# The extended predicative Mahlo universe in Martin-Löf type theory

PETER DYBJER, *Department of Computer Science and Engineering, Chalmers University of Technology, Gothenburg, Sweden.*

ANTON SETZER, *Department of Computer Science, Swansea University, Swansea, UK.*

## Abstract

This paper addresses the long-standing question of the predicativity of the Mahlo universe. A solution, called the extended predicative Mahlo universe, has been proposed by Kahle and Setzer in the context of explicit mathematics. It makes use of the collection of untyped terms (denoting partial functions) which are directly available in explicit mathematics but not in Martin-Löf type theory. In this paper, we overcome the obstacle of not having direct access to untyped terms in Martin-Löf type theory by formalizing explicit mathematics with an extended predicative Mahlo universe in Martin-Löf type theory with certain indexed inductive-recursive definitions. In this way, we can relate the predicativity question to the fundamental semantics of Martin-Löf type theory in terms of computation to canonical form. As a result, we get the first extended predicative definition of a Mahlo universe in Martin-Löf type theory. To this end, we first define an external variant of Kahle and Setzer's internal extended predicative universe in explicit mathematics. This is then formalized in Martin-Löf type theory, where it becomes an internal extended predicative Mahlo universe. Although we make use of indexed inductive-recursive definitions that go beyond the type theory **IIRD** of indexed inductive-recursive definitions defined in previous work by the authors, we argue that they are constructive and predicative in Martin-Löf's sense. The model construction has been type-checked in the proof assistant Agda.

*Keywords*: Martin-Löf type theory, Mahlo, universes, meaning explanations, extended predicativity, predicativity, explicit mathematics, inductive-recursive definitions, indexed induction-recursion, constructive mathematics, Agda, partial functions

## 1 What is predicativity?

Theories with proof-theoretic strength up to $\Gamma_0$ are said to be predicative in Feferman and Schütte's sense. For example, the original version of Martin-Löf type theory with an infinite tower of universes [21] has strength $\Gamma_0$ and is thus predicative in this sense. However, when Martin-Löf [22] added W-types, its proof-theoretic strength [28] increased and the theory became impredicative in Feferman and Schütte's sense. Nevertheless, Martin-Löf still called his new theory 'predicative'. A reason for this is that he [22] provided constructive semantic foundations in terms of computations to canonical form of the expressions of his theory and a resulting account of the sense in which the objects of the theory are 'built up from below'. As we shall see in the next section, Martin-Löf's theory was later extended with progressively stronger constructions, such as superuniverses [25], Mahlo universes [29, 30] and general inductive-recursive definitions [8–11]. The claim is that Martin-Löf's constructive semantic foundations can be extended so that these constructions are covered too and thus can be seen to be predicative. Nevertheless, the predicativity of the Mahlo universe is a non-trivial issue and this paper is aimed at throwing some light on it.

Vol. 00, No. 00, © The Author(s) 2023. Published by Oxford University Press.

https://doi.org/10.1093/logcom/exad022

In what sense are the objects of Martin-Löf's theories built up from below? Consider first natural numbers. These are built up from below, since a natural number is either 0 or (s $n$)[1] where $n$ is already a natural number built up from below. In Martin-Löf's constructive semantics $a : \mathbb{N}$ provided it has canonical form 0 or (s $b$), where $b : \mathbb{N}$. Similarly, we have $a : \mathrm{W}x : B.C$ provided $a$ has canonical form (sup $b\ f$) where $b : B$ and $f\ x : \mathrm{W}x : B.C$ for all $x : C[b/x]$. Just as for natural numbers, we can see elements of W-types as being built up from below, because if we already have built $b$ and we have built ($f\ x$) for all $x : C[b/x]$, then we can build the compound element (sup $b\ f$). This is to be understood in terms of computation to canonical form and can be visualized as building a well-founded tree for (sup $b\ f$) with subtrees $b$ and ($f\ x$) for all $x : C[b/x]$. In the next section, we will present universe constructions. They are predicative in a similar sense as the W-types.

To avoid confusion with Feferman–Schütte's predicativity, one sometimes refers to Martin-Löf's notion as 'extended predicativity'. This is what Kahle and Setzer [20] have in mind when they call their new version of a Mahlo universe in explicit mathematics 'extended predicative'. Their axioms are abstracted from a model construction that describes how their Mahlo universe is built up from below. However, the connection with Martin-Löf type theory and its extended predicativity was left implicit. By implementing explicit mathematics with Kahle and Setzer's Mahlo universe in an extension of Martin-Löf type theory, we make this connection explicit. We here make use of certain strong indexed inductive-recursive definitions going beyond our theory **IIRD** [12] of indexed inductive-recursive definitions. (It is another source of confusion that the term 'inductive-recursive definition' is ambiguous. Sometimes it refers to the type theories of inductive-recursive definitions formally defined in our papers, such as **IR** and **IIRD**. At other times, it refers to definitions which are outside those theories and are still inductive-recursive in that they simultaneously inductively define one or more types and one or more functions by recursion on the way the elements of those types are generated.)

This paper investigates the higher infinite in type theory, where we need to extend the already proof-theoretically strong theories **IR** and **IIRD**.[2] A key point is to be constructive and predicative in Martin-Löf's sense. This amounts to explaining the meaning of the judgments of the theory in terms of computation of canonical forms following the ideas described by Martin-Löf [22, 23] for the basic type theories. However, similar meaning explanations can be given for inductive definitions in type theory more generally. They can also be given for many universe constructions and their generalization to inductive-recursive definitions, although Mahlo universes and similar constructions pose special problems related to the discussion above. The present paper is an attempt to throw light on these problems.

**Plan of the paper.** In Section 2, we recall how universes, including the Mahlo universe, are defined in type theory. In Section 3, we present Feferman's system of explicit mathematics. In Section 4, we present Jäger, Strahm and Studer's [18, 19] axiomatic external Mahlo universe and Tupailo's [34] internal version of it. In Section 5, we present Kahle and Setzer's extended predicative internal

[1] We use in this article functional notation writing (s $n$) for the application of s to $n$. The brackets around s $n$ in running text are used to improve readability. Martin-Löf uses mathematical notation s($n$).

[2] Setzer has shown in [11] that a lower bound for the proof theoretic strength of **IR** is that of KPM. An upper bound has not been worked out yet. However, adapting the set theoretic models of **IR** and **IIRD** to a model in Kripke Platek set theory should determine as an upper bounds for both theories that of KP plus one recursively Mahlo ordinal and finitely many admissibles above it, details still need to be worked out. This leaves open the possibility that **IR** and **IIRD** reaches the strength of type theory with one Mahlo universe. However, one can easily extend the strength of the type theory with one Mahlo universe by adding universes on top of the Mahlo universe and therefore reach a strength which, if the conjectured upper bound is correct, would go beyond the strengths of **IR** and **IIRD**. This means that the principle of a Mahlo universe goes proof-theoretically beyond induction-recursion.

Mahlo universe in explicit mathematics. In Section 6, we present our new external version of Kahle and Setzer's Mahlo universe. In Section 7, we show how to formalize the external version in Martin-Löf type theory with indexed inductive-recursive definitions in a general sense. However, these definitions go beyond our theory **IIRD** of indexed inductive-recursive definitions. The formalization is carried out in the proof assistant Agda [2]. In Sections 8 and 9, we introduce a slightly refined version of the external extended predicative Mahlo first in explicit mathematics and then in type theory. In Section 10, we show that the external extended predicative Mahlo universe in type theory is closed under the rules for axiomatic Mahlo. In Section 11, we argue that this indexed inductive-recursive definition is constructively valid and sketch meaning explanations for it in the sense of Martin-Löf [22, 23]. We end the main part with a conclusion in Section 12. In Appendix A, we give more details about the formalisation of terms in type theory. In Appendix B, we give the complete set of rules for the resulting type theory.

**Git repository.** A git repository of the Agda code used in this paper can be found at [13].

## 2 Universes in type theory

We shall now present the development of universes and inductive-recursive definitions in type theory.

**Universes à la Russell.** In Martin-Löf type theory, a *universe à la Russell* [24] is a type of small types much like a *Grothendieck universe* in set theory. In the logical framework formulation of Martin-Löf type theory, one has a special type Set of sets, where a set should be understood as a small type.

Just as a Grothendieck universe is closed under the basic operations for forming small sets, a type-theoretic universe is closed under the basic constructs for forming sets, such as $\Pi, \Sigma, +$, etc. Type-theoretic universes are also closely related to *large cardinals* in ZF set theory and *admissibles* in Kripke–Platek set theory. Universes increase the proof-theoretic strength of type theory.

**Universes à la Tarski.** Martin-Löf [23] introduced *universes à la Tarski* as an alternative to the original version à la Russell. A universe à la Tarski is a family of sets $(\mathrm{U}, \mathrm{T})$, where U is a set of codes (or names), and T is a decoding function that maps codes to the sets they denote (their extension). For instance, $\widehat{\mathbb{N}} : \mathrm{U}$ is a code for the set of natural numbers, and $\mathrm{T}\ \widehat{\mathbb{N}} = \mathbb{N}$.

Palmgren [25] introduced a *next universe operator* for universes à la Tarski. It maps any family of sets to a universe containing it. In particular, it maps a universe $(\mathrm{U}_i, \mathrm{T}_i)$ to the next universe $(\mathrm{U}_{i+1}, \mathrm{T}_{i+1})$. Given a first universe à la Tarski $(\mathrm{U}_0, \mathrm{T}_0)$, we can construct an internal countable hierarchy of universes $(\mathrm{U}_0, \mathrm{T}_0), (\mathrm{U}_1, \mathrm{T}_1), (\mathrm{U}_2, \mathrm{T}_2), \ldots$ by repeated application of the next universe operator. It is more general than externally indexed towers of universes by giving rise to universes that contain any family of universes indexed over a given set. For instance, one can form a universe containing all $(\mathrm{U}_i, \mathrm{T}_i)$ for $i \in \mathbb{N}$ as elements.

A further step was taken by Palmgren who introduced the *super universe* $(\mathrm{V}_0, \mathrm{S}_0)$, which is a universe closed under the next universe operator.

One can then go on and define a *next super universe operator* that maps any type family to a super universe containing it. In this way, we can form an increasing sequence of super universes $(\mathrm{V}_0, \mathrm{S}_0), (\mathrm{V}_1, \mathrm{S}_1), (\mathrm{V}_2, \mathrm{S}_2), \ldots$. Furthermore, we can introduce a super-super-universe, that is, a universe closed under the next super universe operator. This process can then be iterated so that we get super$^n$-universes.

**External Mahlo universe.** The next step beyond super universes is to build universes $(\mathrm{U}_f, \mathrm{T}_f)$ closed under an arbitrary family operator $f$, and not only when $f$ is the next universe operator, the next super universe operator, etc. The rules for the universes $(\mathrm{U}_f, \mathrm{T}_f)$ can be formalized in Martin-Löf's logical framework, a basic type theory with dependent function types $(x : A) \to B$, dependent product types $(x : A) \times B$ and the type Set of sets. If we define $\mathrm{Fam(Set)} = (A : \mathrm{Set}) \times (A \to \mathrm{Set})$, we can form a universe $(\mathrm{U}_f, \mathrm{T}_f) : \mathrm{Fam(Set)}$ for any $f : \mathrm{Fam(Set)} \to \mathrm{Fam(Set)}$. This formalisation is analogous to the formation of Mahlo cardinals in set theory, and one can call Set therefore a Mahlo universe. It is a large universe, because Set is no longer a set but a type, and therefore, we call it the *external* Mahlo universe.

**Internal Mahlo universe.** An even more radical step is to build a universe which is a set, which contains all the universes $(\mathrm{U}_f, \mathrm{T}_f)$ as subuniverses. We call it the *internal Mahlo universe*. More precisely, one defines a Mahlo universe $(\mathrm{M}, \mathrm{T_M}) : \mathrm{Fam(Set)}$, where $\mathrm{M} : \mathrm{Set}$ and $\mathrm{T_M} : \mathrm{M} \to \mathrm{Set}$ is the decoding. This Mahlo universe has subuniverses $(\mathrm{U}_f, \mathrm{T}_f) : \mathrm{Fam(M)}$ for arbitrary $f : \mathrm{Fam(M)} \to \mathrm{Fam(M)}$, where $\mathrm{Fam(M)} = (A : \mathrm{M}) \times (A \to \mathrm{M})$. This construction was introduced by Setzer [29, 30] and was proof theoretically analysed based on Rathjen's proof theoretic analysis of Kripke–Platek set theory with a recursively Mahlo ordinal [26, 27].

**Induction-recursion.** The general notion of an inductive-recursive definition [8] plays a crucial role in this paper. The authors defined general theories of inductive-recursive definitions **IR** [9, 11] and of indexed inductive-recursive definitions **IIRD** [12]. The consistency of **IR** and **IIRD** was proved by constructing a model in ZF set theory with a Mahlo cardinal and an inaccessible above it [9, 11].

The theory of inductive-recursive definitions is formulated in the setting of Martin-Löf's logical framework. An inductive-recursive definition consists of an inductively defined set $U$ : Set together with a recursively defined map $T : U \to D$, where $D$ is a type. It can be shown that ordinary universes à la Tarski and super$^n$-universes can be defined in the theory **IR**. It can also be shown that the universes $(\mathrm{U}_f, \mathrm{T}_f)$ can be defined as an inductive-recursive definition in **IR**. As already mentioned, Set becomes an external Mahlo universe in this way. However, the internal Mahlo universe $(\mathrm{M}, \mathrm{T_M})$ takes us outside the theory **IR** of inductive-recursive definitions and further increases the proof-theoretic strength of the type theory. It is worth noting that $(\mathrm{M}, \mathrm{T_M})$ still has an inductive-recursive character. As already mentioned above, we must distinguish between induction-recursion in the sense of the theory **IR**, and the more general concept of the simultaneous inductive definition of some sets $U_i$ and decodings $T_i$ defined recursively on the way the elements of the $U_i$ are generated.

**Is the Mahlo universe predicative?** The subuniverses $\mathrm{U}_f$ of the Mahlo universe M can be defined using **IR**. However the introduction rule, which adds a representative $\widehat{\mathrm{U}}_f$ to M, is problematic, because it depends on the *total* functions $f : \mathrm{Fam(M)} \to \mathrm{Fam(M)}$, and this can be considered impredicative. The second author has defined a model for the Mahlo universe [30] in an extension $\mathsf{KPM}^+$ of Kripke Platek set theory, but this model does not provide a predicative justification of its consistency.

An idea for constructing a predicative model came up in a discussion between the second author and Ulrich Berger. The idea is to first build a preliminary version of the universe $(\mathrm{U}_f, \mathrm{T}_f)$ for arbitrary *terms* $f$ including those that do not refer to total functions on families of types. Kahle and Setzer [20] formalized this idea in the setting of Feferman's explicit mathematics [14], a framework where you have access to the collection of all (untyped) terms. They called their construction the *extended predicative Mahlo universe* and we will present it in Section 5. It is an *internal* predicative Mahlo universe analogous to the internal Mahlo universe in type theory presented above. In Section 6, we will present an *external* version of the extended predicative Mahlo universe. It is this external version we will formalize in Martin-Löf type theory in Section 7. So we will go full circle: from a Mahlo universe in type theory to a Mahlo universe in explicit mathematics and then ending up with another Mahlo universe in type theory.

**Inductive vs recursive subuniverses.** Subuniverses (discussed above in connection with the Mahlo universe) can be formed in two ways: as inductive and as recursive subuniverses.

