From Why to How: proof theory since 1950*

JY Girard
tr. T Waring

December 28, 2021

> At the end of the last century, mathematics was gravely menaced by paradoxes; but proof theory was able to return meaning to the traditional terms of reference.

This imaginary quote sums up the ideology of the average proof theorist in 1950. They immediately situate proof theory as a question of fundamentals (elimination of paradoxes) which affirms that the logician gives the deepest meaning to mathematics: this I will call the "why". Later, around 1985, computer science promotes a more pragmatic approach, which I will call the "how": this *how* is a much less noble preoccupation than the *why*, but which demands far more subtle techniques[1].

The domain is organised around the split between a conservative wing, ideologically in decline (while still influential) and an innovative and (sometimes overly) pragmatic wing... this split is not new: its lineaments can be found in the opposition over foundations between Hilbert and Brouwer[2] in the 1920s, a controversy which culminated in the expulsion of Brouwer from the editing of the *Mathematische Annalen*, see [vD90].

1 Prehistory of the why

1.1 Problems of foundations

Set theory is primarily worthwhile for its capacity to unify: it fundamentally announces the unity of mathematics[3]. The utility of this unity is the progressive construction of the rationals, the reals, etc. in terms of sets, which can in their turn be considered as cardinals. This does not really mean that a real number is a set of rationals, and even less a subset of $\mathbb{N}$: the real novelty is the possibility of combining freely all aspects of mathematical reasoning... after all, why else can we count on analytic and algebraic methods in number theory to lead to the same results.

*A translation of Girard's note [Gir00].

[1] Continental France is connected, *because* one can link any city to Paris; but not to dismiss the question of *"how" is France connected* demands the construction of a communication network much less trivial than a web centred on Paris.

[2] Although Brouwer was not strictly a logician, much less a proof theorist, his *intuitionism* appears now as an ancestor of the *how*.

[3] As opposed to physics, formed of islands linked by hazardous bridges.

At the end of the last century, *naive* set theory essentially comes down to the *comprehension schema*, ie the fact that every property A defines a set:

$$\exists X \forall x \, (x \in X \iff A[x]). \tag{1}$$

When, in 1897, Burali-Forti [BF97a, BF97b, vH67] published their paradox[4], it only resulted in a vague menace — not on mathematics as a whole, but on the bridges between different parts of mathematics. And elsewhere, it only took 10 years (Zermelo 1908, see [vH67]) to note that a restriction — all-in-all empirical — of the comprehension schema to elements of a pre-existing set Y

$$\exists X \forall x \, (x \in X \iff A[x] \wedge x \in Y), \tag{2}$$

plus some other particular cases of eq. (1) (power sets, the set of natural numbers) as well as the *Axiom of Choice* permitted one to realise the unity of mathematics without apparent contradiction. 90 years of mathematics formalised inside[5] Zermelo's theory has confirmed the viability of those restrictions.

So, why this great fear of the year 1900? How worried was Hilbert, *reall*, when he stood up to save the edifice of mathematics, or was he using a pretend danger to promote an extreme form of scientism, namely *formalism.* In 1900 — fifteen years before mustard gas — science had only revealed its appearance to Jules Verne, and *scientific* faith knew no bounds. From this came the idea to found mathematics once and for all; by the nature of the scientist *credo* this foundation could only be mathematical, and to avoid auto-justification, it must take the form of a *reduction* of abstract mathematics (set theory) to a very elementary corpus of mathematics (the arithmetic of the ancients).

The so-called *Hilbert Program* traces back to his famous list of problems of 1900, and after a first attempt in 1904 [Hil05, vH67], it takes shape in the 1920s, see [Hil26, vH67]. We can give two versions:

- The demonstrations of consistency: prove that a formal system does not give rise to a contradiction.

- The demonstrations of conservation: prove that a formal system demonstrates no more than the elementary results of the methods *"à la" Papa.*

In addition, only elementary proof methods should be used in establishing one or another of these goals.

By such elementary methods, it is easy to see that these two versions are equivalent: thus consistency is only conservation with respect to an elementary inconsistency such as $0 = 1$. Furthermore, the aspect of "conservation" extends a tradition in number theory: to give elementary proofs, eg to eliminate complex analysis from the prime number theorem. Under this form, Hilbert proposed the conjecture of "purity of methods".

[4] Call an *ordinal* a set which is well-ordered by the relation of belonging; the *set* of ordinals is itself an ordinal, so must belong to itself, and as such is not well-ordered.

[5] Only set theory, by its nature, escapes this limitation.

1.2 The refusal of realism

Foundations force us to respond to the frightening question: when I say that A is true, what does that mean? *Truth* is in fact a notion which rebels against all analysis, as it commutes with all logical operations ($A \wedge B$ is true if A and B are, $\neg A$ is true if A is not, etc[6]) and in particular all justification of the principles of mathematics by their truth is greatly suspect. To convince ourselves of this fact, consider Peano arithmetic, which contains little other — besides visibly true principles of logic — than the schema of demonstration by recurrence, called the *schema of induction*; then, if $A[0]$ is true, and $A[n] \implies A[n+1]$ is true for every n, we are convinced that $A[n]$ is true for every n... but precisely, we have justified induction on A by induction on the truth of A: around in a circle. In this way we are brought to mistrust realism, and, for example, to show that the equation in f, g:

$$f(g(x)) = f(f(x)) \tag{3}$$

does not lead to $f(x) = g(x)$, we will remark — rather than exhibiting functions f and g *ad hoc* — that the consequences of Equation (3) have f simultaneously on both sides.

Having been led to eliminate all reference to truth, one might try to replace them with *provability*. And it is true that the provability of A in a fixed formal system (arithmetic, set theory) has a different allure to its truth. The Hilbert program would have elsewhere justified this substitution completely, at least for propositions of a simple structure, Π^0_1 (see Section 1.3). It is, all the same, necessary to remark that the elimination of truth in texts of proof theory often attains the summits of... hypocrisy. Thus, when one studies Equation (3), one can indeed construct a domain of two elements, $0, 1$, and the functions $f(0) = f(1) = 1$, $g(0) = 0$, $g(1) = 1$ and a finite calculation of truth shows that Equation (3) is true: then $f(0) = g(0)$ has exactly the same epistemological value as induction on the construction of equations[7]. One must never name the devil, and in proof theory the devil is the external world.

