U N I V E R S I T A' D E G L I S T U D I D I S I E N A

DIPARTIMENTO DI MATEMATICA

Via del Capitano, 15

53100 SIENA

Alessandro BERARDUCCI[1] Benedetto INTRIGLIA[2]

A note on coding techniques in $I\Delta_0$

Rapporto Matematico N° 231

GENNAIO 1991

[1]Università di L'Aquila, Dipartimento di Matematica Pura e applicata, Via Vetoio, 67010 Coppito, L'Aquila, Italy
[2]Viale Parioli 41,00197 Roma, Italy.

A note on coding techniques in $I\Delta_0$

by

Alessandro Berarducci[1] and Benedetto Intrigila[2]

$I\Delta_0$ is the subtheory of Peano Arithmetic (PA) where the axiom scheme of induction is restricted to Δ_0-formulas, namely formulas in which every quantification has the form $\exists x \leq t$ or $\forall x \leq t$, where t is a term in the language $0, 1, +, \cdot$ not containing x. It is known that $y = x^n$ ($x, y \in \mathbb{N}$) is definable by a Δ_0-formula in such a way that the basic properties $x^0=1$, $x^{n+1}=xx^n$ are provable in $I\Delta_0$ (cfr. [Be], [GD], [Pu]). In this paper we use the notion of "pseudo-sequence" to present a slight variant of the Δ_0-definition of $y=x^n$ due to Paris (cfr. [GD]) which does not use the chinese remainder theorem. We show that with minor changes the proposed Δ_0-definition generalizes to some cases of matrix exponentiation. As an application we give a Δ_0-definition of the graph of the Fibonacci function, as well as a Δ_0-definition of exponentiation in the ring of gaussian integers. We then prove the basic properties of these functions in $I\Delta_0$.

1. Sequences and pseudo-sequences.

Pudlák [Pu] defined a sequential theory to be a theory T which interprets a reasonable fragment of arithmetic ($I\Delta_0$ or even Robinson's Q) and for which there exists a formula of three variables $s[t] = y$ such that, letting N be the domain of the natural numbers in T, T proves:

[1] Universita' di L'Aquila, dipartimento di Matematica Pura e Applicata, Via Vetoio, 67010 Coppito, L'Aquila, Italy.

[2] Viale Parioli 41, 00197 Roma, Italy.

a) $\forall t \in N \ \forall s \ \exists! y \ (s[t] = y)$

(so we can denote by $s[t]$ the unique y such that $s[t] = y$);

b) $\forall t \in N \ \forall y \ \forall s \ \exists s'$ such that $s'[t] = y \wedge \forall n \in N \ (n \neq t \rightarrow s'[n] = s[n])$.

Intuitively $s[t] = y$ means that y is the t-th element of the sequence coded by s. We modify this notion by dropping the requirement that we can always compute the t-th element $s[t]$ of a sequence s. In exchange we require that sequences can be concatenated, and sequences of one element can always be formed. This can be made precise using the notion of "pseudo-sequence":

1.1. Pseudo-sequences.

Let $M \models T$ and let Atom(x) be a formula (possibly containing parameters from M). We say that M has a coding of pseudo-sequences from Atom if there are formulas with parameters from M, Seq(x) (sequence), $x*y = z$ (concatenation), $\langle x \rangle = y$ (singleton), and $\langle\rangle = y$ (empty sequence), such that, letting f, g, h range over Seq^M, and a, b, x, y range over $Atom^M$, the universal closure of the following is true in M:

1) Concatenation: $\exists ! h \in Seq \ (f*g = h)$;

2) singleton: $\exists ! f \in Seq \ (\langle x \rangle = f)$;

3) empty sequence: $\exists ! f \in Seq \ (\langle\rangle = f)$;

4) Monoid axiom: (Seq, *) is a monoid having $\langle\rangle$ as its neutral element and satisfying the cancellation laws: $f*g = f*h \rightarrow g = h$ and $g*f = h*f \rightarrow g = h$;

5) $\langle x \rangle = \langle y \rangle \rightarrow x = y$;

6) $f \neq \langle\rangle \rightarrow \exists ! x \ \exists ! g \ (f = \langle x \rangle * g)$;

7) $f \neq \langle\rangle \rightarrow \exists ! x \ \exists ! g \ (f = g * \langle x \rangle)$;

1.2. Smullian representation.

We recall that the Smullian representation in base B of a natural number $f \neq 0$ is given by a representation of f in the form $f = \sum_{i \leq n} a_i B^i$ where $1 \leq a_i \leq B$. We think at f as a code for the sequence of the a_i's. The reason we use the Smullian representation rather than the ordinary representation in base B is to avoid problems with leading zeros (cfr. [WP]).