If $(\mathrm{U}_1, \mathrm{T}_1)$ is a recursive subuniverse of $(\mathrm{U}_2, \mathrm{T}_2)$, we define inductive-recursively the elements of $\mathrm{U}_1$ and an embedding $\widehat{\mathrm{T}}_1 : \mathrm{U}_1 \to \mathrm{U}_2$. Then $\mathrm{T}_1$ is defined as $\mathrm{T}_1\ a = \mathrm{T}_2\ (\widehat{\mathrm{T}}_1\ a)$. For instance, we have $\widehat{\mathbb{N}}_2 : \mathrm{U}_2$, $\mathrm{T}_2\ \widehat{\mathbb{N}}_2 = \mathbb{N}$, $\widehat{\mathbb{N}}_1 : \mathrm{U}_1$, $\widehat{\mathrm{T}}_1\ \widehat{\mathbb{N}}_1 = \widehat{\mathbb{N}}_2$, and have $\mathrm{T}_1\ \widehat{\mathbb{N}}_1 = \mathrm{T}_2\ \widehat{\mathbb{N}}_2 = \mathbb{N}$.

In an inductive subuniverse, we define a universe $(\mathrm{U}_1, \mathrm{T}_1)$ directly, and then a constructor $\widehat{\mathrm{T}}_1 : \mathrm{U}_1 \to \mathrm{U}_2$ with recursive equation $\mathrm{T}_2(\widehat{\mathrm{T}}_1\ a) = \mathrm{T}_1\ a$. The example of the natural numbers would result in the same rules for $\widehat{\mathbb{N}}_2$, but we define $\widehat{\mathbb{N}}_1 : \mathrm{U}_1$, $\mathrm{T}_1\ \widehat{\mathbb{N}}_1 = \mathbb{N}$. Note that $\mathrm{T}_2(\widehat{\mathrm{T}}_1\ \widehat{\mathbb{N}}_1) = \mathbb{N}$, therefore $\widehat{\mathrm{T}}_1\ \widehat{\mathbb{N}}_1$ is a second code for the set of natural numbers which is not equal to $\widehat{\mathbb{N}}_2$.

## 3 Explicit mathematics

Explicit mathematics is a theory introduced by Solomon Feferman for formalizing constructive mathematics. He considered both intuitionistic and classical logic. In its intuitionistic form explicit mathematics is an untyped alternative to Martin-Löf type theory. Explicit mathematics is presented as a second-order language, where first-order objects (individuals) can be considered as programs or terms. Second-order quantifiers range over sets which are collections of individuals given by a name. Here names are specific individuals $r$, for which we have a relation $s \dot{\in} r$ for $s$ is an element of $r$. In explicit mathematics, one usually uses 'type' for what we call a 'set'—we prefer to use 'set' in order to avoid confusion with the type-theoretic usage of the word type. Note that the word set has a different meaning in type theory, and we sometimes say 'type-theoretic sets' in order to distinguish them from the sets in explicit mathematics. The use of second-order quantifiers is a technical trick, and it might be possible to avoid it. Actually Tupailo [33] uses a two sorted language instead of a second-order language in his presentation of explicit mathematics.

We will give a brief introduction to explicit mathematics. A detailed presentation can for instance be found in Jäger and Strahm's article [19], from which a substantial part of the material in this section is taken. However we replace the classical axioms for comprehension by the intuitionistic elementary comprehension axioms in Tupailo's article [33].

Explicit mathematics is formulated in a second-order language. We let $U, V, W, X, Y, Z, \ldots$ denote second-order variables and $a, b, c, \ldots, f, x, y, z$ denote first-order variables. In order to improve readability, we will omit the outermost quantifiers in axioms. So all axioms are implicitly quantified over all free first and second-order variables (we will see later that first-order quantifier range over defined elements). Individuals in explicit mathematics are constructed by application from a collection of primitive combinators. We let $r, s, t$ denote terms, and we write $r\ s$ for $r$ applied to $s$ (in running text we usually add brackets for readability writing $(r\ s)$). As usual, parentheses are associated to the left for terms, so $(r\ s\ t)$ denotes $((r\ s)\ t)$. We have the standard combinators $\mathsf{s}$ and $\mathsf{k}$, pairing $\pi$ (with $(r,s) := \pi\ r\ s$), projections $\pi_0$, $\pi_1$, zero $0$, successor $\mathsf{s}_N$ (with $a' := \mathsf{s}_N\ a$), predecessor $\mathsf{p}_N$ for natural numbers and case distinction $\mathsf{d}_N$ on equality. In addition we have combinators for names for various sets: natural numbers $\mathsf{nat}$, identity relation $\mathsf{id}$, inverse image $\mathsf{inv}$, empty set $\dot{\emptyset}$, binary union, intersection, and implication $\dot{\cup}, \dot{\cap}, \dot{\to}$ written infix, intersection and union over sets of pairs $\dot{\bigcap}$, $\dot{\bigcup}$, join $\mathsf{j}$ (which corresponds to $\Sigma$ in type theory) and inductive generation $\mathsf{i}$ (accessible part of a relation). Furthermore, we shall later introduce specific combinators for the constructions associated with the Mahlo universe: for preuniverses $\mathsf{pu}$ and $\mathsf{pu}_{\mathsf{rel}}$, universes $\mathsf{u}$ and the Mahlo universe $\mathsf{M}$. Axioms explicitly refer only to a subset of these combinators.

The atomic formulas are $\bot$ (falsity), $s\downarrow$ for $s$ is defined, $\mathsf{N}(s)$ for $s$ is a natural number, $s = t$, $U = V$, $s \in U$, $\Re(s, U)$ for $s$ is a name for the second order variable $U$ and $\mathsf{PU}(x, a, f)$ for $x$ is an element of the preuniverse depending on $a, f$. $\mathsf{PU}(x, a, f)$ occurs only explicitly in the axioms of the external extended predicative universe.

One defines

- $s \simeq t := s\downarrow \vee t\downarrow \to s = t$.
- We use class notation similar to set theory as an abbreviation. Let $\vec{r} = r_1, \ldots, r_n$, $\vec{s} := s_1, \ldots, s_m$.
  - A predicate $P(\vec{r})$ is a formula with free object variables $\vec{r} = r_0, \ldots, r_n$ and no free second-order variables.
  - For predicates (i.e. relations) $P(r_0, \vec{r})$, we define $r_0 \in P(\vec{r}) := P(r_0, \vec{r})$, and call $P(\vec{r})$ a class. We write $\mathcal{A}(\vec{r}), \mathcal{B}(\vec{s})$ for classes. In case $n = 0$, we write $P$ instead of $P()$, and $\mathcal{A}$ instead of $\mathcal{A}()$.
  - $\{x \mid \varphi(\vec{r}, x)\}$ is the class $\mathcal{A}(\vec{r})$ with $x \in \mathcal{A}(\vec{r}) := \varphi(\vec{r}, x)$.
  - $\{x_1, \ldots, x_n\} := \{x \mid x = x_1 \vee \cdots \vee x = x_n\}$.
  - $\mathcal{A}(\vec{r}) \cup \mathcal{B}(\vec{r})$ is the class with $x \in (\mathcal{A}(\vec{r}) \cup \mathcal{B}(\vec{r})) := x \in \mathcal{A}(\vec{r}) \vee x \in \mathcal{B}(\vec{r})$, similarly for $\mathcal{A}(\vec{r}) \cap \mathcal{B}(\vec{r})$.
  - $\forall x \in \mathcal{A}(\vec{r}).\varphi(x) := \forall x.x \in \mathcal{A}(\vec{r}) \to \varphi(x)$, similarly for $\exists$.
  - $(f : \mathcal{A}(\vec{r}) \to \mathcal{B}(\vec{s})) := \forall x.x \in \mathcal{A}(\vec{r}) \to f\ x \in \mathcal{B}(\vec{s})$.
  - $\mathcal{A}(\vec{r}) \dot{\subseteq} \mathcal{B}(\vec{s}) := \forall x \in \mathcal{A}(\vec{r}).x \in \mathcal{B}(\vec{r})$
  - $\mathcal{A}(\vec{r}) \doteq \mathcal{B}(\vec{s}) := (\mathcal{A}(\vec{r}) \dot{\subseteq} \mathcal{B}(\vec{s}) \wedge \mathcal{B}(\vec{s}) \dot{\subseteq} \mathcal{A}(\vec{r}))$.

The underlying logic is the logic of partial terms $\mathsf{LPT}_{\mathsf{int}}$ (see [7], p. 97ff). It is based on intuitionistic logic extended to second-order logic. So $\mathsf{LPT}_{\mathsf{int}}$ is defined as follows (the following is taken from [7]):

- Standard intuitionistic propositional logic.
- Quantifier axioms:
  - From $B \to A$ one infers $B \to \forall x.A$ (provided $x$ not free in $B$)
  - From $A \to B$ one infers $(\exists x A) \to B$ (provided $x$ not free in $B$)
  - $(\forall x.A) \wedge t\downarrow \to A[x := t]$
  - $A[x := t] \wedge t\downarrow \to \exists x.A$
  - From $B \to A$ one infers $B \to \forall X.A$ (provided $X$ not free in $B$)
  - From $A \to B$ one infers $(\exists X.A) \to B$ (provided $X$ not free in $B$)
  - $(\forall X.A) \to A[X := U]$
  - $A[X := U] \to \exists X.A$
- Equality axioms
  - $x = x \wedge (x = y \to y = x)$
  - $(t \simeq s \wedge \phi(t)) \to \phi(s)$.
  - $t = s \to t\downarrow \wedge s\downarrow$
  - $X = X \wedge (X = Y \to Y = X)$
  - $(X = Y \wedge \phi(X)) \to \phi(Y)$.
- Definedness axioms
  - $R(t_1, \ldots, t_n) \to t_1\downarrow \wedge \cdots \wedge t_n\downarrow$ ($R$ atomic) as a consequence we have strictness $f\ t_1 \cdots t_n\downarrow \to f\downarrow \wedge t_1\downarrow \wedge \cdots \wedge t_n\downarrow$.
  - $c\downarrow$ for any constant $c$
  - $x\downarrow$ for any variable $x$.

The theory $\mathsf{COMB}_{\mathsf{int}}$ of combinatory algebra consists of $\mathsf{LPT}_{\mathsf{int}}$ and the applicative axioms

- $\mathsf{k}\ x\ y = x$
- $\mathsf{s}\ x\ y \downarrow \wedge\ \mathsf{s}\ x\ y\ z \simeq x\ y\ (x\ z)$.
- $\pi_0\ (x, y) = x \wedge \pi_1\ (x, y) = y$.

The theory $\mathsf{ECOMB}_{\mathsf{int}}$ of combinatory logic with explicit representation consist of $\mathsf{COMB}_{\mathsf{int}}$ of combinatory logic together with the axioms for explicit representation:

- $\forall X.\exists x.\Re(x, X)$.
- $\forall x, X, Y.\Re(x, X) \wedge \Re(x, Y) \to X = Y$,
- $(\forall x.x \in X \leftrightarrow x \in Y) \to X = Y$.

The theory $\mathsf{APP}_{\mathsf{int}}$ consists of $\mathsf{ECOMB}_{\mathsf{int}}$ and the axioms for the natural numbers $\mathsf{N}$:

- $0 \in \mathsf{N} \wedge (\mathsf{s}_N : \mathsf{N} \to \mathsf{N})$.
- $\forall x \in \mathsf{N}.\mathsf{s}_N\ x \neq 0 \wedge \mathsf{p}_N\ (\mathsf{s}_N\ x) = x$
- $\forall x \in \mathsf{N}.x \neq 0 \to \mathsf{p}_N\ x \in \mathsf{N} \wedge \mathsf{s}_N\ (\mathsf{p}_N\ x) = x$
- $x, y \in \mathsf{N} \wedge x = y \to \mathsf{d}_N\ u\ v\ x\ y = u$
- $x, y \in \mathsf{N} \wedge x \neq y \to \mathsf{d}_N\ u\ v\ x\ y = v$

In explicit mathematics, some terms are *names* that denote classes of terms. As mentioned before, we have a relation $\Re(x, U)$ meaning '$x$ is a name of the set $U$'. The formulation of explicit mathematics using names and the $\Re$ relation goes back to [15]. We follow the presentation in Kahle and Setzer [20] and define:

$$\begin{array}{lll} s \in \Re := \Re(s) & := & \exists X.\Re(s, X), \\ s \,\dot{\in}\, t & := & \exists X.\Re(t, X) \wedge s \in X, \\ \Re_\Re(s) & := & s \in \Re \wedge \forall x \,\dot{\in}\, s.x \in \Re, \\ (f : s \to s) & := & \forall x.x \,\dot{\in}\, s \to f\ x \,\dot{\in}\, s, \end{array}$$

- We identify elements $r$ of $\Re$ with the class $\{a \mid a \,\dot{\in}\, r\}$, and therefore we can use $r$ in places where a class is required. This would allow us to write $a \in r$, but in order to be in line with current practice in explicit mathematics, we continue using $a \,\dot{\in}\, r$.
- When defining formulas depending on $r \in \Re$, we lift them to formulas depending on a class $\mathcal{A}$ in an obvious way by replacing occurrences of $a \,\dot{\in}\, r$ by $a \in \mathcal{A}$.

The reader might observe the similarity between $\Re$ and a type-theoretic universe à la Tarski. $\Re$ is a class of codes, and $\Re(x, U)$ means that $U$ is the set denoted by $x$. Below we introduce axioms expressing closure of $\Re$ under the basic operations for forming sets (which are classes denoted by names) in explicit mathematics.

The following axioms for names come from Tupailo [34] and imply arithmetic comprehension, see [34] for details.

The theory $\mathsf{EET}_{\mathsf{int}}$ consists of $\mathsf{APP}_{\mathsf{int}}$ and the following axioms for elementary comprehension:

- $\dot{\emptyset} \in \Re \wedge \forall x.\neg(x \,\dot{\in}\, \dot{\emptyset})$.
- $\mathsf{id} \in \Re \wedge \mathsf{id} \doteq \{y \mid \exists x.y = (x, x)\}$.
- $\mathsf{nat} \in \Re \wedge \mathsf{nat} \doteq \mathsf{N}$, with the class $\mathsf{N}$ of natural numbers axiomatized by the axioms above.
- $a \in \Re \to \mathsf{inv}\ f\ a \in \Re \wedge \mathsf{inv}\ f\ a \doteq \{x \mid f\ x \,\dot{\in}\, a\}$.

- $a \in \Re \wedge b \in \Re \to a \dot{\cup} b \in \Re \wedge (a \dot{\cap} b) \in \Re \wedge (a \dot{\to} b) \in \Re$
  $\wedge\ (a \dot{\cup} b) \doteq \{x \mid x \dot{\in} a \vee x \dot{\in} b\}$
  $\wedge\ (a \dot{\cap} b) \doteq \{x \mid x \dot{\in} a \wedge x \dot{\in} b\}$
  $\wedge\ (a \dot{\to} b) \doteq \{x \mid x \dot{\in} a \to x \dot{\in} b\}$
- $a \in \Re \to \dot{\bigcap} a \in \Re \wedge \dot{\bigcup} a \in \Re$
  $\wedge\ (\dot{\bigcap} a) \doteq \{x \mid \forall y.(x,y) \dot{\in} a\}$
  $\wedge\ (\dot{\bigcup} a) \doteq \{x \mid \exists y.(x,y) \dot{\in} a\}$

The theory $\mathsf{EETJ}_{\mathsf{int}}$ extends $\mathsf{EET}_{\mathsf{int}}$ by the the axiom for the join which is the indexed disjoint union of sets. (We take the liberty to use functional notation ($\mathsf{j}\ a\ b$) as opposed to writing it as $\mathsf{j}(a,b)$ using pairing operation $(a,b)$ as in the papers by Jäger, Studer and Strahm—the same applies to $\mathsf{i}$ below.)

- $a \in \Re \wedge (f : a \to \Re) \to \mathsf{j}\ a\ f \in \Re \wedge \mathsf{j}\ a\ f \doteq \{(x,y) \mid x \dot{\in} a \wedge y \dot{\in} f\ x\}$.

The join corresponds to the $\Sigma$-type in Martin-Löf type theory, ($\mathsf{j}\ a\ f$) could be written as $\Sigma x \dot{\in} a.f\ x$ in Martin-Löf notation.

There are two forms of induction for $\mathsf{N}$, set induction and formula induction.[3] In explicit mathematics, one considers both theories with set induction only and with formula induction only, resulting in different proof theoretic strength.

