Manuscrito
46(4) pp.1–47, 2023
10.1590/0100-6045.2023.v46n4.bb

Frege's theory of types

Bruno Bentzen

Abstract It is often claimed that the theory of function levels proposed by Frege in *Grundgesetze der Arithmetik* anticipates the hierarchy of types that underlies Church's simple theory of types. This claim roughly states that Frege presupposes a type of functions in the sense of simple type theory in the expository language of *Grundgesetze.* However, this view makes it hard to accommodate function names of two arguments and view functions as incomplete entities. I propose and defend an alternative interpretation of first-level function names in *Grundgesetze* into simple type-theoretic open terms rather than into closed terms of a function type. This interpretation offers a still unhistorical but more faithful type-theoretic approximation of Frege's theory of levels and can be naturally extended to accommodate second-level functions. It is made possible by two key observations that Frege's Roman markers behave essentially like open terms and that Frege lacks a clear criterion for distinguishing between Roman markers and function names.

1 Introduction

It is often claimed that the theory of function levels introduced by Frege (1962) in his two-volume magnum opus *Grundgesetze der Arithmetik* anticipates the hierarchy of types that underlies the simple theory of types developed by Church (1940).[1] The claim is roughly that a ground type of objects and a type of functions is present in the ideography. I believe the anticipation is articulated more clearly in the words of Martin-Löf:

> The first type structure in the modern sense was introduced by Frege. [...] He did not talk about types, of course, but using modern notation and terminology, we would say that he had a type structure, namely the type structure which is generated by the following clause: If $\alpha_1, ..., \alpha_n$ are types, then $(\alpha_1, ..., \alpha_n)$ is a type. In this clause, n is allowed to be $0, 1$, etc. In particular, for $n = 0$, we get the type $()$, which is Frege's type of objects. Then putting $()$ for each of $\alpha_1, ..., \alpha_n$, we get a type of the form $((), ..., ())$, which is the type of n-ary functions taking n objects into an object again. This can be repeated, of course, arbitrarily high up,

[1] The main proponents of this view are Quine (1955, p.148), Resnik (1965, p.330), Dummett (1973, pp.44–50), Martin-Löf (1993, pp.8–9), and Klev (2014, ch.2,§1.1). It should be noted, however, that an earlier anticipation is found in Schröder, as Church (1976, 407) and Quine (1955, p.148) observe. Landini (1998, §5.4) proposes a formulation of the formal system of *Principia Mathematica* as a simple theory of types.

> so we get, for instance, the important type $((()))$. A function of this type takes a unary function from objects to objects into an object, so it is the type of Frege's course-of-values operator, the Wertverlauf operator. (Martin-Löf, 1993, p.8)

It is instructive to contrast those two theories as follows. In Frege's theory of function levels, each function of one or two arguments is schematically categorized into first, second, and third levels depending on whether its domain of argument is strictly restricted to that of objects, first-level functions, or second-level functions, respectively. On the other hand, in Church's simple type theory only unary functions are formally part of the hierarchy but the formation of functions of multiple arguments is made possible by allowing functions to be values of other functions, in contrast with Frege. More importantly, higher functions are formed through iterations of the function type $\sigma \to \tau$ when σ itself is a function type. The claim above therefore implies that any function name in the expository language of *Grundgesetze* can be interpreted as a term of a function type in the sense of simple type theory. For the lack of a better name I shall call this view the 'function type interpretation'.

The main opponent of the function type interpretation is Church (1976). According to him, the view that Frege has a type of functions is based on a misunderstanding. Church objects that Frege's view of functions as incompleted abstractions that require completion prevents them from being regarded as actual objects or completed abstractions:

> With Frege a function is not properly an (abstract) object at all, but is a sort of incompleted abstraction. A function is *ungesättigt* – is like an incomplete symbol in that it requires something additional, argument or quantifier, to complete its meaning – but nevertheless partakes sufficiently of the nature of an object to be represented by a variable. The notion of a function as an actual object, a completed abstraction in intension, is not contemplated by Frege; [...] (Church, 1976, p.408)

We may thus say with Klev (2014, p.74) that Church opposes the view that a function in the ideography is an object of a type in the sense of simple type theory. However, Klev dismisses Church's accusations by noting that Frege allows his functions be the arguments to other functions and in the scope of quantification, concluding that it would be hard to deny that the functions that constitute Frege's theory of function levels can form a type.

Building on Church's criticism of the function type interpretation, my aim in this paper is to propose and defend an alternative interpretation of Frege's theory of function levels into simple type theory in which first-level function names are viewed as open terms in a context depending on free variables rather than as closed terms of a function type. This interpretation can be naturally extended to second-level functions following Church's formal account of abstraction in terms of rules of inference. Thus, I respond to Klev's challenge concerning the place of higher functions and quantification by interpreting second-level function names as inference rules from open terms as premises to closed terms as conclusions.

In other words, my proposed interpretation, which I call the 'open term interpretation', is developed without a type of functions, as opposed to the widespread function type interpretation of Frege's theory of function levels. Before I put forward the open term interpretation as a more accurate way of understanding Frege's theory of function levels in the simple type-theoretic setting of Church (1940), I will argue that the function type interpretation is problematic for two main reasons: it cannot properly accommodate function names of two arguments and it goes against Frege's firm conviction that functions are incomplete entities that are in need of completion. This is the negative part of the paper. My positive contribution is that to substantiate the open term interpretation we must delve deeper into Frege's accounts of judgment, functionality, and generality in *Grundgesetze*. The following are two important theses that we will establish along the way:

(i) Roman markers in the ideography behave like open terms in simple type theory;

(ii) Frege lacks a criterion for distinguishing between Roman markers and function names.

It is important to note that I am not proposing this type-theoretic rendering of the ideography as a portrayal of Frege's actual views on functions. This is certainly not the case. Still, as it draws out the utmost implications of his conviction that function names are incomplete expressions and sheds light on his confusion between Roman markers and function names, I believe this interpretation should be of interest in its own right. With the open term interpretation we will solve our initial worries that functions of two arguments and Frege's view of functions as incomplete entities cannot be type-theoretically represented. While it has some difficulties dealing with third-level functions, I believe it remains the most faithful approximation of a type-theoretic interpretation of Frege's theory of levels.

Against my undertaking in this paper it may be correctly observed that no function name actually occurs in the object language of the ideography. Instead, function names appear only in the expository language of *Grundgesetze*, while the language of the formal system only contains names of values of functions, which are names for objects.[2] The objection is that when '$\Phi(\Delta)$' appears as a name of an object in the ideography, for example, we cannot say that the expression 'Φ' appears as a name of a function in Frege's object language. For him only an expression requiring completion such as '$\Phi(\xi)$' can be a function name. Frege is explicit about this issue in a letter to Russell from 13 November 1904 to be discussed in Section 3.3. In sum, the problem in question here is that these incomplete names can never occur in the object language of the ideography since they require completion. This observation does not undermine the investigation of Frege's theory of function levels within the context of simple type theory that I aim to undertake in this paper. Frege's theory of

[2] See e.g. Landini (2012, p.19) and Cadet and Panza (2015, pp.16-17). Landini sees the expository language of *Grundgesetze* as a form of metalanguage since for him small Greek letters are metalinguistic variables. It is unclear to me whether Frege had a full grasp of the difference between object language and metalanguage in the same way that we do today. Perhaps the notion of metalanguage is even intelligible to him, given that, as Cadet and Panza note, it goes against his universalist conception of logic. Either way, the fact remains that the expository language of *Grundgesetze* should be distinguished from its object language. For Landini, adopting metalanguage instead of object language does not contradict this universalist concept of logic.

function levels clearly falls outside the object language of *Grundgesetze*, since it is about when his function names can be properly filled, and therefore so does my investigation of its nature. It should be noted that when I speak of the function names of the ideography in what follows, I am referring specifically to the names of functions assumed in the theory of function levels that operates behind the scenes in *Grundgesetze*. For example, if '$\Phi(\Delta)$' and 'Δ' are names in the object language, then '$\Phi(\xi)$' must be a function name in the expository language. The aim of this paper is to investigate how, if possible, the structure of Frege's theory of function levels can be analyzed within the framework of simple type theory. Still, one of the outcomes of my investigation is that Frege's design choice of admitting Roman markers in the object language (as we shall soon see in Section 4) but keeping function names only in the expository language is unjustified given that his views in *Grundgesetze* are deficient in providing a criterion for distinguishing Roman markers and function names.

