imagine infinite many different possibilities to implement an apparently simple task. By the way, the nilpotency order of computations (and thus the complexity involved in normalization proofs), as I'm going to make a bit more clear, will be highly sensible to the choice of P and Q, since the latter discipline the growth of the occurrence space during computation.

We can represent, for instance, the numeral $\underline{2}$, associated to the following proof (where $A_{f(2,4)}, A_{g(3,5)}$ denote fresh locations, respectively, depending on $2, 4$ and $3, 5$),

$$\begin{array}{cl} A_1 \vdash A_2 \quad A_3 \vdash A_4 \quad A_5 \vdash A_6 & \\ \hline\hline A_2 \Rightarrow A_3, A_4 \Rightarrow A_5, A_1 \vdash A_6 & (C) \\ \hline A_{f(2,4)} \Rightarrow A_{g(3,5)}, A_1 \vdash A_6 & \\ \hline A_{f(2,4)} \Rightarrow A_{g(3,5)} \vdash A_1 \Rightarrow A_6 & \end{array} \tag{6.30}$$

as follows: from the 6 × 6 wire,[11] representing the axioms,

$$\begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \tag{6.31}$$

we pass, through two linear isomorphisms $P : |A_2| \oplus |A_4| \to |A_{f(2,4)}|$ and $Q : |A_3| \oplus |A_5| \to |A_{g(3,5)}|$ satisfying (6.29), to the 4 × 4 contracted wire

$$\phi(2) = \begin{pmatrix} 0 & PQ^* & P^* & 0 \\ QP^* & 0 & 0 & Q^* \\ P & 0 & 0 & 0 \\ 0 & Q & 0 & 0 \end{pmatrix} \tag{6.32}$$

We can verify now that $[\phi(2)]\sigma$ still verifies nilpotency, but its order is now bounded by 4, meaning that the occurrence space of σ has been doubled by P and Q during interaction (or, equivalently, that σ has been doubled in $P\sigma P^* + Q\sigma Q^*$, that is, in what we would naturally write σ', σ''): in order to use $\phi(2)$, one has "made space on the blackboard" to create new copies.

On the other hand, suppose one makes the two distinct copies of integer representations interact, i.e. to represent the computation $(\underline{m})\underline{n}$, which in λ-calculus leads to an exponential growth:

$$\begin{aligned}
(\underline{n})\underline{m} \quad &\equiv \quad (\lambda f.\lambda x.(f)^n x)\lambda g.\lambda y.(g)^m y \\
\rightsquigarrow_\beta \quad & \lambda x.(\lambda g.\lambda y.(g)^m y)^{n-k}\lambda y.(x)^{m^k} y \\
= \quad & \lambda x.\, \underbrace{(\lambda g.\lambda y.\, \underbrace{(g)\dots(g)}_{m \text{ times}}\, y)\dots(\lambda g.\lambda y.\, \underbrace{(g)\dots(g)}_{m \text{ times}}\, y)}_{n-k \text{ times}}\, \lambda x.\, \underbrace{(x)\dots(x)}_{m^k \text{ times}}\, y \\
\rightsquigarrow_\beta \quad & \lambda x.\lambda y.\, \underbrace{(x)\dots(x)}_{m^n \text{ times}}\, y \quad \equiv \quad \underline{m^n}
\end{aligned} \tag{6.33}$$

From the viewpoint of formal systems, this poses strictly no problem (since variables and terms can be copied as many times as desired), but, in *GoI*, where every occurrence must receive a distinct location, one needs isomorphisms

[11] To be precise, the wire corresponds to a linear operator u whose *support* is a projection π of dimension 6, i.e. such that $u = \pi u \pi$, and which can thus be thought of as a 6 × 6 matrix. In the following, I'll freely use matrices to represent linear operators of finite support.

whose composition (inducing a group structure) provides enough space to implement the computation (6.33). In this manner, time and space requirements are translated into the algebraic properties of the group of isomorphisms chosen (see Girard 2007 for a discussion).

Now, we can well imagine a tortoise accepting *modus ponens* and demanding for more instructions for its application: since lacking the relevant – though logically arbitrary – isomorphisms, she simply does not know how to write something which would conform to the rule, and so she asks for further explanations. For instance, in (6.33) she could well question the convergence of reductions, since the possibility to write (6.33) undermines the validity of what (6.33) expresses: in order to show convergence of a notation for exponentials, one has to adopt an explicitly exponential notation (look at the underbraces)!