- Set induction is formalised as a single second-order quantified axiom ($\mathsf{T}$-$\mathsf{I}_{\mathsf{N}}$). Since second-order variables range over sets which are classes denoted by names, set induction is induction over sets:

$$\forall X.0 \in X \wedge (\forall x \in \mathsf{N}.x \in X \to \mathsf{s}_N\ x \in X) \to \mathsf{N} \dot{\subseteq} X.$$

- Formula induction ($\mathbb{L}$-$\mathsf{I}_{\mathsf{N}}$) is induction over classes. We could as well call it class induction, but formula induction is well-established terminology in proof theory. So formula induction has for any class $\mathcal{A}$ an axiom:

$$0 \in \mathcal{A} \wedge (\forall x \in \mathsf{N}.x \in \mathcal{A} \to \mathsf{s}_N\ x \in \mathcal{A}) \to \mathsf{N} \dot{\subseteq} \mathcal{A}.$$

The principle of inductive generation $\mathsf{IG}$ defines the accessible part ($\mathsf{i}\ a\ b$) of a relation $b$ with domain $a$ and plays a similar role as the W-type in Martin-Löf type theory. We follow here the formulation in [19]. We define

$$\mathsf{Closed}(a,b,\mathcal{A}) := \forall x \dot{\in} a.(\forall y \dot{\in} a.(y,x) \dot{\in} b \to y \in \mathcal{A}) \to x \in \mathcal{A}.$$

The axioms of inductive generation are defined by

- $a, b \in \Re \to \mathsf{i}\ a\ b \in \Re \wedge \mathsf{Closed}(a,b,\mathsf{i}\ a\ b))$
- $a, b \in \Re \wedge \mathsf{Closed}(a,b,\mathcal{A}) \to \mathsf{i}\ a\ b \dot{\subseteq} \mathcal{A}$.

We define $\mathsf{T}_0^{\mathsf{int}} := \mathsf{EETJ}_{\mathsf{int}} + (\mathsf{IG}) + (\mathbb{L}\text{-}\mathsf{I}_{\mathsf{N}})$, which forms the base theory for the various theories of Mahlo universes discussed in the following.

[3] Set induction is in explicit mathematics usually called type-induction, but we avoid the use of type because of its usage in type theory. Instead we call types sets.

For proof theoretic studies, where one wants to determine the strength of various variations, one can consider

- omitting (IG) which results in meta-predicative theories;
- omitting (IG) and replacing ($\mathbb{L}$-$\mathsf{I_N}$) by ($\mathsf{T}$-$\mathsf{I_N}$).

Here, meta-predicativity is a notion introduced by Jäger for theories which are impredicative in the sense of having strength bigger than the Schütte–Feferman ordinal $\Gamma_0$, but which can be analysed proof theoretically using predicative proof theoretic methods. See the papers by Jäger, Strahm, Studer, Tupailo and others regarding proof theoretic investigations of variations of the above theories. Our formalisation in type theory corresponds to explicit mathematics containing $\mathsf{T}_0^{\mathsf{int}}$.

## 4 Axiomatic Mahlo in explicit mathematics

In this section, we present the axiomatic Mahlo universe as introduced and studied by Jäger, Strahm, Studer [16, 18, 19, 32] and Tupailo's external Mahlo universe [34]. We first introduce the notion of a universe in explicit mathematics, see [17] for a systematic study of universes in explicit mathematics by Jäger, Kahle and Studer.

Universes are names, the elements of which are names. Since sets are classes given by a name, we can say that universes are sets of sets. They are closed under the standard constructs for forming sets in explicit mathematics:

$$\begin{array}{lll} \Gamma_{\mathsf{univ}}(u) & := & \{\mathsf{nat}, \mathsf{id}\} \\ & & \cup \{r \dot{\cap} r', r \dot{\cup} r', r \dot{\to} r', \dot{\bigcap} r, \dot{\bigcup} r \mid r, r' \dot{\in} u\} \\ & & \cup \{\mathsf{inv}\ \ a\, f \mid a \dot{\in} u\} \\ & & \cup \{\mathsf{j}\ r\ s \mid r \dot{\in} u, \forall x \dot{\in} r.s\ x \dot{\in} u\} \\ \mathcal{U}(u) & := & u \in \Re \wedge u \dot{\subseteq} \Re \wedge \Gamma_{\mathsf{univ}}(u) \dot{\subseteq} u. \end{array}$$

**External axiomatic Mahlo in explicit mathematics.** In Section 1, we introduced Setzer's Mahlo universe in type theory, and we discussed both external and internal versions of this notion. A similar construction of an external Mahlo universe was introduced by Jäger, Strahm and Studer [18, 19] in the framework of explicit mathematics. We refer to this as the *axiomatic Mahlo universe*.

We make use of the combinator $\mathsf{u}$ where ($\mathsf{u}\ \ a\ f$) is a universe containing $a$ and closed under $f$ (where $a,f$ are terms). The axiom ($\mathsf{Ax\text{-}M^{ext}}$) for the external axiomatic Mahlo universe is as follows:

$$a \in \Re \wedge (f : \Re \to \Re) \to \mathcal{U}(\mathsf{u}\ a\, f) \wedge a \dot{\in} \mathsf{u}\ a\, f \wedge (f : \mathsf{u}\ a\, f \to \mathsf{u}\ a\, f).$$

Here $\Re$ plays the role of an external Mahlo universe with subuniverses ($\mathsf{u}\ \ a\ f$). This is analogous to the fact that in Martin-Löf type theory, Set plays the role of an external type-theoretic Mahlo universe, see the discussion in Section 1. We remark that we use the name $\mathsf{u}$ (rather than $\mathsf{m}$ used by [18, 19]) to emphasize the analogy between ($\mathsf{u}\ \ a\ f$) and the type-theoretic subuniverse $\mathrm{U}_f$. (An inessential difference is that the former but not the latter contains an element $a$).

**Inconsistency of elimination rules.** Palmgren [25] showed in Theorem 6.1. that adding elimination rules for the type-theoretic Mahlo universe results in an inconsistency. We transfer this result to the axiomatic Mahlo universe in explicit mathematics. All what is required to obtain an inconsistency is the existence of a function which extracts the function $f$ from ($\mathsf{u}\ a\, f$):

LEMMA 4.1
Assume $\mathsf{EETJ}_{\mathsf{int}} + (\mathsf{Ax\text{-}M}^{\mathsf{ext}})$. Then there exists no eliminator $\mathsf{elim} : \Re \to \Re \to \Re$ such that

$$\forall a, x \in \Re. \forall f : \Re \to \Re. \mathsf{elim}(\mathsf{u}\ a\ f)\ x = f\ x.$$

**Proof:** Assume the existence of $\mathsf{elim}$ as stated in the lemma. We choose $\mathsf{dum} := \mathsf{nat}$ as a dummy element of $\Re$. Define $\mathsf{emb} : (\Re \to \Re) \to \Re$, $\mathsf{emb}\ f = \mathsf{u}\ \mathsf{dum}\ f$. Then we get $\mathsf{elim}(\mathsf{emb}\ f)\ x = f\ x$.

We remind ourselves about the recursion operator in the context of the untyped lambda calculus: Assuming $f$, define $t := \lambda x.f\ (x\ x)$, $s := t\ t$, and we get $s = (\lambda x.f\ (x\ x))\ t = f\ (t\ t) = f\ s$. Translating this to our situation, we can define

$$
\begin{array}{lll}
t\_ & := & \lambda f, x.f\ ((\mathsf{elim}\ x)\ x) : (\Re \to \Re) \to \Re \to \Re \\
\mathsf{rec} & : & \lambda f.t_f\ (\mathsf{emb}\ t_f) : (\Re \to \Re) \to \Re \\
 & & \text{and we get} \\
\mathsf{rec}\ f & = & (\lambda x.f\ ((\mathsf{elim}\ x)\ x))\ (\mathsf{emb}\ t_f) = f\ ((\mathsf{elim}\ (\mathsf{emb}\ t_f))\ (\mathsf{emb}\ t_f)) \\
 & = & f\ (t_f\ (\mathsf{emb}\ t_f)) = f\ (\mathsf{rec}\ f) \\
 & & \text{Define} \\
f & := & \lambda x.x\ \dot{\to}\ \dot{\emptyset} : \Re \to \Re \\
a & := & \mathsf{rec}\ f \in \Re \\
 & & \text{and we obtain} \\
a & = & f\ a = a\ \dot{\to}\ \dot{\emptyset}
\end{array}
$$

So if $x \dot{\in} a$ then $x \dot{\in} a\ \dot{\to}\ \dot{\emptyset}$, $x\ \dot{\in} \dot{\emptyset}$, and we get $\bot$.
Therefore $\forall x.\neg(x \dot{\in} a)$, therefore $\mathsf{dum} \dot{\in} a\ \dot{\to}\ \dot{\emptyset} = a$, $\mathsf{dum} \dot{\in} \dot{\emptyset}$ and therefore $\bot$. □

**Internal axiomatic Mahlo in explicit mathematics.** Tupailo [34] introduced an internal axiomatic Mahlo universe $\mathsf{M}$ in explicit mathematics. The axioms $(\mathsf{Ax\text{-}M}^{\mathsf{int}})$ express that $\mathsf{M}$ is a universe containing the subuniverses $(\mathsf{u}\ a\ f)$ for any $a \dot{\in} M$ and $f : \mathsf{M} \to \mathsf{M}$. They are defined as follows:

$$
\begin{array}{ll}
(\mathcal{U}(\mathsf{M})) & \mathcal{U}(\mathsf{M}) \\
(\mathsf{Cl_u\text{-}Ax\text{-}}M^{\mathsf{int}}) & (a \dot{\in} \mathsf{M}\ \wedge\ (f : \mathsf{M} \to \mathsf{M})) \\
 & \to \mathcal{U}(\mathsf{u}\ a\ f) \wedge\ a \dot{\in} \mathsf{u}\ a\ f \wedge\ (f : \mathsf{u}\ a\ f \to \mathsf{u}\ a\ f) \\
 & \quad \wedge \mathsf{u}\ a\ f \dot{\subseteq} \mathsf{M} \wedge\ \mathsf{u}\ a\ f \dot{\in} \mathsf{M}
\end{array}
$$

(We remark that Tupailo only had closure under $f$ but not $a$. We add it here to be consistent with the formalisation by Jäger, Strahm and Studer).

**Remark on $f$ : M → M.** In type theory, the premise for the Mahlo universe construction is $f : \mathrm{Fam}(\mathsf{M}) \to \mathrm{Fam}(\mathsf{M})$. In explicit mathematics using coding and join, we can replace $f$ by $f' : \mathsf{M} \to \mathsf{M}$.

More precisely, for $u \in \Re_{\Re}$ define

$$\mathsf{Fam}_{EM}(u) := \mathsf{j}\ u\ (\lambda y.y\ \to\ u)$$

to be the set of families of sets in $u$ in explicit mathematics. Here, for $a, b \in R$ we define $a \to b := \{f \mid \forall z \dot{\in} a.f\ z \dot{\in} b\}$ by comprehension.

Define the relation $\cong$ of extensional equality on $\mathsf{Fam}_{EM}(u)$ by

$$(a,f) \cong (a',f') :\Leftrightarrow a \doteq a' \wedge \forall x \dot{\in} a.f\ x \doteq f'\ x.$$

LEMMA 4.2
If $u \in \mathcal{U}, f : \mathsf{Fam}_{EM}(u) \to \mathsf{Fam}_{EM}(u)$, then we can define $f' : u \to u$ so that closure under $f'$ implies closure of $f$ up to $\cong$, i.e. $v \in \mathcal{U}$, $v \dot{\subseteq} u$ and $f' : v \to v$ implies

$$\forall x \in \mathsf{Fam}_{EM}(v).\exists y, z \in \mathrm{Fam}(v).y \cong x \wedge f\ y \cong z$$

**Proof:** Let $a \times b := \mathsf{j}\ a\ (\lambda y.b)$ for some fresh variable $y$, and $\{r\}$ be defined by comprehension. Let $\mathsf{p}\ a\,f := (\{0\} \times a)\ \dot{\cup}\ (\{1\} \times \mathsf{j}\ a\,f)$.

If $u \in \mathcal{U}$, $a \dot{\in} u$ and $f : a \to u$, then $\mathsf{p}\ a\,f \dot{\in} u$. Let $\mathsf{q} : \mathsf{Fam}_{EM}(u) \to u$, $\mathsf{q}\ x = \mathsf{p}\ (\pi_0\ x)\ (\pi_1\ x)$.

Let $\mathsf{p}_0\ x := \{z \mid (0,z) \dot{\in} x\}$, and $\mathsf{p}_1\ x\ y := \{z \mid (1,(y,z)) \dot{\in} x\}$. If $x \dot{\in} u$, then $\mathsf{p}_0\ x \dot{\in} u$ and $\mathsf{p}_1\ x \dot{\in} \mathsf{p}_0\ x \to u$. Define $\mathsf{r} : u \to \mathsf{Fam}_{EM}(u)$ as $\mathsf{r}\ x = (\mathsf{p}_0\ x, \mathsf{p}_1\ x)$.

Furthermore, if $(a,g) \dot{\in} \mathsf{Fam}_{EM}(u)$, then $\mathsf{p}_0\ (\mathsf{p}\ a\ g) \doteq a$ and for $y \dot{\in} a$ we have $\mathsf{p}_1\ (\mathsf{p}\ a\ g)\ y \doteq g\ y$. So, $\forall a \dot{\in} \mathsf{Fam}_{EM}(u).\mathsf{r}\ (\mathsf{q}\ a) \cong a$.

If $f : \mathsf{Fam}_{EM}(u) \to \mathsf{Fam}_{EM}(u)$, then define $f' : u \to u$ as $f' = \mathsf{q} \circ f \circ \mathsf{r}$. Assume $v \in \mathcal{U}$, $v \dot{\subseteq} u$ is closed under $f'$. Then it contains for $x \dot{\in} \mathsf{Fam}_{EM}(v)$ the element $f'\ (\mathsf{q}\ x)$ and therefore we have $\mathsf{r}\ (f'\ (\mathsf{q}\ x)) \dot{\in} \mathsf{Fam}_{EM}(v)$. $\mathsf{r}\ (f'\ (\mathsf{q}\ x)) = \mathsf{r}\ (\mathsf{q}\ (f\ (\mathsf{r}\ (\mathsf{q}\ x))))$. Therefore, $\forall x \in \mathsf{Fam}_{EM}(u).\exists y, z \in \mathrm{Fam}(u).y \cong x \wedge f\ y \cong z$. □

We do not expect that closure only up to $\cong$ makes any difference in a proof theoretic analysis: in well-ordering proofs, one works with families of sets of ordinal notations, so closure under a function which is equivalent up to $\cong$ only replaces a set of ordinal notations by an extensionally equal one. Upper bounds formed by models are even easier, when working with $f : \mathsf{M} \to \mathsf{M}$ instead of $f : \mathsf{Fam}_{EM}(\mathsf{M}) \to \mathsf{Fam}_{EM}(\mathsf{M})$. When looking at the literature, it seems unlikely that switching from $f : \mathsf{M} \to \mathsf{M}$ to $f : \mathsf{Fam}_{EM}(\mathsf{M}) \to \mathsf{Fam}_{EM}(\mathsf{M})$ would make any difference regarding the proof theoretic strength.

We note that the replacement of Fam(M) by M cannot be carried out directly in Martin-Löf type theory, since $\mathsf{p}_0$, $\mathsf{p}_1$ cannot be defined in a total setting. One might try to use elimination rules for the Mahlo universe, but as discussed before, we know general elimination rules for the standard Mahlo universe are inconsistent (Palmgren [25], Theorem 6.1). One could think of adding specific elimination rules which allow to define a function M → Fam(M), which map a code $(\widehat{\Sigma}\ a\ b)$ for a $\Sigma$-type to a family of sets $(\mathrm{p}\ a\ b)$ and all other elements to a default family of sets in M. But that would go against the spirit of Martin-Löf type theory where one aims for general principles and avoids ad hoc rules.