The rest of this paper is structured as follows. In Section 2, I introduce some basic concepts from simple type theory, but only loosely following Church's original formulation. Instead my exposition borrows from the presentation of type theory in the tradition of Martin-Löf (1975) and focuses on the role of open terms and his distinction between functions in the old-fashioned and modern sense. I discuss in Section 3 Frege's views on functionality, his theory of function levels, and the function type interpretation, closing this section with four objections to the function type interpretation. To set the stage for the open term interpretation, it will be necessary to establish the theses (i) and (ii) above first. Section 4 is primarily a defense of (i) based on Frege's account of judgment and his use of Roman markers. Section 5 supports (ii) with an examination of Frege's views on functionality and generality. In Section 6 I put together all the elements of our defense of the open term interpretation and explain it in full for first-level and second-level functions, while also addressing its interpretative problem with respect to third-level functions. Section 7 offers some concluding remarks.

2 Open terms and functions in simple type theory

We begin with a brief overview of simple type theory. My exposition shall be based on the tradition of Martin-Löf (1975) with minor differences in terminology and notation. Simple type theory is a formal system restricting the operations of the untyped lambda calculus with the adoption of a type system with some ground types and a type former $\sigma \to \tau$ for functions that take arguments of type σ and result in values of type τ. We write the typing judgment that states that a is a term of type σ as '$a : \sigma$'. Saying that $a : \sigma$ is a judgment simply means that it may carry assertive force and be subject to rules of inference. When such a judgment is asserted it is written preceded by a turnstile symbol following Frege's notation:

$$\vdash a : \sigma.$$

As in modern logic, the turnstile symbol is not part of the object language of the theory. I will say that a judgment $a : \sigma$ is categorical when it is stated without any hypotheses. For the most part, however, a typing judgment takes a general hypothetical form, where a typed term depends on one or more typing assumptions on free variables. In this case it will be written as $a : \sigma\ (\Gamma)$, where Γ is a finite list of typed variables $x_i : \sigma_i$ called the context of the general hypothetical judgment. Notice that when the context of a general hypothetical judgment is empty it just boils down to a categorical judgment.

We also say that a typed term $a : \sigma$ is open when it occurs in a general hypothetical judgment and may depend on some of the variables declared in the context. A typed term occurring categorically is said to be closed, meaning only bound occurrences of variables. A term is open if it is not closed, so some terms may be called open regardless of whether the variables declared in the context actually occur in them. In fact, contexts can be extended freely and a term $a : \sigma$ formed under a specific context may also be said to depend on an additional typed variable. This property is known as the structural rule of weakening:

$$\frac{\vdash a : \sigma \; (\Gamma, \Delta)}{\vdash a : \sigma \; (\Gamma, x : \tau, \Delta).} \tag{1}$$

In the presence of an open term $b : \tau$ depending on a free variable $x : \sigma$, we may obtain a new closed term, when given a closed $a : \sigma$, by substituting a for x in b. This idea is generalized for arbitrary contexts with the substitution principle:

$$\frac{\vdash b : \tau \; (\Gamma, x : \sigma, \Delta) \qquad \vdash a : \sigma \; (\Gamma)}{\vdash b[a/x] : \tau \; (\Gamma, \Delta).} \tag{2}$$

Open terms have no meaning in isolation in the meaning explanations proposed by Martin-Löf (1982) as the intended interpretation of type theory. An open term is said to have type σ when it results in closed terms of type σ after all the typed variables occurring in the context of the general hypothetical judgment are substituted with closed terms of the same type.

2.1 The function type

The definition of the function type $\sigma \to \tau$ involves a well-established collection of rules of inference that we do not need to dwell on in this paper. The reader interested in a comprehensive exposition is referred to Nordström et al. (1990, §19.6). Here it will suffice to note that lambda abstraction, the rule that regulates the construction of functions as lambda terms $\lambda x.a$, acts on an open term a depending on a distinguished free variable x:

$$\frac{\vdash a : \tau \; (\Gamma, x : \sigma)}{\vdash \lambda x.a : \sigma \to \tau \; (\Gamma).} \tag{3}$$

As pointed out by Quine (1940, p.114), this technique is essentially a revival of Frege's distinctive approach to abstraction using value-ranges. In a dictionary entry on abstraction, Church (1942) claims that the smooth-breathing notation $\dot{\varepsilon} f(\varepsilon)$ introduced by Frege to describe value-ranges is one of the precursors of the lambda-notation $\lambda x.a$ he uses for the function terms that inhabit the function type. This is hardly a coincidence, given that, according to Church (1976, p.408), who, again, denies that Frege has a type of functions, it is the value-range that "corresponds to the notion of a function as used in mathematics".

Another point worth contrasting with Frege, to be explored in Section 3, is that although the function type only gives us functions of one argument, functions of multiple arities can be defined in simple type theory through multiple unary functions by means of "Currying", a technique of dispensing with functions of multiple arguments by allowing functions to have other functions as values (Schönfinkel, 1924, §2). For example, the type of binary functions with domains σ and τ and codomain ρ is $\sigma \to (\tau \to \rho)$, meaning that to apply such a function one has to first apply it to a term of σ and only then to a term of τ.

2.2 Functions in the modern and old-fashioned sense

Martin-Löf (1993, pp.37–38) contrasts two notions of function in type theory, noting that functions in the old-fashioned sense are given by open terms in a type $a : \tau \ (x : \sigma)$ and functions in the modern sense given by closed terms in a function type $f : \sigma \to \tau$.[3] The former notion of function is called so because it is related to an old practice, introduced in the 18th century by Bernoulli and Euler, and followed by Lagrange, Fourier, and Boole, among others, of incorporating the argument places in the body of the functional expression. The reader is directed to Rüthing (1984) for a useful chronological compilation of some definitions of the concept of function from Bernoulli to Bourbaki and to Martin-Löf (1993, pp.50–53) for an extensive historical account of the concept of function. From Martin-Löf's perspective, functions in the old-fashioned sense are characterized by their dependence on arguments, examples of which are the functions given as $sin(x)$ or $(2 + 3x^2)x$.

In contrast to the old-fashioned practice, today we often define $f(x) = (2 + 3x^2)$ and then refer to f as a function without its arguments, for example. Functions in the modern sense are characterized by the absence of arguments. Martin-Löf (1993, p.52) attributes the modern notion of function to Dedekind and his definition of a function f as a law that transforms an object a into an object $f(a)$, in which a function is viewed as a self-subsistent mathematical operation that transforms every object into a value. It should be stressed that this modern concept of functions is implicit in Church's account of functions as "completed abstractions". In the modern sense we say that sin or $\lambda x.(2 + 3x^2)x$ are functions instead. Martin-Löf also adds that it is not clear whether Frege can be said to pursue the old-fashioned or modern notion of function in *Grundgesetze* (Martin-Löf, 1993, p.53). Since, as we will see in the next section, Frege views in *Grundgesetze* §1 a function expression as the part that remains when the argument place is not present in the expression, Martin-Löf maintains that Frege was actually aiming at the modern definition of a function. But he acknowledges that Frege writes function names in the old-fashioned manner, thus concluding that Frege did not have a satisfactory way of denoting functions in the modern sense:

> When Frege wants to have an expression for such a function, he does not do what we now are doing, say define $f(x, y) = x^2 + 3y$ and then speak of f as the function in the modern sense. This is the natural thing to do now, but Frege does not have a satisfactory notation for functions in the modern sense. The way he handles it is by introducing these special Greek symbols, ξ and ζ, and he takes $\xi^2 + 3\zeta$ to

[3] This distinction has been discussed in Klev (2019, §4) and more recently in Klev (2023) in the context of the argument move and removal rules proposed by Martin-Löf (1993, pp.59–64) as alternatives to abstraction and application as inference rules for the function type.

> be the expression of the function. Now of course he has after all put these Greek letters into the argument places, so it does not have holes in it any longer, it just looks like the usual expression for an old-fashioned function. (Martin-Löf, 1993, p.53)

In Martin-Löf's terminology, the function type interpretation states that Frege's functions must be understood in the modern sense. That is, a function name '$\Phi(\xi)$' in the theory of levels should be regarded as a closed term of some function type $\tau \to \sigma$. Under the open term interpretation to be defended here, however, Frege's functions must be taken in the old-fashioned sense, meaning that '$\Phi(\xi)$' is an open term of a type σ depending on a type τ. Before we can decide which interpretation should be rejected and which should be endorsed we must take a closer look at Frege's views on functionality.

