A Dependent Set Theory

Wojciech Moczydłowski
Department of Computer Science
Cornell University
Ithaca, NY, 14853, USA
wojtek@cs.cornell.edu

Abstract

Set theories are traditionally based on first-order logic. We show that in a constructive setting, basing a set theory on a dependent logic yields many benefits. To this end, we introduce a dependent impredicative constructive set theory which we call IZF$_D$. Using realizability, we prove that the underlying lambda calculus weakly normalizes, thus enabling program extraction from IZF$_D$ proofs. We also show that IZF$_D$ can interpret IZF with Collection. By a well-known result of Friedman, this establishes IZF$_D$ as a remarkably strong theory, with proof-theoretical power equal to that of ZFC. We further demonstrate that IZF$_D$ provides a natural framework to interpret first-order definitions, thus removing a longstanding barrier to implementing constructive set theories. Finally, we prove that IZF$_D$ extended with excluded middle is consistent, thus paving the way to using our framework in the classical setting as well.

1 Introduction

There are two major foundational frameworks used in mathematics and computer science — set theory and type theory. The former is widely accepted as the foundation of classical mathematics, the latter is being successfully applied in computer science, for the purpose of program verification, programming languages semantics and software engineering.

Both theories are well understood and perform very well in their habitats. Set theory can easily formalize most of the concepts used by mathematicians. The power of modern type theories has exceeded that of Zermelo's set theory [34, 20, 21] and proof assistants based on type theory, such as Coq [32] and Nuprl [10, 6], are successfully used for applications such as extraction of distributed protocols correct-by-construction [8] or formalization of difficult mathematical theorems [14].

We show that combining these two worlds yields many benefits. More specifically, we take a constructive set theory IZF with Replacement (IZF$_R$), and we extend its logic to incorporate several features typical of type theories — dependent implications, conjunctions and what we call restricted Σ-types. We call the resulting "dependent" set theory IZF$_D$ and the underlying lambda calculus λS.

There are several attractive properties of IZF$_D$. First of all, λS weakly normalizes. We prove normalization of λS using realizability, in a spirit close to our previous work [22, 23]. The axiom of choice is used to provide the interpretation of new set terms. The normalization result makes it possible to extract programs from IZF$_D$ proofs.

Second, we show that the combination of dependent features in the logic and Replacement axiom significantly increases the power of a set theory, by showing that IZF$_D$ can prove the axioms of IZF with Collection (IZF$_C$). As known since results of Friedman and Ŝĉedrov [13], Replacement and Collection are not equivalent in the constructive setting. While the proof-theoretic power of IZF$_C$ equals that of ZFC [12], IZF$_R$ is much weaker. It is conjectured in [13] that its consistency can be proved in ZF. Moreover, Collection is a very useful tool in the development of mathematics in constructive set theories; most notably in the treatment of inductive definitions [4, 28]. Thus, IZF$_D$ is a remarkably strong set theory, having the proof-theoretic power of ZFC and all the benefits of Collection at its disposal.

A longstanding, rarely mentioned barrier to utilizing constructive set theories, is the mechanism of first-order definitions. It is an indispensible tool in building the edifice of mathematical knowledge and for implementing set theory. For example, in ZFC, in order to introduce a function symbol for the addition function on natural numbers, one first shows the statement $\phi \equiv \forall m, n \exists ! o.\ ((m \in \omega \land n \in \omega) \to$ "o is a sum of m and n") $\land(\neg(m \in \omega \land n \in \omega) \to o = 0)$. Then a binary symbol $+$ can be added to the signature, along with the defining axiom $\forall m, n.$"if $m, n \in \omega$, then $m + n$ is a sum of m and n, otherwise $m + n$ is 0". In the constructive setting, however, showing ϕ is problematic. Problems arise with any "partial" function symbol, such as

$+$, as extending an intended domain to the entire universe seems to be impossible in general in the constructive setting. Moreover, it is unknown how to perform this procedure while preserving the capability of program extraction. We show that IZF$_D$ provides a means for solving these problems, as dependent implication combined with Σ-types automatically "skolemizes" a set theory. As we prove that the classical counterpart of IZF$_D$ is consistent, the mechanism can also be applied in implementations of classical set theories.

The importance of consistency results in this area cannot be overestimated, as theories tend to dwell close to inconsistency [11, 27]. This is one reason for the restriction of Σ-types we adopt, which amounts to disallowing the standard reduction rule $\pi_1([t, M]) \to t$. Although IZF$_D$ with unrestricted Σ-types enjoys useful proof-theoretic properties, such as Subject Reduction, we show that it is also inconsistent.

The properties of IZF$_D$ make it a viable base for a proof assistant based on set theory with program extraction capability. As the earlier work [22, 23, 29, 20] does not address the problem of definitions, a construction of a prover based on previous approaches is problematic. We witnessed the problems first-hand, while implementing a small prover based on set theory. A unified presentation of IZF$_D$ should facilitate the implementation process. We hope to utilize a logical framework for this purpose.

This paper is organized as follows. In section 2, we introduce informally the theory IZF$_D$. A formal presentation of the underlying lambda calculus λS can be found in section 3, where we also show that IZF$_D$ with unrestricted Σ-types is inconsistent. We define and use realizabity to prove normalization of λS in sections 4 and 5. In section 6 we investigate the power of IZF$_D$ and its classical counterpart. We describe how to use λS to implement first-order definitions in section 7. Related work is discussed in section 8.

2 IZF$_D$

The theory IZF$_D$ is a dependent version of constructive impredicative set theory IZF with Replacement (IZF$_R$), introduced by Myhill in [26]. IZF$_D$ arises by extending the constructive first-order logic of IZF$_R$ with dependent features. As any detailed account of a theory based on dependent logic involves a large amount of syntax, we postpone the formal treatment to the next section and first describe the theory informally.

Intuitively, the axioms of IZF$_D$ are: Empty Set, Pairing, Infinity, Union, Power Set, $\in$-Induction, dependent Separation and dependent Replacement. The underlying logic is an extension of the constructive first-order logic by dependent implications, conjunctions and restricted Σ-types. Formally, IZF$_D$ does not have any axioms in the traditional sense; it is a logic powerful enough to *derive* all the formulas listed in Figure 1. However, since these formulas are helpful in defining and understanding IZF$_D$, we will continue calling them axioms throughout the paper.

- (IN) $\forall a, b.\ a \in b \leftrightarrow \exists c.\ c \in_I b \wedge a = c$
- (EQ) $\forall a, b.\ a = b \leftrightarrow \forall d.\ (d \in_I a \to d \in b) \wedge (d \in_I b \to d \in a)$
- (EMPTY) $\forall c.\ c \in_I \emptyset \leftrightarrow \bot$
- (PAIR) $\forall a, b \forall c.\ c \in_I \{a, b\} \leftrightarrow c = a \vee c = b$
- (INF) $\forall c.\ c \in_I \omega \leftrightarrow c = 0 \vee \exists b \in \omega.\ c = S(b)$
- (SEP$_{\phi(p,a,\overline{f})}$) $\forall \overline{f}, a \forall c.\ c \in_I S_{\phi(p,a,\overline{f})}(a, \overline{f}) \leftrightarrow (p : c \in a) \wedge \phi(p, c, \overline{f})$
- (UNION) $\forall a \forall c.\ c \in_I \bigcup a \leftrightarrow \exists b \in a.\ c \in b$
- (POWER) $\forall a \forall c.\ c \in_I P(a) \leftrightarrow \forall b.\ b \in c \to b \in a$
- (REPL$_{\phi(p,a,b,\overline{f})}$) $\forall \overline{f}, a \forall c.\ c \in_I R_{\phi(p,a,b,\overline{f})}(a, \overline{f}) \leftrightarrow (\forall x.\ (p : x \in a) \to \exists! y.\ \phi(p, x, y, \overline{f})) \wedge (\exists x.\ (p : x \in a) \wedge \phi(p, x, c, \overline{f}))$
- (IND$_{\phi(a,\overline{f})}$) $\forall \overline{f}.\ (\forall a.\ (\forall b.\ b \in_I a \to \phi(b, \overline{f})) \to \phi(a, \overline{f})) \to \forall a.\ \phi(a, \overline{f})$

Figure 1. The axioms of IZF$_D$

The axioms (SEP$_\phi$), (REPL$_\phi$) and (IND$_\phi$) are axiom *schemas*, parameterized by a formula ϕ. The axioms (IN) and (EQ) along with the intensional membership symbol $\in_I$ form the backbone of the Leibniz ($\forall a, b.\ a = b \to \phi(a) \to \phi(b)$) and Extensionality ($\forall a, b.\ (\forall c.\ c \in a \leftrightarrow c \in b) \to a = b$) axioms, which are derivable[1] in our axiomatization. The symbol $\in_I$ needs not be comprehended in order to utilize the theory, as IZF$_D$ can prove all the axioms[1] with $\in_I$ replaced by $\in$. See [23] for more details.

