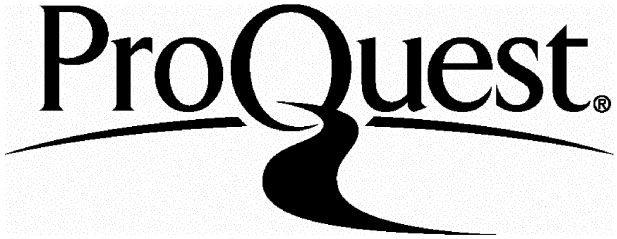
DECIDABLE CLASSES OF RECURSIVE EQUATIONS

by

R. D. Lee

This thesis is submitted for the degree of Doctor of Philosophy of the University of Leicester

1969



Acknowledgements

I should like to thank my supervisor, Professor R.L. Goodstein, for his help and constant encouragement. I should also like to thank George Rousseau, a contemporary at Leicester University for many helpful discussions.

For the typing of the thesis I am grateful to Mrs. J. Hymas and Mrs. L. Hodgson of Essex University whose work speaks for itself and I must thank my wife whose help and support, though strictly non-mathematical, has been invaluable.

Finally, I should like to thank the Science Research Council for financial assistance for three years.

Abstract

Many different formalisations of recursive arithmetic have been proposed, and this thesis is concerned mainly with the system proposed by R.L. Goodstein and known as the Axiom - Free Equation Calculus.

As with all other formal systems of arithmetic with sufficient content, the system is incomplete and recursively undecidable. The interesting questions lie in the completeness and decidability, or otherwise, of fragments of the system. I attempt to answer some of these questions. It happens that some of the problems lead to well known questions in the theory of diophantine equations namely, Hilbert's 10th Problem, The Undecidability of Exponential Diophantine Equations, and the Integer Linear Programming Problem.

In 1943 Kalmar proposed a set of functions called elementary functions, and Ilona Bereczki showed effectively that the class of equations $F = 0$, where F is any elementary function, is undecidable. The class of functions given by Kalmar was, variables, $1, +, ., |a - b|, [a/b], \sum_{y=w}^{z}, \prod_{y=w}^{z}$, but it can easily be shown that this is the same as those formed by composition from $+, ., \dot{-}, \sum_{y=w}^{z}, \prod_{y=w}^{z}$. This latter definition is the one we use.

In his paper, A Decidable Fragment of Recursive Arithmetic, Goodstein showed the class of equations $F = 0$ where F is any function formed by composition from $x + y$, $x.y$ and $1 \dot{-} x$ is decidable.

So I have attempted to extend Goodstein's result with the upper bound provided by the undecidability of the elementary equations. The main results I have obtained are

1. If F is any function formed by composition from $x + y$, $x.y$, $1 \dot{-} x$, $x \dot{-} 1$, $\sum_{y=w}^{z}$, $\prod_{y=w}^{z}$, then $F = 0$ is decidable, and furthermore the provability in the equation calculus of $F = 0$ is decidable and that this class of equations is complete.

2. If F,G are any functions formed from $x + y$, $x.y$, $1 \dot{-} x$, $x \dot{-} 1$, by composition, then the class of equations $F = G$ is decidable.

3. If F,G are any functions formed by composition from $x + y$, $x \dot{-} y$ then the class of equations $F = G$ is decidable.

4. If F.G are any functions formed by composition from $x + y$, $x \dot{-} y$, $x.y$, then the class of equations $F = G$ is decidable if and only if Hilbert's 10th Problem is decidable.

5. If F,G are any functions formed by composition from $x + y$, $x.y$, $\prod_{y=w}^{z}$ then the class of equations $F = G$ is undecidable.

6. Presburger's Algorithm can be used to solve the Integer Linear Programming Problem - the problem was not solved until 1958.

Contents

Chapter 1

§1 Introduction

Many different formalisations of arithmetic have been proposed and this thesis is concerned mainly with the development of recursive arithmetic due to R.L. Goodstein and known as The Axiom-Free Equation Calculus [6] .

