Weak theories of truth and explicit mathematics

Sebastian Eberhard[*] Thomas Strahm[**]

Dedicated to Helmut Schwichtenberg on his retirement

Abstract

We study weak theories of truth over combinatory logic and their relationship to weak systems of explicit mathematics. In particular, we consider two truth theories $\mathsf{T_{PR}}$ and $\mathsf{T_{PT}}$ of primitive recursive and feasible strength. The latter theory is a novel abstract truth-theoretic setting which is able to interpret expressive feasible subsystems of explicit mathematics.

1 Introduction

The theories of truth and explicit mathematics considered in this article are all based on a common applicative ground language for operations in the sense of combinatory logic; operations can freely be applied to other operations and strong principles of recursion are available due to the known expressivity of combinatory algebras. The first order applicative base describes the operational core of Feferman's explicit mathematics, cf. [15, 16, 17]; instead of a predicate N for natural numbers we will consider a predicate W in order to single out those operations which denote binary words.

[*]Institut für Informatik und angewandte Mathematik, Universität Bern, Neubrückstrasse 10, CH-3012 Bern, Switzerland. Email: `eberhard@iam.unibe.ch`

[**]Institut für Informatik und angewandte Mathematik, Universität Bern, Neubrückstrasse 10, CH-3012 Bern, Switzerland. Email: `strahm@iam.unibe.ch`. Homepage: `http://www.iam.unibe.ch/~strahm`

Types (or classifications) in explicit mathematics are extensional collections of operations. They are generated successively and linked to the applicative ground structure by a naming relation: the names of a type constitute its intensional or computational representations. The interplay of types and names on the level of combinatory operations makes the framework of explicit mathematics very expressive.

An alternative means to extend first order applicative theories by a typing discipline is to extend them by a unary truth predicate T and interpret naive set theory by stipulating $x \in a$ as $\mathsf{T}(ax)$. The so-obtained axiomatic frameworks of partial, self-referential truth are rooted in Kripke's seminal work and also yield an interpretation of classical Frege structures (cf. Aczel [1], Beeson [2], and Hayashi and Kobayashi [27]). For detailed background on the type of truth theories considered here, see Cantini [6, 7] and Kahle [30, 31]. Of course, the work on axiomatic truth over combinatory logic is also strongly related to corresponding work in the area of arithmetical truth theories, see e.g. Feferman [18, 21], Friedman and Sheard [25], and Halbach [26].

The focus of the present paper is to discuss various weak (positive) truth theories and systems of explicit mathematics as well as their mutual relationship. Namely we will address two families of theories, capturing the primitive recursive and polynomial time computable functions, respectively. We will see that the truth theories can interpret corresponding systems of explicit mathematics very directly, whereas reverse embeddings of truth theories into explicit mathematics are more elaborate and require additional assumptions.

A further novelty of this paper is the introduction of a natural feasible truth theory T_{PT}, whose provably total operations are the polynomial time computable ones, as is shown in Eberhard [12]. T_{PT} can only reflect initial segments of the class W of binary words, but features unrestricted truth induction; it is obtained as a natural restriction of a truth theory T_{PR} of the strength of primitive recursive arithmetic. Moreover, T_{PT} can interpret very expressive feasible systems of explicit mathematics.

We conclude the introduction with a detailed outline of the paper. In Section

2 we will introduce the basic applicative framework which is common to all systems studied in this paper. Section 3 presents the two central truth theories of this paper, $\mathsf{T_{PR}}$ and $\mathsf{T_{PT}}$. The first one was previously introduced in Cantini [10, 7]. Both systems rely on a form of positive truth and embody truth induction. Whereas $\mathsf{T_{PR}}$ can reflect the whole predicate W of binary words, $\mathsf{T_{PT}}$ only reflects initial segments. In Section 4 we will present two natural systems of explicit mathematics of polynomial and primitive recursive strength, respectively: the system PETJ of Spescha and Strahm [37, 36, 38] and a natural explicit system EPCJ; both of these frameworks are direct subsystems of Feferman's EM_0 plus the join principle (cf. [15, 17]). For the embedding of truth theories into explicit mathematics, further principles will be needed, for example, the existence of universes, and Cantini's uniformity principle. Section 5 will be devoted to mutual embeddings of weak truth theories and systems of explicit mathematics. Firstly, we will see that PETJ and EPCJ are very directly contained in $\mathsf{T_{PT}}$ and $\mathsf{T_{PR}}$, respectively. The reverse embeddings are more difficult: (i) for the direct embedding of $\mathsf{T_{PR}}$ into EPCJ we assume the existence of a universe and the uniformity principle; (ii) the reduction of $\mathsf{T_{PT}}$ to PETJ proceeds via an intermediate leveled truth theory, which in turn can be directly modeled in an extension of PETJ by universes. In Section 6 we turn to an extended discussion of the proof theory of the systems considered in this paper; this includes the review of some known results and a discussion of work under preparation, namely Eberhard's novel realizability interpretation of $\mathsf{T_{PT}}$, which also yields that the extensions of the systems of explicit mathematics mentioned before do not raise the proof-theoretic strength. We conclude this article with an outlook of future work, namely the application of the feasible truth theory $\mathsf{T_{PT}}$ in order to obtain proof-theoretic upper bounds for the unfolding of schematic systems of feasible arithmetic.

2 The basic applicative framework

The theories of truth and explicit mathematics studied in this paper are based on a common applicative base theory. It includes the axioms for a partial or total combinatory algebra and a basic data type of binary words.

2.1 The applicative language L

Our basic language L is a first order language for the logic of partial terms which includes:

- variables $a, b, c, x, y, z, u, v, f, g, h, \dots$
- constants k, s, p, $\mathsf{p_0}$, $\mathsf{p_1}$, $\mathsf{d_W}$, ϵ, $\mathsf{s_0}$, $\mathsf{s_1}$, $\mathsf{p_W}$, $\mathsf{c_\subseteq}$, $*$, $\times$
- relation symbols $=$ (equality), $\downarrow$ (definedness), W (binary words)
- arbitrary term application $\circ$

The meaning of the constants will become clear in the next paragraph.

The terms $(r, s, t, p, q, \dots)$ and formulas $(A, B, C, \dots)$ of L are defined in the expected manner. We assume the following standard abbreviations and syntactical conventions:

$$\begin{array}{rcl}
t_1 t_2 \dots t_n & := & (\dots(t_1 \circ t_2) \circ \dots \circ t_n) \\
s(t_1, \dots, t_n) & := & s t_1 \dots t_n \\
t_1 \simeq t_2 & := & t_1\downarrow \vee\, t_2\downarrow \;\rightarrow\; t_1 = t_2 \\
\langle t \rangle & := & t \\
\langle t_1, \dots, t_{n+1} \rangle & := & \mathsf{p}\langle t_1, \dots, t_n \rangle t_{n+1} \\
t \in \mathsf{W} & := & \mathsf{W}(t) \\
t : \mathsf{W}^k \rightarrow \mathsf{W} & := & (\forall x_1 \dots x_k \in \mathsf{W}) t x_1 \dots x_k \in \mathsf{W} \\
s \leq t & := & \mathsf{c_\subseteq}(1 {\times} s, 1 {\times} t) = 0 \\
s \leq_{\mathsf{W}} t & := & s \leq t \wedge s \in \mathsf{W}
\end{array}$$

In the following we often write $A[\vec{x}]$ in order to indicate that the variables $\vec{x} = x_1, \dots, x_n$ may occur free in A. Finally, let us write $\overline{w}$ for the canonical closed L term denoting the binary word $w \in \mathbb{W}$.

2.2 The basic theory of operations and words B

The applicative base theory B has been introduced in Strahm [40, 41]. Its logic is the *classical* logic of partial terms due to Beeson [2, 3]. The non-logical axioms of B include:

- partial combinatory algebra:

$$\mathsf{k}xy = x, \qquad \mathsf{s}xy\downarrow \wedge \mathsf{s}xyz \simeq xz(yz)$$

- pairing p with projections p_0 and p_1
- defining axioms for the binary words W with ϵ, the binary successors s_0, s_1 and the predecessor p_W
- definition by cases d_W on W
- initial subword relation $\mathsf{c}_\subseteq$
- word concatenation $*$, word multiplication $\times$[1]

These axioms are fully spelled out in [40, 41]. Below we will be mainly interested in extensions of B by the axioms of *totality of application* and *extensionality of operations*:

Totality of application:

(**Tot**) $$(\forall x)(\forall y)(xy\downarrow)$$

Extensionality of operations:

(**Ext**) $$(\forall f)(\forall g)[(\forall x)(fx \simeq gx) \rightarrow f = g]$$

Observe that in the presence of the totality axiom, the logic of partial terms reduces to ordinary classical predicate logic. In the following we write B^+ for the extension of B by (**Tot**) and (**Ext**).

