A Minimal Computational Theory of a Minimal Computational Universe

Arnon Avron and Liron Cohen

[1] Tel Aviv University, Tel-Aviv, Israel.
aa@post.tau.ac.il
[2] Cornell University, Ithaca, NY, USA.
lironcohen@cornell.edu

Abstract. In [3] a general logical framework for formalizing set theories of different strength was suggested. We here employ that framework, focusing on the exploration of *computational* theories. That is, theories whose set of closed terms suffices for denoting every concrete set (including infinite ones) that might be needed in applications, as well as for computations with sets. We demonstrate that already the minimal computational level of the framework, in which only a minimal computational theory and a minimal computational universe are employed, suffices for most, if not all, of applicable mathematics.

1 Introduction

Formalized mathematics and mathematical knowledge management (MKM) are extremely fruitful and quickly expanding fields of research at the intersection of mathematics and computer science (see, e.g., [2,8,21]). The declared goal of these fields is to develop computerized systems that effectively represent all important mathematical knowledge and techniques, while conforming to the highest standards of mathematical rigor. At present there is no general agreement what should be the best framework for this task. However, since most mathematicians view *set theory* as the basic foundation of mathematics, formalized set theories seem to us as the most natural choice.[3] [4]

In [3,4] a logical framework for developing and mechanizing set theories was introduced. Its key properties are that it is based on the usual (type-free) set theoretic language and makes extensive use of statically defined abstract set

[3] Already in [9] it was argued that "a main asset gained from Set theory is the ability to base reasoning on just a handful of axiom schemes which, in addition to being conceptually simple (even though surprisingly expressive), lend themselves to good automated support". More recently, H. Friedman wrote (in a message on FOM on Sep 14, 2015): "I envision a large system and various important weaker subsystems. Since so much math can be done in systems much weaker than ZFC, this should be reflected in the choice of Gold Standards. There should be a few major Gold Standards ranging from Finite Set Theory to full blown ZFC".

[4] Notable set-based automated provers are Mizar [27], Metamath [23] and SETL [28].

terms. Furthermore, it enables the use of different logics and set theories of different strength. This modularity of the system has been exploited in [5], where a hierarchy of set theories for formalizing different levels of mathematics within this framework was presented.

The current paper concentrates on one very basic theory, RST^{FOL}_{HF}, from the above-mentioned hierarchy, and on its minimal model. The latter is shown to be the universe J_2 in Jensen's hierarchy [20]. Both RST^{FOL}_{HF} and J_2 are *computational* (in a precise sense defined below). With the help of the formal framework of [3,4,5] they can therefore be used to make explicit the potential computational content of set theories (first suggested and partially demonstrated in [9]). On the other hand, they also suffice (as we show) for developing large portions of scientifically applicable mathematics [15], especially analysis.[5]

The restriction to a minimal, concrete framework has of course its price. Not all standard mathematical structures are elements of J_2. (The real line is a case in point.) Hence we have to treat such objects in a different manner: as proper classes. Accordingly, in this paper we introduce for the first time classes into the formal framework of [3,4,5], and develop efficient ways for handling them.

The paper is organized as follows: In Section 2 we present the formal framework, define the notions of computational theory and universe, and describe the computational theories which are minimal within the framework. Section 3 is dedicated to the introduction of standard extensions by definitions of the framework, done in a static way. We define the notions of sets and classes in our framework, and describe the way standard set theoretical notions are dealt with in the system. In Section 4 we turn to real analysis, and demonstrate how it can be developed in our minimal computational framework, although the reals are a proper class in it. This includes the introduction of the real line and real functions, as well as formulating and proving classical results concerning these notions.[6] Section 5 concludes with directions for future continuation of the work. *Due to lack of space, all proofs were omitted and are given in the appendix.*

2 Preliminaries

2.1 The Framework

Notation. To avoid confusion, the parentheses $\{\!|\,|\!\}$ are used in our formal languages, while in the meta-language we use $\{\,\}$. We use the letters $X, Y, Z, ...$ for collections; Φ, Θ for finite sets of variables; and $x, y, z, ...$ for variables in the formal language. $Fv(exp)$ denotes the set of free variables of exp, and $\varphi\left\{\frac{t_1}{x_1}, ..., \frac{t_n}{x_n}\right\}$ denotes the result of simultaneously substituting t_i for x_i in φ.

Definition 1 *Let C be a finite set of constants. The language $\mathcal{L}^C_{RST}$ and the associated safety relation $\succ$ are simultaneously defined as follows:*

[5] The thesis that J_2 is sufficient for core mathematics was already put forward in [31].

[6] A few of the claims in Section 4 have counterparts in [5]. The main difference is that in this paper the claims and their proofs have to be modified to handle classes.

– *Terms:*
  - *Every variable is a term.*
  - *Every $c \in C$ is a term (taken to be a constant).*
  - *If x is a variable and φ is a formula such that $\varphi \succ \{x\}$, then $\{\!|x \mid \varphi|\!\}$ is a term ($Fv(\{\!|x \mid \varphi|\!\}) = Fv(\varphi) - \{x\}$).*
– *Formulas:*
  - *If s, t are terms, then $t = s$, $t \in s$ are atomic formulas.*
  - *If φ, ψ are formulas and x is a variable, then $\neg\varphi, (\varphi \wedge \psi), (\varphi \vee \psi)$, $\exists x\varphi$ are formulas.*[7]
– *The safety relation $\succ$:*
  - *If φ is an atomic formula, then $\varphi \succ \emptyset$.*
  - *If t is a term such that $x \notin Fv(t)$, and $\varphi \in \{x \in x, x \in t, x = t, t = x\}$, then $\varphi \succ \{x\}$.*
  - *If $\varphi \succ \emptyset$, then $\neg\varphi \succ \emptyset$.*
  - *If $\varphi \succ \Theta$ and $\psi \succ \Theta$, then $\varphi \vee \psi \succ \Theta$.*
  - *If $\varphi \succ \Theta$, $\psi \succ \Phi$ and $\Phi \cap Fv(\varphi) = \emptyset$ or $\Theta \cap Fv(\psi) = \emptyset$, then $\varphi \wedge \psi \succ \Theta \cup \Phi$.*
  - *If $\varphi \succ \Theta$ and $y \in \Theta$, then $\exists y\varphi \succ \Theta - \{y\}$.*

Notation. We take the usual definition of $\subseteq$ in terms of $\in$, according to which $t \subseteq s \succ \emptyset$. $\{\!|t|\!\}$ denotes the term $\{\!|x \mid x = t|\!\}$, and $s \cup t$ the term $\{\!|x \mid x \in s \vee x \in t|\!\}$.