## 5 Internal extended predicative Mahlo

In order to form the subuniverses $(\mathsf{u}\ a\,f)$ of the internal axiomatic Mahlo universe, we do not really need that $f$ is total on $\mathsf{M}$, only on $(\mathsf{u}\ a\,f)$. A first idea for defining a predicative version of the Mahlo universe is to require that only $(\mathsf{u}\ a\ f)$ is closed under $f$. The problem is that we do not know yet what $(\mathsf{u}\ a\,f)$ is. The solution is to first define the *least preuniverse* $\mathsf{pu}\ a\,f : \mathsf{M}$ for arbitrary terms $a$ and $f$. This is a universe containing $a$ and closed under $f$, provided the elements created are in $\mathsf{M}$.[4] If $(\mathsf{pu}\ a\,f)$ is closed under $a$ and $f$, then it is independent of $\mathsf{M}$, in the sense that we could drop the requirement that any element added to $(\mathsf{pu}\ a\,f)$ needs to be in $\mathsf{M}$, since it is always fulfilled. We can also say that if $(\mathsf{pu}\ a\,f)$ is independent of $\mathsf{M}$, then $(\mathsf{pu}\ a\,f)$ is complete and would not change if we add more elements to $\mathsf{M}$.

[4]More generally, Kahle and Setzer form $\mathsf{pu}$ with an extra argument $v$ for any universe $v$, which we call $(\mathsf{pu}_{\mathsf{rel}}\ a\ f\ v)$, where $\mathsf{pu}\ a\,f = \mathsf{pu}_{\mathsf{rel}}\ a\,f\ \mathsf{M}$. This is not needed until we discuss the elimination rule for $\mathsf{M}$.

The introduction axiom for $(\mathsf{u}\ a\ f)$ as elements of $\mathsf{M}$ for the internal extended predicative Mahlo universe states that if $(\mathsf{pu}\ a\ f)$ is independent, then we form $(\mathsf{u}\ a\ f)$ as a subuniverse of $\mathsf{M}$ closed under $a$ and $f$ and add it to $\mathsf{M}$.

We will now introduce the axioms $(\mathsf{Pred\text{-}M}^{\mathsf{int}})$ for the extended predicative Mahlo universe $\mathsf{M}$ with subuniverses $(\mathsf{u}\ a\ f)$ by the following 3 groups of axioms:

- the axiom $(\mathcal{U}(\mathsf{M}))$, stating that $\mathsf{M}$ is a universe;
- axioms $(\mathsf{pu})$ stating that $(\mathsf{pu}\ a\ f)$ is the least preuniverse closed under $a$ and $f$;
- an axiom $(\mathsf{intro\text{-}u})$ stating that we can form $(\mathsf{u}\ a\ f) \doteq (\mathsf{pu}\ u\ f)$ as an element of $\mathsf{M}$ provided the independence assumption is satisfied.

In the following, we formulate these groups of axioms which will require some auxiliary definitions:

**Axiom $(\mathcal{U}(\mathsf{M}))$.** This axiom is so short that its name is the same as the axiom itself, namely

$$\mathcal{U}(\mathsf{M}).$$

**Preuniverses.** In order to define $(\mathsf{pu}\ a\ f)$, we define a map $u \mapsto \Gamma_{\mathsf{pu}}(a,f,u)$ that takes one step in the iteration creating $(\mathsf{pu}\ a\ f)$. In the next definition, the first line defines the class of potential elements of the preuniverse, and the second line expresses that only elements in $\mathsf{M}$ are added

$$\begin{array}{lcl} \Gamma^{\mathsf{pot}}_{\mathsf{pu}}(a,f,u) & := & \Gamma_{\mathsf{univ}}(u) \cup \{a\} \cup \{f\ y \mid y \mathrel{\dot\in} u\} \\ \Gamma_{\mathsf{pu}}(a,f,u) & := & \Gamma^{\mathsf{pot}}_{\mathsf{pu}}(a,f,u) \cap \mathsf{M}. \end{array}$$

The closure property for $u$ being a preuniverse closed under $a$ and $f$ is

$$\mathcal{C}l_{\mathsf{pu}}(a,f,u) \quad := \quad \forall x \in \Gamma_{\mathsf{pu}}(a,f,u).x \mathrel{\dot\in} u$$

or

$$\mathcal{C}l_{\mathsf{pu}}(a,f,u) \quad := \quad \Gamma_{\mathsf{pu}}(a,f,u) \mathrel{\dot\subseteq} u.$$

The introduction rule of $(\mathsf{pu}\ a\ f)$ states that it satisfies the closure property and the elimination rule states that it is the least one satisfying the closure property. The axioms $(\mathsf{pu})$ are defined as follows:

$$(\mathsf{pu}) \begin{cases} \mathcal{C}l_{\mathsf{pu}}(a,f,\mathsf{pu}\ a\ f) \\ \mathcal{C}l_{\mathsf{pu}}(a,f,\mathcal{A}) \to \mathsf{pu}\ a\ f \mathrel{\dot\subseteq} \mathcal{A} \qquad \text{(for any class } \mathcal{A}\text{).} \end{cases}$$

In order to obtain a metapredicative version, we introduce a restricted version of these axioms

$$(\mathsf{pu}_{res})\ \ \mathcal{C}l_{\mathsf{pu}}(a,f,\mathsf{pu}\ a\ f) \wedge \mathsf{pu}\ a\ f \mathrel{\dot\subseteq} \mathsf{M}.$$

**Independence of $\mathsf{M}$.** If all potential elements in $\Gamma^{\mathsf{pot}}_{\mathsf{pu}}(a,f,u)$ are already in $\mathsf{M}$, then we say that $u$ is *independent* of $\mathsf{M}$, where independence is defined as follows:

$$\mathrm{Indep}(a,f,u) \quad := \quad \Gamma^{\mathsf{pot}}_{\mathsf{pu}}(a,f,u) \mathrel{\dot\subseteq} \mathsf{M}.$$

Once we have $\mathrm{Indep}(a,f,\mathsf{pu}\ a\ f)$, we know that $(\mathsf{pu}\ a\ f)$ does not depend on future elements of $\mathsf{M}$. One can easily derive that $(\mathsf{pu})$ implies the following:

$$\mathrm{Indep}(a,f,\mathsf{pu}\ a\ f) \to \Gamma^{\mathsf{pot}}_{\mathsf{pu}}(a,f,\mathsf{pu}\ a\ f) \mathrel{\dot\subseteq} \mathsf{pu}\ a\ f.$$

The axiom (intro-u), stating that we can form $(\mathsf{u}\ a\ f) \doteq (\mathsf{pu}\ u\ f)$ provided the independence assumption is satisfied, is defined as follows:

$$(\text{intro-u}) \quad \forall a,f.\mathrm{Indep}(a,f,\mathsf{pu}\ a\,f) \to \mathsf{u}\ a\,f \mathrel{\dot\in} \mathsf{M} \wedge \ \mathsf{u}\ a\,f \doteq \mathsf{pu}\ a\,f \ .$$

**The extended predicative Mahlo universe.** The axioms ($\text{Pred-M}^{\text{int}}$) of the extended predicative Mahlo universe are now given as

$$(\text{Pred-M}^{\text{int}}) = (\mathcal{U}(\mathsf{M})) + (\mathsf{pu}) + (\text{intro-u}).$$

This states that $\mathsf{M}$ is a universe, and that if $(\mathsf{pu}\ a\,f)$ is independent of $\mathsf{M}$, then $(\mathsf{u}\ a\,f)$ is an element of $\mathsf{M}$ which has the same elements as $\mathsf{pu}\ a\,f$.

We define as well a restricted form which omits induction over $(\mathsf{pu}\ a\,f)$.

$$(\text{Pred-M}^{\text{int}}_{\text{res}}) = (\mathcal{U}(\mathsf{M})) + (\mathsf{pu}_{res}) + (\text{intro-u})$$

LEMMA 5.1

(a) $(\mathcal{U}(\mathsf{M})) + (\mathsf{pu})$ implies $(\mathsf{pu}_{res})$ and therefore ($\text{Pred-M}^{\text{int}}$) implies ($\text{Pred-}M^{\text{int}}_{\text{res}}$).

(b) ($\text{Pred-M}^{\text{int}}_{\text{res}}$) proves the axioms of the internal axiomatic Mahlo universe

**Proof:** (a) $\mathsf{pu}\ \ a\ \ f\ \dot\subseteq\ \mathsf{M}$ follows by induction over $\mathsf{pu}$ using $\mathcal{A} := \mathsf{M}$. (b) We need to prove ($\mathsf{Cl}_u\text{-Ax-M}^{\text{int}}$). Assume $a\ \dot\in\ \mathsf{M}$ and $f : \mathsf{M} \to \mathsf{M}$. By $\mathcal{U}(\mathsf{M})$ and $\mathsf{pu}\ \ a\ \ f\ \dot\subseteq\ \mathsf{M}$, we get $\mathrm{Indep}(a,f,\mathsf{pu}\ a\,f)$, $\mathsf{u}\ a\,f \mathrel{\dot\in} \mathsf{M}$, $\mathsf{u}\ a\,f \doteq \mathsf{pu}\ a\,f$, $\mathcal{U}(\mathsf{pu}\ a\,f)$, $a \mathrel{\dot\in} \mathsf{pu}\ a\,f$, $f : \mathsf{pu}\ a\,f \to \mathsf{pu}\ a\,f$, which imply the conclusion.

**Remark.** The restricted extended predicative Mahlo universe ($\text{Pred-M}^{\text{int}}_{\text{res}}$) is likely to be meta-predicative, see the discusion in Section 3.

Whether induction over $(\mathsf{pu}\ a\,f)$ lifts the strength beyond meta-predicativity needs to be seen. If we add closure under inductive generation, even if induction over $(\mathsf{pu}\ a\,f)$ is omitted, we can embed the axiomatic Mahlo universe closed under inductive generation into it. Therefore, subject to the 'plausible result' in [35] that $|\mathbf{CZFM}^+| = |\mathbf{KPM}^+|$, we get, by the results of that article a theory of same strength as Martin-Löf type theory with W-type and one Mahlo universe, which goes beyond metapredicativity.

**The least Mahlo universe.** In order to state the elimination rule for $\mathsf{M}$, we need to first define a name $(\mathsf{pu}_{\mathsf{rel}}\ \ a\ \ f\ \ v)$, the elements of which are given in the same way as those of $(\mathsf{pu}\ \ a\ \ f)$ but referring to an arbitrary set of sets $v$ instead of $\mathsf{M}$. We need to relativize the notions involved in the definition of $(\mathsf{pu}\ a\,f)$ to using $v$ instead of $\mathsf{M}$ as well. Assume $v \in \mathfrak{R}_{\mathfrak{R}}$. We define

$$\begin{array}{lcl} \Gamma^{\mathsf{rel}}_{\mathsf{pu}}(a,f,u,v) & := & \Gamma^{\mathsf{pot}}_{\mathsf{pu}}(a,f,u) \cap v \\ \mathcal{C}l^{\mathsf{rel}}_{\mathsf{pu}}(a,f,u,v) & := & \Gamma^{\mathsf{rel}}_{\mathsf{pu}}(a,f,u,v) \mathrel{\dot\subseteq} u \\ \mathrm{Indep}^{\mathsf{rel}}(a,f,u,v) & := & \Gamma^{\mathsf{pot}}_{\mathsf{pu}}(a,f,u) \mathrel{\dot\subseteq} v \end{array}$$

and state the axioms (we lift $\mathcal{C}l^{\mathsf{rel}}_{\mathsf{pu}}$ to having a class as argument instead of a term in a straightforward way)

$$\begin{array}{l} v \in \mathfrak{R} \to \mathcal{C}l^{\mathsf{rel}}_{\mathsf{pu}}(a,f,\mathsf{pu}_{\mathsf{rel}}\ a\ f\ v,v) \\ v \in \mathfrak{R} \wedge \mathcal{C}l^{\mathsf{rel}}_{\mathsf{pu}}(a,f,\mathcal{A},v) \to \mathsf{pu}_{\mathsf{rel}}\ a\ f\ v \subseteq \mathcal{A}. \end{array}$$

The induction axiom (IndM) expressing that $\mathsf{M}$ is the least Mahlo universe is as follows:

$$\mathcal{U}(v)\ \wedge (\forall f,a.\mathrm{Indep}^{\mathsf{rel}}(a,f,\mathsf{pu}_{\mathsf{rel}}\ a\ f\ v,v) \to \mathsf{u}\ a\,f \mathrel{\dot\in} v) \to \mathsf{M} \mathrel{\dot\subseteq} v \ .$$

Note that we obtain an induction axiom for the Mahlo universe, which is in contrast with the axiomatic Mahlo universe, where we have seen in Lemma 4.1 that induction over the Mahlo universe is inconsistent.

## 6 External extended predicative Mahlo

We shall now consider an external version of the extended predicative Mahlo universe. This version is slightly simpler to model in Martin-Löf type theory than the internal version, and we also find it more illuminating. The reason is that when modelling explicit mathematics in type theory, the class of names $\Re$ becomes a type-theoretic universe in a general sense. An external Mahlo universe in explicit mathematics thus becomes an internal Mahlo universe in type theory, while an internal Mahlo universe in explicit mathematics becomes a Mahlo universe in type theory that is both contained in and an element of another universe.

In the external version, the class of names $\Re$ is a Mahlo universe. A preuniverse can no longer be defined as a name, since it is not an element of the Mahlo universe unless it is independent of it. Instead it is defined as a class $\mathsf{PU}(a,f)$ depending on terms $a,f$.

Let $\mathcal{A}$ be a class. We can then define $\Gamma^{\mathsf{pot}}_{\mathsf{pu}}(a,f,\mathcal{A})$ in the same way as we defined $\Gamma^{\mathsf{pot}}_{\mathsf{pu}}(a,f,u)$ in the previous section, except that we replace $y \dot{\in} u$ by $y \in \mathcal{A}$. We then define

$$\begin{array}{lcl} \Gamma^{\Re}_{\mathsf{pu}}(a,f,\mathcal{A}) & := & \Gamma^{\mathsf{pot}}_{\mathsf{pu}}(a,f,\mathcal{A}) \cap \Re \\ \mathcal{C}l^{\Re}_{\mathsf{pu}}(a,f,\mathcal{A}) & := & \Gamma^{\Re}_{\mathsf{pu}}(a,f,\mathcal{A}) \dot{\subseteq} \mathcal{A}. \end{array}$$

The axioms for $\mathsf{PU}$ are as follows:

$$(\mathsf{PU}) \begin{cases} \mathsf{PU}(a,f) \dot{\subseteq} \Re \\ \mathcal{C}l^{\Re}_{\mathsf{pu}}(a,f,\mathsf{PU}(a,f)) \\ \mathcal{C}l^{\Re}_{\mathsf{pu}}(a,f,\mathcal{A}) \rightarrow \mathsf{PU}(a,f) \dot{\subseteq} \mathcal{A}. \end{cases}$$

If $a$ is a name and $f$ maps elements of $\mathsf{PU}(a,f)$ to names, then we say that $(a,f)$ is independent of $\Re$

$$\mathrm{Indep}_{\Re}(a,f) := a \in \Re \wedge (f : \mathsf{PU}(a,f) \rightarrow \Re).$$

The following axiom expresses the closure condition for the external extended predicative Mahlo universe:

$$(\mathrm{Cl}_{\mathsf{PU}}) \quad \mathrm{Indep}_{\Re}(a,f) \rightarrow \mathsf{u}\; a\, f \in \Re \wedge \mathsf{u}\; a\, f \doteq \mathsf{PU}(a,f).$$

By $(\mathsf{PU})$ we have that $f : \mathsf{PU}(a,f) \rightarrow \mathsf{PU}(a,f)$ is equivalent to $f : \mathsf{PU}(a,f) \rightarrow \Re$, therefore $(\mathrm{Cl}_{\mathsf{PU}})$ is, assuming $(\mathsf{PU})$, equivalent to

$$a \in \mathsf{PU}(a,f) \wedge (f : \mathsf{PU}(a,f) \rightarrow \mathsf{PU}(a,f)) \rightarrow \mathsf{u}\; a\, f \in \Re \wedge \mathsf{u}\; a\, f \doteq \mathsf{PU}(a,f).$$

The axioms for the external extended predicative Mahlo universe are defined as

$$(\mathsf{Pred\text{-}M}^{\mathsf{ext}}) = (\mathsf{PU}) + (\mathrm{Cl}_{\mathsf{PU}}).$$

## 7 External extended predicative Mahlo in type theory

We shall now formalize explicit mathematics with the external predicative Mahlo universe in type theory with inductive-recursive definitions. The formalization is implemented in the proof assistant Agda [2]. There are many short introductions to Agda, e.g. in the papers coauthored by the second author. A recommended introduction is Section 2 in [1].