3 Frege's theory of function levels

Now let me return to the discussion of Frege's theory of function levels. There was an influential view among Frege's predecessors that a function is an analytical expression distinguished by its dependence on variables. It can be found, for instance, in the writings of Lagrange and Boole, and, as can be seen in Rüthing (1984), traced back to Euler. In an attempt to avoid the pitfalls of formalism, Frege opposes the idea that functions are expressions already in *Grundgesetze* §1. Frege tells us that one will be tempted to rather say that a function is the meaning or reference of such an incomplete expression. But according to him the problem with this view is that in '$(2 + 3x^2)x$' we find a variable 'x', which does not refer to an object as '2' does, for example, but, as he says, "only indicates one indeterminately".[4]

The argument places of a function name are from this point on are written with small Greek letters such as 'ξ' and 'ζ' throughout *Grundgesetze*. Therefore, such small Greek letters only indicate indeterminately. Landini (2012, §2.6) maintains that these Greek letters are metalinguistic devices used parametrically to facilitate the rule of substitution. They are therefore not part of the object language of the ideography. As it will be seen in Section 4, Frege will also speak of Roman letters as signs that only indicate something indeterminately later in *Grundgesetze* §8. For Frege, these letters have an entirely different purpose.

Because the argument places of a function name do not refer, Frege rejects the above proposal and concludes in *Grundgesetze* §1 that a function is in essence the connection established between the objects whose names we put for a variable and the objects that then appear as referents of the now complete expression. Assuming the old-fashioned convention that the name of a function must always carry an argument place, Frege writes that:

> The nature of the function reveals itself, rather, in the bond that it establishes between the numbers whose signs we put for 'x' and the numbers that then result as the reference of our expression [...] The nature of the *function* lies therefore in that part of the expression that is present without the 'x'. (*Grundgesetze* §1)

[4] "*nur unbestimmt andeutet*" (*Grundgesetze* §1).

3.1 Functions of first, second and third level

The view of functions expressed above is stated with greater precision as the criterion of referentiality proposed for function names in *Grundgesetze* §29. Frege states four separate criteria for first-level function names of one and two arguments, and second- and third-level function names of one argument. Again, all these function names only occur as part of the expository language of *Grundgesetze.* According to Frege, a function name is referential if it always results in a referential name when each of its argument places are filled with referential names of the appropriate level. First-level function names can only be filled with object names, complete expressions which do not in themselves carry argument places. Note that these names of objects actually occur in the formal language. Second- and third-level function names must be filled with first- and second-level function names respectively.

How do the level restrictions work in the ideography? Call a well-formed expression a name. Frege states in *Grundgesetze* §28 that for an expression to be a name, it must either be a primitive, defined, or a compound name. In the ideography, a compound name can only be formed out of a function name by the filling of its argument places with other names in accordance with the level restrictions and formation rules of *Grundgesetze* §30.

In total there are eight primitive names admitted by Frege in *Grundgesetze*, all of which are function names having their own levels. If we ignore for a moment the distinction between functions of one or two arguments and focus on their levels we can describe them as follows. The primitive first-level function names are those for the horizontal, negation, implication, identity, and definite article operator, and their object argument places are marked with the small Greek letters ‘ξ’ or ‘ζ’. Following Frege, I respectively write these names as:

$$\text{‘}{-\!\!\!-\!\!\!-}\,\xi\text{’},\quad \text{‘}{-\!\!\!\!\top\!\!\!\!-}\,\xi\text{’},\quad \text{‘}\begin{array}{l}{-\!\!\!-\!\!\!-\!\!\!\top\!\!\!-\!\!\!-\!\!\!-}\,\zeta\\ \quad\;\;\llcorner\!\!-\!\!\!-\,\xi\end{array}\text{’},\quad \text{‘}\xi=\zeta\text{’},\quad \text{‘}\backslash\xi\text{’}.$$

The primitive second-level function names are those for the first-order universal quantifier and value-range operator. They are written respectively as follows, where their first-level argument places are indicated with the small Greek letter ‘ϕ’:

$$\text{‘}{-\!\!\overset{\mathfrak{a}}{\smile}\!\!-}\,\phi(\mathfrak{a})\text{’},\quad \text{‘}\grave{\varepsilon}\phi(\varepsilon)\text{’}.$$

The only primitive third-level function name considered is that of the second-order quantifier. Its second-level argument place is marked with the small Greek letter ‘μ’. The letter ‘β’, on the other hand, binds the first-level argument place of the second-level function name of one argument used to fill the second-level argument place:

$$\text{‘}{-\!\!\overset{\mathfrak{f}}{\smile}\!\!-}\,\mu_{\beta}\mathfrak{f}(\beta)\text{’}.$$

As the ideography lacks primitive object names, it is only possible to form compound or defined object names. The formation of compound object names, in particular, is done by filling the argument places of second- and third-level function names in the appropriate way. This is precisely how names of truth values (sentences) and value-ranges, the only object names considered by Frege in the formal language, are formed in the ideography.

3.2 The function type interpretation

The theory of levels does prevent a function from having itself as argument, but the doctrine of value-ranges as objects allows self-application to enter through the back door by means of objects of the kind $f(\dot{\varepsilon} f(\varepsilon))$.[5] Quine (1955, p.148) claims that without this peculiar doctrine we cannot reproduce Russell's paradox in the ideography. He concludes that the theory of function levels anticipates, to some degree, the theory of types. Dummett (1973, p.50) states that the Frege's theory of function levels is essentially Russell's simple theory of types formulated in terms of Frege's notion of incomplete expressions. Resnik (1965, §2) maintains that the result of Frege's theory of function levels is a simple hierarchy of types for incomplete entities with differing kinds of incompleteness and, to make the idea precise, he proposes the inductive definition for Frege's types seen earlier in Martin-Löf (1993, p.8):

> To be more precise, we define types inductively as follows: α is a type; if $t_1, ..., t_n$ are types, then so is the ordered n-tuple $\langle t_1, ..., t_n \rangle$ ($n > 0$). The complete entities make up type α. An incomplete entity belongs to the type $\langle t_1, ..., t_n \rangle$ if and only if it may be completed by (and only by) entities of types $t_1, ..., t_n$ taken in that order. Thus, a function of two arguments of type $\langle \alpha, \langle \alpha \rangle \rangle$, for example, can only take as its first argument a member of type α (an object) and as its second argument, a member of type $\langle \alpha \rangle$ (a function). (Resnik, 1965, p.330)

More precisely, the claim is that the theory of function levels amounts to a hierarchy of simple types formulated in the old-fashioned kind, that is, in terms of Frege's view of functions as "incompleted abstractions", to borrow Church's terminology. When spelled out in detail the idea amounts to a function type interpretation. It states that the names occurring in *Grundgesetze* at the object- or meta-level can be understood in the setting of a simple type theory with a ground type of individuals ι and a function type $\sigma \rightarrow \tau$, with the following hierarchy of types reflecting the structure inherent to Frege's theory of function levels:

- Object names have type ι;
- First-level function names have type $\iota \rightarrow \iota$;
- Second-level function names have type $(\iota \rightarrow \iota) \rightarrow \iota$;
- Third-level function names have type $((\iota \rightarrow \iota) \rightarrow \iota) \rightarrow \iota$.

[5] See Frege's letter to Russell of 22 June 1902 (Frege, 1980a, pp.131–133).

The eight primitive names of the ideography described above, for instance, are interpreted as constant terms of the function type corresponding to their level. Thus, for example, we have, as functions of first-, second-, and third-level respectively, the terms

$$-\!\!\!-\!\!\!-\,\xi : \iota \to \iota, \quad -\!\!\!\top\!\!\!-\,\xi : \iota \to \iota, \quad \backslash\xi : \iota \to \iota, \quad \text{and}$$

$$-\!\!\overset{\mathfrak{a}}{\smile}\!\!-\,\phi(\mathfrak{a}) : (\iota \to \iota) \to \iota, \quad \dot{\varepsilon}\phi(\varepsilon) : (\iota \to \iota) \to \iota, \quad \text{and}$$

$$-\!\!\overset{\mathfrak{f}}{\smile}\!\!-\,\mu_{\beta}\mathfrak{f}(\beta) : ((\iota \to \iota) \to \iota) \to \iota.$$

3.3 Objections to the function type interpretation

Before I proceed to the defense of the open term interpretation in the next section, I wish to raise four objections to the function type interpretation stated above. The first two concern functions of two arguments and the challenges the function type interpretation faces in accommodating them. The other two concern the modern view of functions and how it contradicts Frege's belief that functions are incomplete entities that require completion.