The axioms (EMPTY), (PAIR), (INF), (SEP$_\phi$), (UNION), (POWER) and (REPL$_\phi$) all assert the existence of certain classes and have the same form: $\forall \overline{a}. \forall c.\ c \in_I t_A(\overline{a}) \leftrightarrow \phi_A(c, \overline{a})$, where t_A is a function symbol and ϕ_A a corresponding formula for the axiom A. For example, for (POWER), t_{POWER} is P and ϕ_{POWER} is $\forall b.\ b \in c \to b \in a$. We reserve the notation t_A and ϕ_A to denote the term and the corresponding formula for the axiom A.

The underlying logic includes dependent implications and conjuctions, denoted by $(p : \phi) \to \psi$ and $(p : \phi) \wedge \psi$. These can be found in the Separation and Replacement axioms. Their parameterizing formulas can depend on proofs, denoted by p. Intuitively, in IZF$_D$ proofs are a valid subject of discourse. This is the main feature which distinguishes the axioms of IZF$_D$ from traditional axiomatizations. In particular, the axioms of IZF$_R$ are precisely what remains if the schemas are restricted to purely first-order formulas.

[1] For first-order formulas. See the discussion in Section 3.3.

3 The λS calculus

A lambda calculus is an integral part of any type theory. It is a typed programming language, providing means for program extraction capability. At the same time, its types provide the logic of a theory and its terms serve as notation for its proofs.

In this section, we describe in detail the calculus λS constituting IZF$_D$. As common in dependent logics, terms, formulas and proof terms are all defined at the same time. The judgments of the type system of λS induce the set theory IZF$_D$. We proceed to make this introduction precise.

3.1 The terms of λS

The terms of λS are divided into three syntactic categories, encompassing proof terms, set terms and formulas, respectively. We will generally use letters M, N, O, P for proof terms[2], s, t, u for set terms, ϕ, ψ, ϑ for formulas and T, S for arbitrary terms. Thus, whenever one of these symbols is encountered in the text, the reader should assume that it has been generated by the corresponding part of the grammar. There are two kinds of variables. The first one, denoted by letters p, q, x, y, z, intuitively corresponds to the propositional implication. The second one, denoted usually by letters a, b, c, intuitively corresponds to the first-order quantification. We call them *lambda* and *set* variables, respectively. The notation $a, b.\ M$ stands for a term with its variables a, b bound. The notation $\overline{T}$ stands for a sequence of terms. The following abstract grammar defines the terms of λS. The first part generates the proof terms. There are two groups of proof terms. The first group corresponds to the first-order logic with dependent features:

$$M ::= x \mid M\ N \mid \lambda a.\ M \mid \lambda x : \phi.\ M \mid \mathrm{inl}(M) \mid \mathrm{inr}(M) \mid$$
$$\mathrm{fst}(M) \mid \mathrm{snd}(M) \mid [t, M] \mid M\ t \mid \langle M, N \rangle \mid$$
$$\mathrm{case}(M, x : \phi.\ N, x : \psi.\ O) \mid \mathrm{magic}(M) \mid \pi_2^{a.\phi}(M)$$

The second group corresponds to the axioms of set theory:

$$\mathrm{ind}_{a,\overline{f}.\ \phi}(M, \overline{t})$$
$$\mathrm{inProp}(t, u, M) \mid \mathrm{inRep}(t, u, M)$$
$$\mathrm{eqProp}(t, u, M) \mid \mathrm{eqRep}(t, u, M)$$
$$\mathrm{pairProp}(t, u_1, u_2, M) \mid \mathrm{pairRep}(t, u_1, u_2, M)$$
$$\mathrm{unionProp}(t, u, M) \mid \mathrm{unionRep}(t, u, M)$$
$$\mathrm{sep}_{p,a,\overline{f}.\phi}\mathrm{Prop}(t, u, \overline{u}, M) \mid \mathrm{sep}_{p,a,\overline{f}.\phi}\mathrm{Rep}(t, u, \overline{u}, M)$$
$$\mathrm{powerProp}(t, u, M) \mid \mathrm{powerRep}(t, u, M)$$
$$\mathrm{infProp}(t, M) \mid \mathrm{infRep}(t, M)$$
$$\mathrm{repl}_{p,a,b,\overline{f}.\phi}\mathrm{Prop}(t, u, \overline{u}, M) \mid \mathrm{repl}_{p,a,b,\overline{f}.\phi}\mathrm{Rep}(t, u, \overline{u}, M)$$

Intuitively, the Prop and Rep terms correspond to IZF$_D$ axioms. For example, if M is a proof of $t \in_I P(u)$, then $\mathrm{powerProp}(t, u, M)$ is a proof of $t \subseteq u$ and if M is a proof of $t \subseteq u$, then $\mathrm{powerRep}(t, u, M)$ is a proof of $t \in_I P(u)$. As in our previous work [22, 23], we adopt the convention of using axRep and axProp terms to tacitly mean all Rep and Prop terms, for ax being one of in, eq, pair, union, sep, power, inf and repl. With this convention in mind, we can summarize the definition of the Prop and Rep terms as:

$$\mathrm{axProp}(t, \overline{u}, M) \mid \mathrm{axRep}(t, \overline{u}, M),$$

where the number of terms in the sequence $\overline{u}$ depends on the particular axiom.

The second part of the grammar generates the set terms:

$$\begin{aligned} t \quad ::= \quad & a \mid \pi_1^{a.\phi}(M) \mid \emptyset \mid \{t_1, t_2\} \mid \omega \mid P(t) \mid \bigcup t \mid \\ & S_{p,a,\overline{f}.\phi}(t, \overline{t}) \mid R_{p,a,b,\overline{f}.\phi}(t, \overline{t}) \end{aligned}$$

The term $S_{p,a,\overline{f}.\phi}(t, \overline{t})$ intuitively corresponds to the set $\{(p : a \in_I t) \mid \phi(p, a, \overline{f})\}$. The term $R_{p,a,b,\overline{f}.\phi}(t, \overline{t})$ intuitively corresponds to the set $\{y \mid (\forall(p : x \in t)\exists!y.\ \phi(p, x, y, \overline{t})) \land (\exists p : x \in t.\ \phi(p, x, y, \overline{t}))\}$. The term $\pi_1^{a.\phi}(M)$ can be thought of as a dependent version of the Hilbert's epsilon operator $\epsilon a.\ \phi$. These intuitions are justified by the typing system in Section 3.3.

The third part of the grammar generates the formulas of IZF$_D$:

$$\phi \quad ::= \quad \bot \mid (x : \phi) \to \psi \mid (x : \phi) \land \psi \mid \phi \lor \psi \mid \forall a.\ \phi \mid \exists a.\ \phi$$

The formulas $(x : \phi) \to \psi$ and $(x : \phi) \land \psi$ are dependent versions of implication and conjunction. The variable x binds in ψ, which can mention x (inside of $\pi_1^{a.\phi}$ terms). Traditional formulas $\phi \to \psi$ and $\phi \land \psi$ are defined as abbreviations for $(x : \phi) \to \psi$ and $(x : \phi) \land \psi$, where x is fresh.

Definition 3.1 *A* lambda term *is a term generated by the first part of the grammar. A* set term *is a term generated by the second part of the grammar. A* formula *is a term generated by the third part of the grammar.*

The free variables of a term M are denoted by $FV(M)$. The definition of $FV(M)$, as well as the definition of the (capture-avoiding) substitution, follows the grammar in a natural way, taking into account the formulas appearing in subscripts and superscripts of terms. We show two representative cases of the definition:

$$\begin{aligned} FV(\pi_1^{a.\phi}(M)) &= (FV(\phi) \setminus \{a\}) \cup FV(M) \\ FV(\mathrm{ind}_{a,\overline{f}.\phi}(M, \overline{t})) &= (FV(\phi) \setminus \{a, \overline{f}\}) \cup FV(\overline{t}) \cup FV(M) \end{aligned}$$

3.2 The reduction relation

The reduction relation, denoted by →, is deterministic and defined on the lambda terms. It arises from the following reduction rules and evaluation contexts:

[2] The simultaneous usage of P for the power set function symbol should not lead to any confusion.

$$(\lambda x : \phi.\ M)\ N \to M[x := N] \qquad (\lambda a.\ M)\ t \to M[a := t]$$
$$\mathrm{fst}(\langle M, N\rangle) \to M \qquad \mathrm{snd}(\langle M, N\rangle) \to N \qquad \pi_2^{a.\phi}([t, M]) \to M$$
$$\mathrm{case}(\mathrm{inl}(M), x : \phi.\ N, x : \psi.\ O) \to N[x := M]$$
$$\mathrm{case}(\mathrm{inr}(M), x : \phi.\ N, x : \psi.\ O) \to O[x := M]$$
$$\mathrm{axProp}(t, \overline{u}, \mathrm{axRep}(t, \overline{u}, M)) \to M$$
$$\mathrm{ind}_{a,\overline{f}.\phi}(M, \overline{t}) \to \lambda c.\ M\ c\ (\lambda b.\lambda x : b \in_I c.\ \mathrm{ind}_{a,\overline{f}.\phi}(M, \overline{t})\ b)$$

Note that the standard reduction rule $\pi_1^{a.\phi}([t, M]) \to t$ is not present. The reasons for this omission will become clear in Section 3.4.