Inductive-recursive definitions can be seen as an integral part of Martin-Löf type theory. The authors have formalized several extensions of Martin-Löf type theory with inductive-recursive definitions [9, 11, 12]. The second author has also proposed inductive-recursive definitions that go beyond these extensions, e.g. of an *autonomous Mahlo universe* [31]. Similarly, to model the extended predicative Mahlo universe, we will also need to go beyond the authors' theories **IR**, **IIRD**, etc.

First, we formalize the basic ingredients of explicit mathematics. Full details can be found in Appendix A. We define the (type-theoretic) set of combinators (containing nat, zero, succ, id, . . .) and the set of terms with application written $r \cdot s$.

Here, when we say that $A$ is a set in type theory, we mean that the judgement $A$ : Set holds. When we say that $P$ is a predicate in type theory, we mean that the judgement $P : A \to$ Set holds. Moreover, we just say 'predicate' if it is clear from the context that it is a type-theoretic predicate. Classes in explicit mathematics will correspond to type-theoretic predicates.

We define $r \Rightarrow s$ meaning that $r$ reduces to the normal form $s$ and (NF $r$) meaning that $r$ is in normal form. That $r$ is defined, i.e. $r\downarrow$, is formalized as (Normalize $r$) which means that $r$ reduces to a term in normal form. $r \simeq s$ means that $r\downarrow$ if and only if $s\downarrow$ and if $r\downarrow$ and $s\downarrow$ then their their normal forms are the same.

Moreover, we define type-theoretic predicates on terms for all the basic constructions in explicit mathematics: the natural numbers, the identity proofs, union, intersection, etc. We define their sets of normal forms and define the whole set as those whose normal forms are in the specified sets. When quantifying over elements of a set we quantify over the set of normal forms. This implies that the quantifier holds as well for the normal forms of any normalizing term.

We do not present the Agda code for these basic notions here but refer the reader to Appendix A.

The type-theoretic definition of the external Mahlo universe follows the definition in explicit mathematics in Section 6 closely. The universe $\Re$ of names and the preuniverses $\mathsf{PU}(a,f)$ are implemented as indexed inductive-recursive definitions mutually dependent on each other. In order to get unique representations, we define only elements in normal form. We only present the definition of the names in normal form; the collection of all terms in $\Re$ can then be defined as those with a normal form in $\Re$.

**The Agda implementation of the universe of names.** The indexed inductive-recursive definition of $\Re$ is given by a type-theoretic predicate ∈ℜnf on terms together with decoding ∈nf·. (In Agda, we cannot write $\dot\in$, therefore we put the dot after the symbol.) Here $s$ ∈ℜnf means that $s$ is a name in normal form (of a set), and $r$ ∈nf· $s$ means that $r$ is a term in normal form that is an element of the extension of $s$. Note that this definition is mutually dependent on the implementation of the subuniverses $\mathsf{PU}(a,f)$, given as type-theoretic predicates $s$∈PUnf[$a$ , $f$] to be defined below.

The Agda code begins with the key word data [3], which specifies that ∈ℜnf is inductively defined. The rest of the line specifies that ∈ℜnf is a postfix operation and that it is a type-theoretic predicate. The subsequent lines specify four of the name constructors with their types expressing closure under the corresponding set operations in explicit mathematics. (We omit five more constructors: id, ∅ which have definitions similar to nat; ∪, → which have definitions similar to that of ∩; and ⋂ which has a definition similar to ⋃.)

Note that in Agda _ denote positions of arguments of mixfix operations [5]. So we can write $r$ ∈ℜnf for (_∈ℜnf $r$) and ($rp$ ∩ℜ $sp$) for (_∩ℜ_ $rp$ $sp$). Furthermore, arguments of the form $\{x : A\}$ are implicit or hidden arguments in Agda [4]. They are needed in the type signature of a function definition but omitted when applying the function to its arguments if they can be automatically inferred by the type checker. In case an implicit argument $r$ is needed, it can be stated explicitly using the notation $\{r\}$.

```
data _∈ℜnf : Term → Set where
  natℜ   : nat ∈ℜnf
  invℜ   : (f : Term){r : Term} → NF f → r ∈ℜnf → inv f r ∈ℜnf
  _∩ℜ_ : {r r' : Term} → r ∈ℜnf → r' ∈ℜnf → (r ∩· r') ∈ℜnf
  ⋂ℜ     : {r : Term} → r ∈ℜnf → ⋂· r ∈ℜnf
```

We also have a constructor for join with six arguments. The first four express that $r$ is a name and $s$ is a term, both in normal form. The remaining two express that there is a function *sn* that maps an element $x$ in $r$ to a normal form $p$ of $(s \cdot x)$ and a function $sx{\in}R$ that expresses that p is a normal name. (Since normal forms are unique it would be sufficient to say that for the normal form given by *sn*, we have that it is ∈ℜnf. However, that would require some equality reasoning in the rules, and we want to avoid them so that in the meaning explanations the reasoning steps are kept to a minimum.)

```
jℜ : {r : Term}(rp : r ∈ℜnf)(s : Term)(nfs : NF s)
      (sn : (x : Term)(xrp : x ∈nf· rp) → Normalize (s · x))
      (sx∈R : (x : Term)(xrp : x ∈nf· rp)(p : Normalize (s · x))
                → nf p ∈ℜnf)
      → jc r s ∈ℜnf
```

The constructor igℜ is for inductive generation:

```
igℜ : {r s : Term} (rp : r ∈ℜnf)(sp : s ∈ℜnf) → igc r s ∈ℜnf
```

It makes the system impredicative in the proof-theoretic sense. If we remove it, the system is meta-predicative in the sense of Jäger.

The final constructor specifies closure under subuniverses (u $a\,f$). It has six arguments. The first four express that $a$ is a name and $f$ is a term, both in normal form. The last two express that $f$ maps a term $x$ in the preuniverse ∈PUnf[$a$ , $f$] to a normal name $p$ of $f \cdot x$. (As before by uniqueness the normal form is unique but we wanted to keep the reasoning required to a minimum.) These and the third arguments express the *independence condition* in the axiom for introduction of the name u $a\,f \in \Re$ in explicit mathematics.

```
uℜ : (a f : Term)(aR : a ∈ℜnf)(fnf : NF f)
      (fnor : (x : Term)(xpu : (x ∈PUnf[ a , f ]))
               → Normalize (f · x))
      (fxR : (x : Term) (xpu : (x ∈PUnf[ a , f ]))
               (p : Normalize (f · x)) → nf p ∈ℜnf)
      → u a f ∈ℜnf
```

We now give the decoding equations for names, the recursive part of the indexed inductive-recursive definition. Note that the type of the constructor for join refers to ∈ℜnf, hence the simultaneous inductive-recursive nature of the definition.

We first give the six equations expressing closure under the basic set constructions of explicit mathematics (we omit five more equations for id, ∅, ∪, →, ⋂):

```
_∈nf·_ : (r : Term){s : Term} → s ∈ℜnf → Set
r ∈nf· natℜ              = r ∈Nat
r ∈nf· (sℜ ∩ℜ s'ℜ)       = (r ∈nf· sℜ ) ∧ (r ∈nf· s'ℜ )
r ∈nf· ⋂ℜ p              = (y : Term) → < r , y > ∈nf· p
z ∈nf· jℜ rp s nfs n sx∈R
  = z ∈J[ (λ t → t ∈nf· rp ) ,
          (λ t trp s' → s' ∈nf· sx∈R t trp (n t trp)) ]
r ∈nf· invℜ f fn s∈ℜnf = ( f · r) ∈nf· s∈ℜnf
r ∈nf· igℜ rp sp = Ig (λ t → t ∈nf· rp ) (λ t → t ∈nf· sp) r
```

Finally, we give the equation for the final constructor, the one for the name of the subuniverse u *a f* ∈ ℜ:

```
r ∈nf· uℜ a f aR fnf fnor fxR = r ∈PUnf[ a , f ]
```

Note that this equation refers to the preuniverse ∈PUnf[*a* , *f*], which will be defined next.

**Preuniverses.** The preuniverses PU(*a*,*f*) in explicit mathematics are implemented as an indexed inductive-recursive definition of predicates ∈PUnf[*a* , *f*] on terms (where *a*,*f* are terms in normal form) together with decodings PUnf2ℜnf. Here *s* ∈PUnf[*a* , *f*] means that *s* is an element in normal form of the preuniverse for *a*,*f*. The decoding PUnf2ℜnf maps a proof that *s* ∈PUnf[*a* , *f*] to a proof that *s* ∈ℜnf, i.e. it represents the injection PU(*a*,*f*) ⊆ ℜ in explicit mathematics. So in the terminology of Section 2 PU(*a*,*f*) is a recursive subuniverses of ℜ (with the generalization that both ℜ and PU(*a*,*f*) are type-theoretic predicates on terms rather than simple type-theoretic sets, so they are universes indexed over the set of terms).

The first line of the Agda implementation specifies that ∈PUnf[*a* , *f*] is a postfix inductive predicate on terms. The second specifies that it contains the normal term *a*, provided it is in ℜ. The third line specifies that it contains the normal form *s* of *f* ·*x* for any term *x* in the preuniverse, provided *s* is in ℜ

```
data _∈PUnf[_,_] : (r a f : Term) → Set where
  aproof : {a f : Term}(anf : a ∈ℜnf) → a ∈PUnf[ a , f ]
  fproof : {a f x : Term}(xpu : x ∈PUnf[ a , f ])
           (s : Term)(fxred : (f · x) -»NF s)(sℜnf : s ∈ℜnf)
           → s ∈PUnf[ a , f ]
```

Note that the types of these two constructors refer to ∈ℜnf, the inductive part of the previous indexed inductive-recursive definition.

The type-theoretic predicate $\in$PUnf$[a\,,f]$ has six more constructors (we omit the additional five for id, Ø, ∩, →, ⋂), one for each closure condition of a universe in explicit mathematics

```
natpu : {a f : Term} → nat ∈PUnf[ a , f ]
invpu : {a f : Term} (g : Term){r : Term}(nfg : NF g)
        (rpu : r ∈PUnf[ a , f ])
        → inv g r ∈PUnf[ a , f ]
_∩pu_ : {a f r r' : Term}(rpu : r ∈PUnf[ a , f ])
         (r'pu : r' ∈PUnf[ a , f ])
         → (r ∩· r') ∈PUnf[ a , f ]
⋂pu   :  {a f r : Term}(rpu : r ∈PUnf[ a , f ])
         → ⋂· r ∈PUnf[ a , f ]
jpu   :  {a f r : Term}(rp : r ∈PUnf[ a , f ])(s : Term)(nfs : NF s)
         (sn : (x : Term)(xrp : x ∈nf· PUnf2ℜnf rp)
                 → Normalize (s · x))
         (sxPU : (x : Term) (xrp : x ∈nf· PUnf2ℜnf rp)
                 (p : Normalize (s · x))
                 → nf p ∈PUnf[ a , f ])
         → jc r s ∈PUnf[ a , f ]
igpu  :  {a f r s : Term}(rp : r ∈PUnf[ a , f ])
         (sp : s ∈PUnf[ a , f ])
         → igc r s ∈PUnf[ a , f ]
```

Finally, we present the decoding function PUnf2ℜnf. This represents a proof that $\mathsf{PU}(a,f) \subseteq \Re$. The first equation relies on the fact that $a$ can be in the preuniverse only if $a \in \Re$. The second equation uses that the normal form of $f \cdot x$ can only be in PU provided it is in $\Re$

```
PUnf2ℜnf : {a f r : Term} → r ∈PUnf[ a , f ] → r ∈ℜnf
PUnf2ℜnf (aproof pa) = pa
PUnf2ℜnf (fproof xp s fsred sR) = sR
```

The remaining equations use that both $\mathsf{PU}(a,f)$ and $\Re$ are universes closed under the 10 basic set forming constructions of explicit mathematics:

```
PUnf2ℜnf natpu              = natℜ
PUnf2ℜnf (invpu g gnf rp) = invℜ g gnf (PUnf2ℜnf rp)
PUnf2ℜnf (rp ∪pu sp)       = PUnf2ℜnf rp ∪ℜ PUnf2ℜnf sp
PUnf2ℜnf (⋂pu rp)          = ⋂ℜ  (PUnf2ℜnf rp)
PUnf2ℜnf (igpu rp sp)      = igℜ  (PUnf2ℜnf rp) (PUnf2ℜnf sp)
PUnf2ℜnf (jpu rp s nfs sn sxPU)
                 = jℜ (PUnf2ℜnf rp) s nfs sn
                      (λ x xpu nf → PUnf2ℜnf (sxPU x xpu nf) )
```

We again emphasize that the two indexed inductive-recursive definitions depend on each other. The introductory clauses for $a$ and $f$ for $\in$PUnf$[a\,,f]$ refer to $\in\Re$nf. Both the introductory clause for (u $a\ f$) for $\in\Re$nf and its decoding refer to $\in$PUnf$[a\,,f]$. Without these clauses, the two definitions are definable in the type theory of inductive-recursive definitions specified by the authors [10, 12].

In Section 11, we will discuss why the remaining clauses maintain the constructive, predicative validity of the definition. The crux of the matter is the constructor $\mathsf{u}\ a\ f \in \mathfrak{R}$ and its decoding, and the constructors for $a$ and $f$ in the preuniverse $\in$PUnf[$a$ , $f$].

## 8 External extended predicative Mahlo in explicit mathematics, version 2.0

When we provide meaning explanations for the extended predicative Mahlo universe in type theory, we need a more complex argument for the case when we have $\mathrm{Indep}_{\mathfrak{R}}(a,f)$ and add $(\mathsf{u}\ a\ f)$ to $\mathfrak{R}$. This is, because the elements of $(\mathsf{u}\ a\ f)$ are those belonging to $\mathsf{PU}(a,f)$. This requires some additional argument why we can use $\mathsf{PU}(a,f)$, and it does not refer to $\mathfrak{R}$ anymore.

In this revised version, we make this argument explicit. Instead of defining the extension of $(\mathsf{u}\ a\ f)$ as $\mathsf{PU}(a,f)$ we define, once we have established independence, a new (type-theoretic) set $\mathrm{U}(a,f)$, which is defined like $\mathsf{PU}(a,f)$, but does not refer to tests that the elements added are already in $\mathfrak{R}$. Instead we show that it follows from independence that it is a subset of $\mathsf{PU}(a,f)$ and therefore of $\mathfrak{R}$.