1. I deliberately left the hierarchy of types above incomplete. I chose to only include functions of one argument because, as we shall now see, the function type interpretation fails to capture Frege's functions of two arguments properly. One may wonder if we could not simply assign these functions the type $(\sigma \times \tau) \to \rho$, where $\sigma \times \tau$ is the product type which is inhabited by ordered pairs of terms of types σ and τ. This seems to be implicit in the suggestion made by Resnik (1965) and Martin-Löf (1993) in their inductive definition of Frege's types. One problem with this approach is that it does not reflect Frege's process of double completion.

When introducing functions of two arguments in *Grundgesetze* §4, Frege explains that they stand in need of double completion given that a function of one argument is obtained after an initial completion and an object is obtained as a value only after a second completion:

> So far only functions with a single argument have been talked about; but we can easily pass on to *functions with two arguments.* These stand *in need of double completion* insofar as a function with one argument is obtained after their completion by one argument has been effected. Only after yet another completion do we arrive at an object, and this object is then called the *value* of the function for the two arguments. (*Grundgesetze* §4)

There is no particular order for the double completion, so Frege has to use different small Greek letters to make each argument place explicit. Given a function $f(\xi, \zeta)$ and an appropriate argument a, we always have two possible choices: either we apply a to the ξ-argument place and obtain $f(a,\zeta)$ or choose the ζ-argument place and instead obtain $f(\xi,a)$.

However, given a binary function of type $(\sigma \times \tau) \to \rho$ we can only perform a simultaneous application of both arguments together as a pair.[6] It is impossible to provide an initial completion with one argument and then obtain a function of one argument as the result. To get around this difficulty, one might then attempt to assign Frege's functions of two arguments the type $\sigma \to (\tau \to \rho)$ of binary Curried functions instead. Here one problem is that Curried functions have a fixed order of application while for Frege a function $f(\xi,\zeta)$ can be given an initial completion as either $f(a,\zeta)$ or $f(\xi,a)$. Type-theoretically, this is an impossible move because the technique of Currying leaves no room for ambiguity in the determination of the order of application. For $f : \sigma \to (\tau \to \rho)$, the two applications above may be represented as $f(a)$ and $\lambda x.(f(x))(a)$, respectively. This shows that the type of $f(\xi,\zeta)$ cannot possibly take the Curried form $\sigma \to (\tau \to \rho)$, for ξ- and ζ-arguments of respective types σ and τ. Otherwise we could only doubly complete $f(\xi,\zeta)$ in that order.

2. A more serious difficulty with the interpretation of Frege's function of two arguments as Curried functions is that the latter have functions as values, as stressed by Potts (1979). Curried functions are allowed to have a value after a first application but, as seen above, the value of a Fregean function of two arguments is only obtained after the second completion. So, Frege's functions of two arguments cannot be consistently typed with function types.

3. Frege does not give functions the self-subsistent treatment that is demanded by the notion of function in the modern sense, that is, as an object of the type $\sigma \to \tau$. This point is perhaps best made by Church (1976, p.408) in his criticism of Frege's view of function. We can perhaps say in Church's terminology that functions in the modern sense are properly "(abstract) objects" or "completed abstractions" characterized by their absence of arguments. They are given by closed terms without dependence on free variables. Due to their complete nature they can be objects of a type while functions in the old-fashioned sense cannot.

This should be taken as an argument against Martin-Löf conclusion that Frege was actually aiming at a modern definition of function. When Frege states in *Grundgesetze* §1 that a function is given by the part of an expression that is present without the argument place he simply means that the function has an incomplete nature. It is not suggested that a function can be regarded as a mathematical entity that is not in need of an argument. Thus, it seems to me that Frege adheres more closely to the old-fashioned conception of functions. But, as incompleted abstractions, old-fashioned functions cannot be objects of a type.

4. The function type interpretation implies that first-level function names are interpreted as terms of type $\iota \to \iota$, second-level function names as terms of type $(\iota \to \iota) \to \iota$, and third-level function names terms of type $((\iota \to \iota) \to \iota) \to \iota$. But considering that in type theory terms are either open or closed, there are two possible readings of the interpretation.

[6] Simons (2019) notes that Frege does articulate a notion of simultaneous application with ordered pairs in *Grundgesetze* §144, but for double value-ranges and not functions of two arguments.

Open terms seem the correct choice, for they arguably embody in type theory the notion of function in the old-fashioned sense embraced by Frege. However, if a function name '$f(\xi)$' is taken as an open term depending on a free variable 'ξ', its type cannot be $\sigma \to \tau$. Take the definite article operator as an example. In this case, $\backslash x : \iota \to \iota\ (x : \iota)$ would amount to an old-fashioned function that has a function of type $\sigma \to \tau$ as value when the argument place x is substituted by a closed term of type ι. This is not what we are looking for.

Now, if the function type interpretation is read by means of closed terms, we are led to the implausible view that terms with free occurrences of variables like $\backslash x : \iota \to \iota$ are closed. As a result, the function names of *Grundgesetze* would have to be interpreted in modern style. So, if we wish take the function type interpretation seriously, Frege's definite article operator would have to be given by the closed term $\backslash : \iota \to \iota$, for example. However, in a letter to Russell dated 13 November 1904, Frege entertains but ultimately rejects the idea of adopting a modern-style notation for functions. The notation proposed is very similar to the function abstraction notation used for value-ranges, except that the variable binder is indicated with a rough breathing mark, while in value-range names smooth breathing marks are used. So, to give an illustration, the proposal is to write a function such as power of two as '$\dot{\varepsilon}(\varepsilon^2)$' rather than '$\xi^2$' in his standard notation. But Frege says:

> [T]his notation would lead to the same difficulties as my value-range notation and in addition to a new one. For a range of values is supposedly an object and its name a proper name; but '$\dot{\varepsilon}(\varepsilon^2 = 1)$' would supposedly be a function name which would require completion by a sign following it. '$\dot{\varepsilon}(\varepsilon^2 = 1)1$' would have the same meaning as '$1^2 = 1$', and accordingly, '$\dot{\varepsilon}(\varepsilon^2 = 1) \supset$' would have to have the same meaning as '$\supset^2 = 1$', which, however, would be meaningless. '$\dot{\varepsilon}(\varepsilon^2 = 1)$' would be defined only in connection with an argument sign following it, and it would nevertheless be used without one; it would be defined as a function sign and used as a proper name, which will not do. (Frege, 1980a, pp.161–162).

It has been argued by Klement (2003, p.20), Nomoto (2006, p.22), Martin-Löf (2006, p.98) and Klev (2019, 2023) that in the passage above Frege alludes to an abstraction operation that in effect amounts to Church's lambda abstraction (3). Yet, Frege is reluctant to represent the application of a function name by means of juxtaposition because he thinks the lack of argument places in the body of the expression means that the function name can be used as a complete expression. Under Frege's syntactic analysis, it would have to be an object name. Frege insists that it is possible to use '$\dot{\varepsilon}(\varepsilon^2 = 1)$' in isolation as an object name given that it has no occurrences of argument places while '$\xi^2 = 1$' can be viewed as a function name since the expression needs completion. So, to sum up, this closed-term based reading of the function type interpretation is unacceptable to Frege.

Finally, to wrap up the discussion, it should be noted that if by a ground type we mean the type of complete entities, then Frege implies an identification between closed terms and terms of a ground type in his letter. A similar confusion between closed terms and object terms is reflected in Frege's views on the violation of the law of excluded middle, which, as recently discussed by Vanrie (2020), throw light on Frege's rejection of Russell's initial proposal of a theory of types. For Frege, there are no closed terms of the function type.

4 Roman markers in *Grundgesetze*

Since the function type interpretation is untenable, we shall now turn to the development and support of the open term interpretation. The basic idea is that a first-level function name such as '$\backslash\xi$' must be interpreted in the old-fashioned sense as $\backslash x : \iota \ (x : \iota)$. This idea requires careful elaboration and I will return to it on Section 6. First, we must discuss and overcome what we may see as the main challenge to this interpretation, namely, the fact that the role of open term is already played by the so-called "Roman markers" in the ideography. Thus, the open term interpretation conflates both function names and Roman markers into open terms. If function names and Roman markers were to serve different purposes in *Grundgesetze*, this move would be inconsistent with Frege's own views. I will be addressing this problem in the remainder of this section and the next. My answer will be that Frege has no criterion to distinguish between function names and Roman markers.