1.3 Gödel's theorem

Without a doubt, Gödel made the confusion between truth and provability his own, and tried to obtain a contradiction in mathematics: as we know — in a system $\mathcal{T}$ containing a minimum of arithmetic — how to rigorously code and define demonstrability, we can (by a diagonalisation argument[8]) produce a proposition G which formally expresses its own non-provability, in other words, its falsity, which leads to a contradiction. Wait a minute... this follows provided that provability is identified with truth, which would make G a "Cretan statement"[9], ie "this sentence is a lie".

The possibility remains that truth is distinct from provability. At this moment we should be careful. One classes arithmetic propositions into Σ^0_n and Π^0_n: $n \geq 1$ bounds the number of alternating quantifiers of A, of which the first is given by the choice of symbol $\Sigma = \exists$ or $\Pi = \forall$. So Fermat's theorem, the Riemann hypothesis, and statements of consistency are Π^0_1, whereas $L(1, \chi) \neq 0$, and the provability of a proposition are Σ^0_1, and $\forall \chi L(1, \chi) \neq 0$ is Π^0_2. True Σ^0_1

[6] *Translator's note:* see Girard's memorable remark in [Gir11] on "brocolli logic".

[7] This behaviour is even found in Gentzen [Gen36b, Gen69]: a banal truth argument in a finite model is transcribed using cabalistic expressions, and aquires in that moment the stature of a proof of consistency.

[8] Used by Cantor to demonstrate that $\mathcal{P}(\mathbb{N})$ is not countable, and already evident in Russell's paradox, in 1901: see [vH67, pp. 124–125].

[9] *Translator's note:* as in the Epimenedes paradox.

statments are always provable (to demonstrate that $\exists n A(n)$ with A quantifier-free, it suffices to find n and calculate the truth of $A(n)$, which being done, formalises itself), and by duality, provable Π^0_1 statements in $\mathcal{T}$ are true as soon as $\mathcal{T}$ is consistent. The Gödel sentence is Π^0_1, and if it is provable, its provability (ie its negation) is provable as it is Σ^0_1 and true: that is to say, G and $\neg G$ are provable, which makes $\mathcal{T}$ contradictory. We have therefore demonstrated that G is not provable, which is to say G is true. This is the first *incompleteness theorem* (1931), which refutes the "conservation" form of the Hilbert program: in effect, take for $\mathcal{T}$ a system of arithmetic containing exactly the "elementary" methods, which is necessarily consistent... the proposition G follows from the consistency of $\mathcal{T}$ without being able to be obtained in $\mathcal{T}$, ie by elementary methods.

In fact, a little more work (but a lot more care), shows that G may be replaced with the consistency of $\mathcal{T}$: this is the second incompleteness theorem, obtained in process. It refutes the "consistency" version of the Hilbert program. This theorem has become a *tarte à la crème*, deliciously counter-employed[10]. It destroyed (or at least should have destroyed) any hope of realising the famous Hilbert program: to found $\mathcal{T}$ it would be necessary to use more than $\mathcal{T}$, which will never convince more than the true believers.

1.4 The refusal of evidence

The first incompleteness theorem forbids all forms of commutation of provability with negation, and inserts into systems of logic a "gap" between theorems (demonstrable propositions) and antitheorems (refutable propositions), a hole that cannot be "plugged" in a reasonable fashion[11]. We come across this situation in a number of algorithmic problems: so every algorithm which proposes to resolve Hilbert's 10th problem[12] will be either incomplete, or faulty (Matiasevič 1970). More abstractly, the halting problem for a program is not decidable, which is to say that given a program $P[\cdot]$, depending on a parameter n, there is not in general a program $Q[n]$ which can tell us whether $P[n]$ will halt or not.

Let us jump forwards 50 years: around 1980, certain specialists of artificial intelligence are upset that we cannot always respond with "yes" or "no" to every question, and propose that we complete logic so that we can respond in every case — all this in formal contradiction to the incompleteness theorem and its corollaries, that they either denied or were ignorant of. We thus witnessed the outbreak *ex nihilo* of a paralogic[13] which exists in perfect uselessness, but without disappearing completely, because of the pregnancy of a new type of scientism linked to a computer which — they think — would know how to repsond to all problems.

Thus is should not be surprising that Gödel's theorem was not understood at the time: the whole world, including Gödel, searched for escape routes allowing them to get around its difficulty.

1.5 Gentzen

The 1930s were dominated by two proof theorists, both of whom died prematurely: Herbrand (1908–1931) and Gentzen (1909–1945).

[10] And even the object of a best-seller of colossal vulgarity: *Gödel-Escher-Bach*.

[11] In the same way that we cannot "complete" an unbounded operator on a Hilbert space $\mathbb{H}$.

[12] The resolution of Diophantine equations

[13] Non-monotonic "logic", "logic" of defaults, etc.

1.5.1 Sequent calculus

Herbrand's theorem [Her31, Her71, vH67] anticipates Gentzen's theorem (often called the *Hauptsatz*[14]) of which it constitutes a "synthetic" version. Although Herbrand's theorem has certain advantages over Gentzen's result, it is much less flexible in its application, so we will focus on the *Hauptsatz*. This result of 1934, see [Gen35, Gen69], is the most perfect realisation of the Hilbert program: for example it is a result on purity of methods which shows (this is the *subformula property*) that to demonstrate a proposition A, one can restrict to the sub-propositions of A; moreover the result is demonstrated by finitary methods, ie elementary. The rub is that the result only applies to pure logic (ie the *propositional calculus*): it no longer works (or at least only with crutches) as soon as one adds the smallest amount of arithmetic, ie the axioms of induction. As is, the result only applies to the Hilbert program to prove the consistency of ridiculously weak formal systems.

Gentzen's theorem, see appendix B, dominates proof theory, which is far from understanding its full import. From a technical point of view, the rules of logic are rewritten as a *sequent calculus*, based on profound symmetries. A rule (called cut) permits this calculus a deductive flexibility: it expresses in adriot fashion the transitivity of logical consequence, which is to say the possibility of using intermediate results (lemmas) in a proof. Gentzens produced an algorithm (cut elimination) which permits one to effectively replace a proof with a cut-free proof, that is, a practically direct and non-deductive proof. Such proofs do not exist in nature (although now days computers can recover them) for reasons of size and comprehension: since it is impossible to go down from the general to the particular, one demonstrates ten times the same property in similar cases, rather than demonstrating the root property from which these cases arise. These proofs has remarkable properties, which compensate for their artificial character. One can compare sequence calculus to Hamiltonian mechanics: the same insistence on symmetries, and the same inadaption to concrete problems compensated for by a remarkably lofty point of view.