The evaluation contexts describe call-by-need (lazy) evaluation order:

$$\begin{aligned}[\circ] \quad ::= \quad & \mathrm{fst}([\circ]) \mid \mathrm{snd}([\circ]) \mid \mathrm{case}([\circ], x : \phi.\ N, x : \psi.\ O) \mid \\ & \pi_2^{a.\phi}([\circ]) \mid \mathrm{axProp}(t, \overline{u}, [\circ]) \mid [\circ]\ M \mid \mathrm{magic}([\circ])\end{aligned}$$

We distinguish certain λS terms, listed below, as values. The set of λS-values will be denoted by λS_v. In the definition, $t, \overline{u}, \phi, M, N$ are arbitrary terms.

$$\lambda a.\ M \mid \lambda x : \phi.\ M \mid \mathrm{inr}(M) \mid \mathrm{inl}(M) \mid [t, M] \mid \langle M, N\rangle \mid \mathrm{axRep}(t, \overline{u}, M)$$

Definition 3.2 *We write $M \downarrow$ and say that M* normalizes *if the reduction sequence starting from M terminates. We write $M \downarrow v$ if we want to state that v is the term at which this reduction sequence terminates. We write $M \to^* N$ if M reduces to N in some number of steps. The symbol $=_\to$ denotes the smallest equivalence relation extending $\to$.*

3.3 The types of λS

We now introduce the type system for λS. Contexts, denoted by Γ, are finite sequences of pairs (z, T), where z is a variable and T is either a formula or a string Set. The *domain* of a context $\Gamma = z_1 : T_1, \ldots, z_n : T_n$, denoted by $\mathrm{dom}(\Gamma)$, is the set $\{z_1, \ldots, z_n\}$. There are three kinds of typing judgments:

- $\Gamma \vdash t : \mathrm{Set}$, read as "$t$ is a set term in the context Γ".
- $\Gamma \vdash \phi : \mathrm{Form}$, read as "$\phi$ is a formula in the context Γ".
- $\Gamma \vdash M : \phi$, read as: "M is a proof of the formula ϕ in the context Γ".

Recall from Section 2 that $t_A(\overline{u})$ and $\phi_A(t, \overline{u})$ are the term and formula corresponding to the axiom (A) of IZF$_D$.

$$\frac{\Gamma \vdash \phi : \mathrm{Form}}{\Gamma, x : \phi \vdash x : \phi}\ x \notin \mathrm{dom}(\Gamma) \qquad \frac{}{\Gamma, a : \mathrm{Set} \vdash a : \mathrm{Set}}\ a \notin \mathrm{dom}(\Gamma)$$

$$\frac{\Gamma \vdash t, \overline{t} : \mathrm{Set} \quad \Gamma, a, \overline{f} : \mathrm{Set}, p : a \in t \vdash \phi : \mathrm{Form}}{\Gamma \vdash S_{p,a,\overline{f}.\ \phi}(t, \overline{t}) : \mathrm{Set}}$$

$$\frac{\Gamma \vdash t, \overline{t} : \mathrm{Set} \quad \Gamma, a, b, \overline{f} : \mathrm{Set}, p : a \in t \vdash \phi : \mathrm{Form}}{\Gamma \vdash R_{p,a,b,\overline{f}.\ \phi}(t, \overline{t}) : \mathrm{Set}}$$

$$\frac{\Gamma \vdash \overline{u} : \mathrm{Set}}{\Gamma \vdash t_A(\overline{u}) : \mathrm{Set}} \qquad \frac{}{\Gamma \vdash \bot : \mathrm{Form}}$$

$$\frac{\Gamma \vdash t : \mathrm{Set} \quad \Gamma \vdash u : \mathrm{Set}}{\Gamma \vdash t \circ u : \mathrm{Form}}\ \circ \in \{\in_I, =, \in\}$$

$$\frac{\Gamma \vdash \phi : \mathrm{Form} \quad \Gamma \vdash \psi : \mathrm{Form}}{\Gamma \vdash \phi \vee \psi : \mathrm{Form}}$$

$$\frac{\Gamma \vdash \phi : \mathrm{Form} \quad \Gamma, x : \phi \vdash \psi : \mathrm{Form}}{\Gamma \vdash (x : \phi) \circ \psi : \mathrm{Form}}\ \circ \in \{\to, \wedge\}$$

$$\frac{\Gamma, a : \mathrm{Set} \vdash \phi : \mathrm{Form}}{\Gamma \vdash Qa.\ \phi : \mathrm{Form}}\ Q \in \{\forall, \exists\}$$

$$\frac{\Gamma, x : \phi \vdash M : \psi}{\Gamma \vdash \lambda x : \phi.\ M : (x : \phi) \to \psi} \qquad \frac{\Gamma, a : \mathrm{Set} \vdash M : \phi}{\Gamma \vdash \lambda a.\ M : \forall a.\ \phi}$$

$$\frac{\Gamma \vdash M : (x : \phi) \to \psi \quad \Gamma \vdash N : \phi}{\Gamma \vdash M\ N : \psi[x := N]} \qquad \frac{\Gamma \vdash M : \forall a.\ \phi \quad \Gamma \vdash t : \mathrm{Set}}{\Gamma \vdash M\ t : \phi[a := t]}$$

$$\frac{\Gamma \vdash M : \phi \quad \Gamma \vdash N : \psi[x := M]}{\Gamma \vdash \langle M, N\rangle : (x : \phi) \wedge \psi}$$

$$\frac{\Gamma \vdash M : (x : \phi) \wedge \psi}{\Gamma \vdash \mathrm{fst}(M) : \phi} \qquad \frac{\Gamma \vdash M : (x : \phi) \wedge \psi}{\Gamma \vdash \mathrm{snd}(M) : \psi[x := \mathrm{fst}(M)]}$$

$$\frac{\Gamma \vdash t : \mathrm{Set} \quad \Gamma \vdash M : \phi[a := t]}{\Gamma \vdash [t, M] : \exists a.\ \phi}$$

$$\frac{\Gamma \vdash M : \exists a.\ \phi}{\Gamma \vdash \pi_1^{a.\phi}(M) : \mathrm{Set}} \qquad \frac{\Gamma \vdash M : \exists a.\ \phi}{\Gamma \vdash \pi_2^{a.\phi}(M) : \phi[a := \pi_1^{a.\phi}(M)]}$$

$$\frac{\Gamma \vdash M : \phi}{\Gamma \vdash \mathrm{inl}(M) : \phi \vee \psi} \qquad \frac{\Gamma \vdash M : \psi}{\Gamma \vdash \mathrm{inr}(M) : \phi \vee \psi}$$

$$\frac{\Gamma \vdash M : \phi \vee \psi \quad \Gamma, x : \phi \vdash N : \vartheta \quad \Gamma, x : \psi \vdash O : \vartheta}{\Gamma \vdash \mathrm{case}(M, x : \phi.\ N, x : \psi.\ O) : \vartheta}$$

$$\frac{\Gamma \vdash M : \forall c.\ (\forall b.\ b \in_I c \to \phi[a, \overline{f} := b, \overline{t}]) \to \phi[a, \overline{f} := c, \overline{t}] \quad \Gamma \vdash \overline{t} : \mathrm{Set}}{\Gamma \vdash \mathrm{ind}_{a,\overline{f}.\ \phi}(M, \overline{t}) : \forall a.\ \phi[\overline{f} := \overline{t}]}$$