Various extensions of B or B^+ by suitable induction principles on W have been proposed in the past. Most relevant for the systems studied in this article are the theories PT and PR, cf. Strahm [41]. The former includes a form of bounded induction, namely Σ^b_W induction, whereas the latter features induction for arbitrary positive formulas.

Let us turn to the crucial consequences of the axioms about a partial combinatory algebra. For proofs of these standard results, the reader is referred to Beeson [2] or Feferman [15].

[1] $x{\times}y$ signifies the length of y fold concatenation of x with itself; note that we write $x{\times}y$ instead of $\times xy$.

Lemma 1 (Explicit definitions and fixed points)

1. *For each* L *term* t *there exists an* L *term* $(\lambda x.t)$ *so that*

$$\mathsf{B} \vdash (\lambda x.t)\downarrow \ \wedge\ (\lambda x.t)x \simeq t$$

2. *There is a closed* L *term* fix *so that*

$$\mathsf{B} \vdash \mathsf{fix}g\downarrow \ \wedge\ \mathsf{fix}gx \simeq g(\mathsf{fix}gx)$$

Let us quickly remind the reader of two standard models of B, namely the recursion-theoretic model PRO and the term model $\mathcal{M}(\lambda\eta)$. For an extensive discussion of many more models of the applicative basis, the reader is referred to Beeson [2] and Troelstra and van Dalen [43].

Example 2 (Recursion-theoretic model *PRO*) Take the universe of binary words $\mathbb{W} = \{0,1\}^*$ and interpret application $\circ$ as partial recursive function application in the sense of ordinary recursion theory.

Example 3 (The open term model $\mathcal{M}(\lambda\eta)$) Take the universe of open λ terms and consider the usual reduction of the extensional untyped lambda calculus $\lambda\eta$, augmented by suitable reduction rules for the constants other than k and s. Interpret application as juxtaposition. Two terms are equal if they have a common reduct and W denotes those terms that reduce to a "standard" word $\overline{w}$. Note that $\mathcal{M}(\lambda\eta)$ satisfies both (**Tot**) and (**Ext**).

2.3 Provably total functions

We intend to measure the proof-theoretic strength of all the systems treated in this article by ascertaining their provably total functions. In the following let $\mathcal{L}$ be a language extending our first-order language L. The notion of a *provably total function* is introduced for an arbitrary $\mathcal{L}$ theory Th.

Definition 4 *A function* $F : \mathbb{W}^n \to \mathbb{W}$ *is called provably total in an* $\mathcal{L}$ *theory* Th*, if there exists a closed* L *term* t_F *such that*

(i) $\mathsf{Th} \vdash t_F : \mathsf{W}^n \to \mathsf{W}$ *and, in addition,*

(ii) $\mathsf{Th} \vdash t_F \overline{w}_1 \cdots \overline{w}_n = \overline{F(w_1, \ldots, w_n)}$ *for all* $w_1, \ldots, w_n$ *in* $\mathbb{W}$.

The notion of a provably total word function is divided into two conditions (i) and (ii). The first condition (i) expresses that t_F is a total operation from W^n to W, *provably in the* $\mathcal{L}$ *theory* Th. Condition (ii), on the other hand, claims that t_F indeed represents the given function $F : \mathbb{W}^n \to \mathbb{W}$, for each fixed tuple of words $\vec{w}$ in $\mathbb{W}^n$.

To give an example, the provably total functions of the above-mentioned theories PT and PR are the polynomial time computable and primitive recursive functions, respectively.

3 Positive truth

Theories of truth contain a predicate T that mimics the properties of truth. The axiomatization of this predicate relies on a coding mechanism for formulas. In the applicative framework, we code formulas using new constants designating logical operations. In the weak theories of truth discussed in this paper, the Tarski biconditionals hold only for positive formulas. Therefore no liar paradoxes occur.

In the following we will introduce two weak truth theories T_{PR} and T_{PT}. The theories will be presented simultaneously since their axioms differ only slightly.

3.1 The language L_{T} of positive truth

The (first order) language L_{T} is an extension of the language L by

- a new unary predicate symbol T for *truth*
- new individual constants $\dot{=}$, $\dot{\mathsf{W}}$, $\dot{\wedge}$, $\dot{\vee}$, $\dot{\forall}$, $\dot{\exists}$

The new constants allow only the coding of positive formulas since we do not add a constant $\dot{\neg}$ to code negation. As usual, we will use infix notation for $\dot{=}$, $\dot{\wedge}$, and $\dot{\vee}$.

3.2 Two theories of positive truth

All truth theories considered in this article are based on the applicative theory B^+. Accordingly, their underlying logic is simply first order classical predicate logic. The truth axioms for T_{PR} and T_{PT} differ only in the Tarski biconditionals which are available for the word predicate W. The truth axioms for T_{PR} spell out the expected clauses according to the compositional semantics of truth as follows.

Compositionality:

$$
\begin{array}{lll}
(\dot{=}) & \mathsf{T}(x \doteq y) & \leftrightarrow \quad x = y \\
(\dot{\mathsf{W}}_{\mathsf{PR}}) & \mathsf{T}(\dot{\mathsf{W}}x) & \leftrightarrow \quad \mathsf{W}(x) \\
(\dot{\wedge}) & \mathsf{T}(x \mathbin{\dot{\wedge}} y) & \leftrightarrow \quad \mathsf{T}(x) \wedge \mathsf{T}(y) \\
(\dot{\vee}) & \mathsf{T}(x \mathbin{\dot{\vee}} y) & \leftrightarrow \quad \mathsf{T}(x) \vee \mathsf{T}(y) \\
(\dot{\forall}) & \mathsf{T}(\dot{\forall} f) & \leftrightarrow \quad (\forall z)\mathsf{T}(fz) \\
(\dot{\exists}) & \mathsf{T}(\dot{\exists} f) & \leftrightarrow \quad (\exists z)\mathsf{T}(fz)
\end{array}
$$

For the feasible theory T_{PT}, we use instead of $(\dot{\mathsf{W}}_{\mathsf{PR}})$ the following axiom:

$$(\dot{\mathsf{W}}_{\mathsf{PT}}) \qquad\qquad x \in \mathsf{W} \to (\mathsf{T}(\dot{\mathsf{W}}xy) \leftrightarrow y \leq_{\mathsf{W}} x)$$

In contrast to $(\dot{\mathsf{W}}_{\mathsf{PR}})$, it allows only the reflection of initial segments of the set of words. Both theories contain unrestricted truth induction.

Truth induction:

$$\mathsf{T}(r\epsilon) \wedge (\forall x \in \mathsf{W})(\mathsf{T}(rx) \to \mathsf{T}(r(\mathsf{s}_0 x)) \wedge \mathsf{T}(r(\mathsf{s}_1 x))) \to (\forall x \in \mathsf{W})\mathsf{T}(rx)$$

Next we would like to determine the classes of formulas for which the Tarski truth biconditionals hold. In the case of T_{PT} this is the class of so-called simple formulas, which are patterned after similar classes of formulas in explicit mathematics, see [37, 34] and the next section of this paper.

Definition 5 (Simple formulas) *Let A be a positive L_{T} formula and u be a variable not occurring in A. Then the formula A^u which is obtained by replacing each subformula of the form $t \in \mathsf{W}$ of A by $t \leq_{\mathsf{W}} u$ is called simple.*

Next we define coding operations for T_{PR} and T_{PT} which map the positive, respectively the simple formulas to their codes.

Definition 6 *For each positive formula A of L_{T} we inductively define a term $[A]$ whose free variables are exactly the free variables of A:*

$$
\begin{aligned}
[t = s] &:= t \doteq s \\
[\mathsf{T}(t)] &:= t \\
[s \in \mathsf{W}] &:= \dot{\mathsf{W}} s \\
[A \wedge B] &:= [A] \dot{\wedge} [B] \\
[A \vee B] &:= [A] \dot{\vee} [B] \\
[(\forall x) A] &:= \dot{\forall}(\lambda x.[A]) \\
[(\exists x) A] &:= \dot{\exists}(\lambda x.[A])
\end{aligned}
$$

Definition 7 *For each simple formula A^u of L_{T} we inductively define a term $\langle A \rangle$ whose free variables are exactly the free variables of A:*

$$
\begin{aligned}
\langle t = s \rangle &:= t \doteq s \\
\langle \mathsf{T}(t) \rangle &:= t \\
\langle s \leq_{\mathsf{W}} u \rangle &:= \dot{\mathsf{W}} u s \\
\langle A \wedge B \rangle &:= \langle A \rangle \dot{\wedge} \langle B \rangle \\
\langle A \vee B \rangle &:= \langle A \rangle \dot{\vee} \langle B \rangle \\
\langle (\forall x) A \rangle &:= \dot{\forall}(\lambda x.\langle A \rangle) \\
\langle (\exists x) A \rangle &:= \dot{\exists}(\lambda x.\langle A \rangle)
\end{aligned}
$$

We have that $\lambda x.[A]$, respectively $\lambda x.\langle A \rangle$ can be interpreted as the propositional function defined by the formula A. For both theories of truth, the Tarski biconditionals can be proved for the positive, respectively simple formulas.