1.5.2 Proof of consistency

In 1936 Gentzens attacks Peano arithmetic, a system which contains far more than "elementary" methods, and which is genuinely subject to Gödel's theorem. On the basis of his work on sequent calculus, he gives in 1938 [Gen38] a second proof of coherence... by transfinite induction up to the uncountable ordinal ϵ_0[15]. *Grosso modo*, it applies to imperfect cut-elimination theorems, but suffices to assure us of consistency. We remain confounded by the disparity between the ingeniousness of the techniques, and the weakness of the result, at least the "official" result: *"Gentzen established the consistency of induction up to ω by induction up to ϵ_0"*[16].

Gentzen's first consistency proof (1936) [Gen36a, Gen69], is even more unjustifiable from the point of view of foundations: there Gentzen develops nothing more than an interactive interpretation of proofs, seen as strategies for establishing the truth of formulae. This work violates one major rule (the notion of a winning strategy contains a truth predicate) which

[14] This German word means nothing more than the expression *fundamental theorem* used by Herbrand for his own result.

[15] The first fixed point of the function $x \mapsto \omega^x$.

[16] This pettiness due to an important French mathematician as related by Kreisel. We will have occasion to return to this point.

did not exist at the time... It is only with the era of the "how" and the abandonment of fundamentalism that we have found the correct perspective: it gives the first *ludic* interpretation of logic. Incidentally (we will not have the time to return to this) game theory becomes, at the end of the century, in association with linear logic, a major axis of proof theory.

2 The why

The continuation of Gentzen's work with a blind view towards consistency proofs dominated the subject up to — let us say — 1985. I will call this the theory of "why", see footnote 1. About this I propose that Gödel's theorem had rendered the protagonists more or less schizophrenic: for example, when in 1972 I naively declared to Schütte (responding to an opinion of Kreisel, see below) that a proof of the consistency of set theory would not have the least epistemological value, he responded *"okay, but I would still feel a lot better if I had seen one"*. It should also be noted that despite the incompleteness theorem and the fastidious character of "proofs of consistency", they always have an audience, who is without doubt looking for reassurance; thus we see computer scientists — otherwise reasonable — worrying themselves about proofs of consistency, and reacting like Schütte when one tells them that they prove nothing.

Of course, we should note the difference between consistency of set theory, which runs into Gödel's theorem, and the question of the consistency of the axiom of choice (resolved by Gödel in 1938). This result (like that of Cohen in 1963) does not collide in any fashion with the incompleteness theorem, insamuch as the consistency of set theory (without the axiom of choice) is assumed. It takes the form of an elementary reduction of the theory with the axiom of choice to the theory without that axiom (the reduction is done by an explicit translation of one into the other). In other words, Gödel's theorem opposes absolute demonstrations of consistency, but not *relative* coherence results.

2.1 Schütte and the Munich school

Around 1950, proof theory is essentially concentrated in Germany, let us mention Ackermann, Lorenzen and him who will (tardily) get to school, Schütte. We always show the consistencies of arithmetic by way of the eternal ϵ_0, but the methods are a little changed. Schütte permits infinite rules, of the form

$$\text{From } A[n] \text{ for } n = 0, 1, 2, \ldots \text{ deduce } \forall x A[x]$$

for which he demonstrates an excellent analogue of the *Hauptsatz* [Sch77], in which ordinals like ϵ_0 intervene to measure the height of proof trees.

Primarily this is progress on the hypocrisy which characterises certain papers of Gentzen. We have met the insistence of Gentzen on remaining finite (for example his ordinals are replaced by their Cantor normal forms, which are finite expressions), and in particular on avoiding any notion of infinite proof. It is, however, clear that Gentzen thought of his work in terms of infinite proofs, but he preferred to cover his tracks to protect his flanks from the accusation of the abandonment of finitude. Here it is easy to see that these "proofs" can be seen as countably branching proof trees, of which we can, commencing with the conclusion, describe effectively arbitrary finite portions: which means that the finitist dogmas are never violated.

Schütte's approach permits a notable simplification of proof theory, in particular it extends the subformula property to arithmetic, however on the basis of an infinitary logic. This machinery has been applied to extensions of Peano arithmetic to second order, where one allows quantification over sets of integers: such formulae are classed into Σ^1_n and Π^1_n, where here $n \geq 1$ counts alternating quantifications over sets. There is a multitude of such systems, distinguished from one-another by the logical complexity of the formulae A admitted into the comprehension schema (eq. (2)) — so we have always stayed below two quantifications over sets — and incidentally by the finer points of their principles of transfinite induction. Work of certain Americans (Feferman, Friedman, Howard, Tait) in the 1960s have permitted the writing of these diverse systems in the form of *inductive definitions*: by which one means that the sets in question are obtained by certain countable iterations. The first notable extension is due to Takeuti [Tak67] (1967), who treats comprehension over a set-wise quantifier (ie Π^1_1) by way of his own system of ordinals (*ordinal diagrams*), be it somewhat incomprehensible.

It is from the thesis of Jane Bridge (a synthesis of various other work, notably Bachmann and Aczel, 1972), see [Bri75], from which the Munich school will take their ordinal measure scale. Bridge's work (rebaptised as the "Buchholz system") permits Schütte's students: Pohlers, Buchholz, Jäger to analyse multiple inductive definitions in terms of various ordinals. This is a well-established technique, which proceeds without surpise, neither bad (it is correct) nor good (it is always *la même sauce*), see [BFPS81].

This activity continues, be it in a more modest fashion. It should of course be recalled that the fundamental objection which comes to us from Gödel is in no way bypassed by this work: thus, a bigger theory, a larger ordinal, etc. We have in this way seen the creation of catalogues relating more-or-less artificial formal theories with more-or-less gigantic ordinals... there is a certain occultism, numerology in this, as these results do not in general provide any interesting information. This is not entirely true for Gentzen's results, and their immediate extensions, which deal with "reasonable" ordinals: of this Friedman, see eg [Gal91], has remarked that the ordinal associated to a small extension of Peano is a good one[17] because of a famous theorem of Kruskal on the embeddings of finite trees, and since the incompleteness theorem forbids any proof of this theorem in that system.

Let it be said that the notion of infinite proofs poses problems: one must avoid the gag which consists of always postponing the beginning of a proof, that is to say the existence of infinite branches. It is precisely to avoid such branches that the proofs are embedded in an ordinal like ϵ_0. The consistency of these systems results then from the well-foundedness of the ordinals used. This foundation, the notion of provability in infinite logic, is expressed by Π^1_1 formulae: this class of formulae plays for infinite logic the role played by the class Σ^0_1 for finite logic. But this is not all: although we know how to approximate them effectively by arbitrarily large portions, and although we can embed them effetively into an ordinal, how can we be sure of all that? We require an auxillary finite proof of all of this... this gives rise to a lot of coding without a lot of imagination, and by the end we are left with an unfinished impression.

