Unfolding feasible arithmetic and weak truth

Sebastian Eberhard* Thomas Strahm**

August 25, 2012

Abstract

In this paper we continue Feferman's unfolding program initiated in [12] which uses the concept of the unfolding $\mathcal{U}(\mathsf{S})$ of a schematic system S in order to describe those operations, predicates and principles concerning them, which are implicit in the acceptance of S. The program has been carried through for a schematic system of non-finitist arithmetic NFA in Feferman and Strahm [14] and for a system FA (with and without Bar rule) in Feferman and Strahm [15]. The present contribution elucidates the concept of unfolding for a basic schematic system FEA of feasible arithmetic. Apart from the operational unfolding $\mathcal{U}_0(\mathsf{FEA})$ of FEA, we study two full unfolding notions, namely the predicate unfolding $\mathcal{U}(\mathsf{FEA})$ and a more general truth unfolding $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$ of FEA, the latter making use of a truth predicate added to the language of the operational unfolding. The main results obtained are that the provably convergent functions on binary words for all three unfolding systems are precisely those being computable in polynomial time. The upper bound computations make essential use of a specific theory of truth T_{PT} over combinatory logic, which has recently been introduced in Eberhard and Strahm [8] and Eberhard [6] and whose involved proof-theoretic analysis is due to Eberhard [6]. The results of this paper were first announced in [7].

*Institut für Informatik und angewandte Mathematik, Universität Bern, Neubrückstrasse 10, CH-3012 Bern, Switzerland. Email: `eberhard@iam.unibe.ch`. Research supported by the Swiss National Science Foundation.

**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`

1 Introduction

The notion of *unfolding a schematic formal system* was introduced in Feferman [12] in order to answer the following question:

> *Given a schematic system* S*, which operations and predicates, and which principles concerning them, ought to be accepted if one has accepted* S*?*

A paradigmatic example of a schematic system S is the basic system NFA of non-finitist arithmetic. In Feferman and Strahm [14], three unfolding systems for NFA of increasing strength have been analyzed and characterized in more familiar proof-theoretic terms; in particular, it was shown that the full unfolding of NFA, $\mathcal{U}(\mathsf{NFA})$, is proof-theoretically equivalent to predicative analysis. For more information on the path to the unfolding program, especially with regard to predicativity and the implicitness program, see Feferman [13].

More recently, the unfolding notions for a basic schematic system of finitist arithmetic, FA, and for an extension of that by a form BR of the so-called bar rule have been worked out in Feferman and Strahm [15]. It is shown that $\mathcal{U}(\mathsf{FA})$ and $\mathcal{U}(\mathsf{FA} + \mathsf{BR})$ are proof-theoretically equivalent, respectively, to primitive recursive arithmetic, PRA, and to Peano arithmetic, PA.

The aim of the present contribution is to elucidate the concept of unfolding in the context of a natural schematic system FEA for *feasible arithmetic*. We will sketch various unfoldings of FEA and indicate their relationship to weak systems of explicit mathematics and partial truth.

The basic schematic system FEA of feasible arithmetic is based on a language for binary words generated from the empty word by the two binary successors S_0 and S_1; in addition, it includes some natural basic operations on the binary words like, for example, word concatenation and multiplication. The logical operations of FEA are conjunction ($\wedge$), disjunction ($\vee$), and the bounded existential quantifier ($\exists^{\leq}$). FEA is formulated as a system of sequents in this language: apart from the defining axioms for basic operations on words, its heart is a schematically formulated, i.e. open-ended induction rule along the binary words, using a free predicate letter P.

The operational unfolding $\mathcal{U}_0(\mathsf{FEA})$ of FEA extends FEA by a general background theory of combinatory algebra and tells us which operations on words are implicit in the acceptance of FEA. It further includes the generalized substitution rule from Feferman and Strahm [15], which allows arbitrary formulas to be substituted for free predicates in derivable rules of inference such as, for example, the induction rule. We will see that $\mathcal{U}_0(\mathsf{FEA})$ derives the totality of precisely the polynomial time computable functions.

The full predicate unfolding $\mathcal{U}(\mathsf{FEA})$ of FEA tells us, in addition, which predicates and operations on them ought to be accepted if one accepts FEA. It presupposes each logical operation of FEA as an operation on predicates. Predicates themselves are just represented as special operations equipped with an elementhood relation on them. We may further accept the formation of the disjoint union of a (bounded with respect to $\leq$) sequence of predicates given by a corresponding operation. It will turn out that the provably convergent functions of $\mathcal{U}(\mathsf{FEA})$ are still the polynomial time computable ones.

We will also describe an alternative way to define the full unfolding of FEA which makes use of a truth predicate T which mimics the logical operations of FEA in a natural way and makes explicit the requirement that implicit in the acceptance of FEA is the ability to reason about truth in FEA. Using a truth predicate in order to expand a given theory is straightforward and standard approach in the so-called implicitness program: one prominent example is Feferman [11] where the reflective closure of a schematic system is introduced via the famous Feferman-Kripke axioms of partial truth. More recently, in Feferman's original definition of unfolding in [12], a truth predicate is used in order to describe the full unfolding of a schematic system.

The truth unfolding $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$ is obtained by extending the combinatory algebra by a unary truth predicate. Indeed, $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$ contains the predicate unfolding $\mathcal{U}(\mathsf{FEA})$ in a natural way, including the disjoint union operator for predicates. Moreover, the truth unfolding is proof-theoretically equivalent to the predicate unfolding in the sense that its provably convergent functions on the binary words are precisely the polytime functions.

The upper bound computations for both $\mathcal{U}(\mathsf{FEA})$ and $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$ will be obtained via the weak truth theory T_{PT} introduced in Eberhard and Strahm [8] and Eberhard [6], whose analysis and polynomial time upper bound is achieved in Eberhard [6]. The embedding of our two unfolding systems into T_{PT} is rather straightforward, but some special care and additional considerations are needed in order to treat their generalized substitution rules.