1.3. Theorem. Let $M \models I\Delta_0$ and let $B \in M$ be a power of 2. Then the Smullian representation in base B gives, in M, a Δ_0-definable coding of pseudo-sequences of elements < B.

Proof. Define $Seq(f) \leftrightarrow f=f$ and $Atom(x) \leftrightarrow x < B$. So Atom is a formula with parameter $B \in M$. The empty sequence is 0; $\langle x \rangle = x+1$; For $f \neq \langle\rangle$, define $size_B(f) = U \leftrightarrow$ U is a power of B and $U - 1 \leq f \cdot (B - 1) < BU - 1$. So if $f = \sum_{i \leq n} a_i B^i$ with $1 \leq a_i \leq B$ and $n > 0$, then $size_B(f) = B^{n+1}$. If $f = \langle\rangle = 0$, then $size_B(f) = 1$. Define concatenation $f*g$ as follows: $f*g = f + U \cdot g$ where $U = size(f)$. Note that, since B is a power of the prime number 2, then (following an idea of Pudlák [Pu]) we can express "U is a power of B" in a Δ_0-way by: 2 is the only prime divisor of U and $B-1|U-1$. It is clear that these definitions constitute a coding of pseudo-sequences in the standard model. Moreover it can be verified that (1) to (7) can be proved by Δ_0-induction, so we have a coding of pseudo-sequences of elements < B in any model M of $I\Delta_0$. QED.

1.4. Definition. We call "B-sequences", the pseudo-sequences of elements < B coded according to the Smullian representation.

To make explicit the dependence of B in the definition of concatenation we will write sometime $[f^*g]_B$ instead of f^*g.

Note that we can express in $I\Delta_0$ the fact that two B-sequences f, g have the same length by comparing $size_B(f)$ and $size_B(g)$, so without actually computing the lengths of f and g (the length is approximately the logarithm of the size).

1.5. Theorem. $I\Delta_0$ has a Δ_0-definition of pseudo-sequences of arbitrary objects, not necessarily $\leq B$, that is we can find a Δ_0-definition of pseudo-sequences where $Atom(x) \leftrightarrow x=x$.

To prove the theorem we use Smullian representations in base 4 and we interpret a 4-sequence as a sequence of words separated by commas represented by the digit 4. This is made precise in the following definitions.

1.6. Definition. We say that W is a "word" if W does not contain 4 in its Smullian representation in base 4.

1.7. Definition. We say that S is a sequence of words, if the most significant digit in the Smullian representation in base 4 of S is 4, that is S has the form $[A^*\langle 3\rangle]_4$ (recalling that $\langle x\rangle = x+1$). We say that the word W occurs in S if S has the form $S = [A^*\langle 3\rangle^*W^*\langle 3\rangle^*B]_4$ or $S = [W^*\langle 3\rangle^*B]_4$.

A number $x = \sum_{i<n} a_i 2^i$ $(1 \leq a_i \leq 2)$ can be encoded into the word $W(x) = \sum_{i<n} a_i 4^i$, from which we can form the one-word sequence $S = [W(x)^*\langle 3\rangle]_4$. S can then be concatenated as a 4-sequence to other word-sequences. Note

that the map $x \mapsto W(x)$ is into the set of words but not onto since $W(x)$ never has the digit $3 = \langle 2 \rangle$.

To define $y = W(x)$ in a Δ_0-way we first note that the relation $size_2(u)^2 = size_4(v)$ expresses the fact that the length of u in the Smullian representation in base 2, is equal to the length of v in the Smullian representation in base 4. Using this relation we can compare the corresponding coefficients of x and y in base 2 and 4 respectively and impose that they are equal. In other words we can define:

1.8. Definition. For $x \neq 0$ we define $y = W(x)$ iff $size_2(x)^2 = size_4(y)$ and whenever we have a decomposition $x = [u*v]_2$, $y = [z*w]_4$ with $size_2(u)^2 = size_4(z) > 0$, then u and z have the same leading digit $i \in \{0, 1\}$, i.e. we can write $u = [h*\langle i \rangle]_2$ and $z = [g*\langle i \rangle]_4$. For $x = 0$ we define $W(x) = \langle\rangle = 0$.

It is easy to prove by Δ_0-induction that there is always a unique $y \leq 4x^2$ with $y = W(x)$. Morever x is uniquely determined by y.