Lemma 8 (Biconditionals for $\mathsf{T_{PR}}$) *Let A be a positive L_{T} formula. We have*

$$\mathsf{T_{PR}} \vdash \mathsf{T}([A]) \leftrightarrow A$$

Lemma 9 (Biconditionals for $\mathsf{T_{PT}}$) *Let A^u be a simple L_{T} formula. We have*

$$\mathsf{T_{PT}} \vdash u \in \mathsf{W} \rightarrow (\mathsf{T}(\langle A^u \rangle) \leftrightarrow A^u)$$

An interesting consequence of the biconditionals is a second recursion or fixed point theorem for positive, respectively simple predicates. This theorem can be obtained by lifting the fixed point theorem for combinatory logic (cf. Lemma 1) to the truth-theoretic language, cf. Cantini [6, 10].

4 Explicit mathematics

Types in explicit mathematics are collections of operations and must be thought of as being generated successively from preceding ones. They are represented by operations via a suitable *naming relation* $\Re$. Types are extensional and have (explicit) names which are intensional. The formalization of explicit mathematics using a naming relation $\Re$ is due to Jäger [28].

We will present the two weak theories of explicit mathematics EPCJ and PETJ and some extensions thereof. We will describe the two theories simultaneously since their axioms differ only slightly.

4.1 The language $\mathbb{L}$ of explicit mathematics

The language $\mathbb{L}$ is a two-sorted language extending L by

- type variables $U, V, W, X, Y, Z, \ldots$
- binary relation symbols $\Re$ (naming) and $\in$ (elementhood)
- new (individual) constants w (sets of words), id (identity), un (union), int (intersection), dom (domain), all (forall), inv (inverse image), and j (join)

The *formulas* $(A, B, C, \ldots)$ of $\mathbb{L}$ are built from the atomic formulas of L as well as from formulas of the form

$$(s \in X), \quad \Re(s, X), \quad (X = Y)$$

by closing under the propositional connectives and quantification in both sorts of variables. The formula $\Re(s, X)$ reads as "the individual s is a name of (or represents) the type X".

We use the following abbreviations:

$$\begin{aligned}\Re(s) &:= (\exists X)\Re(s, X), \\ s \in t &:= (\exists X)(\Re(t, X) \wedge s \in X).\end{aligned}$$

4.2 Two theories of explicit mathematics

In the following we spell out the axioms of the system EPCJ whose characteristic axioms are elementary positive comprehension and join. The applicative basis of EPCJ is B^+ as for all theories of explicit mathematics studied in this paper. Hence their underlying logic is ordinary two-sorted classical predicate logic.

The following axioms state that each type has a name, that there are no homonyms and that equality of types is extensional.

Ontological axioms:

(O1) $\quad (\exists x)\Re(x, X)$

(O2) $\quad \Re(a, X) \wedge \Re(a, Y) \rightarrow X = Y$

(O3) $\quad (\forall z)(z \in X \leftrightarrow z \in Y) \rightarrow X = Y$

The following axioms provide a finite axiomatization of the schema of positive elementary comprehension and join.

Type existence axioms:

(w_{PR}) $\quad \Re(\mathsf{w}) \wedge (\forall x)(x \in \mathsf{w} \leftrightarrow x \in \mathsf{W})$

(id) $\quad \Re(\mathsf{id}) \wedge (\forall x)(x \in \mathsf{id} \leftrightarrow (\exists y)(x = \langle y, y \rangle))$

(inv) $\quad \Re(a) \rightarrow \Re(\mathsf{inv}(f, a)) \wedge (\forall x)(x \in \mathsf{inv}(f, a) \leftrightarrow fx \in a)$

(un) $\quad \Re(a) \wedge \Re(b) \rightarrow \Re(\mathsf{un}(a, b)) \wedge (\forall x)(x \in \mathsf{un}(a, b) \leftrightarrow (x \in a \vee x \in b))$

(int) $\quad \Re(a) \wedge \Re(b) \rightarrow \Re(\mathsf{int}(a, b)) \wedge (\forall x)(x \in \mathsf{int}(a, b) \leftrightarrow (x \in a \wedge x \in b))$

(dom) $\quad \Re(a) \rightarrow \Re(\mathsf{dom}(a)) \wedge (\forall x)(x \in \mathsf{dom}(a) \leftrightarrow (\exists y)(\langle x, y \rangle \in a))$

(all) $\quad \Re(a) \rightarrow \Re(\mathsf{all}(a)) \wedge (\forall x)(x \in \mathsf{all}(a) \leftrightarrow (\forall y)(\langle x, y \rangle \in a))$

(j.1) $\quad \Re(a) \wedge (\forall x \in a)\Re(fx) \rightarrow \Re(\mathsf{j}(a, f))$

(j.2) $\quad \Re(a) \wedge (\forall x \in a)\Re(fx) \rightarrow (\forall x)(x \in \mathsf{j}(a, f) \leftrightarrow \Sigma(f, a, x))$

where $\Sigma(f, a, x)$ is the formula

$$(\exists y)(\exists z)(x = \langle y, z \rangle \wedge y \in a \wedge z \in fy)$$

The only difference between EPCJ and PETJ is that for PETJ we replace the axiom (w_{PR}) by (w_{PT}).

(w_{PT}) $$a \in \mathsf{W} \;\rightarrow\; \Re(\mathsf{w}(a)) \wedge (\forall x)(x \in \mathsf{w}(a) \leftrightarrow x \leq_{\mathsf{W}} a)$$

In contrast to the comprehension schema available in EPCJ, in PETJ it is not claimed that the collection of binary words forms a type, but merely that for each word a, the collection $\{x \in \mathsf{W} : x \leq a\}$ forms a type, uniformly in a.

Finally, both theories include the principle of type induction along W.

Type induction on W:

$$\epsilon \in X \wedge (\forall x \in \mathsf{W})(x \in X \rightarrow \mathsf{s}_0 x \in X \wedge \mathsf{s}_1 x \in X) \rightarrow (\forall x \in \mathsf{W})(x \in X)$$

The finite axiomatizations of elementary comprehension in EPCJ and PETJ immediately imply corresponding schemes of elementary comprehension with respect to the characteristic formula classes of EPCJ and PETJ, respectively.

Lemma 10 (Positive Comprehension) *Let $A[a, \vec{x}, \vec{X}]$ be a positive $\mathbb{L}$ formula with exactly the free variables displayed which does neither contain the predicate $\Re$ nor second order quantifiers. Then there exists a term $t_A[\vec{x}, \vec{z}]$ with exactly the free variables displayed such that* EPCJ *proves*

$$\Re(\vec{z}, \vec{X}) \;\rightarrow\; \Re(t_A[\vec{x}, \vec{z}]) \wedge (\forall a)\big(a \in t_A[\vec{x}, \vec{z}] \leftrightarrow A[a, \vec{x}, \vec{X}]\big)$$

Given a positive $\mathbb{L}$ formula A which does neither contain the predicate $\Re$ nor second order quantifiers, the restriction A^u is defined in the same way as in Definition 5. The so-obtained formulas are called simple $\mathbb{L}$ formulas.

Lemma 11 (Simple Comprehension) *Let $A^u[u, a, \vec{x}, \vec{X}]$ be a simple $\mathbb{L}$ formula with exactly the free variables displayed. Then there exists a term $t_A[u, \vec{x}, \vec{z}]$ with exactly the free variables displayed such that* PETJ *proves*

$$u \in \mathsf{W} \wedge \Re(\vec{z}, \vec{X}) \;\rightarrow\; \Re(t_A[u, \vec{x}, \vec{z}]) \wedge (\forall a)\big(a \in t_A[u, \vec{x}, \vec{z}] \leftrightarrow A^u[u, a, \vec{x}, \vec{X}]\big)$$

Lemma 10 and Lemma 11 allow us to use set notation. We will sometimes write $\{x : A[x]\}$ instead of t_A where t_A is defined as above.