We end this introduction by giving a short outline of the paper. In Section 2 we describe in detail the basic schematic formulation of feasible arithmetic FEA. In Section 3 we extend FEA to its operational unfolding $\mathcal{U}_0(\mathsf{FEA})$, thus introducing its underlying abstract theory of operations in the sense of a combinatory algebra and the generalized substitution rule. We will show that the polynomial time computable functions are very naturally and directly proved to be total in $\mathcal{U}_0(\mathsf{FEA})$, hence establishing the lower bound for $\mathcal{U}_0(\mathsf{FEA})$. In Section 4 we turn to the full predicate unfolding $\mathcal{U}(\mathsf{FEA})$ of FEA and Section 5 describes the truth unfolding $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$. The final section of the paper is devoted to the upper bound of $\mathcal{U}(\mathsf{FEA})$ and $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$ via the above-mentioned truth theory T_{PT}.

2 The basic schematic system FEA

In this section we introduce the basic schematic system FEA of feasible arithmetic. Its intended universe of discourse is the set $\mathbb{W} = \{0,1\}^*$ of finite binary words and its basic operations and relations include the binary successors S_0 and S_1, the predecessor Pd, the initial subword relation $\subseteq$, word concatenation $\circledast$ as well as word multiplication $\boxtimes$.[1] The logical operations of FEA are conjunction ($\wedge$), disjunction ($\vee$), and bounded existential quantification ($\exists^{\leq}$). As in the case of finitist arithmetic FA, the statements proved in FEA are sequents of formulas in the given language, i.e. implication is allowed at the outermost level.

2.1 The language of FEA

The language $\mathcal{L}$ of FEA contains a countably infinite supply of variables $\alpha, \beta, \gamma, \ldots$ (possibly with subscripts). These variables are interpreted as ranging over the set of binary words $\mathbb{W}$. $\mathcal{L}$ includes a constant ϵ for the empty word, three unary function symbols $\mathsf{S}_0, \mathsf{S}_1, \mathsf{Pd}$ and three binary func-

[1] Given two words w_1 and w_2, the word $w_1 \boxtimes w_2$ denotes the length of w_2 fold concatenation of w_1 with itself.

tion symbols $\circledast$, $\boxtimes$, $\subseteq$.[2] Terms of $\mathcal{L}$ are defined as usual and are denoted by $\sigma, \tau, \ldots$. Further, $\mathcal{L}$ contains the binary predicate symbol $=$ for equality, and an infinite supply $P_0, P_1, \ldots$ of free predicate letters.

The atomic formulas of $\mathcal{L}$ are of the form $(\sigma = \tau)$ and $P_i(\sigma_1, \ldots, \sigma_n)$ for $i \in \mathbb{N}$. The formulas are closed under $\wedge$ and $\vee$ as well as under bounded existential quantification. In particular, if A is an $\mathcal{L}$ formula, then $(\exists \alpha \leq \tau)A$ is an $\mathcal{L}$ formula as well, where τ is not allowed to contain α. Further, as usual for theories of words, we use $\sigma \leq \tau$ as an abbreviation for $1 \boxtimes \sigma \subseteq 1 \boxtimes \tau$, thus expressing that the length of σ is less than or equal to the length of τ. We use $\bar{\alpha}, \bar{\sigma}$, and $\bar{A}$ to denote finite sequences of variables, terms, and formulas, respectively. Moreover, the notation $\sigma[\bar{\alpha}]$ and $A[\bar{\alpha}]$ is used to indicate a sequence of free variables possibly occurring in a term σ or a formula A; finally, $\sigma[\bar{\tau}]$ and $A[\bar{\tau}]$ are used to denote the result of substitution of $\bar{\alpha}$ by $\bar{\tau}$ is those expressions.

2.2 Axioms and rules of FEA

FEA is formulated as a system of sequents Σ of the form $\Gamma \to A$, where Γ is a finite sequence of $\mathcal{L}$ formulas and A is an $\mathcal{L}$ formula. Hence, we have the usual Gentzen-type logical axioms and rules of inference for our underlying restricted language. In particular, the bounded existential quantifier is governed by the following rules of inference, where the usual variable conditions apply:

$$\text{(E1)} \qquad \frac{\Gamma \to \sigma \leq \tau \wedge A[\sigma]}{\Gamma \to (\exists \beta \leq \tau)A[\beta]}$$

$$\text{(E2)} \qquad \frac{\Gamma, \alpha \leq \tau, A[\alpha] \to B}{\Gamma, (\exists \beta \leq \tau)A[\beta] \to B}$$

Further, in our restricted logical setting, we adopt the following rule of term substitution:

$$\text{(S0)} \qquad \frac{\Gamma[\alpha] \to A[\alpha]}{\Gamma[\tau] \to A[\tau]}$$

[2]We assume that $\subseteq$ defines the characteristic function of the initial subword relation. Further, we employ infix notation for these binary function symbols.

The non-logical axioms of FEA state the usual defining equations for the function symbols of the language $\mathcal{L}$, see, e.g., Ferreria [16] for similar axioms. Finally, we have the schematic induction rule formulated for a free predicate P as follows:

$$(\mathsf{Ind}) \qquad \frac{\Gamma \to P(\epsilon) \qquad \Gamma, P(\alpha) \to P(\mathsf{S}_i(\alpha)) \quad (i = 0, 1)}{\Gamma \to P(\alpha)}$$

In the various unfolding systems of FEA introduced below, we will be able to substitute an arbitrary formula for the free predicate letter P.

3 The operational unfolding $\mathcal{U}_0(\mathsf{FEA})$

In this section we are going to introduce the *operational unfolding* $\mathcal{U}_0(\mathsf{FEA})$ of FEA. It tells us which operations from and to individuals, and which principles concerning them, ought to be accepted if one has accepted FEA.

In the operational unfolding, we make these commitments explicit by extending FEA by a partial combinatory algebra. Since it represents any recursion principle and thus any recursive function by suitable terms, it is expressive enough to reflect any ontological commitment we want to reason about. Using the notion of *provable totality*, we single out those functions and recursion principles we are actually committed to by accepting FEA.

Let us explain some properties of the operations we use in the above mentioned extension of FEA. We employ a general notion of (partial) operation, belonging to a universe V including the universe of discourse of FEA. Operations are not bound to any specific mathematical domain, but have to be considered as pre-mathematical in nature. Operations can apply to other operations. Some operations are universal and are naturally self-applicable as a result, like the identity operation or the pairing operation, while some are partial and presented to us on the binary words only. Operations satisfy the laws of a partial combinatory algebra with pairing, projections, and definition by cases.

