Frege's theory of types

Bruno Bentzen

Abstract There is a widespread assumption in type theory that the discipline begins with Russell's efforts to resolve paradoxes concerning the naive notion of a class. My aim in this paper is to argue that Frege's sharp distinction between terms denoting objects and terms denoting functions on the basis of their saturation anticipate a simple type theory, although Frege vacillates between regarding functions as closed terms of a function type and open terms formed under a hypothetical judgment. Frege fails to express his logical views consistently due to his logicist ambitions, which require him to endorse the view that value-ranges are objects.

1 Introduction

There is a widespread assumption in type theory that the discipline begins with Russell's efforts to resolve paradoxes concerning the naive notion of a class since 1903. However, if by a theory of types we understand a logical formalism where every term must have a type and every operation must be restricted to terms of a certain type, then it can be argued that Frege anticipated much of a theory of types in his two-volume book *Grundgesetze der Arithmetik* (1893/1903), which introduces the mature version of the ideography that he intended to use as a formal logical system to vindicate his longstanding logicist claim that arithmetic is reducible to logic.

Of course, never once in his career did Frege advocate a full-fledged doctrine of types, but his sharp distinction between names of objects and names of functions leaves no doubt that his ideography is always implicitly operating on typed terms, for objects and functions form two disjoint categories of things. For Frege, objects are saturated and complete, while functions are unsaturated and in need of completion (*Grundgesetze* I §1). This idea is better understood in the setting of a simple type system with a ground type of individuals ι and a type former for functions $\sigma \rightarrow \tau$, where σ and τ stand for types. The requirement that every term definable in the theory must have a unique type thus incorporates the principle that every term either stands for an object or a function. Given that numbers, truth values, and value-ranges do not exhibit saturation, they cannot be viewed as functions, so, according to this dichotomy, the only alternative left is to regard them as objects

> *Objects* stand opposed to functions. Accordingly, I count as an *object* everything that is not a function, e.g., numbers, truth-values and the value-ranges introduced below. Thus, names of objects, the *proper names*, do not in themselves carry argument places; like the objects themselves, they are saturated. (*Grundgesetze* I §2)

Semantically, this dichotomy is reflected by a sharp distinction between terms that are supposed to refer to objects and terms that are supposed to refer to functions. For variables, Frege determines whether they are intended to indicate an object or a function using specific letter conventions, which are all different for Roman, Greek, and Gothic letters. Ultimately, the type inference is left to the reader and depends on the context, but here are some common examples of his letter conventions:

- Roman object letter: a–e, i–l, n–z;
- Greek object letter: α–ε, Γ–P;
- Gothic object letter: $\mathfrak{a}$–$\mathfrak{e}$;

- first-level Roman function letter: f–h, F–H;
- first-level Greek function letter: Φ–Ψ;
- first-level Gothic function letter: $\mathfrak{f}$–$\mathfrak{h}$;
- second-level Roman function letter: M;
- second-level Greek function letter: Ω.

The distinction between Roman, Greek, and Gothic marks is equally essential to Frege's theory of types since, as I will describe in more detail in the next section, each kind of letter has their own purpose in the ideography. Not only type annotations are implicitly present in the ideography in the form of metavariables ranging over types, but also operational constraints that depend on the type assigned to a term.

Frege famously takes the notion of function as primitive, viewing predicates and relations as functions that assign objects to reified truth values and distinguishing between functions of first and higher level depending on whether they only admit object or functions of lower levels as arguments. That Frege takes those constraints very seriously can be most readily seen from his initial response to the discovery of Russell's paradox, where, in his first letter to Russell, he explains that the circular expression "a predicate is predicated of itself" is not an acceptable term in the ideography because a predicate is a function of first level.[1] More generally, if we adopt the usual notation $a : \sigma$ to express that a is a term of type σ, then it is easy to see that for every term $f : \sigma \to \tau$, the function application $f(f)$ will always be ill-typed. Unfortunately, as value-ranges are taken to be objects, self-application is completely able to enter the ideography through the back door, compromising the coherence of the whole type system of the ideography.

Even though there can be no doubt that Frege failed to arrive at a consistent conception of type constraint, the fact remains that many of his logical insights are still accepted today since the development of Church's simple type theory.

[1] See Frege (1980, p.131–133).

This point has been alluded to by some authors. In particular, Quine (1940) has pointed out that Church revived Frege's conception of function abstraction and predicates with his lambda abstraction and his treatment of predicates as functions to the type of booleans.[2] Later, Quine (1955) comes to recognize Frege's hierarchy of objects, functions of first level, functions of second level and so forth as an anticipation, to some degree, of the theory of types. Potts (1979) explores some similarities and differences between the ideography and the lambda calculus, and Klement (2003) adds detail and precision to the comparison of the role played by value-range and lambda terms in their respective systems, although major emphasis is placed on the influence of Frege's conception of value-ranges in Russell's early work. Simons (2019) examines Frege's use of double value-ranges to provide extensions for binary functions, a technique known as currying in type theory. Despite that, however, I believe Frege's theory of types still has not received the attention it deserves from type theorists, and it is far from obvious how Frege's logical views should be understood from a modern type-theoretic perspective.

Indeed, what I hope to show in the present paper is that more sense can be made of Frege's theory of types than is generally assumed in the type theory literature. In Section 2, I claim that Frege's conventional usage of Roman and Greek marks allows him to distinguish between open and closed terms in a robust way that resembles the type-theoretic distinction between hypothetical and categorical judgments. In Section 3, I investigate Frege's conception of value-ranges as objects and his anticipation of the lambda calculus, and I conclude that Frege's understanding of functions is based on a confusion between open terms and closed terms of the function type. In Section 4, I examine how the view that value-ranges are functions would be in conflict with Frege's conception of identity and his logicist ambition of establishing numbers as logical objects.

