

POWER TYPES IN EXPLICIT MATHEMATICS?

GERHARD JÄGER

Abstract. In this note it is shown that in explicit mathematics the strong power type axiom is inconsistent with (uniform) elementary comprehension and discuss some general aspects of power types in explicit mathematics.

§1. Introduction. Systems for explicit mathematics were introduced in Feferman [2] and further discussed in Feferman [3]. They provide a natural framework for various constructive and semi-constructive approaches to mathematics and play an important role in present day proof theory. In the latter article Feferman also introduces two forms of power type axioms¹ and presents some basic observations concerning their relationship to several subsystems of explicit mathematics.

By an elegant model-theoretic argument it is shown in Feferman [3] that the weak power type axiom is consistent with elementary comprehension and inductive generation. On the other hand, a simple diagonalization argument yields the inconsistency of the weak power type axiom with join.

The recent article Glass [5] deals with the analysis of the proof-theoretic strength of the weak power type axioms. It is proved there that for many systems of explicit mathematics, for example for EM_0 and $EM_0 + (IG)$ and subsystems which are obtained by restricting the induction principles, the addition of the weak power type axiom does not increase their proof-theoretic strength.

The status of the strong power type axiom has been unclear so far and will be settled in this note. We will prove that the existence of strong power types is inconsistent with (uniform) elementary comprehension in explicit mathematics. Actually, even the existence of the strong power type of the empty type leads to an inconsistent theory.

These results about weak and strong power types help to clarify the role of power type constructions in explicit mathematics. In particular, they give some insight into the ontological aspects of various forms of power types.

§2. The theory BAT and the power type axioms. In this section we introduce the basic theory BAT of application and types. By adding a suitable machinery for dealing with the natural numbers, BAT can be easily extended to the elementary explicit type theory EET of Feferman and Jäger [4] and to theories equivalent to systems of explicit mathematics in the older formulation like EM_0 and $EM_0 + (IG)$.

Received December 21, 1995.

¹Actually, what we call power type axiom in this paper, is called power class axiom in [2, 3].

BAT is an auxiliary theory which is tailored for carrying through the following arguments.

The theory BAT is formulated in the second order language \mathcal{L} about individuals (operations) and types. Individuals can be freely applied to each other; hence self-application of individuals is permitted but not necessarily total. Types are supposed to be collections of individuals.

The language \mathcal{L} comprises individual variables a, b, c, x, y, z, \dots and type variables A, B, C, X, Y, Z, \dots (both possibly with subscripts). In addition there are the individual constants \mathbf{k}, \mathbf{s} (combinators), $\mathbf{p}, \mathbf{p}_0, \mathbf{p}_1$ (pairing and unpairing) and \mathbf{c}_e ($e < \omega$), the meaning of which will be explained later.

The *individual terms* ($r, s, t, r_0, s_0, t_0, \dots$) of \mathcal{L} are generated from the individual variables and constants by means of term application: (i) All individual variables and individual constants are individual terms; (ii) if s and t are individual terms, then so is $(s \cdot t)$.

In the following we often abbreviate $(s \cdot t)$ simply as (st) or st and adopt the convention of association to the left so that $s_1 s_2 \dots s_n$ stands for $(\dots (s_1 \cdot s_2) \dots s_n)$. Moreover, we write (t_1, t_2) for $\mathbf{p}t_1 t_2$ and (t_1, t_2, \dots, t_n) for $(t_1, (t_2, \dots, t_n))$.

The relation symbols of \mathcal{L} are the unary \downarrow as well as the binary $=, \in$ and \mathfrak{R} which are used to build the following *atomic formulas* of \mathcal{L} : $t \downarrow$, $(s = t)$, $(s \in A)$, $(A = B)$ and $\mathfrak{R}(s, A)$. Then the atomic formula $t \downarrow$ is read *t is defined* or *t has a value*, and the formula $\mathfrak{R}(s, A)$ is used to express that the individual s *represents* the type A or is a *name* of A .

The *formulas* ($\varphi, \chi, \psi, \varphi_0, \chi_0, \psi_0, \dots$) of \mathcal{L} are generated from these atomic formulas by the usual logical connectives and quantification over individuals and types. A formula of \mathcal{L} is called *elementary* if it contains neither the relation symbol \mathfrak{R} nor bound type variables.

To simplify the notation, we write $(s \simeq t)$ for $((s \downarrow \vee t \downarrow) \rightarrow (s = t))$ and $(A \subset B)$ for $(\forall x)(x \in A \rightarrow x \in B)$. Often we also make use of the shorthand notations $(\exists x \in A)\varphi(x)$, $(\forall x \in A)\varphi(x)$, $(\exists X \subset A)\varphi(X)$ and $(\forall X \subset A)\varphi(X)$, whose meaning is obvious.

The logic of BAT is the (classical) *logic of partial terms* (cf. Beeson [1] or Troelstra and van Dalen [6]), including equality axioms for both sorts. The first order part of BAT comprises the usual axioms which state that the individuals form a partial combinatory algebra and permit pairing and projections.

