Presentation of the scientific activity
Lorenzo Tortora de Falco

June 11, 2018

The aim of this note is to provide some comments on my curriculum vitae and the list of my publications, in order to explain the rationale behind my scientific activity.

In Section 2, I present my main research contributions, that I partitioned referring to four fundamental questions. After a general description of every question, I give some hints on my personal work; this section is thus mainly addressed to logicians. The rest of the note is devoted to describe the general frame of my scientific activity (Section 1) and of my training activity (Section 3).

1 General presentation

If I had to express synthetically my very personal motivations, I would probably say that I am interested in understanding the structures of thought. Of course, this was not at all clear to me at the beginning of my studies: it just slowly showed up, and naturally led me to Logic. The general questions I became interested in were thus the most classical and traditional that one can imagine: the nature of proofs, the notions of contradiction and consistency, the nature of computation, the nature of infinity. . .

My research activity has always been inspired by these naïve but fundamental questions, and I'll try, in the sequel, to position my own work with respect to some possible contemporary presentations of such questions.

The quantitative approach to phenomenons has always been a milestone in (both theoretical and applied) Science. During the twentieth century, the understanding and the use of computation has deeply influenced much research. Logic had a leading role in this evolution: recursion theory is one of the main ingredients in the birth and development of computer science. Later on, in the sixties, the correspondence between proofs and programs (λ-terms) was clearly established by the Curry-Howard isomorphism: a proof of intuitionistic logic is a program, and the execution of this program corresponds to applying the cut-elimination procedure to the proof. This discovery, simple and fundamental, led to a new approach to the above mentioned questions. Gentzen already implicitly suggested that consistency could be related to a process of transformation of proofs: when such a process terminates whatever proof it is applied to, the system is consistent. The Curry-Howard correspondence refines this suggestion going one step further: such a transformation process is (in a precise sense) a computation, so that consistency is related to the termination of some computation. Also, proofs become computational processes and, in order to understand what is a proof π, one can try to to understand what is the computational meaning of the behaviour of π with respect to cut-elimination.

It certainly makes sense to argue that a proof is not *only* a computational process, but it is difficult to deny that the computational point of view on proofs had striking consequences in proof-theory. Indeed, the quest of mathematical invariants of the computational process (cut-elimination in proof-theory, execution of programs in computer science) led to a now wide and well established field of research: denotational semantics, which associates with every formula A a mathematical structure $\mathcal{A}$ and with every proof π of A an element $[\![\pi]\!]$ of $\mathcal{A}$ (called the interpretation of π), in such a way that if π' is a proof of A obtained from π by performing some steps of cut-elimination, then with π and π' is associated the same element of $\mathcal{A}$ (they have the same interpretation: $[\![\pi]\!] = [\![\pi']\!]$). In the eighties, Jean-Yves Girard wanted to define a denotational semantics of System F, a (second order) logical system he introduced 15 years before in order to prove Takeuti's conjecture, which implies the consistency of second order arithmetic. In so doing, Girard introduced the structure of coherent space as a simplification of Scott's denotational model for intuitionistic logic, and discovered that the intuitionistic connectives can be (mathematically) decomposed. Most notably, he realized that this decomposition can be internalized, i.e. it has a logical meaning. The equality[1] $A \rightarrow B = !A \multimap B$ gives birth in [1] to linear logic (LL), a refinement of intuitionistic and classical logic characterized by the splitting of standard connectives ("and" and "or") in two classes (additive and multiplicative), and by the introduction of new connectives (exponentials) which give a *logical* status to the operations of erasing and copying (corresponding to the "structural rules" of intuitionistic and classical logic). This change of viewpoint had a lot of consequences, among which one of the most important is the introduction of proof-nets, a geometric way of representing proofs and computations. From the beginning of proof-theory, the representation of proofs has quite evolved: a proof was first a sequence of formulas (Hilbert), then a tree (Gentzen), and now a graph (Girard's proof-nets). Proof-nets can thus be understood as well-structured objects, allowing a much sharper analysis and deeper understanding of proofs. Thus, LL *is not* another exotic logic, but a canonical object, with a well structured proof-theory, which carries new concepts. After its discovery, LL has shown its relevance for several topics, like the theory of programming languages, implicit complexity, parallelism and concurrency, games and languages, proof-theory, philosophy, categories and algebra, linguistics,. . .

Starting from the second half of the eigthies, a wide international community grew up thanks to various EU projects, like "Typed Lambda-Calculus" (Stimulation programme) from 1991 to 1996, several "Types" Esprit projects since 1989. Linear Logic was among the main research themes developed by these networks. From 1998 until 2002, a TMR research project centered on LL ("Linear Logic in Computer Science") was funded by the EU, with the participation of the most important research groups in Europe in the field at that time. The official sites were: Institut de Mathématiques de Luminy (Marseille), Università di Bologna, University of Cambridge, University of Edinburgh, Universidade de Lisboa, Université Paris 7, Università Roma Tre. Several generations of researchers in Europe benefited from these research networks, and share now a common cultural background. From the beginning, the cooperation on LL has been particularly intensive between Italy and France, with a very strong impact on doctoral training (about 30 PhDs defended or currently being prepared under the joint supervision of italian and french researchers). Since in the meantime LL has become a research area rather than a research theme, I had the idea, together with my french colleague Thomas Ehrhard (CNRS-Université Paris 7), to create a so-called "Groupement de Recherche International" (G.D.R.I.) of the french *Centre National de la Recherche Scientifique (C.N.R.S.)*. A GDRI is not a research project, but rather an institutional structure federating people that are used to work together and do not belong to the same laboratory: we felt that this was precisely what our community needed, after so many years of common work in research and in training. Our proposal was accepted by the CNRS and is funded for 4 years by the CNRS and the italian *Istituto Nazionale di Alta Matematica (I.N.D.A.M.)*, starting from 2015. The official sites of the GDRI "Logique Linéaire" are: CNRS, INDAM, Université Paris 7, Aix-Marseille Université, École centrale de Marseille, École Normale Supérieure de Lyon, Université Lyon 1, Università Roma Tre, Università di Bologna, Università degli studi di Torino. The scientific coordinators

[1] The equality was first established between coherent spaces, and became later an equality between logical formulas.

are Thomas Ehrhard and myself. A quick presentation of the GDRI can be found on the web site of the CNRS:

`http://www.cnrs.fr/ins2i/spip.php?article1935`

Other informations can be found on the web site of the GDRI:

`http://linear-logic.org/en/`

Logic has always been at the crossroad of several disciplines: philosophy, mathematics, and more recently computer science. Referring to contemporary research, one sometimes schematically says that questions come from philosophy and answers use mathematical tools. This might be partially true, but my feeling is that things are much more entangled. The history of logic is scattered by evidences of this entanglement, like the previously described process leading to the birth of LL. People compute, that's a fact. To understand the nature of computation (a philosophical question), the pioneers of computability proposed a mathematical definition of computable function, leading to computers and computer science. It is reasonable to think that Gentzen did not care much of the computational content of its cut-elimination procedure, and the mathematical content of the Curry-Howard isomorphism is almost empty: it is a technically trivial remark, with highly non-trivial philosophical motivations, and highly non-trivial mathematical (and philosophical) consequences. Without Girard's technical work LL would have never been discovered, and on the other hand the understanding and mastering of the new conceptual tools brought by LL need philosophical competence.