Definition 2 *The system RST_C^{FOL} is the classical first-order system with variable binding term operator (vbto; see, e.g., [13]) in $\mathcal{L}_{RST}^C$ which is based on the following set of axioms:* [8]

– *Extensionality:* $\forall z\,(z \in x \leftrightarrow z \in y) \rightarrow x = y$
– *Comprehension Schema:* $\forall x\,(x \in \{\!|x \mid \varphi|\!\} \leftrightarrow \varphi)$
– *Restricted $\in$-induction Schema:*

$$\left(\forall x\left(\forall y\left(y \in x \rightarrow \varphi\left\{\frac{y}{x}\right\}\right) \rightarrow \varphi\right)\right) \rightarrow \forall x\varphi \text{ , for } \varphi \succ \emptyset$$

In case $HF \in C$, the following axioms are added:

– $\emptyset \in HF$ *(where $\emptyset = \{\!|x \in HF \mid x \neq x|\!\}$)*
– $\forall x \forall y\,(x \in HF \wedge y \in HF \rightarrow x \cup \{\!|y|\!\} \in HF)$
– $\forall y\,(\emptyset \in y \wedge \forall v, w \in y.v \cup \{\!|w|\!\} \in y \rightarrow HF \subseteq y)$

Notation. In what follows, in case $C = \emptyset$ we elide C from our notations (e.g., we write RST^{FOL} for $RST_\emptyset^{FOL}$). Also, if $C = \{HF\}$ we simply write RST_{HF}^{FOL}.

An important feature of RST_C^{FOL} is that its first two axioms directly lead (and are equivalent) to the *set-theoretical* β and η reduction rules (see [3]).

In [3] it was suggested that the computationally meaningful instances of the Comprehension Axiom are those which determine the collections they define in an absolute way, independently of any "surrounding universe". In the context of set theory, a formula φ is "computable" w.r.t. x if the collection $\{x \mid \varphi(x, y_1, ..., y_n)\}$

[7] Though the official language does not include $\forall$ and $\rightarrow$, since we assume classical logic we take $\forall x_1 ... \forall x_n\,(\varphi \rightarrow \psi)$ as an abbreviation for $\neg\exists x_1 ... \exists x_n\,(\varphi \wedge \neg\psi)$.

[8] RST^{FOL} can be shown to be equivalent to the system obtained from Gandy's basic set theory [18] by adding to it the Restricted $\in$-induction schema.

is completely and uniquely determined by the identity of the parameters $y_1, ..., y_n$, and the identity of other objects referred to in the formula (all of which are well-determined beforehand). Note that φ is computable for $\emptyset$ iff it is absolute in the usual sense of set theory. In order to translate this idea into an exact, *syntactic* definition, the safety relation is used. Thus, only those formulas which are safe with respect to $\{x\}$ are allowed in the Comprehension Scheme.

Concerning $\in$-induction, even the full one does not seem to be in any conflict with the notion of a computational theory since it only imposes further restrictions on the collection of acceptable sets. Nevertheless, to be on the safe side, we adopt here only a very restricted variation of it. Moreover, we try to avoid (when possible) the use of this axiom, and shall point out the places where it is used.

It is easy to verify that the system RST_C^{FOL} is a proper subsystem of ZF. On the other hand, in [3] it was shown that the full power of ZF can be achieved by simply adding certain syntactic clauses to the definition of the safety relation.

While the formal language allows the use of set terms, it also provides a mechanizable static check of their validity due to the syntactic safety relation. To obtain decidable syntax logically equivalent formulas are not taken to be safe w.r.t. the same set of variables. However, if $\varphi \leftrightarrow \psi$ is provable in RST_C^{FOL}, then so is $x \in \{\!|x \mid \varphi|\!\} \leftrightarrow \psi$. Thus, we freely write $\{\!|x \mid \psi|\!\}$ for of $\{\!|x \mid \varphi|\!\}$ for such φ, ψ.

Definition 3 *Let C be a set of constants.*

1. *A function is called C-rudimentary if it rudimentary relative to the interpretations of the constants in C.* [9]
2. *A C-universe is a transitive collection of sets closed under C-rudimentary functions.*

For simplicity, in what follows we do not distinguish between a C-universe W and a structure for $\mathcal{L}_{RST}^C$ with domain W and an interpretation function I that assigns the obvious interpretations to the symbols $\in$, $=$, the set of hereditary finite sets to HF (if $HF \in C$), and an element in W to every $c \in C$.

Definition 4 *Let v be an assignment in a C-universe W. For a term t and formula φ of $\mathcal{L}_{RST}^C$, a collection $\|t\|_v^W$ and a truth value $\|\varphi\|_v^W \in \{\mathbf{t}, \mathbf{f}\}$ are standardly defined, with the additional clause:* $\|\{\!|x \mid \varphi|\!\}\|_v^W = \left\{a \in W \mid \|\varphi\|_{v[x:=a]}^W = \mathbf{t}\right\}$.[10]

From Corollary 6 below it follows that $\|t\|_v^W$ is an element of W, and $\|\varphi\|_v^W$ denotes the truth value of the formula φ under W and v.
Notation. In case *exp* is a closed expression, we denote by $\|exp\|^W$ the value of *exp* in W, and at times we omit the superscript W and simply write $\|exp\|$.

The following theorem is a slight generalization of a theorem in [4].

Theorem 5 *Let C be a set of constants.*

1. *If F is an n-ary C-rudimentary function, then there exists a formula φ_F of $\mathcal{L}_{RST}^C$ s.t. $Fv(\varphi_F) \subseteq \{y, x_1, ..., x_n\}$, $\varphi_F \succ \{y\}$ and $F(x_1, ..., x_n) = \{y \mid \varphi_F\}$.*

[9] Rudimentary functions are obtained by omitting the recursion schema from the usual list of schemata for primitive recursive set functions (see, e.g., [14]).

[10] $v[x := a]$ denotes the x-variant of v which assigns a to x.

2. *If* φ *is a formula of* $\mathcal{L}^C_{RST}$ *s.t.* $Fv(\varphi) \subseteq \{y_1, ..., y_k, x_1, ..., x_n\}$ *and* $\varphi \succ \{y_1, ..., y_k\}$, *then there exists a* C*-rudimentary function* F_φ *s.t.* $F_\varphi(x_1, ..., x_n) = \{\langle y_1, ..., y_k\rangle \mid \varphi\}$.
3. *If* t *is a term of* $\mathcal{L}^C_{RST}$ *s.t.* $Fv(t) \subseteq \{x_1, ..., x_n\}$, *then there exists a* C*-rudimentary function* F_t *s.t.* $F_t(x_1, ..., x_n) = t$ *for every* $x_1, ..., x_n$.

Corollary 6 *Let* v *be an assignment in a* C*-universe* W.

1. *For a term* t *of* $\mathcal{L}^C_{RST}$, $\|t\|^W_v \in W$.
2. *For a formula* φ *of* $\mathcal{L}^C_{RST}$ *s.t.* $\{y_1, ..., y_n\} \subseteq Fv(\varphi)$:
   (a) *If* $\varphi \succ \{y_1, ..., y_n\}$ $(n > 0)$, $\left\{\langle a_1, ..., a_n\rangle \in W^n \mid \|\varphi\|^W_{v[\boldsymbol{y}:=\overrightarrow{a}]} = \mathbf{t}\right\} \in W$.
   (b) *If* $\varphi \succ \emptyset$ *and* $X \in W$, *then* $\left\{\langle a_1, ..., a_n\rangle \in X^n \mid \|\varphi\|^W_{v[\boldsymbol{y}:=\overrightarrow{a}]} = \mathbf{t}\right\} \in W$.

If t is a closed term s.t. $\|t\|^W = X$, we say that t defines X (X is definable by t).

Corollary 7 *Any* C*-universe is a model of* RST^{FOL}_C.

