



# Gödel's Absolute Proofs and Girard's Ludics Mutual Insights

Gabriella Crocco, Myriam Quatrini

## ► To cite this version:

Gabriella Crocco, Myriam Quatrini. Gödel's Absolute Proofs and Girard's Ludics Mutual Insights. Antonio Piccolomini d'Aragona. Perspectives on Deduction, Springer, inPress, Synthese Library. hal-04003505v2

**HAL Id: hal-04003505**

**<https://hal.science/hal-04003505v2>**

Submitted on 5 Apr 2023

**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.

# Gödel's Absolute Proofs and Girard's Ludics

## Mutual Insights

Gabriella Crocco\* & Myriam Quatrini†

\* Université d'Aix Marseille-CNRS, Centre Gilles Gaston Granger, UMR 7304

† Université d'Aix Marseille-CNRS, Institut de Mathématique de Marseille, UMR 7373

### Abstract

Is it possible to characterize the notion of proof in terms of acts, without focusing on a specific domain of application and a specific linguistic formalization of it? This is the question that this paper addresses through a comparative analysis between two logicians who reflected on this issue: Kurt Gödel and Jean-Yves Girard. A comparative analysis of their respective theoretical frames, their respective results, the similarities and the differences between their methodological assumptions is proposed. More specifically, the aim of the paper is to compare Gödel's notion of absolute proof and Girard's notion of Ludic from their common core: a conception of proof in terms of acts.

## 1 Introduction

This article proposes a comparative analysis between the Gödelian concept of absolute proof and Girard's Ludics. Such a comparison links topics that are not usually discussed together and it involves complex philosophical problems and sophisticated logical tecnics. We hope that we have succeeded in symplifying their presentation without trivializing them. The aim of this analysis is twofold. On the one hand, we would like to give substance to Gödel's scattered indications on the concept of absolute proof through its possible implementation by Ludics, a theory of logic born in the context of theoretical computer science and, more specifically, from the research that gave rise to linear logic. On the other hand, we aim to give a fully epistemological and philosophical reading of some of Girard's results, which are often presented by Girard himself from a technical and methodological perspective, guided essentially by the proofs-as-program paradigm which arose in the 1970's from reflections on the Curry-Howard isomorphism.

This introduction is essentially devoted to explaining the guidelines of our analysis and the motivations that led us to undertake a comparative analysis between two fields of research which at first glance do not appear to be connected. In order to allow the reader to better appreciate our purpose, it seemed to us necessary to briefly introduce the two key notions presented in our title, since they are not widely recognized outside their small communities respectively of Gödel and linear logic scholars. A more thorough discussion will follow in the rest of the article but this first explanation will give us the opportunity to present the structure of our analysis in its main aspects before announcing the plan.

For the time being, let's say that we are concerned here by proofs that "can be treated in a completely absolute way"<sup>1</sup> i.e. absolute logical proofs. By an absolute logical proof<sup>2</sup> Gödel means a proof which does

---

<sup>1</sup>[Gödel1946] in [CWII], page 153

<sup>2</sup>See the end of Gödel's 1946 *Remarks before the Princeton bicentennial conference on problems of mathematics* [CWII], p 152-53. In [Crocco2019], section 2, p 568–70, we qualified this logical sense of the term "absolute" as strong, differentiating it from the weaker,

not depend on a given formal language and on a given formal system expressible in this language, which is therefore stable with respect to any process of diagonalization and extension and which is characterized independently from any particular domain of application involving specific objects and concepts. Since according to Gödel, logic is, by definition, universally applicable<sup>3</sup>, an absolute proof would be a logical one, that is to say:

- a) it would use only logical concepts;
- b) it would be concerned essentially by the acts that these concepts prescribe in the construction of the proof, independently of their linguistic expression; and
- c) it would perform these acts independently of any specific field of application.

It seems to us that although this notion of absolute proof occupies a central place in Gödel's logical and epistemological reflection, it has not received the attention it deserves<sup>4</sup>. Probably this is due to the fact that Gödel, as he himself admits, did not arrive at a satisfactory characterization of it despite it being essential for his own philosophical program which he himself qualifies as conceptual realism. As we will see in Section 1, according to Gödel the concept of absolute logical proof and the concept of concept are mutually definable. As the latter encounters intensional paradoxes, which have not found a satisfactory solution, the concept of proof in its most general sense cannot receive a clear analysis.

Our conviction is that this impasse, explicitly recognized by Gödel, should not prevent us from appreciating the relevance of what motivates Gödel's efforts. There are at least three reasons for Gödel to seek a satisfactory definition of the notion of a logical absolute proof and they seem to us to be of great importance for the current debate about the notion of informal reasoning and informal proofs<sup>5</sup>.

The first reason is related to the necessity of clarifying that logic cannot be reduced to formalised logic. The latter, as developed by the Hilbertian school, is a formidable tool for theoretical clarification, but neither from a historical nor from a theoretical point of view can it represent all the creative aspects of human logical activity. Finding a satisfactory definition of what a non-formalized proof is, would represent an essential step toward such a clarification.

The second reason is related to the necessity of highlighting the essential link between the reduction of logic to formal logic and the elimination of the heuristic dimension from it. The possibility to use logic not only as a method of justification (*ars demonstrandi*) but also as a method of discovery (*ars inveniendi*) was central to Leibniz's logical investigations, to which Gödel explicitly refers. Disentangling the notion of proof from the notion of formal proof (as mechanical procedure), would open new perspectives on such investigations, including heuristic methods, process of cognitive changes in, and conceptual recasts of our theories.

---

mathematical sense of "absolute" discussed by Gödel in connection to absolute provability and definability in mathematics and more specifically set theory. We will come back to this matter in section 1, below. On absolute mathematical proofs see Juliet Kennedy in sections 2-4 of [Kennedy2021].

<sup>3</sup>See [Wang1996], p. 267: "For Gödel, logic deals with formal, in the sense of universally applicable- concepts." and in the same book, Wang's transcriptions of his conversation with Gödel p.267-8; no 8.4.17 and no 8.4.18. See [Crocco2019]-section 2 for the connection between this interpretation of the term "formal" and the notion of strong absolute proof.

<sup>4</sup>See however in [Crocco2019] and [VanAtten2023] the sections concerning in particular the interpretation of [Wang1996] page 188-189, remarks 6.1.13-16.

<sup>5</sup>See on this subject [Rav1999, Giaquinto2007, Mancosu2008, Leitgeb2009, Marfori2010, Montesi-Piccolomini d' Aragona2022]

The third reason is related to the necessity of clearly distinguishing, if possible, a mechanical calculus (i.e. a meaningless sequence of signs manipulated by combinatorial rules) from a creative human proof considered as “a structure of thought” ([Gödel58] and [Gödel72] p 273). Many of Gödel’s writings highlight the central role of this question in his philosophical reflection on the relation between mind and matter. Its pertinence in connection to the present debate on the nature and the power of artificial intelligence seems to us undeniable.

As we will explain in Section 1, because of the great significance of this notion of absolute proof, the fragmentary nature of Gödel’s references to it is certainly disappointing. Our work starts from this assessment, and tries to go beyond it by means of a simple idea: it might be possible to give a clearer exemplification of what an absolute proof could be in a context quite different from Gödel’s philosophical frame. For this reason, we have turned to Jean-Yves Girard’s Ludics and Section 3 of the paper will be devoted to its presentation. For the moment, we can say that Ludics is a theory of logic, which pursues the project of linear logic in the analysis of proofs, in order to give the best account of what Girard calls their computational content. It puts at the heart of this analysis the notion of interaction, thus placing itself in a dialectical (or dialogical) perspective when searching for an agreement in a demonstrative process. In doing so, precisely because it aims to best represent the dynamics of computational processes, it exceeds the notion of mechanical computation and formal demonstration in a direction that seems to us to show clear resonances with the *desiderata* that Gödel states for the notion of absolute proof: independence of language, representation of proofs in terms of acts, independence of any domain of application, stability. These resonances invite us to deepen the comparative analysis and to propose a non-obvious transposition between two diametrically opposed epistemological viewpoints. Gödel’s starting point are proofs as structures of thought, where the meaning of the concepts involved in his proof is an essential element for establishing it; Girard’s starting point are computational processes and their interactions, i.e., their possible compositions in new processes. On Gödel’s framework we start with concepts: abstract entities, timeless, independent of our constructions and conventions (linguistic or not); on Girard’s framework we start from concrete, singular, localized events. And yet, we make the bet that the comparison can be investigated fruitfully provided we keep in mind the differences.

What led us to such a conjectural comparison ? Logical and philosophical reasons.

Firstly, from a logical point of view, Girard’s work springs, at least partially [LGT90, Girard1972], from a deep reflection on Gödel’s system presented in the so called *Dialectica* paper [Gödel58], which is an attempt to overcome formalism in order to give an intuitive and constructive foundation of arithmetics. There is, in some way, a convergence between their works, which is revealed by the logical systems they proposed in order to proof or ensure the consistency of respectively arithmetic (Gödel’s system T) and analysis (Girard’s system F). As we will explain in Section 2, both recognize the necessity to go beyond formal systems, having recourse to abstract notions and methods for ensuring or proving the coherence of mathematics.

Secondly, from a philosophical point of view, we think that one could reformulate in Girardian terms the answer that Gödel makes to Leon Rappaport about the philosophical meaning of his incompleteness theorems. Gödel states:

“My theorems only show that the mechanization of mathematics, i.e., the elimination of the mind and of abstract entities, is impossible, if one wants to have a satisfactory foundation and system of mathematics.” (August 1962, [CWV] (p. 176))

Within Girard’s analytical framework, always claimed by him as non-foundationalist, the elimination of the subject is impossible since the latter is the provider of fundamental intuitions (of the “synthetic *a priori*” says

Girard [Girard2021]) that found mathematical proofs and cannot be justified by a formal proof of coherence. Nevertheless, for Girard it is not a singular “mind” but pluralities of minds that are required to account for our demonstrative practice. In other words, ludics seeks the notion of absolute proof by the characterization of the conditions of possibility of the agreement between subjects in a dialogic argumentative process. There are attempts to give a philosophical interpretation of Girard’s logical work [Joinet2007],[Joinet2023]. Our contribution to this literature, in the conclusion of this article, emphasizes the anti-formalist perspective of both Gödel and Girard, which results in a fundamental agreement on the role they ascribe to the knowing subject and to language in the construction of human knowledge. Although this fundamental agreement that seems to us to be very important, there is a total disagreement between them on the philosophical framework they adopt. Our task here is to clarify the relationship between their anti-formalist conceptions of logical proofs, the centrality they ascribe to the subject in the construction of knowledge, and the opposing ontological perspectives they use to orient their scientific work. As we will discuss in Section 4 of our paper, there is an ontological analogy that has guided our work and that seems to make clear the relationship between their two projects. It can be formulated as follows: Gödel’s notion of absolute logical proof is to Girard’s ludics what conceptual realism is to the nominalism of events. Their respective ontological frameworks are thus completely different, but the notion of logical proof they are seeking for shares important aspects. This common core, revealed by the way they characterize a logical proof, offers a new perspective on their works. Some important aspects of the notion of absolute proof, as Gödel conceives them, are realized in a coherent and effective logical framework in Girard’s ludics. On the contrary, some technicalities of ludics and its *raison d’être* are made clear by their relation to Gödel’s notion of absolute proof. Finally, a non-formalist notion of logical proof is clarified by the comparative analysis of the respective scientific works of these two logicians.

As we already said there are complex philosophical problems and difficult technical results from both Gödel and Girard. This paper attempts to connect them trying to gain mutual insights from their different perspectives on a very essential epistemological notion such as that of logical proof.

The plan of the paper is as follows.

In section 1 we present the different logical and epistemological notions to which Gödel applies the adjective absolute and insist in particular on Gödel’s analysis of absolute computability and absolute provability. The notions of absolute knowledge, absolute definability, absolute provability, absolute computability will be discussed in respect to Gödel’s analysis of concept theory.

Section 2 is devoted to the new theoretical frame that emerges after Gödel’s work on intuitionism and constructivism (as presented in the *Dialectica* paper) and introduces the proofs-as-program paradigm and Girard’s linear logic as motivated by it. It will allow us to introduce some of the key concepts that will be used in Section 3 and also to illustrate the change of perspective in the discussion on the notion of proof which occurred between the first and the second part of the 20th century.

Section 3 presents Ludics and its characteristics most likely to show its convergence with the Gödelian notion of absolute proof.

In Section 4, the conclusion, we come back to the ontological analogy which guided our comparative work and which seems to us to clarify the relationship between the two projects. It has been formulated as follows: Gödel’s notion of absolute proof is to Girard’s Ludics as the realism of concepts is to the nominalism of events.