[17] This ordinal Γ_0 is presented as a linear order, and the question of is to know if it possesses a well-order; the well-ordering of Γ_0 follows from the consistency of theories of which the infinite proofs can be embedded in Γ_0.

2.2 Kreisel

Kreisal dominates this era of proof theory without contest. Strictly speaking, his active period finishes around 1970, but his influence persits for a good twenty years longer. His atitude fundamentally refuses or minimises the value of consistency proofs (a point of view which in hindsight seems like the only tenable one), retaining only the value they add, in other words *"What do they teach us?"*: there is, with Kreisal, already the beginnings of a passage to the *how*. Let us give some examples of his methodology:

- The most stupid of all consistency proofs *"The axioms of arithmetic are true, and the rules preserve truth"* is repudiated with horror by all specialists, as it justifies the axioms by themselves (which we believe in) and uses a truth predicate which is by its nature tautological and infinite. This does not rebut Kreisel, who of this only considers the formal aspect: he restricts himself to a finite number of induction axioms and so the subformula property allows him to *formalise* his argument in arithmetic itself, which thus proves the consistency of finite subsystems.By the incompleteness theorem this demonstrates that arithmetic cannot be finitely axiomatised, which one calls the *reflection schema*, see [KL68] or [Gir87, Chapter 4].

- All consistency proofs proceed by way of a cut-elimination algorithm (see Section 4) which therefore enumerates the input/output functions of the system associated to the Π^0_2 propositions $\forall n \exists m A[n,m]$. In other words, the proofs allows one to analyse the recursive functions (algorithms) of which the systems proves the termination: this is the subject of *provably total* functions: a consistency proof allows the characterisation of these functions, in particular their growth.

- In the same way, an ordinal consistency proof permits the bounding of recursive ordinals of which the system proves the well-foundedness: this is the subject of *provable ordinals*. Again information can be extracted from the proof.

Kreisel's critique has been more recovered than assimilated; thus articles in proof theory "à la Papa" no longer end with *"The theory $\mathcal{T}$ is consistent"*, but with *"The provably well-founded recursive ordinals of $\mathcal{T}$ are smaller than α"* where α is the ordinal used in the consistency proof. We shouldn't see in Kreisel a systematic opponant to Germanic proof theory, since at bottom he only dreamed of sitting the field on a less ideological basis. Thus he reprised the idea of finitist methods extended (this is already evident in Gentzen) by nuancing the cruel words relayed above (Footnote 16): induction up to ω in Peano arithmetic is permitted on propositions of arbitrary logical complexity, so long as we can restrict up to ϵ_0 the induction used in the proof of consistency for very elementary statements (without quantifiers).

Kreisel's insistence was above all the demand of more mathematical substance. He in fact commenced very early to effectively *apply* proof theory to proofs in number theory, eg to Littlwood's theorem concerning the sign changes of $\pi(x) - Li(x)$. The *a priori* possibility of this work comes from the logical complexity of Littlewood's theorem (Π^0_2): one of Kreisel's essential theoretical contributions [Kre51, Kre52] is the remark that classical Π^0_2 theorems are also intuitionistically demonstrable, ie have effective contents[18] (see 3.1). He had tried to train proof

[18] This becomes false for propositions of type *"The equation (E) has a finite number of solutions"*, which are Σ^0_2.

theory in the direction of applications to number theory (logical analysis of non-effective theorems), but without real success; we should remark that this type of application, based on explicit replacements inspired by Gentzen's theorem, easily becomes fastidious and incomprehensible, by saturation with explicit givens which usually mathematics has the good taste to hide. For example, if one is interested in analytic number theory, we hardly cleave to the problem of explicitly analytically continuing a function by way of a path covered by little circles, etc.

2.3 Dilators

This is an isolated experience (1975–1985), inspired by criques of Kreisel. In looking to put some mathematics into this jumble of ordinal encodings of various kinds, I remarked that not only are ordinals direct limits of integers (this is evident), but that most constructions commute with direct limits, which immediately gives them a finitary side, and also with fibred products, which gives normal-form results. The theory of *dilators* (functors of ordinals which preserve direct limits and fibre products) had essentially been developed by myself over 10 years (see [Gir81]), in the course of insisting on geometric aspects absent from the Schütte school (eg infinite proofs obtained as the direct limit of finite proofs etc.) [Gir84]. The implemented tools can without doubt treat the Π^1_2 case, inaccessible to Schütte's methods[19], but to what purpose? At the level where the difference manifests itself, mathematics is very rarified and barely found outside abstract set theory (with which dilators get along well). We ought to fall back on the concrete, the simple: geometric considerations at work in the theory of dilators engendered linear logic, a subject with far more central interest: thus ended the experiment, of which remains one beautiful result: the hierachy comparison theorem of [Gir81], which relates Hardy's inactive hierachy H_α to other hierachies of calculable functions: the index α appears as the direct limit of the $H_\alpha(n)$.

2.4 And now?

This type of proof theory has not disappeared; Schütte's sucessors continued to align lists of theories against a *Panderdivisionen* of ordinals without any breakthrough neither conceptual (explaining what happens) nor in logical complexity (Π^1_2 comprehension).

A certain rear guard is found to be retrained in the predicativist *préchi-précha*. Recall that the origin of predicativity is an idea of Poincaré (taken over by H. Weyl) for explaining paradoxes: we do not have the right to define an object in terms of a set which contains it[20]. In fact — Poincaré or no — this is an idea of remarkable sterility. Like the production of substitutions in proofs of consistency, one finds proposed predicative systems supposed to be "more safe" than our habitual ones. This has barely any sense, firstly as the doubt as to consistency should be either universal, or repudiated as doubtful[21], and further as the aformentioned non-predicative methods are above all a facility. Finally, these correct predicative theories are a little like a carrot diet, very boring, and at bottom do nothing good nor bad... one might consult [Fef75],

[19] The notion of proofs associated to arbitrarily large ordinals however entirely determined by the finite case is much more powerful that the case of ω considered by Schütte: one passes from complexity Π^1_1 to complexity Π^1_2, all while remaining much more "finite".

[20] Kreisel maliciously made the remark that the smallest integer satisfying some property is likewise defined in terms of a set which contains it.