[2]After the discovery by Howard (1980) of a correspondence between propositions and types in the context of constructive logic, this approach has been generalized, giving birth to type families, indexed families of types that associate every term to a type.

2 Categorical and hypothetical judgments

The two forms of judgment that have become standard in modern logic can be traced back to Frege's ideography. First, we have the usual judgment form that states that a proposition is true, which he famously writes in a turnstile notation

$$\vdash A$$

and, second, we have the accompanying judgment form that states that a sentence expresses a proposition, which is initially proposed in the early version of the ideography in *Begriffsschrift* through the notion of the content stoke

$$— A$$

which characterizes a proposition as the content of a turnstile judgment, that is, a "judgeable content". The notion of content was originally the main semantic unit of the ideography, and it was not until after the distinction between sense and reference that a proposition came to be explained as the sense or thought of a sentence. That is, a sentence expresses a proposition and refers to a truth value. When the theory of sense and reference is formally incorporated into the ideography in *Grundgesetze*, the turnstile judgment gets to be explained as the assertion that a sentence refers to the true and the content stroke ends up being treated as a function that refers to the true for the true as argument and the false otherwise (the horizontal).

As a result, it is no longer possible to assert the fact that a sentence expresses a proposition, which is to say that a term expresses a thought, since the ideography does not have a counterpart for the content stoke anymore. In practice, that is achieved through the manipulation of functions into truth values such as the horizontal, negation, identity, conditional, and universal quantifier, which, when fully saturated, are assumed to have a truth value as reference, therefore expressing a thought. Since sentences are handled as terms that refer to objects, and a sentence that refers to exactly one truth value cannot be both true and false, a consistency proof for the ideography can be given by showing that every well-formed term in the system has a unique reference. That is precisely what Frege sets out to do in *Grundgesetze* I §§29-31. This is a point that I shall return to later in this section.

2.1 Unsaturation and hypothetical judgments

The assertion of a turnstile judgment is commonly supported by one or more implicit typing annotations, because the fact that the proposition expressed by a particular sentence is true often depends on assumptions that a letter occurring in the sentence refers to an object or function. In order to ensure full transparency with respect to the assumptions needed in the assertion of a turnstile judgment such as

$$\vdash\!\!\!\!- A \supset (B \supset A)$$

I shall write all typing assumptions as explicit hypotheses on the left-hand side of the turnstile in a modern notation, where $a\,\mathsf{true}$ indicates that a refers to the true

$$A : \iota, B : \iota \vdash A \supset (B \supset A)\,\mathsf{true}.$$

In fact, I shall be treating the relation $a : \sigma$ and predicate $a\,\mathsf{true}$ as forms of judgments as well, meaning that they may have assertive force and can be subject to rules of inference. Thus, to give an example, I may write

$$x : \iota, f : \iota \to \iota \vdash f(x) : \iota$$

to express that $f(x)$ is an object under the assumption that f is a function and x is an object. In type theory, it is common to refer to judgments made under typing assumptions such as the last two illustrated above as hypothetical judgments and judgments made under no typing assumptions such as the first one above as categorical judgments.

Although it would not be wrong to say that Frege makes extensive use of categorical judgments, it would seem that when all the typing annotations of a claim are made explicit in the way I just suggested very few categorical judgments can be found in the ideography. This indicates that most judgments in the ideography are in effect hypothetical ones in our sense, and, as a matter of fact, the notion of hypothetical judgment is implied in the ideography through the adoption of Roman letters since §1 of *Grundgesetze* I, where Frege speaks repeatedly of the unsaturated nature of functions. In particular,

$$(2 + 3x^2)x$$

cannot refer to an object, since, when we substitute the numerals 0, 1, 2, and 3 for the argument place x, we obtain terms referring to different objects, namely, the numbers 0, 5, 28, and 87, respectively. That is because $(2 + 3x^2)x$ is an open rather than a closed term, that is, it is a term with free occurrences of variables. I shall express this fact with the hypothetical judgment

$$x : \iota \vdash (2 + 3x^2)x : \iota$$

that asserts that $(2 + 3x^2)x$ refers to an object provided that x also refers to an object, or, what amounts to the same thing, since every closed term is supposed to have a reference, that $(2 + 3x^2)x$ is a closed object term for a closed object term x.

Every function term is unsaturated in the ideography, but surprisingly not every unsaturated expression, by which Frege appears to mean an open term, as it will be argued in Section 3.3, is accepted as a function term. Perhaps the most straightforward way of seeing this is by observing that every theorem in the ideography is actually a schema formed by open sentences with free occurrences of sentential variables. Curiously, Frege does not view open terms as function terms because he does not accept them as referential terms:

> 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* I §17)

But he gives no reason as to why function terms are referential and open terms are not, since, to the untrained eye, both seem to be unsaturated expressions in the exact same way. The holes in a function term (of first level) are marked using lowercase Greek letters such as ξ or ζ while in an open term we always have Roman letters playing the role. For Frege, however, open terms only serve as indicators of objects. They are what he calls "Roman marks", unsaturated combination of signs that result in a closed term when every Roman letter occurring in it is replaced with a closed term. For an open term, the best we can hope for is that a referential expression is achieved when all their variables are instantiated with closed terms. On the other hand, although Function terms are unsaturated, we can show that they are referential if they always result in a referential expression when properly applied to closed terms.