## 2 Gödel and the absolutes

The adjective “absolute” is used by Gödel from the time of his first writings (published articles, conferences) in the 1930’s<sup>6</sup> up to the 1970’s. Its use is, in most cases, intended to qualify epistemological notions, i.e., notions concerning our knowledge, the theories we build, the properties of the concepts we use. In the following sections, we will first discuss (section 1.1) the notion of absolute knowledge as presented in the 1970’s by Gödel. This will allow us to introduce some aspects of Gödel’s ontology, his conceptual realism, as well as some aspects of his theory of knowledge in order to underline the differences with the conceptual framework of Girard. We will then proceed to the analysis of the logical concepts of absolute computability (section 1.2) and absolute demonstrability (section 1.3), as discussed by Gödel from 1946 onwards together with absolute definability. According to Gödel, definability, computability and demonstrability are logical and epistemological concepts whose clarification is essential for the understanding of the nature and the limits of our knowledge. Definability is related to the human ability of fixing clearly the meaning of a notion in such a way that it can be intersubjectively understandable (“comprehensible by the human mind”). Computability is related to our ability to carry on mechanical and combinatorial operations on meaningless signs and therefore it can be identified with a mechanical procedure that a machine can perform. The ability to perform formal deductions in a formalized axiomatic system obviously belongs to computability. Demonstrability is related to our ability to realize mental constructions, epistemically compelling, and based on the insights we have into the concept involved. Gödel investigates what distinguishes these two last capabilities, because he is convinced that the demonstrative ability of human beings exceeds the capacity of a machine to perform computations. In the sequel our aim is to sketch the way he follows to characterize the respective absolute definitions of computability and demonstrability (disregarding for our purpose the concept of definability) and explain the problems he faced and the paths he tried to overtake them.

### 2.1 Absolute knowledge and perception of concepts

There is in Gödel’s last conversations with Wang<sup>7</sup> a strongly philosophical perspective in the discussion of the concept of absolute knowledge<sup>8</sup>. Gödel places the latter in the context of a rationalist theory of knowledge and an ontology that he himself calls conceptual realism. Concepts have an objective and a subjective

---

<sup>6</sup>The first occurrence of the adjective “absolute” appears in a lecture most probably held in 1931 and published in the third volume of the Collected Works [CWIII]. Here Gödel introduces the notion of “absolute undecidability” which will be widely discussed in the *Vortrag Konzept* of 1951 [CWII] (p 304-323). After presenting his first incompleteness theorem by outlining the main steps of its proof, and before briefly introducing the second one, Gödel states:

“The procedure just sketched furnishes, for every system that satisfies the forementioned assumptions, an arithmetical sentence that is undecidable in that system. **That sentence is, however, not at all absolutely undecidable, rather one can always pass to higher systems in which the sentence in question is decidable.** (Some other sentences, of course, nevertheless remain undecidable). In particular, for example, it turns out that analysis is a system higher in this sense than number theory, and the axiom system of set theory is higher still than analysis. It therefore follows, e.g., that there are number-theoretic problems that cannot be solved with number theoretic, but only with analytic or, respectively set-theoretic methods.” [CWIII], p 35, (the stress is ours).

The second occurrence of the adjective ‘absolute’ can be found in the article: *On the Length of proofs* [Gödel1936a], concerning the notion of a computable numerical function in a formal system  $S_1$  for arithmetic. Considering the possible transfinite hierarchy of stronger higher order systems  $S_i$ , each containing  $S_1$  and able to prove formulas undecidable in the weaker systems, Gödel stresses the fact that passing from a formal system to the next in the hierarchy described there allows one to shorten to a large extent many of the proofs already available. Then Gödel adds: “It can be shown that a function computable in one of the systems  $S_i$ , or even in a system of transfinite order, is computable already in  $S_1$ . Thus, the notion ‘computable’ is in a certain sense ‘absolute’, while almost all metamathematical notions otherwise known (for example provable, definable, and so on) quite essentially depend upon the system adopted.” [CWI], p. 399. The term “absolute” here means stable with respect to successive extensions of the formal systems containing  $S_1$ , to which the function in question belongs.

<sup>7</sup>See [Wang1974] and [Wang1996].

<sup>8</sup>See [Wang1996], 9.2.30-37, pp 302-3

existence: they are respectively structural features of reality (actual or not) and structural features of our mind. Objective concepts are considered by Gödel as acts or principles of activities, for which we cannot foresee all realizations, and which do not presuppose any prior domain of application. As abstract universal entities, they are causes of the objects which instantiate them (i.e., which belong to their extensions). As reality is structured by objects and concepts (the latter being properties and relations), we understand reality insofar as we perceive or have an intuition<sup>9</sup> of its objects and concepts and have some evidences on the properties they have. Subjective concepts are the result in our minds of this perception. They are therefore principles of activities of our mind that we realize as acts at the occasion of our experience. Empirical subjective concepts are obtained by comparing and relating perceptions of objects and of facts of our experience. More abstract concepts (for example those of mathematics) are obtained by using, at the occasion of our experience, an inborn *a priori* intuition such as the one concerning unities and pluralities. The more we enhance our intuition, the more we progress toward an absolute understanding of our concepts which will have the character of evidence.

These three notions of intuition, evidence and absolute knowledge must not be taken in a naive sense. They must be inserted in a dynamic conception of knowledge, which Gödel tries to elaborate from a critical rethinking of the theories of Leibniz and Kant and more generally of classical rationalism.

Intuition is a difficult notion which has been explored in different ways throughout the history of philosophy. Gödel asserts that intuition is not infallible and that it is analogous to a vision of distant objects, all encompassing, but ignoring the details<sup>10</sup>. Intuition always involves anticipations concerning aspects of abstract concepts that we have not yet explored. In this sense it does not give us an immediate, complete and necessary vision of what we are aiming at, but a kind of indication of the direction in which we should proceed. Therefore, the evidence that this fallible intuition offers can only be acquired by degree. There are degrees of evidence, and we can learn strategies and methods to enhance it<sup>11</sup>.

Perception of concepts is much more indirect and mediated than perception of objects, as concepts are primarily perceived as properties of, or relations between, objects. Gödel insisted in different occasions on the fact that it is exactly because of the mediated nature of their perception, that concepts were so often considered as epistemic or linguistic entities instead of elements of reality. From his point of view, the admission of objective concepts as elements of reality and their difference from subjective one allows us to recognize, at the same time, the fallible nature of our knowledge and the true nature of the progress that human mind can accomplish through time<sup>12</sup>.

Our understanding of reality (and therefore our knowledge of it) can be described in terms of the abstractness of the intuition required to understand its concepts. The following list expresses also a hierarchy in generality (from the lower to the higher degree).

- Social and natural sciences (sociology history, biology, chemistry, physics...) are specialized in the sense that they are the result of isolating in a specific way aspects of reality in order to understand what kind of objects can we consider and what kind of concepts can we use to study their properties and relations. These disciplines begin with concepts which are abstracted from experience and use them.

---

<sup>9</sup>We will use in the sequel the terms "intuition" and "perception" of abstract entities as synonymous, without assuming that there is no difference in the use Gödel does of them. Intuition is historically related to the idea of a kind intellectual vision, whereas perception makes reference to a much broader possibility to acquire abstract knowledge. Both stresses, in Gödel, the idea of a human ability to go further what language allow us to express.

<sup>10</sup>See [Wang1974], p85, and [Wang1996], 5.3.29.

<sup>11</sup>Gödel tells Wang: 9.2.35 "We have no absolute knowledge of anything. There are degrees of evidence. The clearness with which we perceive something is overestimated [...]" [Wang1996], p302.

<sup>12</sup>[Wang1996] p. 264 remarks 8.4.19-8.4.20 , p. 390 and p. 302 remarks 9.3.20-9.3.35.

From these specific sciences we have to distinguish two formal sciences, that is mathematics and logic. “Formal” here means sciences whose concepts can be applied to any domain of knowledge.

- Mathematics is formal because it applies to any abstract object, that is for Gödel, to any extension of concept<sup>13</sup>. Mathematics uses concepts but also analyzes them in as much it is concerned by relations and properties between concepts considered extensionally. Actually, mathematics is not concerned by all concepts but only by those concepts whose extension can be treated as an object (i.e., sets<sup>14</sup>), disregarding concepts whose extensions are mere pluralities (i.e., class). In order to form mathematical (subjective) concepts we need the inborn fundamental intuition of pluralities and unities. As mathematics applies to sets, i.e. plurality of things which can form unities, it applies to reality and therefore it is an essential tool for our knowledge of the world<sup>15</sup>.
- Logic analyzes concepts in intensions, the way they are composed by primitive concepts. It goes further in generality than mathematics as it is concerned by concepts not only in as much they are reducible extensionally to sets, these “limiting case of spatio-temporal objects”, but also by concepts in themselves, independently of their extensions, being them classes or sets. Logic is therefore formal because it is universally applicable to any concepts and objects whatsoever. This implies that amongst concepts Gödel distinguishes primitive logical concepts from which all other human (subjective) concepts can be derived. He never gave a complete list of such primitive concepts, but he mentions amongst them the concept of concept, the concept of object, the concept of application, the concept of negation, of conjunction, of existence and of generality.
- Finally, philosophy analyzes concepts in their outmost generality and in relation to the way they can relate and explain all the specific aspects in which reality appears to us. Its fundamental concept is the concept of cause, which involves space, time, the concept of concept, the concept of object and so on. All primitive concepts of each domain of knowledge should be explained and organized in a systematic way by philosophy whose task is to explain the meaning of the world, the reason why it is as it is.

In all these domains we try to understand concepts and objects, try to organize them in a structural way, explaining their relationship and finding the primitive concepts from which the others can be obtained. In other words, we try to understand all the compound concepts we form in terms of the primitive concepts that we perceive, in a fallible way. So, we can reach an absolute knowledge, in as much we arrive to an absolute definition (implicit or explicit) of our concepts. Here, the adjective “absolute” should be understood in the sense of invariance.

Firstly, it means invariance from the theoretical frames we can invent for the concepts we are analyzing. Consequently, it means invariance from the linguistic expression we can give, in a formalized language, to these theoretical frames.

Secondly, it means invariance from the domain of application of our concepts, that can concern objects of whatever kinds but also concepts themselves.

Can we really arrive to such absolute definitions of concepts, and if so in which domain and to which extent ?

---

<sup>13</sup>We will not discuss here this important aspect of Gödel thought which opposes sharply objects and concepts, but see [Wang1996]p. 296

<sup>14</sup>For Gödel mathematics is reducible to sets theory, which in turn, can be interpreted as the discipline which is concerned only by extensions of concepts which are sets. Gödel conjectures that all sets are extensions of concepts, although not all extensions of concepts are sets.

<sup>15</sup>See [Wang1996] p. 300: 9.2.14 “The essence of mathematics is that it consists of generalizations.” and 9.2.15 “In science we generalize. Mathematics describes possibilities of which only few have been realized.”



We know that we have no clear intuition of many primitive concepts of our knowledge (in particular of the philosophical concept of cause and of the logical concept of application). It follows, consequently, that absolute knowledge could only be a regulating ideal for human reason, what it tends towards, what guides it in the construction of its own (fallible) theories but nothing more. Gödel certainly express this idea at least in the conversations with Wang.

Nevertheless, he was convinced that mathematics offers an example of a kind of knowledge where we have an absolute understanding of some of the concept we use. According to Gödel, mathematics is the most striking illustration of a domain where intuition has always been the indispensable ally of logical rigor and formalization. It is only through the dialectic between the intuitive and the formal that mathematics has developed so successfully. However, historical analysis of this success invites us to be cautious. Gödel always asserted that only certain concepts, certain conquests of our knowledge, undeniably deserve the qualification of absolute knowledge since they are stable with respect to any conceptual and theoretical development and independent of the formal systems in which theories including these concepts can be formulated. Gödel gives different examples of what he considers absolute concepts and theories elaborated by our science: the analysis of the concept of natural number by Dedekind and Peano, the theory of Diophantine equations of first degree and second degree with two unknowns, the concept of velocity expressed by  $ds/dt$ , the concept of “size” as opposed to shape, as explained by Peano’s notion of measurement, the concept of continuity, once we have removed the ambiguity between smooth curve and continuous movement<sup>16</sup>. In all these examples, the concepts perceived with rigor after a long work of elaboration appear to us, says Gödel, as “unquestionably unique” and endowed with obvious properties that axioms can make explicit. “Obviousness” here means that we are epistemically constrained to recognize these properties once we have perceived with clarity the concepts in question and therefore that these properties appear to us as necessary, that is to say independent of our conventions and our constructions.

There are many passages in Gödel’s published and unpublished writings in which he comments on the similarity between the abstract intuition necessary for the practice of mathematics and sensible perception<sup>17</sup>. In his view, there is a strict analogy between perceiving a sensible object from different angles and perceiving concepts through their different instantiations, formal or not, which can be shown to be equivalent. This idea, which reminds Husserl’s “eidetic variation”, has a very strong methodological impact in Gödel’s scientific work. The best method for acquiring a stable (absolute) knowledge consists, firstly, in trying to see a notion from different perspectives, from different points of view, making it precise in a rigorous way from all these specific perspectives and, secondly, trying to characterize what is invariant in the multiplicity of points of view so experienced.

It can be certainly possible that what appears to us in a certain place and time to have the characteristic of absoluteness reveals to be the result of a “confused perception”, but Gödel’s rationalistic optimism avoids the conclusion that some aspects of reality will, forever, be obscure and non-transparent for our mind. Progress in time is the horizon in which human knowledge is inscribed, although humanities can arrive “by a kind of miracle” ([Gödel1946] in [CWII], p 150) to some absolute definitions of important concepts.

<sup>16</sup>[Wang1974], p. 325, [Wang1974], p. 85, [Wang1996] 9.2.36, 9.2.37.

<sup>17</sup>Gödel writes: “The similarity between mathematical intuition and physical sense is very striking. It is arbitrary to consider *This is red* an immediate datum, but not so to consider the proposition expressing modus ponens or complete induction (or perhaps some simpler propositions from which the latter follows). For the difference, as far as it is relevant here, consists solely in the fact that in the first case a relationship between a concept and a particular object is perceived, while in the second case it is a relation between concepts.” \*1953/9/ version V, p. CW III, p359.

