# Dialectica: From Gödel to Curry-Howard

Pierre-Marie Pédrot[1]

PPS, $\pi r^2$ team, Univ. Paris Diderot, Sorbonne Paris Cité,
UMR 7126 CNRS, INRIA Paris-Rocquencourt, Paris, France
pierre-marie.pedrot@inria.fr

Originally introduced by Gödel in the eponymous *Dialectica* journal in 1958 [6], the Dialectica transformation was a tentative workaround to the then-perceived cataclysm of the incompleteness theorems. As classical logic could not be considered a firm ground for the foundations of mathematics anymore, one had to rely upon constructive arguments.

Similarly to its predecessor, the double-negation translation, Dialectica aimed at providing classical logic with computational roots, through a transformation of **HA** into system **T** [1]. Unlike the double-negation translation, Dialectica is more fine-grained. Indeed, while retaining the disjunction and existence properties of intuitionistic logic, Dialectica realizes two semi-classical principles, namely Markov's principle (MP) and independence of premises (IP), which, given any decidable proposition $P$ on natural numbers, are usually stated as:

$$\text{MP}\ \frac{\neg(\forall n^{\mathbb{N}}.\, \neg P\, n)}{\exists n^{\mathbb{N}}.\, P\, n} \qquad \frac{(\forall m^{\mathbb{N}}.\, P\, m) \to \exists n^{\mathbb{N}}.\, R\, n}{\exists n^{\mathbb{N}}.\, (\forall m^{\mathbb{N}}.\, P\, m) \to R\, n}\ \text{IP}$$

While representing a major breakthrough at the time of its publication, the Dialectica transformation looks rather *bizarre* by modern standards of proof theory. First, as already observed by then [3], the translation of the contraction rule required the atomic propositions to be decidable. Second, and more worrisome in the Curry-Howard paradigmatic view, the translation does not preserve $\beta$-equivalence.

In her PhD thesis [2], De Paiva proposed a categorical presentation of the Dialectica translation that somehow solved both issues at once. This presentation made the crucial observation that the original Dialectica could be understood as a translation acting over linear logic rather than intuitionistic logic, using Girard's historical call-by-name decomposition of the arrow [4]. This categorical presentation led to more intricate constructions [9, 8], allowing both to factorize models of linear logic from the literature through this Dialectica-like transformation, and to easily design new ones by following the same pattern.

Conversely, and strangely enough, to the best of our knowledge, the Dialectica translation by itself did not benefit from this categorical apparatus. In particular, a clear understanding of the computational effects at work in the translation remained to be found. What does the program corresponding to the translation of a proof actually do?

We answer this legitimate question by providing our own syntactical, untyped presentation of Dialectica in a slightly extended $\lambda$-calculus, based on De Paivas's work. It happens that the very computational content of this translation can be easily explained thanks to the usual Krivine abstract machine (KAM) with closures, in an approach quite similar to the one of classical realizability [10, 12]. Essentially, Dialectica allows to capture the current stack of the machine when accessing a variable in the environment.

This feature can be seen as a weak form of delimited control, embodied by the operator

$$\mathcal{M} : (A \Rightarrow B) \Rightarrow A \Rightarrow {\sim}B \Rightarrow \mathfrak{M}({\sim}A)$$

where ${\sim}X$ denotes the type of stacks of $X$ and $\mathfrak{M}X$ the finite multiset over $X$. Here, stacks are given a first-class citizenship and made inspectable, which is fairly stronger than the usual arrow type of continuations. Given any function $f : A \Rightarrow B$, any argument $t : A$ and any return stack $\pi : {\sim}B$, $\mathcal{M}\, f\, t\, \pi$ computes the multiset of stacks $\{\rho_1, \ldots, \rho_n\}$ where each $\rho_i$ is the current stack of the machine for each corresponding use of $t$ by $f$, delimited by $\pi$.

There is an intriguing mismatch, though. Indeed, the KAM produces the stacks in a definite order, because of the sequentiality of the reduction, but the Dialectica translation does not, because it constructs a multiset instead of a list. Yet, there is no obvious way to tweak the Dialectica transformation to recover the sequentiality in the list of produced stacks. This defect actually seems deeply rooted in the linear decomposition itself.

Our syntactical presentation has the advantage to be compatible with the usual constructions around $\lambda$-calculus. For instance, we can easily apply it to more complicated settings, like dependent types. We obtain a Dialectica translation for $\mathbf{CC}_\omega$ [11] almost trivially. The translation is also applicable to the dependent elimination of inductives, hinting towards a translation for the full-fledged **CIC** system.

In a more general way, the dependently-typed Dialectica gives interesting hindsights into what could be (or not) linear dependent types, and provides more generally enlightening intuitions about effects and continuations in a dependent type theory. Finally, we believe that we could design similar transformations inspired by its computational content able to provide well-behaved versions of delimited control.

# References


[1] Jeremy Avigad and Solomon Feferman. Gödel's functional ("Dialectica") interpretation, 1998.

[2] Valeria de Paiva. A dialectica-like model of linear logic. In *Category Theory and Computer Science*, pages 341–356, 1989.

[3] Justus Diller. Eine Variante zur Dialectica-Interpretation der Heyting-Arithmetik endlicher Typen. *Archiv für mathematische Logik und Grundlagenforschung*, 16(1-2):49–66, 1974.

[4] Jean-Yves Girard. Linear logic. *Theor. Comput. Sci.*, 50:1–102, 1987.

[5] Jean-Yves Girard. *The Blind Spot: Lectures on Logic.* European Mathematical Society, 2011.

[6] Kurt Gödel. Über eine bisher noch nicht benützte Erweiterung des finiten Standpunktes. *Dialectica*, 12:280–287, 1958.

[7] Hugo Herbelin. An intuitionistic logic that proves Markov's principle. *LICS*, pages 50–56, 2010.

[8] J. M. E. Hyland. Proof theory in the abstract. *Ann. Pure Appl. Logic*, 114(1-3):43–78, 2002.

[9] Martin Hyland and Andrea Schalk. Glueing and orthogonality for models of linear logic. *Theor. Comput. Sci.*, 294(1/2):183–231, 2003.

[10] Jean-Louis Krivine. Dependent choice, 'quote' and the clock. *Theor. Comput. Sci.*, 308(1-3):259–276, 2003.

[11] Zhaohui Luo. An extended calculus of constructions, 1990.

[12] Alexandre Miquel. Forcing as a program transformation. In *LICS*, pages 197–206, 2011.

[13] Paulo Oliva. Unifying functional interpretations. *Notre Dame Journal of Formal Logic*, 47(2):263–290, 2006.

[14] Thomas Streicher and Ulrich Kohlenbach. Shoenfield is Gödel after Krivine. *Math. Log. Q.*, 53(2):176–179, 2007.