

Analyticity and syntheticity in type theory revisited

Bruno Bentzen

Abstract I discuss problems with Martin-Löf’s distinction between analytic and synthetic judgments in constructive type theory and propose a revision of his views. I maintain that a judgment is analytic when its correctness follows exclusively from the evaluation of the expressions occurring in it. I argue that Martin-Löf’s claim that all judgments of the forms $a : A$ and $a = b : A$ are analytic is unfounded. As I shall show, when A evaluates to a dependent function type $(x : B) \rightarrow C$, all judgments of these forms fail to be analytic and therefore end up as synthetic. Going beyond the scope of Martin-Löf’s original distinction, I also argue that all hypothetical judgments are synthetic and show how the analytic-synthetic distinction reworked here is capable of accommodating judgments of the forms A type and $A = B$ type as well. Finally, I consider and reject an alternative account of analyticity as decidability and assess Martin-Löf’s position on the analytic grounding of synthetic judgments.

1 Introduction

One thing most philosophers agree on about the analytic-synthetic distinction is that a proposition is analytic when true solely by virtue of its meaning and synthetic otherwise. I believe this is a semantic distinction that deserves to be accounted for not only classically but also within the opposing school of constructive semantics. Roughly, its basic tenets are that meaning and truth are rooted in provability. They can be traced back to the so-called proof explanation of the intuitionistic logical constants established by Brouwer, Heyting, Kolmogorov and many others (Troelstra and van Dalen, 1988). I am particularly interested in the kind of constructive semantics that originated in the work of Martin-Löf (1982) as a generalization of the proof explanation to type-theoretic contexts. The question I want to address in this paper is: how should the analytic-synthetic distinction be drawn?

To be precise, this constructive semantics takes the form of a body of informal meaning explanations used to justify the axioms and rules of inference of constructive type theory, a class of formal systems in which every object is assigned a type and every operation is restricted to objects of a certain type (Martin-Löf, 1975). The type theory is constructive since the objects of a type can be interpreted as proofs of a proposition in such a way that the disjunction property holds and all existential claims have witnesses. In fact, the meaning explanations generalize the proof explanation given that the objects of a type can also be regarded as elements of a set and propositions as sets that consist of their proofs.

The reason I wish to draw special attention to this remarkable form of constructive semantics is that Martin-Löf (1994) also supplemented it with the proposal of a distinction between analytic and synthetic judgments for his constructive type theory. His thoughts on the matter have been particularly influential, and his distinction has been widely accepted by those sympathetic to his constructivism without ever being challenged until now.¹ Proposing a revision of his views on analyticity and syntheticity is precisely my aim in this paper. My criticism of Martin-Löf's distinction will be directed to the following six points:

1. It should concern judgments and not forms of judgments, for the expressions that occur in a form of judgment are metavariables and have no meaning per se;
2. It presupposes a vague characterization of analyticity (and thus also syntheticity), lacking an explanation of what the meaning of an expression actually is;
3. It is based on an unfounded claim that all judgments of the form $a : A$ and $a = b : A$ are analytic due to the vagueness mentioned above;
4. It completely leaves out the issue of the analyticity or syntheticity of judgments of the forms A type and $A = B$ type and, more importantly, hypothetical judgments;
5. It must ultimately be rooted in the meaning explanations, thus decidability in a formal system has nothing to do with the analyticity or syntheticity of a judgment;
6. It is wrong that all synthetic judgments are “grounded on” analytic judgments, so both analyticity and syntheticity are equally basic and important notions.

These points will be thoroughly explained and elaborated upon in due course. The view I will arrive at is that a judgment must be analytic when its correctness is exclusively due to the evaluation of the expressions that occur in the judgment. Given that it is common to understand the meaning of an expression referentially by means of the evaluation of the program it expresses in constructive semantics (Bentzen, 2020), my account of analyticity can be regarded as a refinement of Martin-Löf's working principle that an analytic judgment is correct only by virtue of the meanings of the expressions that occur in it.

I argue that Martin-Löf should not have claimed that all judgments of the form $a : A$ and $a = b : A$ are analytic. Because analyticity is not a property of forms of judgments but judgments themselves, we must examine each judgment and the meaning of its expressions individually to support this claim. Yet, as we shall see, when A evaluates to a dependent function type $(x : B) \rightarrow C$, all judgments of these forms fail to be analytic and therefore end

¹For an excellent historical account of analyticity from Aristotle to the constructive tradition of Martin-Löf and attributions on the level of proposition and judgment, see Sundholm (2013). Therein Martin-Löf's account is acknowledged as “a clear case of Kantian analyticity” (p.33) for its emphasis on decidability.

up as synthetic. Evaluation cannot be decisive for their correctness because of the meaning explanation given for dependent function types, which is dependent on the purely extensional meaning explanation of hypothetical judgments. Going beyond the scope of Martin-Löf's distinction, I also argue that all hypothetical judgments are synthetic and show how my analytic-synthetic distinction accommodates judgments of the forms A type and $A = B$ type. Although there are several instances of analytic judgments of these forms, just as with $a : A$ and $a = b : A$ syntheticity is the norm whenever A evaluates to a dependent function type. To conclude, I consider and reject an alternative account of analyticity as decidability and assess Martin-Löf's position on the analytic grounding of synthetic judgments.

The remainder of this paper is structured as follows. Section 2 gives a general overview of all the forms of judgment of constructive type theory and discusses the analytic-synthetic distinction proposed by Martin-Löf (1994). Section 3 presents the meaning explanations of the basic forms of judgment in detail. In Section 4 I advance my own views through a systematic criticism of Martin-Löf's analytic-synthetic distinction following the points mentioned above. Section 5 replies to some possible objections and general concerns. Finally, Section 6 contains some concluding remarks and outlines directions for future work.

2 Analyticity and syntheticity in Martin-Löf

The aim of this section is to present Martin-Löf's original views on the distinction between analytic and synthetic judgments in constructive type theory. My own critical assessment of his ideas will have to wait until the next sections. Before I begin, however, I want to stress that his analytic-synthetic distinction is drawn in the domain of judgments and not that of propositions as it is common in the literature on classical semantics. Allow me to start by elaborating on the role of judgments in the development of the logic tradition.

Judgments are what can be conferred assertive force and be subject to rules of inference. They convey the idea that one or more objects have a property or stand in a relation. By making a judgment one expresses the knowledge that what is being judged is the case, so judgments may be regarded as objects of knowledge (Martin-Löf, 1985, p.19). When different sorts of properties or relations describe the content of two judgments, we typically say that these judgments have different forms. To know a judgment of a certain form is to know that the objects involved have the properties in question or stand in the relevant relation. I shall also call a judgment that can be made or known a correct judgment.

Throughout the history of logic, various forms of judgment have been observed. For instance, the Aristotelian tradition rests on judgments of the subject-copula-predicate form, S is P , which constitute the premises and conclusions of a syllogism. Judgments of this form express that a particular subject S has a certain property P . Indeed, the Kantian analytic-synthetic distinction is famously proposed for judgments of the S is P form (Kant, 1781, B10). Since especially the works of Bolzano and Frege, modern logic has adopted a new primitive form of judgment, A true, thereby departing from the Aristotelian tradition.² Judgments of this modern form express that a proposition is true and even today continue to figure in both

²See (Martin-Löf, 1985, p.16), (Sundholm, 1994, pp.374–377) and (Sundholm, 2004, pp.441–443).

the premises and the conclusion of an inference. Although Frege stressed the importance of making the truth particle explicit in judgments of this form, as famously demonstrated by his use of the turnstile symbol in his notation, this particle is typically omitted when referring to judgments of this form in the contemporary literature. This has resulted in a confusion between the judgment A true and the proposition A . The former states that a proposition A is true while the latter simply stands for a proposition A that may either be true or not. When one characterizes a true proposition A as analytic or synthetic one is thus saying that the judgment A true is analytic or synthetic. This is especially evident in Frege (1884, §3). Yet, because there is only one form of judgment in the background, A true, it still makes sense to say that the analytic-synthetic distinction is drawn for true propositions. In fact, the distinction still operates at the level of judgments like it does in Kant, the difference is that we are restricting all judgments to the one form that states that a proposition is true.

Propositions are what can serve as truth bearers and are connected by the logical connectives. They lack assertive power and may only figure as the content of A true. Constructively, a proposition is defined by prescribing how its canonical proofs may be formed and how they can be recognized as the same (Martin-Löf, 1987, pp.412–413), that is, by specifying what are the direct ways to prove a proposition and when two such proofs are identical. To illustrate, implication is defined as follows: the canonical way to prove $A \rightarrow B$ is by providing a method that yields a proof of B when applied to a proof of A . Additionally, two such methods are the same when they produce identical proofs of B given a proof of A . The knowledge that something is a proposition therefore amounts to the knowledge of what its canonical proofs are and when two such canonical proofs are the same. Martin-Löf (1975) noticed that the modern one-dimensional judgment structure A true was not enough to effectively capture the subtleties of knowledge in constructive semantics. He then introduced a more complex judgment structure with his constructive type theory in order to accurately express knowledge from a constructive perspective within a formal system.

Constructive type theory expresses knowledge through not one but four fundamental forms of judgment that say that something is a type, two things are the same type, something is an object of a type, and two things are the same objects of a type, respectively:

$$\begin{aligned} &A \text{ type}, & A = B \text{ type}, \\ &a : A, & a = b : A. \end{aligned}$$

Perhaps a few words are in order to motivate these four forms of judgment. First, notice that Martin-Löf diverges from the modern tradition of Frege and Bolzano and regards $a : A$ and $a = b : A$ as primitive forms of judgment, as opposed to A true. Well, this adjustment is only natural from the constructive standpoint, given that these two forms of judgment can be respectively read as “ a is a proof of A ” and “ a and b are equal proofs of A ” and what makes a proposition true is a proof of it. Thus, $a : A$ and $a = b : A$ are more basic than A true. Under this logical reading the other two forms of judgment A type and $A = B$ type simply express that “ A is a proposition” and that “ A and B are equal propositions”. In Section 3, we will see that these forms of judgment extend beyond propositions and proofs to include sets and their elements as well, which can also be interpreted as types and their objects.