By arguing that function terms are unsaturated, Frege not only creates an unnecessary distinction between two classes of unsaturated expressions, but also commits himself to the puzzling thesis that function terms behave like open terms $x : \iota \vdash f(x) : \iota$ despite actually being closed terms of a function type $f : \iota \to \iota$. I will come back to this discussion in the next section after examining Frege's conception of value-ranges and his view of abstracted functions as saturated entities.

2.2 Saturation and categorical judgments

Although not a part of the referential terms of the language, uppercase Greek letters are in sharp contrast with Roman marks in that they are used as a stand-in for referential terms. Heck (1997) views them as auxiliary names that are added to the language subject to the condition that they must refer to some object in the domain. Instead of relying on an assignment of values to free variables, as it common in modern logic since the seminal work of Tarski, Frege makes use of auxiliary names that are assumed only to refer to some object.

The supporting role of auxiliary names is most evident in Frege's proof of referentiality in §§29–31 of *Grundgesetze* I, where he intends to show that every closed term of the ideography is referential by arguing by induction on their structure that $f(\xi)$ is referential if $f(\Delta)$ always has a reference for every object term Δ and that a is referential if $\Phi(a)$ always has a reference for every function term $\Phi(\xi)$. Thus, Frege does think of uppercase Greek letters as terms that are assumed to have a reference, although not only in the domain of objects but that of functions as well. In fact, since every closed term is supposed to be referential, I believe the claim that Δ is an auxiliary name amounts to the hypothesis that it is closed term. Therefore, in a turnstile judgment

$$\vdash\!\!\!-\!\!\!-\, \Delta \supset (\Gamma \supset \Delta)$$

consisting entirely of uppercase Greek letters such as Δ and Γ, what we actually have is a categorical judgment where every auxiliary name is assumed to be closed, which is to say, in other words, that, by hypothesis, they are assumed to be derivable under no typing assumptions. The presuppositions involved in the reasoning above can be expressed more accurately as an inference where the notion of a closed term is made explicit

$$\frac{\vdash A : \iota \quad \vdash B : \iota}{\vdash A \supset (B \supset A)\ \mathsf{true}}.$$

In words, the inference above states that we are able to show that the sentence $A \supset (B \supset A)$ is true provided that we can assert that A and B are closed object terms. When judgments are represented in this way, as I shall do, we no longer have a need to write auxiliary names as uppercase Greek letters.

3 The doctrine of value-ranges

Considering that Frege is determined not to call into doubt his distinction between objects and functions, to question the premise that value-ranges are objects is to embrace the apparently superfluous thesis that the resulting value-range of a function is again a function. In the ideography, the conception of value-ranges as objects can be expressed as the requirement that that the value-range term $\grave{\epsilon}f(\epsilon)$ refers to an object for a function f, which, in our notation, can be more easily articulated via the hypothetical judgment

$$f : \iota \to \iota \vdash \grave{\epsilon}f(\epsilon) : \iota. \tag{1}$$

More precisely, value-ranges are regarded as first-class objects obtained by function abstraction, which, in turn, are incorporated as a function of second level which has the value-range of a first-level unary function as value. In modern terminology, we can think of the value-range $\grave{\epsilon}f(\epsilon)$ as as the graph of the function $f(x)$ for an argument x.

3.1 Why we should not blame Basic law V

The notion of value-range is governed by Basic Law V, the infamous axiom that states that the value-range of the function f is the same as the value-range of the function g just in case f and g have the same values for the same arguments

$$\vdash\!\!\!\!-\!\!\!-\, \grave{\epsilon}f(\epsilon) = \grave{\alpha}g(\alpha) \leftrightarrow \forall(\mathfrak{a})f(\mathfrak{a}) = g(\mathfrak{a}).$$

As I noted in the previous section, this is in fact an axiom schema since the function letters f and g are open terms, so, to begin with, we observe that we have a hypothetical judgment

$$f, g : \iota \to \iota \vdash \grave{\epsilon}f(\epsilon) = \grave{\alpha}g(\alpha) \leftrightarrow \forall(\mathfrak{a})f(\mathfrak{a}) = g(\mathfrak{a})\ \mathsf{true}.$$

Not only Roman letters have implicit typing assumptions, and, indeed, here we have another example with the Gothic letters that are used to mark the variables bound by a quantifier. This convention is patently clear in *Grundgesetze* I §20 where the theorem given as

$$\vdash \forall(\mathfrak{f})(\forall(\mathfrak{a})\mathfrak{f}(\mathfrak{a})) \supset \mathfrak{f}(x)$$

actually stands for the fully explicit theorem schema, which, for convenience, I will henceforth write without Gothic letters,

$$x : \iota \vdash \forall(f : \iota \to \iota)(\forall(a : \iota)f(a)) \supset f(x) \ \mathsf{true}$$

because $x : \iota$ and $f(x) : \iota$ can only mean that $f : \iota \to \iota$. Now, if we apply the same strategy to Basic Law V as well, we are able to bring it to its definite form

$$f, g : \iota \to \iota \vdash \grave{\epsilon}f(\epsilon) = \grave{\alpha}g(\alpha) \leftrightarrow \forall(x : \iota)f(x) = g(x) \ \mathsf{true}.$$

On close inspection, one can see that there is one fundamental assumption concerning the nature of value-ranges that is not stated in this formulation, namely, the condition (1) that value-ranges be objects. As it stands, there is nothing inherently contradictory about Basic Law V per se and this is reflected by the fact that this law has been rediscovered in type theory under the name of function extensionality (see Section 4.1).