[21] Kreisel: *"The doubts as to consistency are more doubtful than the consistency itself"*.

one of those articles where ideology is exercised at the expense of technical content, confined to a collection of banalities.

More recently, proof theorists more oriented towards algorithmic complexity have tried to revive traditional techniques (above all around arithmetic), such as Buss with his *bounded arithmetic* [Bus86]. This is an interesting opening and one that would be placed in the section on *how* if the methods were not so frozen. In particular, we would need a theory of arithmetic proof (bounded or not) with somewhat sharper tools; of course this is difficult, as everything seems to oppose a real cut-elimination theorem (finitary), but the doors on which we do not knock, do not open.

3 Prehistory of the how

Proof theory practically always consists of of a re-writing algorithm of which one demonstrates the termination (cut-elimination, normalisation, etc.). This terminations induces corollaries, such as consistency. This means that on this sole question, termination of an algorithm, concentrates all of the limitations linked to Gödel's theorem, and so a theorist obsessed with the *why* (proof of consistency) would be interested only in this termination. Now let us mourn once and for all the ambitions of the reductionists, fundamentalists, etc., and let us admit the well-foundedness of usual mathematics, in which we can show the termination of our algorithm[22]. So, our algorithm terminates, is that all? No, it can have properties, some of which are remarkable. To commence in studying the properties of the re-writing algorithm — in detail and not in force — is to pass to the *how*, see again Footnote 1.

Let it be said that the question on *how* is not recent in proof theory; but it is only since the 80s that it took precedence over the *why*, as a result of computer science, which overhauled certain questions. We then realised that the study of Gentzen's sequent calculus hat been neglected: we saw it as a module, a black box, necessary for the theory of infinite proofs. In examining it more closely, one discovers a world of incredible richness.

3.1 The constructive tradition

There is in number theory a tradition of *effective theorems*; so to know that a property holds for sufficiently large integers is much weaker than to have a estimate of the sizes of possible exceptions, especially in a time where a computer can check consequential initial segments. As such, the effective / non-effective difference is not superficial, it is a logical difference: one could say that it suffices to verify a property $P[n]$ up to an integer N, chosen so that it is true everywhere (take for N the first exception to P if there is one)... but this is a truism without interest and which has no effective content, observe that I could say this without any knowledge of P. The refusal of such "non-constructions" is the base of Brouwer's *intuitionism* and the resulting logical system, which — let us recall — does not admit the excluded middle $A \vee \neg A$.

Brouwer left us a remarkable idea: a proof is — or rather should be — a construction of an object. For example, he explains that a proof of $A \vee B$ should be a construction of an object which verifies A, or of an object which verifies B (from which we get the refutation of the excluded middle). This should not be interpreted in a stupid way: a proof of $A \vee B$ cannot

[22] The problem being to show this with limited methods.

be obtained by demonstrating one or the other: who — other than a cryptographer — would state $A \vee B$ if they had demonstrated A? Brouwer's requirement is that — in principle at least — one can transform a proof of $A \vee B$ into a demonstration of which is one of A or one of B. In the same way a method will be effective if it gives us a numerical value by default, a manner (algorithm) for obtaining it.

Despite the Don Camillo / Peppone–style opposition between the orientalist mystic Bouwer and the scientist Hilbert, these two points are view have been progresively brought together: in effect cut-free intuitionistic proofs (which are artificial creations and so by the proceeding common-sense remark should not apply[23]) are directly explicit. In summary, intuitionistic logic implicitly produces objects (eg numerical values), which cut-elimination can make explicit.

Under its constructiv*ist* (build a counter-mathematics) fizzled out (last sparks in the 60s with... Kreisel); by contrast constructiv*ity* — that is to say the use outside of any intuitionist sectarian spirit — remains current as we will see.

3.2 Lambda calculus

It is into constructivity that we should insert the technical invention of lambda calculus; there are two types of scientific errors — those which give us nothing and those which have unexpected posterity — and λ-calculus is one of the latter. Around 1930 Church, Curry, Kleene, Rosser, etc. had the absurd idea to construct a naive theory of functions (see appendix A), where every function can be applied to every other and where every expression defines a function. If we recall that a set can be viewed as a characteristic function, this should be seen as nothing more than a new version of naive set theory. And as such Russell's paradox (which is the construction of a fixed point for negation) transposes immediately and mechanically to a method for finding a fixed point for every function[24]. What saves λ-calculus from ridicule is that there is an escape route: the possibility that functions are partial. Lambda calculus becomes in this way an extremely flexible theory of partial algorithms; it is even universal in this respect. In spite of all of this lambda calculus remains a fairly marginal subject until the 60s, where it begins to acquire its letters of nobility, thanks in particular to Böhm, see [Bar84, Kri90].

4 The how

4.1 Around Prawitz

The mathematical activity of Dag Prawitz, philosopher by profession, so not very technical, only extends for a few years (around 1968). He ressurects the system (first considered by Gentzen) of *natural deduction*. In the intuitionist set, this systems works as well as sequent calculus; in fact much better, as if sequent calculus has a fairly messy cut-elimination algorithm (packed with arbitrary choices), natural deduction supresses inessential distinctions, in the manner of a natural quotient of sequent calculus. Therefore, natural deduction has a Church-Rosser property, ie gives a deterministic calculus.

[23] A cut-free intuitionistic proof of $A \vee B$ is a proof of A or a proof of B. It is therefore possible to commute provability and disjunction — contrarily to negation.

[24] The n^{th} avatar of Cantor's diagonal argument.

Relying on earlier work of Heyting, Kolmogorov (around 1930), Curry and Howard state in 1969 what we call the *Curry-Howard isomorphism*, see appedix A, between proofs in natural deduction and a variant of lambda-calculus ("simply typed"). It deals with much more that a bijection, since it transfers already-existing structures: so the intuitionist implication $A \implies B$ becomes the space of functions from A to B, conjunction $A \wedge B$ becomes the Cartesian product of A and B, the cut rule becomes the composition of functions and the normalisation of natural deduction becomes the formal calculus on functions.

In 1970 I introduced System F, see appendix A, which is both a natural deduction and a second-order lambda-calculus, see for example [GLT90]. We are no longer in the *why*: in effect, convergence of the calculi in System F formally implies the consistency of *all*[25] second-order arithmetic, but this termination result is only obtained with the aid of in no way elementary set-theoretic methods.

In the same circle of ideas, Martin-Löf created around 1972 his theory of types [ML84], which remains in the first order, while introducing other degrees of freedom.