4.3 Extensions

In standard models of explicit mathematics, the elementhood and naming relation are constructed in stages; the same applies to standard models of truth theories, see Feferman [15] and Cantini [6]. Beginning with sentences which can be immediately seen to be true, we establish the truth of more complex statements. The truth predicate can then be conceived as joining the truth stage predicates of at least ω many stages. To interpret this object in explicit mathematics, we will define recursively types corresponding to particular stages. However the admissibility of an iterative definition of types presupposes induction on W for the naming predicate to prove that the type constructors work as intended. In the following, we will expand our theories of explicit mathematics such that name induction is possible at least in a restricted way. But instead of expanding these theories by a type reflecting the name predicate, we vote for the weaker and more usual extension by universes. This additional expressive power makes it possible to interpret the truth theories $\mathsf{T_{PR}}$ and $\mathsf{T_{PT}}$ respectively.

A universe is a type U such that:

- U is closed under (positive) elementary comprehension and join;
- All elements of U are names.

To introduce universes precisely we define a closure condition in the following way: $C_{\mathsf{EPCJ}}(z, a)$ holds iff one of the following conditions is satisfied:

- $a = \mathsf{w}$
- $a = \mathsf{id}$
- $(\exists x)(\exists f)(a = \mathsf{inv}(f, x) \wedge x \in z)$
- $(\exists x)(\exists y)(a = \mathsf{un}(x, y) \wedge x \in z \wedge y \in z)$
- $(\exists x)(\exists y)(a = \mathsf{int}(x, y) \wedge x \in z \wedge y \in z)$
- $(\exists x)(a = \mathsf{dom}(x) \wedge x \in z)$
- $(\exists x)(a = \mathsf{all}(x) \wedge x \in z)$

- $(\exists x)(\exists f)\big[a = \mathsf{j}(x, f) \land x \in z \land (\forall y \in x)(fy \in z)\big]$

For PETJ we have to adapt the first condition: we replace $a = \mathsf{w}$ by the formula $(\exists u \in \mathsf{W})(a = \mathsf{w}u)$. We call the modified formula $C_{\mathsf{PETJ}}(z, a)$.

Assuming that z is a name, the formula $(\forall x)(C_{\mathsf{EPCJ}}(z, x) \to x \in z)$ expresses that z names a type that is closed under the type constructors of the theory EPCJ; analogously for PETJ. We abbreviate the formula

$$(\forall x)(C_{\mathsf{EPCJ}}(z, x) \to x \in z) \land (\forall x)(x \in z \to \Re(x)) \land \Re(z)$$

by $\mathsf{U}_{\mathsf{EPCJ}}(z)$; the formula $\mathsf{U}_{\mathsf{PETJ}}(z)$ is defined analogously.

Next assume that the language $\mathbb{L}$ contains two additional constants ℓ_{EPCJ} and ℓ_{PETJ}. The following two axioms state that ℓ_{EPCJ} and ℓ_{PETJ} create an EPCJ or PETJ universe respectively, if applied to a name.

$(\mathsf{U}_{\mathsf{EPCJ}})$ $\quad \Re(a) \;\to\; \mathsf{U}_{\mathsf{EPCJ}}(\ell_{\mathsf{EPCJ}}(a)) \land a \in \ell_{\mathsf{EPCJ}}(a)$

$(\mathsf{U}_{\mathsf{PETJ}})$ $\quad \Re(a) \;\to\; \mathsf{U}_{\mathsf{PETJ}}(\ell_{\mathsf{PETJ}}(a)) \land a \in \ell_{\mathsf{PETJ}}(a)$

In order to keep the notation simple, we write $\mathsf{EPCJ} + \mathsf{U}$ instead of $\mathsf{EPCJ} + \mathsf{U}_{\mathsf{EPCJ}}$ and analogously $\mathsf{PETJ} + \mathsf{U}$ instead of $\mathsf{PETJ} + \mathsf{U}_{\mathsf{PETJ}}$. Similarly, we drop the subscript of ℓ_{EPCJ} and ℓ_{PETJ} if it is clear from the context.

Using the universe it is possible to code the elementhood relation of its types by using a suitable join.

Lemma 12 *There exists a closed term* e *such that for* $\mathsf{Th} = \mathsf{PETJ} + \mathsf{U}$ *or* $\mathsf{EPCJ} + \mathsf{U}$ *we have that* Th *proves*

$$\Re(a) \to \Re(\mathsf{e}(a)) \land (\forall x)\big(x \in \mathsf{e}(a) \leftrightarrow (\exists y)(\exists z)(x = \langle y, z\rangle \land z \in \ell(a) \land y \in z\big)$$

Below, Cantini's uniformity principle (cf. [10]) is needed for the embedding of T_{PR}. It claims for each positive $\mathbb{L}$ formula A

(UP) $\quad (\forall x)(\exists y \in \mathsf{W})A[x, y] \to (\exists y \in \mathsf{W})(\forall x)A[x, y]$

This concludes the description of the relevant extensions of explicit mathematics.

5 Embeddings

We will embed the theories of explicit mathematics with universes into the theories of truth and vice versa. The embedding of the theories of explicit mathematics with universes is straightforward. The reverse embeddings are more difficult and work in a different way for both theories of truth: it seems to be impossible to embed $\mathsf{T_{PT}}$ into $\mathsf{PETJ} + \mathsf{U}$ directly. Instead we embed a leveled theory of truth to which $\mathsf{T_{PT}}$ is reducible by an asymmetric interpretation argument.

In this section we assume an equivalent first order formulation of EPCJ and PETJ. The first order formulations postulate the type-theoretic axioms directly via a unary naming predicate $\Re$ and a binary elementhood relation $\in$ between *individuals*. The first and the second order versions can be mutually embedded for both theories of explicit mathematics. For details about the embedding, see e.g. Spescha [36] or Spescha and Strahm [38].

5.1 Embedding weak theories of explicit mathematics into weak truth theories

For both weak truth theories introduced in the paper, the embedding works completely analogously. We take the embedding of $\mathsf{EPCJ} + \mathsf{U}$ into $\mathsf{T_{PR}}$ as example. The main idea is to interpret the elementhood relation by using the truth predicate and to trivialize the universes. The translation * of a formula $s \in t$ will be

$$\mathsf{T}(t^* s^*).$$

To make this translation work, we have to interpret the type constructors in the right way. The idea is to translate them by predicates, which embody their membership condition.

Definition 13 (Translation of terms) *For each term t of $\mathbb{L}$, its translation t^* into L_{T} is defined recursively on the complexity of t in the following way.*

- *All applicative constants are left untouched.*

- $\mathsf{id}^* \equiv \lambda z.\dot{\exists}\lambda y.\ z \doteq \langle y, y\rangle$
- $\mathsf{w}^* \equiv \lambda z.\ \dot{\mathsf{W}}z$
- $\mathsf{int}^* \equiv \lambda a.\lambda b.\lambda z.\ az \mathrel{\dot{\wedge}} bz$
- $\mathsf{un}^* \equiv \lambda a.\lambda b.\lambda z.\ az \mathrel{\dot{\vee}} bz$
- $\mathsf{inv}^* \equiv \lambda f.\lambda a.\lambda z.\ a(fz)$
- $\mathsf{dom}^* \equiv \lambda a.\lambda z.\ \dot{\exists}\lambda y.a\langle z, y\rangle$
- $\mathsf{all}^* \equiv \lambda a.\lambda z.\ \dot{\forall}\lambda y.a\langle z, y\rangle$
- $\mathsf{j}^* \equiv \lambda f.\lambda a.\lambda z.\ \dot{\exists}\lambda x.\dot{\exists}\lambda y.z \doteq \langle x, y\rangle \mathrel{\dot{\wedge}} ax \mathrel{\dot{\wedge}} (fx)y$
- ${\ell_{\mathsf{EPCJ}}}^* \equiv \lambda a.\lambda z.0 \doteq 0$
- $st^* \equiv s^*t^*$

Formulas are translated in the following way: atomic formulas commute with * except for the formulas of the shape $s \in t$ whose translation is $\mathsf{T}(t^*s^*)$ and the formulas of the shape $\Re(s)$ whose translation is $0 = 0$. The translation commutes with negation, propositional connectives and quantifiers.

For this translation, the embedding theorem below can be proved without difficulties. Since the name predicate is interpreted trivially, the translations of the universe axioms hold in T_{PR}. Moreover, the translation can be modified in the obvious way in order to provide an embedding of $\mathsf{PETJ} + \mathsf{U}$ into T_{PT}.