Lemma 8 *[5] The following notations are available in* RST^{FOL} *(i.e. they can be introduced as abbreviations in* $\mathcal{L}_{RST}$ *and their basic properties are provable in* RST^{FOL}*):* $\emptyset$, $\langle t_1, ..., t_n\rangle$, $\{\!\{t_1, ..., t_n\}\!\}$, $\{\!\{x \in t \mid \varphi\}\!\}$ *(provided* $\varphi \succ \emptyset$ *and* $x \notin Fv(t)$*),* $\{\!\{t \mid x \in s\}\!\}$ *(provided* $x \notin Fv(s)$*),* $s \times t$, $s \cup t$, $s \cap t$, $s - t$, $\cup t$, $\cap t$, $\pi_1(t)$, $\pi_2(t)$, $Dom(t)$, $Im(t)$, $\iota x.\varphi$ *(provided* $\varphi \succ \{x\}$*),* $\lambda x \in s.t$ *(provided* $x \notin Fv(s)$*).*

2.2 Computational Theories and Universes

Computations within a set of objects require concrete representations of these objects. Accordingly, we call a theory *computational* if its set of closed terms induces in a natural way a minimal model of the theory, and it enables the key properties of these elements to be provable within it. Next we provide a more formal definition for the case of set theories which are defined within our general framework. Note that from a Platonist point of view, the set of closed terms of such a theory $\mathcal{T}$ induces some subset $\mathcal{S}_\mathcal{T}$ of the cumulative universe of sets V, as well as some subset $\mathcal{M}_\mathcal{T}$ of any transitive model $\mathcal{M}$ of $\mathcal{T}$.

Definition 9

1. *A theory* $\mathcal{T}$ *in the above framework is called* computational *if the set* $\mathcal{S}_\mathcal{T}$ *it induces is a transitive model of* $\mathcal{T}$*, and the identity of* $\mathcal{S}_\mathcal{T}$ *is absolute in the sense that* $\mathcal{M}_\mathcal{T} = \mathcal{S}_\mathcal{T}$ *for any transitive model* $\mathcal{M}$ *of* $\mathcal{T}$ *(implying that* $\mathcal{S}_\mathcal{T}$ *is actually a* minimal *transitive model of* $\mathcal{T}$*).*
2. *A set is called computational if it is* $\mathcal{S}_\mathcal{T}$ *for some computational theory* $\mathcal{T}$.

The most basic computational theories are the two minimal theories in the hierarchy of systems developed in [5]. This fact, as well as the corresponding computational universes, are described in the following three results from [5].

Proposition 10 *Let* J_1, J_2 *be the first two universes in Jensen's hierarchy [20].*

1. *J_1 is a model of RST^{FOL}.*
2. *J_2 with the interpretation of HF as J_1 is a model of RST^{FOL}_{HF}.*

Theorem 11

- *$X \in J_1$ iff there is a closed term t of $\mathcal{L}_{RST}$ s.t. $\|t\|^{J_1} = X$.*
- *$X \in J_2$ iff there is a closed term t of $\mathcal{L}^{HF}_{RST}$ such that $\|t\|^{J_2} = X$.*

Corollary 12 *RST^{FOL} and RST^{FOL}_{HF} are computational, and J_1 and J_2 are their computational universes.*

Now J_1, the minimal computational universe, is the set of hereditary finite sets. This universe captures the standard data structures used in computer science, like strings and lists. However, in order to be able to capture computational structures with infinite objects, we have to move to RST^{FOL}_{HF}, whose computational universe, J_2, seems to be the minimal universe that suffices for this purpose. RST^{FOL}_{HF} still allows for a very concrete, computationally-oriented interpretation, and it is appropriate for mechanical manipulations and interactive theorem proving. Moreover, as noted in the introduction, its corresponding universe J_2 is rich enough for a systematic development of applicable mathematics.

3 Static Extensions by Definitions

When working in a minimal computational universe such as J_2 (as done in the next section), many of the standard mathematical objects (such as the real line and real functions) are only available in our framework as proper classes. Thus, in order to be able to formalize standard theorems regarding such objects we must enrich our language to include them. Introducing classes into our framework, however, is a part of the more general method of extensions by definitions which is an essential part of every mathematical research and its presentation. Now, there are two principles that govern this process in our framework. First, the static nature of our framework demands that conservatively expanding the language of a given theory should be reduced to the use of *abbreviations*. Second, since the introduction of new predicates and function symbols creates new atomic formulas and terms, one should be careful that the basic conditions concerning the underlying safety relation $\succ$ are preserved. Thus only formulas φ s.t. $\varphi \succ \emptyset$ can be used for defining new predicate symbols.

We start with the problem of introducing new unary predicate symbols.[11] In standard practice such extensions are carried out by introducing a new unary predicate symbol P and either treating $P(t)$ as an abbreviation for $\varphi(t)$ for some formula φ, or (what is more practical) adding $\forall x\,(P(x) \leftrightarrow \varphi)$ as an axiom to the (current version of the base) theory, obtaining by this a conservative theory in the extended language. However, in the set theoretical framework it is possible and frequently more convenient to uniformly use class terms, rather than introduce a new predicate symbol each time. Thus, instead of writing "$P(t)$" one uses

[11] The use of n-ary predicates can standardly be reduced, of course, to unary predicates.

an appropriate class term S and writes "$t \in S$". Whatever approach is chosen – in order to respect the definition of a safety relation, class terms should be restricted so that "$t \in S$" is safe w.r.t. $\emptyset$. Accordingly, we extend our language by incorporating class terms which are objects of the form $\{\!|x \,\hat{|}\, \varphi|\!\}$, where $\varphi \succ \emptyset$. The use of these terms is done in the standard way. In particular, $t \in \{\!|x \,\hat{|}\, \varphi|\!\}$ (where t is free for x in φ) is equivalent to (and may be taken as an abbreviation for) $\varphi\left\{\frac{t}{x}\right\}$. It should be emphasized that a class term is not a valid term in the language, only a definable predicate. The addition of the new notation does not enhance the expressive power of $\mathcal{L}^C_{RST}$, but only increases the ease of using it.

A further conservative extension of the language that we shall use incorporates free class variables, $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{Z}$, and free function variables, $\boldsymbol{F}, \boldsymbol{G}$, into $\mathcal{L}^C_{RST}$ (as in free-variable second-order logic [29]). These variables stand for arbitrary class or function terms (the latter is defined in Def. 20), and they may only appear as *free* variables, *never to be quantified*. We allow occurrences of such variables inside a formula in a class term or a function term. One may think of a formula with such variables as a schema, where the variables play the role of "place holders", and whose substitution instances abbreviate official formulas of the language (see Example 2). In effect, a formula $\psi(\boldsymbol{X})$ with free class variable $\boldsymbol{X}$ can be intuitively interpreted as "for any *given* class X, $\psi(X)$ holds". Thus, a free-variable formulation has the flavor of a universal formula. Therefore, this addition allows statements about *all* potential classes and *all* potential functions.

We define $\left\|\{\!|x \,\hat{|}\, \varphi|\!\}\right\|^W_v = \left\{a \in W \mid \|\varphi\|^W_{v[x:=a]} = \mathbf{t}\right\}$. We say that the class term defines the latter collection (which might not be an element of W).