In my own research experience, the interaction with philosophers, pushed me to be always extremely careful on the many implicit presuppositions hiding in my approach. Very often, I realised that the scientific surrounding paradigm had a much stronger influence on my work than I could imagine. In [2] and [3], for example, we stress the difficulties often hidden behind widely accepted conventions.

2 Research activity

Usually a publication of mines establishes some new results in a particular topic, and I tried to partition my contributions referring to four fundamental questions, which give the title to subsections. In every subsection, after the title and between brackets, one can find my works which (mainly) refer to that question. Three exceptions are the works written in italian: [4], [2], [3].

I wrote the paper [4] after a seminar held in the mathematics department of the university Roma Tor Vergata and several discussions with my friend and colleague Flavio D'Alessandro. He suggested me to write a small paper containing an informal presentation of some results that were not known from the majority of the italian philosophers, mathematicians and computer scientists.

I consider the book in two volumes ([2] and [3]) the outcome of a considerable effort. The aim was to give a complete presentation of the basic results of logic usually taught in a master course, but with an approach entirely different from the standard one, following a proof-theoretical route, and inspired by some of the more recent advances in logic (see also the reviews of MathSciNet and of Zentralblatt for more details). In both volumes, a long introduction presents the results and methods discussed in the rest of the book in a philosophical perspective.

2.1 The nature of (classical) proofs ([5],[6],[7],[8],[9])

For a long time, the Curry-Howard correspondence was restricted to intuitionistic logic. It seems that one of the reasons was some kind of confusion between the so-called "non constructive" aspects of classical logic and the impossibility of extracting a computational content from a classical proof. The situation radically changed at the beginning of the nineties, when Griffin ([10]) showed that in the programming language Scheme, the constant call/cc can be typed by Peirce's law ($((A \to B) \to A) \to A$). From the proof-theoretical point of view, the main difficulty in the extraction of the computational content from classical proofs lies in the fundamental non-determinism of Gentzen's procedure of cut-elimination for classical logic: to compute one has to make choices, and there does not seem to exist a canonical way to make them.

The decomposition of intuitionistic connectives provided by LL is still relevant at the computational level: the computational process underlying LL (linear cut-elimination[2]) is itself a decomposition of the computational process underlying intuitionistic logic (intuitionistic cut-elimination). Thus, LL allows a sharper analysis of the computational process in intuitionistic logic, and induces also to ask the question: "is it possible to compute with classical proofs?"

Apart from the obvious theoretical interest, to understand the computational mechanisms at work in classical logic has also a more practical purpose: to see if any new programming construct shows up in extending the Curry-Howard isomorphism to classical logic. A lot of work has been done in this field starting from from 1990 (by Krivine, Parigot, Girard, Coquand, Murthy, Felleisen, Danos, Joinet, Schellinx, Berardi, Barbanera, ...), and although much of this work is not based on it, LL seems to play a pre-eminent role: because it can be used as a looking-glass, on the one hand LL suggests a reasonable way to make choices in the cut-elimination procedure of classical logic, and on the other hand it organizes known solutions in a simple pattern that makes apparent the how and why of their making, thus acting as a unifying and clarifying tool (see [11] and [12]). Another approach was conducted by Jean-Louis Krivine: starting from the idea that a proof of a "mathematically interesting" theorem has a "computationally interesting" content, Krivine has worked on concrete proofs of well-known theorems trying to understand how the various programs (λ-terms) associated with these proofs behave (see for example [13]).

My personal contribution to the subject is contained in the papers [5],[6],[7],[8],[9]. With Jean-Baptiste Joinet and Harold Schellinx ([5],[6]), we proved that, provided one follows the suggestions given by LL in defining the procedure of cut-elimination in classical logic, every computation terminates (strong normalization property) and leads to a unique normal form (confluence property). These results are achieved for all the (rather general) ways of performing cut-elimination in Gentzen's LK suggested by the translations of classical into linear logic. With Myriam Quatrini and Olivier Laurent ([7],[8]), we precisely isolate the fragment of LL used by Girard in [11] (the polarized fragment), and exploiting the properties of focalization and reversion of linear proofs, we prove that the previously mentioned cut-elimination procedure for LK perfectly fits cut-elimination of polarized proof-nets; some more semantic considerations allow to recover Girard's LC ([11]) within Gentzen's LK. In [9], I try to give a computational meaning to the choices involved in the classical cut-elimination procedure. This is a rather unexplored issue: we now know that when performing cut-elimination in classical logic, it is possible to make choices in a "reasonable" way, but there is still no canonical way to make them. At first glance, this seems to indicate that we should try to consider a non-deterministic cut-elimination procedure for classical logic: a very interesting attempt in this direction is [14]. But using the linear looking-glass, one sees that these choices become choices in the translation of certain classical rules in LL, and this suggests that the computational object which lies "behind" a classical proof *is not* a linear proof but a superimposition of linear proofs, and that a classical cut-elimination step corresponds to one or several cut-elimination step(s) in the "linear slices" superimposed by the classical proof. It is technically difficult to make sense from these (a bit fanciful) remarks, and in [9] I just try to settle a framework where to state the problem in a clear way.

[2] Instead of "cut-elimination", one often uses the word "normalization", coming (through the Curry-Howard isomorphism) from the theory of the λ-calculus.

If none of the my recent published works directly deals with this topic, this is because I feel that a significant new advance in the field needs sharper tools than the ones we have right now. For example, there is no trace of the geometrization of computations in any of the systems proposed up to now to formalize classical proofs. If we could define a solid notion of "classical proof-net", then, following Krivine's idea, a proof of a "mathematically interesting" theorem should have a "geometrically interesting" structure. And we could even dream of classifying mathematical proofs by means of the geometric properties of their computational content. . .