3.2 Currying and double value-ranges

On an equal footing with Frege's classification between first and higher level is his distinction between unary and binary functions in the ideography. Clearly, unary functions can be only possibly annotated as $f : \iota \to \iota$, but how should we write the type of a binary function? Binary functions act on a pair of arguments but may also be partially saturated with a single term, resulting in a unary function:

> 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* I §4)

This partial saturation creates some ambiguity in the interpretation of function application and suggests that Frege actually thinks of binary functions as unary functions from objects to unary functions, an idea that was later on rediscovered by Schönfinkel (1924) and further developed in Curry and Feys (1958). In other words, Frege was the first to make use of the concept of currying, the idea in combinatory logic, lambda calculus, and type theory that functions of multiple arguments can be entirely dispensed with if we allow functions to have other functions as values. Partial saturation means that a binary function g actually has the type

$$g : \iota \to (\iota \to \iota)$$

However, in general, a binary function may be partially saturated in two different ways, and, to make the argument order explicit, Frege always writes a binary function term g as $g(\xi, \zeta)$ instead, meaning that, given a term $a : \iota$, he can indicate the two resulting unary functions as $g(a, \zeta)$ and $g(\xi, a)$. In the lambda calculus, the use of explicit argument-marks are unnecessary because function abstraction leaves no room for ambiguity in the determination of the arguments of a function, and in this way the unary functions above may be represented as $\lambda x.g(a)(x)$ and $\lambda x.g(x)(a)$, respectively. Simons (2019) explains in detail that Frege's ideography had a very similar feature for handling application in curried functions with the use of the double value-ranges introduced in *Grundgesetze* I §36. Since, for instance, $g(a, \zeta)$ is a unary function, we can form its value-range $\grave{\epsilon} g(a, \epsilon) : \iota$. Now, by removing $a : \iota$ from this term, a term formation method explained in §30, we can form a new unary function $\grave{\epsilon} g(\zeta, \epsilon)$, whose value-range is $\grave{\alpha}\grave{\epsilon} g(\alpha, \epsilon) : \iota$, the double value-range of the binary function $g(\xi, \zeta)$.

In view of the fact that value-ranges are objects, it is not possible to apply them to objects. Instead, Frege has a special purpose application function, introduced in §34, which is not a built-in primitive operation on a par with the ordinary function application, which from $f : \sigma \to \tau$ and $a : \sigma$ results in $f(a) : \tau$, but, instead, is a definable binary function of the ideography, derived using the definite description function of §11. I shall write this application function as an open term

$$x, y : \iota \vdash x \cap y : \iota$$

in order to avoid the implicit convention of adopting lowercase Greek letters for argument-places. Now, $x \cap y$ refers to $f(x)$ if y is a value-range $\grave{\epsilon} f(\epsilon)$ and to the false otherwise.[3] In fact, we have an explicit equality that holds for value-ranges

$$f : \iota \to \iota, x : \iota \vdash x \cap \grave{\epsilon} f(\epsilon) = f(x) \ \mathsf{true} \tag{2}$$

and, given that a double value-range is just a value-range with a doubly-iterated function abstraction, for any binary function $g : \iota \to (\iota \to \iota)$, and terms $a, b : \iota$, the following equality holds

$$\begin{aligned} b \cap a \cap \grave{\alpha}\grave{\epsilon} g(\alpha, \epsilon) &= b \cap \grave{\epsilon} g(a, \epsilon) \\ &= g(a, b). \end{aligned}$$

This eliminates the ambiguity in the order of application of a binary function, allowing Frege to explicitly differentiate between the terms $\grave{\alpha}\grave{\epsilon} g(\alpha, \epsilon)$ and $\grave{\alpha}\grave{\epsilon} g(\epsilon, \alpha)$, which generally refer to distinct objects when g is not commutative, since

$$b \cap a \cap \grave{\alpha}\grave{\epsilon} g(\alpha, \epsilon) = g(a, b) \quad \text{but} \quad b \cap a \cap \grave{\alpha}\grave{\epsilon} g(\epsilon, \alpha) = g(b, a).$$

There has been one instance, pointed out by Simons (2019), where Frege articulates a notion of simultaneous application of double value-ranges with ordered pairs. The idea is put forward later in §144, together with the definition of a pairing function $x; y$ via application iteration as $\grave{\epsilon}(x \cap y \cap \epsilon)$ so that

[3] Actually, when y is not a value-range, $x \cap y$ refers to the value-range of a function whose value for every argument is the false, which, according to the stipulations of §10 is the false itself (see Section 4.2).

$$\begin{aligned}\grave{\alpha}\grave{\epsilon}g(\alpha,\epsilon)\cap a;b &= \grave{\alpha}\grave{\epsilon}g(\alpha,\epsilon)\cap\grave{\epsilon}(a\cap b\cap\epsilon)\\ &= b\cap a\cap\grave{\alpha}\grave{\epsilon}g(\alpha,\epsilon)\\ &= b\cap\grave{\epsilon}g(a,\epsilon)\\ &= g(a,b).\end{aligned}$$

In type theories with a product type $\sigma\times\tau$, whose terms are ordered pairs $(a,b):\sigma\times\tau$ for $a:\sigma$ and $b:\tau$, currying is commonly expressed as a logical equivalence between the types $\sigma\times\tau\to\upsilon$ and $\sigma\to(\tau\to\upsilon)$, which states the existence of one function that transforms a binary function into its curried form and one function that takes a curried function and transforms it back into its binary form. Frege's conception of currying could not possibly be better stated with his identification of simultaneous and iterated applications, except that since $x;y$ is a value-range, ordered pairs are not terms of a product type. They are objects like any others, and since a function $f:\iota\to\iota$ has to be defined for all objects (see Section 4.1), any such f can take an ordered pair as argument and still be unary.