As with all other formal systems of arithmetic with sufficient content, the system is incomplete and recursively undecidable. The interesting questions lie in the completeness and decidability or otherwise of fragments of the system. We attempt to answer some of these questions and in the process provide decision procedures for identities between certain number theoretic functions and discuss the relationship these problems have to classical problems in diophantine equations such as Hilbert's 10th problem, the integer linear programming problem and the undecidability of the class of exponential diophantine equations. [2]

§2 Results so far obtained

In his paper "A Decidable Fragment of Recursive Arithmetic", Goodstein shows that if F is any function formed by composition from the functions $x + y$, $x.y$, $1\dot{-}x$, then the equation $F = 0$ is provable if and only if $F = 0$ for the values 0,1 of its variables. i.e. the fragment of arithmetic containing just these equations is decidable and complete. (Decidable in this context means that one can decide if any equation $F = 0$ is provable in the Equation Calculus).

Kalmar [9] proposed a class of functions which he called elementary functions. This class of functions is in fact equivalent to the class formed from the initial equations $x+y$, $x.y$, $x\dot{-}y$ by composition, bounded summation and bounded multiplication (II)[1].

---

[1] That these classes of functions are equivalent is easily proved (see Kleene P 285). [10]

Given a general recursive function R, we know that there is no decision procedure by which we may decide for which y, (Ex)R(x,y) holds. In unpublished work, Ilona Bereczki showed that this holds where "general recursive" is replaced by "elementary".

I attempt in this thesis to extend Goodstein's results to larger classes of equations, in the knowledge that the class can never include all the equations F = 0 where F is elementary.

§3 The Integer Linear Programming Problem

In attempting to prove that the class of equations F = 0, where F is formed by composition from x + y and $x \dot{-} y$, is decidable, I showed that the problem was equivalent to solving the Integer Linear Programming Problem. I subsequently discovered that it was also equivalent to Presburger's Algorithm, for deciding that fragment of formal arithmetic containing addition but not multiplication. The question arose: "Is Presburger's Algorithm any use for solving the General Integer Linear Programming Problem?" the special interest being that Presburger's result was produced in 1929, and the Integer Linear Programming Problem was first solved by R. Gomory [3] in 1958. In fact Presburger's Algorithm does provide an algorithm for solving the programming problem, and I have derived an algorithm, from Presburger's result, specially for programming purposes, and this is explained in Appendix B.

§4 Different Formulations of The Axiom Free Equation Calculus

This system of primitive recursive arithmetic is formalised by R.L. Goodstein in several different ways in his book Recursive Number Theory. The one I shall use (referred to as R) has no axioms other than the recursive definition of functions e.g. a + 0 = a, a + Sb = S(a + b).

Its rules of inference are

Sb_1 $\qquad \dfrac{F(x) = G(x)}{F(A) = G(A)}$

Sb_2 $$\frac{A = B}{F(A) = F(B)}$$

T $$\frac{\begin{array}{l}A = B\\ A = C\end{array}}{B = C}$$

where F(x), G(x) are recursive functions, and A,B,C are recursive terms. Also we take as a rule of inference the primitive recursive uniqueness rule

U $$\frac{F(Sx) = H(x,F(x))}{F(x) = H^{x}F(0)}$$

where the iterative function $H^{x}t$ is defined by the primitive recursion $H^{0}t = t$, $H^{Sx}t = H(x,H^{x}t)$.

In an alternative formulation, Goodstein attempts to dispense with U, which he does at the expense of introducing one new rule of inference

E $$\frac{F(0) = 0,\ F(Sx) = F(x)}{F(x) = 0}$$

and 2 axioms

A $$a + (b \dot{-} a) = b + (a \dot{-} b)$$

and

P $$Sa \dot{-} Sb = a \dot{-} b$$

I have attempted to provide an alternative formulation in which U is not assumed. Also I attempted to eliminate any use of the substitution rule Sb_2 but there are 4 special cases of this rule that I could not avoid using.