2.2 From static to dynamic consistency ([15],[16],[17])

Traditionally, one says that an axiomatic theory T is inconsistent when there exists a formula A such that from T one can derive both A and $\neg A$. A significant part of the research activity of a logician of the first half of the nineteenth century was to prove that such or such other theory was consistent, i.e. one cannot prove both A and $\neg A$, for any formula A. It happened to me to hear a sentence like "Logicians are those people who spend their lives working on things that do not exist" (the reference is to the proof of $A \wedge \neg A$). Thanks to LL, we can now positively say that "proofs" of contradictions do exist, and they are of the most genuine interest. Recall the already mentioned crucial dynamic aspect of Gentzen's consistency "proof"[3] of first order Peano Arithmetic: the cut-elimination procedure allows to transform a proof into a proof without cuts. The Curry-Howard isomorphism corroborates the importance of this transformation process. LL reverses the order of priorities: first comes the cut-elimination process, then come the proofs. Indeed, the (logically) correct LL proofs (proof-nets) are particular cases of proof-structures, and the cut-elimination procedure can be applied to all proof-structures. There exist several ways to characterize, among proof-structures, the (logically) correct ones: such a property is called correctness criterion. One of these criterions is often referred to as "interactive": a proof-structure is a proof-net iff it "interacts well" with the environment (see for example [18]). The full completeness theorem of Girard's ludics ([19]) is an even stronger example of the same approach: there are a lot of "agents" (say paraproofs) that interact freely, some of them are winners the others (the great majority) are losers. Roughly speaking, a paraproof of A is a proof when, as an interacting agent, it is a winner: it has a "good" interactive behaviour with respect to every paraproof of $\neg A$. The interaction between a paraproof of A and a paraproof of $\neg A$ is the cut-elimination procedure applied to the paraproof of a contradiction (of the empty sequent in Gentzen's terminology). Another example of the crucial importance of contradictions comes from Girard's approach to the logical nature of bounded time complexity (see Subsection 2.4): in [4], I try to explain, in an informal way, how using Russell's paradox Girard could taylor the exponentials of his "light linear logic" ([20]). He uses to say, after this discovery, "Un paradoxe est une clef, dont on ne connaît pas la serrure" (which sounds less nice in my english translation: a paradox is a key of which we do not know the lock). Summing up, a modern approach to consistency is the study of the properties of cut-elimination, and specially of cut-elimination of "proofs" of contradictions!

My personal contribution to the subject is contained in the papers [15],[16],[17], where I obtain new results on LL cut-elimination. In [15] and [16], I concentrate on the additive connectives of LL: despite the fact that the additives do not have a satisfactory representation in proof-nets, I prove that cut-elimination enjoys a confluence property (any proof-net has a unique normal form) for a significant (second order) fragment of LL, even though it was known that such a property does not hold in the general case. In [17], we give the first complete proof of strong normalization for full second order LL: Girard's original proof

[3]The use of inverted commas on the word proof has here a different meaning than the one it has for the previous occurrence of the same word. In the first case, as explained in the sequel, it indicates that there can exist proof-structures of both A and $\neg A$, but at least one of the two is not logically correct. In the second case, it indicates that -in accordance with Gödel's incompleteness theorems- Gentzen uses the induction principle on the ordinal ε_0, a principle which is not available in first order Peano Arithmetic.

([1]) used a standardization theorem (also called conservation theorem in the theory of the λ-calculus) which was not proven. I believe that the reason why our proof could answer a question which remained open for many years is that we were able to use the very elegant idea of Gandy ([21]) to infer strong normalization (every computation terminates) from weak normalization (some computation terminates). Such a method yielded a conceptual clarification, since we could clearly distinguish the combinatorial part of the proof (our contribution: the standardization/conservation theorem) from the proof of consistency (weak normalization: essentially contained in [1], using Girard's reducibility candidates).

2.3 Identity of proofs ([22],[23],[24],[25],[26],[27])

I find it very exciting to attack this old and traditional question from an interactive point of view. My favourite tool to identify/distinguish proofs is denotational semantics. As already pointed out, it is through denotational semantics that LL was discovered, and more precisely through a back-and-forth from syntactic objects (representations of proofs) and semantic ones (functions or more generally morphisms of some categories). The categorical framework helps to keep in mind that our interpretation of proofs is compositional, and thus the interpretation of π is indeed intended to account for all the possible interactions (through cut-elimination) of π with its environment.

For proof-nets, but more generally for any syntax $\mathcal{S}$ endowed with some rewrite rules, there is a natural syntactic equivalence relation which identifies objects with the same computational behaviour: under "reasonable hypothesis", $\pi \sim \pi'$ if and only if there exists π'', a computation starting from π and another one starting from π' both ending in π''. On the other hand, every denotational model $\mathcal{D}$ of LL, but more generally of any syntax $\mathcal{S}$, induces a semantic equivalence relation: $\pi \sim_{\mathcal{D}} \pi'$ if and only if $[\![\pi]\!]_{\mathcal{D}} = [\![\pi']\!]_{\mathcal{D}}$ (that is π and π' have the same interpretation in the model $\mathcal{D}$). It is then very natural to ask whether or not the two equivalence relations coincide for the known denotational models of LL. When this is the case, we say that the model is injective: for π and π' cut-free (and with atomic axioms), if $[\![\pi]\!]_{\mathcal{D}} = [\![\pi']\!]_{\mathcal{D}}$, then $\pi = \pi'$. In categorical terms, injectivity corresponds to faithfulness of the interpretation functor from proof-nets (more generally from the syntax $\mathcal{S}$) to $\mathcal{D}$. In the framework of the λ-calculus, which can be seen as a (little) subsystem of LL, every denotational model is injective. Actually, these "separation" results are among the cornerstones of the whole theory of the λ-calculus: from Böhm's theorem in the pure setting ([28]) to Friedman and Statman's results in the typed one ([29],[30]).

Maybe a bit surprisingly, I was the first to precisely address and study this question for proof-nets ([31]), for three well-known models of LL: the coherent set-based model (that gave birth to LL), the coherent multiset-based model, and the relational model. The coherent multiset-based model ([11]) is a variant of the original coherent set-based model, while the relational model is obtained by "forgetting coherence": it turned out that LL could be interpreted in the category of sets and relations[4]. In the sequel, I will denote by $[\![\pi]\!]_{Rel}$ (resp. $[\![\pi]\!]_{Coh}$) the interpretation of π in the relational (resp. multiset-based coherent) model of LL. In [22] and [23], I show that both the original coherent model and its multiset variant are not injective for Multiplicative and Exponential Linear Logic ($MELL$). The fragment $MELL$ of LL was chosen both because of its expressive power (it obviously contains the λ-calculus but also the fragments of LL used to translate classical logic) and because the theory of proof-nets of $MELL$ was considered solid (in particular cut-elimination was known to be confluent and strongly normalizing). Intuitively, the (cut-free and with atomic axioms) proof-nets π and π' that I found, such that $[\![\pi]\!]_{Coh} = [\![\pi']\!]_{Coh}$ whereas $\pi \neq \pi'$, "interact" in the same way with any possible environment, which suggests that "the coherent model is right, syntax

[4] Notice that the standard interpretation of the typed λ-calculus in the category $\mathbf{Set}$, where the objects are sets and the morphsims are functions between sets, does not yield a denotational model of LL.

is wrong". Following the back-and-forth perspective, this phenomenon witnesses the fact that, despite the tremendous improvement represented by the syntax of proof-nets w.r.t. sequent calculus, it still needs some improvement to really capture the interactive nature of proofs. I believe these counterexamples are strongly related to the property of (non)-connectedness of the proof-nets, but if I could turn this intuition into the following conjecture, I could not prove it: the coherent multiset-based model is injective for $MELL$ without weakenings. Since the weakening rule is the unique responsible for the absence of connectedness in proof-nets, the proof of such a conjecture would be a strong evidence that connectedness is essential to grasp the interactive nature of proofs.