Theorem 14 $\mathsf{EPCJ}+\mathsf{U}$ *and* $\mathsf{PETJ}+\mathsf{U}$ *are contained in* T_{PR} *and* T_{PT} *via the* * *translation or a slight modification thereof, respectively.*

Let us mention that this embedding theorem also holds for expansions of explicit mathematics by positive uniformity if the truth theories are expanded analogously.

5.2 Embedding of T_{PR} into $\mathsf{EPCJ} + \mathsf{U} + \mathsf{UP}$

First, we define types corresponding to levels of truth. We construct a truth-level type τw for each word w. Using join we can then collect all these truth-level types. Because of UP the resulting type satisfies the translated truth axioms.

The types τw for the truth levels all consist of tuples of three elements. The first element contains a code for a logical symbol of L_{T}. All these codes are assumed to be different words. The second and the third element stand for the terms the logical constant is applied to. The third element is sometimes only a placeholder. Let us define the bottom truth level $\tau\epsilon$ as

$$\{\langle a,b,c\rangle | (a = \ulcorner \dot{=} \urcorner \wedge b = c) \vee (a = \ulcorner \dot{\mathsf{W}} \urcorner \wedge \mathsf{W}(b) \wedge c = \epsilon)\}.$$

The types for the higher truth levels are defined recursively (using the fixed point theorem of B) in the following way:

$$\begin{aligned}\tau(\mathsf{s}_i w) := \tau w \cup \{\langle a,b,c\rangle | \ & \big(a = \ulcorner \dot{\wedge} \urcorner \wedge b \in \tau w \wedge c \in \tau w\big) && \vee \\ & \big(a = \ulcorner \dot{\vee} \urcorner \wedge [b \in \tau w \vee c \in \tau w]\big) && \vee \\ & \big(a = \ulcorner \dot{\exists} \urcorner \wedge (\exists x)(bx \in \tau w) \wedge c = \epsilon\big) && \vee \\ & \big(a = \ulcorner \dot{\forall} \urcorner \wedge (\forall x)(bx \in \tau w) \wedge c = \epsilon\big) && \}\end{aligned}$$

To justify the type notation, we have to show that the above given terms τw are names for each $w \in \mathsf{W}$. Only then, the type constructors work in the intended way and indeed name the above displayed types. Since $\Re$-induction is not available, we use type induction with the universe $\ell_{\mathsf{EPCJ}}(\mathsf{id})$ for this purpose. It is easy to see that $\tau\epsilon \in \ell_{\mathsf{EPCJ}}(\mathsf{id})$ and

$$(\forall w \in \mathsf{W})(\tau w \in \ell_{\mathsf{EPCJ}}(\mathsf{id}) \rightarrow \tau(\mathsf{s}_i w) \in \ell_{\mathsf{EPCJ}}(\mathsf{id}))$$

hold. We apply type induction and use the fact $(\forall x)\big(x \in \ell_{\mathsf{EPCJ}}(\mathsf{id}) \rightarrow \Re(x)\big)$ to get the desired result.

Now the stage is set to define a translation $^\circ$ of T_{PR} into $\mathsf{EPCJ} + \mathsf{U} + \mathsf{UP}$. In particular, we translate the truth predicate using the above defined hierarchy of types.

Definition 15 (Translation of terms) *For each term t of L_T, its translation t° is defined inductively on its complexity in the following way.*

- *All applicative constants are left untouched.*
- $\dot{=}^\circ \equiv \lambda x.\lambda y.\langle \ulcorner \dot{=} \urcorner, x, y\rangle$
- $\dot{\mathsf{W}}^\circ \equiv \lambda x.\langle \ulcorner \dot{\mathsf{W}} \urcorner, x, \epsilon\rangle$
- $\dot{\wedge}^\circ \equiv \lambda x.\lambda y.\langle \ulcorner \dot{\wedge} \urcorner, x, y\rangle$
- $\dot{\vee}^\circ \equiv \lambda x.\lambda y.\langle \ulcorner \dot{\vee} \urcorner, x, y\rangle$
- $\dot{\exists}^\circ \equiv \lambda x.\langle \ulcorner \dot{\exists} \urcorner, x, \epsilon\rangle$
- $\dot{\forall}^\circ \equiv \lambda x.\langle \ulcorner \dot{\forall} \urcorner, x, \epsilon\rangle$
- $(st)^\circ \equiv s^\circ t^\circ$

Definition 16 (Translation of formulas) *For each formula A of L_T, its translation A° is defined inductively in the following way.*

- $(s = t)^\circ \equiv s^\circ = t^\circ$
- $(s \in \mathsf{W})^\circ \equiv s^\circ \in \mathsf{W}$
- $\mathsf{T}(t)^\circ \equiv t^\circ \in \mathsf{dom}(\mathsf{inv}(\lambda x.\langle \mathsf{p_1} x, \mathsf{p_0} x\rangle, \mathsf{j}(\mathsf{W}, \tau)))$
- *The translation commutes with negation, propositional connectives and quantifiers.*

Note that by the type axioms of $\mathsf{EPCJ} + \mathsf{U}$, we have

$$t \in \mathsf{dom}(\mathsf{inv}(\lambda x.\langle \mathsf{p_1} x, \mathsf{p_0} x\rangle, \mathsf{j}(\mathsf{W}, \tau))) \leftrightarrow (\exists w \in \mathsf{W})(t \in \tau(w))$$

We are now ready to state the embedding of $\mathsf{T_{PR}}$ into $\mathsf{EPCJ} + \mathsf{U} + \mathsf{UP}$.

Theorem 17 $\mathsf{T_{PR}}$ *is contained in* $\mathsf{EPCJ} + \mathsf{U} + \mathsf{UP}$ *via the* $^\circ$ *translation.*

Proof. It is clear that the translations of the applicative axioms hold in $\mathsf{EPCJ} + \mathsf{U}$. Further, we can show that the translation of truth induction holds in $\mathsf{EPCJ} + \mathsf{U}$ using inv. So let us check the translations of the truth axioms. The direction from right to left is always trivially fulfilled except for $(\dot\forall)$. Its translation is in $\mathsf{EPCJ} + \mathsf{U}$ equivalent to

$$(\forall x)(\exists w \in \mathsf{W})(fx \in \tau w) \to (\exists w \in \mathsf{W})(\langle \ulcorner \dot\forall \urcorner, f, \epsilon \rangle \in \tau w).$$

Using UP, from the antecedens we can derive the existence of a $w \in \mathsf{W}$ such that $(\forall x)(fx \in \tau w)$. This implies that $\langle \ulcorner \dot\forall \urcorner, f, \epsilon \rangle$ is in the successor truth level type $\tau(\mathsf{s}_i w)$.

The direction from left to right is always proved in the same way. We sketch the proof for the $\dot\wedge$-axiom. Let $\dot-$ be defined in the following way.

- $x \dot- \epsilon := x$
- $x \dot- \mathsf{s}_i w := \mathsf{p}_\mathsf{W}(x \dot- w)$

Let us assume the lefthand side of the $\dot\wedge$-axiom. Its translation implies in $\mathsf{EPCJ} + \mathsf{U}$

$$\langle \ulcorner \dot\wedge \urcorner, a, b \rangle \in \tau w$$

for a $w \in \mathsf{W}$ unequal ϵ. We define the formula $A[x]$ as[2]

$$\langle \ulcorner \dot\wedge \urcorner, a, b \rangle \in \tau(w \dot- x) \vee (\exists y \in \mathsf{W})(y \subset w \wedge a \in \tau y \wedge b \in \tau y).$$

Clearly, this formula is progressive in W due to the construction of the truth level types. By type induction we get

$$\langle \ulcorner \dot\wedge \urcorner, a, b \rangle \in \tau\epsilon \vee (\exists y \in \mathsf{W})(y \subset w \wedge a \in \tau y \wedge b \in \tau y).$$

Since the bottom level of truth does not contain tuples of the form $\langle \ulcorner \dot\wedge \urcorner, a, b \rangle$, the second disjunct has to be true. This immediately implies the lefthand side of the $\dot\wedge$-axiom. $\Box$

Note that we needed only the existence of one single universe to prove this embedding. In addition, similarly as described in the next paragraph for T_{PT}, it is also possible to reduce T_{PR} via an intermediate leveled truth theory to $\mathsf{EPCJ} + \mathsf{U}$. This results in a reduction of T_{PR} to $\mathsf{EPCJ} + \mathsf{U}$ which does not depend on the uniformity principle.

[2]We use $s \subset t$ as abbreviation for $\mathsf{d}_\mathsf{W}(0,1,s,t) = 1 \wedge \mathsf{c}_\subseteq st = 0$

5.3 Reduction of T_{PT} to $\mathsf{PETJ} + \mathsf{U}$

Unfortunately an embedding similar to the one in the previous subsection does not seem to be possible in this case. This is because we cannot collect the truth levels for all words, since join can have only initial segments of the type of words as index type.