I think I have said all I need to put Martin-Löf's distinction into proper perspective. Classically, it is not harmful to locate the analytic-synthetic distinction on the level of true propositions because there is only one form of judgment, A true. But with the complex judgment structure required by constructive semantics, as Martin-Löf understands it, we are forced to look for a more elaborate analytic-synthetic distinction that attributes analyticity and syntheticity to judgments having the forms A type, $A = B$ type, $a : A$, and $a = b : A$. This understanding is sufficient for us to comprehend his central claims. I leave a fully detailed account of the meaning explanations of these forms of judgment to the next section.

2.1 Analytic judgments

Martin-Löf embraces the now widespread view that an analytic judgment is one which is correct only in virtue of the meanings of the terms that occur in it (Martin-Löf, 1994, p.88). According to him, this "current formulation" of analyticity, as he calls it, is in essential agreement with the Kantian insight that analytic judgments are explicative. After a brief exposition of some basic rules of inference valid in his constructive type theory, which I shall omit here since they are not directly relevant to our discussion, Martin-Löf claims that the forms of judgment $a : A$ and $a = b : A$ are both analytic:

Now, whatever type α that we choose, the two forms of judgement $a : \alpha$ and $a = b : \alpha$ are both analytic. You see, if a judgement of one of these two forms is evident at all, then it is evident solely by virtue of the meanings of the terms that occur in it. (Martin-Löf, 1994, p.93)

The reason given is simply that the correctness for such forms of judgment follows from the meanings of the expressions that occur in them and nothing else. This is unfortunately all Martin-Löf tells us about analyticity on this occasion. But later in his paper, he adds that the analytic-synthetic distinction is crucial for a proper understanding of the incompleteness and undecidability phenomena in logic, observing that judgments of the two analytic forms $a : A$ and $a = b : A$ are decidable and, consequently, complete too:

[...] if you have a judgement of one of the two analytical forms, that is, one of the two forms $a : \alpha$ and $a = b : \alpha$, then it can be checked, or decided, whether or not that judgement is derivable by means of the formal rules, and the algorithm for doing that is what the computer scientists call the type checking algorithm, which is so to speak the formal, or logical, heart of the computer proof systems that I just mentioned. (Martin-Löf, 1994, p.97)

For Martin-Löf the "logic of analytic judgments", as he calls it, is complete and decidable because there is an algorithm to check whether $a : A$ and $a = b : A$ are derivable. The kind of decidability referred to here therefore amounts to derivability in a formal system. This is an important point that I shall soon return to in Section 4.5. The requirement that typechecking be decidable in the theory is closely related to Kreisel's (1962) dictum that we must recognize a proof when we see one, given the propositions-as-types correspondence. Note that the completeness of the "logic of analytic judgments" follows because decidability guarantees that every judgment $a : A$ and $a = b : A$ is either correct or not.

2.2 Synthetic judgments

Having concluded that the forms of judgment $a : A$ and $a = b : A$ are analytic, Martin-Löf asks where do synthetic judgments come from. To understand his account of syntheticity it is necessary we first discuss how his constructive semantics incorporates the tenet that provability has conceptual priority over truth. The truth of a proposition is expressed with the introduction of a fifth form of judgment A true that is explained to be correct provided some judgment $a : A$ is correct in the presence of some a . In other words, the new form of judgment A true is derived from the primitive $a : A$ and the notion of existence. Some comments on the role of existence in Martin-Löf's thought will be helpful.

Existence must enter the analysis of judgments of the form A true because for Martin-Löf a proposition A is true only if there exists a proof of A . It is imperative to acknowledge that existence cannot be examined through the use of the existential quantifiers here. This attempt would only lead to an infinite regress of questioning what it means for the resulting existentially quantified proposition to be true (Martin-Löf, 1993, pp.138–139). We should instead make use of the traditional philosophical notion of concept instantiation, viewing, as Martin-Löf (1994, p.93) says, the existence of A in terms of objects falling under A . This passage holds special philosophical significance as it illustrates how Martin-Löf treats concepts and types as synonymous philosophical notions. He even explicitly alludes to a Kantian interpretation of types as categories (Martin-Löf, 1994, p.91). Unfortunately, to delve any further into this difficult topic would take us too far astray, so I will simply take his claim for granted and think of objects falling under concepts as objects belonging to a type.³ The suggestion is then that we can say that A is true whenever the type A is inhabited. To put it differently, whenever it is the case that some a is an object of A , it follows that A is true. This meaning stipulation can be more clearly stated in the form of a semantic rule:

$$\frac{a : A}{A \text{ true.}}$$

Martin-Löf argues that the form of judgment A true is the synthetic form of judgment. Conceptual analysis alone does not suffice to establish correctness, because A true is correct only when there exists an a such that $a : A$ is correct, so to know that A true one has to go beyond what is contained in the judgment and find an object of the relevant type:

The synthetic form of judgement is precisely the existential form of judgement that I have just introduced. [...] To see why, we have to ask ourselves, what does its evidence rest on? Is it merely on conceptual analysis, or do we have to go beyond what is contained entirely within the judgement in order to make it evident? Well, its evidence rests on a construction: you see, we arrive at an existential judgment α exists, say through the construction of an object which falls under the concept α , that is, through the construction of the concept [...] in Kantian terminology. (Martin-Löf, 1994, p.94)

³I refer the reader to (Martin-Löf, 1993, pp.20–22) for an in-depth comparison between types and categories.

Since according to Martin-Löf the synthetic judgment is obtained by suppressing part of the underlying analytic judgment, he concludes that every synthetic judgment is “grounded on” an analytic judgment in this sense (Martin-Löf, 1994, p.95). He sees this as the reason for why the notion of analyticity is more basic and important than that of syntheticity. What does it mean to say that A true suppresses part of the underlying analytic judgment? Another point made by Martin-Löf is that the judgment A true is some kind of incomplete or abbreviated judgment, where leave out the object a of the judgment $a : A$ and choose to retain only the relevant information that A is inhabited (Martin-Löf, 1993, p.139).

Later in the paper when discussing incompleteness and undecidability, Martin-Löf states that the source of the phenomena is the fact that in the synthetic form of judgment A true the object a such that $a : A$ claimed to exist has been suppressed:

[...] we have to search for an object a of type α , and, even if we limit ourselves to a particular formal system, that search procedure will not necessarily terminate, let alone then if we put no limitation on the axioms that we are allowed to invoke in the course of that procedure. (Martin-Löf, 1994, pp.97–98)

Incompleteness and undecidability have the same source, Martin-Löf suggests. If we fix a formal system and consider a type A that can be formulated in it, he says, then it may be the case that the judgment A true is not derivable just because there is no $a : A$ that can be expressed in the system. However, by considering an extension of the original system with other axioms or rules of inference, it is possible that now there is a derivation of $a : A$. Martin-Löf connects this observation with Gödel’s discovery:

It was this phenomenon that Gödel discovered by taking the original system to be first order arithmetic, say, and constructing a particular first order arithmetical proposition A which is such that there is no proof of A in first order arithmetic [...] (Martin-Löf, 1994, p.98)

3 Overview of the meaning explanations

The analytic-synthetic distinction is drawn in terms of meaning, so a closer look on how the meanings of these judgments are explained in constructive type theory will be helpful in my assessment of Martin-Löf’s distinction. My exposition follows the earliest formulation of the meaning explanations by Martin-Löf (1982) based on a lower-order type structure. The only difference is that I adopt an intensional approach to type equality to simplify the presentation following Constable et al. (1985) and Allen (1987). This early formulation differs from the later version of the meaning explanations considered by Martin-Löf (1994) on top of a higher-order structure that places sets and propositions at the bottom level as ground types and builds higher types as function types. However, these are minor differences that are completely orthogonal to my criticism of Martin-Löf’s analytic-synthetic distinction and my own characterization of analyticity and syntheticity. I address objections against the neutrality of my particular formulation of the meaning explanations in Section 5.

The meaning explanations fix the intended interpretation of constructive type theory with a systematic account of what a judgment of each one of its forms means. This is done with an explanation of what one knows when one knows a judgment of one of the forms. So if \mathcal{J} is a judgment its meaning explanation goes as “to know \mathcal{J} is to know ...” Roughly, we explain the four primitive forms of judgment according to the following pattern:

- (i) To know A type is to know that A evaluates to a canonical type;
 To know a canonical type is to know
 - (a) how to construct a canonical object of it;
 - (b) how to show that two of its canonical objects are equal;
- (ii) To know $A = B$ type is to know that A and B evaluate to the same canonical type;
- (iii) To know $a : A$ is to know that a evaluates to a canonical object in A ;
- (iv) To know $a = b : A$ is to know that a and b evaluate to the same canonical object in A .

We do not need to add A true to the list above because the semantic rule from Section 2.2 allows for us to explain its meaning directly from the meaning explanation of $a : A$. Now, since each judgment in constructive type theory can depend, or not, on some hypotheses, the meaning explanations will actually fall into two groups. When there is dependence on a finite list of hypotheses, judgments are said to be hypothetical and when no assumptions are made they are said to be categorical. If the judgment is hypothetical, the list of hypotheses it depends on will be displayed within braces to the right as $\mathcal{J}(\mathcal{J}_1, \dots, \mathcal{J}_n)$.

3.1 Categorical judgments

The meaning explanations of categorical judgments begin with the specification of a native model of computation as the point of departure. This is the most fundamental assumption underlying the meaning explanations because objects will be assigned to types depending on their computational behavior. For example, the meaning of the categorical judgment $a : A$ will be explained based on what values a and A evaluate to. It follows from this order of priority that the model of computation taken as primitive in the explanation of type membership must itself be an untyped one to avoid circularity. It can be specified with the prescription of the syntax of a primitive programming language and its operational semantics.