3.1 The language $\mathcal{L}_1$

The language $\mathcal{L}_1$ is an expansion of the language $\mathcal{L}$ including new constants k, s, π, p_0, p_1, d, tt, ff, e, ϵ, s_0, s_1, pd, $\mathsf{c}_\subseteq$, $*$, $\times$, and an additional countably infinite set of variables $x_0, x_1, \ldots$.[3] The new variables are supposed to range over the universe of operations and are usually denoted by $a, b, c, x, y, z, \ldots$. The $\mathcal{L}_1$ terms $(r, s, t, \ldots)$ are inductively generated from variables and constants of $\mathcal{L}$ and $\mathcal{L}_1$ by means of the function symbols of FEA and the application operator $\cdot$. We use the usual abbreviations for applicative terms and abbreviate $s \cdot t$ as (st), st or $s(t)$ as long as no confusion arises. We further adopt the convention of association to the left so that $s_0 s_1 \cdots s_n$ stands for $(\cdots(s_0 s_1) \cdots s_n)$; we sometimes write $s(t_0, \ldots, t_n)$ for $st_0 \cdots t_n$. We have $(s = t)$, $s\downarrow$ and $P_i(\bar{s})$ for $i \in \mathbb{N}$ as atoms of $\mathcal{L}_1$. The formula $s\downarrow$ is interpreted as definedness of s. The formulas $(A, B, C, \ldots)$ are built from the atoms as before using $\vee, \wedge$ and the bounded existential quantifier, where as above the bounding term is a term of $\mathcal{L}$ not containing the bound variable.

For s a term of $\mathcal{L}_1 \setminus \mathcal{L}$ we write $s \leq \tau$ for $(\exists \beta \leq \tau)(s = \beta)$. We use the pairing operator π to introduce n-tupling $\langle t_1, \ldots, t_n \rangle$ of terms as usual.

3.2 Axioms and rules of $\mathcal{U}_0(\mathsf{FEA})$

The operational unfolding $\mathcal{U}_0(\mathsf{FEA})$ is formulated as a system of sequents $\Gamma \to A$ of formulas in the language $\mathcal{L}_1$. $\emptyset \to A$ will just be displayed as A. Apart from the axioms for FEA, $\mathcal{U}_0(\mathsf{FEA})$ comprises the following axioms and rules of inference.

I. Applicative counterpart of the initial functions.

(1) $\mathsf{s}_i \alpha = \mathsf{S}_i(\alpha), \quad \mathsf{pd}\alpha = \mathsf{Pd}(\alpha)$,

(2) $* \alpha\beta = \alpha \circledast \beta, \quad \times \alpha\beta = \alpha \boxtimes \beta, \quad \mathsf{c}_{\subseteq} \alpha\beta = \alpha \subseteq \beta$.

II. Partial combinatory algebra, pairing, definition by cases.

(3) $\mathsf{k}ab = a$,

(4) $\mathsf{s}ab\downarrow, \quad \mathsf{s}abc \simeq ac(bc)$,

(5) $\mathsf{p}_0 \langle a, b \rangle = a, \quad \mathsf{p}_1 \langle a, b \rangle = b$,

(6) $\mathsf{d}ab\,\mathsf{tt} = a, \quad \mathsf{d}ab\,\mathsf{ff} = b$.

[3] These variables are syntactically different from the $\mathcal{L}$ variables $\alpha_0, \alpha_1, \ldots$.

III. Equality on the binary words.

(7) $\mathsf{e}\alpha\beta = \mathtt{tt} \vee \mathsf{e}\alpha\beta = \mathtt{ff}$,

(8) $\mathsf{e}\alpha\beta = \mathtt{tt} \leftrightarrow \alpha = \beta$.[4]

The operational unfolding of FEA includes the rules of inference of FEA (extended to the new language). In addition, in analogy to the rule (S0), we have the following new substitution rule for terms of $\mathcal{L}_1$:

(S1)
$$\frac{\Gamma[u] \to A[u]}{\Gamma[t], t\downarrow \to A[t]}$$

The next useful substitution rule (S2) can be derived easily from the other axioms and rules. It tells us that bounded terms can be substituted for word variables:[5]

(S2)
$$\frac{\Gamma[\alpha] \to A[\alpha] \quad \Gamma[t] \to t \leq \tau}{\Gamma[t] \to A[t]}$$

Finally, $\mathcal{U}_0$(FEA) includes the generalized substitution rule for derived rules of inference as it is developed in Feferman and Strahm [15]. Towards a more compact notation, let use write $\Sigma_1, \Sigma_2, \ldots, \Sigma_n \Rightarrow \Sigma$ to denote a rule of inference with premises $\Sigma_1, \ldots, \Sigma_n$ and conclusion Σ. We let $A[\bar{B}/\bar{P}]$ denote the formula $A[\bar{P}]$ with each subformula $P_i(\bar{t})$ replaced by $\bar{t}\downarrow \wedge B_i[\bar{t}]$, where the length of $\bar{t}$ equals the arity of P_i. The generalized substitution rule (S3) can now be described as follows: Assume that the rule of inference $\Sigma_1, \Sigma_2, \ldots, \Sigma_n \Rightarrow \Sigma$ is derivable from the axioms and rules at hand. Then we can adjoin an arbitrary substitution instance

(S3)
$$\Sigma_1[\bar{B}/\bar{P}], \ldots, \Sigma_n[\bar{B}/\bar{P}] \Rightarrow \Sigma[\bar{B}/\bar{P}]$$

as new rule of inference to our system. Here $\bar{P}$ and $\bar{B}$ are finite sequences of free predicates and $\mathcal{L}_1$ formulas, respectively. Note that the notion of *derivability of a rule of inference* is dynamic as one unfolds a given system.

[4] To be precise, this equivalence is a shorthand for the two sequents $\mathsf{e}\alpha\beta = \mathtt{tt} \to \alpha = \beta$ and $\alpha = \beta \to \mathsf{e}\alpha\beta = \mathtt{tt}$.

[5] Note that for an $A[\alpha]$ with α occurring in a bound and a term $t \in \mathcal{L}_1 \setminus \mathcal{L}$, the rule (S2) cannot be derived because then $A[t]$ is not a formula.