4.1 Roman markers as open terms

To begin with, let me explain why Roman markers behave as open terms in the ideography. Roman markers are incomplete expressions depending on Roman letters. They are not names but only expressions that convey generality. This straightforward interpretation that the ideography has Roman letters as free variables in the modern syntactic sense is found in Kemp (1998, §1), Klement (2001, p.38), Heck (2010, §3), and Landini (2012, p.32,§5.1).

We will soon see below that Roman markers can only result in names when all their Roman letters are substituted with appropriate names. Yet, Frege frequently states his axioms and theorems with Roman markers to emphasize their general applicability. This can be seen in many places, but especially in *Grundgesetze* §47 in his summary of the basic laws and in his proof of the derived laws in, for example, in *Grundgesetze* §§49–52. This usage is to be expected once we observe that these are actually axiom and theorem schemes and thus cannot concern names of the object language. There is no precise mention of Roman markers when the first three rules of inference of the system are exposed in *Grundgesetze* §§14-16. But it is already clear in *Grundgesetze* §17 from Frege's examination of how the Barbara syllogism fits in the ideography that his rules of inference need Roman markers (Klement, 2001, pp.38–40).

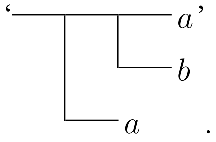
The axioms schemes and rules of inferences in simple type theory are, in the same way, formulated by means of open terms and their applicability is not limited to closed terms. When we think of the Roman letters of the ideography as typed variables and its names as closed terms, this property can be expressed type-theoretically by the substitution rule (2). Likewise, following Martin-Löf (1982), an open term is said to have a given type when it always results in closed terms of that type after all their typed variables are substituted with closed terms of the same type. Formally, the only noticeable difference between Roman markers and open terms is that the former always have occurrences of Roman letters but, due to weakening, free typed variables need not occur in the latter. Despite the converse not being true, every Roman marker in *Grundgesetze* can be viewed as an open term.

4.2 Roman letters and indication

In *Grundgesetze* §17 Frege, emphasizes that Roman letters are used to express generality in a judgment and have scope over the entire expression. Frege adds in a footnote that the “use of Roman letters is hereby explained only for the case in which a judgment-stroke occurs.” We may then say with Landini (2012) that each occurrence of the same variable is to be regarded as having the same referent once a referent is assigned and this holds throughout judgments of the whole proof. Frege stipulates that Roman letters do not refer but merely indicate:

> I shall call *names* only those signs or combinations of signs that refer to something. Roman letters, and combinations of signs in which those occur, are thus not *names* as they merely *indicate*. A combination of signs which contains Roman letters, and which always results in a proper name when every Roman letter is replaced by a name, I will call a *Roman object-marker*. In addition, a combination of signs which contains Roman letters and which always results in a function-name when every Roman letter is replaced by a name, I will call a *Roman function-marker* or *Roman marker* of a function. (*Grundgesetze* §17)

Here to say that a Roman marker indicates means that it always result in a name that refers to an object or function when all its Roman letters are replaced with appropriate names. Now, for example, we can say that the following expression is a Roman object marker because when ‘a’ and ‘b’ are replaced by object names the resulting names have objects as references:



This is not the first time Frege speaks of indication as opposed to reference for expressions. Recall that in *Grundgesetze* §1 Frege stresses that the argument places of a function name do not refer but only “indicate indeterminately”. The same thing is said of Roman letters in a preliminary discussion on generality in *Grundgesetze* §8. In §17 Frege states that both Roman and Gothic letters do not refer to an object but only indicate it. Klement (2001, p.36) thinks that to indicate for Frege is to reveal the kind of supplementation that is needed in an incomplete expression, and concludes that Roman and Gothic letters are semantically no different from Frege’s small Greek letters. I believe this is correct, though for reasons of space I will only focus on Roman and small Greek letters here. Frege sees the difference between a generality of an identity in ‘$\Phi(x) = \Psi(x)$’ and an identity in ‘$2^2 = 4$’ as manifested in the presence of a letter ‘x’ that “only indicates indeterminately” in the former, since every sign has a determinate reference in the latter. Roman letters not only merely indicate, they also indicate indeterminately, just as the small Greek letters do. In *Grundgesetze* §1 Frege explains that argument places of a function name indicate indeterminately because “[f]or different number-signs put in place of ‘x’ we generally obtain different references.” This is precisely what it means to say that a Roman marker indicates.

Now, the matter of the presence of Roman markers in the object language of the ideography is a delicate one, as Cadet and Panza (2015) emphasize. Unlike small Greek letters, Roman letters can occur in a judgment in *Grundgesetze*, so while function names are only part of the expository language, Roman markers belong to the object language. Even Roman letters indicating functions are allowed in the object language (Landini, 2012, §§2.1–6). The problem is that Roman markers are not accepted as names in *Grundgesetze*, and, as a result, do not correspond exactly to the terms of a modern formal system. In his reconstruction of the ideography as a modern predicate language, Landini avoids this problem by interpreting names as closed terms and allowing Roman markers to be open terms. This resembles the type-theoretic reading of Roman markers I am proposing here. However, Landini's approach gives rise to a new interpretative challenge as, since his analysis is grounded on a predicate language, he is led to interpret names preceded by Frege's turnstile sign as formulas (i.e. wffs) of a formal system (Landini, 2012, §2.1,§2.6). I will have more to say about the problem with this interpretation after an analysis of Frege's account of judgment.

4.3 Frege's account of judgment

Given that generality is expressed in a judgment, a few words on Frege's account of judgment may be in order. First of all, let me emphasize that all axioms and theorems of the ideography take the form of judgments that state that a thought is true. For Frege, a judgment is the acknowledgment of the truth of a thought, or, equivalently, the acknowledgment that the proposition expressed by a sentence refers to the True:

> with '$2+3 = 5$' only a truth-value is designated, without its being said which one of the two it is. Moreover, if I wrote '$(2+3 = 5) = (2 = 2)$' and presupposed that one knows that $2 = 2$ is the True, even then I would not thereby have asserted that the sum of 2 and 3 is 5; rather I would only have designated the truth-value of: that '$2 + 3 = 5$' refers to the same as '$2 = 2$'. We are therefore in need of another special sign in order to be able to assert something as true. To this end, I let the sign precede the name of the truth-value, in such a way that, e.g., in
>
> $$\text{`}\vdash\!\!\!— 2^2 = 4\text{'}$$
>
> it is asserted that the square of 2 is 4. I distinguish the *judgement* from the *thought* in such a way that I understand by a *judgement* the acknowledgement of the truth of a *thought*. (*Grundgesetze* §5)

I want to stress that "acknowledgment" is a success word. It is clear that for Frege a judgment "is not the mere grasping of a thought, but the admission of its truth" (Frege, 1892, fn.7). Therefore, in Frege a judgment is always synonymous with a correct judgment. This reveals that the turnstile sign of *Grundgesetze* has a double role of expressing a judgment while at the same time making the judgment expressed (Hare, 1970). Formally speaking, a judgment in the ideography consists of a sentence preceded of a turnstile sign:

$$\vdash\!\!\!— a.$$

Since sentences are names of objects, note that Frege does not respect the now standard distinction between terms and formulas in modern predicate logic. Landini (2012, §2.2) maintains that this orthodox reading is incorrect because, under his interpretation, formulas can be formed by appending a turnstile sign to terms of the ideography. So, he sees '$\vdash\!\!-\!\!-\, a$' as a formula whenever 'a' is a term of the formal language.[7] I am afraid this interpretation neglects the madeness associated with Frege's judgment stroke. Landini thinks to convey correctness with Frege's turnstile sign is to conflate matters of syntax with those of semantics in the ideography. To be fair, Landini admits that Frege can never be found attaching the turnstile sign to what he believes is a false sentence, but counters that "Frege only says that attaching a turnstile 'aims' to make a judgment" (Landini, 2012, p.31). Landini emphasizes Frege does express his false Basic Law V with a turnstile. So, in his view, a turnstile sign can flank not just names of the True but any term. To conclude, Landini (2012, p.34) contrasts the use of the turnstile sign '$\vdash$' in modern logic with Frege's use of his turnstile sign '$\vdash\!\!-\!\!-$'. The former is attached to a formula of the language to form a thesis (axiom or theorem) only expressible in the metalanguage, while, under his view, the latter is attached to a term to form a formula in the object language. Implicit in Landini's defense of his formula interpretation of judgments is the premise that Frege's turnstile sign is part of the object language. But as '$\vdash\!\!-\!\!-\, a$' goes beyond the formal theory by saying that the thought expressed by 'a' in the object language refers to the True, I see this as evidence that it is not part of the object language. If that is correct, there is no confusion between syntax and semantics, as Landini worries. The fact that Frege used a turnstile sign to express Basic Law V in *Grundgesetze* simply means, as Whitehead and Russell (1997, p.8) point out, that he can be convicted of error.