Definition 13 *Let X be a collection of elements in W.*
- *X is a $\succ$-*set *if there is a closed term that defines it. If X is a $\succ$-set, $\widetilde{X}$ denotes some closed term that defines it.*
- *X is a $\succ$-*class *if there is a closed class term that defines it. If X is a $\succ$-class, $\bar{X}$ denotes some closed class term that defines it.*

Note that, by Corollary 6, if X is a *$\succ$-set* then $X \in W$.

Proposition 14 *The following holds:*
1. *Every $\succ$-set is a $\succ$-class.*
2. *The intersection of a $\succ$-class with a $\succ$-set is a $\succ$-set.*
3. *Every $\succ$-class that is contained in a $\succ$-set is a $\succ$-set.*

Remark 15 A semantic counterpart of our notion of a $\succ$-class was used in [31], and is there called an ι-class. It is defined as a definable subset of J_2 whose intersection with any element of J_2 is in J_2. The second condition in this definition seems somewhat ad hoc. More importantly, it is unclear how it can be checked in general, and what kind of set theory is needed to establish that certain collections are ι-classes. The definition of a $\succ$-class used here is, in contrast, motivated by and based on purely syntactical considerations. It is also a simplification of the notion of ι-class as by Prop. 14(2) every $\succ$-class is an ι-class.[12]

[12] Two other ideas that appear in the sequel were adopted from [31]: treating the collection of reals as a proper class, and the use of codes for handling certain classes.

Proposition 16 *The following holds:*

- *Let Y be a $\succ$-set. If $\varphi \succ \emptyset$ and $Fv(\varphi) \subseteq \{x\}$, then $\{x \in Y \mid \varphi\}$ is a $\succ$-set.*
- *If $\varphi \succ \{x_1, ..., x_n\}$, then $\{\langle x_1, ..., x_n\rangle \mid \varphi\}$ is a $\succ$-set.*

Proposition 17 *For every n-ary C-rudimentary function f there is a term t with $Fv(t) \subseteq \{x_1, ..., x_n\}$ s.t. for any $\langle A_1, ..., A_n\rangle \in W^n$, f returns the $\succ$-set $\|t\|^W_{[x_1:=A_1,...,x_n:=A_n]}$.*

Proposition 18 *If X, Y are $\succ$-classes, so are $X \cup Y$, $X \cap Y$, $X \times Y$, $J_2 - X$, and $P_{J_2}(X) = \{z \in J_2 \mid z \subseteq X\}$.*

For a class term s we denote by 2^s the class term $\{z \mid z \subseteq s\}$. Note that for any assignment v in W and class term s, $\|2^s\|^W_v$ is equal to $P_W\left(\|s\|^W_v\right)$, i.e., the intersection of the power set of $\|s\|^W_v$ and W. This demonstrates the main difference between set terms and class terms. The interpretation of set terms is absolute, whereas the interpretation of class terms might not be (though membership in the interpretation of a class term is absolute).

Definition 19 *A $\succ$-*relation from *a $\succ$-*class *X to a $\succ$-*class *Y is a $\succ$-class A s.t. $A \subseteq X \times Y$. A $\succ$-relation is called* small *if it is a $\succ$-set.*

Next we extend our framework by the introduction of new function symbols. This poses a new difficulty. While new relation symbols are commonly introduced in a static way, new function symbols are usually introduced *dynamically*: a new function symbol is made available after appropriate existence and uniqueness theorems had been proven. However, one of the main guiding principles of our framework is that its languages should be treated exclusively in a *static* way. Thus function symbols, too, are introduced only as abbreviations for definable operations on sets.[13]

Definition 20

- *For a closed class term T and a term t of $\mathcal{L}^C_{RST}$, $\lambda x \in \mathsf{T}.t$ is a* function term *which is an abbreviation for $\{z \mid \exists x \exists y\,(z \doteq \langle x, y\rangle \wedge x \in \mathsf{T} \wedge y = t)\}$.*[14]
- *$A \succ$-class F is called a* $\succ$-function on a $\succ$-class *X if there is a function term $\lambda x \in \mathsf{T}.t$ such that $X = \|\mathsf{T}\|$, $Fv(t) \subseteq \{x\}$ and $F = \|\lambda x \in \mathsf{T}.t\|$. t is called a term which* represents *F.*
- *$A \succ$-class is called a $\succ$-function if it is a $\succ$-function on some $\succ$-class.*
- *$A \succ$-function is called* small *if it is a $\succ$-set.*

It should nevertheless be emphasized that the framework in [31] is exclusively based on semantical considerations, and it is unclear how it can be turned into a formal theory like ZF or PA (and it is certainly not suitable for mechanization as is).

[13] In this paper, as in standard mathematical textbooks, the term "function" is used both for collections of ordered pairs and for set-theoretical operations (such as $\cup$).

[14] We abbreviate by $z \doteq \langle x, y\rangle$ and $\langle x, y\rangle \doteq z$ the two formulas that are provably equivalent to $z = \langle x, y\rangle$ and $\langle x, y\rangle \in z$ and are safe w.r.t. $\{x, y\}$ which were introduced in [5].

Note that the standard functionality condition is always satisfied in a $\succ$-*function*. *Terminology.* In what follows, claiming that an object is *available in* RST_C^{FOL} *as a* $\succ$-*function (*$\succ$-*relation)* means that it is definable as a $\succ$-function *(*$\succ$-*relation)* in $\mathcal{L}_{RST}^C$, and that its basic properties are provable in RST_C^{FOL}.[15]

Proposition 21 *Let* X, Y *be* $\succ$-*classes and* R *a* $\succ$-*relation from* X *to* Y.

1. R *is small iff* $Dom(R)$ *and* $Im(R)$ *are* $\succ$-*sets.*
2. $R^{-1} = \{\langle y, x\rangle \mid \langle x, y\rangle \in R\}$ *is available in* RST_C^{FOL} *as a* $\succ$-*relation from* Y *to* X. *If* R *is small, then so is* R^{-1}.
3. *If* $Z \subseteq X$ *and* $U \subseteq Y$ *are* $\succ$-*classes, then* $R \cap (Z \times U)$ *is available in* RST_C^{FOL} *as a* $\succ$-*relation from* Z *to* U .

Proposition 22 *A* $\succ$-*set is a function according to the standard mathematical definition (a single-valued relation) iff it is a small* $\succ$-*function.*

Notation. Let $F = \left\|\lambda x \in \bar{X}.t\right\|$ be a $\succ$-function. We employ standard β-reduction for λ terms. Thus, we write $F(s)$ for $t\left\{\frac{s}{x}\right\}$ if s is free for x in t. Hence $F(s) = y$ stands for $t\left\{\frac{s}{x}\right\} = y$, and so if $y \notin Fv[t] \cup Fv[s] \setminus \{x\}$, then $F(s) = y \succ \{y\}$.

Proposition 23 (Replacement axiom in class form) *Let* F *be a* $\succ$-*function on a* $\succ$-*class* X. *Then for every* $\succ$-*set* $A \subseteq X$, $F[A] = \{F(a) \mid a \in A\}$ *is a* $\succ$-*set.*

Below is a natural generalization of Def. 20 to functions of several variables.