The new axioms $(\mathrm{Cl}^2_{\mathsf{PU}})$, which replace $(\mathrm{Cl}_{\mathsf{PU}})$, are as follows:

We add a new predicate $\mathrm{U}(a,f)$ for terms $a,f$ and have the rules

$$\begin{aligned}
&\mathrm{Indep}_{\mathfrak{R}}(a,f) \to \mathrm{U}(a,f) \subseteq \mathsf{PU}(a,f) \wedge \Gamma^{\mathsf{pot}}_{\mathsf{pu}}(a,f,\mathrm{U}(a,f)) \subseteq \mathrm{U}(a,f)\\
&\mathrm{Indep}_{\mathfrak{R}}(a,f) \wedge \Gamma^{\mathsf{pot}}_{\mathsf{pu}}(a,f,\mathcal{A}) \dot{\subseteq} \mathcal{A} \to \mathrm{U}(a,f) \dot{\subseteq} \mathcal{A}\\
&\mathrm{Indep}_{\mathfrak{R}}(a,f) \to \mathsf{u}\ a\ f \in \mathfrak{R} \wedge \mathsf{u}\ a\ f \doteq \mathrm{U}(a,f).
\end{aligned}$$

The axioms $(\mathsf{Pred\text{-}M}^{\mathsf{ext},2})$ for the external extended predicative Mahlo universe Vers 2.0 consist of $(\mathsf{PU})$ and $(\mathrm{Cl}^2_{\mathsf{PU}})$.

## 9 External extended predicative Mahlo in type theory, version 2.0

Since independence occurs very often in this version, we give a separate definition

```
data Indep (a f : Term) : Set where
  indep : (aR : a ∈ℜnf)(fnf : NF f)
          (fnor : (x : Term)(xpu : (x ∈PUnf[ a , f ]))
                  → Normalize (f · x))
          (fxR :  (x : Term) (xpu : (x ∈PUnf[ a , f ]))
                  (p : Normalize (f · x)) → nf p ∈ℜnf)
          → Indep a f
```

Then we define, depending on independence, the universes $\in$Unf[$a, f, indpt$]: First we have closure under $a$ and $f$, which now does not need to check whether it is in $\in\mathfrak{R}$nf:

```
data _∈Unf[_,_,_] : (r a f : Term)(indep : Indep a f)
                     → Set where
  aproofu : {a f : Term}{indep : Indep a f}
            → a ∈Unf[ a , f , indep ]
  fproofu  : {a f : Term}{indep : Indep a f}
            (r : Term)
            (rpu : r ∈Unf[ a , f , indep ])
            (s : Term)(fxred : (f · r) -»NF s)(sℜnf : s ∈ℜnf)
            → s ∈Unf[ a , f , indep ]
```

The remaining rules are closure under set operators which are as before, see the Appendix B for the complete set of rules.

The embedding of $\in$Unf[$a, f, indpt$] into $\in$PUnf[$a$, $f$] is given as follows. First we define it for closure under $a$ and $f$, which makes use of independence

```
Unf2PUnf : {a f r : Term}{indp : Indep a f}
           → r ∈Unf[ a , f , indp ] → r ∈PUnf[ a , f ]
Unf2PUnf {a} {f} {.a} {indep aR fnf fnor fxR} aproofu
           = aproof aR
Unf2PUnf {a} {f} {s} {indep aR fnf fnor fxR}
                     (fproofu r rpu s frxred sR)
           = fproof (Unf2PUnf rpu) s frxred (fxR r (Unf2PUnf rpu)
                   (normalize s frxred))
```

The remaining set constructions can be found in Appendix B.

The introduction and equality rule for (u $a$ $f$) in version 2.0 are as follows

```
uℜ : (a f : Term)(indp : Indep a f) → u a f ∈ℜnf

r ∈nf· uℜ a f indp = r ∈Unf[ a , f , indp ]
```

## 10 Closure of the extended predicative Mahlo universe under the axiomatic Mahlo rules

We show that the external extended predicative Mahlo universe (Vers. 2.0) is closed under the rules for the axiomatic Mahlo universe. (The same should apply to Vers. 1.)

The rules for the axiomatic Mahlo universe express that if $a \in \mathfrak{R}$ and $f : \mathfrak{R} \to R$, then (u $a$ $f$) $\in \mathfrak{R}$. In type theory, the closure condition is as follows:

```
data Clos (a f : Term) : Set where
  clos : (aR : a ∈ℜnf)
         (fnf : NF f)
         (fnor : (x : Term) → x ∈ℜnf
                 → Normalize (f · x))
         (fxR :  (x : Term) → x ∈ℜnf
                 → (p : Normalize (f · x)) → nf p ∈ℜnf)
         → Clos a f
```

We prove that closure implies independence

```
clos2Indep : (a f : Term)(cl : Clos a f)
           → Indep a f
clos2Indep a f (clos aR fnf fnor fxR)
         = indep aR fnf ( λ x xpunf   → fnor x (PUnf2ℜnf xpunf))
                          λ x xpunf p → fxR x (PUnf2ℜnf xpunf) p
```

It follows that $\mathfrak{R}$ is closed under u:

$$\begin{aligned}&\mathsf{uAx} : (a\ f : \mathsf{Term})\ (cl : \mathsf{Clos}\ a\ f)\\ &\quad \rightarrow \mathsf{u}\ a\ f \in\!\mathfrak{R}\mathsf{nf}\\ &\mathsf{uAx}\ a\ f\ cl = \mathsf{u}\mathfrak{R}\ a\ f\ (\mathsf{clos2Indep}\ a\ f\ cl)\end{aligned}$$

We show that u $a\,f$ is a subset of $\mathfrak{R}$

$$\begin{aligned}&\mathsf{uAxaf}\!\subseteq\!\mathfrak{R} : (a\ f : \mathsf{Term})\ (cl : \mathsf{Clos}\ a\ f)\\ &\qquad (y : \mathsf{Term})(yp : y \in\!\mathsf{nf}\cdot\ (\mathsf{uAx}\ a\ f\ cl))\\ &\qquad \rightarrow y \in\!\mathfrak{R}\mathsf{nf}\\ &\mathsf{uAxaf}\!\subseteq\!\mathfrak{R}\ a\ f\ cl\ y\ yp = \mathsf{PUnf2}\mathfrak{R}\mathsf{nf}\ (\mathsf{Unf2PUnf}\ yp\ )\end{aligned}$$

Moreover, u $a\,f$ contains $a$

$$\begin{aligned}&\mathsf{aproofAx} : (a\ f : \mathsf{Term})(cl : \mathsf{Clos}\ a\ f)\\ &\qquad \rightarrow a \in\!\mathsf{nf}\cdot\ (\mathsf{uAx}\ a\ f\ cl)\\ &\mathsf{aproofAx}\ a\ f\ cl = \mathsf{aproofu}\end{aligned}$$

and is closed under $f$

$$\begin{aligned}&\mathsf{fproofAx} : (a\ f : \mathsf{Term})(cl : \mathsf{Clos}\ a\ f)\\ &\qquad (r : \mathsf{Term})\\ &\qquad (rpu : r \in\!\mathsf{nf}\cdot\ (\mathsf{uAx}\ a\ f\ cl))\\ &\qquad (s : \mathsf{Term})(fxred : (f\ \cdot\ r)\ \text{-»}\mathsf{NF}\ s)\\ &\qquad \rightarrow s \in\!\mathsf{nf}\cdot\ (\mathsf{uAx}\ a\ f\ cl)\\ &\mathsf{fproofAx}\ a\ f\ (\mathsf{clos}\ aR\ fnf\ fnor\ fxR)\ x\ xp\ s\ fr\text{->}s\\ &\quad = \mathsf{fproofu}\ x\ xp\ s\ fr\text{->}s\ (fxR\ x\ (\mathsf{PUnf2}\mathfrak{R}\mathsf{nf}\ (\mathsf{Unf2PUnf}\ xp))\ (\mathsf{normalize}\ s\ fr\text{->}s))\end{aligned}$$

(u $a\,f$) is closed under the universe operators. We show here only a few examples. Closure under nat is as follows:

$$\begin{aligned}&\mathsf{natAx} : (a\ f : \mathsf{Term})(cl : \mathsf{Clos}\ a\ f)\\ &\qquad \rightarrow \mathsf{nat} \in\!\mathsf{nf}\cdot\ (\mathsf{uAx}\ a\ f\ cl)\\ &\mathsf{natAx}\ a\ f\ cl = \mathsf{natu}\end{aligned}$$

Closure under $\dot{\cap}$

$$\begin{aligned}&\cap\mathsf{Ax} : (a\ f : \mathsf{Term})(cl : \mathsf{Clos}\ a\ f)(r\ r' : \mathsf{Term})\\ &\qquad (rpu : r \in\!\mathsf{nf}\cdot\ \mathsf{uAx}\ a\ f\ cl)\\ &\qquad (r'pu : r' \in\!\mathsf{nf}\cdot\ \mathsf{uAx}\ a\ f\ cl)\\ &\qquad \rightarrow (r \cap\cdot\ r') \in\!\mathsf{nf}\cdot\ \mathsf{uAx}\ a\ f\ cl\\ &\cap\mathsf{Ax}\ a\ f\ cl\ r\ r' = \_\cap\mathsf{u}\_\end{aligned}$$

Closure under $\bigcap$

$$\begin{aligned}&\textstyle\bigcap\mathsf{ax} : (a\ f : \mathsf{Term})(cl : \mathsf{Clos}\ a\ f)(r : \mathsf{Term})\\ &\qquad (rpu : r \in\!\mathsf{nf}\cdot\ \ \mathsf{uAx}\ a\ f\ cl)\\ &\qquad \rightarrow (\textstyle\bigcap\cdot\ r) \in\!\mathsf{nf}\cdot\ \mathsf{uAx}\ a\ f\ cl\\ &\textstyle\bigcap\mathsf{ax}\ a\ f\ cl\ rpu = \bigcap\mathsf{u}\end{aligned}$$

Closure under j

$$
\begin{aligned}
&\mathsf{jax} : (a\ f : \mathsf{Term})(cl : \mathsf{Clos}\ a\ f)\\
&\qquad (r : \mathsf{Term})(rp : r \in\!\mathsf{nf}\cdot\ \mathsf{uAx}\ a\ f\ cl)\\
&\qquad (s : \mathsf{Term})(nfs : \mathsf{NF}\ s)\\
&\qquad (sn \quad : (x : \mathsf{Term}) \to x \in\!\mathsf{nf}\cdot\ \mathsf{uAxaf}{\subseteq}\Re\ a\ f\ cl\ r\ rp\\
&\qquad\qquad\quad \to \mathsf{Normalize}\ (s\ \cdot\ x))\\
&\qquad (sxU : (x : \mathsf{Term}) \to (xrp : x \in\!\mathsf{nf}\cdot\ \mathsf{uAxaf}{\subseteq}\Re\ a\ f\ cl\ r\ rp)\\
&\qquad\qquad\quad (p : \mathsf{Normalize}\ (s\ \cdot\ x))\\
&\qquad\qquad\quad \to \mathsf{nf}\ p \in\!\mathsf{nf}\cdot\ \mathsf{uAx}\ a\ f\ cl)\\
&\qquad \to (\mathsf{jc}\ r\ s) \in\!\mathsf{nf}\cdot\ \mathsf{uAx}\ a\ f\ cl\\
&\mathsf{jax}\ a\ f\ cl\ r = \mathsf{ju}
\end{aligned}
$$

## 11 Meaning explanations for the type theoretic extended predicative Mahlo Universe

We simultaneously explain the meaning of $\in\!\Re\mathsf{nf}$ and $\in\!\mathsf{PUnf}[a\,,f]$ for any $a, f$. Simultaneously, we explain the meaning of

- $r \in\!\mathsf{nf}\cdot\ s$ for any term $r$ and any element $s$ we have determined as an element of $\in\!\Re\mathsf{nf}$,
- proofs $\mathsf{PUnf2}\Re\mathsf{nf}\ r$ as an element of $\in\!\Re\mathsf{nf}$ for any element $s$ we have determined by proof $r$ as an element of $\in\!\mathsf{PUnf}[a\,,f]$.

For the standard set formers of explicit mathematics, we have that once we have constructed the elements of $\in\!\Re\mathsf{nf}$ from which it is formed, we introduce the new constructed element of $\in\!\Re\mathsf{nf}$ and give its meaning as determined by the meaning explanation for the set formers of explicit mathematics. Once we have introduced the elements of $\in\!\mathsf{PUnf}[a\,,f]$ from which it is constructed, it is an element of $\in\!\mathsf{PUnf}[a\,,f]$, and the meaning of $\mathsf{PUnf2}\Re\mathsf{nf}$ applied to it is the proof that it is an element of $\in\!\Re\mathsf{nf}$ as defined before.

Furthermore, if $a$ is an element of $\in\!\Re\mathsf{nf}$, then $a$ is an element of $\in\!\mathsf{PUnf}[a\,,f]$ and $\mathsf{PUnf2}\Re\mathsf{nf}$ returns the proof that $a \in\!\Re\mathsf{nf}$. If $x$ is an element of $\in\!\mathsf{PUnf}[a\,,f]$, and $f\ x$ has a normal form $b$ and $b$ is an element of $\in\!\Re\mathsf{nf}$, then $b$ is an element of $\in\!\mathsf{PUnf}[a\,,f]$, and the proof object is mapped to the proof object that $b$ is in $\in\!\Re\mathsf{nf}$.

Assume that $a$ is an element of $\in\!\Re\mathsf{nf}$, $f$ is in normal form, and for any element $x$ in $\in\!\mathsf{PUnf}[a\,,f]$, we have that $(f\ x)$ has a normal form $b$ which is in $\in\!\Re\mathsf{nf}$. We call this definition *indp*

We define a set $\in\mathsf{Unf}[a, f, indpt]$ and for any proof $p$ that an element is in $\in\mathsf{Unf}[a, f, indpt]$ a proof ($\mathsf{UnfPUnf}\ p$) that it is an element of $\in\!\mathsf{PUnf}[a\,,f]$: For every standard set construction from explicit mathematics, we construct new elements of $\in\mathsf{Unf}[a, f, indpt]$ and map the proof that it is $\in\mathsf{Unf}[a, f, indpt]$ to the proof that it is in $\in\!\mathsf{PUnf}[a\,,f]$. By *indp* we have a proof that $a$ is in $\in\!\Re\mathsf{nf}$ and therefore a proof $p$ that $a\ \in\!\mathsf{PUnf}[a\,,f]$. Therefore we state a proof that $a\ \in\mathsf{Unf}[a, f, indpt]$ and map it to $p$. If $x$ is an element of $\in\mathsf{Unf}[a, f, indpt]$, then it is an element of $\in\!\mathsf{PUnf}[a\,,f]$, and therefore by *indp* there exists an element $b$ in normal form which is the reduct of $f\ x$ and is $\in\!\Re\mathsf{nf}$ and therefore $\in\!\mathsf{PUnf}[a\,,f]$. Let $p$ be that proof. Then we state that $b$ is $\in\mathsf{Unf}[a, f, indpt]$ and we map its proof to $p$. That completes the explanation of $\in\mathsf{Unf}[a, f, indpt]$. Now we state that ($\mathsf{u}\ \ a\ f$) is an element of $\Re$ and $r\ \in\!\mathsf{nf}\cdot\ \mathsf{u}\ a\ f$ if $r{\in}\mathsf{Unf}[a, f, indpt]$.

## 12 Conclusion

In this paper, we reviewed Kahle and Setzer's extended predicative Mahlo universe [20] which aims to provide a more predicative justification of Jäger, Strahm and Studer's original Mahlo universe in explicit mathematics. First we introduced an external variant of Kahle and Setzer's internal extended predicative Mahlo universe. When formalized in Martin-Löf type theory, this external version corresponds to an internal Mahlo universe in type theory. Then we formalized the external extended predicative Mahlo universe in an extension of Martin-Löf type theory with certain indexed inductive-recursive definitions which go beyond the indexed inductive-recursive definitions definable in our theory **IIRD** [12]. We have also given a second version of the external extended predicative Mahlo universe in Martin-Löf type theory. In this version, it is more transparent that the subuniverses (u $a$ $f$) are fully closed under $a$ and $f$ without depending on the fact that each generated element is already a name. We also showed that this version is closed under the rules for the axiomatic Mahlo universe in explicit mathematics. Finally, we have given meaning explanations for the external extended predicative Mahlo universe, justifying that it is indeed constructive and predicative in Martin-Löf's extended sense.