4.2 Typed lambda calculus

It is only today that one see this work from a modern, non-reductionist viewpoint. In the reductionist view the termination of calculations in System F is an outcome without posterity. It is a compilation on System F which was to bring out the interest of computer scientists at the beginning of the 80s. In effect, this system, despite a restricted number (five) of primitives, permits the definitions of the "given types" (integers, lists, booleans, trees, etc.) by way of logical formulae. The implication between two given types $D \implies E$ is the type of total algorithms with inputs from D and outputs E. One sees that logical formulae play the role of specifications (which given type is accepted, what happens to it) and guarantees the absence of loops (termination). In addition, the presence of second-order objects gives the specifications a great flexibility: for example if $\mathrm{list}(\alpha)$ is the type corresponding to finite lists of objects of type α, the reversal of lists can be expressed without reference to α, as an object of type

$$\forall\alpha(\mathrm{list}(\alpha) \implies \mathrm{list}(\alpha)),$$

it is this that we call *polymorphism*. Finally, a forgetful functor associates to expressions in System F a lambda term (in the original lambda calculus of Church) which has the same algorithmic content. This deals with the development of the concept of *functional* programming, in opposition to the dominant paradigm of *imperative* programming[26].

In line with System F we have seen the birth of diverse typed lambda calculi, such as the *calculus of constructions*, due to Coquand [CH85], a computer-science oriented synthesis of System F with Martin-Löf type theory. The ambitions are clear: establish a powerful system of functional programming, where for the moment we only have pre-industrial products, such as COQ (developed around Gérard Huet at INRIA Rocquencourt). *Grosso modo* we propose to extract programs from mathematical proofs: one demostrates the existence of a solution to some problem, and we compile this proof into an algorithm for calculating the solution. The

[25] Recall again that all pseudo-effective methods using ordinals are restricted to complexity Π^1_2.

[26] Imperative programming uses instructions which indicate the actions to effect, here erase the register, and which do not correspond in any way to a functional calculus.

advantage of this method is that it produces programs which are mathematically guaranteed to be correct. There are, however, multiple problems in need of resolution:

- Firstly, the proofs in question must be constructive (intuitionist), which forces us to exist in a mathematical universe where certain very simple theorems are not yet verified. At least for Π^0_2 propositions we may yse the remark of Kreisel, but we then collide with arduous problems in the classical / intuitionist conversion[27].
- In addition, one needs to be able to formally write out a proof; this supposes the setting up of formal systems of abbreviations in every area[28] in which the computer works.
- Finally, one needs to extract a program; in principle the lambda-term associated to a given certified code.

All that is simplistic; so the program which extracts from the aforementioned proof that France is connected (see Footnote 1) is the SNCF network centre in Paris that we know... is not particularly effective. This is why one has a tendancy to complicate the schema above: nothing replaces a real algorithmic idea, and so it is proposed (in particular by J.-L. Krivine) to start with an algorithm (obtained by "pifomètre"), then to demonstrate that it answers the question and continue as above. The final lambda term is a sort of certified compilation of the original algorithm. One sees that the subject of typed lambda calculus is resolutely oriented away from fundamentalism, until it becomes (or at least furiously tries to become) applied.

Perhaps the theory does not quite find its account... we'll see; for example, Krivine's *storage operators* [Kri94] use certain intuitionist proofs so as to modify the execution type of an algorithm (eg force the evaluation of an argument before any calculation).

One must do a spell on the paradigm of *logic programming*[29]. From the beginning the idea is simple: *"state the program and PROLOG will do the rest"*. This is again the result of Gentzen, but exploited in an orthogonal way: one can reduce the idea of a calculuation to that of a proof (this is implicit in the proof of Gödel's theorem) in a very simple system which satisfies the *Hauptsatz*. We then pose the problem of proving certain formulae automatically, which is plausible as the sub-formula property dramatically reduces the search-space, see appendix B; on the other hand the logical aspect guarantees against any errors. There is an evident limitation: the algorithm will not alway converge, otherwise the question of provability would be decidable in contradiction to Gödel's theorem and its corollaries. There is above all technological evidence: one proposes a universal algorithm, an all-terrain vehicle which is fatally inferior to algorithms which use an idea[30]. It would have been wise to confine PROLOG to "exploration"-type algorithms, eg compiling police files... Alas, we wanted to give moustaches to logic, by adding to research "control instructions" one permits a programmer to use their cleverness. The result is that one does not control anything anymore from the logical point of view, and the

[27] A classical proof only defines an algorithm *modulo* a non-deterministic reduction to an intuitionist system; in this classical logic is never constructive, as the algorithmic content of proofs is not explicit in the demonstration.

[28] My strong-normalisation theorem for System F has been entirely machine-formalised by Berardi; the conceptual parts proved relatively easy, but by contrast the "obvious" parts have us a hard time.

[29] Which had its hour of glory with the enthusiasm of Japanese industrials in the 80s, now fallen out in favour of a radical scam, soft "logic".

[30] If I pose logically the problem of sorting a list, it is impossible that the application of generic techniques in automatic proof would give me the efficiency of "quicksort".

slogan we began with is recognised as a humbug. However the idea deserved better: it will lack a little modesty — confined to certain types of task — and above all theoreticians... it is staggering to consider that the theory of logic programming exists almost entirely in the work of Herbrand and Gentzen, see eg [GLT90].

4.3 Denotational semantics

Let us return to the lambda calculus, of which the typed version is isomorphic (by Curry-Howard) to intuitionistic logic. Although it is very robust, it presents itself as a formal functional calculus, without a "naive" interpretation. More precisely, the original calculus of Church (untyped) admits no set-theoretic interpretation for reasons of circularity (eg the identity function satisfied $I(a) = a$ for every a, including $a = I$), while the typed calculus does admit a trivial interpretation in sets of which the cardinals explode well beyond the continuum, in contradiction to the local finiteness of the calculus.

One can see the typed lambda calculus as a *cartesian-closed category*, where $\mathrm{Hom}(X, Y)$ is itself an object of the category, which is expressed by way of a certain number of canonical isomorphisms. Where then can we find these categories (besides the hopeless category of sets)?

The idea of a topological interpretation is natural; however it will be necessary to give the space $\mathrm{Hom}(X, Y)$ of continuous functions from X to Y a topology, and there is no miracle: one runs into contradictory requirements of simple and uniform convergence. This is why the solution proposed by Scott (and also Ershov) in 1969, see eg [Sco76], is not topological except in name: the opens of *Scott domains* are final segments of certain lattices. These topologies are T_0 — the weakest form of separation — and non-uniformisable, which avoids having to choose between the two topologies on the space of functions. At any rate, while the topological aspect is a little doubtful, this *denotational semantics* has the good taste to remain within a reasonable cardinal (at most the continuum) and to function (thanks to an approximation by direct limits) in the untyped case, see [Bar84].