The model of computation adopted here is a form of untyped lambda-calculus extended with constants for some types and their respective constructors and eliminators. They are the dependent function type, equality type, empty type, and natural number type, which will suffice for the point I want to make in this paper. Our syntax is as follows:

$$\begin{aligned} \text{var} &:= x_0 \mid x_1 \mid x_2 \mid \dots \\ \text{expr} &:= \text{var} \mid \lambda \text{var}. \text{expr} \mid \text{app}(\text{expr}, \text{expr}) \mid (\text{var} : \text{expr}) \rightarrow \text{expr} \\ &\quad \mid \text{refl} \mid \text{ld}(\text{expr}, \text{expr}, \text{expr}) \mid \text{void} \mid 0 \mid \text{s}(\text{expr}) \\ &\quad \mid \text{natrec}(\text{expr}, \text{expr}, \text{expr}) \mid \text{nat} \end{aligned}$$

Now that we have fixed the syntax of our programming language, let me briefly introduce the four types we will be talking about before we proceed. It is easier to explain them by reference to the sets and propositions that each of them corresponds to:

- We can think of the dependent function type $(x : A) \rightarrow B$ as the Cartesian product of a family of sets B indexed by A , set-theoretically. When the type family B is constant and does not depend on objects of A this is simply the set of functions from A to B . Under the propositions-as-types paradigm (Howard, 1980), the type $(x : A) \rightarrow B$ plays the role of universal quantification and can be translated as “all A s are B s”. It also boils down to the implication $A \rightarrow B$ when there is no dependence, hence the arrow notation. Objects of this type can be constructed as lambda functions $\lambda x.a$ and they can be used by means of function application with the eliminator $\text{app}(f, a)$.
- The equality type $\text{Id}(A, a, b)$ is the type of identifications of objects a and b of the type A . One straightforward interpretation in set theory views it as a set that is nonempty when a and b are equal elements of A and empty otherwise. At the same time, it can be regarded as a proposition that expresses the identity of a and b and is governed by Leibniz’s Law. Objects of this type are then just reflexivity proofs refl . Notice that there is no constant associated with an elimination rule for equality. There is no need for one because the meaning explanations validate the reflection rule (see Section 3.2), which allows us to assert $a = b : A$ when given a $p : \text{Id}(A, a, b)$ under any assumptions, and this already serves as the elimination rule for this type.
- The empty type void is the type that has no constructors. Its counterpart in set theory is the empty set and, in logic, the proposition it corresponds to is absurdity. The elimination rule for this type validated by the meaning explanations is a stronger version of *ex falso quodlibet* which states that given an object of void any judgment \mathcal{J} can be derived (see Section 3.2). Thus, again, there is no need to include a constant to act as an elimination rule for it in our programming language.
- The natural number type nat is the type that has a zero 0 and successor $s(n)$ constructor. Its eliminator $\text{natrec}(n, a, f)$ generalizes the principle of mathematical induction by roughly allowing us to define dependent functions over natural numbers by recursion. The type nat has an obvious interpretation in set theory as the set of natural numbers. Drawing inspiration from the propositions-as-types paradigm, one may also want to think of nat as a proposition that has countably many proofs 0 and $s(n)$ (Martin-Löf, 1993, p.163), but this reading is so unnatural in that it has absolutely no noticeable counterpart in logic. It is for this reason I am inclined to think of the propositions-as-types not as an isomorphism, as it is often referred to, but as an embedding of propositions into types. The term “isomorphism” suggests an interpretation that goes in both ways but there is no question that many types normally considered as part of constructive type theory (especially inductive types and universes) fail to have a natural counterpart in logic as propositions. It is of course expected that a foundational system of mathematics like constructive type theory go beyond logic. It is in this sense that we say that the meaning explanations generalize the proof explanation.

We may now endow our language with an operational semantics. The collection of valid programs in this language is determined by a binary transition relation over expressions with no occurrence of free variables (closed expressions). The relation is reflexive on the closed expressions that cannot be computed any further and are taken as values. This relation, written as $M \mapsto M'$ to indicate that M transitions to M' , can be defined as the smallest relation closed under the three groups of reduction rules given below:

$$\begin{array}{c}
\overline{(x : M) \rightarrow N \mapsto (x : M) \rightarrow N} \quad \overline{\lambda x.M \mapsto \lambda x.M} \\
\overline{\text{ld}(M, N, O) \mapsto \text{ld}(M, N, O)} \quad \overline{\text{refl} \mapsto \text{refl}} \quad \overline{\text{void} \mapsto \text{void}} \\
\overline{\text{nat} \mapsto \text{nat}} \quad \overline{0 \mapsto 0} \quad \overline{s(M) \mapsto s(M)} \\
\\
\frac{M \mapsto M'}{\overline{\text{app}(M, N) \mapsto \text{app}(M', N)}} \\
\overline{\text{app}(\lambda x.M, N) \mapsto M[N/x]} \quad \overline{\lambda x.\text{app}(M, x) \mapsto M} \\
\\
\frac{M \mapsto M'}{\overline{\text{natrec}(M, N, O) \mapsto \text{natrec}(M', N, O)}} \\
\overline{\text{natrec}(0, M, N) \mapsto M} \quad \overline{\text{natrec}(s(M), N, O) \mapsto (O(M))(\text{natrec}(M, N, O))}
\end{array}$$

Instead of using distinct kinds of letters a, b, f, n, A, B for the metavariables to suggest the expressions they stand for, as I did in the discussion above, I insisted to write them all uniformly as M, N , etc. to emphasize the fact that computation is untyped.

How should the reduction rules above be read? The first group of rules determines what expressions are values using a lazy evaluation strategy, which postpones the reduction of an expression until it becomes necessary to determine its value. This means that an expression may be a value even when some of its parts may not. One example of a value is $s(2 + 2)$ since it is already a successor, even though we could still evaluate $2 + 2$ further. The second and third groups describe the reduction rules for dependent functions and natural numbers respectively, telling us what happens when a function is applied to an argument and when a recursor operates on a natural number. For example, $\text{app}(\lambda x.x + 2, 2)$ reduces to $2 + 2$ given that the function that adds two yields two plus two when applied to two as argument. To reduce $\lambda x.\text{app}(M, x)$ to M we require that M has no free occurrences of x as usual. The reduction rules for natrec simply reflect the principles of definition by recursion. They state, in particular, that $m + 0$ reduces to m and $m + s(n)$ reduces to $s(m + n)$. For better readability, I am treating the addition sign as an abbreviation for the addition operation $m + n$ that can be defined by recursion on the second argument as $\text{natrec}(n, m, \lambda k.\lambda f.s(f))$. I am also using numerals to abbreviate $s(0)$ as 1, $s(1)$ as 2, and so on. Together, all these rules make closed expressions into programs that when executed may output a value after a finite number of reduction steps. If there is a sequence of reductions that starts with M and ends with value M' , we say that M evaluates to M' and write $M \downarrow M'$. Reductions need not always simplify an expression and evaluations may even go forever and never terminate.

3.1.1 Typehood and type equality

Knowing that A is a type means knowing that A evaluates to a canonical type. To begin with, therefore, I shall explain what being a canonical type means. Recall that the meaning explanations are always given in terms of what it is to know something. Knowing that A is a canonical type is to know how to construct a canonical object of A and the conditions under which two canonical objects of A are equal. To illustrate the idea, we have:

- We know that $(x : A) \rightarrow B$ is canonical because we prescribe that the function $\lambda x.b$ is a canonical object of it provided that b is an object of B given some object x of A ; moreover, two functions $\lambda x.b$ and $\lambda x.c$ are equal canonical objects of this canonical type when b and c are equal objects of B assuming that x is some object of A .
- We know that $\text{Id}(A, a, b)$ is canonical because we prescribe that refl is a canonical object of it provided a and b are equal objects of A ; moreover, refl is equal to itself.
- We know that void is canonical because we prescribe it has no canonical objects. I should stress that this stipulation does tell us how to construct objects of this type and when they are equal, namely, under absolutely no circumstances.
- We know that nat is a canonical type because we prescribe that 0 is a canonical object of it and that so is $s(n)$ when n is an object of nat (which may be non-canonical); moreover, 0 is equal to itself and $s(n)$ equals $s(m)$ when n and m are equal objects of nat .

In general, the meaning of a judgment of the form A type is explained as: to know that A is a type is to know that A evaluates to an A' that we know is a canonical type. Nevertheless, the precise format of the explanation depends on the canonical type under consideration. Our task at hand is to provide a concrete explanation of the meaning of A type for each of the canonical types introduced above, namely, dependent function types $(x : A) \rightarrow B$, equality types $\text{Id}(A, a, b)$, the empty type void , and natural number type nat .

After that, we will need to explain the meaning of judgments of the form $A = B$ type. These judgments are known whenever A and B evaluate to the same canonical type, so, again, the particular form of the explanation is contingent on the canonical types in question. Instead of giving two separate meaning explanations, first explaining A type and then $A = B$ type, I observe that the explanation of the meaning of A type is just an instance of the more comprehensive meaning explanation of the judgment $A = B$ type in the specific case where A and B happen to be syntactically identical as expressions.⁴ Thus, to avoid unnecessary repetition, I shall focus exclusively on the meaning explanation of $A = B$ type from this point on, implicitly treating the meaning explanation of A type as that of $A = A$ type.

⁴See e.g. Allen (1987, p.6), Constable et al. (1985, p.143), Angiuli et al. (2017, §3.1). This reduction is made possible by the fact that our meaning explanation of type equality is intensional. In Martin-Löf (1982), an extensional account of $A = B$ type is favored. I shall briefly compare both approaches in Section 5.

We explain the meaning of the type equality judgment $A = B \text{ type}$ by saying that to know that A and B are equal types is to know that they evaluate to the same canonical type. For atomic types such as the empty type void and natural number type nat which cannot be broken down into smaller parts this simply amounts to syntactic equality of canonical types. The idea can be more clearly illustrated in the rule format displayed below. To avoid confusion, one should not forget that these are semantic stipulations and not rules of inference:

$$\frac{A \downarrow \text{nat} \quad B \downarrow \text{nat}}{A = B \text{ type}} \quad \frac{A \downarrow \text{void} \quad B \downarrow \text{void}}{A = B \text{ type.}}$$

For compound types such as the dependent function and equality types, which are composed of other types as their parts, the meaning explanation above tells us that type equality is dependent on that of the canonical types that their parts evaluate to. In particular, we have:

$$\frac{A \downarrow (x : A') \rightarrow C' \quad B \downarrow (x : B') \rightarrow D' \quad A' = B' \text{ type} \quad C' = D' \text{ type } (x : A')}{A = B \text{ type}}$$

$$\frac{A \downarrow \text{Id}(A', a, c) \quad B \downarrow \text{Id}(B', b, d) \quad A' = B' \text{ type} \quad a = b : A' \quad c = d : B'}{A = B \text{ type.}}$$

Since the evaluation of a closed expression may not always terminate, the problem of whether or not A is a type according to the meaning explanations is not decidable, even though in the formalism the judgment $A \text{ type}$ is often decidable. Indeed, as we shall soon see, the meaning of judgments of the form $a : A$ and $a = b : A$ will be explained by means of evaluation as well, so they too are undecidable in general. This discrepancy with the intended interpretation is due to the fact that the axioms and rules of inference chosen for the formal system may not be rich enough to fully capture this aspect of the meaning explanations. The failure of decidability in the meaning explanations is a highly significant point, in view of the role it plays in Martin-Löf's analytic-synthetic distinction. I am not contradicting his claims that $a : A$ and $a = b : A$ are decidable in a particular formal system. I am stating that they fail to be decidable in its intended interpretation, the meaning explanations. Another important semantic element responsible for this lack of decidability is the explanation of the meaning of hypothetical judgments. I shall go back to this point in Section 3.2.