3.3 Frege's simple type theory

Recall that the real purpose of the restrictions of a type system is to ensure that operations are applied only to arguments of the intended domain, making sure that well-typed terms are always well-behaved in a certain sense. Given that every well-formed term is well-typed and vice-versa, for Frege, good behavior means that a term definable in the ideography is referential, and, as mentioned in the previous section, Frege goes to great lengths in *Grundgesetze* I §§29–31 to show that this is the case. Unfortunately, his proof cannot go through due to the contradictions found in the ideography, which, as noted by Frege himself in his first letter to Russell (see footnote 1), arises in the form of the self-application $\grave{\epsilon}f(\epsilon)\cap\grave{\epsilon}f(\epsilon)$ which is coreferential with $f(\grave{\epsilon}f(\epsilon))$ for any $f:\iota\to\iota$. Now, if we instantiate the application theorem (2) with the function $\neg x\cap x$ and its corresponding value-range, we immediately arrive at a contradiction, a closed term of the form $\neg a = a$ that simultaneously refers to both the true and false.

It is therefore fair to assume that Frege's theory of types turned out to be inconsistent due to his flawed conception of value-ranges as objects (1), which undermines all his efforts to separate objects from functions. If the most prudent way out of the contradiction is to view value-ranges not as objects but as functions, the only problem is that value-ranges are themselves determined by functions for Frege, so such a stipulation would make the distinction between functions and value-ranges lose its purpose:

$$f : \iota \to \iota \vdash \grave{\epsilon} f(\epsilon) : \iota \to \iota$$

It was Church (1940) who first realized with his lambda calculus how to adequately dissolve the dichotomy between functions and value-ranges and capture Frege's intuition that we form value-ranges by abstraction on unsaturated expressions. The idea simply involves a functional abstraction on open terms, which are then regarded as closed function terms via the introduction of an abstract binding operation that ranges over all their free occurrences of variables. That is, instead of abstracting functions to form saturated objects, we abstract unsaturated objects to form functions. If we were to express this view in a Fregean style, it would be as

$$\frac{x : \iota \vdash f(x) : \iota}{\vdash \grave{x} f(x) : \iota \to \iota}. \tag{3}$$

Instead of Frege's smooth-breathing, however, which is recognized by Church (1942) himself as one of the precursors of his lambda-notation, I shall stick to Church's $\lambda x.f(x)$. Moreover, as the explicit use of hypothetical judgments already determines what variables an open term may depend on, (3) has a certain redundancy which may be completely eliminated by rephrasing it as

$$\frac{x : \iota \vdash f : \iota}{\vdash \lambda x.f : \iota \to \iota}. \tag{4}$$

As a result of the identification of value-ranges with functions in the above sense, the ideography no longer would need two distinct forms of function application, the primitive $f(x)$ for functions and the derived $x \cap f$ for value-ranges. Instead, we need only one notion of application $\mathsf{app}(f, x)$, a unifying operation that comes with the same typing structure as $f(x)$ in being restricted to a function and an object

$$f : \iota \to \iota, x : \iota \vdash \mathsf{app}(f, x) : \iota \tag{5}$$

and that inherits, at the same time, the computation rule of $a \cap \grave{\epsilon} f(\epsilon)$ which in §34 is stipulated to denote the same as $f(a)$, as stated in (2). In the lambda calculus, this computation is the so-called β-reduction rule, which roughly states that the application of a lambda term to a closed term results in a closed term where all occurrences of the abstracted variable are replaced with the term applied

$$\frac{x : \iota \vdash f : \iota \qquad \vdash a : \iota}{\vdash \mathsf{app}(\lambda x.f, a) = f[a/x] : \iota} \tag{6}$$

Notice that this computation rule expresses the same idea as Frege's stipulation of §34, apart from the fact that it deals with function rather than object terms. Even another main rule of Church's lambda calculus, α-conversion, the stipulation that two lambda terms that use different variable names are still the same, was already envisioned by Frege (1891), who famously declares that we can write a function like '$x^2 - 4x$' as '$y^2 - 4y$' without altering its sense.[4] For the sake of brevity, I shall simply write $f(x)$ for $\mathsf{app}(f, x)$.

Except by their restriction to the domain of the type of individuals ι, the rules described here are all present in the simple type theory of Church (1940), an extended version of the lambda calculus with a type system composed of a type of individuals, functions, and truth values. It is remarkable that the two turning points that determine the success of Church's

[4]I have stressed this point in Bentzen (forthcoming). For a more focused investigation of Frege's theory of sense and reference in the setting of type theory see Martin-Löf (2001) and Bentzen (2020b).

approach over Frege's are the ideas that (i) functions are formed by abstraction on open terms, a requirement which, to some extent, can be traced back to Frege, since, as it will be seen in the remainder of this section, function terms are treated as if they were open terms in the ideography, and that (ii) value-range terms are function terms, meaning that value-ranges should refer to functions. For now, I will defer the discussion of this second point to the next section, where I explain how it would go directly against Frege's logicist agenda.

In addition to the α- and β-rules, I would like to consider another important rule not found in Church (1940), but studied extensively in Curry and Feys (1958). Considering that Frege's value-range terms are formed by abstraction on function rather than open terms, one may argue that the direct representation of the value-range $\grave{\epsilon}f(\epsilon)$ should be the lambda-term $\lambda x.f(x)$, where f is a function. However, if x does not occur in f, the distinction between $\lambda x.f(x)$ and f turns out to be unnecessary extensionally speaking, since both functions will have the same value for the same arguments. This intuition is captured by the rule known as η-reduction

$$f : \iota \to \iota \vdash \lambda x.f(x) = f : \iota \to \iota \tag{7}$$

which was not originally included in the untyped lambda calculus for the semantic reason that while the left side of the equality is a function, the right side may not be in Church's intended interpretation (Curry and Feys, 1958, p.92). Clearly, η-reduction cannot have a representation in the ideography, because if value-ranges are objects, so it would not make sense to ask whether they could be functions.