Clearly, using the generalized substitution rule, the induction rule in its usual form can be derived for an arbitrary $A \in \mathcal{L}_1$:

$$\frac{\Gamma \to A[\epsilon] \qquad \Gamma, A[\alpha] \to A[\mathsf{S}_i(\alpha)] \quad (i = 0, 1)}{\Gamma \to A[\alpha]}$$

Moreover, the usual substitution rule for sequents, $\Sigma[\bar{P}] \Rightarrow \Sigma[\bar{B}/\bar{P}]$ can be obtained as an admissible rule of inference. This ends the description of the operational unfolding $\mathcal{U}_0(\mathsf{FEA})$ of FEA.

Next we want to show that the polynomial time computable functions can be proved to be total in $\mathcal{U}_0(\mathsf{FEA})$. We call a function $F : \mathbb{W}^n \to \mathbb{W}$ provably total in a given axiomatic system whose language includes $\mathcal{L}_1$, if there exists a closed $\mathcal{L}_1$ term t_F such that (i) t_F defines F pointwise, i.e. on each standard word, and, moreover, (ii) there is a $\mathcal{L}$ term $\tau[\alpha_1, \ldots, \alpha_n]$ such that the assertion

$$t_F(\alpha_1, \ldots, \alpha_n) \leq \tau[\alpha_1, \ldots, \alpha_n]$$

is provable in the underlying system. Thus, in a nutshell, F is provably total iff it is provably and uniformly bounded.

Lemma 1 *The polynomial time computable functions are provably total in the operational unfolding* $\mathcal{U}_0(\mathsf{FEA})$.

Proof. We use Cobham's characterization of the polynomial time computable functions (cf. [5, 4]): starting off from the initial functions of $\mathcal{L}$ and arbitrary projections, the polynomial time computable functions can be generated by closing under composition and bounded recursion. First of all, the initial functions of $\mathcal{L}$ and projections represented using lambda abstraction are obviously total. Closure of the provably total functions under composition is established by making use of the substitution rules (S1) and (S2) as well as the fact that the $\mathcal{L}$ functions are provably monotone. In order to show closure under bounded recursion, assume that F is defined by bounded recursion with initial function G and step function H, where τ is the corresponding bounding polynomial.[6] By the induction hypothesis, G and H are provably total via suitable $\mathcal{L}_1$ terms t_G and t_H. Using the recursion or fixed point

[6] We can assume that only functions built from concatenation and multiplication are permissible bounds for the recursion.

theorem of the partial combinatory algebra, we find an $\mathcal{L}_1$ term t_F which provably in $\mathcal{U}_0(\mathsf{FEA})$ satisfies the following recursion equations for $i = 0, 1$:

$$\begin{aligned} t_F(\bar{\alpha}, \epsilon) &\simeq t_G(\bar{\alpha}) \mid \tau[\bar{\alpha}, \epsilon], \\ t_F(\bar{\alpha}, \mathsf{s}_i(\beta)) &\simeq t_H(t_F(\bar{\alpha}, \beta), \bar{\alpha}, \beta) \mid \tau[\bar{\alpha}, \mathsf{s}_i(\beta)] \end{aligned}$$

Here $\mid$ is the usual truncation operation such that $\alpha|\beta$ is α if $\alpha \leq \beta$ and β otherwise. Now fix $\bar{\alpha}$ and let $A[\beta]$ be the formula $t_F(\bar{\alpha}, \beta) \leq \tau[\bar{\alpha}, \beta]$ [7] and simply show $A[\beta]$ by induction on β. Thus F is provably total in $\mathcal{U}_0(\mathsf{FEA})$ which concludes the proof of the lower bound lemma. $\square$

4 The full predicate unfolding $\mathcal{U}(\mathsf{FEA})$

In this section we will define the full predicate unfolding $\mathcal{U}(\mathsf{FEA})$ of FEA. It tells us, in addition, which predicates and operations on predicates ought to be accepted if one has accepted FEA. By accepting $\mathcal{U}_0(\mathsf{FEA})$ one implicitly accepts an equality predicate and operations on predicates corresponding to the logical operations of $\mathcal{U}_0(\mathsf{FEA})$. Finally, we may accept the principle of forming the predicate for the disjoint union of a (bounded) sequence of predicates given by an operation.

As before the equality predicate and the above-mentioned operations will be given as elements of an underlying combinatory algebra which is extended by a binary relation $\in$ for elementship, so predicates are represented via classifications in the sense of Feferman's explicit mathematics [9, 10]. We additionally use a relation Π to single out the operations representing predicates one is committed to by accepting FEA.

The language $\mathcal{L}_2$ of $\mathcal{U}(\mathsf{FEA})$ is an extension of $\mathcal{L}_1$ by new individual constants id (identity), inv (inverse image), con (conjunction), dis (disjunction), leq (bounded existential quantifier), and j (bounded disjoint unions); further new constants are $\pi_0, \pi_1, \ldots$ which are combinatorial representations of free predicates. Finally, $\mathcal{L}_2$ has a new unary relation symbol Π in order to single out the predicates we are committed to as well as a binary relation symbol $\in$ for elementhood of individuals in predicates. The terms of $\mathcal{L}_2$ are generated

[7] Recall that by expanding the definition of the $\leq$ relation, the formula $A[\beta]$ stands for the assertion $(\exists \gamma \leq \tau[\bar{\alpha}, \beta])(t_F(\bar{\alpha}, \beta) = \gamma)$.

as before but now taking into account the new constants. The formulas of $\mathcal{L}_2$ extend the formulas of $\mathcal{L}_1$ by allowing new atomic formulas of the form $\Pi(t)$ and $s \in t$.

The axioms of $\mathcal{U}(\mathsf{FEA})$ extend those of $\mathcal{U}_0(\mathsf{FEA})$ by the following axioms about predicates.

I. Identity predicate

(1) $\Pi(\mathsf{id})$,

(2) $x \in \mathsf{id} \to \mathsf{p}_0 x = \mathsf{p}_1 x \wedge x = \langle \mathsf{p}_0 x, \mathsf{p}_1 x \rangle$,

(3) $\mathsf{p}_0 x = \mathsf{p}_1 x, x = \langle \mathsf{p}_0 x, \mathsf{p}_1 x \rangle \to x \in \mathsf{id}$.