We define concatenation $u*v$ of two sequences u, v of words as their concatenation as 4-sequences. The new singleton operation is given by $x \mapsto [W(x)*\langle 3 \rangle]_4$. With this setting we get, in $I\Delta_0$, a notion of pseudo-sequences of arbitrary elements, i.e. we can set $Atom(x) \leftrightarrow x=x$, thus proving theorem 1.5. It can actually be proved that $I\Delta_0$ is sequential also in the sense of [Pu], but this requires more involved techniques that basically amount to "count" the number of words in a pseudo-sequence (this can be done using results from [PWW]). However pseudo-sequences are already enough to give a Δ_0-definition of $y=x^n$ as we next show.

2. Exponentiation.

We want to give a Δ_0-definition of the relation $y = x^n$. The idea is to employ the relations $x^{2n} = (x^n)^2$ and $x^{2n+1} = (x^n)^2 \cdot x$.

2.1. Definition. Let $x > 1$. We say that C is a computation of $y = x^n$ if C is a sequence of words (in the sense of def. 1.6) such that, letting $J(x, y)$ be a suitable word encoding the pair (x, y), we have:

1) Each word occurring in C has the form $J(u,i)$ (the idea is that $u = x^i$);
2) If $J(u, i)$ and $J(v, j)$ are consecutive words in C (i.e. C has the form $A*J(u, i)*\langle 3\rangle*J(v, j)*\langle 3\rangle*B$ where A and B are sequences of words), then either $v = u^2$ and $j = 2i$, or $v = u^2x$ and $j = 2i+1$.
3) The last word in C is $J(y, n)$ and the first one is $J(1, 0)$.

A simple definition of J is given by $J(u, i) = [W(u)*\langle 2\rangle*W(i)]_4$. (Another possibility is to use a bijective pairing function.) A crucial observation to make is that, for $x > 1$, the computation C of $y = x^n$ can be bounded by a (standard) polynomial in y. We will see later that we can take a polynomial $P(y) = x^{13} + R$ of degree 13. We can therefore give the following Δ_0-definition of $y = x^n$.

2.2. Definition. For $x > 1$ define $y = x^n \leftrightarrow \exists C \leq P(y)$ "C is a computation of $y = x^n$". For $x = 0, 1$ define $y = x^n$ by cases in the usual way.

The unicity of computations can easily be proved by Δ_0-induction, namely one proves by Δ_0-induction on C that if C is a computation of $y = x^n$ and $C' \leq C$ is a computation of $z = x^n$, then $C = C'$ (hence $y = z$). The polynomial P must be chosen to be big enough so that if $C \leq P(y)$ is a computation of $y =$

x^n, then there are computations $C' \le P(y^2)$ of $y^2 = x^{2n}$ and $C'' \le P(y^2x)$ of $y^2x = x^{2n+1}$. In other words computations can always be prolonged one more step remaining within the polynomial bound.

For a precise evaluation of the polynomial P we must extimate the sizes of the computation C' and C''. By definition we must have $C' = C^*W(y^2)^*\langle 2\rangle^*W(2n)^*\langle 3\rangle$, $C'' = C^*W(y^2x)^*\langle 2\rangle^*W(2n+1)^*\langle 3\rangle$. To give a polynomial bound of C and C' we need the following properties of the $size_B$ function (provable by Δ_0-induction):

2.3. Proposition.

i) $size_B(f^*g) = size_B(f) \cdot size_B(g)$;

ii) $f \le B \cdot size_B(f)$;

iii) $f \ne 0 \to size_B(f) \le Bf$;

iv) $size_2(x) \le 4size_4(x)$;

v) $size_4(W(x)) = size_2(x)^2 \le 16size_4(x)^2$.

For n and y big enough (i.e. $n, y \ge K$ for a suitable $K \in \mathbb{N}$ which could be easily computed in advance) the words $W(y^2)^*\langle 2\rangle^*W(2n)^*\langle 3\rangle$ and $W(y^2x)^*\langle 2\rangle^*W(2n+1)^*\langle 3\rangle$ are bounded by $W(y^3)^*W(y^3)$, hence $size_4(C)$ and $size_4(C')$ are bounded by $size_4(C) \cdot size_4(W(y^3))^2$ which is $\le size_4(C) \cdot 4^4 size_4(y^3)^4 \le 4^9Cy^{12}$. Hence $C', C'' \le 4^{10}Cy^{12}$ (for $y \ge K$). It is now clear that a polynomial of the form $y^{13} + R$, with $R \in \mathbb{N}$ big enough, will do the job. We have thus proved that computations can always be extended one more step, that is:

2.4. Lemma. $y = x^n \to (y^2 = x^{2n}$ and $y^2x = x^{2n+1})$.

We also need the following "inversion formulas":

2.5. Lemma. $u = x^{2n} \rightarrow \exists y\, (u = y^2 \wedge y = x^n)$ and $u = x^{2n+1} \rightarrow \exists y\, (u = y^2 x \wedge y = x^n)$.