Since for Frege value-range terms always refer to objects while for Church lambda terms should be interpreted as functions, and, moreover, that for Frege function application must result in an object while for Church functions are allowed to be values, it can be argued, following Potts (1979), that their approaches to function abstraction are fundamentally different. I am inclined to disagree with Potts on this point, for I do not see the two reasons given above as compelling motivations for distinguishing value-range and lambda terms. I shall address the second reason first. As I have indicated in Section 3.2, Frege's binary functions are essentially of second level. Frege even considers third-level functions in the ideography and, as correctly pointed out by Quine (1955), the only reason why Frege does not adopt a hierarchy of higher functions is because he sees no need for it: his conception of value-ranges as objects allows him to always reduce higher-level functions to objects.

Finally, Klement (2003) calls attention to the fact that Frege has once entertained the idea of having a function abstraction device for function terms in a letter to Russell of 13 November 1904, where Frege employs a rough-breathing notation $\acute{\epsilon}(\epsilon^2 = 1)$ for the function term that he would otherwise write as $\xi^2 = 1$:

> But this 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 '$\acute{\epsilon}(\epsilon^2 = 1)$' would supposedly be a function name which would require completion by a sign following it. '$\acute{\epsilon}(\epsilon^2 = 1)1$' would have the same meaning as '$1^2 = 1$', and accordingly, '$\acute{\epsilon}(\epsilon^2 = 1) \supset$' would have to have the same meaning as '$\supset^2 = 1$', which, however, would be meaningless. '$\acute{\epsilon}(\epsilon^2 = 1)$' 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, 1980, p.161–162).

Although Frege clearly anticipates the developments of lambda calculus in this passage, he quickly abandons the proposal of indicating function terms by abstraction because it would be incompatible with his resolve that the nature of function consists in its unsaturatedness. For him, it is possible to use $\acute{\epsilon}(\epsilon^2 = 1)$ in isolation as an object term because it has no occurrences of argument-places while $\xi^2 = 1$ is a proper function term because the expression itself requires completion.

Put differently, in the quotation above Frege suggests that function terms must be unsaturated because they are formed by incomplete expressions and I believe that this leaves no grounds for doubt that he completely confuses function terms $f : \iota \to \iota$ with open terms $x : \iota \vdash f(x) : \iota$. If we think of Frege's function terms as open terms, whose argument-places are specified as typing assumptions, then the terms that should be assigned to the function type are his value-range terms, when properly reinterpreted as lambda terms in the sense described earlier. It would seem to me that this confusion is one of the factors that motivates Frege to make the bold claim that value-ranges are objects (but see the Section 4.2). On the other hand, I should mention that this reading of Frege's function terms as open terms is not fully consistent with all the aspects of the ideography I have discussed so far. In particular, Frege's account of functions of second and third level is not amenable to this interpretation, since open terms, which are represented as hypothetical judgments, cannot be part of other hypothetical judgments in any way.[5] In the final analysis, it seems that Frege vacillates between the treatment of function terms as open terms and closed terms of a function type.

4 Why value-ranges cannot be objects

Finally, before I conclude, I would like to discuss what I see as the main reason why Frege could not endorse the view that value-ranges are functions, an observation that goes beyond his conviction that value-range terms are saturated. Of relevance to this is Frege's conception of identity as a first-level relation, his lack of a direct identity criterion for functions, and conviction that no identity statements can ever be made about functions.

4.1 Identity is a first-level relation

Frege is known for conceiving identity as an all-inclusive relation in the domain of all objects, or, more precisely, for holding that an identity statement can be formed for any two objects terms in the ideography. This happens to be a generalization of

[5] But note that in (4) I rendered function abstraction, a second-level function, as an inference rule that takes an open term to a closed term.

his principle of complete determination that the definition of a predicate must say whether it is true or false for all objects. In *Grundgesetze* II §65, Frege is clear that this principle of complete determination is expressed by the requirement that every function term must have a reference. It goes without saying that this applies to the identity function as well

$$x, y : \iota \vdash x = y : \iota$$

Frege's criterion of referentiality of *Grundgesetze* I §29 states that to determine the reference of a function term $x : \iota \vdash f : \iota$ is to determine the reference of $f(a) : \iota$ for a closed $a : \iota$.[6] More precisely, to determine the reference of this function term, which is in fact a binary relation, it suffices to determine the truth value of $a = b$ for any two closed terms $a, b : \iota$.

That identity is restricted to the domain of objects is very clearly expressed in Frege's writings. Bearing in mind that one should never be allowed to speak of two functions as being the same according to Frege, when a mathematician expresses the view that two functions are identical he or she is, strictly speaking, incurring in a type mismatch error. Actually, what he or she should have in mind is the idea of two functions being coextensional, as noted in Ruffino (2003), which we may articulate as the fact that f and g always have the same value for the same arguments, that is, in the form of the first-level relation of pointwise function equality. Still, since the two halves of Basic Law V are taken to express the same sense, but in a different way (Frege, 1891, p.27), and the value-range of a function is an object, its graph, we can express coextensionality more directly in first-order terms via function abstraction, as an identity statement between the value-ranges of f and g.