These results are presented in Appendix A.

§5 Definition of terms and notation

Decidability When an equation, a set of equations or a formal system is said to be decidable we shall normally mean that it is decidable whether or not it is true in the Standard Model. Sometimes we shall mean that it is decidable whether or not it is provable in a particular formal system. Where the latter meaning is intended it will either be obvious from the

context or will be said to be decidable in the second sense.

Provability This will usually mean provable in the Equation Calculus.

Complete A formal system, a fragment of a formal system or simply a class of equations from a formal system, will be said to be complete when every equation in that class of equations which is true is also provable in that formal system

Equations or Identities? Although the formal system we are concerned with is called the Equation Calculus the statements in the system are identities rather than equations. We shall abide by the conventions and call them equations and use the equation sign =, rather than ≡. The latter sign ≡, is used particularly in Appendix B to denote equivalence modulo some natural number, and occasionally to denote logical equivalence.

Basic Functions In describing a set of functions e.g. "All the functions obtained by composition from $x + y$, $x.y$, $x \dot{-} y$," it is assumed always, that the class of functions includes the functions $Z(x) = 0$, $Sx = x + 1$, and $U_i^r(x_1,\ldots x_n) = x_r$. These 3 functions will be referred to as the basic functions.

Description of classes of functions If a class of functions is obtained from $x + y$, $x.y$, $x \dot{-} y$ by composition, the class of functions will be denoted by $C(+,.,\dot{-})$ and $x + y$, $x.y$, $x \dot{-} y$ are called the initial functions. The class $C(+,.,1\dot{-},\dot{-}1,\Sigma,\Pi)$ will refer to the functions obtained from the functions $x + y$, $x.y$, $1 \dot{-} x$, $x \dot{-} 1$ by composition and bounded addition and bounded multiplication and in this particular case we shall abbreviate $C(+,.,1\dot{-},\dot{-}1,\Sigma,\Pi)$ by C'.

With regard to proof in the Equation Calculus, because of the considerable length of such proofs, I have in most cases proved only one equation as an example and said that proofs of the others are similar.

Chapter 2

§1. Introduction

R.L. Goodstein in his paper "A Decidable Fragment of Recursive Arithmetic" [7] provided a decision procedure for the provability of equations $F = 0$ where $F \in C(+, \cdot, 1\dot{-})$, (for notation see p 4). He shows that $F = 0$ is provable if and only if $F = 0$ whenever the variables of F take the value 0 or 1. Hence if $F = 0$ is true (in the Standard Model), $F = 0$ is provable, so this fragment of Recursive Arithmetic is Complete as well as decidable,

We know (see p 2) that the class of equations $G = H$ for any elementary functions G, H, is undecidable. Since $G = H$ is equivalent to $(G \dot{-} H) + (H \dot{-} G) = 0$, for all elementary functions G,H, the class of equations $F = 0$ for all elementary functions F, is the same as the class $G = H$ G, H elementary.

So in seeking the largest possible class of equations F for which $F = 0$ is decidable we must drop at least one of the initial functions from which the elementary functions are obtained by composition. In this Chapter we omit the recursive difference function $x \dot{-} y$ (though we are able to add special cases of it) and provide a decision procedure for the equations $F = 0$ where $F \in C(+, \cdot, \Sigma, \pi, 1\dot{-}, \dot{-}1)$ - henceforth referred to as C'.

We first give an alternative proof of Goodstein's results to illustrate the essential simplicity of the method (§2), and then apply this method to the larger class of equations $F = 0$ where $F \in C'$ to give a decision procedure for truth (§3) and finally (§4) show that the decision procedure for truth is also one for provability in the Equation Calculus R, and hence that this fragment of the Equation Calculus is complete.