Denotational semantics is not a simple graph associated to functions; it takes into account an algorithmic aspect, ie the input / output dependence on pieces of information. Typically, it distinguishes between the accidentally constant function $f(n) = n - n$ and the deliberately constant function $f(n) = 0$ which however have the same graph. In other words, denotational semantics permits one — in a limited but genuine fashion — to distinguish multiple algorithms computing the same function. Thus, it was possible to work on the notion of a *sequential* algorithm by way of purely denotational considerations; the condition of *stability* due to Berry (1978) is satisfied by the denotational semantics of sequential algorithms.

The modelisation of System F poses a new problem, in the form of polymorphic types, and here the approach of Scott finds its natural limit. For example the function "reverse" of type $\forall\alpha(\mathrm{list}(\alpha) \implies \mathrm{list}(\alpha))$ takes as an argument a type σ to become the function which reverses lists with entries of type σ. To avoid circularity, I tried, inspired by dilators, to say that reversal is in fact defined on finite domains, then extended by a direct limit to any types. Unfortunately a Scott domain is not a direct limit of finite domains... and one needs to change the semantics, to introduce *coherence spaces*: these are simplified Scott domains which are induced by graphs: Berry's stability condition appears as the preservation of certain fibred products.

4.4 Linear logic

That which can do the most can the least; coherence spaces are also a model of first-order logic. They are sufficiently simple that one can amuse oneself with explicit calculations (which is more problematic with Scott's semantics, being very redundant). These calculations quickly make clear the need of abbreviations for intermediate non-logical configurations... or rather those without logical status. For example, coherence spaces develop an all-in-all decent linear algebra, and the notion of a linear function from X to Y aquires a natural sense. In fact, if we interpret the rules of intuitionist logic in linear algebra[31], eg implication as the space of linear maps, one discovers that almost all rules of sequent calculus pass this test, with the exception of two principles:

- *Weakening*, which permits the introduction of fictive hypotheses (if B then $A \implies B$) and which translates into fictive functional dependencies $f(x) = b$: linearity does not suffice for this, and we must admit affine functions.

- *Contraction*, which allows reusing hypotheses (if $A \wedge A \implies B$ then $A \implies B$), ie I have the right to use A twice in order to demonstrate B, and to only count it once. Here, we are required to take a function — to begin with bilinear — $f(x, x')$, and make a function of a single argument, which can only be $f(x, x)$, which one finds to be quadratic... thus we violate linearity[32].

Here one finds that Gentzen placed these two principles in a separate group of rules — which he called *structural*, thinking without doubt that they were obvious[33] — which do not depend on the logical connectives. In particular, one can modify logic to omit these rules: thus we obtain the first group of rule for *linear logic*. At this point we remark that conjunction is broken down into two forms: *multiplicative* (the tensor product $A \otimes B$) and *additive* (the direct sum $A \& B$[34]). Implication becomes linear implication $A \multimap B$. The difference in the two conjunction results from the management of linearity: from $A \multimap B$ and $A \multimap C$ one can deduce $A \multimap B \& C$, but only $A \otimes A \multimap B \otimes C$[35]. Repetitions of hypotheses (expressed by the tensor) are not free, and one therefore sees that linear implication corresponds to a single use of the hypothesis... better, $A \multimap B$ expresses that starting with A I have B, but I do not keep A. This is then a *causal* vision of logical deduction, which opposes the perpetuity of traditional truth in philosophy and mathematics. We have here truths that are volatile, contingent and dominated by the idea of *resources* and of action.

The second group of connectives is constituted by the exclamation point $!A$[36] which represents the symmetric algebra; this construction permits the linearisation of multilinear functions, by a formal change of domain, and as such we accept weakening and contraction for formulae of the form $!A$: usual implication becomes $!A \multimap B$, and we can see $!A$ as the perrenialisation

[31] This is not metaphorically linear algebra: we know now how to interpret logic in Banach spaces.

[32] In Banach spaces, one needs to use analytic functions on the unit ball to "recover contraction".

[33] There is a third, *exchange* which states the commutativity of logic (one can permute hypotheses) and of which the rejection gives us a non-commutative logic which is problematic.

[34] In Banach spaces, normed by l^∞; the l^1 norm on the direct sum defines the dual connective to &, ie $A \oplus B$.

[35] From this point of view one can see that linear logic questions the commutativity of provability and conjunction.

[36] And its dual $?A$.

of A, ie that the resources on A are potentially infinite[37]. The isomorphism $!(A\&B) \simeq !A \otimes !B$ suggests the name *exponential* for this group.

Finally, last but not least, linear negation, based on the idea of the dual space[38], inducing an involution which associates to each connective a mirror connective. Concretely, negation corresponds to the "action / reaction" duality and not at all to not doing the action: typically read / write, send / receive are litigants of linear negation.

Linear logic is truly a proof theory of the how, ie

- Linear logic adheres to the dynamic, and permits the representation of the state of abstract machines using provability, this usual logic cannot do without encasing the configuration in a mostrous temporal gangue. From this point of view, we approach imperative programming, which is exploited in a linear version of logic programming (moving points on a screen, changing their colour, etc) which gives this scene its first logical development (Jean-Marc Andreoli, for Xerox in Grenoble).

- Linear negation in fact nixes the directionality in logic, ie the distinction hypothesis / conclusion, or even input / output. This makes linear logic a fundamental tool for the study of asynchronous parallelism. There is non-trivial mathematics therein, in particular, the first geometric approach to proofs, *proof nets*.

4.5 And tomorrow?

There is hardly a doubt that computer-scientific applications will be developed. All the same, we are still waiting for genuine industrial products, but there are stirrings, as we have seen...

As to theoretical developments, there will be before anything the semantic method[39]. Proof theory ought to divest from all syntax, any bureaucracy[40]. Thus denotational semantics is not completely satisfactory, as it does not close in on itself: one would need completeness theorems, which is to say a justification of the logical rules without the least recourse to a "space of truth", whatever that is; this question and the neighbouring questions of sequentialism, of *full abstraction* carry much support. It is the same for questions of dynamic semantics which can take on the appearance of *game semantics* — of which the the precursor is (as always) Gentzen, see Section 1.5.2 — of *abstract machines*, for example that of Krivine, or more mathematically of *geometry of interaction*, [GLR95] formulated in terms of star-algebras. The mathematical unification of these forms of semantics is the principal problem which arises.