$$\frac{\Gamma \vdash M : \phi_A(t, \overline{u}) \quad \Gamma \vdash t, \overline{u} : \mathrm{Set}}{\Gamma \vdash \mathrm{axRep}(t, \overline{u}, M) : t \in_I t_A(\overline{u})} \qquad \frac{\Gamma \vdash M : t \in_I t_A(\overline{u})}{\Gamma \vdash \mathrm{axProp}(t, \overline{u}, M) : \phi_A(t, \overline{u})}$$

$$\frac{\Gamma \vdash M : \exists c.\ c \in_I u \wedge t = c}{\Gamma \vdash \mathrm{inRep}(t, u, M) : t \in u} \qquad \frac{\Gamma \vdash M : t \in u}{\Gamma \vdash \mathrm{inProp}(t, u, M) : \exists c.\ c \in_I u \wedge t = c}$$

$$\frac{\Gamma \vdash M : \forall d.\ (d \in_I t \to d \in u) \wedge (d \in_I u \to d \in t)}{\Gamma \vdash \mathrm{eqRep}(t, u, M) : t = u}$$

$$\frac{\Gamma \vdash M : t = u}{\Gamma \vdash \mathrm{eqProp}(t, u, M) : \forall d.\ (d \in_I t \to d \in u) \wedge (d \in_I u \to d \in t)}$$

$$\frac{\Gamma \vdash M : \bot}{\Gamma \vdash \mathrm{magic}(M) : \phi} \qquad \frac{\Gamma \vdash S : T}{\Gamma, a : \mathrm{Set} \vdash S : T}\ a \notin \mathrm{dom}(\Gamma)$$

$$\frac{\Gamma \vdash S : T \quad \Gamma \vdash \phi : \mathrm{Form}}{\Gamma, x : \phi \vdash S : T}\ x \notin \mathrm{dom}(\Gamma)$$

Lemma 3.3 *If $\Gamma \vdash S : T$, then $FV(S) \cup FV(T) \subseteq \mathrm{dom}(\Gamma)$. Moreover, for any $(x, \phi) \in \Gamma$, $FV(\phi) \subseteq \mathrm{dom}(\Gamma)$.*

We write $\Gamma \vdash T : S$, when this judgment can be derived using the typing rules. The theory IZF$_D$ arises from the typing system, by considering the formulas ϕ such that $\vdash M : \phi$ for some term M, to be provable in IZF$_D$. Although IZF$_D$ might seem formidable at the first sight, we remark that its complexity does not surpass that of other formal systems intended for general use [32, 25, 18].

Most of the rules are standard. The typing system incorporates the definition of formulas and terms of set theory. The term $\pi_1^{a.\phi}$ can be thought of as a version of the Hilbert's epsilon operator, as it provides a witness to any provable existential quantifier. For example, if $\vdash M : \exists a.\ a = P(\omega)$, then $\pi_1^{a.\ a=P(\omega)}(M)$ is "the" A such that $A = P(\omega)$, as we have $\vdash \pi_2^{a.\ a=P(\omega)}(M) : \pi_1^{a.\ a=P(\omega)}(M) = P(\omega)$. In fact, a dependent version of the Hilbert's axiom is provable, as it is easy to see that $\vdash \lambda x : \exists a.\ \phi.\ \pi_2^{a.\phi}(x) : (x : \exists a.\ \phi) \to \phi[a := \pi_1^{a.\phi}(x)]$.

The $\pi_1^{a.\phi}$ operator is non-extensional — from the facts that $M : \exists a.\ \phi$, $N : \exists a.\ \psi$ and $O : \forall a.\ \phi \leftrightarrow \psi$ we cannot derive $\pi_1^{a.\phi}(M) = \pi_1^{a.\psi}(N)$. Because of this, there are instances of the Leibniz axiom not provable in IZF$_D$, such as $a = b \to \pi_1^{c.\phi(a)}(M) \in e \to \pi_1^{c.\phi(b)}(M) \in e$. The effect spreads to the extensional $\in$-Induction, Separation and Replacement axiom schemas — for example, there are formulas ϕ such that $c \in S_{\phi(p,x)}(a) \leftrightarrow p : c \in a \wedge \phi(p, c)$ is not provable. However, one can axiomatize IZF$_D$ with extensional $\in$-Induction axiom with no harm to the developments in the paper. Furthermore, versions of Separation and Replacement insulated against non-extensionality can be derived, for example $c \in S_{\phi(p,x)}(a) \leftrightarrow c \in a \wedge \exists d.\ d = c \wedge q : d \in a \wedge \phi(q, d)$. Finally, it is unclear if any of the unprovable instances would be useful in mathematical practice. We hope to further investigate the interaction between $\pi_1^{a.\phi}$ terms and extensionality in the future.

3.4 Inconsistency of unrestricted Σ-types

There are two natural rules missing from λS: the reduction rule $\pi_1^{a.\phi}([t, M]) \to t$ and the typing rule:

$$\frac{\Gamma \vdash M : \psi}{\Gamma \vdash M : \phi}\ \phi =_{\to} \psi \qquad (*)$$

Let IZF$_D^\Sigma$ denote IZF$_D$ extended with these rules. Unlike IZF$_D$, IZF$_D^\Sigma$ enjoys nice proof-theoretical properties, such as Subject Reduction. However, as the following theorem shows, it also suffers the property of being inconsistent.

Theorem 3.4 *IZF$_D^\Sigma$ is inconsistent.*

Proof Recall first that in set theories, $0 = \emptyset, 1 = \{\emptyset\}$. For the informal proof, consider the set $B = \{x \in 1 \mid \exists a.\ a = a\}$. We can show that for any p proving $x \in B$, there is exactly one y which witnesses the formula $\exists a.\ a = a$, namely the set A used for proving p. Formally, we set $y = \pi_1^{a.\ a=a}(\mathrm{snd}(\mathrm{sep}_{\exists \mathrm{a.\ a=a}}\mathrm{Prop}(x, 1, p)))$. By the Replacement axiom, all these y's can be collected in one set C. Now take any set D and use it to show that $\exists a.\ a = a$ and furthermore that $0 \in B$. Applying (*) to the y corresponding to this proof, we easily find that $D \in C$. Therefore C contains all sets and thus is a subject to the Russell's paradox.[3]

For the formal proof, we only present the relevant terms and provable judgments. Let eqRefl denote the term corresponding to the proof of $\forall a.\ a = a$, let 0in1 denote the term corresponding to the proof of $0 \in 1$ and let russ denote the proof term corresponding to the proof of $\forall a.\ (\forall b.\ b \in a) \to \bot$. The terms are probably best read in a bottom-up fashion.

$$\begin{array}{rcl}
B & \equiv & S_{\exists a.\ a=a}(1) \\
t & \equiv & \pi_1^{a.\ a=a}(\mathrm{snd}(\mathrm{sepProp}(x, 1, p))) \\
M & \equiv & \langle \mathrm{eqRefl}\, t, \lambda z.\ \lambda q : z = t.\ q \rangle \\
N & \equiv & \lambda x.\ \lambda p : x \in B.[t, M] \\
\vdash N & : & \forall x.(p : x \in B) \to \exists! y.\ y = t \\
C & \equiv & R_{p,x,y.\ y=t}(B) \\
P & \equiv & \mathrm{sepRep}(0, 1, \langle \mathrm{0in1}, [a, \mathrm{eqRefl}\, a] \rangle) \\
a : \mathrm{Set} \vdash P & : & 0 \in_I B \\
Q & \equiv & \lambda a.\ \mathrm{replRep}(a, B, \langle N, [0, \langle P, \mathrm{eqRefl}\, a \rangle] \rangle \\
\vdash Q & : & \forall a.\ a \in C \\
\vdash \mathrm{russ}\, C\, Q & : & \bot \qquad\qquad \Box
\end{array}$$

4 Realizability

Let ZFO be the Zermelo-Fraenkel set theory extended with the binary relational symbol $<$ and the axiom stating that $<$ well-orders the universe. In this section we work in ZFO. Although ZFO might seem excessive as a metatheory for the purpose of proving normalization of a constructive system, we remark that with a bit more effort and slightly more obscure presentation, we could carry out the proof in ZFC.