Still in [23], I proved that the coherent (multiset-based) model is injective for a significant fragment of LL (strictly wider than the λ-calculus) and I conjectured that the relational model is injective for whole $MELL$. In the LL community, this last conjecture received a much wider interest than the previous one, and there were many attempts to solve it by several students of mine and colleagues (see for example [32], [33], [34], [35],. . .). This is partly due to the fact that it is pleasant to have a positive result for full $MELL$, but mainly to the increasing importance of the relational model in the LL landscape. In particular, the works of Thomas Ehrhard on the differential extensions of the λ-calculus and LL constitute a spectacular improvement in the understanding of the relational model. In [36], the author introduced finiteness spaces, a denotational model of LL (and the λ-calculus) which interprets formulas by topological vector spaces and proofs by analytical functions: in this model the operations of differentiation and Taylor expansion make sense. Still through some back-and-forth between syntax and semantics, Ehrhard and Regnier ([37, 38, 39]) internalized these operations in the syntax and thus introduced the differential linear logic $DiLL_0$, allowing a more subtle analysis of the resources consumption during the cut-elimination process. At the syntactic level, the Taylor expansion decomposes a proof-net in a (generally infinite) formal sum of $DiLL_0$ proof-nets, each of which contains resources usable only a fixed number of times. It is easy to see, when π is a cut-free proof-net with atomic axioms, that the elements of π's Taylor expansion are equivalence classes of elements of the relation $[\![\pi]\!]_{Rel}$. This entails that two (cut-free and with atomic axioms) proof-nets have the same interpretation in the relational model iff they have the same Taylor expansion. Thus, if we can recover a cut-free proof-net with atomic axioms π from (one or several elements of) its Taylor expansion, then for every cut-free proof-net with atomic axioms π' one has that $[\![\pi]\!]_{Rel} = [\![\pi']\!]_{Rel}$ implies $\pi = \pi'$. In the light of the differential approach, it is clear (and well-known) that the element of order 1 in the Taylor expansion of a λ-term is enough to entirely determine the λ-term: if two λ-terms t_1 and t_2 have the same element of order 1 in their Taylor expansion, then $t_1 = t_2$ (a proof is in [27], but probably in many other papers). Some years ago, I supervised the post-doctoral staying in Roma Tre of Daniel de Carvalho, and our common work led to a new result ([24]): given two *connected* $MELL$ proof-nets π_1 and π_2, if there exists an appropriate $DiLL_0$ proof-net, whose order depends on π_1 and π_2, which occurs in the Taylor expansions of both π_1 and π_2, then $\pi_1 = \pi_2$. In [25], we could improve this result: given two *connected* $MELL$ proof-nets π_1 and π_2, if π_1 and π_2 have the same element of order 2 in their Taylor expansions, then $\pi_1 = \pi_2$. In [40], de Carvalho proved the full conjecture: given two proof-nets π_1 and π_2, if there exists an appropriate $DiLL_0$ proof-net, whose order depends on π_1 and π_2, which occurs in the Taylor expansions of both π_1 and π_2, then $\pi_1 = \pi_2$. These results suggest the following interesting hierarchy, where the order of the element of the Taylor expansion allowing to distinguish different proof-nets is a measure of the amount of information which is enough to separate different computational behaviours:

1. full $MELL$, for which there does not seem to be a way to bound in advance the complexity of the element of the Taylor expansion allowing to distinguish two different proof-nets;
2. connected $MELL$ (containing the λ-calculus) for which the element of order 2 of the Taylor expansion of a proof-net is enough to entirely determine the proof-net;

3. the λ-calculus, for which the element of order 1 of the Taylor expansion of a λ-term is enough to entirely determine the λ-term.

The work [27] can be seen as an evidence of the power of the injectivity property. As already mentioned (Section 2.2), the original representation of proofs as proof-nets is unsatisfactory is presence of the additives, which is one of the reasons for the difficulty of the proof of the strong normalization theorem ([17]). Since the birth of LL, computing with the additives has always been a challenge. In [27], we introduce a notion of sliced proof-net for the polarized fragment of linear logic. We prove that this notion yields computational objects, sequentializable in the absence of cuts. We then show how the injectivity property of denotational semantics guarantees the "canonicity" of sliced proof-nets, and prove injectivity for the fragment of polarized linear logic corresponding to simply typed lambda-calculus with pairing.

In the attempt to stick as close as possible to the interactive nature of proofs, in [31] and [26], I modify the syntax of proof-nets introducing the notion of "additive multibox", internalizing in the syntax the reversibility property of the additive conjuction $\&$. A similar idea was later used in the syntax of Girard's ludics ([19]).

In the near future, I believe that we can expect to establish a precise connection between the connectedness of proof-nets and their "good" interactive behaviour.

2.4 Time in logic ([41], [42],[43],[44])

"Time goes by" and "nothing stands the test of time", people use to say (and sing). However, as a matter of fact, we often implicitly pretend that time does not exist: in mathematics a theorem is eternal, and the laws of logic are established once for all and forever. Of course this is debatable, a debate as captivating as difficult, and I do not dare entering into it here. What I want to point out is that the computational approach to proof-theory let time slowly slip into logic. I think logic has to be grateful to computer science for having transformed the original question that gave birth to recursion theory (and thus to computer science itself) "what is computable?" into its modern version "what is *really* computable?", leading to the research area called computational complexity. The concrete need of computer science to answer questions in reasonable time and space led proof-theorists to wonder if there exists a logical structure behind time-bounded or/and space-bounded computations[5]. Through the Curry-Howard correspondence, the question amounts to characterizing, within logical proofs, the ones for which the procedure of cut-elimination terminates in bounded time/space. Here LL, and more precisely the geometric representation of proofs as proof-nets, has a clear advantage: while in other systems one would try to limit the use of the cut rule, in LL one can hope to find a geometric parameter related to bounded complexity, with no need to restrict communication between proofs (the cut rule). This is the breakthrough achieved in [20]: roughly speaking, when the *depth* of a proof-net remains constant, meaning that cut-elimination is performed depth-by-depth, it is possible to keep control on the complexity of the procedure. Actually, proof-nets are structures more complex than graphs: boxes are a way to identify subgraphs that can be duplicated or erased during cut-elimination, and the depth of a node is the maximum number of nested boxes containing the node. The cut-elimination procedure applied to the proof-net corresponding to Russell's paradox (in LL extended with the full comprehension scheme) is cyclic, and Girard noticed that this is tightly related to the change of depth of the proof-nets during cut-elimination. He then proves that suitable restrictions allow to identify a system such that the functions "representable" in it are exactly the functions computable in polynomial time by a (deterministic) Turing machine. This characterization of a complexity class, independent from any reference to Turing machines, is typical of