II. Inverse image predicates

(4) $\Pi(a) \to \Pi(\mathsf{inv}(f, a))$,

(5) $\Pi(a), x \in \mathsf{inv}(f, a) \to fx \in a$,

(6) $\Pi(a), fx \in a \to x \in \mathsf{inv}(f, a)$.

III. Conjunction and disjunction

(7) $\Pi(a), \Pi(b) \to \Pi(\mathsf{con}(a, b))$,

(8) $\Pi(a), \Pi(b), x \in \mathsf{con}(a, b) \to x \in a \wedge x \in b$,

(9) $\Pi(a), \Pi(b), x \in a, x \in b \to x \in \mathsf{con}(a, b)$,

(10) $\Pi(a), \Pi(b) \to \Pi(\mathsf{dis}(a, b))$,

(11) $\Pi(a), \Pi(b), x \in \mathsf{dis}(a, b) \to x \in a \vee x \in b$,

(12) $\Pi(a), \Pi(b), x \in a \vee x \in b \to x \in \mathsf{dis}(a, b)$.

IV. Bounded existential quantification

(13) $\Pi(a) \to \Pi(\mathsf{leq} a)$,

(14) $\Pi(a), \langle y, \alpha \rangle \in \mathsf{leq}(a) \to (\exists \beta \leq \alpha)(\langle y, \beta \rangle \in a)$,

(15) $\Pi(a), (\exists \beta \leq \alpha)(\langle y, \beta \rangle \in a) \to \langle y, \alpha \rangle \in \mathsf{leq}(a)$.

V. Free predicates

(16) $\Pi(\pi_i)$,

(17) $\langle \bar{x} \rangle \in \pi_i \to P_i(\bar{x}), \quad P_i(\bar{x}) \to \langle \bar{x} \rangle \in \pi_i.$

Further, the full unfolding $\mathcal{U}(\mathsf{FEA})$ includes axioms stating that a bounded sequence of predicates determines the predicate of the disjoint union of this sequence. We use the following three rules to axiomatize the join predicates in our restricted logical setting.

VI. Join rules [8]

$$\frac{\Gamma, \beta \leq \alpha \to \Pi(f\beta)}{\Gamma \to \Pi(\mathsf{j}(f, \alpha))} \tag{18}$$

$$\frac{\Gamma, \beta \leq \alpha \to \Pi(f\beta)}{\Gamma, x \in \mathsf{j}(f, \alpha) \to x = \langle \mathsf{p}_0 x, \mathsf{p}_1 x \rangle \wedge \mathsf{p}_0 x \leq \alpha \wedge \mathsf{p}_1 x \in f(\mathsf{p}_0 x)} \tag{19}$$

$$\frac{\Gamma, \beta \leq \alpha \to \Pi(f\beta)}{\Gamma, x = \langle \mathsf{p}_0 x, \mathsf{p}_1 x \rangle, \mathsf{p}_0 x \leq \alpha, \mathsf{p}_1 x \in f(\mathsf{p}_0 x) \to x \in \mathsf{j}(f, \alpha)} \tag{20}$$

The rules of inference of $\mathcal{U}_0(\mathsf{FEA})$ are also available in $\mathcal{U}(\mathsf{FEA})$. In particular, $\mathcal{U}(\mathsf{FEA})$ contains the generalized substitution rule (S3): the formulas $\bar{B}$ to be substituted for $\bar{P}$ are now in the language of $\mathcal{L}_2$; the rule in the premise of (S3), however, is required to be in the language $\mathcal{L}_1$.[9] This concludes the description of the predicate unfolding $\mathcal{U}(\mathsf{FEA})$ of FEA.

5 The truth unfolding $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$

In this section we describe an alternative way to define the full unfolding of FEA. The truth unfolding $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$ of FEA makes use of a truth predicate T which reflects the logical operations of FEA in a natural and direct way. We will see that the full predicate unfolding $\mathcal{U}(\mathsf{FEA})$ is directly contained in $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$.

As in the last section, we want to make the commitment to the logical operations of FEA explicit. This is done by introducing a truth predicate for

[8] In the formulation of these rules, it is assumed that β does not occur in Γ.

[9] This last restriction is imposed since predicates may depend on $\bar{P}$.

which truth biconditionals defining the truth conditions of the logical operations hold. The axiomatization of the truth predicate relies on a coding mechanism for formulas. In the applicative framework, this is achieved in a very natural way by using new constants designating the logical operations of FEA. The language $\mathcal{L}_\mathsf{T}$ of $\mathcal{U}_\mathsf{T}(\mathsf{FEA})$ extends $\mathcal{L}_1$ by new individual constants $\dot{=}$, $\dot{\wedge}$, $\dot{\vee}$, $\dot{\exists}$, as well as constants $\pi_0, \pi_1, \ldots$. In addition, $\mathcal{L}_\mathsf{T}$ includes a new unary relation symbol T. The terms and formulas of $\mathcal{L}_\mathsf{T}$ are defined in the expected manner. Moreover, we will use infix notation for $\dot{=}, \dot{\wedge}$ and $\dot{\vee}$.

The axioms of $\mathcal{U}_\mathsf{T}(\mathsf{FEA})$ extend those of $\mathcal{U}_0(\mathsf{FEA})$ by the following axioms about the truth predicate T:

$$
\begin{array}{llcl}
(\dot{=}) & \mathsf{T}(x \mathrel{\dot{=}} y) & \leftrightarrow & x = y \\
(\dot{\wedge}) & \mathsf{T}(x \mathrel{\dot{\wedge}} y) & \leftrightarrow & \mathsf{T}(x) \wedge \mathsf{T}(y) \\
(\dot{\vee}) & \mathsf{T}(x \mathrel{\dot{\vee}} y) & \leftrightarrow & \mathsf{T}(x) \vee \mathsf{T}(y) \\
(\dot{\exists}) & \mathsf{T}(\dot{\exists}\alpha x) & \leftrightarrow & (\exists \beta \leq \alpha)\mathsf{T}(x\beta) \\
(\pi_i) & \mathsf{T}(\pi_i(\bar{x})) & \leftrightarrow & P_i(\bar{x})
\end{array}
$$

It is easy and natural to assign $\mathcal{L}_\mathsf{T}$ terms to $\mathcal{L}_\mathsf{T}$ formulas in the following way.