Moreover, if the madeness associated with the judgment stroke is denied, and Frege just meant to say that attaching a turnstile sign to a name "aims" to make a judgment, as Landini maintains, one may wonder how similar the roles of the turnstile in *Grundgesetze* and the horizontal in *Begriffsschrift* would be. Expressing that something can become a judgment was precisely the original role of the horizontal in *Begriffsschrift* §2. In the formal system of *Begriffsschrift* $-\!\!-\!\!-\, A$ expresses that "the content A is judgeable", which, from a purely syntactic perspective, is to say it is a formula (Bentzen, 2020, §2). Furthermore, given that all Frege has to say about the madeness of judgments in *Grundgesetze* §5 and *Begriffsschrift* §2 amounts to essentially the same, it seems to me that for a conclusive vindication of the formula interpretation of judgments one has to explain how it can tell the judgments and judgeable contents (formulas) of *Begriffsschrift* apart. One may respond that the formula interpretation of judgments only applies to *Grundgesetze* because in *Begriffsschrift* the judgment stroke already flanks a formula while in *Grundgesetze* it flanks a term. But that would be to overlook that Frege sees a sentence as a name of a truth value in *Grundgesetze.*

Back to the discussion of generality in *Grundgesetze* §17. Frege states that a Roman marker can be transformed into a universally quantified sentence, thereby explaining what seems to be the only purpose of Roman markers in the ideography, formally speaking. This transition is illustrated as a rule of universal generalization that allows us to pass from a judgment presented with a Roman marker of a truth value to a judgment presented with a sentence:

[7] So does Klement (2001, p.30), except that Landini's view is purely syntactic.

$$\frac{\vdash\!\!-\, \Phi(x)}{\vdash\!\overset{\mathfrak{a}}{\smile}\!-\, \Phi(\mathfrak{a}).} \tag{4}$$

Like any other rules of inference in the ideography, universal generalization operates on judgments as the premises and conclusions of the rule.[8] Therefore, Frege implies that we have a judgment in the premise of this rule. Despite Frege's implicit assumption, a second look reveals that the account of judgment outlined in *Grundgesetze* §5 is only applicable in the context of sentences, not Roman markers. How can we acknowledge the truth of a thought when a Roman marker, not being a sentence, neither express a thought nor refers? It is clear that a "judgment" of this kind has to be understood in a different way. What we can acknowledge here is that no matter how the Roman letters are replaced with names the resulting name always refers to the True. It is not an acknowledgment of a true thought but of true thoughts whenever appropriate names are given. I will argue in Section 5 that this is the explanation that Frege should have adopted for such judgments. It will be seen below, however, that a different account is hinted at in *Grundgesetze* §32.

Let me introduce some terminology for the sake of convenience. If we think of Roman markers as open terms and names as closed terms following Landini (2012), but use simple type theory as the framework of our investigation instead of the predicate calculus, then we can see that judgments in the ideography are also either categorical or general hypothetical. So, when referring to the ideography in what follows I call a judgment "categorical" if it is composed of a turnstile sign followed by a name of a truth value or "general hypothetical" if a turnstile precedes a Roman marker that indicates a truth value. To give a concrete example, these are categorical and hypothetical judgments of the ideography, respectively:

$$\vdash\!\!-\, \dot{\varepsilon}(\varepsilon = \varepsilon) \qquad\qquad \vdash\!\!\begin{array}{l} \!-\!\!\!-\!\!\!-\, f(x) = f(y) \\ \llcorner\!-\, x = y. \end{array}$$

Frege is well aware of the subtle distinction involved here and seems to use it quite skillfully. It is explicitly stated, for instance, in *Grundgesetze* §26 and §32, that a judgment in the ideography can only take the categorical and general hypothetical forms described above. Frege opens *Grundgesetze* §17 with a remark that Barbara cannot be derived from his previous rules of inference when the premises and conclusion are understood categorically in terms of universally quantified sentences, but only when represented general-hypothetically.

Since the account of judgment given in *Grundgesetze* §5 is not applicable to general hypothetical judgments, Frege attempts to justify their presence in the ideography in an alternative way in *Grundgesetze* §32. Therein it is observed that there are in fact only categorical judgments in the ideography as long as we consider that with the rule of universal generalization (4) a Roman marker can be transformed into a general sentence:

[8] It is worth noting that the ideography lacks a rule of conditional proof, which explains why Frege's universal generalization rule can be formulated without any limitations (Landini, 2012, p.33).

> Now, a concept-script proposition consists of a judgement-stroke with a name, or a Roman marker, of a truth-value. However, such a marker is transformed into the name of a truth-value when German letters are introduced for the Roman ones with concavities put in front, in accordance with §17. If we suppose this has been carried out, then there is only the case where the proposition is composed of the judgement-stroke and a name of a truth-value. (*Grundgesetze* §32)

It is unclear how to connect this passage with the view exposed in *Grundgesetze* §5 in terms of acknowledgments, but we may presume that with a general hypothetical judgment $\vdash\!\!-\!\!-\, \Phi(x)$ we acknowledge, although only indirectly, the truth of $-\!\overset{\mathfrak{a}}{\smile}\!-\,\Phi(\mathfrak{a})$ instead. One problem with this account will be discussed next in Section 5: Frege appears to incur circularity.

5 Functionality and generality in the ideography

What is the rationale behind Frege's stipulation that only function names but not Roman markers are allowed to refer as incomplete expressions? What criterion does Frege have in the expository language of *Grundgesetze* to tell them apart? They certainly differ in that the argument places of a function name are marked with small Greek letters while the incompleteness of a Roman marker is expressed with Roman letters. But to adopt this minor syntactic convention as the criterion for determining that function name may refer but Roman markers may only indicate would be unacceptable. It could be argued that function names only occur in the metalanguage but Roman markers are part of the object language. However, the question again arises as to what criteria are used to support his decision of doing so. If Frege's distinction between them is meaningful, a satisfactory explanation cannot simply rely on his arbitrary convention of using small Greek letters as metavariables.

To my mind, the problem is that both small Greek and Roman letters are attributed similar semantic roles as signs that "only indicate indeterminately". How do we distinguish those two kinds of incomplete expressions, function names and Roman markers, without begging the question by appealing to the very stipulation in question that Roman markers do not refer like function names but merely indicate? Frege does not provide a direct answer for this question, but the following criteria for the distinction may be drawn from his expository remarks on function names and Roman markers already discussed in the two previous sections:

- *functionality*: a function name expresses functionality as something that establishes a connection between the objects whose names we put for a small Greek letter and the objects that result as referents of the resulting name, as seen in Section 3;

- *generality*: a Roman marker expresses generality in a judgment with its Roman letters as something that can be transformed into a general sentence, as seen in Section 4.

In this section I will argue that Frege lacks a clear criterion for distinguishing between function names and Roman markers because his views on functionality apply to Roman markers and the same can be said of his views on generality applying just as well to function names.

First, let us note that the criterion of functionality laid down in *Grundgesetze* §1, and quoted and discussed in Section 3, applies to Roman markers. We have already seen in Section 4.2 that a Roman marker can relate the objects whose names we put for a Roman letter and the objects that appear as referents of the substituted expression. If Roman markers are not functional just as function names are, then functionality cannot be manifested in the part of an incomplete expression that remains without the variables, here either marked with small Greek or Roman letters, both of which, according to Frege, only indicate indeterminately. Why is '$\xi = \zeta$' functional but not '$a = b$', for instance, considering that both expressions result in referential names when their variables are replaced with referential names? The parts of the expressions that remain without the variables are syntactically identical, so we can say with Frege that the "nature of the function" must "lie" in both incomplete expressions. That is, it must be admitted, if we wish to conform to Frege's criterion of functionality, that both function names and Roman markers are functional in *Grundgesetze*.