5.3.1 The leveled truth theory $\mathsf{T}^{\ell}_{\mathsf{PT}}$

Because of the above mentioned reasons, we have to reduce the truth theory T_{PT} to a leveled truth theory $\mathsf{T}^{\ell}_{\mathsf{PT}}$. This means that the predicate T in $\mathsf{T}^{\ell}_{\mathsf{PT}}$ is a binary predicate, whose first argument is written as superscript and interpreted as truth level. This superscript displays the maximal complexity of formulas the corresponding unary truth predicate can reflect. The logical axioms and rules of $\mathsf{T}^{\ell}_{\mathsf{PT}}$ are the usual ones. $\mathsf{T}^{\ell}_{\mathsf{PT}}$ has the following non-logical axioms.

- $a \in \mathsf{W} \to (\mathsf{T}^a(x \doteq y) \leftrightarrow x = y)$
- $a, b \in \mathsf{W} \to (x \leq_{\mathsf{W}} b \leftrightarrow \mathsf{T}^a(\dot{\mathsf{W}}bx))$
- $a \in \mathsf{W} \to \big(\mathsf{T}^{\mathsf{s}_i a}(x \mathbin{\dot\vee} y) \leftrightarrow \mathsf{T}^a(x) \vee \mathsf{T}^a(y)\big)$
- $a \in \mathsf{W} \to \big(\mathsf{T}^{\mathsf{s}_i a}(x \mathbin{\dot\wedge} y) \leftrightarrow \mathsf{T}^a(x) \wedge \mathsf{T}^a(y)\big)$
- $a \in \mathsf{W} \to \big(\mathsf{T}^{\mathsf{s}_i a}(\dot\exists f) \leftrightarrow (\exists z)\mathsf{T}^a(fz)\big)$
- $a \in \mathsf{W} \to \big(\mathsf{T}^{\mathsf{s}_i a}(\dot\forall f) \leftrightarrow (\forall z)\mathsf{T}^a(fz)\big)$
- $a_0, a_1 \in \mathsf{W} \wedge a_0 \leq a_1 \wedge \mathsf{T}^{a_0}(x) \to \mathsf{T}^{a_1}(x)$

Additionally, we have truth induction in the following form:

$$\mathsf{T}^{p\epsilon}(f\epsilon) \wedge (\forall x \in \mathsf{W})\big[\mathsf{T}^{px}(fx) \to \mathsf{T}^{p(\mathsf{s}_0 x)}(f(\mathsf{s}_0 x)) \wedge \mathsf{T}^{p(\mathsf{s}_1 x)}(f(\mathsf{s}_1 x))\big] \\ \to (\forall x \in \mathsf{W})(\mathsf{T}^{px}(fx))$$

In the following, p will always be a polynomial.

For the asymmetric interpretation of T_{PT} in $\mathsf{T}^{\ell}_{\mathsf{PT}}$, we bound the truth level and the W predicate simultaneously. We work as usual with sequent style formulations of T_{PT} and $\mathsf{T}^{\ell}_{\mathsf{PT}}$ which we call T_{PT} and $\mathsf{T}^{\ell}_{\mathsf{PT}}$ as well. We assume that in these calculi all axioms are formulated for terms to guarantee a sufficient cut elimination.

Definition 18 (Asymmetric interpretation) *Let A be a positive L_{T} formula and let a, b be variables. The formula $A^{a,b}$ is defined in the following way.*

- $(t \in \mathsf{W})^{a,b} \equiv t \leq_{\mathsf{W}} a$
- $\mathsf{T}(t)^{a,b} \equiv \mathsf{T}^{b}(t)$

Other atomic formulas are untouched by the asymmetric interpretation. The asymmetric interpretation commutes with propositional connectives and quantifiers.

Next we state the crucial asymmetric interpretation lemma of T_{PT} into $\mathsf{T}^{\ell}_{\mathsf{PT}}$. An immediate consequence of the lemma is that the provably total functions of T_{PT} are contained in the provably total functions of $\mathsf{T}^{\ell}_{\mathsf{PT}}$.

Lemma 19 *Let $\Gamma \Rightarrow \Delta$ be a positive sequent which has a proof of depth k in T_{PT} containing only positive formulas. Then there exists a polynomial p of degree $2^{(2^k)}$ such that*[3]

$$\mathsf{T}^{\ell}_{\mathsf{PT}} \vdash a, b \in \mathsf{W}, \Gamma^{a,b} \Rightarrow \Delta^{pa,pa*b}$$

Proof. We show the lemma by induction on the depth of the positive proof. The only difficult case is induction. In this case we have by induction hypothesis polynomials p, q_0, q_1 of degree $2^{(2^k)}$ with the following properties.

(1) $\quad \mathsf{T}^{\ell}_{\mathsf{PT}} \vdash a, b \in \mathsf{W}, \Gamma^{a,b} \Rightarrow \mathsf{T}^{pa*b}(r\epsilon), \Delta^{pa,pa*b}$

(2) $\quad \mathsf{T}^{\ell}_{\mathsf{PT}} \vdash a, b \in \mathsf{W}, \Gamma^{a,b}, \mathsf{T}^{b}(rx), x \leq_{\mathsf{W}} a \Rightarrow \mathsf{T}^{q_i a*b}(r(\mathsf{s}_i x)), \Delta^{q_i a, q_i a*b}$

Let q be a polynomial that bounds q_0 and q_1. We define the polynomial g as

$$g(x, y) := p(x) * (q(x) \times \mathsf{s}_0 y).$$

[3]We use the notation $pa * b$ to denote the term $*(pa, b)$. Analogously for similar notations.

Because of monotonicity of the asymmetric interpretation the following hold:

(3) $\quad \mathsf{T}^{\ell}_{\mathsf{PT}} \vdash a, b \in \mathsf{W}, \Gamma^{a,b} \Rightarrow \mathsf{T}^{ga\epsilon*b}(r\epsilon), \Delta^{ga\epsilon,ga\epsilon*b}$

(4) $\quad \mathsf{T}^{\ell}_{\mathsf{PT}} \vdash a, b \in \mathsf{W}, \Gamma^{a,gax*b}, \mathsf{T}^{gax*b}(rx), x \leq_{\mathsf{W}} a \Rightarrow$
$\mathsf{T}^{ga(\mathsf{s}_i x)*b}(r(\mathsf{s}_i x)), \Delta^{qa,ga(\mathsf{s}_i x)*b}$

We use again monotonicity of the asymmetric interpretation to unify the side formulas and thus get:

(5) $\quad \mathsf{T}^{\ell}_{\mathsf{PT}} \vdash a, b \in \mathsf{W}, \Gamma^{a,b} \Rightarrow \mathsf{T}^{ga\epsilon*b}(r\epsilon), \Delta^{gaa,gaa*b}$

(6) $\quad \mathsf{T}^{\ell}_{\mathsf{PT}} \vdash a, b \in \mathsf{W}, \Gamma^{a,b}, \mathsf{T}^{gax*b}(rx), x \leq_{\mathsf{W}} a \Rightarrow$
$\mathsf{T}^{ga(\mathsf{s}_i x)*b}(r(\mathsf{s}_i x)), \Delta^{gaa,gaa*b}$

These are the premises for an induction over an initial segment of W which can be proved admissible as usual. After using induction, monotonicity delivers the following.

(7) $\quad \mathsf{T}^{\ell}_{\mathsf{PT}} \vdash a, b \in \mathsf{W}, \Gamma^{a,b}, x \leq_{\mathsf{W}} a \Rightarrow \mathsf{T}^{gaa*b}(rx), \Delta^{gaa,gaa*b}$

Since gaa is a polynomial of degree $2^{(2^k)} + 1$ in a this is the desired result. Note that the degree $2^{(2^{k+1})}$ for the bounding polynomial is needed for the cut rule. $\square$

5.3.2 Embedding of $\mathsf{T}^{\ell}_{\mathsf{PT}}$ into $\mathsf{PETJ} + \mathsf{U}$

In analogy to the previous embedding, we construct a closed term τ such that for all $w \in \mathsf{W}$ the type $\tau(w)$ corresponds to the truth level w. We set

$$\tau(\epsilon) := \{\langle a, b, c\rangle | a = \ulcorner \dot{=} \urcorner \wedge b = c \vee a = \ulcorner \dot{\mathsf{W}} \urcorner \wedge \langle c, \mathsf{w}(b)\rangle \in \mathsf{e}(\mathsf{id})\}.$$

The types for the higher truth levels are defined recursively as before:

$$\begin{aligned}\tau(\mathsf{s}_i w) := \tau w \cup \{\langle a, b, c\rangle | \quad & \big(a = \ulcorner \dot{\wedge} \urcorner \wedge b \in \tau w \wedge c \in \tau w\big) && \vee \\ & \big(a = \ulcorner \dot{\vee} \urcorner \wedge [b \in \tau w \vee c \in \tau w]\big) && \vee \\ & \big(a = \ulcorner \dot{\exists} \urcorner \wedge (\exists x)(bx \in \tau w) \wedge c = \epsilon\big) && \vee \\ & \big(a = \ulcorner \dot{\forall} \urcorner \wedge (\forall x)(bx \in \tau w) \wedge c = \epsilon\big) && \}\end{aligned}$$

As above, we show that these levels are all names by type induction with the universe $\ell(\mathsf{e}(\mathsf{id}))$.