As a matter of fact, even when Frege seems to be explicitly speaking of an identity criterion for functions he recognizes it as a relation of second level that must be distinguished from

[6]The same strategy appears in Martin-Löf's (1982) meaning explanations of the hypothetical judgments of his type theory, but it is unclear whether it was inspired by Frege's approach. Either way, this puts more weight to my allegation that Frege mistreats function terms as open terms.

the usual identity of objects. In his posthumous comments on sense and reference, written between 1892 and 1895, he makes this point very clear, adding that such a second-level relation expresses the same sense as pointwise function equality:

> For every argument the function $x^2 = 1$ has the same (truth-) value as the function $(x+1)^2 = 2(x+1)$ i.e. every object falling under the concept *less by 1 than a number whose square is equal to its double* falls under the concept *square root of 1* and conversely. If we expressed this thought in the way that we gave above, we should have
>
> $$\alpha^2 = 1 \overset{\alpha}{\asymp} (\alpha+1)^2 = 2(\alpha+1)$$
>
> What we would have here is that second level relation which corresponds to, but should not be confused with, equality (complete coincidence) between objects. If we write it $\forall(\mathfrak{a})(\mathfrak{a}^2 = 1) = (\mathfrak{a}+1)^2 = 2(\mathfrak{a}+1)$, we have expressed what is essentially the same thought, construed as an equation between values of functions that holds generally. (Frege, 1979, p.121)[7]

Put another way, this means that Frege's second-level relation term for function correspondence $f(\alpha) \overset{\alpha}{\asymp} g(\alpha)$ expresses, via Basic Law V, the same sense as the first-level identity statement between value-range terms $\grave{\epsilon}f(\epsilon) = \grave{\alpha}g(\alpha)$. Here we see that the conception of value-ranges as objects is used in a crucial way as a technical device to escape the restriction that one is not allowed to speak of identical functions.

Notice that if we were to follow the developments of the previous section of using lambda terms for value-range terms then we would have to make some adjustments to Basic Law V accordingly, as we would have an illegitimate identity statement

[7] It is believed that Frege's explanation of this new notation was given in the lost first part of the manuscript (see Frege (1979, p.121, fn.1)). I took the liberty to modernize Frege's quantifier notation.

holding between two functions on the left-hand side of the equivalence, either as

$$f, g : \iota \to \iota \vdash \lambda x.f(x) = \lambda y.g(y) \leftrightarrow \forall(x : \iota) f(x) = g(x) \; \mathsf{true}$$

or, equivalently, in the presence of η-reduction, as

$$f, g : \iota \to \iota \vdash f = g \leftrightarrow \forall(x : \iota) f(x) = g(x) \; \mathsf{true}.$$

The latter principle is known as function extensionality in dependent type theory, and, as it can be seen, it is nothing more than a paraphrase of Basic Law V. Dependent type theory is flavor of type theory developed by Martin-Löf (1975) that extends simple type theory with the introduction of dependent types, a concept that allows for an elegant treatment of quantifiers and identity, and universe types, types whose terms are themselves types. Identity is conceived as a sortal relation that is limited to two terms of a same type. Comparing incomparables is not allowed, for it does not make sense to ask whether two terms of a different type are equal. Actually, just as envisaged by Frege, identity of objects and identity of functions are regarded as different relations. The difference is that in dependent type theory no identity relation is taken to be more fundamental than the other.[8]

But it must be emphasized that in the ideography there is no sortal identity and a first-level relation of sameness cannot be confused with a second-level one. Although the above formulation of function extensionality is meaningless for Frege, we can certainly avoid the so-to-speak fallacious identity statement $f = g$ on its left-hand side if we state

[8]Surprisingly, many forms of dependent type theory are unable to prove function extensionality, even though the principle is validated by its intended semantics, the meaning explanations (Bentzen, forthcoming). Homotopy type theory (UFP, 2013) has been gaining acceptance as a foundational language for mathematics strong enough for proving not only function extensionality but also that isomorphic objects are equal. However, as the theory has to abandon the meaning explanations as its informal interpretation, its philosophical coherence is open to question (Ladyman and Presnell, 2016; Bentzen, 2020a).

the principle in terms of the second-level relation expression $f(\alpha) \overset{\alpha}{\asymp} g(\alpha)$ instead, which, is not only co-referential but has the same sense as the pointwise function equality statement $\forall(x : \iota)f(x) = g(x)$. Since there is no violation of the principle that identity is a first-order relation, it is fair to regard this principle as Frege's conception of function extensionality.

4.2 Two problems of identity

For Frege, numbers are extensions of concepts, a supposedly logical conception of class defined as value-ranges of predicates. Indeed, recall that Frege's whole program presupposes the derivation of the concept of number from purely logical means. This definition is first envisaged in §68 of *Grundlagen*, Frege's philosophical masterpiece, but only after his proposal of two well-known tentative definitions. His first definitional attempt is not seriously considered, and appears to serve only to motivate his claim that numbers are self-subsistent objects.