Second, what can be said about the applicability of the generality criterion to function names rather than only Roman markers? Recall that the generality expressed by Roman markers is formally made precise with the transition (4) from *Grundgesetze* §17. Now consider a transition where now we have a passage from a function name to a general sentence, stating that given a function $\Phi(\xi)$ assumed to always have the True as a value for every argument, we can apply the second-level function $-\overset{\mathfrak{a}}{\smile}-\,\phi(\mathfrak{a})$ to obtain a true thought $-\overset{\mathfrak{a}}{\smile}-\,\Phi(\mathfrak{a})$:

$$\frac{\vdash\!\!-\!\!-\,\Phi(\xi)}{\vdash\!\overset{\mathfrak{a}}{\smile}\!-\,\Phi(\mathfrak{a}).} \tag{5}$$

Of course, this controversial transition is nowhere to be found in *Grundgesetze* and it is completely ungrammatical from the standpoint of its object language. Readers will no doubt object that we cannot see this as a transition on the same level as (4) because in (5) we do not actually have a judgment in the premise. But I believe it would be a mistake to think that we are in a better position to justify (4) as a transition. In its premise we have a Roman marker and $\vdash\!\!-\!\!-\,\Phi(x)$ is not a categorical judgment either. For this reason, Frege cannot justify (4) unless he offers in advance an explanation of what makes its premise a judgment.

This amounts to giving an account of general hypothetical judgments. But earlier in Section 4 we saw Frege defending in *Grundgesetze* §32 the presence of general hypothetical judgments in the ideography by reducing them to categorical judgments using the very transition (4) that transforms Roman markers into general sentences. This is a circular strategy. It attempts to explain the introduction of general hypothetical judgments by taking for granted that (4) is a valid transition; yet this implies that both its conclusion and its premise, which happens to take a general hypothetical form, are judgments of the ideography.

This is why I suggested in Section 4 that it is more promising to understand a general hypothetical judgment as an acknowledgment that an incomplete expression expresses a truth thought whenever it is filled with referential names of suitable levels. Surely $\vdash\!\!-\!\!-\,\Phi(\xi)$ can also be understood in those terms if the function has the True as value for every argument.

But even if one wants to stick to Frege's circular account of general hypothetical judgments I see no reason to accept (4) as a valid transition but not (5), unless one begs the question by contending that Roman markers are not function names. Given that both Roman markers and function names seem equally transformable into general sentences, generality cannot be a reliable criterion to distinguish between Roman markers and function names.

In sum, the claim that function names refer but Roman markers fail to do so is unfounded, given that Frege does not present any convincing criterion to distinguish between Roman markers and function names. By insisting on distinguishing function names and Roman markers, Frege creates a puzzling distinction between two classes of incomplete expressions. This observation paves the way for the open term interpretation: no conflict should arise in the reading of both function names and Roman markers as open terms. It also reveals that Frege could not be justified in keeping function names outside the object language and Roman markers in the object language of *Grundgesetze*.

6 The open term interpretation

The open term interpretation may be briefly described as the type-theoretic rendering of Frege's view of functions as incomplete entities. It embraces Church's (1976) reading of Frege's functions as incompleted abstractions and addresses Klev's (2014) concern about cases of functions being the arguments to other functions by giving an interpretation of second-level functions without function types. Now, the correlation between function names in *Grundgesetze* and open terms in simple type theory should be clear at this point. Following the functionality criterion laid down in Section 5, a function name in *Grundgesetze* must establish a connection between the objects whose names are put for a variable and the objects that are values of the complete expression. But, following Martin-Löf (1982), this is exactly what it means for an open term with explicit occurrences of variables to have a type.

Thus, if we were to view the primitive first-level function names of *Grundgesetze* as terms of a simple type theory, they would have to be open terms of a ground type and not closed terms of the function type. The resulting formalism would be a fragment of simple type theory that lacks a function type and the weakening property, but includes the following typing judgments, where ι is a type of individuals and o a type of truth values:

$$\vdash \text{———}\, x : o\ (x : \iota), \qquad \vdash \text{—⊤—}\, x : o\ (x : \iota), \qquad \vdash \begin{array}{l} \text{——┬——}\, y : o\ (x, y : \iota), \\ \qquad\ \text{└——}\, x \end{array}$$

$$\vdash x = y : o\ (x, y : \iota), \qquad \vdash \backslash x : \iota\ (x : \iota).$$

Moreover, because every truth value is an object, we require that:

$$\frac{\vdash a : o\ (\Gamma)}{\vdash a : \iota\ (\Gamma).}$$

Here open terms inhabit ground types and first-level functions are open terms whose argument places are designated by free typed variables. We must reject weakening because it goes against Frege's conviction that a function name must have explicit argument places. If we had this property it would not be possible to view open terms as function names. Given a closed term such as $\overset{,}{x}(\text{——}\, x) : \iota$ (a term for a value-range now written in simplified notation without small Greek letters), we could obtain, by weakening, an open term $\overset{,}{x}(\text{——}\, x) : \iota\ (y : \iota)$ that would otherwise be a function name of one argument, except that no argument places actually occur in the expression, in contrary to Frege's view.

Function application takes the form of substitution (2) of open terms. One benefit of this view is that application for functions of two arguments is handled in a way that resembles the method of double completion introduced in *Grundgesetze* §4. From $f(x, y) : \iota\ (x, y : \iota)$ and $a : \iota$ we may obtain either $f(x, a) : \iota\ (x : \iota)$ or $f(a, y) : \iota\ (y : \iota)$. Only after yet another completion, say, with $b : \iota$, we arrive at a closed term $f(a, b) : \iota$. Being a closed term, this is called the value of the function. Just as with Frege, there is no fixed application order for functions of two arguments, but we only arrive at a value after a double completion. As noted before, Curried functions fail to represent Frege's view in both respects.

Finally, it should be mentioned that under this open term interpretation the definition of new functions (in the old-fashioned sense) can be done by uniformly substituting complex function signs for object language function variables. This corresponds to Landini's (2012) interpretation of comprehension of functions in *Grundgesetze*. Substitution is made possible with the introduction of the following rule of inference, where a is free for x in f:

$$\frac{\vdash f[a/x] : \tau\ (\Gamma) \qquad \vdash a : \sigma}{\vdash f : \tau\ (\Gamma, x : \sigma).}$$

6.1 Second-level function names

The view of first-level functions as open terms bridges the gap between the approaches to function abstraction in the ideography and simple type theory. In both systems, the operation may be viewed as taking an open term depending on a distinguished free variable and resulting in a closed term. The main difference is that for Frege the abstracted term is in the ground type of individuals and not in a function type, unlike in (3). So instead we have:

$$\frac{\vdash a : \iota\ (x : \iota)}{\vdash \overset{,}{x}a : \iota.} \tag{6}$$

However, the reading of Frege's function names as open terms becomes less simple whenever higher order functions are concerned due to the absence of function types. Open terms are formed by means of general hypothetical judgments, and those cannot be part of other general hypothetical judgments in the same way that higher-level functions may take lower-level functions as arguments. Fortunately, there is a very natural way to circumvent our limitation and extend our interpretation to second-level functions. I have just explained the value-range operator as an inference rule that takes an open term in the type of individuals ι to a closed term in that type. The same can be said of first-order quantification using the type of truth values o. I now use modern notation for the sake of convenience:

$$\frac{\vdash f : o \ (x : \iota)}{\vdash \forall (x : \iota) f : o.} \tag{7}$$

Another way in which this is a natural interpretation is the following. Following the observation from Section 4 that Roman markers behave as open terms, (7) also interprets the transition (4) from Roman markers to general sentences proposed in *Grundgesetze* §17, revealing that Frege's distinction between Roman markers and function names is ultimately superfluous when seen through the lenses of simple type theory. Both are open terms that can be transformed into closed terms representing value-range or quantified propositions.

This interpretation sheds light on a noteworthy relation between function abstraction and quantification. Under a correspondence famously discovered by Howard (1980) between propositions and types, the terms of the function type can be regarded as proofs of implications. Just as simple type theory can be viewed as a generalized form of propositional logic, a more sophisticated framework known as "constructive type theory" (Martin-Löf, 1975) generalizes predicate calculus. The theory features dependent types, a constructive extension of Frege's conception of predicates as functions that assign objects to reified truth values. From a suitable open term of a type we can always either obtain a value-range term from (6) or a quantified term from (7), but never both in a single rule. In dependent type theory, this conflict is resolved by keeping function abstraction and quantification apart with one rule that at the same time emphasizes both the functionality and generality of an open term:

$$\frac{\vdash a : f \ (\Gamma, x : \sigma)}{\vdash \lambda x.a : \forall (x : \sigma) f \ (\Gamma).} \tag{8}$$

Under the propositions-as-types correspondence, quantification $\forall(x : \sigma)f$ can be seen as a generalized type of functions $(x : \sigma) \to f$ whose values for any $a : \sigma$ have type $f[a/x]$. Thus, f is an indexed family of types associating every term of a base type to another type. This also means that instead of $f(x) : o \ (x : \iota)$, in constructive type theory we require $f(x) : \mathcal{U} \ (x : \iota)$, where $\mathcal{U}$ is a universe type, a type whose terms are themselves types. Of course, I am not suggesting that Frege would be prepared to accept (8). I doubt he could even make sense of it as a rule of inference in his wildest dreams. All I wish to point out is that, since lambda terms are function terms, Frege's struggle with the distinction between function names and

Roman markers, which are both here open terms, is finally settled in constructive type theory by keeping functionality on the level of terms and generality on the level of types.