Definition 2 *For each formula A of $\mathcal{L}_\mathsf{T}$ we inductively define an $\mathcal{L}_\mathsf{T}$ term $[A]$ whose free variables are exactly the free variables of A:*

$$
\begin{array}{rcl}
[t = s] & := & t \mathrel{\dot{=}} s \\
[P_i(\bar{t})] & := & \pi_i(\bar{t}) \\
[\mathsf{T}(t)] & := & t \\
[A \wedge B] & := & [A] \mathrel{\dot{\wedge}} [B] \\
[A \vee B] & := & [A] \mathrel{\dot{\vee}} [B] \\
[(\exists \alpha \leq \tau) A[\alpha]] & := & \dot{\exists}\tau(\lambda\alpha.[A[\alpha]])
\end{array}
$$

The following lemma can be proved by a trivial induction on the complexity of formulas.

Lemma 3 (Tarski biconditionals) *Let A be a $\mathcal{L}_\mathsf{T}$ formula. Then we have*

$$\mathcal{U}_\mathsf{T}(\mathsf{FEA}) \vdash A \leftrightarrow \mathsf{T}([A])$$

This lemma shows that in our weak setting, full Tarski biconditionals can be achieved without having to type the truth predicate. Of course, this is due to the fact that negation is only present at the level of sequents.

We close this section by noting that the generalized substitution rule ($\mathsf{S3}$) can be stated in a somewhat more general form for $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$. Recall that in $\mathcal{U}(\mathsf{FEA})$, the rule in the premise of ($\mathsf{S3}$) is required to be in $\mathcal{L}_1$. Due to the fact that each $\mathcal{L}_{\mathsf{T}}$ formula can be represented by a term, we can allow rules in $\mathcal{L}_{\mathsf{T}}$ in the premise of the generalized substitution rule, as long as we substitute formulas and associated terms for the predicates P_i and constants π_i simultaneously. In the following we denote by $\Sigma[\overline{B}/\overline{P};\overline{t}/\overline{\pi}]$ the simultaneous substitution of the predicates $\overline{P}$ by the formulas $\overline{B}$ and of the constants $\overline{\pi}$ by the $\mathcal{L}_{\mathsf{T}}$ terms $\overline{t}$. The generalized substitution rule for $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$ can now be stated as follows. Assume that the rule $\Sigma_1,\dots,\Sigma_n \Rightarrow \Sigma$ is derivable with the axioms and rules at hand. Assume further that the terms $\overline{t_B}$ correspond to the $\mathcal{L}_{\mathsf{T}}$ formulas $\overline{B}$ according to the lemma above. Then we can adjoin the rule

$$\Sigma_1[\overline{B}/\overline{P};\overline{t_B}/\overline{\pi}],\dots,\Sigma_n[\overline{B}/\overline{P};\overline{t_B}/\overline{\pi}] \Rightarrow \Sigma[\overline{B}/\overline{P};\overline{t_B}/\overline{\pi}]$$

as a new rule of inference to our unfolding system $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$. This concludes the description of $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$.

It is easy to see that the full predicate unfolding $\mathcal{U}(\mathsf{FEA})$ is contained in the truth unfolding $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$. The argument proceeds along the same line as the embedding of weak explicit mathematics into theories of truth in Eberhard and Strahm [8], which will also be described in some detail in the next section.

6 Proof-theoretical analysis

In this section we will find a suitable upper bound for $\mathcal{U}(\mathsf{FEA})$ and $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$ thus showing that their provably total functions are indeed computable in polynomial time. We will obtain the upper bound via the weak truth theory T_{PT} introduced in Eberhard and Strahm [8] and Eberhard [6], whose detailed and very involved proof-theoretic analysis is carried out in [6]. To be precise, we consider a slight (conservative) extension of T_{PT} which facilitates the treatment of the generalized substitution rule.

Let us sketch the theory T_{PT}; for a more detailed description, the reader is referred to [8, 6]. For a more extensive survey on similar kinds of theories in a stronger setting, see Cantini [1] and Kahle [17]. T_{PT} is based on a total version of the basic applicative theory B for words which was developed in Strahm [18]. In particular, we have a word predicate W which is interpreted as the type of binary strings, constants for some simple functions on the words and a computationally complete combinatory algebra. T_{PT} contains, in addition, a unary truth predicate T which formalizes a compositional truth predicate, where we have constants for the basic logical operations as in the case of the truth unfolding above. The axioms for this predicate T are as usual for theories of truth over an applicative setting with the exception of the axiom for the word predicate. Only *bounded* elementship in the words can be reflected by T, thus the low proof theoretic strength of T_{PT}.[10] In the axioms below, $y \leq_{\mathsf{W}} x$ is short for $y \leq x \wedge y \in \mathsf{W}$. The truth axioms now read as follows:

$$\begin{array}{llcl}
(\dot{=}) & \mathsf{T}(x \doteq y) & \leftrightarrow & x = y \\
(\dot{\mathsf{W}}) & x \in \mathsf{W} \to (\mathsf{T}(\dot{\mathsf{W}}xy) & \leftrightarrow & y \leq_{\mathsf{W}} x) \\
(\dot{\wedge}) & \mathsf{T}(x \mathbin{\dot{\wedge}} y) & \leftrightarrow & \mathsf{T}(x) \wedge \mathsf{T}(y) \\
(\dot{\vee}) & \mathsf{T}(x \mathbin{\dot{\vee}} y) & \leftrightarrow & \mathsf{T}(x) \vee \mathsf{T}(y) \\
(\dot{\forall}) & \mathsf{T}(\dot{\forall} f) & \leftrightarrow & (\forall x)\mathsf{T}(fx) \\
(\dot{\exists}) & \mathsf{T}(\dot{\exists} f) & \leftrightarrow & (\exists x)\mathsf{T}(fx)
\end{array}$$

In addition, T_{PT} contains unrestricted truth induction on the binary words:

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

It is shown in Eberhard [6] that the provably total operations of T_{PT} are precisely the polynomial time computable functions.[11] Moreover, T_{PT} proves

[10] We note that T_{PT} can be seen as a polynomial time analogue of a theory of truth of primitive recursive strength studied in Cantini [2, 3].