## 2.2 Absolute computability

Having sketched out Gödel's general philosophical framework for absolute knowledge, we can now turn to the application of this adjective to a mathematical and logical concept, computability, which is deeply related to demonstrability. In his paper *Remarks before the Princeton bicentennial conference on problems in mathematics*, written in 1946 [CWII], pp 150-153, Gödel states that Turing's concept of computability has for the first time provided an absolute definition of an epistemologically interesting notion (computability) and that it would be desirable to look for such absolute definitions for other epistemologically interesting notions such as demonstrability and definability.

The starting point is therefore the notion of Turing's computability, intended, first of all, as mathematical computability: the ability to solve a problem by a mathematical effective calculus on mathematical objects, but also and more generally as the ability to solve a problem whatsoever by a mechanical procedure (logical computability). As we explained before, this latter interpretation makes the notion of computability a special case of the notion of demonstrability and motivated Gödel to start the analysis of absolute demonstrability with this special case. It will also allow us to better understand what is at stake in the shift from the mathematical absolute concept of mathematical procedure, to the logical absolute one, i.e., from an analysis centered on objects to an analysis in terms of concepts (acts).

In the beginning of the 1946 paper, considering mathematical computability, Gödel means by absolute computability "one not depending on the formalism chosen", in other words, stable with respect to any possible conceptual framework in which it can be transposed<sup>18</sup>. Gödel emphasizes on many other occasions the relevance he attaches to the definition of computability given by Turing<sup>19</sup>. There is evidence of the fact that Turing's work offers, from Gödel's point of view, a totally new analysis of the concept of mechanical procedure because it brought out clearly that the intuitive notion of mechanical procedure does not require that it should always terminate or succeed. "A something unsuccessful procedure, if sharply defined, still is a procedure" (Wang 1974, p. 84).

Of course, other demonstrably equivalent definitions had been proposed: by Gödel himself through the concept of recursive function in his own work of 1931 and by Church, through the lambda-calculus. However, Gödel strongly emphasizes the novelty of Turing's approach, the only one that confers a true generality to his incompleteness results. This novelty lies essentially in the fact that Turing's analysis concentrates on the acts performed during the execution of a computation, completely neglecting both the objects on which one computes and the language in which the computation algorithm is expressed in order to be executed by a machine. Indeed, on the one hand, the notion of general recursiveness obtained by the definition of partial recursive functions is based on the mathematical concept of functions of integers on integers. It is therefore not at all independent of the object domain, which is presupposed. Starting with numbers and with functions on them, our intuition led us directly to the notion of total functions and therefore to ending computations. The general notion of computation comes only later, in a negative form, as a partial function. On the other

---

<sup>18</sup>In a note added in 1965 and referring to the expression "one not depending from the formalism chosen" Gödel says about arithmetical computability: "To be more precise: a function of integers is computable in any formal system containing arithmetic if and only if it is computable in arithmetic, where a function  $f$  is computable in  $S$  if there is in  $S$  a computable term representing  $f$ ." [CWII], p. 150.. According to Gödel \*1951, [CWIII]page 305, we interpret here the second occurrence of "arithmetic" in the sentence "if and only if it is computable in arithmetic" as arithmetic proper, that is, the body of those mathematical propositions which hold in an absolute sense, without any further hypothesis. There is no reason to interpret it as meaning "formal arithmetic", as it was explicitly the case in 1936a [CWI], pp 397-399. Forcing the latter interpretation leads J. Kennedy to understand absoluteness as confluence, i.e., "stability of a concept across conceptually distinct **formal** frameworks" (the emphasis is our) [Kennedy2021], p. 74. According to us, considering what Gödel says about absolute knowledge, the kind of stability at which Gödel is pointing is conceptual stability i.e. stability across distinct conceptual or informal axiomatic frameworks. Stability across distinct formal frameworks is a particular case of general conceptual stability.

<sup>19</sup>[CWI], p. 195, note of August 28, 1963 to Gödel 1931, [CWI], p. 370, Postscript of June 3, 1964 to Gödel 1934, section 3.1 in [Wang1974].

hand, the notion of lambda computability presupposes as a given the formal language of the lambda calculus and it cannot provide, according to Gödel, the guarantee intrinsically linked to a conceptual analysis carried out independently of the language<sup>20</sup>. Finally, what attests to Gödel's emphasis on Turing's conceptual analysis, as opposed to other definitions, is the well-known fact that Gödel did not accept Church's thesis until Turing's definition of a mechanical procedure gave it a foundation independent from language.

Once again, and this is the important point for Gödel, what allows Turing to arrive at his absolute definition is the fact of aiming at the acts of a human calculator, while totally neglecting the aspects that are accessory to the calculation, i.e., what he calculates and in which symbolism he writes his calculations and the algorithmic rules that he must apply. By ignoring these two aspects, he can exclusively focus his attention on the processual aspect of the computation. The latter can then be analyzed in terms of simple operations (reading/writing, moving to the left or to the right, applying a rule based on the observation of what he reads in a box) whose mechanical nature is unquestionable for him. These operations can then be chained together to accomplish arbitrarily complex calculations, whose mechanical nature derives directly from the nature of the simple operations exhaustively listed<sup>21</sup>. Once this analysis in terms of simple acts is accomplished, the problem of finding a formal language becomes relatively trivial. Then, the very fact of having arrived at this completely absolute definition of the concept of computation allows the properties of a mechanical process, "i.e., capable of being performed by a machine, or by a man in a mechanical (non-creative) way<sup>22</sup>", to emerge with clarity.

Synthesising Gödel analysis and following the suggestions of Wang<sup>23</sup> and of Judson Webb<sup>24</sup>, we can stress three properties of the acts involved in calculation that we think are worth mentioning before moving on to the notion of absolute demonstrability (of which the notion of computation is only a particular case).

The first of these properties is determinism: the choice of application of the rules of the computation is determined by the current state of the computation without being modified either by the interpretation of the context, or by the meaning of the signs or the rules. What to do at each step is prescribed by the operational rules (indicated by the transition function) which must be exhaustively listed and whose application is totally determined by the analysis of the signs.

The second property is finiteness: finiteness of the different types of states of the machine (or computer) when performing the computation; finiteness of the number of symbols that are written at a given moment; finiteness of the operations that can be performed at each step of the computation.

The third is locality: passage from one computational step to another and from one state of the machine to another is always locally determined. To proceed to the next step, the machine only needs to process a very small part of the information at a time. When it takes a decision, it only looks (step by step) at a finite and bounded part of the data and thanks to the transition function it proceeds in the computation from one step to another.

The conclusions we can draw from this analysis are crucial. These general properties emerge naturally from the analysis of the absolute concept of calculability, insofar as it allows us to abstract from any language and any domain of application. As they emerge, they indicate the direction in which one can look for the characterization of a proof that cannot be reduced to a calculus. A proof that a creative human being, but not a machine, would be able to carry out, should be non-deterministic, in the sense that the application of a rule should depend on the meaning of the concepts used, or non-local, if conducted in such a way as to

---

<sup>20</sup>See Wang1974, p. 91

<sup>21</sup>See in particular note 29 [Wang1974], pp. 101-102

<sup>22</sup>[Wang1974], p. 91

<sup>23</sup>see chapter II, section 3, of his 1974 book devoted to the concept of mechanical procedures, and discussed in depth with Gödel

<sup>24</sup>See introduction of Remark 3 of Gödel's 1972a, [CWII], pp 292- 304, and more specifically p 298.

account for global features of the demonstrative process, or non-finitistic. Gödel expresses similar ideas in his 1972 note, about Turing’s conclusions on the identity between machine and mind. We can reformulate Gödel’s point of view in his remark 3 as follows<sup>25</sup>. A finite human mind can go beyond the stepwise, local procedures of a machine, because it is capable of a sort of global *Gestalt*, and by abstract thought it can find global solutions which it could not arrive at by local procedures. To collect in an abstract notion an infinity of more concrete notions perfectly mastered constitutes a strategy which proves promising. Our human reason proceeds only in time and proceeding in time means proceeding essentially step by step. However, our capacity for reflection (of self-application of thought to our own thoughts) is constitutive of the plasticity of our brain. We can harvest and summarize (creatively) into one new non-mechanical process the obvious procedures in our possession.

### 2.3 Absolute demonstrability and its open problems

As we mentioned before, this absolute definition of the concept of computability should encourage us, says Gödel in 1946, to look for absolute definitions of other epistemologically interesting concepts like the concept of demonstrability and the concept of definability. In the light of what has been explained in section 1.1, the epistemological interest of these concepts appears clearly. An absolute definition of the concept of definability would give us an adequate formulation of the “comprehensibility by our mind” of any concept ([CWII], p. 152). And following Gödel’s comment in \*1951 in [CWIII], an absolute definition of demonstrability would give us an adequate formulation of the notion of any demonstration that the human mind can conceive in whatever domain. More specifically, in this latter case, if we come to a better understanding of the general concept of proof (i.e., the concept of logical absolute proof) we could prove that the whole of what we can prove (in whatever domain, including mathematics) is consistent. If this were the case, we would be able to prove our own coherence and thus recognize the correctness of our own conclusions, so that the difference between minds and machines would be clarified<sup>26</sup>.

In the same 1946 paper, symmetrically to what he has done for computability, Gödel analyzes the possibility of an absolute definition of provability in the domain of mathematics, before considering at the end the general case for logical absolute provability. At the beginning he states two fundamental remarks that clarify respectively independence from language and stability from any extension and variation of the domain of objects, within the context of mathematical absolute provability.

The first remark highlights a phenomenon of incompleteness which Gödel will talk about at length in the 1951 article. Unlike computability, no formalism can specify demonstrability, at least if it includes mathematical demonstrability. Indeed, the incompleteness theorems show that whatever the chosen formalism is:

“The contemplation of this very formalism gives rise to new axioms exactly as evident and justified as those with which you started and [...] this process of extension could be iterated into the transfinite” [Gödel1946] p. 151

So, if we aim at the notion of absolute demonstrability in mathematics, in no way can we reduce it to formal demonstrability in a system.

The second remark emphasizes the dialectic of the formal and the intuitive in mathematics. Indeed Gödel insists on the interest that these successive formal extensions can have in the research for the absolute concept of demonstrability. The analysis of what we do in order to pass from a formal system to a stronger one, by a possibly infinite iteration process, can allow us to perceive regularities. The analysis of these regularities would help us to perceive the non-mechanical process that would allow us to describe and bring together

---

<sup>25</sup>[CWII], p. 306

<sup>26</sup>Gödel tells Wang: 6.1.17 “When one speaks of mind one does not mean a machine (in any general sense) but a machine that recognizes itself as right” [Wang1996], p189.

what is conceptually essential in these successive extensions. This is what Gödel says for mathematical provability. What about logical provability?

It has been pointed out [Crocco2019], pp. 568-569, how, at the end of the same paper, after having suggested different ways to conceive such absolute notions of demonstrability and definability in the field of mathematics and set theory, Gödel evokes the possibility of giving a completely absolute definition of these notions, beyond the field of set theory as well as of any field. Actually, he states:

You may notice that, in both examples I gave [demonstrability and definability], the concepts arrived at or envisaged were not absolute in the strictest sense, but only with respect to a certain system of things, namely the sets as conceived in axiomatic set theory [...]. The question whether the two epistemological concepts considered, or any others, can be treated in a completely absolute way is of an entirely different nature. [CWII], 152-3.

In what sense is the definition of a *completely absolute notion of demonstrability*, according to Gödel, of an *entirely different nature* than what can be proposed for mathematics and set theory?

Firstly, how should we understand the expression “completely absolute”? We already mentioned that, for Gödel, logic can be characterized as what is universally applicable, without having any regard to the specificity of the things we are considering. He says explicitly that logic can be considered the science which is presupposed by all the others and “which contains the ideas and principles underlying all sciences” [CWII], p.119, and he repeats to Wang that logic is universally applicable to any system of things and to any domain. Therefore, a completely absolute concept of demonstrability is a logical absolute concept of demonstrability<sup>27</sup>.

Secondly, why does Gödel assert that the problems posed by such a completely absolute and therefore logical concept of demonstrability are of an entirely different nature? Why should the nature of the question of a logical absolute proof be totally different from the nature of the question of mathematical absolute provability?

The only clue for understanding such a statement are the scattered and rather obscure remarks contained in the conversations with Wang (especially 6.1.12-6.1.16). We will not go into the exegesis of these texts, but we can mention some aspects of the theses presented there.

Gödel asserts that the concept of absolute proof and the concept of concept are mutually definable. His reasoning seems to be built up as follows. To understand a proof implies understanding the logical relations between the concepts that are involved in it and to understand a concept means to understand the proofs in which it can be involved. The usual concepts are analyzable in terms of simple primitive concepts and the absolute proofs in terms of simple primitive acts, prescribed by the primitive concepts involved in them. Therefore, the concept of concept can be grasped if we understand what is a correct proof for the proposition containing it. In its turn, the concept of absolute proof will be understood when we are able to recognize through it what are the primitive concepts (the primitive acts) from which all the others can be specified<sup>28</sup>.

The concept of concept is afflicted by intensional paradoxes which have not yet found a satisfactory solution, unlike extensional paradoxes (of set theory) and semantic paradoxes (like the liar) which only involve truth in a language and can be solved by the Tarskian distinction between language and metalanguage<sup>29</sup>.