## Appendix A: More details regarding the formalization of explicit mathematics in type theory

We will formalize a (type-theoretic) set Term of terms corresponding to the set of terms in explicit mathematics. Moreover, we formalize reduction rules and normal forms.

Note that in explicit mathematics quantifiers are over defined terms. Here these correspond to terms which reduce to normal form. Technically, we define the set of names in normal form, and for each name its elements in normal form. A name is then a name which reduces to a term in normal form. Elements of a name are those which reduce to a term in normal form.

We start with a set of constructors and combinators, where combinators include the constructors. We add one or more suffixes c to the constructors, so that we can use the name without the suffixes for a function which operates like the combinator but can be applied to the arguments (and for the $s$ and $k$-combinators, we keep the suffix to avoid clashes with variables).

```
data Constructor : Set where
  πcc zerocc succc natcc idcc invcc ∅cc ∩ccc ∪ccc -»ccc ⋂ccc : Constructor
  ⋃ccc ucc pucc mahlocc jccc igccc : Constructor

data Combinator : Set where
  scc kcc π₀c π₁c primc plusc : Combinator
  cons : Constructor → Combinator
```

We now define a term as a combinator applied to a list of terms

```
data Term : Set where
  _·_ : (c : Combinator)(tl : TermList) → Term

data TermList : Set where
  [] : TermList
  _::_ : (t : Term)(tl : TermList) → TermList
```

Using Agda patterns/syntax construct, one can now redefine them so that they look like simple applications (we give only some examples)

$$
\begin{aligned}
&\mathsf{pattern\ nat} && = \mathsf{natc} \cdot\cdot\ [] \\
&\mathsf{pattern\ kc}\ r\ s\ l && = \mathsf{kcc} \cdot\cdot\ (r :: s :: l\,) \\
&\mathsf{pattern\ sc}\ r\ s\ t\ l && = \mathsf{scc} \cdot\cdot\ (r :: s :: t :: l) \\
&\mathsf{syntax}\ \pi\ r\ s && = <\ r\ ,\ s\ > \\
&\mathsf{syntax}\ \cap\mathsf{c}\ r\ s && = r \cap\cdot\ s
\end{aligned}
$$

Since terms are combinators applied to lists of terms, application is a defined operation

$$
\begin{aligned}
&\_\cdot\_ && : \mathsf{Term} \to \mathsf{Term} \to \mathsf{Term} \\
&\_\cdot\mathsf{L}\_ && : \mathsf{Term} \to \mathsf{TermList} \to \mathsf{Term}
\end{aligned}
$$

The result type of one step reductions will be either terms in normal form or terms which can be reduced further. The resulting type $\mathsf{NF}\uplus\mathsf{Red}$ is

$$
\begin{aligned}
&\mathsf{data\ NF}\uplus\mathsf{Red} : \mathsf{Set\ where} \\
&\quad \mathsf{nft}\ : (t : \mathsf{Term}) \to \mathsf{NF}\uplus\mathsf{Red} \\
&\quad \mathsf{red}\ : (t : \mathsf{Term}) \to \mathsf{NF}\uplus\mathsf{Red}
\end{aligned}
$$

Now we define the one step reduction of a term

$$
\begin{aligned}
&\mathsf{reduce} : \mathsf{Term} \to \mathsf{NF}\uplus\mathsf{Red} \\
&\mathsf{reduce}\ (\mathsf{kc}\ s\ t\ tl) && = \mathsf{red}\ (s \cdot\mathsf{L}\ tl) \\
&\mathsf{reduce}\ (\mathsf{sc}\ r\ s\ t\ tl) && = \mathsf{red}\ (((r \cdot t) \cdot (s \cdot t)) \cdot\mathsf{L}\ tl) \\
&\mathsf{reduce}\ (\mathsf{prim}\ f\ g\ \mathsf{zero}\ tl) && = \mathsf{red}\ (f \cdot\mathsf{L}\ tl) \\
&\mathsf{reduce}\ (\mathsf{prim}\ f\ g\ (\mathsf{suc}\ n)\ tl) && = \mathsf{red}\ (((g \cdot n) \cdot (\mathsf{prim}\ f\ g\ n\ [])) \cdot\mathsf{L}\ tl) \\
&\mathsf{reduce}\ (\pi_0\ (\pi\ r\ s :: l)) && = \mathsf{red}\ (r \cdot\mathsf{L}\ l) \\
&\mathsf{reduce}\ (\pi_1\ (\pi\ r\ s :: l)) && = \mathsf{red}\ (s \cdot\mathsf{L}\ l)
\end{aligned}
$$

In all other cases, a term is reduced by reducing the terms it is applied to (where the definition is in such way that the term is in normal form if all of the terms the combinator is applied to are already in normal form)

$$
\mathsf{reduce}\ (c \cdot\cdot\ tl) = c \cdot\mathsf{nfl}\ (\mathsf{reduceL}\ tl)
$$

$$
\begin{aligned}
&\mathsf{reduceL} : \mathsf{TermList} \to \mathsf{NF}\uplus\mathsf{RedList} \\
&\mathsf{reduceL}\ [] && = \mathsf{nfl}\ [] \\
&\mathsf{reduceL}\ (t :: tl) && = \mathsf{reducenfl}\ (\mathsf{reduce}\ t)\ tl
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{reducenfl} : \mathsf{NF}\uplus\mathsf{Red} \to \mathsf{TermList} \to \mathsf{NF}\uplus\mathsf{RedList} \\
&\mathsf{reducenfl}\ (\mathsf{nft}\ t)\ tl = \mathsf{reducenfnfl}\ t\ (\mathsf{reduceL}\ tl) \\
&\mathsf{reducenfl}\ (\mathsf{red}\ t)\ tl = \mathsf{redl}\ (t :: tl)
\end{aligned}
$$

$$
\begin{aligned}
&\mathsf{reducenfnfl} : \mathsf{Term} \to \mathsf{NF}\uplus\mathsf{RedList} \to \mathsf{NF}\uplus\mathsf{RedList} \\
&\mathsf{reducenfnfl}\ t\ (\mathsf{nfl}\ tl) && = \mathsf{nfl}\ (t :: tl\,) \\
&\mathsf{reducenfnfl}\ t\ (\mathsf{redl}\ tl) && = \mathsf{redl}\ (t :: tl)
\end{aligned}
$$

A term reduces to a term in normal form, if it is in normal form or its one step reduct reduces to a term in normal form

```
_-»NF_ : Term → Term → Set
t -»NF t' = nfred2NF (reduce t) t'

data nfred2NF : NF⊎Red → Term → Set where
  isnf : {t : Term}                → nfred2NF (nft t) t
  step : {s t : Term} → s -»NF t → nfred2NF (red s) t
```

A term is in normal form, if the application of reduce returns (nft $t$)

```
NF : (t : Term) → Set
NF t = (reduce t) ≡ (nft t)
```

A term is defined (or normalises) if it reduces to a term in normal form

```
data Normalize (t : Term) : Set where
  normalize : (r : Term) → t -»NF r → Normalize t

nf : {t : Term} → Normalize t → Term
nf (normalize r p) = r
```

We define the relation $r \simeq s$ as a record type [6],
(so, if $p : r \simeq s$, then $p$ .norDir1 : Normalize $r$ → Normalize $s$).

```
record _≃_ (r s : Term) : Set where
  constructor simeq
  field
    norDir1 : Normalize r → Normalize s
    norDir2 : Normalize s → Normalize r
    eqnor   : (p : Normalize r) → (q : Normalize s) → nf p ≡ nf q
```

Because reduction is deterministic, we can easily show uniqueness of normal forms

```
unique2NF : {r s t : Term} → r -»NF s → r -»NF t → s ≡ t
unique2NF rs rt = unique2NF⊎Red rs rt

unique2NF⊎Red : {r : NF⊎Red}{s t : Term} → nfred2NF r s → nfred2NF r t → s ≡ t
unique2NF⊎Red isnf isnf = refl
unique2NF⊎Red (step x) (step x₁) = unique2NF⊎Red x x₁
```

## Appendix B: Complete rules for the external extended predicative Mahlo Universe in type theory, version 2.0

```
data _∈Natnf : Term → Set where
  zerop : zero ∈Natnf
```

```
  sucp  : {t : Term} → t ∈Natnf → suc t ∈Natnf

data _∈Nat (s : Term) : Set where
  red   : {t : Term} → s -»NF t → t ∈Natnf → s ∈Nat

data _∈Idnf : (s : Term) → Set where
  idp : {t : Term} → NF t → < t , t > ∈Idnf

data _∈Id (s : Term) : Set where
  red : {t : Term}(tf : s -»NF t)(snat : t ∈Idnf) → s ∈Id


data ElJnf (X : Term → Set) (Y : (t : Term) → X t → Term → Set) : Term → Set where
  jcons : {t t' : Term} (tp : X t)(t'p : Y t tp t') → ElJnf X Y (< t , t' >)

data ElJ (X : Term → Set) (Y : (t : Term) → X t → Term → Set) : Term → Set where
  red : {s t : Term}(tf : s -»NF t)(tJ : ElJnf X Y t) → ElJ X Y s

data Ig (X : Term → Set) (Y : Term → Set) : Term → Set where
  cig : {s : Term}→ X s → ((y : Term) → Y (< y , s >) → Ig X Y y)
        → Ig X Y s

_∈Jnf[_,_] : (t : Term)(X : Term → Set) (Y : (t : Term) → X t → Term → Set) → Set
t ∈Jnf[ X , Y ] = ElJnf X Y t

_∈J[_,_] : (t : Term)(X : Term → Set) (Y : (t : Term) → X t → Term → Set) → Set
t ∈J[ X , Y ] = ElJ X Y t


data _∈ℜnf : Term → Set where
  natℜ   : nat ∈ℜnf
  idℜ    : id ∈ℜnf
  ∅ℜ     : ∅ ∈ℜnf
  invℜ   : (f : Term){r : Term} → NF f → r ∈ℜnf → inv f r ∈ℜnf
  _∩ℜ_   : {r r' : Term} → r ∈ℜnf → r' ∈ℜnf → (r ∩· r') ∈ℜnf
  _∪ℜ_   : {r r' : Term} → r ∈ℜnf → r' ∈ℜnf → (r ∪· r') ∈ℜnf
  _-»ℜ_  : {r r' : Term} → r ∈ℜnf → r' ∈ℜnf → (r -» r') ∈ℜnf
  ⋂ℜ     : {r : Term} → r ∈ℜnf → ⋂· r ∈ℜnf
  ⋃ℜ     : {r : Term} → r ∈ℜnf → ⋃· r ∈ℜnf
  jℜ     : {r : Term}(rp : r ∈ℜnf)(s : Term)(nfs : NF s)
           (sn : (x : Term)(xrp : x ∈nf·     rp)
                 → Normalize (s · x))
           (sx∈R : (x : Term)(xrp : x ∈nf· rp)(p : Normalize (s · x))
                 → nf p ∈ℜnf)
           → jc r s ∈ℜnf
  igℜ    : {r s : Term} (rp : r ∈ℜnf)(sp : s ∈ℜnf) → igc r s ∈ℜnf
  uℜ     : (a f : Term)(indp : Indep a f) → u a f ∈ℜnf
```

```
_∈nf·_ : (r : Term){s : Term} → s ∈ℜnf → Set
r ∈nf· natℜ           = r ∈Nat
r ∈nf· idℜ            = r ∈Id
r ∈nf· ∅ℜ             = ⊥
r ∈nf· (sℜ ∩ℜ s'ℜ)    = (r ∈nf· sℜ ) ∧ (r ∈nf· s'ℜ )
r ∈nf· (sℜ ∪ℜ s'ℜ)    = (r ∈nf· sℜ ) ∨ (r ∈nf· s'ℜ )
r ∈nf· (sℜ -»ℜ s'ℜ) = (r ∈nf· sℜ ) → (r ∈nf· s'ℜ )
r ∈nf· ⋂ℜ p           = (y : Term) → < r , y > ∈nf· p
r ∈nf· ⋃ℜ p = ∃[ y ∈ Term ] ( < r , y > ∈nf· p)
z ∈nf· jℜ rp s nfs n sx∈R =
       z ∈J[ (λ t → t ∈nf· rp ) ,
              (λ t trp s' → s' ∈nf· sx∈R t trp (n t trp)) ]
r ∈nf· invℜ f fn s∈ℜnf = (f · r) ∈nf· s∈ℜnf
r ∈nf· igℜ rp sp = Ig (λ t → t ∈nf· rp ) (λ t → t ∈nf· sp) r
r ∈nf· uℜ a f indp = r ∈Unf[ a , f , indp ]


data _∈PUnf[_,_] : (r a f : Term) → Set where
  aproof : {a f : Term}(anf : a ∈ℜnf) → a ∈PUnf[ a , f ]
  fproof : {a f x : Term}(xpu : x ∈PUnf[ a , f ])
           (s : Term)(fxred : (f · x) -»NF s)(sℜnf : s ∈ℜnf)
           → s ∈PUnf[ a , f ]
  natpu  : {a f : Term} → nat ∈PUnf[ a , f ]
  idpu   : {a f : Term} → id ∈PUnf[ a , f ]
  ∅pu    : {a f : Term} → ∅ ∈PUnf[ a , f ]
  invpu  : {a f : Term} (g : Term){r : Term}(nfg : NF g)
            (rpu : r ∈PUnf[ a , f ])
            → inv g r ∈PUnf[ a , f ]
  _∩pu_ : {a f r r' : Term}(rpu : r ∈PUnf[ a , f ])(r'pu : r' ∈PUnf[ a , f ])
            → (r ∩· r') ∈PUnf[ a , f ]
  _∪pu_ : {a f r r' : Term}(rpu : r ∈PUnf[ a , f ])(r'pu : r' ∈PUnf[ a , f ])
            → (r ∪· r') ∈PUnf[ a , f ]
  _-»pu_ : {a f r r' : Term}(rpu : r ∈PUnf[ a , f ])(r'pu : r' ∈PUnf[ a , f ])
            → (r -» r') ∈PUnf[ a , f ]
  ⋂pu   :   {a f r : Term}(rpu : r ∈PUnf[ a , f ]) → ⋂· r ∈PUnf[ a , f ]
  ⋃pu   :   {a f r : Term}(rpu : r ∈PUnf[ a , f ]) → ⋃· r ∈PUnf[ a , f ]
  jpu    :   {a f r : Term}(rp : r ∈PUnf[ a , f ])(s : Term)(nfs : NF s)
            (sn : (x : Term)(xrp : x ∈nf· PUnf2ℜnf rp)
                   → Normalize (s · x))
            (sxPU : (x : Term) (xrp : x ∈nf· PUnf2ℜnf rp)
                   (p : Normalize (s · x))
                   → nf p ∈PUnf[ a , f ])
            → jc r s ∈PUnf[ a , f ]
  igpu   :   {a f r s : Term}(rp : r ∈PUnf[ a , f ])(sp : s ∈PUnf[ a , f ])
            → igc r s ∈PUnf[ a , f ]
```

```
PUnf2ℜnf : {a f r : Term} → r ∈PUnf[ a , f ] → r ∈ℜnf
PUnf2ℜnf   (aproof pa)     = pa
PUnf2ℜnf   (fproof xp s fsred sR) = sR
PUnf2ℜnf   natpu           = natℜ
PUnf2ℜnf idpu              = idℜ
PUnf2ℜnf ∅pu               = ∅ℜ
PUnf2ℜnf (invpu g gnf rp) = invℜ g gnf (PUnf2ℜnf rp)
PUnf2ℜnf (rp ∩pu sp)       = PUnf2ℜnf rp ∩ℜ PUnf2ℜnf sp
PUnf2ℜnf (rp ∪pu sp)       = PUnf2ℜnf rp ∪ℜ PUnf2ℜnf sp
PUnf2ℜnf (rp -»pu sp)      = PUnf2ℜnf rp -»ℜ PUnf2ℜnf sp
PUnf2ℜnf (⋂pu rp)          = ⋂ℜ (PUnf2ℜnf rp)
PUnf2ℜnf (⋃pu rp)          = ⋃ℜ (PUnf2ℜnf rp)
PUnf2ℜnf (igpu rp sp)      = igℜ (PUnf2ℜnf rp) (PUnf2ℜnf sp)
PUnf2ℜnf (jpu rp s nfs sn sxPU)
           = jℜ (PUnf2ℜnf rp) s nfs sn
                 λ x xpu p → PUnf2ℜnf (sxPU x xpu p )

data Indep (a f : Term) : Set where
  indep : (aR : a ∈ℜnf)(fnf : NF f)
          (fnor : (x : Term)(xpu : (x ∈PUnf[ a , f ]))
                  → Normalize (f · x))
          (fxR :  (x : Term) (xpu : (x ∈PUnf[ a , f ]))
                  (p : Normalize (f · x))
                  → nf p ∈ℜnf)
          → Indep a f

data _∈Unf[_,_,_] : (r a f : Term)(indep : Indep a f)
                      → Set where
  aproofu    : {a f : Term}{indep : Indep a f}
               → a ∈Unf[ a , f , indep ]
  fproofu    : {a f : Term}{indep : Indep a f}
                (r : Term)(rpu : r ∈Unf[ a , f , indep ])
                (s : Term)(fxred : (f · r) -»NF s)(sℜnf : s ∈ℜnf)
                → s ∈Unf[ a , f , indep ]
  natu   : {a f : Term}{indep : Indep a f} → nat ∈Unf[ a , f , indep ]
  idu    : {a f : Term}{indep : Indep a f}
           → id ∈Unf[ a , f , indep ]
  ∅u     : {a f : Term}{indep : Indep a f} → ∅ ∈Unf[ a , f , indep ]
  invu   : {a f : Term}{indep : Indep a f} (g : Term){r : Term}
           (nfg : NF g)(rpu : r ∈Unf[ a , f , indep ])
             → (inv g r) ∈Unf[ a , f , indep ]
  _∩u_ : {a f : Term}{indep : Indep a f}{r r' : Term}
          (rpu :  r ∈Unf[ a , f , indep ])
          (r'pu : r' ∈Unf[ a , f , indep ])
          → (r ∩· r') ∈Unf[ a , f , indep ]
  _∪u_ : {a f : Term}{indep : Indep a f}{r r' : Term}
```

```
           (rpu : r ∈Unf[ a , f , indep ])
           (r'pu : r' ∈Unf[ a , f , indep ])
           → (r ∪· r') ∈Unf[ a , f , indep ]
_-»u_ : {a f : Term}{indep : Indep a f}{r r' : Term}
           (rpu : r ∈Unf[ a , f , indep ])
           (r'pu : r' ∈Unf[ a , f , indep ])
             → (r -» r') ∈Unf[ a , f , indep ]
⋂u : {a f : Term}{indep : Indep a f}{r : Term}
           (rpu : r ∈Unf[ a , f , indep ])
           → (⋂· r) ∈Unf[ a , f , indep ]
⋃u : {a f : Term}{indep : Indep a f}{r : Term}
           (rpu : r ∈Unf[ a , f , indep ])
           → (⋃· r) ∈Unf[ a , f , indep ]
ju   : {a f : Term}{indep : Indep a f}
           {r : Term}(rp : r ∈Unf[ a , f , indep ])
           (s : Term)(nfs : NF s)
           (sn : (x : Term) → x ∈nf· PUnf2ℜnf (Unf2PUnf rp)
                   → Normalize (s · x))
           (sxU : (x : Term) → (xrp : x ∈nf· PUnf2ℜnf (Unf2PUnf rp))
                   (p : Normalize (s · x))
                   → nf p ∈Unf[ a , f , indep ])
           → (jc r s) ∈Unf[ a , f , indep ]
igu : {a f : Term}{indep : Indep a f}{r s : Term}
           (rp : r ∈Unf[ a , f , indep ])(sp : s ∈Unf[ a , f , indep ])
           → (igc r s) ∈Unf[ a , f , indep ]
```

```
Unf2PUnf : {a f r : Term}{indp : Indep a f}
              → r ∈Unf[ a , f , indp ] → r ∈PUnf[ a , f ]
Unf2PUnf {a} {f} {.a} {indep aR fnf fnor fxR} aproofu
              = aproof aR
Unf2PUnf {a} {f} {s} {indep aR fnf fnor fxR}
                        (fproofu r rpu s frxred sR)
              = fproof (Unf2PUnf rpu) s frxred (fxR r (Unf2PUnf rpu)
                        (normalize s frxred))
Unf2PUnf idu = idpu
Unf2PUnf (invu g gnf rp) = invpu g gnf (Unf2PUnf rp)
Unf2PUnf natu             = natpu
Unf2PUnf ∅u               = ∅pu
Unf2PUnf (ju rp s nfs sn sxPU)
             = jpu (Unf2PUnf rp) s nfs sn
               (λ x xpu nf → Unf2PUnf (sxPU x xpu nf ) )
```