Definition 4.1 *If $\phi(a)$ is a ZFO formula, then* "the first a such that ϕ" *is defined to be:*

- *The empty set, if there is no A such that $\phi(A)$.*
- *The smallest set A in the ordering $<$ such that $\phi(A)$ holds, otherwise.*

Our realizers are lambda terms of λS. The set of realizers arises as an image of the term-erasing map which replaces all set terms in a term by $\emptyset$. The reason for the erasure is that set terms play no part in reductions and eliminating them makes the account much cleaner. We leave the judgment whether this presentation is better than those in [22, 23] to the reader. The result of erasure on the term T will be denoted by T'. It is defined inductively in an obvious way. We show several representative cases:

$$x' = x \qquad a' = \emptyset \qquad (M\ N)' = M'\ N' \qquad (\lambda a.\ M)' = \lambda a.\ M'$$

[3] Russell's paradox is not necessary to derive contradiction, as $\in$-induction together with $C \in C$ is also contradictory.

$$(\pi_1^{a.\phi}(M))' = \emptyset \qquad (t_A(\overline{u}))' = \emptyset \qquad (\lambda x : \tau.\, M)' = \lambda x : \tau'.\, M'$$
$$(\pi_2^{a.\phi}(M))' = \pi_2^{a.\phi'}(M') \quad (\mathrm{axRep}(t, \overline{u}, M))' = \mathrm{axRep}(t', \overline{u'}, M')$$

Definition 4.2 *The set R consists of all closed terms in the range of the erasing map. A* realizer *is any element of R.*

We state several easy properties of the erasure map.

Lemma 4.3 *For any M, M' does not have any free set variables.*

Lemma 4.4 *R is closed under reductions: if $M \in R$ and $M \to N$, then $N \in R$.*

Lemma 4.5 *If M' normalizes, then so does M.*

4.1 Realizability relation

We proceed to define the realizability relation $M \Vdash_\rho \phi$, read as "M realizes ϕ", where M is a realizer and ϕ comes from the extended language L defined below. The definition and presentation are based heavily on our previous work [22, 23], originally inspired by McCarty's thesis [19].

Definition 4.6 *A set A is a λ-name iff A is a set of pairs (v, B) such that $v \in \lambda S_v \cap R$ and B is a λ-name.*

In other words, λ-names are sets hereditarily labelled by realizers that are λS values.

Definition 4.7 *The class of λ-names is denoted by V^λ.*

Formally, V^λ is generated by the following transfinite inductive definition on ordinals:

$$V^\lambda_\alpha = \bigcup_{\beta < \alpha} P(\lambda S_v \times V^\lambda_\beta) \qquad V^\lambda = \bigcup_{\alpha \in \mathrm{ORD}} V^\lambda_\alpha$$

We now extend the language of IZF$_D$ to encompass all λ-names as constants. We also restrict the formulas by allowing only the elements of R as arguments of $\pi_1^{a.\phi}([\circ])$. We call the resulting class-sized language L. Thus, the grammar is extended and modified by:

$$t ::= A \mid \pi_1^{a.\phi}(R) \mid \ldots$$

From now on until the end of this section, symbols M, N, O, P range exclusively over realizers, letters a, b, c vary over set variables in the language, letters A, B, C vary over λ-names, letters ϕ, ψ over formulas in L and the letter ρ varies over finite partial functions from set variables to V^λ. We call such functions *environments*.

Definition 4.8 *For any formula ϕ of L, any set term t of L and ρ defined on all free variables of ϕ and t, we define by metalevel induction a realizability relation $M \Vdash_\rho \phi$ in an environment ρ and a meaning of a term $\llbracket t \rrbracket_\rho$ in an environment ρ.*

- $\llbracket a \rrbracket_\rho \equiv \rho(a)$
- $\llbracket A \rrbracket_\rho \equiv A$
- $\llbracket \omega \rrbracket_\rho$. *Omitted. See [23] for details.*
- $\llbracket \pi_1^{a.\phi}(M) \rrbracket_\rho$ *is the first A such that $M \downarrow [\emptyset, N]$ and $N \Vdash_\rho \phi[a := A]$.*
- $\llbracket t_A(\overline{u}) \rrbracket_\rho \equiv \{(\mathrm{axRep}(\emptyset, \overline{\emptyset}, N), B) \in R \times V^\lambda_\gamma \mid N \Vdash_\rho \phi_A(B, \llbracket \overline{u} \rrbracket_\rho)\}$. *The definition of the ordinal γ is similar to the one in [23].*
- $M \Vdash_\rho \bot \equiv \bot$
- $M \Vdash_\rho t \in_I s \equiv M \downarrow v \wedge (v, \llbracket t \rrbracket_\rho) \in \llbracket s \rrbracket_\rho$
- $M \Vdash_\rho t = s$ *and* $M \Vdash_\rho t \in s$ *are defined together by $\in$-induction. See [23] for details.*
- $M \Vdash_\rho \phi \vee \psi \equiv (M \downarrow \mathrm{inl}(M_1) \wedge M_1 \Vdash_\rho \phi) \vee (M \downarrow \mathrm{inr}(M_1) \wedge M_1 \Vdash_\rho \psi)$
- $M \Vdash_\rho (x : \phi) \wedge \psi \equiv M \downarrow \langle M_1, M_2 \rangle \wedge (M_1 \Vdash_\rho \phi) \wedge (M_2 \Vdash_\rho \psi[x := M_1])$
- $M \Vdash_\rho (x : \phi) \to \psi \equiv (M \downarrow \lambda x : \_.\ M_1) \wedge \forall N.\ (N \Vdash_\rho \phi) \to (M_1[x := N] \Vdash_\rho \psi[x := N])$
- $M \Vdash_\rho \exists a.\ \phi \equiv M \downarrow [\emptyset, N] \wedge \exists A.\ N \Vdash_\rho \phi[a := A]$
- $M \Vdash_\rho \forall a.\ \phi \equiv M \downarrow \lambda a.\ N \wedge \forall A.\ N \Vdash_\rho \phi[a := A]$

It is not difficult to show that the definition of realizability is well-founded. Therefore, (metalevel) inductive proofs on the definition of realizability are justified, such as the proof of the following lemma:

Lemma 4.9 $\llbracket t[a := s] \rrbracket_\rho = \llbracket t[a := \llbracket s \rrbracket_\rho] \rrbracket_\rho = \llbracket t \rrbracket_{\rho[a := \llbracket s \rrbracket_\rho]}$ *and $M \Vdash_\rho \phi[a := s]$ iff $M \Vdash_\rho \phi[a := \llbracket s \rrbracket_\rho]$ iff $M \Vdash_{\rho[a := \llbracket s \rrbracket_\rho]} \phi$.*

Proof Proceed as in [23], using Lemma 4.3 in the case $t = \pi_1^{b.\phi}(M)$. □

The following two easy lemmas state that realizability behaves similarly to saturated sets as far as reductions and normalization are concerned:

Lemma 4.10 *If $(M \Vdash_\rho \phi)$ then $M \downarrow$.*

Lemma 4.11 *If $M \to^* M'$ then $M' \Vdash_\rho \phi$ iff $M \Vdash_\rho \phi$.*

Realizability is also invariant with respect to reductions of lambda terms inside of set terms and formulas:

Lemma 4.12 *If $M \to^* N$, then $\llbracket t[x := M] \rrbracket_\rho = \llbracket t[x := N] \rrbracket_\rho$ and $O \Vdash_\rho \phi[x := M]$ iff $O \Vdash_\rho \phi[x := N]$.*

The following keystone in the normalization proof is proved exactly as in [23].

Lemma 4.13 $(M, C) \in \llbracket t_A(\overline{u}) \rrbracket_\rho$ *iff* $M = \mathrm{axRep}(\emptyset, \overline{\emptyset}, N)$ *and* $N \Vdash_\rho \phi_A(C, \llbracket \overline{u} \rrbracket_\rho)$.

5 Normalization

We are now ready to prove that λS normalizes, thus enabling program extraction from IZF$_D$ proofs. The environments in this section are finite partial functions which map set variables to V^λ and lambda variables to realizers. Any such environment can be used as a realizability environment by ignoring the mapping of lambda variables.

Definition 5.1 *For any term T with free lambda variables $x_1, \ldots, x_n$ and ρ defined on $x_1, \ldots, x_n$, $T[\rho]$ denotes $T[x_1 := \rho(x_i), \ldots, x_n := \rho(x_n)]$.*

For any formula ϕ of IZF$_D$ there is a natural corresponding formula $\hat{\phi}$ of L which results by replacing every $\pi_1^{a.\psi}(M)$ occuring in ϕ by $\pi_1^{a.\psi}(M')$.