Therefore, because of our present inability to find a solution to these intensional paradoxes no analysis of the notion of absolute provability can be rigorously proposed. As Gödel says to Wang ([Wang1974], note 14 on page 328) the concept of proof “in its most general sense” (i.e., the concept of logical absolute proof) is not, in the present state of our knowledge, clear enough to support self-reference. Intensional paradoxes occur and we are unable to isolate the cases in which self-reference leads to contradiction. In the framework

<sup>27</sup>To stress this difference, as announced in the introduction, note 1, we decided in [Crocco2019], pp. 568-70 to qualify logical absolute provability as strong absolute provability. On the contrary mathematical absolute provability (or absolute provability in a specific domain) can be qualified as weak absolute demonstrability, because it just implies independence from formal language and from formal systems and therefore stability.

<sup>28</sup>See [VanAtten2023], note 71, for a slightly different explanation of this assertion by Gödel.

<sup>29</sup>See on the intensional paradoxes [Wang1996], pp 278-280, and in particular 8.6.25

of Gödel’s conceptual realism these intensional paradoxes (on the notion of proof, proposition and concept) show, as all paradoxes do, that a notion is there but that we cannot yet clearly perceive it. As the essential difficulty of the theory of concepts is self-application, so the essential difficulty of the notion of absolute proof is the application of proofs to proofs. In the absence of any restriction, the theory of concepts and thus the theory of absolute proof meet the obstacle of paradoxes.

### 3 Casting a new frame of analysis

In this second part we pass from the Gödelian questionnement on absolute to a brief presentation of the ideas which will irrigate the Girard’s thought on logic, in which we will scrutinize traces of the notion of absoluteness.

We start this section by considering fonctionnal interpretations of arithmetics and analysis. The two logicians contributed to this domain by setting two important functional systems, namely the system  $T$  of Gödel and the system  $F$  of Girard. Both are relevant for our purpose. Developing  $T$ , Gödel realized his first attempt to overcome formalism while preserving constructivity and then hoping to approach a notion of absolute proof in arithmetics, while System  $F$  is the very starting point from which Girard made explicite concepts which will be relevant for formulating Linear Logic and, much latter, Ludics.

Then, we continue the section by focusing on the proofs-as-programs paradigm, the heart for the change in perspectives in proof theory which occurred between 1933 and 1972. Gödel began to think about the premises of the System  $T$  from 1933, after the second incompleteness theorem. In 1972, Girard established the  $F$ -System which extends the parallel between proofs and programs to the second order and from which Girard developed his thought on Logic, based on a radical account of dynamic dimension of Logic.

Finally, we will say a few words on how linear logic arose, based in the proof-as-program paradigm. Proofs-as-programs paradigm and linear logic are both necessary for understanding the rest of this paper. We will focus among the notions of both these domains the ones which are needed to understand both Ludics and our comments on Ludics.

#### 3.1 From Gödel’s system $T$ to Girard’s system $F$

Gödel’s indications on the concept of absolute proof in mathematics are rather tenuous but, moreover, for the notion of logical absolute proof, paradoxes constitute for the moment a dead end. There are thus serious objective difficulties in giving substance to this notion of absolute proof.

As Wang stresses in his introduction to *A logical journey, from Gödel to philosophy*<sup>30</sup>, despite this impasse, Gödel is convinced that there is a feasible path to follow in order to make progress in this investigation: the study of the different ways of conceiving the notion of constructive evidence for proofs.

We have seen how the intuitive concept of computability, obtained thanks to the notion of a Turing machine, allows us to “perceive” the properties of this concept. How to pursue an analogous approach? We could look for the intuitive concept of absolute proof by letting ourselves be guided by the successive extensions of the notion of constructive evidence in the domain of mathematics. The path is passable.

On the one hand, any strictly absolute reliable notion can only arise, from Gödel’s point of view, from an extreme idealization of constructive processes (Wang 1995, 7.1.7, 7.1.9, 7.1.11) and the different stages of this idealization correspond to different philosophical options, each of which has its share of truth.

On the other hand, the mind, (able to recognize its own correction), can conceive more and more powerful systems thanks to which the epistemic constraint of evidence, necessary to the practice of mathematics, can

---

<sup>30</sup>[Wang1996], pp 12-13

show itself in all its different facets. Here we are again faced with the idea of grasping regularities in the comparison of the different steps of construction of a growing hierarchy of abstract systems generated by a controlled and non-local (non-mechanical) process.

This idea is indeed at work in the whole of Gödel's reflections from 1933 to 1972. Immediately after the publication of his incompleteness theorems, Gödel shows his interest in the different constructively acceptable ways to overcome the limitations of formal systems [1933o, 1935, [Gödel41]]. There is obviously a technical (mathematical) interest in this question, but there is also a crucial epistemological issue: as in the case of successive extensions of a formal system, the study of ways of extending the notion of constructive evidence in mathematics can be valuable for the understanding of the way our mind works in all its demonstrative practices. Once something is recognized or accepted as self-evident, a comparative study of the different ways of understanding or constructively extending this self-evidence can play an essential heuristic role. It is in this direction that we should read all Gödel's forays into constructive mathematics, and finally his work on the system  $T$ .

The system  $T$ , presented in his paper in honor of Bernays in 1958, focuses on the notion of intuitive clear abstract concepts sufficient for the foundation of arithmetic. Gödel's incompleteness theorems show that the notion of concrete intuition (based on the combinatorial perception of space-temporal arrangement of symbols) required by Hilbert's finitism is too weak. Moreover, for different reasons, explained in detail in [Gödel38], Gödel is dissatisfied with the notion of intuitionistic constructive proof admitted by Heyting's intuitionistic logic. In particular, the interpretation of the intuitionistic implication is admissible only on the basis of a very abstract concept of absolute derivation, which is for Gödel a sort of jump into a too high level of abstraction. System  $T$  shows that the notion of functional of finite type is the right candidate for ensuring the coherence of arithmetic, on an evident secure basis, without renouncing the classical framework. Gödel's argument is as follows: the coherence of classical arithmetics may be reduced to the one of  $T$ , via an interpretation of the intuitionistic arithmetics to which classical one is reducible. Gödel associates with each formula  $A$  of Heyting's (intuitionistic) arithmetics ( $HA$ ), a formula  $\exists x \forall y A_T(x, y)$  of  $T$ , where  $x$  and  $y$  are sequences of variables of functionals of finite type, in such a way that if  $HA$  proves  $A$  then it is possible to find a sequence of terms  $t$  such that  $T(t, y)$  is provable in  $T$ . Therefore the coherence of  $HA$  is a consequence of the coherence of  $T$ , the latter being not doubtful for Gödel.

In some way, Girard's system  $F$  is an extension of system  $T$ : the result of coherence is extended from arithmetics to analyse. Manipulating functionals of finite type –that is a sufficiently intuitive notion since based on explicit actions– Gödel succeeds in proving coherence of intuitionistic and classical first order arithmetics without conforming to the intuitionistic system. Taking system  $T$  from its formulation in lambda-calculus, called by Troeslra the 'term model'<sup>31</sup>, Girard extends it into a second order lambda calculus so obtaining the system  $F$ . The important result in the context of lambda calculus, is the *strong normalisation theorem*: performing all calculi inside some term, whatever is the order according to which these calculi are performed, we are always sure to obtain a result in a finite number of steps. In order to overcome the impossibility to find a combinatory proof of such a result for first order lambda calculus, Tait introduced in [Tait1967] the notion of *reducibility*. The reducibility property is associated with terms of a given type and it is defined inductively on types. Having proved that reducible terms are strongly normalisable and that all terms are reducible, Tait obtained strong normalisation for System  $T$ . To use such a method in second order, a step further towards abstraction is necessary. Girard introduces the notion of *candidate of reducibility*: a set of typed terms potentially reducible. In particular, candidate of reducibility for types quantified on types, i.e. types on the form  $\Lambda \alpha T$  – as for example  $\Lambda \alpha \alpha \Rightarrow \alpha$  – are defined using a quantification on the candidates on reducibility of type  $\alpha$  in addition to a quantification on types. Among the potential

---

<sup>31</sup>[Troeslra1996] in [CWII] p 234

candidates of reducibility, the relevant reducibility is retrieved and the result of strong normalization for System  $F$  is obtained.

Aside a technical continuity, there is a continuity between the methodological and epistemological positions of the two logicians. To define System  $T$ , Gödel emancipates himself from the formalist constraint of finitude. In system  $F$ , Girard also allows himself the manipulation of highly infinite objects (sets of sets of reducible terms of each type, up to types quantified over types), provided that there exists a representative one that behaves well. Both logicians allow abstraction, provided it is controlled.

Nevertheless, the passage from  $T$  to  $F$  accentuates a break. The Gödelian approach is concerned with the philosophical foundations of logic and mathematics. The Girardian approach is aimed at both a better comprehension and an extension of the use of Proof theory in the development of Computer Sciences. The point of divergence is manifest when we consider the positioning of the two logicians with respect to the lambda calculus. For Gödel, the lambda-calculus is a less accomplished attempt than the notion of Turing's machine to capture the notion of computation. For Girard, the lambda-calculus was the relevant formal system to make explicit the proofs-as-programs paradigm. In fact, the irruption of this paradigm is the split line which separates two moments, two traditions in Proof Theory. Until Curry-Howard work, proofs were used to prove statements. The proof-as-program paradigm gave a new role to logic and proofs. It enabled to conceive proofs as calculi and vice versa to use logic to understand what a calculus is. As we will see, starting from this theoretical achievement, Girard will try to represent as explicitly as possible the computational content of formal proofs, in order to study them as possible objects of more abstract (geometrical) properties such as symmetry, duality, stability, closure, saturation etc. . .

## 3.2 The Proofs-as-Programs paradigm

After the deep correspondence between formal proofs and programs was noted [Curry1934, Howard69], many works contributed to consolidate this bridge between proof theory, theory of computation and theoretical computer science. In general, this correspondence between intuitionistic natural deduction and typed lambda calculus is considered as an isomorphism because the bijection between lambda-terms and formal proofs scrupulously respects the dynamic processes taking place on them.

To understand the main features of this correspondence between typed lambda-calculus and intuitionistic natural deduction, it is enough to consider small fragments of both domains: on one side a typed lambda-calculus where the types are just built from basic types and only the arrow as constructor; on the other side, the intuitionistic minimal logic: the formulas are built from atomic formulas, using only the implication as connective.

The first level of correspondence is the one between formulas and types. One may associate with each formula a type: starting by associating with each atomic formula  $A$  a type  $A^*$ , and by extending the correspondence to each formula by setting  $(A \Rightarrow B)^* = A^* \Rightarrow B^*$ . Then, the correspondence between terms and proofs is essentially based on two straightforward similarities, dual with each other:

1) the first similarity is between, on the one hand, the rule of introduction of the implication which captures in a proof of  $B$  the hypothesis  $A$  in order to introduce  $A \Rightarrow B$ , and on the other hand, the operation of abstraction of the lambda calculus, which constructs a function of  $A^* \Rightarrow B^*$ , denoted  $\lambda x.t$ , by binding a variable  $x$  of type  $A$  when it is used in the construction of a term  $t$  of type  $B$ .

2) the second similarity is between, on the one hand the elimination rule of implication, which links a proof of  $A$  and a proof of  $A \Rightarrow B$ , to produce a proof of  $B$ , and on the other hand, the operation of application, which occurs in lambda calculus between a term  $N$  of type  $A^*$  and a term  $M$  of type  $A^* \Rightarrow B^*$  to produce a term  $(M)N$  of type  $B^*$ .

These similarities are made explicit by representing the construction of  $\lambda$ -terms as proofs of natural deduc-



	Intuitionistic Natural Deduction	Typed Lambda-Calculus
Introduction/abstraction	$\frac{\overline{\vdash A} \quad \vdots \quad \vdash B}{\vdash A \Rightarrow B}$	$\frac{\overline{\vdash x : A} \quad \vdots \quad \vdash t : B}{\vdash \lambda x.t : A \Rightarrow B}$
Elimination/application	$\frac{\vdots \quad \vdots \quad \vdash A \quad \vdash A \Rightarrow B}{\vdash B}$	$\frac{\vdots \quad \vdots \quad \vdash N : A \quad \vdash M : A \Rightarrow B}{\vdash (M)N : B}$

The turnstile symbol  $\vdash$  indicates the derivation of some formula in the left column and the construction of a  $\lambda$ -term in the right one.

Figure 1: The correspondence introduction/abstraction and elimination/application

tion annotated by terms, as shown in figure 1.

From such observations, it is easy to associate with each proof a  $\lambda$ -term. The correspondence between a proof  $\pi$  of a formula  $F$  and a  $\lambda$ -term  $\pi^*$  of type  $F^*$  is defined by induction on  $\pi$ : if  $\pi$  is proof reduced to the single formula  $A$ , then  $\pi^*$  is a variable of type  $A^*$ . If  $\pi$  is a proof of  $A \Rightarrow B$ , obtained from a proof  $\rho$  of  $B$  in which  $A$  is a hypothesis, then  $\pi^* = \lambda x.\rho^*$ . Finally, if  $\pi$  is a proof of  $B$ , obtained from a proof  $\rho$  of  $A \Rightarrow B$  and a proof  $\eta$  of  $A$ , then  $\pi^* = (\rho^*)\eta^*$ . The interesting thing about this static correspondence is that it extends into a dynamic correspondence. To explain this point, we have to specify which are the dynamic processes that we call calculus in each domain.