3.1.2 Type membership and member equality

We explain the meaning of $a : A$ under the assumption that A is a type. Knowing that a is an object of the type A , assuming the knowledge that A evaluates to a canonical type A' , is knowing that a evaluates to a canonical object of the canonical type A' . We shall also derive this meaning explanation from a more general meaning explanation for $a = b : A$ in the particular case where a and b are syntactically identical. Therefore, we shall only be concerned with the meaning explanation for $a = b : A$ in what follows.

Knowing from the assumption that A is a type that it evaluates to the canonical type A' , we say that to know that $a = b : A$ is to know that a and b both evaluate to equal canonical objects of A' , where we have stipulated above which canonical objects belong to which types and when they are equal. The meaning explanation of $a = b : A$ is immediately clear in cases when objects evaluate to strictly the same canonical object:

$$\frac{A \downarrow \text{nat} \quad a \downarrow 0 \quad b \downarrow 0}{a = b : A.}$$

For other pairs of objects of the same type that evaluate to more complex canonical objects that need not be strictly identical to one another, we just demand the knowledge that the parts of their values be equal objects of the same type. With this, we are given meaning explanations for all objects of the natural number, dependent function, equality, and empty type. The latter only gets covered through a vacuous explanation that consists of no rules since the empty type has no canonical objects that can be evaluated to. Thus the explanations are:

$$\frac{A \downarrow \text{nat} \quad a \downarrow s(a') \quad b \downarrow s(b') \quad a' = b' : \text{nat}}{a = b : A}$$

$$\frac{A \downarrow (x : A') \rightarrow C' \quad a \downarrow \lambda x.c \quad b \downarrow \lambda x.d \quad c = d : C' (x : A')}{a = b : A}$$

$$\frac{A \downarrow \text{Id}(A', a', c') \quad a \downarrow \text{refl} \quad b \downarrow \text{refl} \quad a' = c' : A'}{a = b : A.}$$

Note that the meaning explanations for dependent function types given above for $a = b : A$ and earlier for $A = B$ type presuppose a meaning explanation for hypothetical judgments. I shall now explain what the meaning of a hypothetical judgment is.

3.2 Hypothetical judgments

We will be exclusively concerned with hypothetical judgments that depend on a finite list of type membership judgments $x_1 : A_1, \dots, x_n : A_n$ where each x_i is a variable. We may think of the closed expressions that occur in a categorical judgment as complete programs in the sense that they can be executed in isolation anytime. Then the open expressions which occur in a hypothetical judgment depending on a finite list of typed variables are best seen as incomplete programs that need parameters for their execution. We can then think of the hypotheses $x_1 : A_1, \dots, x_n : A_n$ as variable declarations. These incomplete programs behave functionally given that they only serve to yield complete programs whenever other complete programs are passed as parameters of appropriate types to replace their variables.

Hypothetical judgments are explained by induction on the length of their list of declared variables that for the sake of brevity we shall denote as Γ . Since the base case is that of a categorical judgment, it suffices that we focus our attention on the inductive step. Assuming that we already know what a hypothetical judgment $\mathcal{J}(\Gamma)$ means for any judgment \mathcal{J} , we can deal with the inductive case through the stipulation that to know $\mathcal{J}(\Gamma, x : A)$ is to know that $\mathcal{J}[M/x](\Gamma)$ for any $M : A(\Gamma)$. Put differently, this means that

$$\mathcal{J} (x_1 : A_1, \dots, x_n : A_n)$$

is correct when every single categorical judgment $\mathcal{J}[a_1/x_1, \dots, a_n/x_n]$ is correct for all

$$a_1 : A_1, \dots, a_n : A_n[a_1/x_1, \dots, a_{n-1}/x_{n-1}].$$

We also require additional conditions demanding that hypothetical judgments have functional extensionality with respect to type and object equality (Martin-Löf, 1982, pp.164–166), but the simplified picture we have drawn here will be enough for our purposes.

To illustrate how the meaning explanation of hypothetical judgments works, I want to call attention to two of its notable implications that result in undecidability. They involve the validation in the meaning explanations of the reflection rule and judgmental ex falso rule mentioned earlier in Section 3.1. First, the reflection rule states that if the hypothetical judgment that a certain equality type is inhabited is known

$$p : \text{Id}(A, x, y) (x_1 : A_1, \dots, x_n : A_n)$$

then the hypothetical judgment that the relevant objects are equal is also known

$$x = y : A (x_1 : A_1, \dots, x_n : A_n).$$

This rule is valid, for according to the explanation of the meaning of hypothetical judgments, the assumption that the former judgment is known means that the categorical judgment $p[a_1/x_1, \dots, a_n/x_n] : \text{Id}(A[a_1/x_1, \dots, a_n/x_n], x[a_1/x_1, \dots, a_n/x_n], y[a_1/x_1, \dots, a_n/x_n])$ is known no matter what closed objects $a_1 : A_1, \dots, a_n : A_n[a_1/x_1, \dots, a_{n-1}/x_{n-1}]$ we choose. This, in turn, as seen in Section 3.1.2, means we know that $p[a_1/x_1, \dots, a_n/x_n] \downarrow \text{refl}$ and $x[a_1/x_1, \dots, a_n/x_n] = y[a_1/x_1, \dots, a_n/x_n] : A[a_1/x_1, \dots, a_n/x_n]$ (Dybjer, 2012, §11.2.7).

Second, we note that a stronger version of the ex falso quodlibet rule in which the explosion takes place at the level of judgments is valid. Specifically, as there are no closed objects $a : \text{void}$ to account for, since void has no canonical objects, any hypothetical judgment $\mathcal{J} (x : \text{void})$ is vacuously correct for any judgment \mathcal{J} expressible in our language. Or, more generally, the judgment $\mathcal{J} (x : A)$ is known for any type A that is known not to be inhabited.⁵

Both the reflection and generalized ex falso rules are causes of undecidability of judgments of the form $a : A$ or $a = b : A$ in the meaning explanations, given that we cannot generally decide when their correctness is a consequence of one of these rules. That would imply a decision method for when a type is inhabited, since the reflection rule asks when equality types are inhabited and the judgmental ex falso rule when types are not inhabited.

⁵I thank Göran Sundholm for pointing out to me that knowledge of noninhabitation is required.

4 Our account of analyticity and syntheticity

Now that we have discussed the meaning explanations in the previous section, it is time to take a critical look at the analytic-synthetic distinction proposed by Martin-Löf (1994) and already examined at length in Section 2. In what follows, I will level six charges against his distinction and overcome them by proposing and defending my own. My criticism is motivated by six fundamental questions that I will ask about his distinction:

1. Does it do justice to the “current formulation” of analytic judgment as one which is correct solely by virtue of the meanings of the expressions involved?
2. Does it explain what the meaning of an expression in the “current formulation” is?
3. Can the meaning explanations justify the central claims that the judgments of the two analytic forms are $a : A$ and $a = b : A$ and that the synthetic form is A true?
4. Does it conclusively attribute analyticity or syntheticity to all correct judgments?
5. Can the connections drawn between decidability and analyticity be defended by appealing to the meaning explanations?
6. Can the claim that synthetic judgments are “grounded on” analytic ones be supported?

All questions will be answered negatively due to the problems indicated in Section 1. I shall address each problem in turn in the subsections to follow while advancing, at the same time, an alternative account of analyticity and syntheticity that can remedy them.

Given that I will establish my proposal through a systematic critique and reconstruction of Martin-Löf’s ideas, a major part of my argumentation will be negative. Yet, a crucial positive argument underlies my discussion of the first two questions, which runs as follows: firstly, if a judgment is analytic when correct only in virtue of the meaning of the expressions occurring in it and synthetic otherwise then we should examine what the expressions in the judgment in question mean before attributing analyticity or syntheticity; secondly, the meaning of an expression here should be understood as reference, which, from a constructive standpoint, is traditionally understood as evaluation of programs. Recall that evaluation is conceived by means of the computational model described in Section 3.1. Therefore, the analytic-synthetic distinction should center around the evaluation of expressions occurring in the judgment. This is how I will arrive at my own definition of analytic and synthetic judgments:

Definition 1 (Analyticity and Syntheticity) *A judgment is analytic when its correctness is entirely a consequence of the evaluation of the expressions that occur in it and nothing else. A judgment is synthetic when its correctness cannot be established just by evaluation.*

As I hope to show, my definition of analyticity and syntheticity is simply a more careful rendering of the “current formulation” of analyticity invoked by Martin-Löf (1994, p.88) when the relevant notion of meaning is investigated more thoroughly. After we analyze what the meaning of an expression is, we realize that some judgments of the form $a : A$ and $a = b : A$ cannot be correct only in virtue of meaning, that we should also investigate the analytic or synthetic character of judgments of the form A type and $A = B$ type, that questions of decidability are peripheral to those of analyticity and syntheticity, and that synthetic judgments are not grounded on analytic ones, revealing that both are basic notions.