[11] As usual for applicative systems, we call a function $F : \mathbb{W}^n \to \mathbb{W}$ provably total in T_{PT}, if there exists a closed term t_F such that (i) t_F defines F pointwise, i.e. on each standard word, and, moreover, (ii) the following assertion is provable in T_{PT}:

$$x_1 \in \mathsf{W}, \ldots, x_n \in \mathsf{W} \to t_F(x_1, \ldots, x_n) \in \mathsf{W}$$

the Tarski biconditionals for formulas that contain only bounded occurrences of the word predicate, e.g. formulas of the form $s \leq_{\mathsf{W}} t$. The corresponding terms for such formulas A are denoted by $\ulcorner A \urcorner$, see [8, 6] for details.

In order to deal with the generalized substitution rule below, we will consider a slight extension $\mathsf{T}^*_{\mathsf{PT}}$ of T_{PT}. Its language extends the one of T_{PT} by the predicates $P_0, P_1, \dots$ and the constants $\pi_0, \pi_1, \dots$. It contains the additional axiom $\mathsf{T}(\pi_i \bar{x}) \leftrightarrow P_i(\bar{x})$ for every $i \in \mathbb{N}$. Since no other axioms for the P predicates and the π constants are present, $\mathsf{T}^*_{\mathsf{PT}}$ is clearly a conservative extension of T_{PT}.

Next we describe a direct embedding of $\mathcal{U}(\mathsf{FEA})$ into $\mathsf{T}^*_{\mathsf{PT}}$ which resembles a standard embedding of a theory of explicit mathematics into a theory of truth: We translate the elementship relation with help of the truth predicate and the type constructors by formulating their elementhood condition as in Eberhard and Strahm [8]. Nevertheless, we have to consider some peculiarities of our system: we take special care of the FEA function constants which are not present in the language of T_{PT} and map the two kinds of variables to disjoint sets of T_{PT} variables.

Definition 4 (Translation * of $\mathcal{L}_2$ terms) *The translation of $\mathcal{L}_2$ terms is given inductively on their complexity.*

- *Let c be an applicative constant. Then $c^* \equiv c$.*
- *Let α_i be an $\mathcal{L}$ variable. Then $\alpha_i^* \equiv x_{2i}$.*
- *Let x_i be a variable of $\mathcal{L}_1 \setminus \mathcal{L}$. Then $x_i^* \equiv x_{2i+1}$.*
- $\mathsf{leq}^* \equiv \lambda a.\lambda z.z \doteq \langle \mathsf{p_0} z, \mathsf{p_1} z\rangle \mathbin{\dot\wedge} \dot\exists \lambda y.\dot{\mathsf{W}}(\mathsf{p_1} z) y \mathbin{\dot\wedge} a\langle \mathsf{p_0} z, y\rangle$
- $\mathsf{id}^* \equiv \lambda z.z = \langle \mathsf{p_0} z, \mathsf{p_1} z\rangle \mathbin{\dot\wedge} \mathsf{p_0} z \doteq \mathsf{p_1} z$
- $\mathsf{con}^* \equiv \lambda a.\lambda b.\lambda z.\ az \mathbin{\dot\wedge} bz$
- $\mathsf{dis}^* \equiv \lambda a.\lambda b.\lambda z.\ az \mathbin{\dot\vee} bz$
- $\mathsf{inv}^* \equiv \lambda f.\lambda a.\lambda z.\ a(fz)$
- $\mathsf{j}^* \equiv \lambda f.\lambda a.\lambda z.z = \langle \mathsf{p_0} z, \mathsf{p_1} z\rangle \mathbin{\dot\wedge} \dot{\mathsf{W}} a(\mathsf{p_0} z) \mathbin{\dot\wedge} f(\mathsf{p_0} z)(\mathsf{p_1} z)$
- $\pi_i^* \equiv \pi_i$

- *Let t be $s_0 s_1$. Then $t^* \equiv s_0^* s_1^*$.*
- *Let G be an n-ary $\mathcal{L}$ function symbol, g_{App} its applicative analogue, and $\overline{t}$ a sequence of terms of suitable arity. Then $G(\overline{t})^* \equiv g_{App}\overline{t^*}$.*

For the translation of $\mathcal{L}_2$ formulas, we interpret elementship using the truth predicate as usual and trivialize the relation Π.

Definition 5 (Translation * of $\mathcal{L}_2$ formulas) *The translation of $\mathcal{L}_2$ formulas is given inductively on their complexity.*

- $\Pi(s)^* \equiv 0 = 0$
- $(s = t)^* \equiv s^* = t^*$
- $(s \in t)^* \equiv \mathsf{T}(t^* s^*)$
- $(\exists \alpha \leq \tau)A[\alpha]^* \equiv (\exists \alpha^* \leq_{\mathsf{W}} \tau^*)A^*[\alpha^*]$
- *The translation commutes with the connectives $\wedge$ and $\vee$.*

The translation * is extended in the obvious way to sequences and sequents of $\mathcal{L}_2$ formulas. Further, for the statement of the embedding theorem below, the following notation is handy.

Definition 6 *Let $\Diamond$ be an $\mathcal{L}_2$ term, formula or sequence of formulas. Then $\overline{y}(\Diamond) \in \mathsf{W}$ denotes the sequence $x_{n_0} \in \mathsf{W}, \ldots, x_{n_m} \in \mathsf{W}$ where the x_{n_i} enumerate the variables with even subscripts occurring freely in $\Diamond^*$.*

The next two lemmas will be used in the proof of the embedding theorem below. Lemma 7 can be proved by a trivial induction on the complexity of the FEA term. Lemma 8 can be proved by induction on the complexity of A. For the case where A is of the form $(\exists \alpha \leq \tau)B[\alpha]$, we use lemma 7.