We are now ready to modify the translation $^\circ$ from the last subsection in order to provide a translation from $\mathsf{T}^\ell_{\mathsf{PT}}$ into $\mathsf{PETJ} + \mathsf{U}$.

Definition 20 (Translation of terms) *For each term t of L_{T} its translation t° into $\mathbb{L}$ is defined in the same way as above except that we put*

$$\dot{\mathsf{W}}^\circ \equiv \lambda x.\lambda y.\langle \ulcorner \dot{\mathsf{W}} \urcorner, x, y\rangle$$

Definition 21 (Translation of formulas) *For each formula A of L_{T} its translation A° is defined inductively in the following way.*

- $(s = t)^\circ \equiv s^\circ = t^\circ$
- $(s \in \mathsf{W})^\circ \equiv s^\circ \in \mathsf{W}$
- $\mathsf{T}^s(t)^\circ \equiv \langle t^\circ, \tau(0 \times s^\circ)\rangle \in \mathsf{e}(\mathsf{e}(\mathsf{id}))$
- *The translation commutes with negation, propositional connectives and quantifiers.*

Similarly to the previous embedding theorem, we now obtain the following result.

Theorem 22 $\mathsf{T}^\ell_{\mathsf{PT}}$ *is contained in* $\mathsf{PETJ} + \mathsf{U}$ *via the* $^\circ$ *translation.*

Proof. The translations of the truth axioms can be proved as before; note that the induction formula is equivalent to a simple formula. Because of the leveling the uniformity principle is not needed. An easy induction establishes $v, w \in \mathsf{W} \wedge v \subset w \wedge \mathsf{T}^v(x) \to \mathsf{T}^w(x)$, which implies the translation of monotonicity. $\square$

Note that the previous lemma and theorem readily imply that the provably total functions of T_{PT} are contained in those of $\mathsf{PETJ} + \mathsf{U}$.[4]

[4]The theory T_{PT} can also be reduced to an extension of polynomial strength of PETJ by a *single* universe and an additional type constructor to deal with sharply bounded universal quantification. In this case the intermediate reduction theory is a twice leveled theory of truth whose second level designates the maximal value of s to which formulas of the form $t \leq_{\mathsf{W}} s$ are reflected. The corresponding truth levels can be interpreted in this extension of PETJ without using $\mathsf{e}(\mathsf{id})$.

6 Proof-theoretic analysis

In this section we give an overview of the existing and forthcoming literature regarding the proof theory of weak systems of truth and explicit mathematics, including the ones discussed in this paper.

6.1 Weak systems of explicit mathematics

Let us start by briefly reviewing some previous proof-theoretic work regarding systems of explicit mathematics of strength PRA. Early systems of flexible typing of this strength are extensively studied by Feferman. In [18, 19] he proposes a program to use explicit mathematics to analyze properties of functional programs. In the realm of pure applicative theories, natural systems of strength PRA are proposed and analyzed in Feferman and Jäger [22] and Jäger and Strahm [29]. Theories that are strongly related to the system EPCJ are considered in Krähenbühl [33]. Let us note that all upper bound computations for the systems mentioned above proceed via embeddings into suitable subsystems of Peano arithmetic of strength PRA.

Let us now turn to the discussion of the proof theory of systems of explicit mathematics of polynomial strength. Purely first order systems were proposed and analyzed in Calamai [4], Cantini [8, 9, 10, 5], Kahle and Oitavem [32], and Strahm [39, 40, 41].[5] For those theories, a direct embedding into feasible subsystems of arithmetic does not seem to be possible, as, for example, already the standard interpretation of equality of terms translates into a proper Σ^0_1 statement in arithmetic. Extensions of these weak applicative theories formulated in the full language of explicit mathematics and featuring weak forms of elementary comprehension were first introduced and studied in Spescha and Strahm [37]. In particular, the system PET of types and names[6] has been proposed whose provably total functions are exactly the polytime functions. The PhD thesis of Spescha [36] gives a uniform treatment of various weak systems of explicit mathematics in the spirit of PET, possibly

[5]For a very different and interesting approach to feasibility in the context of Heyting arithmetic and using notions of ramification and linearity for proof terms, see Schwichtenberg [35].

[6]In the notation of this paper, PET is PETJ without the join axioms.

augmented by the axiom of join. A new syntactical approach to the analysis of these systems is proposed via a novel realizability interpretation for the language of types and names, see also the article Spescha and Strahm [38]. The very recent work of Probst [34] solves the delicate and difficult problem of showing that the provably total operations of the system PET with the join axiom and *classical* logic are still the polynomial time computable ones. The article Strahm [42] surveys most of these results. However, let use mention that the techniques used in these papers do not validate expansions of PETJ by universes or by the assertion $\forall\Re$ claiming that everything is a name. These additional principles, however, can be treated by Eberhard's new realizability techniques [12] discussed in the next paragraph.

6.2 Weak truth theories

First examples of very expressive and natural truth theories of strength PRA are presented in Cantini [7]. The proof-theoretic tools used in this article are the technique of asymmetric interpretation as well as subtle formalizations in the subsystem of Peano arithmetic with induction restricted to Σ^0_1 statements. In his more recent [10], Cantini studies a rich family of truth theories of strength PRA including additional principles such as positive choice and uniformity. Special emphasis is put on the reduction of classical truth theories to their intuitionistic counterparts using a forcing relation. The computational content of the intuitionistic truth theories is analyzed by means of suitable realizability techniques. A direct companion to [10] is Cantini [11], which deals with further extensions of the theories in [10]. Let us note that our truth theory T_{PR} can be embedded into PRA plus Σ^0_1 induction by using the formalized term model construction used in the proof of Theorem 9 in Cantini [11].

Let us now turn to the discussion of truth theories of polynomial strength. First examples are treated in Cantini [10], where the truth predicate is used as a guiding technical tool in order to deal with additional principles. One of the core differences between our system T_{PT} and the weak truth theories in [10] is the fact that in T_{PT} we have unrestricted truth induction. This very liberal induction principle makes the proof-theoretic analysis of T_{PT} complicated. The realization approach used in [10] does not work; in particular, there are provable sequents which require exponential realization functions. Nevertheless, it is possible to show that T_{PT} is feasible by using a new realization approach which is developed in Eberhard [12]. The main idea is to use pointers thanks to which the same piece of realization information can be used to realize subformulas of several formulas. This allows to represent and manipulate realization information more efficiently. Using this modified realization approach, one finds polynomial time realization functions for each provable positive sequent. Also the strengthening of T_{PT} by UP can be realized by the same technique. Finally, let us mention that the same approach can also be used in order to realize extensions of the system PETJ of explicit mathematics. In particular, the system $\mathsf{PETJ} + \mathsf{U}$ used in the embeddings above can be realized using the techniques developed in [12].

6.3 Summary of proof-theoretic results

Let us conclude this section by summarizing the results about the proof-theoretic strength of the theories of truth and explicit mathematics considered in this article.

Theorem 23 (Systems of primitive recursive strength) *The provably total functions of the following theories are the primitive recursive ones:*

1. EPCJ*, possibly augmented by* UP *and* U*;*
2. T_{PR}*, possibly augmented by* UP*.*

Theorem 24 (Systems of polynomial strength) *The provably total functions of the following theories are the polynomial time computable ones:*

1. PETJ*, possibly augmented by* UP *and* U*;*
2. T_{PT}*, possibly augmented by* UP*.*

Let us note that for both theories of explicit mathematics, U is a consequence of $\forall\Re$, since under this assumption, ℓa can be interpreted as $\{x : x = x\}$ for any a. Moreover, Eberhard's realization approach trivializes the name predicate and hence can handle $\forall\Re$. Finally, we mention that in the context of $\mathsf{T_{PR}}$, Cantini [10] deals with further choice and reflection principles which do not raise the strength of $\mathsf{T_{PR}}$.