Using these facts one can prove the following:

2.3. Theorem($I\Delta_0$). $y = x^n \rightarrow\ yx = x^{n+1}$ and $u = x^{n+1} \rightarrow \exists y\, (y = x^n \wedge u = yx)$.

Proof. We distinguish two cases:

Case 1. Suppose $n > 0$ is even, say $n = 2m$. Let $y = x^{2m}$. Then $\exists$ z such that $y = z^2$ and $z = x^m$. It follows that $z^2 x = x^{2m+1}$. Hence $yx = x^{2m+1}$. Now suppose $u = x^{2m+1}$, then $u = z^2 x$ with $z = x^m$. Hence $z^2 = x^{2m}$. So u has the form yx with $y = x^{2m} = x^n$.

Case 2. n odd. Similar. QED.

The above theorem implies that in every model of $I\Delta_0$, the exponential function can be defined on a initial segment of the model closed under successor (and containing 0).

Note that, once we have exponentiation, we can define in a Δ_0-way "y is the n-th digit in the Smullian representation in base B of x". Thus we can now refer to the n-th element of a B-sequence and compute its length. Moreover we can speak of B-sequences where B is not necessarily a power of 2 (such a requirement was introduced in order to be able to express "U is a power of B"; with exponentiation at our disposal this is no longer a problem.)

3. Matrix exponentiation.

The advantage of the above approach is that it can be used with minor changes to define exponentiation of certain $k \times k$ matrices with integer entries (k standard). For simplicity consider a 2×2 matrix A. Let $J(a, b, c, d, i)$ be a word encoding the matrix $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ and the number i. To simplify the notation we also write $J(D, i)$ instead of $J(a, b, c, d, i)$ when D is the matrix $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$.

3.1. Definition. A computation C of $B = A^n$ consists of a sequence of words of the form $J(a, b, c, d, i)$ (whose intended meaning is $\begin{pmatrix} a & b \\ c & d \end{pmatrix} = A^i$). In complete analogy with the case of integer exponentiation we require that if $J(D, i)$ and $J(E, j)$ are two consecutive words of C, then either $j = 2i$ and $E = D^2$, or $j = 2i+1$ and $E = D^2A$. We also require that the last word of C is $J(B, n)$ and the first one is $J(I, 0)$ where I is the identity matrix. Note that "C is a computation of $B = A^n$" can be defined by a Δ_0-formula.

A convenient definition of $J(a, b, c, d, i)$ is given by the word $[W(a)*\langle 2\rangle*W(b)*\text{-}\langle 2\rangle*W(c)*\langle 2\rangle*W(d)*\langle 2\rangle*W(i)]_4$. Sequences of consecutive words are separated by $\langle 3\rangle$ (as usual). We have: $B = A^n \leftrightarrow \exists\, C$ (C is a computation of $B = A^n$). It is easy to see that C can be bounded by a polynomial in α^n where α is an integer coding the matrix A (note that α is polynomially related to the maximum of the entries of A). So in the particular case when α^n can be bounded by a polynomial in β, where β is a code for B, then we can reason as in the case of integer exponentiation to give a polynomial bound on C in terms of (a code for) B, thus obtaining a Δ_0-definition.

However in cases like $\begin{pmatrix} 1 & n \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}^n$ the maximum of the entries of the matrix $B = \begin{pmatrix} 1 & n \\ 0 & 1 \end{pmatrix}$ grows only linearly with n and the above method fails. So it is natural to ask: is $B = A^n$ always Δ_0-definable?

While an answer to the above question is still lacking, we can make use of the favorable cases of matrix exponentiation to give a Δ_0-definition of the relation $y = F(n)$, where $F(n)$ is the n-th Fibonacci number. A similar method will yield a Δ_0-definition of exponentiation in the ring of gaussian integers.