- Let us recall that a cut, in natural deduction, occurs inside a proof of conclusion  $B$  obtained from two proofs: a first one,  $\eta$  of conclusion  $A$  and a second one  $\rho$  of conclusion  $A \Rightarrow B$  where this latter conclusion has just been introduced<sup>32</sup>. Such a cut triggers a calculus, i.e., a rewriting process which replaces the initial proof by the proof  $\rho$  of  $B$  where we replace each occurrence of the hypothesis  $A$  by the proof  $\eta$ . This operation of substitution may introduce new cuts, in that case the process continues, otherwise it stops on the result of the calculus: a proof of  $B$  without a cut.
- In  $\lambda$ -calculus, what triggers a calculus inside a term is the occurrence of a redex, i.e., a term of the form  $(\lambda x.t)N$ . Such a redex is the occasion of an elementary step of calculus: *the rewriting of  $(\lambda x.t)N$  into  $t[N/x]$* , i.e., the term obtained by replacing each occurrence of  $x$  inside  $t$  by  $N$ . Such a substitution may create new redexes, in that case the calculus goes on, otherwise it stops on a result: a term without redex.

The *dynamic part of the correspondence* is as follows: the proof obtained after the cut-elimination, i.e., when we eliminate the two intro/elimination rules and substitute in the proof  $\rho$  of  $B$ , the hypothesis  $A$  by the proof  $\eta$  of  $A$ , is exactly the one associated with the term  $\rho^*[\eta^*/x]$  obtained from  $(\lambda x.\rho^*)\eta^*$  after the reduction, as shown in figure 2.

<sup>32</sup>This situation where two rules of introduction and elimination of the same implication follow each other immediately, is also called *detour*.

$$\begin{array}{ccc}
 \frac{}{\vdash A} & & \eta \\
 \eta & \vdots & \vdots \\
 \vdots & \vdots & \vdash A \\
 \vdots & \vdots & \vdots \\
 \vdash A \quad \frac{}{\vdash B} & \vdash A \Rightarrow B & \vdots \\
 \hline \vdash B & \text{becomes (after cut elimination)} & \vdash \beta
 \end{array}$$
$$\frac{\displaystyle \frac{\vdash x : A}{\vdash \rho^* : \beta} \quad \vdash \eta^* : A}{\vdash (\lambda x \rho^*) \eta^* : \beta} \quad \text{normalizes in} \quad \frac{\vdash \eta^* : A}{\vdash \rho^*[\eta^*/x] : \beta}$$

The dynamical correspondence described above is the core of Curry-Howard’s paradigm: as the normalization is a calculus starting from terms containing redexes towards terms without redex, cut-elimination is a calculus starting from proofs containing cuts towards proofs without cuts. So, when we speak about the computational content of proofs, we designate the ability of proofs to be supports of calculus while the calculus is the cut elimination process. Let us notice that such an ability is not reduced to the case of natural deduction as our presentation might suggest. We could have presented the process of calculus in sequent calculus, in a perfectly equivalent way, and in fact, the word “cut-elimination” comes from Gentzen’s sequent calculus.

### 3.3 A very brief look at the emergence of Linear Logic

17

Let  $\pi$  and  $\rho$  be two *LK*-proofs of the same formula  $A$ . It is possible to derive the following *LK*-proof:

$$\begin{array}{c}
 \pi \qquad \qquad \rho \\
 \vdots \qquad \qquad \vdots \\
 \hline \vdash A \qquad \vdash A \\
 \hline \vdash A, B \qquad B \vdash A \\
 \hline \vdash A, A \\
 \hline \vdash A
 \end{array}$$

where the rules used above are the following: the ones denoted “ $\vdash \vdash \vdash_w$ ” are *weakening* rules, the one denoted by “ $\vdash \vdash \vdash_c$ ” is a contraction rule and “ $\vdash \vdash \vdash_{cut}$ ” denotes a cut rule. For eliminating this, the process indeterministically may choose to rewrite it either into  $\pi$  or into  $\rho$  and there is no clear way to prefer one over the other.

Figure 3: Lafont’s critical pair

is either non deterministic or merges all the proofs of a given formula: like the proofs  $\pi$  and  $\eta$  in the example, each two proofs of the same formula are possible results of a calculus starting from a unique proof. The calculus is either arbitrary non deterministic, or degenerated: all steps are pairwise equal and the rewriting process is nothing else but equality. The process of cut-elimination is therefore meaningless from a computational point of view.

Such a problem does not occur in intuitionistic logic since the intuitionistic sequent calculus is not symmetric: it does not enable structural rules on the right part of sequents, and the critical pairs disappear.

Linear Logic exceeds the limitations of classical sequent calculus cut elimination. Choosing to deal with weakening and contraction in a logical way instead of a structural one, Girard substituted a fine duality for a too-brutal symmetry. In linear logic, two unary connectives, called *exponential* and denoted “ $!$ ” and “ $?$ ”, dual from each other, enable to manage contraction and weakening -and make disappear structural critical pairs<sup>33</sup>. Here *dual with each other* refers to the linear negation which is denoted by  $(-)^{\perp}$  and which is such that  $(?A)^{\perp} = !A^{\perp}$  and  $(!A)^{\perp} = ?A^{\perp}$ .

Let us recall, in a few words, the main features of Linear Logic. Once contraction and weakening are strictly limited to exponential formulas, it becomes necessary to refine logical connectives. Indeed, it is no longer equivalent to introduce a conjunction between two formulas when both are proved either in the same context or in different ones; it is no longer equivalent to justify a disjunction either by choosing to justify one of the two disjuncts, or by keeping the alternative as long as possible. Therefore, Linear Logic distinguishes between two conjunctions “ $\otimes$  and  $\&$ ” and two disjunctions “ $\wp$  and  $\oplus$ ” respectively called multiplicative conjunction and additive conjunction, multiplicative disjunction and additive disjunction. These binary logical connectives are pairwise dual with each other. Endly, the linear implication uses the multiplicative version of disjunction:  $A \multimap B = A^{\perp} \wp B$ . The linear implication is called resource-sensitive, making explicit that in the proof of  $A \multimap B$ , the hypothesis  $A$  is used exactly once.

By defining the sequent calculus, Gentzen offered the possibility of studying the properties of formal proofs by studying the properties of the formal systems in which they can be defined. Continuing this approach, Girard can, with the linear sequent calculus, give a finer account of the logical concepts hitherto

<sup>33</sup>In the example described in figure 3, you are only able to introduce by weakening a formula  $?B$  in the left side proof  $\pi$  and a formula  $!B$  in the right one  $\rho$ . A cut is therefore no more possible and the problem disappeared.

handled. The linear connectives are indeed refinements of the traditional classical and intuitionistic ones. We saw above how linear conjunctions and disjunctions are refinements of classical ones. Let us mention also the best known of these refinements: the decomposition of the intuitionistic implication into two connectives:  $A \Rightarrow B = !A \multimap B$ , making explicit that in the proof of  $!A \multimap B$ , the hypothesis  $A$  can be used an arbitrary number of times. This refinement of the intuitionistic implication into a still more “algorithmic” formulation – insofar as the possibility of using an arbitrary number of times the argument of a function is represented by a logical connective – prefigures the change of epistemological perspective on proofs that linear logic will allow and that Girard’s reflection will affirm more and more clearly – and to which we will return in section 4.

Although Girard has always emphasized the methodological and technical concerns of his positions, they nonetheless resonate with the philosophical concerns that motivated Gödel when he set up System  $T$  insofar as linear logic is a way to subsume intuitionistic constructivity and classical symmetry. As in intuitionistic systems, the elimination of cuts in linear logic inherits the properties of finitism, determinism and locality required by formalism, i.e., the properties that make this procedure a calculus. For that reason, linear logic and intuitionistic logic are both said to be constructive. Moreover, the framework of Linear Logic also inherits good properties of symmetry from the classical framework: like classical logic, linear logic verifies de Morgan’s laws; linear negation is involutive, like the classical one. As many subsequent works attest, the new methodological tools offered by Linear Logic make it possible to refine the comprehension and the manipulation of classical [Girard1991] and intuitionistic formal proofs [Danos-and-all1997], also merging both in a “unified” logic [Girard1993].

However, a radical change of perspective accompanies this new expression of formal proofs. There is no object in linear logic, and therefore no more predicates expressing relations between objects. More precisely, even if a first order linear logic may be formulated, first order individuals and predicates are not primarily present in Linear logic. We essentially find in it processes for which static usual identification by the use of identical linguistic names must be corroborated by a dynamic identification and negation has too be interpreted as a dual process: rather than a duality true/false, linear negation is a duality in terms of actions: read/write, open/close, give/receive, propose/accept, etc.

This change of perspective has been made possible by the new approach in proof theory driven by the *proofs-as-programs* paradigm; it is accompanied by the emergence of new methodological tools, providing in turn the possibility of new achievements in the understanding of proofs.

- The refinement of binary connectives suggests new properties. The distinction multiplicative/additive connectives may be read in terms of acts, and we then remark that while the introduction of multiplicative disjunction and additive conjunction are *reversible*<sup>34</sup>, additive disjunction and multiplicative conjunction are not reversible. The two reversible connectives are called negative while the non reversible ones are called positive. This is called polarity. Such a difference is very interesting when we study cut elimination process: the rewriting step in the case of a cut on one reversible connective is deterministic, it is non deterministic in the case of a non reversible connective step. Such a difference may be equivalently formulated in the case of proof search: there is no choice for deciding how to continue a proof search when one has to decompose a negative formula, while you have to make a decision to go on a proof search for a positive one.

- This notion of polarity has allowed us to formulate technical properties such as *focalization*: roughly speaking, a proof is focalized when it obeys the following constraint: when constructing from the conclusion to the axioms, if there is a negative formula, decompose it until reaching its positive subformulas, otherwise

---

<sup>34</sup>One knows exactly how to rebuild the premises of one such introduction rule from its conclusion.

choose a positive formula and decompose it until reaching its negative subformulas, and start again. Focalization was made explicit by J.-M. Andreoli [Andreoli1992] for proof search objectives: indeed, focalization is particularly interesting from a heuristic point of view. If a formula is provable in linear logic then it possesses a focalized proof. This fact allows us to drastically reduce the search space of a proof: one just needs to look for a focalized proof.

- *Proof nets* are presented by Girard as an expression of proofs that allows linear logic what natural deduction allows intuitionistic logic: a way of giving an account of proofs that remains as faithful as possible to what they are in informal practice<sup>35</sup>. As we have already mentioned above talking about linear negation, linear proofs can increasingly be thought of in terms of action, as consisting of acts rather than of rules.

Proof nets present themselves as an original expression of linear proofs. This new expression highlights original properties of them. Manipulating proof nets drastically reduces the number of proofs, drastically relaxes the constraint of sequentiality. Besides an indisputable practical advantage: the possibility of parallelism in the transformations allowed by the proofs, we also obtain an important methodological advantage. In sequent calculus as well as in natural deduction, the correction of formal proofs is a natural consequence of their construction: the inductive definition of formal proofs builds only correct proofs. On the opposite, the induction definition of proofnets builds correct and incorrect proofs. The correction depends on the satisfaction of a global criterion concerning the network as a whole. This opens up the possibility of considering the notion of “pre-proof” which will prove fruitful for the continuation of Girard’s research program and which we will return to in the next section.

## 4 Traces of the concept of absolute proof in Girard’s Ludics

The concept of absolute proof is not part of Girard’s vocabulary and concerns. The proofs which the father of Linear Logic speaks about are first of all formal proofs. The reflection that he has explicitly pursued since the end of the 1980s, i.e., to exhibit and manipulate at best the computational content of formal proofs, seems to be far removed from the epistemic and philosophical questions that Gödel aimed at in his analysis of the notion of absolute proof.

Yet Ludics is a new step in Girard’s reflection that deserves to be interrogated from a slightly different point of view, not the natural and already well-developed one that proof theory drives with the *proofs-as-programs* paradigm, but rather the one not yet explored of the possible epistemological new insights that Ludics could offer for the analysis of the concept of proof. This is what we propose to do by exploring traces of the concept of absolute proof in Ludics.

But before such an exploration, we present some of the original features of this theory.

### 4.1 The basic objects of Ludics

Ludics is the theory of “Logic”, developed by Girard in the early 2000s [Girard01], a theory that he himself qualifies as a *theory of interaction*. Undoubtedly, the interaction is that of calculation, the one which starts when a function and an argument are put in contact, and which unfolds, when the process of calculation is convergent, until a result is obtained. But even more abstractly, the interaction that logic accounts for is the one of the dialogue between a proponent who tries to justify a position and an opponent who tries to refute it. The central object of Ludics, called “design”, subsumes as elements of interaction: function *vs* argument, proof *vs* counter-proof, proponent *vs* opponent, . . . Designs are constituted by primitive pieces of interaction,

---

<sup>35</sup>Gentzen said that natural deduction restored the practice of mathematical proof.

called **actions** in Ludics. Finally, actions themselves are defined by keeping only the most primitive and abstract elements of an act in an interactive situation:

1. With regard to the points of view of the proponent and of the opponent, an action is in turn either produced or received.  
Therefore, the actions are polarized. One says that an action is positive when it is produced, and negative when it is received.
2. The focus of such an action is the locus where the action occurs with respect to the interaction. Such a focus may be initial, as at the start of a dialogue, or may have been created by a previous action.
3. The action creates new foci that are available for subsequent acts.

For example, the following sequence is an alternated succession of three such acts; we remark that the first two open only a new focus while the third one opens four foci:

$$(+, L_0, \{L_1\})(-, L_1, \{L_2\})(+, L_2, \{L_{31}, L_{32}, L_{33}, L_{34}\})$$

Recalling that the building of designs is inspired by proof search, such a sequence is represented by using the sequent calculus proof tree representation that has to be read from the bottom up:

$$\frac{\frac{\frac{L_{31} \vdash \quad L_{32} \vdash \quad L_{33} \vdash \quad L_{34} \vdash}{\vdash L_2}}{L_1 \vdash}}{\vdash L_0}$$

Moreover, there is another action: an action called *daïmon*. Indicating the decision to stop an interaction, it is a positive action which neither needs any focus, nor creates new foci and which is denoted by  $\dagger$ .