Lemma 7 *Let τ be an $\mathcal{L}$ term. Then we have*

$$\mathsf{T}^*_{\mathsf{PT}} \vdash \overline{y}(\tau) \in \mathsf{W} \rightarrow \tau^* \in \mathsf{W}.$$

Lemma 8 *Let A be an $\mathcal{L}_2$ formula. Then we have*

$$\mathsf{T}^*_{\mathsf{PT}} \vdash \overline{y}(A) \in \mathsf{W} \rightarrow \mathsf{T}(\ulcorner A^* \urcorner) \leftrightarrow A^*.$$

We are now ready to state the main embedding lemma of $\mathcal{U}(\mathsf{FEA})$ into $\mathsf{T}^*_{\mathsf{PT}}$ and sketch its proof.

Lemma 9 (Embedding lemma) *Assume* $\mathcal{U}(\mathsf{FEA}) \vdash \Gamma \to A$. *Then we have*

$$\mathsf{T}^*_{\mathsf{PT}} \vdash \overline{y}(\Gamma, A) \in \mathsf{W}, \Gamma^* \to A^*.$$

Proof.(Sketch) In order to prove the lemma, one shows a stronger assertion, namely that the * translation of each derivable rule of $\mathcal{U}(\mathsf{FEA})$ is also derivable in $\mathsf{T}^*_{\mathsf{PT}}$. Let us exemplary discuss some crucial examples. First, let us look at * translations of axioms of $\mathcal{U}(\mathsf{FEA})$ and distinguish the following cases:

(i) The translations of the axioms about the (word) function symbols of $\mathcal{L}$ hold, because the $\mathcal{L}$ variables are assumed to range over words;

(ii) The translations of the axioms about the applicative combinators clearly hold;

(iii) The translations of the axioms about the correspondence between the function symbols and the applicative constants follow directly from the definition of the translation;

(iv) The translations of the axioms about the predicate constructors hold because of their translation by suitable elementhood conditions and because of the trivial interpretation of the relation Π;

(v) The translations of the axioms $P_i(\bar{x}) \leftrightarrow \langle \bar{x} \rangle \in \pi_i$ clearly hold.

Towards the treatment of the generalized substitution rule, assume that the rule with premises $\Gamma_i[\overline{P}] \to A_i[\overline{P}]$ for $1 \leq i \leq m$ and conclusion $\Gamma[\overline{P}] \to A[\overline{P}]$ is derivable in $\mathcal{U}(\mathsf{FEA})$. Let us look at the * translation of the proof for derivability which is a proof of derivability in $\mathsf{T}^*_{\mathsf{PT}}$ by induction hypothesis. It can be easily seen that for each sequence of formulas $\overline{B} \in \mathcal{L}_2$ we still have a proof if we replace each occurrence of P_i by B_i^* and each occurrence of π_i by $[B_i^*]$ and add $\overline{y}(\overline{B}) \in \mathsf{W}$ to each antecedent. Here, we use lemma 8 to justify induction and the substituted P biconditionals. Thus the * translation of the rule with conclusion $\Gamma[\overline{B}] \to A[\overline{B}]$ and premises $\Gamma_i[\overline{B}] \to A_i[\overline{B}]$ for $1 \leq i \leq m$ is derivable in $\mathsf{T}^*_{\mathsf{PT}}$ as desired. This ends the treatment of the generalized substitution rule and hence the proof sketch of the embedding lemma. □

The embedding lemma immediately implies that each function which is provably total in $\mathcal{U}(\mathsf{FEA})$ is also provably total in $\mathsf{T}^*_{\mathsf{PT}}$ in the usual sense. Since $\mathsf{T}^*_{\mathsf{PT}}$ is conservative over T_{PT} and the latter proves totality exactly for the polynomial time computable functions (cf. Eberhard [6]) this delivers the desired upper bound for the unfoldings $\mathcal{U}_0(\mathsf{FEA})$ and $\mathcal{U}(\mathsf{FEA})$. Together with Lemma 1, we obtain sharp proof theoretic bounds.

Theorem 10 *The provably total functions of $\mathcal{U}_0(\mathsf{FEA})$ and $\mathcal{U}(\mathsf{FEA})$ are exactly the polynomial time computable functions.*

An embedding of $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$ into $\mathsf{T}^*_{\mathsf{PT}}$ can be found in a very similar way as for $\mathcal{U}(\mathsf{FEA})$. Just interpret the constants $\dot{=}$, $\dot{\wedge}$ and $\dot{\vee}$ as themselves and $\dot{\exists}$ as $\lambda y.\lambda z.\dot{\exists}\lambda x.\dot{\mathsf{W}}yx \mathbin{\dot{\wedge}} zx$. Thus we obtain the following theorem.

Theorem 11 *The provably total functions of $\mathcal{U}_{\mathsf{T}}(\mathsf{FEA})$ are exactly the polynomial time computable functions.*

This concludes the computation of the upper bounds and hence the proof-theoretic analysis of our various unfolding systems.

References

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

[2] 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.

[3] 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.

[4] Clote, P. Computation models and function algebras. In *Handbook of Computability Theory*, E. Griffor, Ed. Elsevier, 1999, pp. 589–681.

[5] Cobham, A. The intrinsic computational difficulty of functions. In *Logic, Methodology and Philosophy of Science II*. North Holland, Amsterdam, 1965, pp. 24–30.

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

[7] Eberhard, S., and Strahm, T. Towards the unfolding of feasible arithmetic (Abstract). *Bulletin of Symbolic Logic 18*, 3 (2012), 474–475.

[8] Eberhard, S., and Strahm, T. Weak theories of truth and explicit mathematics. In *Logic, Construction, Computation*, U. Berger, H. Diener, P. Schuster, and M. Seisenberger, Eds. Ontos Verlag, 2012, pp. 157–184.

[9] 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.

[10] 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.

[11] Feferman, S. Reflecting on incompleteness. *Journal of Symbolic Logic 56*, 1 (1991), 1–49.

[12] 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.

[13] Feferman, S. Predicativity. In *The Oxford Handbook of Philosophy of Mathematics and Logic*, S. Shapiro, Ed. Oxford University Press, 2005, pp. 590–624.

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

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

[16] Ferreira, F. Polynomial time computable arithmetic. In *Logic and Computation, Proceedings of a Workshop held at Carnegie Mellon University, 1987*, W. Sieg, Ed., vol. 106 of *Contemporary Mathematics*. American Mathematical Society, Providence, Rhode Island, 1990, pp. 137–156.

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

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