Lemma 24 *If* $X_1, ..., X_n$ are $\succ$-*classes and* t *is a term s.t.* $Fv(t) \subseteq \{x_1, ..., x_n\}$, *then* $F = \left\|\lambda x_1 \in \bar{X}_1, ..., x_n \in \bar{X}_n.t\right\|$ *is available in* RST_C^{FOL} *as a* $\succ$-*function on* $X_1 \times ... \times X_n$. *(where* $\lambda x_1 \in \bar{X}_1, ..., x_n \in \bar{X}_n.t$ *is an abbreviation for* $\left\{\langle\langle x_1, ..., x_n\rangle, t\rangle \mid \langle x_1, ..., x_n\rangle \in \bar{X}_1 \times ... \times \bar{X}_n\right\}$*).*

Corollary 25 *Every* C-*rudimentary function is available in* RST_C^{FOL} *as a* $\succ$-*function.*

Proposition 26 *Let* F *be a* $\succ$-*function on a* $\succ$-*class* X.

1. F *is small iff* X *is a* $\succ$-*set.*
2. *If* Y_0 *is a* $\succ$-*class, then* $F^{-1}[Y_0] = \{a \in X \mid F(a) \in Y_0\}$ *is a* $\succ$-*class. If* F *is small, then* $F^{-1}[Y_0]$ *is a* $\succ$-*set.*
3. *If* $X_0 \subseteq X$ *is a* $\succ$-*class, then* $F \restriction_{X_0}$ *is available in* RST_C^{FOL} *as a* $\succ$-*function.*
4. $G \circ F$ *is available in* RST_C^{FOL} *as a* $\succ$-*function on* X, *in case* G *is a* $\succ$-*function on a* $\succ$-*class* Y *and* $Im(F) \subseteq Y$.
5. *If* G *is a* $\succ$-*function on a* $\succ$-*class* Y *and* F *and* G *agree on* $X \cap Y$, *then* $G \cup F$ *is available in* RST_C^{FOL} *as a* $\succ$-*function on* $X \cup Y$.
6. *If* Z *is a* $\succ$-*class then the identity on* Z *and any constant function on* Z *are available in* RST_C^{FOL} *as* $\succ$-*functions.*

[15] The "basic properties" of a certain object is of course a fuzzy notion. However, it is not difficult to identify its meaning in each particular case, as will be demonstrated in several examples below.

4 Real Analysis in J_2

It is not difficult to formalize the definitions, claims, and proofs of this section in our formal framework. These translations are straightforward, but rather tedious. Hence we shall omit them, with the exception of a few outlined examples.

4.1 The Natural Numbers

We follow the standard construction of the natural numbers: $0 := \emptyset$; $n+1 := S(n)$, where $S(n) = n \cup \{n\}$. Each $n \in \mathbb{N}$ is a $\succ$-set, and $\mathbb{N}$ (the set of natural numbers) is contained in the interpretation of HF.

In mainstream mathematics, as well as in standard computerized theorem provers, the collection of natural numbers is taken as a basic object. This is because it constitutes a well-understood, computational concept. Now, the computational universe associated with RST^{FOL} is J_1, in which $\mathbb{N}$ is available only as a proper $\succ$-class. To solve this, in RST^{FOL}_{HF} a special constant HF was added, whose axioms ensure (as far as possible on the first-order level) that it is to be interpreted as the set of hereditary finite sets. These axioms in fact replace the usual infinity axiom of ZF. This increases the computational power of the theory and captures the natural numbers as a $\succ$-set. Thus, in what follows we restrict our attention to the computational theory RST^{FOL}_{HF} and its computational universe J_2. Therefore, for readability, we simply write $\|exp\|_v$ instead of $\|exp\|^{J_2}_v$.

The induction rule is available in RST^{FOL}_{HF}, but only for $\varphi \succ \emptyset$.

Proposition 27 $\vdash_{RST^{FOL}_{HF}} (\varphi(0) \wedge \forall x (\varphi \to \varphi(S(x)))) \to \forall x \in \widetilde{\mathbb{N}}.\varphi$, *for* $\varphi \succ \emptyset$.

Basic properties of the natural numbers which can be formulated in the language of first-order Peano arithmetics are provable in RST^{FOL}_{HF} using the restricted induction principle given in Prop. 27. This is because in their translation to $\mathcal{L}^{HF}_{RST}$, all the quantifications are bounded in $\mathbb{N}$, and thus they are safe w.r.t. $\emptyset$. [16]

4.2 The Real Line

The standard construction of $\mathbb{Z}$, the set of integers, as the set of ordered pairs $(\mathbb{N} \times \{0\}) \cup (\{0\} \times \mathbb{N})$ can be easily carried out in RST^{FOL}_{HF}, as can the usual construction of $\mathbb{Q}$, the set of rationals, in terms of ordered pairs of relatively prime integers. There is also no difficulty in defining the standard orderings on $\mathbb{Z}$ and $\mathbb{Q}$ as small $\succ$-relations, as well as the standard functions of addition and multiplication as small $\succ$-functions. The main properties of addition and multiplication are provable in RST^{FOL}_{HF}, as the standard proofs by induction can be carried out within it. Furthermore, all the basic properties of $\mathbb{Z}$ and $\mathbb{Q}$ (such as $\mathbb{Q}$ being a dense unbounded field) are straightforwardly proven in RST^{FOL}_{HF}.

Now we turn to the standard construction of the real line using Dedekind cuts. Since it is well known that the real line and its open segments are not

[16] It can be shown that the power of full induction over $\mathbb{N}$ (i.e. for *any* formula φ) can be achieved by adding to RST^{FOL}_{HF} the full $\in$-induction scheme.

absolute, they cannot be $\succ$-sets, only proper $\succ$-classes. Thus the collection of real numbers in RST^{FOL}_{HF} will not be a term but merely a *definable predicate.*[17]

Let $\psi(u) = \forall x, y \in \widetilde{\mathbb{Q}}.x \in u \wedge y < x \rightarrow y \in u$, $\varphi(u) = \neg\exists x \in u \forall y \in u.y \leq x$.

Definition 28 (The Reals) $\mathbb{R}$ *is* $\left\| \{u \in \overline{P_{J_2}(\mathbb{Q}) \setminus \{\emptyset, \mathbb{Q}\}} \mid \psi(u) \wedge \varphi(u)\} \right\|$.

The above term is a valid class term as $P_{J_2}(\mathbb{Q}) \setminus \{\emptyset, \mathbb{Q}\}$ is a $\succ$-class, and $\varphi, \psi \succ \emptyset$.

Note that the $\succ$-class $\mathbb{R}$ is not the "real" real-line (if such a thing really exists). However, it does contain all *computable* real numbers, such as $\sqrt{2}$ and π (see [5]). *Notation.* We employ the following notations: $\mathbb{Q}^+ = \{q \in \mathbb{Q} \mid 0 < q\}$, $\mathbb{R}^+ = \{r \in \mathbb{R} \mid 0 < r\}$, $(a, b) = \{r \in \mathbb{R} \mid a < r < b\}$ and $[a, b] = \{r \in \mathbb{R} \mid a \leq r \leq b\}$, for a, b real numbers. [18]

Proposition 29 *The following holds:*

1. *The standard ordering $<$ on $\mathbb{R}$ is available in RST^{FOL}_{HF} as a $\succ$-relation.*
2. *The standard addition and multiplication of reals are available in RST^{FOL}_{HF} as $\succ$-functions.*

We next show that the least upper bound principle is provable in RST^{FOL}_{HF} for $\succ$-subsets of $\mathbb{R}$.