[5]We use in the sequel the generic expression "bounded complexity" when we do not want to specify whether we speak of time-bounded complexity or space-bounded complexity.

the research area called "Implicit Computational Complexity"; and as such [20] has been the starting point of several works of (theoretical) computer scientists working in the field. From the logical point of view, [20] draws the main "boulevards" of a logical approach to complexity, based on the idea that the expressive power of a logical system is the complexity of its cut-elimination procedure. Notice, by the way, that the above mentioned sentence ("Un paradoxe est une clef, dont on ne connaît pas la serrure") perfectly suits Girard's methodology: with Russell's paradox is associated an infinite reduction sequence and inconsistency can be though as an upper bound to arbitrary big complexity. The paradox allows then to identify the good invariant (depth), and opens the lock of a logical approach to bounded complexity. Fascinating.

The technical part of [20] can be simplified, as noticed by several authors, and Girard's solution appeared quickly to be one among the possible ones (for some alternatives, see for example [45],[46],[47],[48],[49],[50],...). Many questions and problems arose from [20], one of the main ones being the search of a denotational semantics suitable for LLL. Since I found very elegant the (geometric) invariant found by Girard (depth), I tried to find a semantic counterpart of this invariant. In [41], we use the tools developed in [23] and give a semantic characterization of time-bounded complexity proofs. We introduce the notion of obsessional clique in the relational model of LL and show that restricting the morphisms of the category of sets and relations to obsessional cliques yields models of ELL and SLL, two logical systems allowing to represent, respectively, exactly the functions computable in elementary and polynomial time. Conversely, we prove that these models are relatively complete: an LL proof whose interpretation is an obsessional clique is always an ELL/SLL proof. In [42], we study in an abstract setting the property of stratification: in [20] strata coincide with depths, and bounded complexity is related to perform cut-elimination at a fixed depth (stratum by stratum). We show that stratification may be formulated independently of depth; when it is forced to be connected to depth, it yields interesting complexity properties.

A different and much more concrete approach to the semantics of time-bounded complexity is possible: the basic idea is to measure by semantic means the complexity of the cut-elimination procedure applied to any proof, regardless to its computational complexity. The aim is to compare different computational behaviours and to learn afterwards something on the very nature of bounded complexity. In [51], Daniel de Carvalho could compute the execution time of an untyped λ-term from its interpretation in the relational model. Since the interpretation of λ-terms in such a model is a particular case of the interpretation of LL proof-nets, we had the idea to apply this technique to LL, and we could in such a way adapt to the already mentioned fragment $MELL$ of LL some classical "qualitative" results well-known in the restricted setting of the λ-calculus, and prove quantitative results for untyped $MELL$ proof-nets. More precisely, in [43] we give a semantic account of the execution time (i.e. the number of cut-elimination steps leading to the cut free form) of an untyped $MELL$ proof-net. We first prove that: 1) a proof-net is head-normalizable (i.e. normalizable at depth 0) if and only if its interpretation in the relational model is not empty and 2) a proof-net is weakly normalizable if and only if its exhaustive interpretation (a suitable restriction of its interpretation) is not empty. We then give a semantic measure of execution time: we prove that we can compute the number of cut-elimination steps leading to a cut free normal form of the proof-net obtained by connecting two cut free proof-nets by means of a cut link, from the interpretations of the two cut free proof-nets. In [44], we showed that this approach can be applied to study strong normalization too: we prove that given two cut free untyped proof-nets of $MELL$, by means of their interpretations in the relational model, one can: 1) first determine whether or not the proof-net obtained by cutting the two proof-nets is strongly normalizable 2) then (in case it is strongly normalizable) compute the maximal length of the reduction sequences starting from that proof-net. As a rather nice by-product, we obtain an easy proof of strong normalization of typed $MELL$ which, instead of using confluence, relies on the invariance of the interpretation w.r.t. cut-elimination.

In both cases of weak and strong normalization ([43] and [44]), we use the crucial notion of experiment, introduced in [1] but extensively used only much later (after [31]). Experiments correspond to type derivations in the tradition of Intersection Types Systems, an approach to the λ-calculus introduced and studied in Torino since the seventies (see for example [52],[53],[54],[55],[56]). And indeed, in the restricted framework of the λ-calculus, we already mentioned the similar results obtained for weak normalization (our source of inspiration: [51]) while for strong normalization similar results are presented in [57]. Type derivations are a way to label tree-like derivations (in the Gentzen style), while experiments are a way to label graphs. Daniel de Carvalho and I are currently working on a natural improvement of the results obtained in [43] and [44]: exploiting the non arborescent structure of proof-nets, one should strongly reduce the amount of information used to predict the length of the computational process, a fact which would immediately yield a similar result for the λ-calculus too.

3 Training activity

If it goes without saying that the training of PhD students is very close to the research activity, I want to mention here that in my experience this has often been the case for master courses too, as witnessed by the content of the already mentioned book ([2],[3]). The effort we tried to make in order to constantly motivate and criticise the ideas, the methods and the techniques we use, has taught me a lot on the basis of my own discipline.

3.1 Training of researchers

Vito Michele Abrusci has been one among the main promoters of Linear Logic in Italy, since its birth in 1986. As soon as he arrived in Roma Tre (1996), he tried to spread these new ideas in several PhD courses. We met during my PhD studies in Paris, and when he proposed me to join him in Roma Tre (1999), it was clear to both of us that one of the main issues was the training of young researchers. In Roma Tre, we now have a well-established and truly interdisciplinary experience of PhD training in logic. Since when I was hired as a post-doc (2001) until the end of 2012, all the members of our research group were in the philosophy department, and we moved to the department of "Matematica e Fisica" in January 2013. This actually did not change much in the PhD training: since the beginning, we had PhD students with a mathematical background, others with a philosophical background. Our research group has always dedicated a lot of attention and energy to the training of young researchers. From 2006, we had 15 PhD students in logic in Roma Tre, the majority of whom prepared joint thesis ("cotutelles" with french Universities): Damiano Mazza, Michele Pagani, Gabriele Pulcini, Paolo Di Giamberardino, Maria Teresa Medaglia, Paolo Tranquilli, Giulia Frezza, Antonio Mosca, Marco Romano, Mattia Petrolo, Giulio Guerrieri, Andrei Dorman, Paolo Pistone, Arnaud Valence and Stefano Del Vecchio. Two among these students are french (Dorman and Valence) and participated to our PhD for the specific expertise of our group. Some highlights of our PhD training activities are the following (more informations can be found on our web site: http://logica.uniroma3.it/):