I. Partial combinatory algebra, pairing and projections.

- (1) $\mathbf{k}xy = x$,
- (2) $\mathbf{s}xy \downarrow \wedge \mathbf{s}xyz \simeq xz(yz)$,
- (3) $(x, y) \downarrow \wedge \mathbf{p}_0(x, y) = x \wedge \mathbf{p}_1(x, y) = y$.

The first group of second order axioms of BAT claim that each type has a name, that different names represent different types and that types are extensional. On the other hand, the representation of types via names is considered intensionally.

II. Explicit representation and extensionality.

- (4) $(\forall X)(\exists y) \mathfrak{R}(y, X)$,
- (5) $\mathfrak{R}(a, B) \wedge \mathfrak{R}(a, C) \rightarrow B = C$,
- (6) $(\forall x)(x \in A \leftrightarrow x \in B) \rightarrow A = B$.

BAT also permits uniform elementary comprehension. In connection with the naming process in this scheme it is convenient to make use of the following conventions: (i) We assume that there is some arbitrary but fixed standard assignment of Gödel numbers to the formulas of \mathcal{L} . (ii) We assume further that v_0, v_1, \dots and V_0, V_1, \dots are arbitrary but fixed enumerations of the individual and type variables. If φ is an \mathcal{L} formula with no other individual variables than v_0, \dots, v_m and no other type variables than V_0, \dots, V_n and if $\vec{x} = x_0, \dots, x_m$ and $\vec{Y} = Y_0, \dots, Y_n$, then we write $\varphi[\vec{x}, \vec{Y}]$ for the \mathcal{L} formula which results from φ by a simultaneous replacement of vi by x_i and V_j by Y_j ($0 \leq i \leq m, 0 \leq j \leq n$). (iii) Moreover, if $\vec{x} = x_0, \dots, x_n$ and $\vec{X} = X_0, \dots, X_n$, then $\mathfrak{R}(\vec{x}, \vec{Y})$ stands for $\bigwedge_{i=1}^n \mathfrak{R}(x_i, X_i)$.

III. Elementary comprehension. Let $\varphi[x, \vec{y}, \vec{Z}]$ be an elementary \mathcal{L} formula and e its Gödel number; then we have:

$$(ECA.1) \quad (\exists X)(\forall x)(x \in X \leftrightarrow \varphi[x, \vec{a}, \vec{B}]),$$

$$(ECA.2) \quad \mathfrak{R}(\vec{b}, \vec{B}) \wedge (\forall x)(x \in A \leftrightarrow \varphi[x, \vec{a}, \vec{B}]) \rightarrow \mathfrak{R}(c_e(\vec{a}, \vec{b}), A).$$

Thus the constants c_e make the naming of types generated by elementary comprehension uniform in the parameters of the defining formula. Of course this process depends on the chosen Gödel numbering and enumerations of variables of \mathcal{L} , but this is obviously not a relevant restriction.

REMARK 1. If we omit the axiom (ECA.2) in the previous comprehension scheme, then we obtain the non-uniform version of elementary comprehension.

The basic theory BAT of application and types is defined to be the \mathcal{L} theory which consists of the axioms (I)–(II) plus elementary comprehension (ECA.1) and (ECA.2). The restriction of BAT to non-uniform elementary comprehension, i.e., BAT without (ECA.2), is called nu-BAT.

Strictly speaking, the only type terms in the language \mathcal{L} are the type variables. However, if $\varphi[x, \vec{a}, \vec{B}]$ is an elementary \mathcal{L} formula, then it is often convenient to write $\{x : \varphi[x, \vec{a}, \vec{B}]\}$ for the type whose existence is established by (ECA.1), and we will freely make use of this more flexible notation.

Now we turn to the power type axioms. The *weak power type axiom* (Pow)⁻ is the universal closure of the following formula:

$$(Pow)^- \quad (\exists X)[(\forall y \in X)(\exists Y \subset A)\mathfrak{R}(y, Y) \wedge (\forall Y \subset A)(\exists y \in X)\mathfrak{R}(y, Y)].$$

It states that for every type A there exists a type X which consists of names for subtypes of A and contains at least one name for each subtype of A . This is in sharp contrast to the *strong power type axiom* (Pow)⁺ which claims that for each type A there exists a type X which consists exactly of all names of all subtypes of A . It is the universal closure of the following formula:

$$(Pow)^+ \quad (\exists X)(\forall y)(y \in X \leftrightarrow (\exists Y \subset A)\mathfrak{R}(y, Y)).$$

The uniform version of the weak (strong) power type axioms is easily obtained by adding further individual constant π and an axiom which says that πa is a name of a weak (the strong) power type of A provided that a is a name of A .