6.2 Third-level function names

One objection to the open term interpretation is that is unclear how the approach to second-level functions could be generalized to third-level functions because we obviously do not have something that operates on rules of inferences in type theory. Such an interpretation would therefore have to be studied at a metainferential level, a task which I am not willing to undertake here since I am not confident about its philosophical significance. The purpose of this paper is not to provide an entirely accurate representation of Frege's theory of function levels assuming simple type theory in the background. Instead, the objective is to analyze and compare the function type and open term interpretations in order to determine which one aligns more closely with Frege's theory of function levels.

7 Concluding remarks

I have discussed two representations of Frege's theory of function levels in simple type theory, namely, the function type and open term interpretations. The former states that a type of functions is presupposed at the level of the expository language of *Grundgesetze*. While this interpretation is very influential in the literature, it leaves out function names of two arguments and goes against Frege's view of function names as incomplete expressions. On the other hand, Frege's lack of criterion to distinguish function names from Roman markers and his account of functionality seem to favor the open term interpretation. Thus understood, *Grundgesetze* only presupposes a ground type of individuals ι and truth-values o assigned to every closed term in the language by virtue of the fact that they are complete expressions. There are technical complications in extending the open term interpretation to functions of third level, but it brings Frege's and Church's theories of abstraction together under one common framework for open terms and sheds new light on the type-theoretic rendering of the view that function names are incomplete expressions.

References

Bruno Bentzen. Sense, reference, and computation. *Perspectiva Filosofica*, 47(2):179–203, 2020. URL http://philsci-archive.pitt.edu/17408/.

Méven Cadet and Marco Panza. The logical system of Frege's Grundgestze: A rational reconstruction. *Manuscrito*, 38:5–94, 2015. URL https://doi.org/10.1590/0100-6045.2015.v38n1.mcemp.

Alonzo Church. A formulation of the simple theory of types. *The Journal of Symbolic Logic*, 5 (2):56–68, 1940. URL https://www.jstor.org/stable/2266170.

Alonzo Church. Abstraction. *The Dictionary of Philosophy, New York: Philosophical Library*, 1942.

Alonzo Church. Schröder's anticipation of the simple theory of types. *Erkenntnis*, 10(3): 407–411, 1976. ISSN 01650106, 15728420. URL http://www.jstor.org/stable/20010523. Originally presented at the Congress of the Unity of Science in 1939.

Michael Dummett. *Frege: Philosophy of language*. Harvard University Press, 1973.

Gottlob Frege. Über Sinn und Bedeutung. *Zeitschrift für Philosophie und philosophische Kritik*, 100:25–50, 1892. Translated in Frege (1980b, pp.56–78).

Gottlob Frege. *Grundgesetze der Arithmetik, Begriffsschriftlich Abgeleitet (I/II).* Georg Olms, Hildesheim, 1962. Translated in Frege (2013).

Gottlob Frege. *Philosophical and Mathematical Correspondence.* University of Chicago Press, Chicago, 1980a. Translated by H. Kaal.

Gottlob Frege. *Translations from the Philosophical Writings of Gottlob Frege.* Blackwell, Oxford, 3 edition, 1980b. Translated by Geach and Black.

Gottlob Frege. *Basic laws of arithmetic.* Oxford University Press (UK), 2013.

R. M. Hare. Meaning and speech acts. *Philosophical Review*, 79(1):3–24, 1970. URL https://doi.org/10.2307/2184066.

Richard G Heck. Frege and semantics. In Tom Ricketts and Michael Potter, editors, *The Cambridge companion to Frege.* Cambridge University Press, 2010.

William A. Howard. The formulae-as-types notion of construction. In J. P. Seldin and J. R. Hindley, editors, *Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism*, pages 479–490. Academic Press, London, 1980.

Gary Kemp. Propositions and reasoning in Russell and Frege. *Pacific philosophical quarterly*, 79:218–235, 1998. URL https://doi.org/10.1111/1468-0114.00059.

Kevin Klement. *Frege and the Logic of Sense and Reference.* Routledge, 2001. URL https://www.tandfonline.com/doi/abs/10.1080/0144534031000076237.

Kevin Klement. Russell's 1903-1905 anticipation of the Lambda Calculus. *History and Philosophy of Logic*, 24(1):15–37, 2003. URL https://doi.org/10.1080/0144534031000076237.

Ansten Klev. *Categories and Logical Syntax.* PhD thesis, Universiteit Leiden, 2014.

Ansten Klev. Name of the sinus function. In Pavel Arazim, editor, *The Logica Yearbook 2018*, pages 149–159. College Publications, London, 2019.

Ansten Klev. Spiritus asper, lambda, and the notion of function. *Notre Dame Journal of Formal Logic*, 64(2):205–223, 2023. URL https://doi.org/10.1215/00294527-10670089.

Gregory Landini. *Russell's hidden substitutional theory.* Oxford University Press, USA, 1998.

Gregory Landini. *Frege's notations: What they are and how they mean.* History of Analytic Philosophy. Pallgrave MacMillan, Basingstoke, 2012.

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.

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.

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. Comments on Prof. Kazuyuki Nomoto's Paper. *Annals of the Japan Association for Philosophy of Science*, 14(2):98–99, 2006. URL https://doi.org/10.4288/jafpos1956.14.98.

Kazuyuki Nomoto. The Methodology and Structure of Gottlob Frege's Logico-philosophical Investigations. *Annals of the Japan Association for Philosophy of Science*, 14(2):73–97, 2006. URL https://doi.org/10.4288/jafpos1956.14.73.

Bengt Nordström, Kent Petersson, and Jan M Smith. *Programming in Martin-Löf's type theory*, volume 200. Oxford University Press, Oxford, 1990.

Timothy C Potts. 'The grossest confusion possible?'– Frege and the lambda-calculus. *Revue Internationale de Philosophie*, 33(130):761–785, 1979. URL https://www.jstor.org/stable/23944070.

W. V. Quine. Alonzo Church. A formulation of the simple theory of types. The journal of Symbolic Logic, vol. 5 (1940), pp. 56–68. *Journal of Symbolic Logic*, 5(3):114–115, 1940. . URL https://doi.org/10.2307/2266866.

W. V. Quine. On Frege's way out. *Mind*, 64(254):145–159, 1955. URL https://www.jstor.org/stable/2251464.

Michael Resnik. Frege's theory of incomplete entities. *Philosophy of Science*, 32(3/4):329–341, 1965. URL https://doi.org/10.1086/288057.

Dieter Rüthing. Some Definitions of the Concept of Function from Joh. Bernoulli to N. Bourbaki. *The Mathematical Intelligencer*, 6(4):72–77, 1984.

Moses Schönfinkel. Über die Bausteine der mathematischen Logik. *Mathematische Annalen*, 92(3-4):305–316, 1924.

Peter Simons. Double value-ranges. In Philip A. Ebert and Marcus Rossberg, editors, *Essays on Frege's Basic Laws of Arithmetic*, pages 167–181. Oxford University Press, Oxford, 2019.

Wim Vanrie. Why did Frege reject the theory of types? *British Journal for the History of Philosophy*, pages 1–20, 2020. URL https://doi.org/10.1080/09608788.2020.1807908.

Alfred N. Whitehead and Bertrand Russell. *Principia Mathematica to* 56*, volume 2. Cambridge University Press, 1997.

Acknowledgments I would like to thank Ansten Klev, Marco Panza, Marco Ruffino, Will Stafford, Göran Sundholm, Wim Vanrie, and two anonymous reviewers for their helpful comments on early drafts of this paper. Most of the material here has been presented in different online seminars and conferences over the past three years, in particular the OCIE Seminar in the History and Philosophy of Mathematics and Logic, Logicians in Quarantine, I Colóquio NULFA-NuLFiC de Lógica e Filosofia da Matemática, and Chá das 5. I also thank Alessandro Duarte, Erich Reck, and Edward Zalta for stimulating discussions. This research was partly supported by the AFOSR grant FA9550-18-1-0120 and the Lumina quaeruntur fellowship number LQ300092101 from the Czech Academy of Sciences.

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