- in 2001, we submitted a joint project of "Scuola dottorale/Ecole doctorale" with Université de la Méditerranée (Aix-Marseille 2), that was funded by Università italo-francese, in the framework of the Vinci program (Bando 2001), responsible: Paul Ruet (CNRS-Université de la Méditerranée). Thanks to this project we could pay the trips and staying of our students in Marseille for long periods;
- during the academic year 2002/03, we organized a PhD course having as title "Introduction to Linear Logic". It was a very successful course, with students coming from other italian universities (Siena in

particular) to follow our lessons;

- in 2004, in the framework of the Vinci program "cattedra De Giorgi-Venturi Bando 2004/05" of Università italo-francese, we submitted a project for a PhD course held by Jean-Yves Girard (CNRS-Université de la Méditerranée). Our project was funded and Girard could spend 3 months in Roma Tre. These three months of lecturing, together with the simultaneous composition of lectures notes, led Girard to write and publish a book: "The blind spot" ([58]);

- in april-may 2009, Roma Tre accepted our proposal to invite Giuseppe Longo (CNRS-Ecole Normale Supérieure Paris) using the funds dedicated to visiting professors and international teaching. The students could then benefit of a course on several aspects of incompleteness ("otto lezioni sull'incompletezza").

Like the other members of the group, I contributed to the training of all the students. Six among them chose to work on topics very close to my research interests (I was the official supervisor of three of them) and our interactions were more intensive:

- Damiano Mazza (now CNRS researcher in the logic group of Université Paris 13) worked on implicit computational complexity in his master thesis in Engineering (Roma Tre), that I supervised. He gave remarkable contributions to this research field (much after the end of his PhD, we wrote together the paper [42]) and he became a specialist of Lafont's interaction nets. He is now acknowledged by the international community for his expertise in his research field;

- Michele Pagani (now professor in the équipe "Preuves, Programmes et Systèmes" of Université Paris 7) was a student of Corrado Mangione (philosophy department, Università di Milano). During his PhD, he worked on proof-nets and denotational semantics. Our cooperation has been particularly intensive after his PhD defense ([17] and [43]). He then turned to Differential Linear Logic and Differential λ-calculus, of which he is now among the worldwide experts. After some years as Maître de Conférences in the logic group of Université Paris 13, he was hired as full professor in Paris 7;

- Paolo Di Giamberardino defended his master thesis in philosophy in Roma Tre under the supervision of Vito Michele Abrusci. For his master thesis, he worked on ludics, a topic he developed in his PhD too, but he also worked on proof-nets (including additives), which has been our main topic of discussion. He now works in Rome, in a private company;

- Paolo Tranquilli defended his master thesis in mathematics in Roma Tre under the supervision of Marco Pedicini. Tranquilli is a brilliant researcher, with the great ability of jumping with ease from abstract category theory to the most concrete implementations. During his PhD, that I supervised together with Antonio Bucciarelli (University Paris 7), he worked on additive proof-nets and differential nets, proving technically difficult results. He spent one year of post-doc in Ecole Normale Supérieure de Lyon and two years of post-doc in Bologna. He now works in Switzerland in a private company;

- Giulio Guerrieri defended his master thesis in philosophy in Roma Tre (under my supervision) and he also obtained a master degree in "Logique et Fondements de l'Informatique" in Paris 7. In his PhD, that I supervised together with Thomas Ehrhard (CNRS-University Paris 7), he studied the call-by-value λ-calculus in a Linear Logic perspective (mainly following the works of Thomas Ehrhard) and the injectivity property of the relational model through the Taylor expansion. After his PhD, we were able to give a more global definition of Taylor expansion of a proof-net, which was crucial in the presentation of the result contained in [25]. I believe this alternative point of view is among the main contributions of the paper. Giulio Guerrieri was Attaché Temporaire d'Enseignement et de Recherche for two years in Paris 7, then post-doc for one year, still in Paris 7 (Equipe PPS), later he had a post-doc position between Roma Tre and Aix-Marseille Université (Equipe de Logique de la Programmation), a post-doc position in Oxford (supervised by Luke Ong), and he has now a post-doc position in Bologna;

- Andrei Dorman obtained a master degree in "Logique et Fondements de l'Informatique" in Paris 7, and I met him during one of my staying in Paris. Since he was interested in Linear Logic and in our research group, I suggested him to apply for a PhD position. We started working on the LL approach to Implicit Computational Complexity, but he then turned his attention to concurrency and more precisely to three extensions of Lafont's interaction nets: multiport multiwire and multiruled interaction nets. It has been a pleasure to work with him, even if he decided to quit the research activity after his PhD. He now works in Paris in a private company.

Apart from the training of PhD students, as part of an international community which is used to frequent and very intensive exchanges, we had several post-doc students in Roma Tre during these years, and several pre-doc students. I was the referee of the PhD thesis of Daniel de Carvalho (defended in Marseille), and I later supervised his post-doc staying in Roma Tre. Our cooperation was very fruitful: [43],[24],[44]. Daniel was among the most interested researchers in the question of injectivity I addressed for LL, and he eventually solved one of the main conjectures on this topic ([40]). Among the recent pre-doc staying in Roma Tre, let me mention Luc Pellissier (Ecole Normale Supérieure Cachan), whose short staying was also very fruitful in the long run ([25]). The research projects funded by the italian research ministry (COFIN, PRIN,...) played for some years an important role in the training of young researchers: until the size of these projects needed not being too big, we could federate all the italian researchers on LL, and several generations of young researchers in Torino, Bologna, Udine, Verona (and Roma Tre) could meet on a regular basis and share their ideas.

Thanks to all these networks, it happened that during the PhD defense of Paolo Tranquilli on Differential nets, Giuseppe Longo was our guest in Roma Tre, and he could attend the defense. With his experience, he realized that something really new and interesting was happening around these research themes, and as editor in chief of the journal *Mathematical Structures in Computer Science*, he proposed me to take care of a special issue on these themes. I accepted, and even if it took quite some time to go through the whole process (from the call for papers to the collection of the final versions) a special issue having as title "Differential Linear Logic, Nets, and other quantitative and parallel approaches to proof-theory" is now available online and the printed version will soon appear; I hope it will become a reference for the researchers in the field. I find it interesting to conclude this section on training of researchers with this little story, illustrating the relevance, for my research activity, of the training activity.