Theorem 30 *It is provable in RST^{FOL}_{HF} that every nonempty $\succ$-subset of $\mathbb{R}$ that is bounded above has a least upper bound in $\mathbb{R}$. Furthermore, the induced mapping (l.u.b) is available in RST^{FOL}_{HF} as a $\succ$-function.*

Theorem 30 only states that $\succ$-*subsets* of $\mathbb{R}$ have the least upper bound property. Thus, it is insufficient for the development of most of standard mathematics in RST^{FOL}_{HF}. The reason is that even the most basic substructures of $\mathbb{R}$, like the intervals, are not $\succ$-sets, but proper $\succ$-classes in RST^{FOL}_{HF}. Hence, a stronger version of the theorem, which ensures that the least upper bound property holds for standard $\succ$-subclasses of $\mathbb{R}$, is needed. Theorem 40 below provides such an extension, but it requires some additional definitions and propositions.

First we consider $\succ$-classes $U \subseteq \mathbb{R}$ which are open. These $\succ$-classes are generally not $\succ$-sets (unless empty), since they contain an interval of positive length, which is a proper $\succ$-class and thus cannot be contained in a $\succ$-set (see Prop. 14(3)). Clearly, there is no such thing as a $\succ$-set of $\succ$-classes, as a proper $\succ$-class can never be an element of another $\succ$-set or $\succ$-class. However, the use of coding (following [30], [31][19]) allows us, for example, to replace the meaningless statement "the union of a $\succ$-set of $\succ$-classes is a $\succ$-class" with "given a $\succ$-set of codes for $\succ$-classes, the union of the corresponding $\succ$-classes is a $\succ$-class".

The coding technique we use is based on the standard mathematical notation for a "family of sets", $(A_i)_{i \in I}$, where I is a set of indices and A_i is a set for each

[17] As noted in Footnote 6, some of claims in the sequel have counterparts in [5]. However, the minimality restriction on the universe employed in this paper, which in turn requires the use of classes, makes a crucial difference.

[18] Notice that $\mathbb{Q}^+$ is a $\succ$-set and $\mathbb{R}^+$ is a $\succ$-class.

[19] In [31] such codings are called "proxies".

$i \in I$. In RST^{FOL}_{HF} we cannot construct the collection of all such A_i's if A_i is a $\succ$-class for some $i \in I$. Thus, we treat the $\succ$-set I as a code for the "family of classes" $(A_i)_{i \in I}$. In fact, we mainly use the union of such families, i.e., $\bigcup_{i \in I} A_i$.

Definition 31 *For any $p \in \mathbb{R}$ and $q \in \mathbb{R}^+$, the open ball $B_q(p)$ is the $\succ$-class $\{r \in \mathbb{R} \mid |r - p| < q\}$.*

Definition 32 *Let $U \subseteq \mathbb{R}$ be a $\succ$-class. If there exists a $\succ$-set $u \subseteq \mathbb{Q} \times \mathbb{Q}^+$ s.t. $U = \bigcup_{\langle p,q \rangle \in u} B_q(p) = \{r \in \mathbb{R} \mid \exists p, q\, (\langle p, q \rangle \in u \wedge |r - p| < q)\}$, then U is called* open *and u is a* code *for U.*

In what follows, the formalizations in RST^{FOL}_{HF} are carried out as follows:

- To quantify over open $\succ$-classes: $Qu \subseteq \widetilde{\mathbb{Q} \times \mathbb{Q}^+}$ $(Q \in \{\forall, \exists\})$.
- To decode the open $\succ$-class whose code is u:

$$dec(u) := \{\!|r \in \bar{\mathbb{R}} \,\hat{|}\, \exists p, q\, (\langle p, q \rangle \check{\in} u \wedge |r - p| < q)|\!\}$$

- To state that a class variable $\boldsymbol{U}$ is an open $\succ$-class:

$$Open(\boldsymbol{U}) := \exists u \subseteq \widetilde{\mathbb{Q} \times \mathbb{Q}^+}.\boldsymbol{U} = dec(u)$$

Proposition 33 *The following are provable in RST^{FOL}_{HF}:*

1. *For any $\succ$-set $u \subseteq \mathbb{R} \times \mathbb{R}^+$, $\{r \in \mathbb{R} \mid \exists p, q\, (\langle p, q \rangle \in u \wedge |r - p| < q)\}$ is an open $\succ$-class.*
2. *The open ball $B_q(p)$ is an open $\succ$-class for any $p \in \mathbb{R}$ and $q \in \mathbb{R}^+$.*

Proposition 34 *The following are provable in RST^{FOL}_{HF}:*

1. *The union of a $\succ$-set of open $\succ$-classes is an open $\succ$-class. i.e, given a $\succ$-set of codes of open $\succ$-classes, the union of the corresponding open $\succ$-classes is an open $\succ$-class.*
2. *The intersection of finitely many open $\succ$-classes is an open $\succ$-class.*

Example 1. As an example of the use of the coding technique, we demonstrate the formalization of Prop. 34(1):

$$\forall z.(\forall x \in z.x \subseteq \widetilde{\mathbb{Q} \times \mathbb{Q}^+}) \to \exists w \subseteq \widetilde{\mathbb{Q} \times \mathbb{Q}^+}.dec(w) = \{\!|r \,\hat{|}\, \exists x \in z.r \in dec(x)|\!\}$$

Definition 35 *A $\succ$-class $X \subseteq \mathbb{R}$ is* closed *if $\mathbb{R} - X$ is open.*

Lemma 36 *Let $X \subseteq \mathbb{R}$ be a $\succ$-class and $A \subseteq X$ be a $\succ$-set. The following are equivalent in RST^{FOL}_{HF}:*

1. *Every open ball about a point in X intersects A.*
2. *Every open $\succ$-class that intersects X also intersects A.*

Example 2. As an example of a full formalization which uses class variables, the formalization of the Lemma above is:

$$\phi := \boldsymbol{X} \subseteq \bar{\mathbb{R}} \to \forall a \subseteq \boldsymbol{X} \Big(\forall x \in \boldsymbol{X} \forall \varepsilon \in \bar{\mathbb{R}^+}\, (B_\varepsilon(x) \cap a \neq \emptyset) \leftrightarrow \forall u \subseteq \widetilde{\mathbb{Q} \times \mathbb{Q}^+}\, (dec(u) \cap \boldsymbol{X} \neq \emptyset \to dec(u) \cap a \neq \emptyset)\Big)$$