Definition 5.2 *For a lambda term M, we write $\overline{M}$ to denote $M'[\rho]$, when ρ is clear from the context. Also, for a formula ϕ of IZF$_D$, we write $\overline{\phi}$ to denote $\hat{\phi}[\rho]$.*

Lemma 5.3 *For any ρ, $\overline{T[z := S]} = \overline{T}[z := \overline{S}]$.*

Definition 5.4 *For a sequent $\Gamma \vdash M : \phi$, $\rho \models \Gamma$ means that ρ is defined on $\mathrm{dom}(\Gamma)$, for all $(a_i, \mathrm{Set}) \in \mathrm{dom}(\Gamma)$, $\rho(a_i) \in V^\lambda$ and for all $(x_i, \phi_i) \in \Gamma$, $\rho(x_i) \Vdash_\rho \overline{\phi_i}$.*

Theorem 5.5 (Normalization) *If $\Gamma \vdash O : \vartheta$ then for all $\rho \models \Gamma$, $\overline{O} \Vdash_\rho \overline{\vartheta}$.*

Proof We proceed by metalevel induction on $\Gamma \vdash O : \vartheta$. Note first that by Lemmas 3.3 and 4.3, $\overline{O}$ is closed, thus it is a realizer. We only show the new cases compared to [23]. Case $\Gamma \vdash O : \vartheta$ of:

- $$\frac{\Gamma \vdash M : \exists a.\, \phi}{\Gamma \vdash \pi_2^{a.\phi}(M) : \phi[a := \pi_1^{a.\phi}(M)]}$$

  By Lemma 5.3, $\overline{(\phi[a := \pi_1^{a.\phi}(M)])} = \overline{\phi}[a := \pi_1^{a.\overline{\phi}}(\overline{M})]$. By the inductive hypothesis, $\overline{M} \Vdash_\rho \exists a.\, \overline{\phi}$, so $\overline{M} \downarrow [\emptyset, N]$ and there is some A such that $N \Vdash_\rho \overline{\phi}[a := A]$. Furthermore, $[\![\pi_1^{a.\overline{\phi}}(\overline{M})]\!]_\rho$ is the first A such that $\overline{M} \downarrow [\emptyset, Q]$ and $Q \Vdash_\rho \overline{\phi}[a := A]$, so also $N \Vdash_\rho \overline{\phi}[a := [\![\pi_1^{a.\overline{\phi}}(\overline{M})]\!]_\rho]$. By Lemma 4.9, $N \Vdash_\rho \overline{\phi}[a := \pi_1^{a.\overline{\phi}}(\overline{M})]$. Since $\pi_2^{a.\overline{\phi}}(\overline{M}) \to^* N$, by Lemma 4.11 $\pi_2^{a.\overline{\phi}}(\overline{M}) \Vdash_\rho \overline{\phi}[a := \pi_1^{a.\overline{\phi}}(\overline{M})]$, which shows the claim.

- $$\frac{\Gamma \vdash M : \phi \quad \Gamma \vdash N : \psi[x := M]}{\Gamma \vdash \langle M, N\rangle : (x : \phi) \wedge \psi}$$

  By the inductive hypothesis, $\overline{M} \Vdash_\rho \overline{\phi}$ and $\overline{N} \Vdash_\rho \overline{\psi[x := M]}$, thus also $\overline{N} \Vdash_\rho \overline{\psi}[x := \overline{M}]$, which is precisely what needs to be shown.

- $$\frac{\Gamma \vdash M : (x : \phi) \wedge \psi}{\Gamma \vdash \mathrm{fst}(M) : \phi}$$

  The proof is the same as in [22, 23].

- $$\frac{\Gamma \vdash M : (x : \phi) \wedge \psi}{\Gamma \vdash \mathrm{snd}(M) : \psi[x := \mathrm{fst}(M)]}$$

  By the inductive hypothesis, $\overline{M} \downarrow \langle M_1, M_2\rangle$ and $M_2 \Vdash_\rho \overline{\psi}[x := \overline{M_1}]$. As $\mathrm{snd}(\overline{M}) \to^* M_2$, by Lemma 4.11 it suffices to show that $M_2 \Vdash_\rho \overline{\psi[x := \mathrm{fst}(M)]}$, which is equivalent to $M_2 \Vdash_\rho \overline{\psi}[x := \mathrm{fst}(\overline{M})]$. Since $\mathrm{fst}(\overline{M}) \to^* M_1$ and $\overline{\psi[x := \mathrm{fst}(M)]} = \overline{\psi}[x := \mathrm{fst}(\overline{M})]$, Lemma 4.12 shows the claim.

- $$\frac{\Gamma \vdash M : (x : \phi) \to \psi \quad \Gamma \vdash N : \phi}{\Gamma \vdash M\ N : \psi[x := N]}$$

  By the inductive hypothesis, for some ϕ_1, $\overline{M} \downarrow \lambda x : \phi_1.\, M_1$, $\overline{N} \Vdash_\rho \overline{\phi}$ and for all $P \Vdash_\rho \overline{\phi}$, $M_1[x := P] \Vdash_\rho \overline{\psi}[x := P]$. Thus in particular $M_1[x := \overline{N}] \Vdash_\rho \overline{\psi}[x := \overline{N}]$. As $\overline{M\ N} = \overline{M}\ \overline{N} \to^* (\lambda x : \phi_1.\, M_1)\ \overline{N} \to M_1[x := \overline{N}]$, Lemmas 4.11 and 5.3 show the claim.

- $$\frac{\Gamma, x : \phi \vdash M : \psi}{\Gamma \vdash \lambda x : \phi.\, M : (x : \phi) \to \psi}$$

  Take any $\rho \models \Gamma$. We need to show that for any $N \Vdash_\rho \overline{\phi}$, $\overline{M}[x := N] \Vdash_\rho \overline{\psi}[x := N]$. Take any such N. Since $\rho[x := N] \models \Gamma, x : \phi$, by the inductive hypothesis $M'[\rho[x := N]] \Vdash_\rho \hat{\psi}[\rho[x := N]]$. It is easy to see that this is equivalent to $\overline{M}[x := N] \Vdash_\rho \overline{\psi}[x := N]$.

- The cases corresponding to the axRep and axProp terms are handled as in [22, 23], using Lemma 4.13.□

Corollary 5.6 (Normalization) *If $\vdash M : \phi$, then $M' \downarrow$ and thus also $M \downarrow$.*

Corollary 5.7 *IZF$_D$ is consistent.*

Proof If $\vdash M : \bot$, then $\overline{M} \Vdash_\rho \bot$, which is not the case. □

5.1 Program extraction

We now briefly explain how to use the normalization result for the purpose of program extraction from IZF$_D$ proofs. For a natural number n, let $\overline{n}$ denote the IZF$_D$ numeral corresponding to n. We will need the following instance of Lemma 4.13:

Lemma 5.8 $(M, C) \in [\![\omega]\!]_\rho$ *iff* $M = \mathrm{infRep}(\emptyset, N)$ *and* $N \Vdash_\rho C = 0 \vee \exists y.\, y \in \omega \wedge C = S(y)$.

An easy consequence is Numerical Existence Property for the realizability model:

Lemma 5.9 *If $M \Vdash_\rho \exists y \in \omega.\ \phi$, then one can obtain a number n and a realizer O such that $O \Vdash_\rho \phi[y := \overline{n}]$.*

Proof[Sketch] The process of obtaining n and O is essentially the procedure described in the proof of Numerical Existence Property for IZF$_R$ in [22], using Lemma 5.8 and replacing applications of Term Existence Property by applications of Definition 4.1. □

We show one example of extraction, referring the reader to [9] for more general account, including extraction of higher-order functions. Suppose IZF$_D \vdash M : \forall x \in \omega \exists y \in \omega.\ \phi$. From this proof, we extract a function $f : \mathrm{nat} \to \mathrm{nat}$ which works as follows. It takes a natural number n as an argument. It constructs an IZF$_D$ proof $\vdash N : \overline{n} \in \omega$. Then $\vdash M\ \overline{n}\ N : \exists y \in \omega.\ \phi[x := \overline{n}]$. By Theorem 5.5, $\overline{M}\ \emptyset\ \overline{N} \Vdash_\rho \exists y \in \omega.\ \overline{\phi}[x := \overline{n}]$. Using Lemma 5.9, we obtain a natural number m along with a realizer O such that $O \Vdash_\rho \overline{\phi}[x, y := \overline{n}, \overline{m}]$. The function f returns m.

The key property of IZF$_D$, which makes it possible to utilize our account from [9], is Term Existence Property, internalized by $\pi_1^{a.\phi}$ terms.

6 The properties of IZF$_D$

In this section, we relate IZF$_D$ and its classical counterpart to well-known first-order set theories.