§2. Alternative Proof of Goodsteins Results:-

The idea behind this proof is quite simple. For a function $F \varepsilon C(+, \cdot, 1\dot{-})$ if we are interested only in whether $F = 0$ or $F \neq 0$ for particular values of the variables, and not the actual value of F if $F \neq 0$, it matters only whether the values of the variables are 0 or not.

This is illustrated by the following tables where ϕ represents "non-zero".

x	Sx
0	ϕ
ϕ	ϕ

x	Z(x)
0	0
ϕ	0

x_i	$U_i^n(x_1 \ldots x_n)$
0	0
ϕ	ϕ

+	0	y ϕ
0	0	ϕ
x		
ϕ	ϕ	ϕ

$\cdot$	0	y ϕ
0	0	0
x		
ϕ	0	ϕ

x	$1\dot{-}x$
0	ϕ
ϕ	0

So if $F(x_1, \ldots x_n) \varepsilon C(+, \cdot, 1\dot{-})$ to decide whether or not $F(x_1, \ldots x_n) = 0$ we consider the value of F for the 2^n different cases given by $X_i = 0$, $x_i = \phi$ for $i = 1 \ldots n$. If for all these cases $F = 0$, then $F = 0$ for all possible values of the variables.

Thus the deciding (for truth) of $F = 0$ has been reduced to a finite number (2^n) of cases, each of which is clearly decidable.

We now set down this argument formally to enable us eventually to translate it into a decision procedure for provability, and to facilitate generalisation of the result in §3 and §4.

Define

$$\alpha_1(x) = 1 \dot{-} (1 \dot{-} x) \tag{1}$$

Then α_1 maps 0 onto 0, and everything else onto 1

Theorem 2.2.1 If $F(x_1 \ldots x_n) \varepsilon C(+, \cdot, 1\dot{-})$ then

$$\alpha_1 F(x_1, \ldots x_n) = \alpha_1 F(\alpha_1 x_1, \ldots \alpha_1 x_n) \tag{2}$$

(i.e. this equation is true, not (as yet) provable)

Proof The following equations can be seen to be true, simply by considering the values with the variables zero and non-zero

$$\alpha_1(Sx) = \alpha_1 S(\alpha_1 x) \quad (3)$$

$$\alpha_1(Z(x)) = \alpha_1 Z(\alpha_1 x) \quad (4)$$

$$\alpha_1 U_i^n(x_1,\ldots x_n) = \alpha_1 U_i^n(\alpha_1 x_1,\ldots \alpha_1 x_n) \quad (5)$$

$$\alpha_1(x + y) = \alpha_1(\alpha_1 x + \alpha_1 y) \quad (6)$$

$$\alpha_1(x \cdot y) = \alpha_1(\alpha_1 x \cdot \alpha_1 y) \quad (7)$$

$$\alpha_1(1 \dot{-} x) = \alpha_1(1 \dot{-} \alpha_1 x) \quad (8)$$

So the theorem holds for the initial functions.

Suppose now that

$$f(x_1,\ldots x_n) = g(h_1(x_1\ldots x_n), \ldots\ldots h_r(x_1\ldots x_n)) \quad (9)$$

and the theorem holds for the functions g, h_i, $i = 1 \ldots r$.

i.e. $\alpha_1 g(y_1,\ldots y_r) = \alpha_1 g(\alpha_1 y_1,\ldots \alpha_1 y_r)$ (10)