The execution of **an interaction** is an alternated succession of actions; the interaction continues as long as the action produced by one party corresponds exactly to an action that the other party is willing to receive. Finally, either an agreement is made explicit -by means of the action  $\dagger$ - or the interaction is said to be divergent.

A **design** then consists of such alternating successions of acts of interaction: a design is a set of alternated sequences of acts that are sufficiently compatible with each other to be joined together. Just as a strategy is the union of parts anticipated by a single player and enables him to deal with his opponent, a design is the union of interactions anticipated by a single subject to defend a position against different opponents.

Examples of such *designs* are given in figures fig. 4 and fig. 5. These latter figures illustrate the modelling of dialogues in Ludics. The interaction in the first example is a legal dispute, in the second one we may moreover observe an interaction corresponding to a calculus. We start with a dialogical sequence of statements in natural language and instead of translating them in a formal language, we analyse them in terms of designs (sequences of alternative acts): for each participant to the dialogue the design represents a possible strategy of interaction.

## 4.2 A Theory of Logic

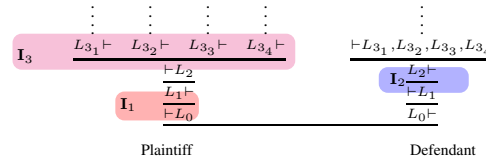
Ludics is built by pushing as far as possible the new perspectives on proofs allowed by the original tools of linear logic:

- It uses the possibility of manipulating, rather than formal proofs, some objects that only claim to be such proofs and still have to satisfy explicit criteria to fully acquire this status –such a possibility first appeared

We show below the modelling in Ludics that we presented in [Fouquere-Quatrini2013]. The dialogue is a legalk debate starting with the following four interventions:

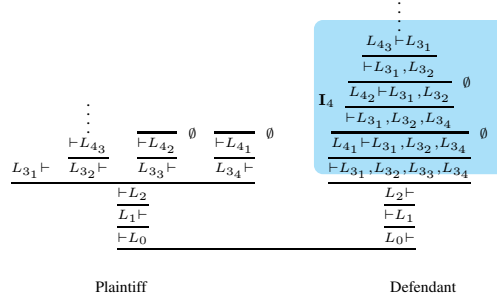
- $I_1$  : I claim that defendant owes me 500 euro.
- $I_2$  : I dispute plaintiff's claim.
- $I_3$  : Defendant owes me 500 euro by  $r_1$  since we conclude a valid sales contract,  
I delivered but defendant did not pay.
- $I_4$  I concede that plaintiff delivered and I did not pay,  
but I dispute that we have valid contract.

We represent below the two steps of the interaction unfolding between the two designs respectively associated with each speaker. The design on the left gives an account of the dialogue seen from the *Plaintiff's* viewpoint, while the one on the right gives an account of the dialogue seen from the *Defendant's* viewpoint.



The first three interventions each consist of a simple act; the first two interventions each create only one place that the opponent will be able to grasp to continue the exchange; the third intervention,  $I_3$ , creates four places, one creating a space (a branch) to eventually specify the law  $r_1$ , a second one to exchange around the valid contract that is said to have been concluded, a third one for the fact that the product has been delivered and finally a fourth one for the fact that payment was not made.

The fourth intervention is more elaborated:



This intervention  $I_4$  is made up of five elementary acts: the first two concede that the product has been supplied, so the branch on which a continuation of the exchange would have been possible on this point is closed in the Plaintiff's design, the empty set of possible places of continuation matches this closure. Similarly, the following two acts remove in both designs any possibility of suing on the grounds that payment has not been made. Finally, in the last act of its intervention, the Defendant gives the floor back to the Plaintiff to justify the validity of the contract that was allegedly entered into.

Figure 4: Formalization in Ludics of a juridic debate.

We consider a very short dialogue:

To the question asked by  $P$ : “All the *Kpelle* cultivate rice, Mister Smith does not cultivate rice. Is Mister Smith a *Kpelle*?”

$N$  answers: “I do not know Mister Smith.”

This dialogue may be said to be **divergent**. The non correspondence between the act produced by  $N$  and the two acts that  $P$  is ready to receive makes explicit this divergence.

$$\begin{array}{c}
 \frac{\frac{\frac{\vdash L_{no} \quad \vdash L_{yes}}{L_R \vdash}}{\vdash L_Q} \text{---} Q}{\vdash L_Q} \\
 \text{P's dialogue acts}
 \end{array}
 \quad
 \begin{array}{c}
 \frac{\frac{L_{not-know} \vdash}{\vdash L_R} \text{---} R}{L_Q \vdash} \\
 \text{N's dialogue acts}
 \end{array}$$

It is possible to rebuild the expectations of  $P$  when she asked her question. To answer the question “Is Mister Smith a *Kpelle*?”,  $N$  is supposed to translate two pieces of information in two designs:

$$\begin{array}{c}
 \frac{\frac{\frac{\vdash C(a)}{\vdash C(a)} \text{---} \emptyset}{K(a) \vdash C(a)} \text{---} \emptyset}{\vdots} \\
 \vdots \\
 \frac{\vdots}{K(x) \vdash C(x)}
 \end{array}
 \quad
 \begin{array}{c}
 \frac{\frac{\frac{\vdash C(b)}{\vdash C(b)} \text{---} \emptyset}{K(b) \vdash C(b)} \text{---} \emptyset}{\vdots} \\
 \vdots \\
 \frac{\vdots}{C(smith) \vdash}
 \end{array}
 \quad \text{and} \quad
 \begin{array}{c}
 \frac{\frac{\vdash}{\vdash} \text{---} \dagger}{\vdash} \text{---} \emptyset \\
 C(smith) \vdash
 \end{array}$$

*If  $x$  is a kpelle  $x$  cultivates rice.*
*Smith does not cultivate rice.*

Then  $N$  is supposed to use these designs to perform logical calculi.

$$\begin{array}{c}
 \frac{\frac{\frac{\vdash C(smith)}{\vdash C(smith)} \text{---} \emptyset}{\vdash \mathbf{K(smith)}} \text{---} \emptyset}{\vdash \mathbf{K(smith)}} \text{---} \emptyset
 \end{array}
 \quad
 \frac{\frac{\frac{\vdash C(smith)}{\vdash C(smith)} \text{---} \emptyset}{\mathbf{K(smith)} \vdash C(smith)} \text{---} \emptyset}{\vdash \mathbf{K(smith)}} \text{---} \emptyset
 \quad \mapsto \quad
 \frac{\frac{\vdash}{\vdash} \text{---} \dagger}{\vdash} \text{---} \emptyset$$

First calculus : from the interaction between the (proofs of the) propositions “Mister Smith is a *Kpelle*” and “If Mister Smith is a *Kpelle*, Mister Smith cultivates rice.” results in a (proof of the) proposition “Mister Smith cultivates rice”.

$$\begin{array}{c}
 \frac{\frac{\frac{\vdash C(smith)}{\vdash C(smith)} \text{---} \dagger}{\vdash \mathbf{C(smith)}} \text{---} \emptyset}{\vdash \mathbf{C(smith)}} \text{---} \dagger
 \end{array}
 \quad
 \frac{\frac{\frac{\vdash}{\vdash} \text{---} \dagger}{\vdash} \text{---} \emptyset}{\mathbf{C(smith)} \vdash} \text{---} \dagger
 \quad \text{results in} \quad
 \frac{\frac{\vdash}{\vdash} \text{---} \dagger}{\vdash} \text{---} \dagger$$

Second calculus, the addition of a data “Smith is a *Kpelle*” yields a *contradiction*.

$N$  is therefore suggested to answer “no” in order to avoid this contradiction.

Figure 5: Needed calculi to answer questions.



with the notion of proof net.

- It uses the expression of logical steps in terms of acts. As described in section 2, the internalization in the binary linear connectives of their contextual conditions of introduction allows to read the logical steps of a linear proof in terms of commitment vs availability according to the polarity of the connective.
- It uses the possibility given by the *focalisation property* of Linear Logic to reduce drastically the search space of a proof.

Summing up, designs are pre-proofs and more precisely focalized pre-proofs, and even more *abstractions of focalized pre-proofs*. Here *abstraction* indicates that, from the notion of pre-proof, we only retain that it is a strategy of justification with respect to successive challenges. We do not manipulate formulas as in a formal proof, but only actions: there are no fixed propositions, but only choices, either the choice of the form of decomposition (next steps to explore towards an agreement) or the choice to stop the interaction.

Provided with such objects, Ludics allows us to recover the elements of a formal system: formulas, formal proofs and truth.

We find the **logical formulas** as sets of such pre-proofs that we can gather because they share compatible choices: they may be gathered to defend a same position against different attacks provided that they make the same choices in the same situation. Specifically, the objects that will correspond to logical formulas (i.e., the propositions) are called *behaviours*. They are sets of designs having the same *counter-designs*, a design  $\mathcal{E}$  being a counter-design of a design  $\mathcal{D}$  when the interaction between both  $\mathcal{D}$  and  $\mathcal{E}$  converges. For being a behaviour, such set of designs has to verify a property of stability: every design having the same counter-designs already belongs to the behaviour<sup>36</sup>.

We find **formal proofs** amongst the designs belonging to behaviours associated with formulas. Proofs are *good* designs, that means *winning designs*, i.e., those that never make concessions during the interactions with all possible counterdesigns<sup>37</sup>. In other words, proofs are justifications that successfully pass all attempts to challenge them.

We find **true propositions** amongst the behaviours in which some designs satisfy the required properties for being a proof.

In the seminal article [Girard01] where Girard defines Ludics, a full completeness result is demonstrated. A correspondence is set between the second order multiplicative, additive linear sequent calculus:  $MALL_2$  and the objects of Ludics. The full completeness result qualifies this correspondence: it means that not only the formal proofs are associated with good designs but a good design belonging to the interpretation of a  $\Pi_1$  linear formula is the interpretation of a proof of this formula. Let us notice that  $MALL_2$  is a linear sequent calculus without the exponential connectives, specifically, a sequent calculus without the contraction rule. This has some consequences on the potential interest of Ludics for modelling calculus, which we shall return to in the following section.

Finally, note that in the context of the proof-as-program paradigm, in which Ludics has arisen, mathematical proofs have not a key role for the investigation of logical proofs. In the Proof Theoretical frame, formal proofs may be declined as juridic demonstrations or grammatical analysis as well as justification of theorems. In Ludics, Designs may be attempts to justify mathematical formulas as well as natural language utterances. In fact, besides formal mathematical proofs, Ludics enables to retrieve Logic not only beyond mathematics but also beyond formal Logic.

<sup>36</sup>While in general, if we add a new design to any set of designs  $\mathbf{C}$ , the set of shared counter designs -called the orthogonal of  $\mathbf{C}$ -decreases and dually  $\mathbf{C}$  increases, a behaviour is a set of designs equal to its bi-orthogonal.

<sup>37</sup>Other technical properties are also needed: proofs are designs which unfold without useless detours, and which are uniform in their infinite development.

### 4.3 Beyond formal proof

Ludics emerges in the perspective of giving the best account of the cut elimination process and to this effect it retains from the formal proofs only those elements which support this process, one of the motivations being that such a process is a calculus. Actually, the objects to which original Ludics gives birth exceed this strict mission in accounting for calculus. They exceed the nature of objects that we may grasp just as a combination of signs and that we may mechanically recognise.

First of all, the objects of Ludics are *infinite*. Not only the basic objects, the designs, are infinite trees in height and width, but also the sets of such objects, which are candidates to play the role of formulas, are infinite.

Then, the objects of Ludics are *abstract*. Certainly, they are defined from elements of a sure materiality: namely, actions make explicit the passage from a formula to its sub formulas by a simple link between an address and its subaddresses. An address is then a place on which a formula can occur. But such a formula is not known in advance, it is not fixed and besides it does not necessarily exist<sup>38</sup>.

Finally, the objects of Ludics are *non-local*. More precisely, the properties that allow us to express which of these candidates are indeed logical formulas, which of the other candidates are indeed proofs, are expressed by appealing to operations that involve an infinite number of other objects.

We are quite far from formal proofs, and from the standards of Proof Theory which was intended to be a strict finite manipulation of signs. How can we understand these objects<sup>39</sup> ?

One of the originalities of Ludics is that it is obtained thanks to what we could qualify as an “ontological reversal”. Usually in Logic, and particularly in Proof Theory, the notion of formula is the primitive one. Proofs, and notably formal proofs, are made of formulas, and finally, we consider amongst the formal rules constituting proofs the cut rule –the one that we abstracted as *interaction*. In Ludics, on the contrary, we get the notion of formula only at the end of the process of analysis, which starts from the primitive notion of interaction, passes through those of pre-proofs to arrive at pre-formulas, then formulas and finally proofs. Ludics adopts a much more primitive level than that of formal logic, taking the objects of logic in their interactional manifestation, before their linguistic shaping, as shown by the examples given in the previous subsection.