We now demonstrate how to obtain a formula in the basic $\mathcal{L}_{RST}^{HF}$ by replacing each appearance of a class term or variable with the formula it stands for. First, we explain the translation of $x \in \bar{\mathbb{R}}$ to $\mathcal{L}_{RST}^{HF}$. One iteration of the translation entails $x \in \overline{P_{J_2}(\mathbb{Q}) \setminus \{\emptyset, \mathbb{Q}\}} \wedge \varphi(x) \wedge \psi(x)$ for φ, ψ as in Def. 28. A second iteration yields $R(x) := x \subseteq \tilde{\mathbb{Q}} \wedge x \neq \tilde{\mathbb{Q}} \wedge x \neq \emptyset \wedge \varphi(x) \wedge \psi(x)$ which is in $\mathcal{L}_{RST}^{HF}$. For the translation of ϕ, first substitute $\hat{\{}x \,|\, \theta\hat{\}}$ for $\boldsymbol{X}$, where $\theta \succ \emptyset$. Proceeding with the translation steps results in the following formula (scheme) of $\mathcal{L}_{RST}^{HF}$, for $\theta \succ \emptyset$:

$$\forall b\,(\theta(b) \to R(b)) \to \forall a\,((\forall z.z \in a \to \theta(z)) \to \forall x\,(\theta(x) \to \forall \varepsilon\,((R(\varepsilon) \wedge 0 < \varepsilon) \to$$
$$\exists w.\,|w - x| < \varepsilon \wedge w \in a \leftrightarrow \forall u \subseteq \widetilde{\mathbb{Q} \times \mathbb{Q}^+}\,(\exists w.R(w) \wedge \exists p, q\,(\langle p, q\rangle \check{\in} u \wedge |w - p| < q) \wedge$$
$$\theta(w)) \to \exists w.R(w) \wedge \exists p, q\,(\langle p, q\rangle \check{\in} u \wedge |w - p| < q) \wedge w \in a)$$

Remark 37 When we say that a theorem about a $\succ$-class or a $\succ$-function is provable in RST_{HF}^{FOL} (as in Lem. 36), we mean that it can be formalized and proved as a scheme. That is, that its proof can be carried out in RST_{HF}^{FOL} using a uniform scheme. The one exception is theorems about open $\succ$-classes, which due to the coding machinery can be fully formalized and proved in RST_{HF}^{FOL}.

Definition 38 *Let $X \subseteq \mathbb{R}$ be a $\succ$-class, and $A \subseteq X$ a $\succ$-set. A is called* dense in X *if one of the conditions of Lemma 36 holds. X is called* separable *if it contains a dense $\succ$-subset.*

Proposition 39 *It is provable in RST_{HF}^{FOL} that an open $\succ$-subclass of a separable $\succ$-class is separable.*

Now we can finally turn to prove a more encompassing least upper bound theorem.

Theorem 40 *It is provable in RST_{HF}^{FOL} that every nonempty separable $\succ$-subclass of $\mathbb{R}$ that is bounded above has a least upper bound in $\mathbb{R}$.*

Definition 41 *A $\succ$-class $X \subseteq \mathbb{R}$ is called an* interval *if for any $a, b \in X$ s.t. $a < b$: if $c \in \mathbb{R} \wedge a < c < b$ then $c \in X$.*

Proposition 42 *It is provable in RST_{HF}^{FOL} that a non-degenerate interval is separable. If it is also bounded above then it has a least upper bound.*

Proposition 43 *Let $X \subseteq \mathbb{R}$ be a $\succ$-class. It is provable in RST_{HF}^{FOL} that X is connected (i.e. cannot be disconnected by two open $\succ$-classes) iff it is an interval.*

4.3 Real Functions

Definition 44 *Let X be a $\succ$-class. A* $\succ$-sequence *in X is a $\succ$-function on $\mathbb{N}$ whose image is contained in X.*

Lemma 45 *It is provable in RST_{HF}^{FOL} that Cauchy $\succ$-sequences in $\mathbb{R}$ converge to limits in $\mathbb{R}$. The induced map (lim) is available in RST_{HF}^{FOL} as a $\succ$-function.*

Proposition 46 *It is provable in RST_{HF}^{FOL} that if $X \subseteq \mathbb{R}$ is closed, then every Cauchy $\succ$-sequence in X converges to a limit in X.*

Next we want to study sequences of functions, but Def. 44 cannot be applied as is, since $\succ$-functions which are proper $\succ$-classes cannot be values of a $\succ$-function (in particular, of a $\succ$-sequence). Instead, we use the standard Uncurrying procedure.

Definition 47 *For X, Y $\succ$-classes, a* $\succ$-sequence of $\succ$-functions *on X whose image is contained in Y is a $\succ$-function on $\mathbb{N} \times X$ with image contained in Y.*

Proposition 48 *Any point-wise limit of a $\succ$-sequence of $\succ$-functions on a $\succ$-class $X \subseteq \mathbb{R}$ whose image is contained in $\mathbb{R}$ is available in RST^{FOL}_{HF} as a $\succ$-function.*

Next we turn to continuous real $\succ$-functions. One possibility of doing so, adopted e.g., in [30,32], is to introduce codes for continuous real $\succ$-functions (similar to the use of codes for open $\succ$-classes). This is of course possible as such $\succ$-functions are determined by their values on the $\succ$-set $\mathbb{Q}$. However, we prefer to present here another approach, which allows for almost direct translations of proofs in standard analysis textbook into our system. This is done using free function variables. Accordingly, the theorems which follow are schemes. Implicitly, the previous sections of this paper can also be read and understood as done in this manner. Therefore, in what follows we freely use results from them.

Definition 49 *Let $X \subseteq \mathbb{R}$ be a $\succ$-class and let F be a $\succ$-function on X whose image is contained in $\mathbb{R}$. F is called a* continuous real $\succ$-function *if:*

$$\forall a \in X \forall \varepsilon \in \mathbb{R}^{+} \exists \delta \in \mathbb{R}^{+} \forall x \in X. \left|x - a\right| < \delta \rightarrow \left|F\left(x\right) - F\left(a\right)\right| < \varepsilon$$

Proposition 50 *Let $X \subseteq \mathbb{R}$ be a $\succ$-class and F be a $\succ$-function on X whose image is contained in $\mathbb{R}$. It is provable in RST^{FOL}_{HF} that if for every open $\succ$-class $B \subseteq \mathbb{R}$, there is an open $\succ$-class A s.t. $F^{-1}\left[B\right] = A \cap X$, then F is continuous.*

Lemma 51 *The following are provable in RST^{FOL}_{HF}:*

1. *The composition, sum and product of two continuous real $\succ$-functions is a continuous real $\succ$-function.*
2. *The uniform limit of a $\succ$-sequence of continuous real $\succ$-functions is a continuous real $\succ$-function.*

Theorem 52 (Intermediate Value Theorem) *Let F be a continuous real $\succ$-function on an interval $[a, b]$ with $F\left(a\right) < F\left(b\right)$. It is provable in RST^{FOL}_{HF} that for any $d \in \mathbb{R}$ s.t. $F\left(a\right) < d < F\left(b\right)$, there is $c \in [a, b]$ s.t. $F\left(c\right) = d$.*

Theorem 53 (Extreme Value Theorem) *Let F be a continuous real $\succ$- function on a non-degenerate interval $[a, b]$. It is provable in RST^{FOL}_{HF} that F attains its maximum and minimum.*

The next step is to introduce in RST^{FOL}_{HF} the concepts of differentiation, integration, power series, etc, and develop their theories. It should now be clear that there is no difficulty in doing so. Since a thorough exposition obviously could not fit in one paper we omit it here, but use some relevant facts in what follows.

We now show that all elementary functions that are relevant to J_2 are available in RST^{FOL}_{HF}. Even though for every real number y in J_2, $\lambda x \in \mathbb{R}.y$ is available in RST^{FOL}_{HF} as a $\succ$-function, not all constant functions on the "real" real line are available in J_2. The reason is that $\lambda x \in \mathbb{R}.y$ does not exists in J_2 for *every* "real" number y (simply since not every "real" real number is available in RST^{FOL}_{HF}).