3.2. Fibonacci numbers.

We recall the definition: $F(0) = 0$, $F(1) = 1$, $F(n+2) = F(n+1) + F(n)$. It is well known that this can be reduced to matrix exponentiation as follows: first put the recurrence relation for $F(n)$ in the form of a 2×2 system of linear equations

$$\begin{cases} F(n+1) = F(n) + G(n) \\ G(n+1) = F(n) \end{cases}$$

with the initial conditions $F(0) = 0$, $G(0) = 1$. Next put the system in matrix form: $\begin{pmatrix} F(n+1) \\ G(n+1) \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} F(n) \\ G(n) \end{pmatrix}$. Deduce that $\begin{pmatrix} F(n) \\ G(n) \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n \begin{pmatrix} 0 \\ 1 \end{pmatrix}$. Thus to give a Δ_0-definition of $y = F(n)$ it is enough to give one of $B = A^n$ where $A = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$. Since the entries of A^n grow exponentially with n, it is not difficult give a polynomial bound on the computation C needed to compute $B = A^n$, thus obtaining a Δ_0-definition of $B = A^n$ for this particular matrix A.

Some details: suppose that C is a computation of $B = \begin{pmatrix} a & b \\ c & d \end{pmatrix} = A^n$ (where $A = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$).We can prove by Δ_0-induction on C that $a \geq b = c \geq d$, so $a = \max\{a, b, c, d\}$. Consider the computations C' and C'' of $B' = A^{2n}$ and $B'' = A^{2n+1}$ respectively. Then $C' = C * J(B', 2n) * \langle 3 \rangle$ and $C'' = C * J(B'', 2n+1) * \langle 3 \rangle$ where B' and B'' are the matrices B^2 and B^2A respectively. Letting a' and a'' be the maximum of the entries of B' and B'' respectively, we have $a^2 \leq a' \leq a'' \leq 4a^2$. Thus $J(B', 2n)*\langle 3\rangle$ and $J(B'', 2n+1)*\langle 3\rangle$ are bounded, for n big enough, by $W(4a^2)*W(4a^2)*W(4a^2)*W(4a^2)*W(4a^2)$ (a copy of $W(4a^2)$ for each entry of the matrix, plus one more copy to majorize the remaining ingredients of $J(B', 2n)*\langle 3\rangle$ and $J(B'', 2n+1)*\langle 3\rangle$). It follows that $size_4(C')$ and $size_4(C'')$ are $\leq size_4(C)\cdot size_4(W(4a^2))^5 \leq size_4(C)\cdot 4^{20}a^{10}$. Thus C and C' are $\leq 4^{22}Ca^{10}$. If we now choose a polynomial $P(a) = a^{11}+R$ with $R\in \mathbb{N}$ big enough, it follows, from the assumption $C \leq P(a)$, that $C' \leq P(a')$ and $C'' \leq P(a'')$ respectively. So if we define, for n big enough, $B = A^n \leftrightarrow \exists\, C$ ($C \leq$ (maximum of $B)^{11}+R$ and C is a computation of $B = A^n$), we have that $\{n \mid \exists\, B\, (B = A^n)\}$ defines, in every model of $I\Delta_0$, an initial segment J closed under successor and containing 0. Moreover on this initial segment $A^{n+1} = A^nA$. From this we obtain a Δ_0-definition of $y = F(n)$ such that for all $n \in J$, $F(n+2) = F(n+1) + F(n)$.

3.3. Gaussian integers.

Gaussian integers can be identified with matrices of the special form $\begin{pmatrix} a & b \\ -b & a \end{pmatrix}$ where $\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$ represents the imaginary unit. Any gaussian integer has norm 0, 1 or ≥ 2. If A is a matrix corresponding to a gaussian integer with norm ≥ 2, then the norm of A^n is $\geq 2^n$ and we can give a Δ_0-definition of $B = A^n$ using the techniques of the previous section. The remaining cases correspond to the gaussian integers 1, -1, i, -i, and 0. In these cases A^n is cyclic with period 4 and we can define exponentiation in a trivial manner.

References

[Be] J. H. Bennet, On Spectra, Ph.D. dissertation, Princeton University 1962.

[GD] H. Gaifman and C. Dimitracopoulos, Fragments of Peano's Arithmetic and the MRDP theorem, in Logic and Algorithmic, Geneve 1982, 187-296.

[PWW] J. B. Paris, A. J. Wilkie, and A. R. Woods, Provability of the pigeonhole principle and the existence of infinitely many primes, Journal of Symbolic Logic, vol. 53, n. 4, 1988, pp. 1235-1244.

[Pu] P. Pudlák, A definition of Exponentiation by a bounded Arithmetical formula, Commentationes Mathematicae Universitatis Carolinae, 24, 4; 1983.

[WP] A. J. Wilkie and J. B. Paris, On the scheme of induction for bounded arithmetical formulas, Annals of Pure and Applied Logic, vol. 35 (1987), pp. 261-302.