But then, what do the objects of Ludics tell us about proofs? First of all, that the proofs are not reducible to formal proofs. One can undoubtedly give a relevant account of almost all the behaviors, the properties, the uses, the role of proofs by manipulating them, by writing them, by using them as formal proofs. However, all is not said with the formalization. By proposing to circumscribe the elements of formalization to their sole role as a support for interaction, Ludics brings out the interstices between segments reducible to formalization: some notions are correctly restituted by formalization, others are not. Amongst these features that are not related to formalism, we underline:

- *The dimension of the subject*. It is perhaps the processual aspect of Ludics which allows us to highlight that, in order to succeed in characterizing a proof, it is necessary to give back its place to the subject. To better understand such a statement, we have to remember how ludical objects arise. As seen before, the characterization of proofs in Ludics proceeds in several steps: first there are sketches of proofs, which are not yet proofs but only investigations; then these investigations are gathered in a single intention, that of specifying what it is proved; intention which in its turn is smoothed out, i.e., represented by a formula; and finally it is possible to verify that there are among the drafts, *successful* proofs.

---

<sup>38</sup>In [Fouquere-Quatrini2020] we characterized which among the behaviours do correspond to logical formula.

<sup>39</sup>Beyond the undeniable methodological role they have played in the fields of the theory of computation and game semantics by bringing out original concepts, such as those of internal completeness or incarnation.

Indeed the subject emerges from the way Ludics manipulates these sketches of proofs: “actions” or “acts” are the primitive elements of which these sketches are made. It is the subject who then decides on the eventual positions admitted in a given context of interaction, in other words, at what moment one decides that the investigation stops, not on an abandonment but on an assured position. Finally, it is the subject who commits himself to delimiting the scope of the intention in question. In other words, what other drafts does one decide to consider with which the turn of the question will be considered sufficient.

Actually, *intersubjectivity* is a dimension at least as relevant as the one of the subject. In his process of justification, the subject is motivated by the injunction to confront attempts at rebuttal, or even simple requests for clarification from other subjects. It is only once all these potential opponents admit the arguments of the proponent that the justification of this latter becomes valid.

- *The independence with respect to language.* We have seen above that, when we put everything on the table, as Ludics invites us to do, we can see the limits of formalization in order to investigate what proofs are. We see also the limits of language. Rather, we notice that language is not presupposed in the definition of proofs: to define the drafts of proofs, Ludic needs to keep only instructions of investigation paths. For example, the logical proof par excellence, the identity-proof (see figure 6), not only is the same for all formulas, but does contain nothing more than the expression of this plasticity: no matter the form of what is said, what is important is to ensure the identity of the formulas by the identity of the acts. It is only in a second time, when we decide to identify regularities and to stop there, that we recover the notion of formulas and of proofs establishing them.

Let us observe that the independance with respect to domains of objects is, in Ludics, inherent to the independance with respect to language. As there is no formula *a priori*, there is no object of predication, then there is no object at all.

- Finally, a property of *stability* is required for the objects of logic. It is only once we have succeeded in capturing an invariant that the notion of “logical formula” emerges. It is necessary to saturate the set of designs that can be deployed from the same position in a **behaviour**: a set which is invariant insofar it is stable with respect to any possible new convergent interactions. The notion of proof must then be understood from this perspective. It is consubstantial with that of proposition, i.e., the statement of which it is a proof. A proof belongs to the heart of the set of justifications that it contributes to stabilize as a statement and is essential to the form of this statement, it allows to ensure the validity of this statement against any possible attempt at refutation.

The richness of Ludics is to allow us a back and forth between the level of the most concrete and local argumentative situations and that of the most abstract and general theoretical definitions in order to pose logical objects as invariants of the interaction. How can we understand this notion of invariance? A set of designs only gains the status of behaviour once we decide that it is sufficiently relevant to the exploration of a position and that we can close it with respect to interaction. In other words, only then does it become stable (invariant) and gain the status of a proposition. Therefore, only then can the elements of the set of designs acquire the status of proofs.

Consequently, more than a formal system useful to define and manipulate formal proofs, Ludics provides us with both epistemological and euristic approach of proofs, enabling at the same time a very abstract and a very concrete viewpoint on proofs. The notion of design highlights a concept of proof that depends as much on the statement it demonstrates as the statement itself depends on its justifications. At the same time, designs are arrangements of the basic actions needed to support reasoning at the lowest level of interaction.

According to this theoretical approach, we may outline some features of the concept of absolute proof. A proof in Ludics is a good candidate for embodying such a concept. We find such ludical proofs among preproofs (designs), which inherit of the properties of designs: infiniteness, abstraction, non locality, independence with respect to language, independence with respect to domains of objects, relevance with respect to intersubjectivity, stability. Moreover, a proof in Ludics inherits of the properties of a successful design (proof), that is: a justification that unfolds itself without any useless detours towards any request for clarification, leaving nothing in the dark, ready to respond to any attempt at rebuttal.

This is not enough, we have just seen that a proof is attached to what is demonstrated. If a proof is absolute, what it proves is absolutely demonstrable. It is difficult to conceive of any other situation than that of a logical proof.

Among the logical proofs, in Ludics, a special attention may be paid to the proof of identity, represented in figure 6. In ludics, there is not an identity proof for each formula but a generic identity proof (up to localisation). Let us emphasize that this generic identity proof is an infinite proof, in width and in height, at the same time that it is perfectly regular, indicating the perfect symmetry between the acts that have to be performed to decompose a formula, either when it is analyzed or when it is assumed. In other terms, the identity proof in ludics is the juxtaposition in a unique design, in a given locus, of every identity formulas. The first rule of this design subsumes in a sole rule all possible decompositions of a given formula. Then, above each choice of decomposition, the symmetry is restored by decomposing according to a schema which has to be identical with the initial one.

In Ludics, the form of proofs is very much constrained by the rigidity of polarisation. Thus, in a logical proof, the bulk is that of the identity proof with the exception of the few previous rules used to determinate the form of the statement being proved. Ludics makes explicit how much justifying a formula is in fact establishing an identity. Figure 7 shows an example of such a logical proof and its differences from the identity proof.

In some way, the ludical identity proof would thus appear as the matrix of any absolute proof. It is interesting that we find here the idea dear to Leibniz of analyticity being reduced to identity. Gödel subscribed to this idea in his article on Russell [Gödel1944]. Still more resonating with what says Gödel in this article, Ludics enlightens the importance of identity in the course of proof, while it does not disqualify the role of logical connectives that remain essential for restituting meaning.

## 5 Conclusion

As announced in the introduction, this article has attempted to analyze Girard's Ludics in the light of the Gödelian concept of absolute proof. This analysis gives substance to Gödel's indications on the concept of absolute logical proof through the implementation offered by Ludics but also in order to give a fully epistemological and philosophical reading of Girard's results.

The analysis of section 3 has allowed us to stress some resonances between absolute and ludical proofs. As required by the notion of absolute proof, ludical objects, which are by nature independent from any domain of application, gain their logical status of propositions and proofs, provided that they satisfy the properties of stability and independence from language. Moreover, ludical logical proofs resonate with the gödelian idea that meaning is relevant also for tautologies, whose proofs are -quasi- reducible to identity.

What gives full meaning to these resonances? Our idea is the following. The objective differences between the two approaches are due to different ontological presuppositions used by the two authors. The

The following design is the generic identity-proof of every formula

$$\begin{array}{c}
 \begin{array}{ccc}
 \mathcal{F}ax_{\eta.1,\xi.1} & \mathcal{F}ax_{\eta.n,\xi.n} & \mathcal{F}ax_{\eta.k,\xi.k} \quad \mathcal{F}ax_{\eta.m,\xi.m} \\
 \vdots & \vdots & \vdots \quad \vdots \\
 \eta.1 \vdash \xi.1 & \eta.n \vdash \xi.n & \eta.k \vdash \xi.k \quad \eta.m \vdash \xi.m \\
 \hline
 \vdash \xi.1, \dots, \xi.n, \eta & \dots & \vdash \xi.k, \dots, \xi.m, \eta
 \end{array} \\
 \hline
 \mathcal{F}ax_{\xi,\eta} = & \xi \vdash \eta
 \end{array}$$

The loci  $\xi$  and  $\eta$  are intended to be the places of any formulas. If we suppose that  $X$  is the formula placed in  $\xi$  and  $Y$  the one placed in  $\eta$ , we consider that  $X$  may be decomposed as a (infinite) disjunctive formula and the first actions  $(-, \xi, \{i_1, \dots, i_p\})$ ,  $\dots$ ,  $(-, \xi, \{j_1, \dots, j_k\})$  are every possible choices of the subformulas of this disjunction. Then if we focus on one of these actions, for example  $(-, \xi, \{1, \dots, n\})$ , it indicates, on one hand any choice of a sub-formula among the disjunction. On the other hand, it constrains the decomposition of this sub-formula into  $X_1 \wedge \dots \wedge X_n$ . The next action is then  $(+, \eta, \{1, \dots, n\})$  which constrains an identical decomposition for  $Y$  into  $Y_1 \wedge \dots \wedge Y_n$ . In this way we made explicit next subdesigns, each of them being based on  $\eta_i \vdash \xi_i$ . All these subdesigns are themselves the repetitions of the same processes: as  $X$  and  $Y$  were constrained having the same two first steps of decomposition,  $Y_i$  and  $X_i$  are constrained having the same decomposition. Finally  $X$  and  $Y$  have the same decomposition, they are two localization of the same logical formula.

Figure 6: Identity Proof

$$\begin{array}{c}
 \begin{array}{ccc}
 \mathcal{F}ax & & \mathcal{F}ax \\
 \vdots & & \vdots \\
 L_{\neg A} \vdash L_{\neg A} \quad L_{\neg B} \vdash L_{\neg B} & & \mathcal{F}ax \\
 \hline
 \vdash L_{A \wedge B}, L_{\neg A}, L_{\neg B} & & \vdots \\
 \hline
 L_{\neg(A \wedge B)} \vdash L_{\neg A}, L_{\neg B} & & L_{\neg C} \vdash L_{\neg C} \\
 \hline
 \vdash L_{\neg A}, L_{\neg B}, L_{\neg C}, L_{(A \wedge B) \wedge C} \\
 \hline
 \vdash L_{B \wedge C} \vdash L_{\neg A}, L_{(A \wedge B) \wedge C} \\
 \hline
 \vdash L_{\neg A}, L_{\neg(B \wedge C)}, L_{(A \wedge B) \wedge C} \\
 \hline
 L_{A \wedge (B \wedge C)} \vdash L_{(A \wedge B) \wedge C}
 \end{array}
 \quad
 \begin{array}{ccc}
 \mathcal{F}ax & & \mathcal{F}ax \\
 \vdots & & \vdots \\
 L_{\neg A} \vdash L_{\neg A} \quad L_{\neg B} \vdash L_{\neg B} & & \mathcal{F}ax \\
 \hline
 \vdash L_{A \wedge B}, L_{\neg A}, L_{\neg B} & & \vdots \\
 \hline
 L_{A \wedge B} \vdash L_{A \wedge B} & & L_{\neg C} \vdash L_{\neg C} \\
 \hline
 \vdash L_{\neg(A \wedge B)}, L_{\neg C}, L_{(A \wedge B) \wedge C} \\
 \hline
 L_{(A \wedge B) \wedge C} \vdash L_{(A \wedge B) \wedge C}
 \end{array}
 \end{array}$$

The right side proof is one branch of the generic identity proof (Fax), while the left side one is neither a Fax nor a branch of Fax. The sole difference between the two is the succession of two rules that mark a forced language rearrangement.

Figure 7: Proof of the associativity of conjunction

similarities are due to the fact that both share a common analysis on the insufficiencies of formal proofs and both aim, from different perspectives, for a notion of proof in terms of acts. The key difference is that, for Gödel, the fundamental notion is concept, whereas for Girard it is interaction. But interactions are a kind of events and as we said in the introduction, we have been guided by the following analogy: Gödel's notion of absolute proof is to Girard's Ludics as the realism of concepts is to the nominalism of events. It is time to explain this idea in more detail, since we now have some elements to justify it.

For Gödel's conceptual realism we have underlined its fundamental aspects. Concepts are considered by Gödel as acts or principles of activities, for which we cannot foresee all realizations, and which do not presuppose any prior domain of application. The characteristic of a logical concept is its universal applicability. The concept of absolute proof must therefore be analyzed in terms of the fundamental acts involved in any demonstrative activity, independently of any linguistic expression of it. The difficulty Gödel encountered in the realization of a general theory of absolute proofs is related to the problem of self-application.

Hence our question: can the notion of absolute proof be found in a theoretical framework completely different from the realism of concepts?

To answer this question, let us first try to illustrate nominalism of events and try to explain in what sense it seems to us the ontological position best suited to account for the implicit presuppositions of Girard's logical analysis. First, let us say that the term "nominalism of events" comes from Jules Vuillemin and his investigation of what he calls the fundamental forms of predication, i.e., the plural and fundamentally different ways of reaching agreement on the truth conditions (or assertability conditions, if we prefer) of propositions [Vuillemin 1996, Vuillemin 1986]. Vuillemin considers the analysis of these forms, based on what he calls a general semiology, as an indispensable prelude to the analysis of philosophical plurality. Speakers must agree on the truth conditions that predication is assumed to aim at, if their communication is to be successful. When we assert that something is such (when we predicate something about something) we presuppose that those who are listening to us can agree with us on what leads us to such an assertion. However, there is, according to Vuillemin, an important element that can come into play in the determination of the conditions of possibility of agreement. This element is the materiality of the sign, the very enunciation of a statement belonging to a given linguistic code. This enunciation is an event, which happens in a given place and time and in the presence of the speakers. Let us see the consequences of such an assertion.