$\alpha_1 h_i(x_1,\ldots x_n) = \alpha_1 h_i(\alpha_1 x_1,\ldots \alpha_1 x_n)$ for $i = 1 \ldots r$ (11)

then

$$\begin{aligned}
\alpha_1 f(x_1,\ldots x_n) &= \alpha_1 g(h_1(x_1\ldots x_n),\ldots h_r(x_1,\ldots x_n)) && \text{from (9)}\\
&= \alpha_1 g(\alpha_1 h_1(x_1,\ldots x_n),\ldots \alpha_1 h_r(x_1,\ldots x_n)) && \text{" (10)}\\
&= \alpha_1 g(\alpha_1 h_1(\alpha_1 x_1\ldots \alpha_1 x_n)\ldots \alpha_1 h_r(\alpha_1 x_1\ldots \alpha_1 x_n) && \text{" (11)}\\
&= \alpha_1 g(h_1(\alpha_1 x_1\ldots \alpha_1 x_n),\ldots h_r(\alpha_1 x_1\ldots \alpha_1 x_n)) && \text{" (10)}\\
&= \alpha_1 f(\alpha_1 x_1,\ldots \alpha_1 x_n) && \text{" (9)}
\end{aligned}$$

So the equation (2) is preserved under composition according to the schema (9), and is true for the initial functions, and hence holds for all $F \varepsilon C(+, \cdot, 1\dot{-})$.

Theorem 2.2.2 If $F(x_1\ldots x_n) \varepsilon C(+, \cdot, 1\dot{-})$ then $F(x_1\ldots x_n) = 0$ if and only if $F(\xi_1,\ldots \xi_n) = 0$ for all $\xi_i = 0,1$ $(i = 1\ldots n)$ and hence $F(x_1\ldots x_n) = 0$ is decidable.

Proof $F(x_1,\ldots x_n) = 0 \longleftrightarrow \alpha_1 F(x_1\ldots x_n) = 0$

$\longleftrightarrow \alpha_1 F(\alpha_1 x_1,\ldots \alpha_1 x_n) = 0$ from Theorem 2.2.1

$\longleftrightarrow F(\alpha_1 x_1,\ldots \alpha_1 x_n) = 0$

$\longleftrightarrow F(\xi_1,\ldots \xi_n) = 0$ for all $\xi_i = 0,1 (i=1\ldots n)$

$F(\xi_1,\ldots\xi_n)$ can now be evaluated for each of the 2^n cases given by $\xi_i = 0,1$ and hence $F(x_1,\ldots x_n) = 0$ is decidable

---

So we can decide the truth of $F = 0$ for $F \varepsilon C(+, \cdot, 1\dot{-})$; we turn now to deciding the provability of $F = 0$ in the Equation Calculus (i.e. system R).

Theorem 2.2.3 If $F(x_1,\ldots x_n)\ \varepsilon C(+, \cdot, 1\dot{-})$ then $F(x_1,\ldots x_n) = 0$ is provable in R if and only if $F(\xi_1,\ldots\xi_n) = 0$ is true for $\xi_i = 0,1$, and hence the fragment of R consisting of equations $F = 0$ for $F \varepsilon C(+, \cdot, 1\dot{-})$ is decidable.

Proof In this proof we make explicit use of rules of inference of R, Sb_1, Sb_2, T, U_1 (see [6]) and the introductory equations. In addition we shall use the schema

$$E_3.\quad \frac{F(0) = G(0)}{F(x) = G(x)}$$

which Goodstein ([6] p.108) shows can be derived in R.

First, equations (3) - (8) in Theorem 2.2.1 are all provable in R. We prove (6), the rest are similar (or easier).

Let $F(y) = \alpha_1(x + y) = 1 \dot{-} (1 \dot{-} (x + y))$

" $G(y) = \alpha_1(\alpha_1 x + \alpha_1 y) = 1 \dot{-} (1 \dot{-} ((1 \dot{-} (1 \dot{-} x)) + (1 \dot{-} (1 \dot{-} y)))$

Then $F(Sy) = 1 \dot{-} (1 \dot{-} (x + Sy))$

$= 1 \dot{-} (1 \dot{-} S(x + y))$ using $a + Sb = S(a + b)$ and Sb_2

$= 1 \dot{-} (0 \dot{-} (x + y))$ " $Sa \dot{-} Sb = a \dot{-} b$ [6] p.106

$= 1 \dot{-} 0$ and Sb_2.