$$\mathsf{Unf2PUnf}\ (rp \cap\mathsf{u}\ sp) = \mathsf{Unf2PUnf}\ rp \cap\mathsf{pu}\ \mathsf{Unf2PUnf}\ sp$$
$$\mathsf{Unf2PUnf}\ (rp \cup\mathsf{u}\ sp) = \mathsf{Unf2PUnf}\ rp \cup\mathsf{pu}\ \mathsf{Unf2PUnf}\ sp$$
$$\mathsf{Unf2PUnf}\ (rp\ \text{-»u}\ sp) = \mathsf{Unf2PUnf}\ rp\ \text{-»pu}\ \mathsf{Unf2PUnf}\ sp$$
$$\mathsf{Unf2PUnf}\ (\textstyle\bigcap\mathsf{u}\ rp) = \bigcap\mathsf{pu}\ (\mathsf{Unf2PUnf}\ rp)$$
$$\mathsf{Unf2PUnf}\ (\textstyle\bigcup\mathsf{u}\ rp) = \bigcup\mathsf{pu}\ (\mathsf{Unf2PUnf}\ rp)$$
$$\mathsf{Unf2PUnf}\ (\mathsf{igu}\ rp\ sp) = \mathsf{igpu}\ (\mathsf{Unf2PUnf}\ rp)\ (\mathsf{Unf2PUnf}\ sp)$$

‘

## Funding

Anton Setzer was supported by STSM grant E-COST-GRANT-CA20111-f6450ef0 “Formalisation of Meaning Explanations in Agda” and by the “Training School: Dedukti school & Women in EuroProofNet” E-COST-TRAINING_SCHOOL-CA20111-240622-53c72694-94240eb3-dc2e-11ec-b8f2-0204f3b8d66d, both through COST Action: CA20111 - European Research Network on Formal Proofs.

## References

[1] A. Abel, S. Adelsberger and A. Setzer. Interactive programming in Agda – Objects and graphical user interfaces. *Journal of Functional Programming*, **27**, e8, 2017. https://doi.org/10.1017/S0956796816000319.

[2] Agda Community. *Welcome to Agda's documentation!*, Retrieved 15 January 2023. https://agda.readthedocs.io/.

[3] Agda Team. *Data Types*, Retrieved 17 January 2023. Agda Language Reference, available at https://agda.readthedocs.io/en/v2.6.2.2.20230105/language/data-types.html.

[4] Agda Team. *Implicit Arguments*, Retrieved 17 January 2023. Agda Language Reference, available at https://agda.readthedocs.io/en/v2.6.2.2.20230105/language/implicit-arguments.html.

[5] Agda Team. *Mixfix Operators*, Retrieved 17 January 2023. Agda Language Reference, available at https://agda.readthedocs.io/en/v2.6.2.2.20230105/language/mixfix-operators.html.

[6] Agda Team. *Record Types*, Retrieved 17 January 2023. Agda Language Reference, available at https://agda.readthedocs.io/en/v2.6.2.2.20230105/language/record-types.html.

[7] M.J. Beeson. *Foundations of constructive mathematics: metamathematical studies*. Springer-Verlag, 1985. https://link.springer.com/book/10.1007/978-3-642-68952-9.

[8] P. Dybjer. A general formulation of simultaneous inductive-recursive definitions in type theory. *Journal of Symbolic Logic*, **65**, 525 – 549, 2000. https://doi.org/10.2307/2586554.

[9] P. Dybjer and A. Setzer. A finite axiomatization of inductive-recursive definitions. In *Typed Lambda Calculi and Applications*, J.-Y. Girard. ed., volume 1581 of *Lecture Notes in Computer Science*, pp. 129–146. Springer, 1999. https://doi.org/10.2307/2586554.

[10] P. Dybjer and A. Setzer. Indexed induction-recursion. In *Proof Theory in Computer Science*, R. Kahle, P. Schroeder-Heister and R. Stärk., eds, volume 2183 of *Lecture Notes in Computer Science*, pp. 93–113. Springer Berlin/Heidelberg, 2001. http://dx.doi.org/10.1007/3-540-45504-3_7.

[11] P. Dybjer and A. Setzer. Induction-recursion and initial algebras. *Annals of Pure and Applied Logic*, **124**, 1–47, 2003. https://doi.org/10.1016/S0168-0072(02)00096-9.

[12] P. Dybjer and A. Setzer. Indexed induction-recursion. *Journal of Logic and Algebraic Programming*, **66**, 1–49, 2006. http://dx.doi.org/10.1016/j.jlap.2005.07.001.

[13] P. Dybjer and A. Setzer. *Extended predicative Mahlo type theory*, January 2023. Git repository of Agda code. Available at https://gitlab.com/setzer/extended-predicative-mahlo-type-theory/.

[14] S. Feferman. A language and axioms for explicit mathematics. In *Algebra and Logic*, J. Crossley., ed, volume 450 of *Lecture Notes in Mathematics*, pp. 87–139. Springer, 1975. http://dx.doi.org/10.1007/BFb0062852.

[15] G. Jäger. Induction in the elementary theory of types and names. In *CSL '87*, E. Börger, H. Büning and M. Richter., eds, volume 329 of *Lecture Notes in Computer Science*, pp. 118–128. Springer Berlin/Heidelberg, 1988. http://dx.doi.org/10.1007/3-540-50241-6_33.

[16] G. Jäger. Metapredicative and explicit Mahlo: a proof-theoretic perspective. In *Logic Colloquium 2000. Proceedings of the Annual European Summer Meeting of the Association for Symbolic Logic, held in Paris, France July 23 - 31, 2000*, R. Cori, A. Razborov, S. Todorčević and C. Wood., eds, Lecture Notes in Logic, pp. 272 – 293. A K Peters. Association for Symbolic Logic, 2005. https://www.taylorfrancis.com/chapters/edit/10.1201/9781439865750-10/metapredicative-explicit-mahlo-proof-theoretic-perspective-gerhard-j\%C3\%A4ger.

[17] G. Jäger, R. Kahle and T. Studer. Universes in explicit mathematics. *Annals of Pure and Applied Logic*, **109**,141–162, 2001. https://doi.org/10.1016/S0168-0072(00)00057-9.

[18] G. Jäger and T. Strahm. Upper bounds for metapredicative Mahlo in explicit mathematics and admissible set theory. *Journal of Symbolic Logic*, **66**, 935–958, 2001. http://dx.doi.org/10.2307/2695054.

[19] G. Jäger and T. Studer. Extending the system $\mathsf{T}_0$ of explicit mathematics: the limit and Mahlo axioms. *Annals of Pure and Applied Logic*, **114**, 79–101, 2002. http://dx.doi.org/10.1016/S0168-0072(01)00076-8.

[20] R. Kahle and A. Setzer. An extended predicative definition of the Mahlo universe. In *Ways of Proof Theory*, R. Schindler., ed, Ontos Series in Mathematical Logic, pp. 309–334, Berlin, Boston, 2010. De Gruyter. http://dx.doi.org/10.1515/9783110324907.315.

[21] P. Martin-Löf. An intuitionistic theory of types: predicative part. In *Logic Colloquium '73, Proceedings of the Logic Colloquium*, H.E. Rose and J.C. Shepherdson., eds, volume 80 of *Studies in Logic and the Foundations of Mathematics*, pp. 73–118. North-Holland, 1975. https://doi.org/10.1016/S0049-237X(08)71945-1.

[22] P. Martin-Löf. Constructive mathematics and computer programming. In *Logic, Methodology and Philosophy of Science VI, Proceedings of the Sixth International Congress of Logic, Methodology and Philosophy of Science, Hannover 1979*, L. J. Cohen, J. Loś, H. Pfeiffer and K.-P. Podewski., eds, volume 104 of *Studies in Logic and the Foundations of Mathematics*, pp. 153–175. North-Holland, 1982. http://dx.doi.org/10.1016/S0049-237X(09)70189-2.

[23] P. Martin-Löf. *Intuitionistic Type Theory*, volume 1 of *Studies in Proof Theory*. Bibliopolis, 1984.

[24] P. Martin-Löf. An intuitionistic theory of types. In *Twenty-five years of constructive type theory (Venice, 1995)*, G. Sambin and J. M. Smith., eds, volume 36 of *Oxford Logic Guides*, pp. 127–172. Oxford University Press, 1998. https://global.oup.com/academic/product/twenty-five-years-of-constructive-type-theory-9780198501275?cc=gb&lang=en&.

[25] E. Palmgren. On universes in type theory. In *Twenty-five years of constructive type theory: proceedings of a congress held in Venice, October 1995*, G. Sambin and J.M. Smith., eds, volume 36, pp. 191–204. Oxford University Press, 1998. https://global.oup.com/academic/product/twenty-five-years-of-constructive-type-theory-9780198501275?cc=gb&lang=en&.

[26] M. Rathjen. Ordinal notations based on a weakly Mahlo cardinal. *Archive for Mathematical Logic*, **29**, 249–263, 1990. https://doi.org/10.1007/BF01651328.

[27] M. Rathjen. Proof-theoretic analysis of KPM. *Archive of Mathematical Logic*, **30**, 377–403, 1991. https://doi.org/10.1007/BF01621475.

[28] A. Setzer. *Proof theoretical strength of Martin-Löf Type Theory with W-type and one universe*. PhD thesis, Mathematisches Institut, Universität München, Munich, Germany, 1993. Available from http://www.cs.swan.ac.uk/~csetzer/articles/weor0.pdf.

[29] A. Setzer. Extending Martin-Löf type theory by one Mahlo-universe. *Archive for Mathematical Logic*, **39**, 155–181, 2000. http://dx.doi.org/10.1007/s001530050140.

[30] A. Setzer. Universes in type theory part I – Inaccessibles and Mahlo. In *Logic Colloquium '04*, A. Andretta, K. Kearnes and D. Zambella., eds, Lecture Notes in Logic, pp. 123–156. Association of Symbolic Logic, Lecture Notes in Logic 29, Cambridge University Press, 2008. http://dx.doi.org/10.1017/CBO9780511721151.009.

[31] A. Setzer. *Universes in type theory Part II – Autonomous Mahlo*, 2010. Available fromhttp://www.cs.swan.ac.uk/~csetzer/articles/modelautomahlomain.pdf.

[32] T. Strahm. Wellordering proofs for metapredicative Mahlo. *Journal of Symbolic Logic*, **67**, 260–278, 2002. http://dx.doi.org/10.2178/jsl/1190150043.

[33] S. Tupailo. Realization of analysis into explicit mathematics. *Journal of Symbolic Logic*, **66**, 1848–1864, 2001. https://doi.org/10.2307/2694980.

[34] S. Tupailo. Realization of constructive set theory into explicit mathematics: a lower bound for impredicative Mahlo universe. *Annals of Pure and Applied Logic*, **120**, 165–196, 2003. https://doi.org/10.1016/S0168-0072(02)00065-9.

[35] S. Tupailo. Realization of constructive set theory into explicit mathematics: a lower bound for impredicative Mahlo universe. *Annals of Pure and Applied Logic*, **120**, 165–196, 2003. https://doi.org/10.1016/S0168-0072(02)00065-9.

Received 17 January 2023