In contrast, there is no denying that Frege does seem to struggle to establish the legitimacy of his second attempt, a contextual definition via the so-called Hume's Principle, that says that the number that falls under the predicate f is equal to the number that falls under the predicate g iff f and g are in one-to-one correspondence, a relation of second level that is sketched informally in §§71–72 of *Grundlagen*. We thus have

$$\#xf(x) = \#xg(x) \leftrightarrow f \approx g. \tag{HP}$$

Frege eventually rejects this tentative definition because it does not rule out the possibility of Julius Caesar being a number, and I have argued elsewhere that this strange objection is raised for the reason that it cannot ensure our epistemic grip on numbers as logical objects (Bentzen, 2019). Surely, it can hardly be argued that Hume's Principle is a candidate logical law and, in a letter to Russell of 28 July 1902, Frege expressly states that he sees the notion of value-range as the only possible foundation for our apprehension of logical objects:

> I myself was long reluctant to recognize value-ranges

> and hence classes; but I saw no other possibility of placing arithmetic on a logical foundation. But the question is, how do we apprehend logical objects? And I have found no other answer to it than this, we apprehend them as extensions of concepts, or more generally, as value-ranges of functions. (Frege, 1980, pp. 140–141)

Frege's transsortal identification of truth values with value-ranges proposed in *Grundgesetze* I §10, one of the most extensively studied sections of the book, makes it clear that in his view value-ranges are the fundamental logical objects that populate the universe of arithmetic. To resolve a referential indeterminacy affecting the notion of value-range, Frege stipulates that the true is equal to the value-range of a function that always has the true as value for every argument and a similar specification is given for the false.

This is a curious section that has caused considerable confusion among Frege scholars because the Julius Caesar objection is generally regarded as the semantic problem that Hume's Principle does not determine the truth value of mixed-identity statements of the form $\#xf(x) = a$. This common interpretation of the Julius Caesar objection, however, is incapable of explaining why, in §10, when a similar problem of indeterminacy is encountered, now with respect to Basic Law V and the reference of mixed-identity statements of the form $\grave{\epsilon}f(\epsilon) = a$, which means that their truth value is yet to be decided, Frege feels entitled to restrict his solution to the domain of logical objects with his stipulation that some truth values are value-ranges, completely ignoring the question of whether a value-range could be identical to an urelement.

This dilemma can be resolved by noting that, since the Julius Caesar objection in *Grundlagen* just calls into question whether Hume's Principle succeeds in establishing beyond all doubt that we apprehend numbers are logical objects, once value-ranges are already accepted as logical objects, there is no need to worry about urelements anymore. All that we need to do is to determine the reference of the function $\grave{\epsilon}f(\epsilon) = x$, and, according to §29, to do so is to determine the truth value of the closed sentence $\grave{\epsilon}f(\epsilon) = a$ for every closed term $a : \iota$. Seeing that, prior to the transsortal identification of §10, every closed object term in the ideography is supposed to refer to either a value-range or truth value, and Basic Law V already takes care of value-range terms, it is enough to decide whether $\grave{\epsilon}f(\epsilon) = a$ is true or false for a closed sentence $a : \iota$.

The position that value-ranges are functions may be able to prevent the occurrence of paradox threats in the ideography and, considering that it is founded on the assumption that we perform function abstraction on open terms, it may even be in line with Frege's tendency to consider his function terms as open terms, as I have already mentioned. Still, this position would be of no use to Frege. More than anything else, Frege felt he had to commit himself to the existence of value-ranges in order to define numbers logically and make his logicist program plausible, but to conform to his thesis that numbers are objects, value-ranges have to be objects as well. Frege came remarkably close to the formulation of simple type theory as we know it, but in the end he failed to express his theory of types consistently due to his logicist ambitions.

References

Bruno Bentzen. Frege on Referentiality and Julius Caesar in Grundgesetze Section 10. *Notre Dame Journal of Formal Logic*, 60(4):617–637, 2019.

Bruno Bentzen. What types should not be. *Philosophia Mathematica*, 28(1):60–76, 2020a.

Bruno Bentzen. Sense, reference, and computation, 2020b. Preprint. URL: https://sites.google.com/site/bbentzena/publications.

Bruno Bentzen. On different ways of being equal, forthcoming. *Erkenntnis*. URL: https://link.springer.com/article/10.1007/s10670-020-00275-8.

Alonzo Church. A formulation of the simple theory of types. *The Journal of Symbolic Logic*, 5(2):56–68, 1940.

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

Haskell B. Curry and Robert Feys. *Combinatory logic*, volume 1. North-Holland Amsterdam, 1958.

Philip A Ebert and Marcus Rossberg. *Gottlob Frege: Basic laws of arithmetic*. Oxford University Press (UK), 2013.

Gottlob Frege. *Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens*. L. Nebert, Halle a.S., 1879.

Gottlob Frege. *Funktion und Begriff*. Hermann Pohle, Jena, 1891.

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

Gottlob Frege. Comments on sense and meaning. In Hans Hermes, Friedrich Kambartel, and Friedrich Kaulbach, editors, *Posthumous writings*, pages 118–125. Basil Blackwell, Oxford, 1979.

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

Richard G. Heck. Grundgesetze der Arithmetik I §§29–32. *Notre Dame J. Formal Logic*, 38(3):437–474, 07 1997. . URL `https://doi.org/10.1305/ndjfl/1039700749`.

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.

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

James Ladyman and Stuart Presnell. Does Homotopy Type Theory Provide a Foundation for Mathematics. *The British Journal for the Philosophy of Science*, 2016.

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. The sense/reference distinction in constructive semantics. Paper read at Leiden (transc. by B. Jespersen), 2001.

Timothy C Potts. 'The Grossest Confusion Possible?'–Frege and the Lambda-calculus. *Revue Internationale de Philosophie*, pages 761–785, 1979.

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. .

Willard V Quine. On Frege's way out. *Mind*, 64(254):145–159, 1955.

Marco Ruffino. Why Frege Would Not Be a Neo-Fregean. *Mind*, 112(445):51–78, 2003. ISSN 00264423, 14602113. URL http://www.jstor.org/stable/3093824.

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

Peter Simons. Double value-ranges. *Essays on Frege's Basic Laws of Arithmetic*, page 167, 2019.

UFP (The Univalent Foundations Program). Homotopy type theory: Univalent foundations of mathematics, 2013.