4.1 Analyticity according to the “current formulation”

Recall that the so-called “current formulation” of analyticity states that an analytic judgment is correct solely by virtue of the meaning of the expressions that occur in it. Is it upheld in his distinction as Martin-Löf (1994, p.93) claims it is? I do not think so. Some passages quoted in Section 2 suggest that he attributes analyticity to the two forms of judgment $a : A$ and $a = b : A$ themselves, not to the actual judgments expressible in the language that have these forms. For instance, $0 : \text{nat}$ or $s(0) = s(0) : \text{nat}$. The crucial observation is that forms of judgments are not judgments themselves. Being a form of judgment means that whenever the metavariables are replaced with expressions of the language the result is a judgment. So $a : A$ and $a = b : A$ are forms of judgment but not in fact judgments themselves, given that the metavariables a , b , and A that occur in them are not part of the syntax of the language. Martin-Löf also seems to attribute syntheticity to the form of judgment A true, rather than to judgments of this form such as $\text{ld}(\text{nat}, 2 + 2, 4)$ true. This practice of relocating the distinction to the realm of forms of judgments is illustrated in Martin-Löf (1994) in many places through statements and expressions such as:

“[...] the two forms of judgement $a : \alpha$ and $a = b : \alpha$ are both analytic.” (p.93)

“The synthetic form of judgement is precisely the existential form of judgement that I have just introduced.” (p.94)

“[...] it is only [A is true] which is a form of synthetic judgment.” (p.96)

“[...] a judgement of one of the two analytical forms [...]” (p.97)

To my mind, this is an inaccurate way of phrasing the analytic-synthetic distinction. If analyticity is indeed correctness by virtue of the meaning of expressions in a judgment, as Martin-Löf insists with his allusion to the “current formulation”, then it can never be predicated of forms of judgments. The expressions that occur in a form of judgment are metavariables that have no meaning. Forms of judgments are not even bearers of correctness, so phrases like the “synthetic form of judgment” or “form of analytic judgment” reveal a misleading *façon de parler*. To offer a charitable reading of Martin-Löf’s claim, one could say that all judgments that have the forms $a : A$ and $a = b : A$ are analytic and all judgments having the form A true are synthetic. I believe this is the most profitable way of understanding his claim, so this is how I will interpret his distinction in the rest of this paper. Yet, his terminology of analytic and synthetic forms indicates that the form takes priority over the content of a judgment in questions of analyticity and syntheticity.

In this regard, Martin-Löf breaks with the tradition of Kant, Frege, Carnap and others, where two judgments may share the same form but still have different analytic or synthetic status depending on their content. For instance, in Kant's (1781, B11) view, you can have the judgment "all bodies are extended" as analytic and "all bodies are heavy" as synthetic, even though both ultimately share the same form S is P . Kant says that the former is analytic because one needs only to analyze the concept of a body to find extension therein. In the latter we cannot find heaviness as a predicate no matter how hard we analyze the subject. These examples indicate that, generally, to determine analyticity or syntheticity for a judgment having the form S is P in Kant we must first analyze the content of the judgment, which in this case is the particular subject S and predicate P involved in the judgment. And in Frege's writings, to give another example, we can see that some judgments having the same form A true can be analytic while others are synthetic, as he believed that the truths of arithmetic were analytic but those of geometry were synthetic (Frege, 1884, §§89–90). The distinction does not rest on the form of the judgment. This suggests that the analytic-synthetic distinction has been traditionally conceived as a contentual one: a judgment is analytic or synthetic just in virtue of the meanings of the expressions contained in it and not its form.

Unlike the philosophical tradition, Martin-Löf's distinction is purely formal. What decides whether a judgment is analytic or synthetic is not the meanings of the expressions involved but the form the judgment has, A type, $A = B$ type, $a : A$, $a = b : A$, or A true. If two judgments share the same form they are either both analytic or synthetic. This is a questionable approach, for one cannot adhere to the "current formulation" and focus on the meaning of expressions while endorsing this formal attitude. When a judgment is analytic, in Martin-Löf's view, it is due to having one of his two "analytical forms", $a : A$ or $a = b : A$. So how can its analyticity be derived from the meaning of the expressions? To properly adhere to the "current formulation" we must first examine what the expressions occurring in a judgment mean and then attribute analyticity or syntheticity to it depending of their meaning. Note that this is precisely the approach of Definition 1, where meaning is explained as evaluation. My definition of analyticity and syntheticity is thus well aligned with the tradition. However, it remains to be seen why meaning should be understood as evaluation.

4.2 Meaning as evaluation

In Section 2 we discussed Martin-Löf's claim that all judgments of forms $a : A$ and $a = b : A$ are analytic because the correctness of a judgment of these forms is "solely by virtue of the meanings of the terms that occur in it" (Martin-Löf, 1994, p.93). Unfortunately, this claim is presented without any accompanying argument in support of it. Even worse, it has an alarming air of vagueness, as it is unclear what it is for a judgment to be correct solely by virtue of the meaning of the expressions occurring in it. What is the meaning of an expression? To support Martin-Löf's claim this notion of meaning of expression must be clarified.

This matter is not settled in the meaning explanations. They say what a judgment means, but tell nothing directly about the meaning of an expression occurring in a judgment. This is a gap that we must fill ourselves to understand the imprecise account of analyticity Martin-Löf appeals to in his claim that all $a : A$ and $a = b : A$ are analytic. We shall do this as follows. There are two primary dimensions of meaning for expressions, sense and reference. Which one is most fitting to clarify the notion of meaning Martin-Löf needs in his claim?

It is a commonly accepted view in constructive semantics that the sense of an expression is the program it expresses when endowed by an operational semantics and that its reference is determined by evaluation as the value of the program it expresses. This program-value interpretation of sense and reference was suggested by Dummett (1978, p.133). His ideas were further developed in a constructive setting in 1992 at a talk delivered in Leiden whose transcript was published recently (Dummett, 2021). In 2001, his interpretation was refined and adapted to constructive type theory by Martin-Löf (2021).⁶ For our purposes, it will suffice to mention that the reading of meaning as sense emphasizes the way the program expressed by an expression computes its value. The reading of meaning as reference stresses the value output by the program, regardless of how it was computed. For example, the senses of $2 + 2$ and 4 are different but they have the same reference, as both evaluate to 4 . As we shall see, the referential reading of meaning squares better with the conceptualization of analyticity presupposed by Martin-Löf and it is thus the notion of meaning we shall adopt.

Referentially, the meaning of any expression M definable in our programming language is understood by means of the evaluation $M \downarrow M'$ in accordance to the operational semantics we have already prescribed in Section 3.1. Therefore, to say that a judgment of the forms $a : A$ or $a = b : B$ is analytic would be the same as saying that when it is correct, its correctness is entirely due to the evaluation of the expressions a and A , or a, b, A , and B , respectively. This referential reading seems to be implicit, for example, in Sundholm (2004):

$[a : \alpha \text{ and } a = b : \alpha]$ have the required analytic character. It is enough to have a and α in order to decide whether $a : \alpha$ (and similarly for judgements of equality. For both kinds of judgement, the means of decision utilizes evaluation to, and inspection of, relevant canonical forms). (Sundholm, 2004, p.456)

Under the referential reading, Martin-Löf's claim that all judgments of the forms $a : A$ or $a = b : B$ are analytic has in common with the meaning explanations given for these forms of judgment the fact that their correctness is a consequence of evaluation. This may seem to count as evidence in favor of his claim. However, upon closer inspection, although a few correct judgments of the forms $a : A$ or $a = b : B$ indeed come out as analytic, some do not. This can be observed after a more detailed examination of the meaning explanations.

4.3 Revisiting Martin-Löf's claims

There is a number of simple cases in which analyticity is easy to see. The correctness of several judgments of the form $a : A$ or $a = b : A$ can be determined after some reflection on the meaning of the expressions involved, that is, by evaluating them. I will focus mainly on the type of natural numbers here because the computation of arithmetical expressions is a simple task we are all sufficiently familiar with. Take, for example, the correct judgment $0 : \text{nat}$. From the meanings of 0 and nat alone it can be concluded that $0 \downarrow 0$ and $\text{nat} \downarrow \text{nat}$. This is all that is needed to know to immediately check that the judgment $0 : \text{nat}$ is correct according to the zero clause from Section 3.1.2, given that this is a valid instance:

⁶A full discussion of sense and reference in type theory would take us well beyond the scope of this paper. I refer the reader to Bentzen (2022, 2020).

$$\frac{\text{nat} \downarrow \text{nat} \quad 0 \downarrow 0}{0 : \text{nat}.}$$

Similarly, we can show that any judgment $n : \text{nat}$ is analytic, where, for the sake of brevity, a numeral n is used as a short for a string of n -many successors $s(\dots)$ followed by 0 . The meaning of the expressions occurring in, say, $1 : \text{nat}$, is, given that 1 abbreviates $s(0)$, fixed by the evaluations $0 \downarrow 0$, $s(0) \downarrow s(0)$ and $\text{nat} \downarrow \text{nat}$, which include all expressions that have occurrences in the judgment. Then, since we have just seen that the correctness of $0 : \text{nat}$ follows entirely from the meanings of 0 and nat , we have all we need to show its correctness using the successor clause from Section 3.1.2 via the instance:

$$\frac{\text{nat} \downarrow \text{nat} \quad s(0) \downarrow s(0) \quad 0 : \text{nat}}{s(0) : \text{nat}.}$$

It is not hard to check the analyticity of $2 + 2 = 4 : \text{nat}$ and similar arithmetical judgments. Again, $m + n$ abbreviates $\text{natrec}(n, m, \lambda k. \lambda f. s(f))$. Correctness follows directly from the meanings of the expressions occurring in it because $2 + 2 \downarrow 4$ and $4 : \text{nat}$ is analytic. I take this as sufficient evidence that all judgments of the form $a : A$ and $a = b : A$ are analytic when A is nat . To be exact, this claim extends to any particular type A such that $A \downarrow \text{nat}$.

Simple cases like this in which all we need to do to determine correctness is a calculation from the meanings of the expressions were perhaps what Martin-Löf had in mind. But what happens when A evaluates to a more complex type like dependent function types?