The justification of this enterprise is the progressive discovery of hidden structures, in what presents *a priori* as pure symbol pushing, but which tend to give credit to the idea that proofs are nothing but the verbalisation of physical structures gifted from dynamics, which is furthermore codified by Gentzen's theorem. This hypothesis permits us to hope that serious bridges might be established with physics, and expecially which quantum physics.

---

[37] Which is the case for mathematical truth, which does not wear out.

[38] In Banach spaces, one needs to give the "dual".

[39] A semantic analysis of the word "semantic" makes clear that it is almost free of meaning; we only use it here for convenience, in opposition to the old syntactic tradition.

[40] Who was it that said: *"In every proof theorist there is a sleeping bureaucrat."*?

5 Two technical appendices

Translator's note: see [GLT90], Chapter 3 for the lambda calculus (A), and Chapter 5 for sequent calculus (B).

References

[Bar84] H. Barendregt. *The lambda-calculus, its syntax and semantics*. North Holland, 1984.

[BF97a] C. Burali-Forti. Sulle classi ben ordinate. *Rendiconti del Circolo matematico di Palermo*, 11:260, 1897.

[BF97b] C. Burali-Forti. Una questione sui numeri transfiniti. *Rendiconti del Circolo matematico di Palermo*, 11:154–164, 1897.

[BFPS81] W. Buchholz, S. Feferman, W. Pohlers, and W. Sieg. *Iterated Inductive Definitions and Subsystems of Analysis: Recent Proof-theoretical studies*. Springer Verlag, 1981.

[Bri75] J. Bridge. A simplification of the Bachmann method for generating large countable ordinals. *Journal of Symbolic Logic*, 40:171–185, 1975.

[Bus86] S. Buss. *Bounded arithmetic*. Bibliopolis, 1986.

[CH85] T. Coquand and G. Huet. Constructions: a higher order proof system for mechanizing mathematics. In *EUROCAL '85*. Springer Verlag, 1985.

[Fef75] S. Feferman. A language and axioms for explicit mathematics. In *Algebra and Logic*, pages 87–139. Springer Verlag, 1975.

[Gal91] J. Gallier. What is so special about Kruskal's theorem and the ordinal Γ_0? *Annals of Pure and applied logic*, 53:199–260, 1991.

[Gen35] G. Gentzen. Untersuchungen über das logische schliessen. *Mathematische Zeitschrift*, 39:176–210, 405–431, 1935.

[Gen36a] G. Gentzen. Die widerspruchsfreiheit der reinen zahlentheorie. *Mathematische Annalen*, 112:493–431, 1936.

[Gen36b] G. Gentzen. Die widerspruchsfreiheit der stufenlogik. *Mathematische Zeitschrift*, 41.3:357–366, 1936.

[Gen38] G. Gentzen. Neue fassung des widerspruchsfreiheitbeweises für die reine zahlentheorie. *Forschungen zue Logik und zur Grundlegung der exakten Wissenschaften, nouvelle série*, 4:19–44, 1938.

[Gen69] G. Gentzen. *The collected works of Gehrard Gentzen*. North Holland, 1969.

[Gir81] J. Girard. Π^1_2-logic, part I: dilators. *Annals of Mathematical Logic*, 21:75–219, 1981.

[Gir84] J. Girard. The Ω-rule. In *Proceedings of the International Congress of Mathematicians*, pages 307–321, 1984.

[Gir87] J. Girard. *Proof-theory and logical complexity I*. Bibliopolis, 1987.

[Gir00] J. Girard. Du pourquoi au comment: la thèorie de la démonstration de 1950 à nos jours. In *Les mathématiques 1950-2000*, pages 515–545. Birkhauser, 2000. Available at `https://girard.perso.math.cnrs.fr/theodem.pdf`.

[Gir11] J.-Y. Girard. *The Blind Spot: Lectures on Logic.* European Mathematical Society Publishing House, 2011.

[GLR95] J.-Y. Girard, Y. Lafont, and L. Regnier, editors. *Advances in Linear Logic.* Cambridge University Press, 1995. Volume 222 of *London Mathematical Society Lecture Note Series.*

[GLT90] J.-Y. Girard, Y. Lafont, and P. Taylor. *Proofs and types.* Cambridge University Press, 1990.

[Her31] J. Herbrand. Sur le problème fondamental de la logique mathématique. *Sprawozdania z posiedzen Towarzystwa Naukowego Warszawskiego, Wyzdial III*, 24:12–56, 1931.

[Her71] J. Herbrand. *Collected Works.* Harvard University Press, 1971.

[Hil05] D. Hilbert. *Uber die Grundlagen der Logik und die Arithmetik.* Verhandlungen des Dritten Internationalen Mathematiker-Kongresses in Heidelberg, 1905.

[Hil26] D. Hilbert. Uber das unendlische. *Mathematische Annalen*, 95:161–190, 1926.

[KL68] G. Kreisel and A. Levy. Reflection principles and their use for establishing the complexity of logical systems. *Zeitschrift für Mathetische Logik*, 14:97–142, 1968.

[Kre51] G. Kreisel. On the interpretation of non-finitistic proofs i. *Journal for Symbolic Logic*, 16:241–267, 1951.

[Kre52] G. Kreisel. On the interpretation of non-finitistic proofs ii. *Journal for Symbolic Logic*, 17:43–58, 1952.

[Kri90] J.-L. Krivine. *Lambda-calcul, types et modèles.* Masson, 1990.

[Kri94] J.-L. Krivine. Classical logic, storage operators and second order lambda-calculus. *Annals of Pure and Applied Logic*, 68:53–78, 1994.

[ML84] P. Martin-Löf. *Intuitionistic Type Theory.* Bibliopolis, 1984.

[Sch77] K. Schütte. *Proof-theory.* Springer Verlag, 1977.

[Sco76] D. Scott. Data types as lattices. *SIAM Journal of Computing*, 5:522–587, 1976.

[Tak67] G. Takeuti. Consistency proofs for subsystems of classical analysis. *Annals of Mathematics*, 86:299–348, 1967.

[vD90] D. van Dalen. The war of the frogs and the mice, or the crisis of the mathematische annalen. *Mathematical Intelligencer*, 12:17–31, 1990.

[vH67] J. van Heijenoort. *From Frege to Gödel.* Harvard University Press, 1967.