3.2 Training of master students

During my PhD in Paris, I could benefit a lot from regular meetings in the framework of international projects. It was then very natural for me to first take part and then promote research networks and PhD training networks. After some years, I realized that it was possible to extend this practice to the lower level (master), for objective and subjective reasons: on the one hand in Europe the institution of joint/double degrees was (at least in principle) encouraged, and on the other hand we could rely on the familiarity we had with our french colleagues. Above all, I felt that a joint multidisciplinary master program in logic could be the answer to a(n implicit) request of the students. Indeed, after several years of lectures to a mixed audience of philosophers and mathematicians, it was clear that small groups of very motivated students attended our master courses, and these courses were not enough for the high level training in logic we aimed at reaching. In 2009, my colleague Paul Ruet and I prepared a proposal of double degree in logic between Université de la Méditerranée (now Aix-Marseille Université) and Università Roma Tre (mathematics or[6] philosophy in Roma and mathematics in Marseille). The project was approved by the two institutions in 2010 and funded by Università italo-francese twice in the framework of the Vinci program (Bando 2009 and Bando 2013). Despite my long experience as Erasmus coordinator (since 2002), I must confess that, from the administrative point of view, it was a nightmare to set up this training program. I am nevertheless proud of this effort, because the students having followed it up to now (around 10) were of very high quality: half of them philosophers and half of them mathematicians, many of them came from other italian universties and matriculated at Roma Tre precisely to follow this program. The majority of the students having completed this program has defended (or is preparing) a PhD thesis in Italy or in France. By the way, since two diplomas are granted by two universities, there are two master thesis defenses, one for each institution, so that every student has two tutors, one in each country, with all the very positive consequences this has for students. More informations can be found on the site: http://logica.uniroma3.it/ tortora/CurriculumBinazLogica.html.

It occasionally happened to me to supervise students preparing their master thesis in other universities: several times with Roma La Sapienza, and a couple of times with foreign institutions (Université de la Méditerranée and Ecole Normale Supérieure de Cachan).

An outcome of the master training experience of Abrusci and myself is the logic book ([2],[3]). At first sight, the fact that the book is written in italian might appear a bit contradictory with the previously mentioned international training activity. A first reason is that our lecture notes were in italian, but another one is that the different approach we propose in the book was (initially) mainly addressed to the italian community. Not to speak of our pleasure to (sometimes) write in such a beautiful language! Anyway, Springer already proposed us to translate the book in english, and we plan to have (sooner or later) an english version.

References

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

[2] Vito Michele Abrusci and Lorenzo Tortora de Falco. *Logica. Volume 1. Dimostrazioni e modelli al primo ordine.* Milan: Springer, 2014.

[3] Vito Michele Abrusci and Lorenzo Tortora de Falco. *Logica. Volume 2: Incompletezza, teoria assiomatica degli insiemi.* Milan: Springer, 2018. to appear.

[4] Lorenzo Tortora de Falco. Sulla struttura logica del calcolo. *Rendiconti di Matematica e delle sue applicazioni*, Serie VII, vol. 26:367–404, 2006.

[5] Jean-Baptiste Joinet, Harold Schellinx, and Lorenzo Tortora de Falco. Strong normalization for all-style LK^{tq}. In *Proceedings of the conference Theorem proving with Analytic Tableaux and Related Methods, 5th International Workshop, TABLEAUX 96*, volume 1071 of *Lecture Notes in Artificial Intelligence*, pages 226–243. Springer, 1996.

[6] Jean-Baptiste Joinet, Harold Schellinx, and Lorenzo Tortora de Falco. SN and CR for free-style LK^{tq}: linear decorations and simulation of normalization. *Journal of Symbolic Logic*, 67 (1):162–196, 2002.

[6]This should be read as an *exclusive* or!

[7] Myriam Quatrini and Lorenzo Tortora de Falco. Polarisation des preuves classiques et renversement. *Compte-Rendu de l'Académie des Sciences de Paris*, 323:113–116, 1996.

[8] Olivier Laurent, Myriam Quatrini, and Lorenzo Tortora de Falco. Polarized and focalized linear and classical proofs. *Annals of Pure and Applied Logic*, 134, Issues 2-3:95–316, 2005.

[9] Lorenzo Tortora de Falco. Denotational semantics for polarized (but non-constrained) LK by means of the additives. In 5^{th} *Kurt Goedel Colloquium KGC'97, computational logic and proof theory*, volume 1289 of *Lecture Notes in Computer Science*, pages 290–304. Springer, 1997.

[10] Timothy Griffin. The formulae-as-types notion of control. In *Proceedings of the 17th ACM Symposium on Principles of Programming Languages (POPL)*, Association for Computing Machinery, pages 47–57, 1990.

[11] Jean-Yves Girard. A new constructive logic: Classical logic. *Math. Struct. Comput. Sci.*, 1(3):255–296, 1991.

[12] Vincent Danos, Jean-Baptiste Joinet, and Harold Schellinx. A new deconstructive logic: Linear logic. *J. Symb. Log.*, 62(3):755–807, 1997.

[13] Jean-Louis Krivine. Une preuve formelle et intuitionniste du théorème de complétude de la logique classique. *Bull. Symb. Log.*, 2(4):405–421, 1996.

[14] Franco Barbanera and Stefano Berardi. A symmetric lambda calculus for classical program extraction. *Inf. Comput.*, 125(2):103–117, art. no. 0025, 1996.

[15] Lorenzo Tortora de Falco. Generalized standardization lemma for the additives. In *Linear '96 Tokyo meeting*, volume 3 of *Electronic Notes in Theoretical Computer Science*. Springer, 1996.

[16] Lorenzo Tortora de Falco. Additives of linear logic and normalization- part I: a (restricted) church-rosser property. *Theoretical Computer Science*, 294/3:489–524, 2003.

[17] Michele Pagani and Lorenzo Tortora de Falco. Strong normalization property for second order linear logic. *Theoret. Comput. Sci.*, 411(2):410–444, 2010.

[18] Denis Bechet. Minimality of the correctness criterion for multiplicative proof nets. *Math. Struct. Comput. Sci.*, 8(6):543–558, 1998.

[19] Jean-Yves Girard. Locus solum: From the rules of logic to the logic of rules. *Math. Struct. Comput. Sci.*, 11(3):301–506, 2001.

[20] Jean-Yves Girard. Light linear logic. *Inf. Comput.*, 143(2):175–204, art. no. ic982700, 1998.

[21] Robin Oliver Gandy. Proofs of strong normalization. In *To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, Seldin, J., Hindley, J. (Eds.), pages 457–477. Academic Press Limited, 1990.

[22] Lorenzo Tortora de Falco. Coherent obsessional experiments for linear logic proof-nets. *Bulletin of Symbolic Logic*, 7, Issue 1:154, 2000.