Martin-Löf's claim that all $a : A$ and $a = b : A$ are analytic breaks down because it leaves out any type A that evaluates to a dependent function type, $A \downarrow (x : B) \rightarrow C$. This failure is due to the meaning explanation given for hypothetical judgments. I remarked in Section 3.1.2 that all meaning explanations involving dependent function types depend on the meaning explanation for hypothetical judgments. That is, in cases where $A \downarrow (x : B) \rightarrow C$, $a : A$ is correct only if $a \downarrow \lambda x. c$ and the hypothetical judgment $c : C (x : B)$ is correct. More generally, the correctness of $a = b : A$ follows only when $c = d : C (x : B)$ is correct, assuming that $a \downarrow \lambda x. c$ and $b \downarrow \lambda x. d$. To sum up, when A evaluates to a dependent function type $(x : B) \rightarrow C$, the judgments $a : A$ and $a = b : A$ are analytic only if so are the hypothetical judgments $c : C (x : B)$ and $c = d : C (x : B)$ they respectively presuppose. But the problem is that no hypothetical judgments can be analytic. Their correctness is never in virtue of evaluation alone but due to their intrinsic "extensional" nature: the fact that all resulting categorical judgments remain correct for all objects passed as parameters.

Recall from Section 3.2 that the meaning explanation for hypothetical judgments is given functionally, reducing, for instance, the correctness of

$$a : A (x_1 : A_1, \dots, x_n : A_n)$$

to the correctness of every categorical judgment

$$a[a_1/x_1, \dots, a_n/x_n] : A[a_1/x_1, \dots, a_n/x_n]$$

that holds for all

$$a_1 : A_1, \dots, a_n : A_n[a_1/x_1, \dots, a_{n-1}/x_{n-1}].$$

So, given how the correctness of multiple categorical judgments needs to be checked before we can establish the correctness of judgments involving dependent function types, we can see why a conceptual analysis by means of evaluation cannot be conclusive. In fact, we could never expect evaluation to determine the correctness of a hypothetical judgment in general, for, as we stipulated in Section 3.1, only closed expressions can be evaluated.

Let me now turn to a few concrete cases where any hope for analyticity is extinguished by the semantic stipulation for hypothetical judgments. Recall that we can, for example, draw the absurd conclusion that $1 : \text{nat} \rightarrow \text{nat}$ under the assumption that $x : \text{void}$. This is an instance of the judgmental *ex falso* rule validated in Section 3.2. Thus, we have

$$\lambda x.1 : \text{void} \rightarrow (\text{nat} \rightarrow \text{nat}).$$

But it does not matter how we try to analyze the meanings of the expressions involved. We will never be able to assert this judgment based on evaluation alone. Actually no further evaluation is possible since $\lambda x.1 \downarrow \lambda x.1$ and $\text{void} \rightarrow (\text{nat} \rightarrow \text{nat}) \downarrow \text{void} \rightarrow (\text{nat} \rightarrow \text{nat})$. Yet this information does not suffice to determine that $\lambda x.1 : \text{void} \rightarrow (\text{nat} \rightarrow \text{nat})$ is correct. The key issue is that its correctness depends on the correctness of $1 : \text{nat} \rightarrow \text{nat}$ ($x : \text{void}$). So, as indicated above, we need to justify its correctness with an extensional argument that checks the correctness of every resulting categorical judgment for all objects as parameters. The point is that we need to go beyond the meanings of the expressions involved.

One may wonder if the above example is not too specific since functions from empty domains correspond to vacuously true implications under the propositions-as-types interpretation. I believe a more common example of a function will provide a more general and illuminating picture of the failure of analyticity. The simplest example of such a synthetic judgment may be the judgment that the identity function is a function $\lambda x.x : A \rightarrow A$. It is correct, but its correctness does not follow from the evaluation of the expressions alone. Again, $\lambda x.x$ and $A \rightarrow A$ are already values, that is, $\lambda x.x \downarrow \lambda x.x$ and $A \rightarrow A \downarrow A \rightarrow A$. The judgment is only correct due to the trivial extensional argument that for any $a : A$ we have $a : A$.

The example above suggests that all judgments of the form $a : A$ are synthetic when A evaluates to a dependent function type. Judgments of the form $a = b : A$ can be synthetic too. What I am trying to convey can already be seen for unary arithmetic functions $\text{nat} \rightarrow \text{nat}$. Using the definition of addition by recursion on the second argument mentioned earlier, consider two extensionally equal functions, one that adds two to something $\lambda x.x + 2$ and one that adds something to two $\lambda x.2 + x$. The judgment that expresses their equality

$$\lambda x.x + 2 = \lambda x.2 + x : \text{nat} \rightarrow \text{nat}$$

is correct because so is the hypothetical judgment

$$x + 2 = 2 + x : \text{nat} (x : \text{nat}).$$

That is, for any closed number $n : \text{nat}$, $n + 2 = 2 + n : \text{nat}$ is correct. But, again, the equality of these two functions is not a consequence of the evaluation of the expressions involved. From the facts that $\text{nat} \rightarrow \text{nat} \downarrow \text{nat} \rightarrow \text{nat}$, $\lambda x.x + 2 \downarrow \lambda x.x + 2$, and $\lambda x.2 + x \downarrow \lambda x.2 + x$ alone there is no way to know that $\lambda x.x + 2 = \lambda x.2 + x : \text{nat} \rightarrow \text{nat}$. We must appeal to extensionality and check that the equality holds for every natural number $n : \text{nat}$. But, of course, this can never be checked analytically, that is, by means of evaluation alone.

In summary, upon closer interpretation and examination, Martin-Löf's assertion that all judgments of the forms $a : A$ and $a = b : A$ are analytic lacks a solid foundation. Some judgments of these forms are indeed analytic as he observes. But others are synthetic. All judgments of these forms are synthetic when A evaluates to a dependent function type.

4.4 Martin-Löf's missing judgments

Another problem with Martin-Löf's distinction is that it does not exhaustively apply to all judgments expressible in the language of constructive type theory. It neglects the status of hypothetical judgments and the forms A type and $A = B$ type, in particular. But, first of all, why should we expect that Martin-Löf's analytic-synthetic distinction be exhaustive?

One might attempt to counter my objection by noting that Kant (1781) originally only considered categorical judgments, so Martin-Löf is simply following the Kantian tradition. I hesitate to think this is a convincing analogy for two reasons:

- Martin-Löf already omits categorical judgments of the forms A type and $A = B$ type, which are needed to establish the correctness of all $a : A$ and $a = b : A$. As long as we adhere to the “current formulation”, viewing analyticity as correctness from the meaning of expressions, I do not see how one can claim that $a : A$ and $a = b : A$ are analytic without understanding the status of the more basic A type and $A = B$ type.
- Unlike in the Kantian background, the correctness of hypothetical judgments is a precondition for the correctness of some of Martin-Löf's categorical judgments. These are not only the instances of $a : A$ and $a = b : A$ we have considered before, where A evaluates to a dependent function type, but also those of A type and $A = B$ type. Again, there is no way to examine how these categorical judgments are correct solely by virtue of the meaning of their expressions without extending one's examination to the hypothetical judgments that their correctness presuppose.

The account proposed here is general enough to be applicable to any correct judgment. I just showed in Section 4.3 that all judgments $a : A$ and $a = b : A$ will fail to be analytic whenever $A \downarrow (x : B) \rightarrow C$, for their correctness is not verifiable purely by evaluation. It is implied in my previous considerations that every hypothetical judgment is synthetic. This is the real reason of the failure of analyticity for $a : A$ and $a = b : A$ when $A \downarrow (x : B) \rightarrow C$.

What about the typehood and type equality judgments? I trust that the reader can see that $\text{void} = \text{void}$ type and $\text{Id}(\text{nat}, 0, 0)$ type are analytic, for example. Is it possible for a judgment of these forms to be synthetic? Yes, due to the syntheticity of hypothetical judgments, A type and $A = B$ type will be synthetic whenever $A \downarrow (x : B) \rightarrow C$. One simple example of a synthetic judgment of one of these forms is $(x : \text{nat}) \rightarrow \text{Id}(\text{nat}, x, x)$ type. Its correctness owes to the correctness of the hypothetical judgment $\text{Id}(\text{nat}, x, x)$ type $(x : \text{nat})$.

4.5 Decidability and incompleteness

Perhaps my demonstration of the failure of analyticity for some judgments of the forms $a : A$ and $a = b : A$ may be taken as evidence that a completely distinct characterization of meaning of expression underlies Martin-Löf's (1994) claim. So, in his defense, it may be objected that, since his conclusions are erroneous under our referential reading of meaning as evaluation, he must have endorsed a different account of analyticity.

The closest thing to an argument that could provide a justification for his claim of analyticity is found in his discussion of decidability (Martin-Löf, 1994, pp.97–99). This is not the first time analyticity has been connected with decidability in the philosophical tradition. Indeed, Wittgenstein thought that we should be able to decide by a mechanical calculation of symbols alone whether a proposition is analytic, as Sundholm (1990, pp.59–60) notes. What is special about Martin-Löf's view is that the analytic status he attributes to $a : A$ and $a = b : A$ is intimately related to our possession of a type-checking algorithm which can decide whether or not they are derivable in a fixed formal system for constructive type theory.

Clearly, the meaning of an expression has to be understood along the same lines if this connection between analyticity and decidability is to be taken seriously. Martin-Löf does not explain in what sense of “meaning” the decidability of $a : A$ and $a = b : A$ leads to correctness in virtue of the meaning of the expressions that occur in them. But one thing we can say for sure is that meaning has to be understood in terms of the information needed to decide whether these judgments are derivable or not. For example, we can say that the meaning of an expression consists of all the axioms and rules of inference of the formal system in consideration that say something about this expression. Needless to say, making this approach to meaning fully precise requires a bit more work than this. But, no matter how much we refine the idea, the formal line of reasoning it hinges on is too problematic. The correlation with decidability inevitably put matters of analyticity at the level of the particular formal system our type-checking algorithm is based on. Nevertheless, the analytic-synthetic distinction is supposed to have a semantic, as opposed to a formal nature.⁷

⁷The distinction drawn here also has strong epistemic overtones carried over from the meaning explanations.