By inflating the argument, as relying on higher forms of the *modus ponens* schema (or some schema defining exponential functions), still the tortoise would not know how to accept the conclusion, since she would find herself in a completely isomorphic situation: the instructions she needs are inexorably left implicit in the description of logical rules; in order to investigate what it is to follow the rules of logic, then, one has to take into consideration the concrete manipulation of symbols in time and space.

To conclude, I add that, from a purely mathematical perspective, the purpose to characterize in purely algebraico-geometrical terms the subjectivity of formal systems and to describe logically their interactions has already led to some considerable results (concerning the implicit characterization of complexity classes like *ELEMENTARY*, *P* or *NL*, see for instance Girard (2012), Girard (2011a), and Aubert and Seiller (2012)) and constitutes one of the most promising technical aspects of research on *transcendental syntax*.

References

Abrusci, Vito Michele and Paul Ruet. 2000. Non-commutative logic I: the multiplicative fragment. *Annals of Pure and Applied Logic* 101(1): 29–64.

Aubert, Clément and Thomas Seiller. 2012. Characterizing co-NL by a group action. arxiv preprint abs/1209.3422.

Carroll, Lewis. 1895. What the Tortoise said to Achilles. *Mind* 104(416): 691–693.

Carnap, Rudolf. 1950. Empiricism, semantics and ontology. *Revue Internationale de Philosophie* 4: 20–40.

Cozzo, Cesare . 2004. Rule-following and the objectivity of proof. In *Wittgenstein today*, ed. A. Coliva and E. Picardi. Padova: Il poligrafo.

Dummett, Michael. 1973. *Frege's philosophy of language*. Cambridge: Harvard University Press.

Dummett, Michael. 1978. Wittgenstein's philosophy of mathematics (1959). In *Truth and other enigmas*. Cambridge: Harvard University Press.

Dummett, Michael. 1991. *The logical basis of metaphysics*. New York: Columbia University Press.

Engel, Pascal. 2005. Dummett, Achilles and the tortoise. HAL-SHS preprint, ijn_00000571.

Girard, Jean-Yves. 1987a. Linear logic. Theoretical Computer Science. 50: 1–102.

Girard, Jean-Yves. 1987b. Multiplicatives. In *Logic and computer science: new trends and applications*, ed. G. Lolli, 11–34. Rosenberg & Sellier.

Girard, Jean-Yves. 1989. Geometry of interaction 1: Interpretation of system F. *Studies in Logic and the Foundations of Mathematics* 127: 221–260.

Girard, Jean-Yves. 2001. Locus solum: from the rules of logic to the logic of rules. *Mathematical Structures in Computer Science* 11(3): 301–506.

Girard, Jean-Yves. 2007. *Le point aveugle Tome 2, Cours de logique, Vers l'imperfection*. Hermann: Vision des sciences.

Girard, Jean-Yves. 2011a. Geometry of interaction V: logic in the hyperfinite factor. *Theoretical Computer Science* 412(20): 1860–1883.

Girard, Jean-Yves. 2011b. La syntaxe transcendantale, manifeste. http://iml.univ-mrs.fr/~girard/syntran.pdf.

Girard, Jean-Yves. 2012. Normativity in logic. In *Epistemology versus ontology – essays on the philosophy and foundations of mathematics in honour of Per Martin-Löf*, 243–263. Dordrecht: Springer.

Joinet, Jean-Baptiste. 2009. Ouvrir la logique au mond. In *Ouvrir la logique au mond: Philosophie et Mathématique de l'interaction*. Hermann et CCI-Cerisy.

Joinet, Jean-Baptiste. 2011. Logique et métaphysique. In *O que è metafisica?* Editora da Universidade Federal do Rio Grande do Norte.

Prawitz, Dag. 1973. Towards a foundation of a general proof theory. In *Proceedings of the fourth international congress for logic, methodoly and philosophy of science*, Bucharest, 1971, vol. 74, 225–250.

Quine, Willard Van Orman. 1984. Truth by convention (1936). In *Philosophy of matemathics: selected readings*, ed. H. Putnam, P. Benacerraf. Cambridge: Cambridge University Press.

Searle, John. 1969. *Speach acts: an essay in the philosophy of language*. New York: Cambridge University Press.

Wittgenstein, Ludwig. 1969. *On certainty*, ed. by G.E.M, Anscombe and G.H von Wright. Basil Blackwell.

Wittgenstein, Ludwig. 2001. *Tractatus logico-philosophicus (1921)*. Routledge.

Wittgenstein, Ludwig. 2009. *Philosophical investigations (1953)*. Wiley-Blackwell. German text, with an English translation by G.E.M. Anscombe, P.M.S. Hacker and Joachim Schulte.