Definition 54 *The collection of J_2-elementary functions is defined like the standard elementary functions (see, e.g., [26]), replacing the constant functions by J_2-constant functions, which are $\lambda x \in \mathbb{R}.c$ where c is a real in J_2.*

Proposition 55 *Let F be a continuous, strictly monotone real $\succ$-function on a real interval. Then it is provable in RST^{FOL}_{HF} that the inverse function F^{-1} is available in RST^{FOL}_{HF} as a $\succ$-function, and its continuity is provable in RST^{FOL}_{HF}.*

Proposition 56 *All J_2-elementary functions are available in RST^{FOL}_{HF}. Also, any piece-wise defined function with finitely many pieces such that its restriction to any of the pieces is a J_2-elementary function, is available in RST^{FOL}_{HF}.*

5 Conclusion and Further Research

In this paper we showed that a minimal computational framework is sufficient for the development of applicable mathematics. Of course, a major future research task is to implement and test the framework. A critical component of such implementation will be to scale the cost of checking the safety relation. We then plan to use the implemented framework to formalize even larger portions of mathematics, including first of all more analysis, but also topology and algebra.

Another important task is to fully exploit the computational power of our computational theories. This includes finding a good notion of canonical terms, and investigating various reduction properties such as strong normalization. We intend to try also to profit from this computational power in other ways, e.g., by using it for proofs by reflection as supported by well-known proof assistant like Coq [10], Nuprl [12] and Isabelle/HOL [25].

An intuitionistic variant of the system RST^{FOL}_{C}, RST^{iFOL}_{C}, can be also considered. It is based on intuitionistic first-order logic (which underlies constructive counterparts of ZF, like CZF [1] and IZF [7]), and is obtained by adding to RST^{FOL}_{C} the axiom of Restricted Excluded Middle: $\varphi \vee \neg\varphi$, where $\varphi \succ \emptyset$. This axiom is computationally acceptable since it simply asserts the definiteness of absolute formulas. The computational theory RST^{iFOL}_{HF} should allow for a similar formalization of constructive analysis (e.g., [24]).

Further exploration of the connection between our framework and other related works is also required. This includes works on: computational set theory [1,7,9,17,24], operational set theory [16,19], and rudimentary set theory [6,22].

Another direction for further research is to consider larger computational structures. This includes J_ω or even J_{ω^ω} (which is the minimal model of the minimal computational theory based on ancestral logic [4,11]). On the one hand, in such universes standard mathematical structures can be treated as sets. On the other hand, they are more comprehensive and less concrete, thus include more objects which may make computations harder.

References

1. P. Aczel and M. Rathjen. Notes on constructive set theory. Technical Report 40, Mittag-Leffler, 2001.
2. J. Avigad and J. Harrison. Formally verified mathematics. *Communications of the ACM*, 57(4):66–75, 2014.
3. A. Avron. A framework for formalizing set theories based on the use of static set terms. In A. Avron, N. Dershowitz, and A. Rabinovich, editors, *Pillars of computer science*, LNCS 4800, pages 87–106. Springer, 2008.
4. A. Avron. A new approach to predicative set theory. In R. Schindler, editor, *Ways of Proof Theory*, pages 31–63. Onto Series in Mathematical Logic, Verlag, 2010.
5. A. Avron and L. Cohen. Formalizing scientifically applicable mathematics in a definitional framework. *Journal of Formalized Reasoning*, 9(1):53–70, 2016.
6. A. Beckmann, S. R. Buss, and SD. Friedman. Safe recursive set functions. *The Journal of Symbolic Logic*, 80(3):730–762, 2015.
7. M. J. Beeson. *Foundations of constructive mathematics: Metamathematical studies*, volume 6. Springer Science & Business Media, 2012.
8. J. J. J. Campbell, J. C. G. Dos Reis, P. S. M. Wenzel, and V. Sorge. Intelligent computer mathematics. 2008.
9. D. Cantone, E. Omodeo, and A. Policriti. *Set theory for computing: from decision procedures to declarative programming with sets*. Springer, 2001.
10. A. Chlipala. *Certified Programming with Dependent Types*. MIT Press, Cambridge, MA, 2013.
11. L. Cohen and A. Avron. The middle ground–ancestral logic. *Synthese*, pages 1–23, 2015.
12. R. L. Constable, S. F. Allen, M. Bromley, R. Cleaveland, et al. *Implementing mathematics with the Nuprl proof development system*. Prentice Hall, 1986.
13. J. Corcoran, W. Hatcher, and J. Herring. Variable binding term operators. *Mathematical Logic Quarterly*, 18(12):177–182, 1972.
14. K. Devlin. Constructibility, volume 6 of perspectives in mathematical logic, 1984.
15. S. Feferman. Why a little bit goes a long way: Logical foundations of scientifically applicable mathematics. In *PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association*, pages 442–455. JSTOR, 1992.
16. S. Feferman. Operational set theory and small large cardinals. *Information and Computation*, 207(10):971 – 979, 2009.
17. H. Friedman. Set theoretic foundations for constructive analysis. *Annals of Mathematics*, 105(1):pp. 1–28, 1977.
18. R. O. Gandy. Set-theoretic functions for elementary syntax. In *Proc. Symp. in Pure Math*, volume 13, pages 103–126, 1974.
19. G. Jäger and R. Zumbrunnen. Explicit mathematics and operational set theory: Some ontological comparisons. *The Bulletin of Symbolic Logic*, 20(3):275–292, 2014.
20. R. B. Jensen. The fine structure of the constructible hierarchy. *Annals of Mathematical Logic*, 4(3):229–308, 1972.
21. F. D. Kamareddine. *Thirty five years of automating mathematics*, volume 28. Springer, 2003.
22. A.R.D. Mathias, N.J. Bowler, et al. Rudimentary recursion, gentle functions and provident sets. *Notre Dame Journal of Formal Logic*, 56(1):3–60, 2015.
23. N. Megill. *Metamath: A Computer Language for Pure Mathematics*. Elsevier Science, 1997.

24. J. Myhill. Constructive set theory. *The Journal of Symbolic Logic*, 40(03):347–382, 1975.
25. T. Nipkow, L. C. Paulson, and M. Wenzel. *Isabelle/HOL: a proof assistant for higher-order logic*, volume 2283. Springer, 2002.
26. R. H. Risch. Algebraic properties of the elementary functions of analysis. *American Journal of Mathematics*, 101(4):743–759, 1979.
27. P. Rudnicki. An overview of the mizar project. In *Proceedings of the 1992 Workshop on Types for Proofs and Programs*, pages 311–330, 1992.
28. J. T. Schwartz, R. B. Dewar, E. Schonberg, and E. Dubinsky. *Programming with Sets; an Introduction to SETL*. Springer-Verlag New York, Inc., 1986.
29. S. Shapiro. *Foundations without Foundationalism: A Case for Second-Order Logic: A Case for Second-Order Logic*. Oxford University Press, 1991.
30. S. G. Simpson. *Subsystems of second order arithmetic*, volume 1. Cambridge University Press, 2009.
31. N. Weaver. Analysis in J_2. unpublished manuscript, 2005.
32. H. Weyl. *Das Kontinuum: Kritische Untersuchungen über die Grundlagen der Analysis*. W. de Gruyter, 1932.