Since the intended semantics of constructive type theory is the meaning explanations, the analytic-synthetic distinction must be rooted in reflections on the meaning explanations and not on the particulars of one formal system they justify. My proposed account of analyticity in terms of evaluation conforms to this principle, but the one discussed here violates it. Its formal approach is incapable of offering a sustainable answer because the same meaning explanations can be used to justify vastly different formal systems (e.g. Martin-Löf (1975, 1982)). All of them are imperfect approximations of the meaning explanations. They can never portray the full picture due to Gödel’s incompleteness theorems. Once the set of all correct judgments encompasses all arithmetical truths, as it should under the standard interpretation of the natural numbers in the meaning explanations, it ceases to be recursively enumerable. The totality of correct judgments outstrips a single body of axioms and rules of inference.⁸

Each one of these different formal systems for constructive type theory simply isolates a limited range of semantic properties of the meaning explanations it emphasizes. That is, from a certain formal viewpoint, if we are careful enough to see that only the “right” axioms and rules of inference are included in the proof system, the result will indeed be a formalism in which $a : A$ and $a = b : A$ are decidable (Martin-Löf, 1975). However, decidability only holds because the axioms and rules of inference that we chose to accept are incapable of fully capturing this rich aspect of the meaning explanations.

As it was seen in Section 3, the judgments $a : A$ and $a = b : A$ are undecidable from the semantic perspective of the meaning explanations. It is thus to be expected that these judgments will be undecidable in a different but still justifiable formal system. One such a formalism with undecidable type-checking is outlined in Martin-Löf (1982) with the adoption of the reflection rule as the elimination rule for the equality type.

The greatest challenge for this tentative formal account of analyticity as decidability in some formal system is how to decide which axioms and rules of inference count as the “right” ones when other rules that lead to undecidability are validated by the meaning explanations. As far as I can see, a proper defense would involve abandoning the meaning explanations completely in favor of another intended constructive semantics or at least restricting its decisive role in the justification of type-theoretic formal systems to some extent.⁹

One could, for instance, maintain that formal systems where $a : A$ and $a = b : A$ are decidable have a privileged status for some other reason extraneous to the meaning explanations. To be fair, I am willing to concede that decidability of type-checking is a useful feature to have for practical purposes like formal proof verification in the mechanization of mathematics.

⁸The germ of incompleteness can be found in Brouwer’s early writings from 1905–1907. In his student notebooks he explicitly states that the totality of all mathematical theorems is “denumerably unfinished” (Brouwer, 1905–1907, VIII-44, p.308). In the dissertation, the same assertion is made for the totality of all possible constructions (Brouwer, 1907, p.149). Carnap reports that Gödel’s work on the incompleteness theorems was stimulated by Brouwer’s conviction that mathematics is not formalizable (Wang, 1990, p.84). Likewise, the meaning explanations of constructive type theory are not entirely formalizable. In the words of Robert Harper, “constructive mathematics is not metamathematics” (Harper, 2013).

⁹The latter path is taken in Klev (2019, §10) to justify the omission of some eta-rules in a formalism.

Perhaps Kreisel’s (1962) dictum that we should recognize a proof when we see one best expresses why decidability is held in high regard by many as an ideal proof-theoretic feature. Implicit in this welcoming attitude toward decidability is a rough treatment of $a : A$ as “ a is a formal proof of the theorem A ”. Is there a chance that with some refinements to the idea this could give rise to a different intended semantics for constructive type theory that could rival the meaning explanations? Well, worries similar to those expressed on page 9 would arise, considering that it is hard to think of judgments not restricted to logic such as $2 + 2 : \text{nat}$ as formal proofs of the theorem nat . Such judgments are unproblematic in the meaning explanations where all objects are programs and $a : A$ means “ a is a program with values in A ”. Though every formal proof can be regarded as a program by the very finitistic nature of a formal system, not every program should be identified with a formal proof. Until a solution to this problem is found, it is unclear how this treatment of objects as formal proofs could lead to an interpretation for all types considered in constructive type theory.

4.6 Analytic and synthetic grounding

Finally, I want to emphasize that Martin-Löf’s (1994) view that every synthetic judgment of the form A true is “grounded on” an analytic one is false under our proposed account. I do not dispute his conclusion that all A true are synthetic, since it is not possible to establish their correctness with some $a : A$ only by evaluating A . But, as we have seen, it is possible that the judgment $a : A$ that makes A true correct is synthetic. This will happen in any truth judgment involving the dependent function type. To give an example, notice that we have both synthetic judgments in the premise and the conclusion of the following valid rule:

$$\frac{\lambda x. \lambda y. x : A \rightarrow (B \rightarrow A)}{A \rightarrow (B \rightarrow A) \text{ true.}}$$

So we end up with two classes of synthetic judgments of the form A true, namely, those that can be grounded on an analytic judgment $a : A$ and those that cannot. In this sense, judgments of the first class are more fundamentally synthetic than those of the second class. Is this subtle difference between so-to-speak “synthetic-synthetic” and “synthetic-analytic” judgments of the form A true able to capture any useful philosophical distinction? I believe so, though for reasons of space I can only offer a short example to convince the reader. Via the propositions-as-types paradigm, my proposal allows us to distinguish the syntheticity of any logical laws saying something about implication or universal quantification from that of simpler judgments like nat true or $\text{ld}(\text{nat}, 0, 0) \text{ true}$ which have an analytic grounding. Martin-Löf’s account of syntheticity is unable to explain this difference.

5 Some possible objections

Before I conclude, I would like to address some possible objections and general concerns that may arise regarding the legitimacy of my criticism of Martin-Löf’s views and some implications of the analytic-synthetic distinction I proposed above.

5.1 Particulars of the meaning explanations

It may be objected that my critical remarks rely heavily on the particular formulation of the meaning explanations favored in Section 3. The version I presented is a variant of the original formulation of Martin-Löf (1982) that adopts an intensional explanation of type equality judgments. In his own meaning explanations, however, he explains type equality extensionally by saying instead that the judgment $A = B$ type is known whenever the knowledge of $a : A$ follows from the knowledge of $a : B$ and vice-versa. This extensional explanation is rejected in this paper for going against what I see as the most fundamental principle of the meaning explanations: the core idea that computational behavior determines meaning is entirely absent in the extensional account of type equality. That is, unlike its intensional counterpart I adopt, the meaning of $A = B$ type is not determined by taking into account how closed expressions are evaluated to their canonical forms. For this reason, under the extensional approach we cannot explain A type as $A = A$ type. Still, none of my arguments in Section 4 are contingent upon the adoption of intensional type equality.

I have also opted to follow the early meaning explanations of Martin-Löf (1982), wherein the elements of a set and proofs of a proposition are treated as objects of a type. This approach, commonly referred to as a lower-order formulation, is distinct from the later higher-order formulation proposed by Martin-Löf (1993), which employs a complex type structure where the elements of a set and proofs of a proposition are distinct from objects of a type. It also features a typed evaluation relation instead. This higher-order structure is assumed in the background of Martin-Löf's (1994, pp.91–94) distinction. I avoided it here due to its significant conceptual complexity, which I believe would complicate the presentation. Moreover, I am not entirely convinced the presupposition of a typed evaluation relation in an explanation of what it means to be a type is a coherent philosophical move to make. This is an issue worth exploring, but due to space constraints, I cannot do so here. Most importantly, my adoption of a lower-order formulation does not result in any loss of generality in my arguments. The only features of the meaning explanations I rely on in my refutation of Martin-Löf's claim that all $a : A$ and $a = b : A$ are analytic in Sections 4.3 and 4.5 are found in both the lower- and higher-order formulation. In particular, the explanation of the meaning of hypothetical judgments and thus the semantical undecidability of $a : A$ and $a = b : A$. The interested reader can compare (Martin-Löf, 1982, pp.164–166) with (Martin-Löf, 1993, pp.35–39).

5.2 Why the formal approach is left behind

Recall I accept Martin-Löf's claim that all judgments of the form A true are synthetic. If some A true were analytic, their correctness would follow from the evaluation of A alone. But to make this judgment one needs an object a such that the judgment $a : A$ is correct. As I reject Martin-Löf's formal approach in Section 4.1, I must add that the syntheticity of every judgment of the form A true is not a consequence of the form of judgment that they all share, but our failure to establish their correctness based only on an analysis of the meaning of A . I follow the contentual tradition of Kant, Frege and others, since two judgments having the same form can have different analytic or synthetic status according to my proposal. I see the type membership $0 : \text{nat}$ as analytic but $\lambda x.x : A \rightarrow A$ as synthetic, for instance. This is a result of the difference in meaning of the expressions occurring in them.

5.3 Typehood judgments cannot be synthetic

It may be objected that my claim that some judgments of the forms A type or $A = B$ type are synthetic is counterintuitive or even contradictory. To say that something is a type is, in some sense, to say that it is meaningful, but this, as I claim, may not always follow from an analysis of the meaning of the expressions. There is a confusion between the meaningfulness an expression is supposed to have when it is known to be a type and the notion of meaning of expression I pin down to understand analyticity with. When A type is correct, A is meaningful in that we know from the meaning explanations what it means for it to be a type. This cannot be the notion of meaning we invoke in our discussion of analyticity. If it were, we would be forced to say, for instance, that nat true is analytic. When one knows that nat is a type one knows how to construct a canonical object of it. Thus, from the meaningfulness nat has as a type one could conclude that $0 : \text{nat}$ and therefore that nat true . Under my view, the meaning of an expression is the value it expresses as a program.

5.4 Empirical judgments

Given that the status of identity judgments is a recurring concern in classical approaches to the analytic-synthetic distinction, one may wonder how the distinction proposed in this paper can be applied to identity judgments beyond the ideal realm of mathematics. For example, what is the status of a judgment such as $\text{Hesperus} = \text{Phosphorus} : \text{Planet}$? One problem is how to understand types of empirical objects and the correctness of judgments about them. Elsewhere Martin-Löf (2014, p.5) states that to explain what an empirical proposition is one has to first explain the notion of an empirical quantity. Instead of giving a definition he provides examples such as the eye color of some individual, the length of a table, or the number of geese that cackled on the Capitol a particular night in 390 BC. Martin-Löf adds that these empirical quantities all have their own types. For eye colors we have a type that contains brown, blue, green and so on. For lengths and geese we have natural numbers. We can then perhaps say that Mercury, Venus, Earth, Mars, Jupiter, Saturn, Uranus, and Neptune are all empirical quantities of the type Planet. However, a more serious problem is that it is less clear how we should evaluate empirical quantities. Martin-Löf (2014, p.5) contrasts calculation for mathematical objects with “experiments” for empirical objects in a broad sense. For example, taking a look at my eyes to tell what color they are, or using a ruler to measure the height of a table both count as experiments in his view. If we follow this line of reasoning then, because an experiment reveals that both Hesperus and Phosphorus turn out to be the planet Venus, we should take $\text{Hesperus} = \text{Phosphorus} : \text{Planet}$ as analytic. They are non-canonical empirical quantities of that type, so to speak. But to provide a comprehensive analysis of this proposal, a rigorous examination of empirical types is necessary. One might think a viable approach for refining the conception of experimentation could be the formulation of a general theory of intuition and fulfillment-chains in the sense of Husserl (1970, VI, §18). However, some strong objections to the idea that Husserl’s views on intuition are compatible with a constructivist interpretation have been raised recently (Bentzen, 2023a). In any case, further investigation of judgments outside of mathematics is beyond the scope of this paper.