When we say "It's raining" we use the noise that our statement makes at the moment we pronounce it. This noise propagates like a material wave and is therefore capable of fixing the center of a system of space-time coordinates that the sound wave of the utterance induces around it. The enunciation itself "it is raining" functions as a predicate that applies to the coordinates of the enunciation. The event *Rain*, is not instantiated but localized in a form sufficient for individuation with respect to the present space and time of the enunciation. The localization is thus pragmatically fixed, the time of a dialogue, in a concrete context. Dialogical correlation is thus recognized as essential for this form of predication, which Vuillemin calls circumstantial. In this form, then, interaction between speakers is necessary for the determination of the truth conditions of the statements. The "I" who utters is the one who determines the reference point. The relativity of the choice of the marker to the dialogue partners makes the truth value depend on the circumstances of the utterance.

The ontology that circumstantial predication calls for is therefore that of events. What is, from the point of view of the theory of knowledge, an event?

An event is always local, concrete, occurring in determined circumstances. If there are "laws" that govern a universe of events, they must, first of all, concern the identity of events occurring in different circumstances, and this will only be possible if we analyze the punctual modifications that each event can produce in its environment.

From Stoic physics, the philosophy of antiquity which Vuillemin classifies as adopting nominalism of

events as ontology, Vuillemin points out the following ideas:

- a) Superposition phenomena were central to the Stoic analysis of tides. The latter are analyzed and predicted on the basis of superpositions at the same place and time of more elementary periodic events.
- b) The stoic analysis of the acoustic phenomena had led them to take into account the interference phenomena. The latter consists in the capacity that two events have to cancel each other out or to reinforce each other, by superimposing themselves under defined conditions.
- c) The Stoic analysis of celestial phenomena is independent of a previous substantial ontology. In modern terms one could say that the Stoics have approached the concept of field laws. A complex phenomenon is analyzed as an abstract whole whose periodic variations must be identified as a function of time.

Predicting, for Stoic physics, therefore involves focusing on the correlations between processes without entering into an ontological investigation on the nature of the causes or possible “substances” supporting the events. The necessity of the physical laws is then reducible to invariance in space and time, an invariance that only global structural conditions can guarantee.

This detour through Stoic physics and its notion of physical law, besides evoking some of Girard’s forays into contemporary physics<sup>40</sup>, is essentially intended to make more explicit the notion of interaction. This is the starting point towards the notion of absolute proof in Girard’s framework and it seems to us that interactions can be considered as events.

An interaction is a local, concrete and deterministic event triggered by a subject (human or not) which has its own dynamics. If now, from our point of view of humans, we ask ourselves how to use interactions to construct proofs, we find ourselves in the same situation of the Stoic physicist who wants to make prediction.

We must therefore:

- (a) Accept that the identity of meaning of our assertions or demonstrative acts is only circumstantial and that any process of identification presupposes finely determining the conditions of transfer from one place of enunciation to another.
- b) That these conditions of identity can only be established in terms of invariance (stability) with respect to the abstract determination of a space of modifications, established according to our needs (proofs and counter-proofs).

Considering interactions as events enables us to give a more precise analysis of the elementary acts which constitute proofs. The Girardian idea to found intersubjectivity on interaction and on agreement conditions gives substance to the Gödelian idea of absolute proof while avoiding the impasse of the self-application of concepts.

Nevertheless, it seems to us that there is a kind of echo of this problem also in Ludics: initially, ludical objects did not directly fulfill the mission assigned to them to support the calculation, insofar as the rule of contraction had difficulty finding its place in the original formulation of Ludics. However, the rule of contraction, which allows us to simulate repetition and iteration, is necessary for computation.

Interaction inside a process is not self application but requires internalizing knowledge that some sequences of actions are identical. As we pointed out, the identity between two events is the main issue of

---

<sup>40</sup>It would be interesting to complete our frame of analyse with a section "Girard and the Relatives" which would have been the counterpart of our first section. In such a section we would have analysed the relation between Girardian conception on physics and his conception on logics. In particular, it would be important to understand which aspect of Ludics need to be overpassed in order for Girard to continue his program : transposing the conceptual shift realized by the mathematics of quantum physics to Logic. However, we decided not to weigh down the content of this article, leaving for another occasion this more general epistemological analysis. For the presentation of this program see [Girard2021]

nominalism of events. For this reason, perhaps, we find a similar problem in Ludics. Recognizing, in a proof, two occurrences of the same formula and merging them into one seems to be unanalyzed in terms of elementary actions. Extending Ludics to encompass the duplication/merging phenomena that the contraction rule allows has been possible only with certain inflections of the theory: either by reintroducing the Language as suggested by K. Terui [Terui2011], or by abandoning the strict dialogical framework by introducing neutral actions as suggested by C. Faggian and M. Basaldella [Faggian-Basaldella2011].

The connection we propose between nominalism of events and Ludics, which relies notably on the dialogical approach and on the emancipation from language, would no longer be so direct for these alternative versions. It would be interesting to explore deeper such a question by investigating more precisely what is the place of contraction (which presupposes a kind of identity) in the epistemological reflection on proofs and by questioning the possible similarities with the problem of self-application.

## References

- [Andreoli1992] J.-M. Andreoli, *Logic Programming with Focusing Proofs in Linear Logic*, J. Log. Comput, vol 2, n)3, pp 297-347, 1992.
- [deBruijn68] N. de Bruijn, *Automath, a language for mathematics*, Department of Mathematics, Eindhoven University of Technology, TH-report 68-WSK-05. Reprinted in revised form, with two pages commentary, in: *Automation and Reasoning*, vol 2, Classical papers on computational logic 1967 1970, Springer Verlag, 1983, pp. 159–200.
- [Catta-Piccolomini d’Aragonna-to appear] D. Catta, A. Piccolomini d’Aragona, *Informal and Absolute Provability: From Kreisel and Gödel To Prawitz And Girard*, forthcoming.
- [Crocco2019] G. Crocco, *Informal and Absolute Proofs: Some Remarks from a Gödelian Perspective*, Topoi 38, pp 561–575, 2019.
- [Curry1934] H. B. Curry, *Functionality in Combinatory Logic*. Proceedings of the National Academy of Sciences of the United States of America. 20 (11): 584–590, 1934.
- [Curry1958] H. B. Curry, B. Haskell, Feys, Robert (1958), *Combinatory Logic. Studies in Logic and the Foundations of Mathematics*, Craig, William (ed.), North-Holland Publishing Company, 1958.
- [Joinet2007] J.-B. Joinet, *Sur le temps logique*, in J.-B. Joinet (dir.), *Logique, Dynamique et Cognition*, Coll. Logique, langage, sciences, philosophie, Publications de la Sorbonne, Paris, p. 31–49, 2007.
- [Joinet2023] J.-B. Joinet, *Nature et logique de G. Gentzen à J.-Y. Girard*, *Logique et Analyse*, n. 234, 2016, pp. 157–71. JSTOR, <https://www.jstor.org/stable/26767822>. Accessed 22 Feb. 2023.
- [Danos-and-all1997] V. Danos, J.-B. joinet, H. Schellinx, *A deconstructive logic: linear logic*, The Journal of Symbolic Logic, Volume 62, Issue 3, pp. 755 - 807, September 1997.
- [Faggian-Basaldella2011] M. Basaldella, C. Faggian, *Ludics with Repetitions (Exponentials, Interactive Types and Completeness)*, Proceedings - Symposium on Logic in Computer Science 7(2), April 2011.
- [Fouquere-Quatrini2013] C. Fouqueré and M. Quatrini, *Argumentation and Inference: A Unified Approach*, Baltic International Yearbook of Cognition, Logic and Communication, vol. 8, n)4, pp 1-41, 2013.



- [Fouquere-Quatrini2020] C. Fouqueré and M. Quatrini, *Study of Behaviours via visitable paths*, Logical Methods in Computer Science 14(2), May 2020.
- [Giaquinto2007] M. Giaquinto *Visual thinking in mathematics, an epistemological study*, Oxford University Press 2007.
- [Girard1972] J.-Y. Girard, *Interprétation fonctionnelle et élimination des coupures de l'arithmétique d'ordre supérieur*, Thèse de doctorat, Université Paris 7, Diderot, 1972
- [Girard1987] J.-Y. Girard, *Linear Logic*, Theor. Comput. Sci. vol 50, pp 1-102, 1987
- [Girard1991] J.-Y. Girard, *A new constructive logic: classical logic*, Mathematical Structures in Computer Science, Volume 1, Issue 3 , pp. 255 - 296, November 1991.
- [Girard1993] J.-Y. Girard, *On the unity of logic*, Annals of Pure and Applied Logic 59 (3), pp 201-217, 1993
- [Girard01] J.-Y. Girard, *Locus Solum: From the rules of logic to the logic of rules*, in Mathematical Structures in Computer Science (vol.11 n3), 2001, pp 301-506.
- [Girard2021] J.-Y. Girard. *Transcendental syntax I : deterministic case*. Mathematical Structures in Computer Science, 2015. Computing with lambda-terms. A special issue dedicated to Corrado Böhm for his 90th birthday.
- [LGT90] Girard, J.-Y., Lafont, Y., and P.Taylor, *Proofs and types*, volume 7 of Cambridge tracts in theoretical computer science. Cambridge University Press, Cambridge, 1990.
- [Gödel1936a] K. Gödel, *On the length of proofs*, in CW I, pp 397–398.
- [Gödel1946] K. Gödel, *Remarks before the Princeton centennial conference on problems of mathematics* (1946), in CWII, pp150–153.
- [Gödel38] K. Gödel, *Lecture at Zilsel's*, in CW III. (\* 1938a), pp 87–113.
- [Gödel41] K. Gödel, *In what sense is intuitionistic Logic constructive?*, in CW II. (1941), pp 241–251.
- [Gödel1944] K. Gödel, *Russell's mathematical logic* , in CW II. (1944), pp 119–141.
- [Gödel58] K. Gödel, *On a hitherto unutilized extension of the finitary standpoint* , in CW II. (1958), pp 87–113.
- [Gödel72] K. Gödel, *On an extension of finitary mathematics which has not yet been used* , in CW II. (1972), pp 271–280.
- [CWI] K. Gödel, *Collected Works*, volume I, Publications 1939-1936, S Feferman, J. Dawson, S. C. Kleene, G.H. Moore, R.M. Solovay, J. van Heijenoort, Oxford University Press, New York 1986.
- [CWII] K. Gödel, *Collected Works*, volume II, Publications 1938-1974, S Feferman, J. Dawson, S. C. Kleene, G.H. Moore, R.M. Solovay, J. van Heijenoort, Oxford University Press, New York 1990.
- [CWIII] K. Gödel, *Collected Works*, volume III, Unpublished Essays and lectures, S Feferman, J. Dawson, W. Goldfarb, Ch. Parsons, R.M. Solovay, Oxford University Press, New York 1995.

- [CWV] K. Gödel, *Collected Works*, volume V, Correspondence H-Z; S, Feferman, J. Dawson, W; Goldfarb, Ch. Parsons, W. Sieg (eds), Oxford University Press, 2003.
- [Howard69] W. Howard, A. William *The formulae-as-types notion of construction* [original paper manuscript from 1969], in Seldin, Jonathan P., Hindley, J. Roger (eds.), *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, Academic Press, pp. 479–490, 1980.
- [Kennedy2021] J. Kennedy *Gödel, Tarski and the Lure of Natural language*, Cambridge University Press, 2021.
- [Leitgeb2009] H. Leitgeb *On formal and informal provability*, In Ø Linnebo, O. Bueno (eds) *New waves on Philosophy of mathematics* Palgrave MacMillan London 2009.
- [Mancosu2008] P. Mancosu, *The Philosophy of mathematical Practice*, Oxford University Press 2008.
- [Marfori2010] M. Antonutti-Marfori *Informal Proofs and Mathematical rigour* *Studia Logica*, 96(2) pp261–272, 2010.
- [Montesi-Piccolomini d’Aragona2022] F. Montesi, A. Piccolomini d’Aragona, *Prawitz’s Semantics And Walton’s Argument Schemes: A Tentative Reading And Application of Kreisel’s Informal Rigour*, in P. Cantù, B. Halimi, G. Heinzmann, F. Patras, *Objectivity In Mathematics, special issue of Noesis*, 2022.
- [Rav1999] Y. Rav *Why do we prove theorems?*, *Philos Math* 7 (1) pp 5–41 1999.
- [Tait1967] W.W. Tait. *Intensional interpretations of functionals of finite type I*. *The Journal of Symbolic Logic*, 32(2):198D212, 1967.
- [Terui2011] Kazushige Terui, *Computational ludics*, *Theor. Comput. Sci.* vol 412 n;20, pp 2048-2071, 2011.
- [Troelstra1996] A.S. Troelstra, *Introduction to Gödel 58 and Gödel 72*, in CWII, page 234.
- [VanAtten2023] M. van Atten *Natural constructive proofs of  $A$  via  $A \rightarrow B$ , proof paradoxes, and impredicativity*, forthcoming.
- [Vuillemin1996] J. Vuillemin, *Necessity or Contingency: The Master Argument*, Center for the Study of Language and Inf., Chicago University Press, 1996 (First french edition, Les éditions de minuit, 1984).
- [Vuillemin1986] J. Vuillemin, *What are philosophical systems?*, Cambridge university press, 1986
- [Wang1974] H. Wang *From Mathematics to Philosophy*, Routledge & Kegan Paul, 1974.
- [Wang1996] H. Wang *Logical Journey From Gödel to Philosophy*, MIT Press, 1996.