Theorem 6.1 *IZF$_D$ interprets IZF$_C$.*

Proof The precise formulation of the claim is: if IZF$_C \vdash \phi$, then for some term M, IZF$_D \vdash M : \phi$. We formulate IZF$_C$ as IZF$_R$ extended with the Collection axiom schema:

$$(\mathrm{COLL}_{\phi(x,y,\overline{f})})\ \forall \overline{f}.\ \forall a.\ (\forall x \in a \exists y.\ \phi) \to \exists b.\ \forall x \in a \exists y \in b.\ \phi$$

To show that IZF$_D$ interprets IZF$_R$, we first need to prove that it interprets the rules of first-order logic. Most of them are present in the type system of λS as special cases when dependencies are not used. The only missing rule is elimination of the existential quantifier:

$$\frac{\Gamma \vdash \exists a.\ \phi \quad \Gamma \vdash \forall a.\ \phi \to \psi}{\Gamma \vdash \psi}\ a \notin FV(\psi)$$

It is easy to show that in IZF$_D$ the following rule is admissible, that is if assumptions are derivable, then so is the conclusion:

$$\frac{\Gamma \vdash M : \exists a.\ \phi \quad \Gamma \vdash N : \forall a.\ \phi \to \psi}{\Gamma \vdash N\ (\pi_1^{a.\phi}(M))\ (\pi_2^{a.\phi}(M)) : \psi}\ a \notin FV(\psi)$$

Second, we need to give the interpretation of IZF$_R$ terms in IZF$_D$ and show that they satisfy the respective axioms. This is straightforward, as it suffices to add extraneous binders for Separation and Replacement terms. For example, we interpret $\{x \in a \mid \phi\}$ as $\{p : x \in a \mid \phi\}$, where p is fresh.

The only nontrivial thing left is the interpretation of the Collection axiom. Intuitively, it follows from Replacement, as using dependent implication and $\pi_1^{a.\phi}$ terms, we can transform a proof p of $\forall x \in a \exists y.\ \phi$ into $\forall q : x \in a \exists ! y.\ \phi \wedge y = \pi_1^{a.\phi}(p\ x\ q)$. Formally, we exhibit the proof terms. To increase readability, we display $\forall x.\ (p : x \in a) \to \phi$ as $\forall p : x \in a.\ \phi$, $\exists x.\ (p : x \in a) \wedge \phi$ as $\exists p : x \in a.\ \phi$ and $R_\phi(t, \overline{t})$ as $\{y \mid (\forall p : x \in t \exists ! y.\ \phi[\overline{f} := \overline{t}]) \wedge (\exists p : x \in a.\ \phi[\overline{f} := \overline{t}])\}$.

$$\begin{array}{rcl}
M_1 & \equiv & \langle \pi_2^{y.\phi}(p\ x\ q), \mathrm{eqRefl}\ \pi_1^{y.\phi}(p\ x\ q) \rangle \\
M_2 & \equiv & \lambda z.\ \lambda r : \phi[y := z] \wedge z = \pi_1^{y.\phi}(p\ x\ q).\ \mathrm{snd}(r) \\
M_3 & \equiv & \lambda x.\ \lambda q : x \in a.\ [\pi_1^{y.\phi}(p\ x\ q), \langle M_1, M_2 \rangle] \\
M_4 & \equiv & [x, \langle q, \langle \pi_2^{y.\phi}(p\ x\ q), \mathrm{eqRefl}\ \pi_1^{y.\phi}(p\ x\ q) \rangle \rangle] \\
M_5 & \equiv & \mathrm{replRep}(\pi_1^{y.\phi}(p\ x\ q), a, \overline{f}, \langle M_3, M_4 \rangle) \\
M_6 & \equiv & \lambda x.\ \lambda q : x \in a.\ [\pi_1^{y.\phi}(p\ x\ q), \langle M_5, \pi_2^{y.\phi}(p\ x\ q) \rangle] \\
\psi & \equiv & \phi \wedge y = \pi_1^{y.\phi}(p\ x\ q) \\
t & \equiv & \{y \mid (\forall q : x \in a \exists ! y.\ \psi) \wedge \exists q : x \in a.\ \psi\} \\
N & \equiv & \lambda \overline{f}.\ \lambda p : \forall x \in a \exists y.\ \phi.\ [t, M_6] \\
\vdash N & : & \forall \overline{f}.\ (\forall x \in a \exists y.\ \phi) \to \exists b.\ \forall x \in a \exists y \in b.\ \phi
\end{array} \quad \square$$

Therefore, by results of Friedman [12], the proof-theoretical strength of IZF$_D$ equals that of ZFC.

We now consider a classical version of IZF$_D$. Let ZF$_D$ be IZF$_D$ extended with the excluded middle axiom EM. We show that ZF$_D$ is consistent. For this purpose, take a formulation of ZFO with set terms, such as IZF$_R$ from [22] + EM + "the universe is well-ordered by $<$". Define an erasure map on formulas and set terms of λS, which returns formulas and set terms of ZFO. The representative cases of the definition follow, where $\iota a.\phi$ denotes "the first a such that ϕ":

$$\overline{a} = a \quad \overline{\pi_1^{a.\phi}(M)} = \iota a.\phi \quad \overline{\emptyset} = \emptyset \quad \overline{\{t_1, t_2\}} = \{\overline{t_1}, \overline{t_2}\} \quad \overline{\omega} = \omega$$

$$\overline{(p : \phi) \to \psi} = \overline{\phi} \to \overline{\psi} \quad \overline{S_{p,a,\overline{f}.\phi}(u, \overline{t})} = \{a \in \overline{u} \mid \overline{\phi(z, a, \overline{t})}\}$$

$$\overline{R_{p,a,b,\overline{f}.\phi}(u, \overline{t})} = R_{a,b,\overline{f}.\ \overline{\phi}}(\overline{u}, \overline{\overline{t}})$$

With the map at hand, we can easily prove by induction on the proof the consistency result:

Theorem 6.2 *If ZF$_D \vdash t$: Set, then t is a term of ZFO. If ZF$_D \vdash M : \phi$, then ZFO $\vdash \overline{\phi}$. Thus ZF$_D$ is consistent.*

Theorem 6.3 *ZF$_D$ interprets ZF.*

Proof By Theorem 6.1, IZF$_D$ interprets IZF$_C$. Since ZF=IZF$_C$ + EM, the claim follows. □

7 The definition mechanism

A crucial feature of first-order set theories, which makes formalization of mathematics so convenient, is the mechanism of definitions. If a theory T proves a statement

$\exists ! a.\ \phi(a)$, then a new constant c can be introduced along with the defining axiom $\phi(t)$. More importantly, if a statement $\forall \overline{x} \exists ! y.\ \phi(x, y)$ is provable, the introduction of a new function symbol f along with the axiom $\forall \overline{x}.\ \phi(x, f(x))$, is justified. The Definition Elimination theorem guarantees the safety of this extending process.

There are several problems with this approach. First, it adds an extra layer on top of first-order logic, which makes the account and possible implementation more difficult. Moreover, the "domain" of new function symbols is the entire universe: in set theory, nothing prevents a user from using terms such as $5 + P(\omega)$. More importantly, it is not known how to use this mechanism while preserving the capability of program extraction. Finally, there is also a more insidious, fundamental problem with "partial" function symbols, which seems to make it impossible in general to utilize the mechanism in constructive set theories.

Consider a very simple example — a definition of the function symbol f corresponding to the function which assigns 5 to every natural number. The standard approach[4] defines f to be $\emptyset$ on any set out of its intended domain. Thus, in order to introduce f, one first needs to prove the formula

$$\phi \equiv \forall x \exists y.\ ((x \in \omega \to y = 5) \land (x \notin \omega \to y = \emptyset)) \land \\ \forall z.\ ((x \in \omega \to z = 5) \land (x \notin \omega \to z = \emptyset)) \to z = y.$$

Constructively, a proof of the second part of ϕ is problematic. Note that in order to obtain any information about z, one first needs to know whether $x \in \omega$ or not and this knowledge is unavailable in the constructive world. We conjecture that it is impossible to prove ϕ in constructive set theories.

IZF$_D$ provides a solution to all these problems. For our example, we can simply prove a formula $\vdash M : \forall x \in \omega \exists ! y.\ y = 5$. Then M itself can be used to provide a "typed" function symbol, as for any set x along with a proof p of its membership in ω, we can show that $\pi_1^{y.\ y=5 \land \forall z.\ z=5 \to z=y}(M\ x\ p)$ is the unique set equal to 5.

The benefits of this approach are manifold. First, nonsense applications of term symbols, such as $5 + P(\omega)$, are outlawed, as all new function symbols are automatically "typed". Second, all features necessary for providing the mechanism of definitions are already present in IZF$_D$, so there is no need for description and implementation of an extra layer on top of the theory. Last, but definitely not least, the program extraction capability remains intact.

The price to pay is the possibly problematic interaction with the Leibniz axiom, discussed in Section 3.3. It remains to be seen, however, if any significant difficulties would arise in practice.