6 Concluding remarks

I have proposed an analytic-synthetic distinction that remedies all the six problems with Martin-Löf’s proposal mentioned in Section 1. My account of analyticity and syntheticity is supported by the meaning explanations and a constructive reading of meaning as evaluation.

It would be important to investigate in future work to what extent the correctness of synthetic judgments requires appeal to intuition, a theme that was so central in Kant. For that we would first have to develop a notion of intuition for constructive type theory. With some modifications that can overcome a problem of ontological explosion raised in Bentzen (2023b), perhaps Brouwer’s theory of the intuition of twofoldness may be a promising start.

Acknowledgments I thank Ansten Klev, Antonio Piccolomini d’Aragona, Will Stafford, Göran Sundholm, Colin Zwanziger, and two anonymous reviewers for helpful comments on an early version of this paper. I am especially indebted to Bob Harper. My understanding of the meaning explanations and the computational approach to type theory has been shaped by many long conversations with him over the years. This work was funded by the grant 23YJRC04ZD-2YB from the Zhejiang Federation of Humanities and Social Sciences.

References

- Stuart Allen. *A non-type-theoretic definition of Martin-Löf’s types*. PhD thesis, Cornell University, 1987.
- Carlo Angiuli, Robert Harper, and Todd Wilson. Computational higher-dimensional type theory. *Proceedings of the 44th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL ’17*, 52(1):680–693, 2017. URL <https://www.cs.cmu.edu/~cangiuli/papers/chtt.pdf>.
- Bruno Bentzen. Sense, reference, and computation. *Perspectiva Filosofica*, 47(2):179–203, 2020. URL <http://philsci-archive.pitt.edu/17408/>.
- Bruno Bentzen. On different ways of being equal. *Erkenntnis*, 87(4), 2022. URL <https://doi.org/10.1007/s10670-020-00275-8>.
- Bruno Bentzen. Propositions as intentions. *Husserl Studies*, 39:143–160, 2023a. URL <https://doi.org/10.1007/s10743-022-09323-3>.
- Bruno Bentzen. Brouwer’s intuition of twofoldness and constructions in separable mathematics. *History and Philosophy of Logic*, pages 1–21, 2023b. URL <https://doi.org/10.1080/01445340.2023.2210908>. Online first.
- L.E.J. Brouwer. Student notebooks, 1905–1907. Digital transcriptions by John Kuiper available at: <http://www.cs.ru.nl/F.Wiedijk/brouwer/transcriptie.pdf>.
- L.E.J. Brouwer. *Over de Grondslagen der Wiskunde*. PhD thesis, Universiteit van Amsterdam, 1907. Translated in (Brouwer, 1975, pp.13–101).
- L.E.J. Brouwer. *Collected Works 1. Philosophy and Foundations of Mathematics*, 1975.
- R. L. Constable, S. F. Allen, H. M. Bromley, W. R. Cleaveland, J. F. Cremer, R. W. Harper, D. J. Howe, T. B. Knoblock, N. P. Mendler, P Panangaden, J. T. Sasaki, and S. F Smith. *Implementing Mathematics with The Nuprl Proof Development System*. Prentice-Hall, 1985. URL <https://www.nuprl.org/book/>.
- Michael Dummett. Sense and reference from a constructivist standpoint. *The Bulletin of Symbolic Logic*, 27(4):485–500, 2021. URL <https://doi.org/10.1017/bsl.2021.60>. Transcription of a talk given at Leiden University on September 26, 1992.

- Michael A. Dummett. Frege's distinction between sense and reference. In *Truth and Other Enigmas*, pages 116–144. Harvard Univ. Press, Cambridge, 1978. Original work published in 1975.
- Peter Dybjer. Program testing and the meaning explanations of intuitionistic type theory. In Peter Dybjer, Sten Lindström, Erik Palmgren, and Göran Sundholm, editors, *Epistemology versus Ontology*, pages 215–241. Springer, 2012. URL https://doi.org/10.1007/978-94-007-4435-6_11.
- Gottlob Frege. *Die Grundlagen der Arithmetik: eine logisch mathematische Untersuchung über den Begriff der Zahl*. w. Koebner, 1884.
- Robert Harper. Constructive mathematics is not metamathematics. <https://existentialtype.wordpress.com/2013/07/10/constructive-mathematics-is-not-meta-mathematics/>, 2013. Online. Accessed: 12-08-2022.
- William A. Howard. The formulae-as-types notion of construction. In J. P. Seldin and J. R. Hindley, editors, *To H. B. Curry: Essays on Combinatory Logic, Lambda-Calculus and Formalism*, pages 479–490. Academic Press, London, 1980.
- Edmund Husserl. *Logical Investigations (I/II)*. Routledge and Kegan Paul, London, 1970. translated by J. N. Findlay. Vol. I first published in 1900, vol. II in 1901.
- Immanuel Kant. *Kritik der reinen Vernunft*, 1781. Revised Second Edition published by Johann Friedrich Hartknoch in 1787. English translation in Paul Guyer and Allen Wood. *The Cambridge Companion to Kant's Critique of Pure Reason*. Cambridge University Press, 2010.
- Ansten Klev. Eta-rules in Martin-Löf type theory. *Bulletin of Symbolic Logic*, 25(3):333–359, 2019. URL <https://doi.org/10.1017/bsl.2019.21>.
- Georg Kreisel. Foundations of intuitionistic logic. *Logic, methodology, and the philosophy of science*, pages 198–210, 1962.
- Per Martin-Löf. An intuitionistic theory of types: predicative part. In H. E. Rose and J. C. Shepherdson, editors, *Logic Colloquium '73 : Proceedings of the logic colloquium, Bristol*, pages 73–118. North-Holland, Amsterdam, New York, Oxford, 7 1975. URL [https://doi.org/10.1016/S0049-237X\(08\)71945-1](https://doi.org/10.1016/S0049-237X(08)71945-1).
- Per Martin-Löf. Constructive mathematics and computer programming. In *Logic, methodology and philosophy of science, VI (Hannover, 1979)*, volume 104 of *Stud. Logic Found. Math.*, pages 153–175. North-Holland, Amsterdam, 1982. URL [https://doi.org/10.1016/S0049-237X\(09\)70189-2](https://doi.org/10.1016/S0049-237X(09)70189-2).
- Per Martin-Löf. On the meanings of the logical constants and the justifications of the logical laws. In *Atti degli incontri di logica matematica. Scuola di Specializzazione in Logica Matematica*, volume 2, pages 203–281. Università di Siena, Siena, 1985. URL <https://ncatlab.org/nlab/files/MartinLofOnTheMeaning96.pdf>. Reprinted in *Nordic Journal of Philosophical Logic*, 1(1), pp.11–60.
- Per Martin-Löf. Truth of a proposition, evidence of a judgement, validity of a proof. *Synthese*, 73(3):407–420, 1987. URL <https://doi.org/10.1007/BF00484985>.
- Per Martin-Löf. Philosophical aspects of intuitionistic type theory, 1993. URL <https://pml.flu.cas.cz/uploads/PML-LeidenLectures93.pdf>. Transcriptions of lectures given at Leiden University from 23 September to 16 December 1993.
- Per Martin-Löf. Analytic and synthetic judgments in type theory. In P. Parrini, editor, *Kant and Contemporary Epistemology*, volume 54 of *The University of Western Ontario Series in Philosophy of Science*, pages 87–99. Kluwer Academic Publishers, Dordrecht, 1994.

- Per Martin-Löf. Truth of empirical propositions, 2014. URL <https://pml.flu.cas.cz/uploads/PML-Leiden04Feb14.pdf>. Transcriptions of lectures given at Leiden University on February 4, 2014.
- Per Martin-Löf. The sense/reference distinction in constructive semantics. *Bulletin of Symbolic Logic*, 27(4):501–513, 2021. URL <https://doi.org/10.1017/bsl.2021.61>. Transcription of a talk given at Leiden University on 25 August, 2001.
- Göran Sundholm. Sätze der logik: An alternative conception. In Haller Rudolf and J. Brandl, editors, *Wittgenstein – Towards a re-evaluation, Proceedings 14th International Wittgenstein Symposium. Kirchberg am Wechsel, 13–20 August 1989*, pages 59–61. Verlag Holder-Pichler-Tempsky, Vienna, 1990.
- Göran Sundholm. Ontologic versus epistemologic: Some strands in the development of logic, 1837–1957. In *Logic and Philosophy of Science in Uppsala: Papers from the 9th International Congress of Logic, Methodology and Philosophy of Science*, pages 373–384, Dordrecht, Netherlands, 1994. Kluwer Academic Publishers. URL https://doi.org/10.1007/978-94-015-8311-4_25.
- Göran Sundholm. Antirealism and the roles of truth. In I. Niiniluoto, M. Sintonen, and J. Woleński, editors, *Handbook of epistemology*, pages 437–466. Kluwer Academic Publishers, Netherlands, 2004. URL https://doi.org/10.1007/978-1-4020-1986-9_11.
- Göran Sundholm. Containment and variation; two strands in the development of analyticity from Aristotle to Martin-Löf. In M. van der Schaar, editor, *Judgement and the Epistemic Foundation of Logic*, pages 23–35. Springer, 2013. URL https://doi.org/10.1007/978-94-007-5137-8_3.
- Anne S. Troelstra and Dirk van Dalen. *Constructivism in mathematics. Vol. I*, volume 121 of *Studies in Logic and the Foundations of Mathematics*. North-Holland, Amsterdam, 1988.
- Hao Wang. *Reflections on Kurt Gödel*. MIT Press, 1990.

School of Philosophy
 Zhejiang University
 Hangzhou, China
bbentzen@zju.edu.cn