REMARK 2. As mentioned above, it follows from an argument in Feferman [3] that the theory $\text{BAT} + (\text{Pow})^-$, even extended by the usual axioms about the natural numbers and inductive generation (IG), is consistent. It is also shown there that the weak power type axiom is inconsistent with the join axiom of explicit mathematics. In addition, Glass [5] provides a proof-theoretic analysis of the weak power type axiom and shows that adding $(\text{Pow})^-$ to many systems of explicit mathematics (without join) does not increase their proof-theoretic strength.

§3. The names of the empty type. In this section we show that the strong power type axiom is inconsistent with BAT. This follows from the more general result that the names of the empty type provably do not form a type in BAT.

It is an obvious consequence of elementary comprehension that there exists the empty type $\{x : x \neq x\}$, which we simply denote as \emptyset . Accordingly, $\mathfrak{R}(a, \emptyset)$ expresses that the individual a is a name of the empty type.

THEOREM 3. *The theory BAT proves that there is no type which consists exactly of the names of the empty type, i.e.,*

$$\text{BAT} \vdash \neg(\exists X)(\forall x)(x \in X \leftrightarrow \mathfrak{R}(x, \emptyset)).$$

PROOF. We begin with the simple observation that there exists an individual term t which has the property

$$\text{BAT} \vdash \mathfrak{R}(a, A) \rightarrow \mathfrak{R}(ta, \{x : x = a \wedge x \notin A\}).$$

Here we make use of the fact that BAT comprises uniform elementary comprehension; in nu-BAT a similar argument seems not to be possible.

Now we work informally in BAT and assume that the names of the empty type form a type. This means that there exists a type E so that

$$a \in E \leftrightarrow \mathfrak{R}(a, \emptyset)$$

for all individuals a . Elementary comprehension with type parameter E then yields the existence of the type $R := \{x : tx \notin E\}$, and we have

$$a \in R \leftrightarrow \neg\mathfrak{R}(ta, \emptyset)$$

for all individuals a . Finally let r be a name of R . Then it is easy to verify that $r \in R$ if and only if $r \notin R$. Hence we have a contradiction, and the names of the empty type cannot form a type. \neg

This theorem immediately implies that the empty type cannot possess a strong power type if one works in the framework of BAT. Hence we obtain the following corollary.

COROLLARY 4. *The strong power type axiom is provably false in the theory BAT, i.e.,*

$$\text{BAT} \vdash \neg(\text{Pow})^+.$$

§4. Final remarks. The previous corollary makes it clear that the strong power type axiom is not justified in explicit mathematics. To be precise: The strong power type axiom is not justified in systems of explicit mathematics which use the uniform version of elementary comprehension. It is still open whether the strong power type axiom is compatible with non-elementary comprehension, i.e., whether $\text{nu-BAT} + (\text{Pow})^+$ is consistent.

The relevance of this question depends on the estimation whether one would be willing to sacrifice the uniformity of elementary comprehension for the existence of strong power types. However, uniform elementary comprehension is very much in the spirit of explicit mathematics and contained in nearly all systems, which are discussed in the literature.

Also the status of the weak power type axiom in explicit mathematics is rather delicate and one may argue whether it agrees with the philosophical principles inherent in explicit mathematics. It only claims that every type A has a weak power type B which consists of names for subtypes of A and contains at least one name for each subtype of A . However, this B is not further specified and there is no criterion to find out whether the name of a subtype of A belongs to B .

Technically speaking, the problem is as follows. Suppose that B is a weak power type of A and that we know that c is a name of a subtype of A . Then we cannot assume that c belongs to B . So there is no way to really exploit the existence of weak power types, and therefore it is not surprising that it does not increase the mathematical strength of explicit mathematics.

To sum up: Neither the strong nor the weak power type axiom seem to provide a convincing approach to power types in explicit mathematics. Therefore there remains the question whether there is an alternative, possibly intermediate form, which is more interesting.

REFERENCES

- [1] M. J. BEESON, *Foundations of constructive mathematics*, Springer-Verlag, 1985.
- [2] S. FEFERMAN, *A language and axioms for explicit mathematics*, *Algebra and logic* (J. N. Crossley, editor), Lecture Notes in Mathematics, no. 450, Springer-Verlag, 1975.
- [3] ———, *Constructive theories of functions and classes*, *Logic colloquium '78* (M. Boffa, D. van Dalen, and K. McAloon, editors), North-Holland, 1979.
- [4] S. FEFERMAN and G. JÄGER, *Systems of explicit mathematics with non-constructive μ -operator, Part II*, *Annals of Pure and Applied Logic*, vol. 79 (1996).
- [5] TH. GLASS, *On power set in explicit mathematics*, this JOURNAL, vol. 61 (1996), no. 2, pp. 468–489.
- [6] A. S. TROELSTRA and D. VAN DALEN, *Constructivism in mathematics*, vol. I, North-Holland, 1988.

INSTITUT FÜR INFORMATIK UND ANGEWANDTE MATHEMATIK
 UNIVERSITÄT BERN
 NEUBRÜCKSTRASSE 10
 CH-3012 BERN, SWITZERLAND

E-mail: jaeger@iam.unibe.ch