8 Related work

The theories IZF$_C$ and IZF$_R$ are well-investigated. Research up to 1985 is presented in [7, 33]. Recent results include demonstration of the disjunction, numerical existence and related properties for IZF$_C$ extended with various choice axioms [29] and normalization of IZF$_R$ extended with inaccessible sets [22, 23].

There is a significant amount of research on connections between type and set theories. Aczel [2, 3] described mutual interpretations of variants of CZF and Martin-Löf type theory. Werner [34] did the same thing for Zermelo set theory and Calculus of Constructions. Miquel [20, 21] investigated embeddings of impredicative set theories without $\in$-induction axiom schema in type theories. Howe [17] investigated an extension of the set theoretic universe with type-theoretical constructs in order to validate the type theory of Nuprl.

Modifications of logic underlying set theory were investigated before. Agerholm and Gordon [5, 15] studied classical higher-order set theory HOL-ST. They did not find a clear advantage of HOL-ST over first-order ZF. A map theory [16] provides a unified framework for sets and computation. An ongoing research on algebraic set theory [24] investigates set theories based on category theory. There are also set theories based on linear logics [30, 31].

There are several known paradoxes in type theories [11, 27], which show that extending the power of type theory is a precarious activity, easily leading to contradiction. The particulars of our paradox seem to be unrelated to these results, as we utilize very set-theoretical combination of Russell's paradox, Replacement and Separation axioms.

9 Conclusion

We have shown that extending set theory with type-theoretic features yields many benefits, from both theoretical and practical points of view. We leave several questions open:

- Can the normalization proof be conducted in IZF$_D$?
- Can constructive mathematics be smoothly developed in IZF$_D$?
- Can IZF$_D$ be useful in system development along the lines of the B-Tool [1]?
- The inconsistency in Section 3.4 utilizes a dependent nature of the Replacement axiom. Is there a consistent, possibly weaker, dependent set theory with unrestricted Σ-types?
- Is IZF$_D$ conservative over IZF$_C$?

[4] We have recently discovered an alternative approach which would work in strong, impredicative set theories such as IZF$_R$. However, it is not applicable to weaker set theories such as CZF.

While we conjecture that the answers to the first two questions are positive, we do not have intuitions regarding the rest of them.

Acknowledgements I would like to thank my advisor, Bob Constable, for suggestions and support, Richard Shore and Andrew Myers for valuable discussions and comments and anonymous referees for helpful comments.

References

[1] J.-R. Abrial. *The B-book: assigning programs to meanings*. Cambridge University Press, New York, NY, USA, 1996.

[2] P. Aczel. The type theoretic interpretation of constructive set theory. In A. MacIntyre, L. Pacholski, and J. Paris, editors, *Logic Colloquium '77*. North Holland, 1978.

[3] P. Aczel. On relating type theories and set theories. In *TYPES '98: Selected papers from the International Workshop on Types for Proofs and Programs*, pages 1–18, London, UK, 1999. Springer-Verlag.

[4] P. Aczel and M. Rathjen. Notes on constructive set theory. Technical Report 40, Institut Mittag-Leffler (The Royal Swedish Academy of Sciences), 2000/2001.

[5] S. Agerholm and M. J. C. Gordon. Experiments with ZF Set Theory in HOL and Isabelle. In *Proc. of the 8th Int. Workshop on Higher Order Logic Theorem Proving and Its Applications*, pages 32–45, London, UK, 1995. Springer-Verlag.

[6] S. Allen, M. Bickford, R. Constable, R. Eaton, C. Kreitz, L. Lorigo, and E. Moran. Innovations in computational type theory using Nuprl. *Journal of Applied Logic*, 4(4):428–469, 2006.

[7] M. Beeson. *Foundations of Constructive Mathematics*. Springer-Verlag, 1985.

[8] M. Bickford and R. L. Constable. A Logic of Events. Technical Report TR2003-1893, Cornell University, 2003.

[9] R. Constable and W. Moczydłowski. Extracting Programs from Constructive HOL Proofs via IZF Set-Theoretic Semantics. In *Proc. 3rd Int. Joint Conf. on Automated Reasoning (IJCAR 2006)*, volume 4130 of *LNCS*, pages 162–176. Springer, 2006.

[10] R. L. Constable et al. *Implementing Mathematics with the Nuprl Proof Development System*. Prentice-Hall, NJ, 1986.

[11] T. Coquand. An Analysis of Girard's Paradox. In *Proc. 1st Ann. IEEE Symposium on Logic in Computer Science*, pages 227–236. IEEE Computer Society Press, June 1986.

[12] H. Friedman. The consistency of classical set theory relative to a set theory with intuitionistic logic. *Journal of Symbolic Logic*, 38:315–319, 1973.

[13] H. Friedman and A. Ŝĉedrov. The lack of definable witnesses and provably recursive functions in intuitionistic set theories. *Advances in Mathematics*, 57:1–13, 1985.

[14] G. Gonthier. A computer-checked proof of the Four Colour Theorem. Preprint, 2005.

[15] M. Gordon. Set Theory, Higher Order Logic or Both? In *TPHOLs '96: Proc. of the 9th Int. Conf. on Theorem Proving in Higher Order Logics*, volume 1125 of *LNCS*, pages 191–202. Springer-Verlag, 1996.

[16] K. Grue. Map theory. *Theor. Comput. Sci.*, 102(1):1–133, 1992.

[17] D. J. Howe. Semantic foundations for embedding HOL in Nuprl. In M. Wirsing and M. Nivat, editors, *Algebraic Methodology and Software Technology*, volume 1101 of *LNCS*, pages 85–101. Springer-Verlag, Berlin, 1996.

[18] C. Kreitz. *The Nuprl Proof Development System, Version 5: Reference Manual and User's Guide*. Department of Computer Science, Cornell University, December 2002.

[19] D. McCarty. *Realizability and Recursive Mathematics*. D.Phil. Thesis, University of Oxford, 1984.

[20] A. Miquel. A Strongly Normalising Curry-Howard Correspondence for IZF Set Theory. In *Proc. of 12th Ann. Conf. of the EACSL (CSL 2003)*, volume 2803 of *LNCS*, pages 441–454. Springer, 2003.

[21] A. Miquel. Lambda-Z: Zermelo's Set Theory as a PTS with 4 Sorts. In J.-C. Filliâtre, C. Paulin-Mohring, and B. Werner, editors, *TYPES*, volume 3839 of *Lecture Notes in Computer Science*, pages 232–251. Springer, 2004.

[22] W. Moczydłowski. Normalization of IZF with Replacement. In *Proc. 15th Ann. Conf. of the EACSL (CSL 2006)*, volume 4207 of *Lecture Notes in Computer Science*. Springer, 2006.

[23] W. Moczydłowski. A Normalizing Intuitionistic Set Theory with Inaccessible Sets. Technical Report TR2006-2051, Cornell University, 2006. In submission.

[24] I. Moerdijk and E. Palmgren. Type theories, toposes and constructive set theory: predicative aspects of AST. *Annals of Pure and Applied Logic*, 114:155–201, 2002.

[25] M. Muzalewski. *An Outline of PC Mizar*. Foundation of Logic, Mathematics and Informatics, Mizar User Group, Brussels, 1993.

[26] J. Myhill. Some properties of intuitionistic Zermelo-Fraenkel set theory. In *Cambridge Summer School in Mathematical Logic*, volume 29, pages 206–231. Springer, 1973.

[27] A. M. Pitts. Non-trivial power types can't be subtypes of polymorphic types. In *4th Annual Symposium on Logic in Computer Science*, pages 6–13. IEEE Computer Society Press, Washington, 1989.

[28] M. Rathjen. Generalized inductive definitions in constructive set theory. In L. Crosilla and P. Schuster, editors, *From Sets and Types to Topology and Analysis: Towards Practicable Foundations for Constructive Mathematics*. Oxford University Press, 2005.

[29] M. Rathjen. Metamathematical properties of intuitionistic set theories with choice principles. 2006. Manuscript, available from the web page of the author.

[30] M. Shirahata. *Linear Set Theory*. PhD thesis, 1994.

[31] K. Terui. Light affine set theory: A naive set theory of polynomial time. *Studia Logica*, 77(1):9–40, 2004.

[32] The Coq Development Team. *The Coq Proof Assistant Reference Manual – Version V8.0*, Apr. 2004.

[33] A. Ščedrov. Intuitionistic set theory. In *Harvey Friedman's Research on the Foundations of Mathematics*, pages 257–284. Elsevier, 1985.

[34] B. Werner. Sets in types, types in sets. In *TACS '97: Proc. of the 3rd Int. Symposium on Theoretical Aspects of Computer Software*, pages 530–546. Springer-Verlag, 1997.