$= 1.$

$G(Sy) = 1 \dot{-} (1 \dot{-} ((1 \dot{-} (1 \dot{-} x)) + (1 \dot{-} (1 \dot{-} Sy)))$

$= 1 \dot{-} (1 \dot{-} ((1 \dot{-} (1 \dot{-} x)) + 1))$

$= 1 \dot{-} (1 \dot{-} S(1 \dot{-} (1 \dot{-} x)))$

$= 1 \dot{-} 0$

$= 1$

So $F(Sy) = G(Sy)$.

Now $F(0) = 1 \dot{-} (1 \dot{-} x)$

$G(0) = 1 \dot{-} (1 \dot{-} (1 \dot{-} (1 \dot{-} x)))$

Let $H(x) = 1 \dot{-} (1 \dot{-} x)$

and $K(x) = 1 \dot{-} (1 \dot{-} (1 \dot{-} (1 \dot{-} x)))$

Then $H(0) = 0 = K(0)$ and $H(Sx) = 1 = K(Sx)$

Hence by E_3, $H(x) = K(x)$ i.e. $F(0) = G(0)$

So $F(0) = G(0)$.

And $F(Sy) = G(Sy)$.

Hence by E_3 $F(y) = G(y)$. (12)

i.e. $\alpha_1(x + y) = \alpha_1(\alpha_1 x + \alpha_1 y)$

Similarly, using results established in [6] we can derive the remainder of (3) - (8). Further if (10) (11) are provable in R, then we can see that (9) is also provable, for the proof given in Theorem 2.2.1 can be carried out in R, by use of the appropriate substitution schema.

So if $F(x_1,\ldots x_n) \in C(+, \cdot, 1\dot{-})$ then $\alpha_1 F(x_1,\ldots x_n) = \alpha_1 F(\alpha_1 x_1,\ldots \alpha_1 x_n)$ is provable in R.

Substituting $\xi_i = 0, Sx_i$ in (2), using Sb_2, we get 2^n equations

$$\alpha_1 F(\xi_1,\ldots \xi_n) = \alpha_1 F(\alpha_1 \xi_1,\ldots \alpha_1 \xi_n)$$

But $\alpha_1 \xi_i$ is provably equal to 0 or 1, and so if $F(x_1,\ldots x_n) = 0$ for $x_i = 0,1$, $i = 1\ldots n$, the 2^n equations

$$\alpha_1 F(\xi_1,\ldots \xi_n) = 0$$

are provable.

Hence the 2^n equations

$$F(\xi_1,\ldots \xi_n) = 0$$

are provable.

Hence

$$F(0,\xi_2,\ldots \xi_n) = 0 \quad \text{and} \quad F(Sx_1,\xi_2,\ldots \xi_n) = 0$$

Hence by U_1,

$$F(x_1,\xi_2,\ldots\xi_n) = 0$$

By repeating the procedure we get successively $F(x_1,x_2,\xi_3,\ldots\xi_n) = 0$,

$F(x_1,x_2,x_3,\xi_4,\ldots\xi_n) = 0 \ldots\ldots F(x_1,x_2,\ldots x_n) = 0$

So if $F(x_1,\ldots x_n) = 0$ is true for $x_i = 0,1$, $i = 1\ldots n$, then

$F(x_1\ldots x_n) = 0$ is provable in R. The converse is also clearly true.

Corollary The fragment of R consisting of all equations $F = 0$ for $F\varepsilon C(+, \cdot, 1\dot{-})$ is complete.

Proof If $F(x_1\ldots x_n) = 0$ is true for all values of $x_1\ldots x_n$, then it is certainly true for $x_i = 0,1$ for all i and hence by Theorem 2.2.3 is provable.

§3. Decidability of the class of equations F = 0 for $F\varepsilon C'$

It now requires only a natural extension of the methods of §2 to establish a decision procedure for this much larger class of equations.

We define the concept of the height of a function of C'. (λ_F denotes the height of F.)