7 Concluding remarks

We have studied two natural truth-theoretic frameworks over combinatory logic and their relationship to weak systems of explicit mathematics. We have seen that the embedding of explicit mathematics into truth theories is very straightforward, whereas the *direct* reverse embeddings require further (natural) extensions of explicit mathematics and sometimes even intermediate reduction steps. The corresponding extensions of explicit mathematics do not increase the proof-theoretic strength of the underlying systems.

The newly proposed feasible truth theory $\mathsf{T_{PT}}$ is also an important reference theory in our recent work on Feferman's unfolding program (see Feferman [20] and Feferman and Strahm [23, 24]): in Eberhard and Strahm [13, 14], the system $\mathsf{T_{PT}}$ plays a crucial role in order to obtain proof-theoretic upper bounds for the full unfolding $\mathcal{U}(\mathsf{FEA})$ of a natural schematic system FEA of feasible arithmetic.

References

[1] Aczel, P. Frege structures and the notion of proposition, truth and set. In *The Kleene Symposium* (1980), J. Barwise, H. Keisler, and K. Kunen, Eds., North-Holland, pp. 31– 59.

[2] Beeson, M. J. *Foundations of Constructive Mathematics: Metamathematical Studies*. Springer, Berlin, 1985.

[3] Beeson, M. J. Proving programs and programming proofs. In *Logic, Methodology and Philosophy of Science VII*, Barcan Marcus et. al., Ed. North Holland, Amsterdam, 1986, pp. 51–82.

[4] Calamai, G. *Proof-theoretic contributions to computational complexity*. PhD thesis, University of Siena, 2008.

[5] Cantini, A. A footnote on the Parsons-Mints-Takeuti theorem. Talk at *Recent Trends in Proof Theory*, Bern, July 2008.

[6] Cantini, A. *Logical Frameworks for Truth and Abstraction.* North-Holland, Amsterdam, 1996.

[7] Cantini, A. Proof-theoretic aspects of self-referential truth. In *Tenth International Congress of Logic, Methodology and Philosophy of Science, Florence, August 1995*, Maria Luisa Dalla Chiara et. al., Ed., vol. 1. Kluwer, September 1997, pp. 7–27.

[8] Cantini, A. Feasible operations and applicative theories based on $\lambda\eta$. *Mathematical Logic Quarterly 46*, 3 (2000), 291–312.

[9] Cantini, A. Polytime, combinatory logic and positive safe induction. *Archive for Mathematical Logic 41*, 2 (2002), 169–189.

[10] Cantini, A. Choice and uniformity in weak applicative theories. In *Logic Colloquium '01*, M. Baaz, S. Friedman, and J. Krajíček, Eds., vol. 20 of *Lecture Notes in Logic*. Association for Symbolic Logic, 2005, pp. 108–138.

[11] Cantini, A. Remarks on applicative theories. *Annals of Pure and Applied Logic 136* (2005), 91–115.

[12] Eberhard, S. A feasible theory of truth over combinatory logic. Preliminary draft, February 2011.

[13] Eberhard, S., and Strahm, T. Unfolding feasible arithmetic and weak truth. Submitted for publication.

[14] Eberhard, S., and Strahm, T. Towards the unfolding of feasible arithmetic (Abstract). *Bulletin of Symbolic Logic* (to appear).

[15] Feferman, S. A language and axioms for explicit mathematics. In *Algebra and Logic*, J. Crossley, Ed., vol. 450 of *Lecture Notes in Mathematics*. Springer, Berlin, 1975, pp. 87–139.

[16] Feferman, S. Recursion theory and set theory: a marriage of convenience. In *Generalized recursion theory II, Oslo 1977*, J. E. Fenstad, R. O. Gandy, and G. E. Sacks, Eds., vol. 94 of *Stud. Logic Found. Math.* North Holland, Amsterdam, 1978, pp. 55–98.

[17] Feferman, S. Constructive theories of functions and classes. In *Logic Colloquium '78*, M. Boffa, D. van Dalen, and K. McAloon, Eds. North Holland, Amsterdam, 1979, pp. 159–224.

[18] Feferman, S. Logics for termination and correctness of functional programs. In *Logic from Computer Science*, Y. N. Moschovakis, Ed., vol. 21 of *MSRI Publications*. Springer, Berlin, 1991, pp. 95–127.

[19] Feferman, S. Logics for termination and correctness of functional programs II: Logics of strength PRA. In *Proof Theory*, P. Aczel, H. Simmons, and S. S. Wainer, Eds. Cambridge University Press, Cambridge, 1992, pp. 195–225.

[20] Feferman, S. Gödel's program for new axioms: Why, where, how and what? In *Gödel '96*, P. Hájek, Ed., vol. 6 of *Lecture Notes in Logic*. Springer, Berlin, 1996, pp. 3–22.

[21] Feferman, S. Axioms for the determinateness of truth. *Review of Symbolic Logic 1* (2008), 204–217.

[22] Feferman, S., and Jäger, G. Systems of explicit mathematics with non-constructive μ-operator. Part I. *Annals of Pure and Applied Logic 65*, 3 (1993), 243–263.

[23] Feferman, S., and Strahm, T. The unfolding of non-finitist arithmetic. *Annals of Pure and Applied Logic 104*, 1–3 (2000), 75–96.

[24] Feferman, S., and Strahm, T. Unfolding finitist arithmetic. *Review of Symbolic Logic 3*, 4 (2010), 665–689.

[25] Friedman, H., and Sheard, M. An axiomatic approach to self-referential truth. *Annals of Pure and Applied Logic 33*, 1 (1987), 1–21.

[26] Halbach, V. *Axiomatic Theories of Truth.* Cambridge University Press, 2011.

[27] Hayashi, S., and Kobayashi, S. A new fomalization of Feferman's system of functions and classes and its relation to Frege structurs. *International Journal of Foundations of Computer Science 6*, 3 (1995), 187–202.

[28] Jäger, G. Induction in the elementary theory of types and names. In *Computer Science Logic '87*, E. Börger, H. Kleine Büning, and M.M. Richter, Eds., vol. 329 of *Lecture Notes in Computer Science*. Springer, Berlin, 1988, pp. 118–128.

[29] Jäger, G., and Strahm, T. Totality in applicative theories. *Annals of Pure and Applied Logic 74*, 2 (1995), 105–120.

[30] Kahle, R. *Applikative Theorien und Frege-Strukturen.* PhD thesis, Institut für Informatik und angewandte Mathematik, Universität Bern, 1997.

[31] Kahle, R. *The Applicative Realm.* Habilitation Thesis, Tübingen, 2007. Appeared in Textos de Mathemática 40, Departamento de Mathemática da Universidade de Coimbra, Portugal, 2007.

[32] Kahle, R., and Oitavem, I. An applicative theory for FPH. In *Proceedings Third International Workshop on Classical Logic and Computation CL&C* (2010), S. van Bakel, S. Berardi, and U. Berger, Eds., vol. 47 of *EPTCS*.

[33] Krähenbühl, J. Explicit mathematics with positive existential comprehension and join. Master's thesis, Institut für Informatik und angewandte Mathematik, Universität Bern, 2006.

[34] Probst, D. The provably terminating operations of the subsystem PETJ of explicit mathematics. *Annals of Pure and Applied Logic 162*, 11 (2011), 934–947.

[35] Schwichtenberg, H. An arithmetic for polynomial time computation. *Theoretical Computer Science 357* (2006), 202–214.

[36] Spescha, D. *Weak systems of explicit mathematics.* PhD thesis, Universität Bern, 2009.

[37] Spescha, D., and Strahm, T. Elementary explicit types and polynomial time operations. *Mathematical Logic Quarterly 55*, 3 (2009), 245–258.

[38] Spescha, D., and Strahm, T. Realizability in weak systems of explicit mathematics. *Mathematical Logic Quarterly 57*, 6 (2011), 551–565.

[39] Strahm, T. Polynomial time operations in explicit mathematics. *Journal of Symbolic Logic 62*, 2 (1997), 575–594.

[40] Strahm, T. *Proof-theoretic Contributions to Explicit Mathematics.* Habilitationsschrift, University of Bern, 2001.

[41] Strahm, T. Theories with self-application and computational complexity. *Information and Computation 185* (2003), 263–297.

[42] Strahm, T. Weak theories of operations and types. In *Ways of Proof Theory*, R. Schindler, Ed. Ontos Verlag, 2010, pp. 441–468.

[43] Troelstra, A., and van Dalen, D. *Constructivism in Mathematics*, vol. II. North Holland, Amsterdam, 1988.

Bern, March 1, 2012