[23] Lorenzo Tortora de Falco. Obsessional experiments for linear logic proof-nets. *Mathematical Structures in Computer Science*, 13, Issue 1:799–855, 2003.

[24] Daniel de Carvalho and Lorenzo Tortora de Falco. The relational model is injective for multiplicative exponential linear logic (without weakenings). *Ann. Pure Appl. Logic*, 163(9):1210–1236, 2012.

[25] Giulio Guerrieri, Luc Pellissier, and Lorenzo Tortora de Falco. Computing connected proof(-structure)s from their taylor expansion. In *Proceedings of the conference "Formal Structures for Computation and Deduction" (FSCD 2016), Porto (Portugal), 22-25 june 2016*, Leibniz International Proceedings in Informatics (LIPIcs), 2016.

[26] Lorenzo Tortora de Falco. The additive multiboxes. *Annals of Pure and Applied Logic*, 120, Issue 1-3:65–102, 2003.

[27] Olivier Laurent and Lorenzo Tortora de Falco. Slicing polarized additive normalization. In Thomas Ehrhard, Jean-Yves Girard, Paul Ruet, and Philip Scott, editors, *Linear Logic in Computer Science*, volume 316 of *London Mathematical Society Lecture Note Series*, pages 247–282. Cambridge University Press, 2004.

[28] C. Böhm. Alcune proprietà delle forme β-η-normali nel λ- K-calcolo. *Pubblicazioni dell'Istituto per le Applicazioni del Calcolo*, 696, 1968.

[29] H. Friedman. Equality between functionals. In Rohit Parikh, editor, *Logic Colloquium*, volume 453 of *Lecture Notes in Mathematics*, pages 22–37. Springer Berlin, 1975.

[30] R. Statman. Completeness, invariance and λ-definability. *J. Symb. Logic*, 47(1):17–26, 1982.

[31] Lorenzo Tortora de Falco. *Réseaux, cohérence et expériences obsessionnelles*. Ph.D. thesis, Université Paris VII, January 2000.

[32] Pierre Boudes. Unifying static and dynamic denotational semantics. Technical report, Institut de Mathématiques de Luminy, 2004.

[33] Michele Pagani. Proofs, Denotational Semantics and Observational Equivalences in Multiplicative Linear Logic. *Mathematical Structures in Computer Science*, 17(2):341–359, 2007.

[34] D. Mazza and M. Pagani. The separation theorem for differential interaction nets. In *Proceedings of the 14th LPAR International Conference*, volume 4790 of *LNAI*. Springer, 2007.

[35] M. Pagani and C. Tasson. The Taylor Expansion Inverse Problem in Linear Logic. In *Proceedings of the 24th Annual IEEE Symposium on Logic in Computer Science*, 2009.

[36] T. Ehrhard. Finiteness spaces. *Math. Struct. Comp. Science*, 15(4):615–646, 2005.

[37] T. Ehrhard and L. Regnier. The differential lambda-calculus. *Theoretical Computer Science*, 309(1-3):1–41, 2003.

[38] T. Ehrhard and L. Regnier. Differential interaction nets. *Theoretical Computer Science*, 364(2):166–195, 2006.

[39] T. Ehrhard and L. Regnier. Uniformity and the Taylor expansion of ordinary lambda-terms. *Theoretical Computer Science*, 403(2-3):347–372, 2008.

[40] Daniel de Carvalho. The relational model is injective for multiplicative exponential linear logic. In *Computer science logic 2016*, volume 62 of *LIPIcs. Leibniz Int. Proc. Inform.*, pages Art. No. 41, 19. Schloss Dagstuhl. Leibniz-Zent. Inform., Wadern, 2016.

[41] Olivier Laurent and Lorenzo Tortora de Falco. Obsessional cliques: a semantic characterization of bounded time complexity. In *Proceedings of the twenty-first annual IEEE symposium on Logic In Computer Science (LICS '06)*, 2006.

[42] Pierre Boudes, Damiano Mazza, and Lorenzo Tortora de Falco. An abstract approach to stratification in linear logic. *Information and Computation*, 241:32–61, 2015.

[43] D. de Carvalho, M. Pagani, and L. Tortora de Falco. A semantic measure of the execution time in linear logic. *Theoret. Comput. Sci.*, 412(20):1884–1902, 2011.

[44] Daniel de Carvalho and Lorenzo Tortora de Falco. A semantic account of strong normalization in linear logic. *Information and Computation*, 248:104–129, 2016.

[45] A. Asperti. Light affine logic. In *Thirteenth annual IEEE symposium on logic in computer science. Held in Indianapolis, IN, USA, June 21–24, 1998*, pages 300–308. Los Alamitos, CA: IEEE Computer Society, 1998.

[46] Vincent Danos and Jean-Baptiste Joinet. Linear logic and elementary time. *Information and Computation*, 183(1):123–137, May 2003.

[47] Yves Lafont. Soft linear logic and polynomial time. *Theoretical Computer Science*, 318(1–2):163–180, June 2004.

[48] Damiano Mazza. Linear logic and polynomial time. *Math. Struct. Comput. Sci.*, 16(6):947–988, 2006.

[49] Kazushige Terui. *Light Logic and Polynomial Time Computation*. PhD thesis, Keio University, March 2002.

[50] Kazushige Terui. Light affine lambda calculus and polynomial time strong normalization. *Archive for Mathematical Logic*, 46(3-4):253–280, 2007.

[51] Daniel de Carvalho. *Sémantiques de la logique linéaire et temps de calcul*. Thèse de doctorat, Université Aix-Marseille II, 2007.

[52] M. Dezani-Ciancaglini. A type theory for $\lambda - \beta$-normal forms. In *Proceedings of the Symposium Informatica 75, Bled Slovenia*. 1975.

[53] M. Coppo and M. Dezani-Ciancaglini. A new type assignment for λ-terms. *Arch. Math. Logik Grundlag.*, 19(3-4):139–156, 1978/79.

[54] Henk Barendregt, Mario Coppo, and Mariangiola Dezani-Ciancaglini. A filter lambda model and the completeness of type assignment. *J. Symbolic Logic*, 48(4):931–940 (1984), 1983.

[55] S. Ronchi Della Rocca and B. Venneri. Principal type schemes for an extended type theory. *Theoret. Comput. Sci.*, 28(1-2):151–169, 1984.

[56] Simona Ronchi Della Rocca. Principal type scheme and unification for intersection type discipline. *Theoret. Comput. Sci.*, 59(1-2):181–209, 1988. International Joint Conference on Theory and Practice of Software Development (Pisa, 1987).

[57] Alexis Bernadet and Stéphane Jean Lengrand. Non-idempotent intersection types and strong normalisation. *Log. Methods Comput. Sci.*, 9(4):46, 2013.

[58] Jean-Yves Girard. *The blind spot*. European Mathematical Society (EMS), Zürich, 2011. Lectures on logic.