The basic functions S(x), Z(x), $U_r^n(x_1\ldots x_n)$ are of height 0.

If F, G are of heights λ_F, λ_G respectively we define the height of $F + G$, and $F \cdot G$ to be $\max\{\lambda_F, \lambda_G\}$, and the heights of $1 \dot{-} F$, $F \dot{-} 1$, $\sum_{y<x} F(y)$, and $\prod_{y<x} F(y)$ to be λ_F, λ_F+1, λ_F+1, and λ_F+1 respectively.

Thus the height of any function $F\varepsilon C'$ is effectively defined and calculable.

Define $\alpha_i x$ by

$$\alpha_i x = i \dot{-} (i \dot{-} x)$$

Then α_i maps $0,1,\ldots i$ onto themselves and $x \geqslant i$ onto i.

Before proving our main result we prove an important preliminary result, analogous to Theorem 2.2.1.

Theorem 2.3.1 If $F \in C'$ then

$$\alpha_i F(x_1,\ldots x_n) = \alpha_i F(\alpha_q x_1,\ldots \alpha_q x_n)$$

for $i \geqslant 1$ and $q = (\lambda_F + i)^2$

(We require this result only for $i = 1$ but this stronger result is as easy to prove.)

Proof We first list some properties of α_i that are easily seen to be true by considering the 4 ranges $x, y < j$; $x \geqslant j$ and $y \neq j$; $x < j$ and $y \geqslant j$; $x, y \geqslant j$.

If $j \geqslant i \geqslant 1$, then

$$\alpha_i Sx = \alpha_i S(\alpha_j x) \tag{13}$$

$$\alpha_i Z(x) = \alpha_i Z(\alpha_j x) \tag{14}$$

$$\alpha_i U_r^n(x_1 \ldots . x_n) = \alpha_i U_r^n(\alpha_j x, \ldots {}_j x_n) \tag{15}$$

$$\alpha_i(\alpha_j x) = \alpha_i x = \alpha_j(\alpha_i x) \tag{16}$$

$$\alpha_i(x + y) = \alpha_i(\alpha_j x + \alpha_j y) \tag{17}$$

if $j \geqslant 1$ $$\alpha_i(1 \dot{-} x) = \alpha_i(1 \dot{-} \alpha_j x) \tag{18}$$

if $j \geqslant i + 1$ $$\alpha_i(x \dot{-} 1) = \alpha_i(\alpha_j x \dot{-} 1) \tag{19}$$

We also need the following results

$$\alpha_i \sum_{x<y_1} F(\alpha_j x) = \alpha_i \sum_{x<y_2} F(\alpha_j x) \quad \text{for } y_1, y_2 \geqslant i + j \tag{20}$$

$$\alpha_i \prod_{x<y_1} F(\alpha_j x) = \alpha_i \prod_{x<y_2} F(\alpha_j x) \quad " \quad " \quad " \tag{21}$$

To prove these we use the following results

$$\alpha_i(x + ry) = \alpha_i(x + sy) \quad \text{for } r, s \geqslant i \tag{22}$$

$$\alpha_i(x \cdot y^r) = \alpha_i(x \cdot y^s) \quad " \quad " \tag{23}$$

(22) obviously holds for $y = 0$, and if $y \geqslant 1$ both sides become i

(23) is obvious for $x = 0$ and for $y = 0$ or 1; if $x \geqslant 1$ and $y \geqslant 2$, then $x.y^r$ and $x.y^s$ exceed i, and hence $\alpha_i(x.y^r) = \alpha_i(x.y^s) = i$

Now to prove (20) we observe that

$$\alpha_i \sum_{x<y_r} F(\alpha_j x) = \alpha_i\Big(\sum_{x<j} F(x) + (y_r - j)\, F(j)\Big) \quad \text{for } r = 1, 2 \tag{24}$$

but now $y_r - j \geq i$ and hence by (22) the expressions given by $r = 1,2$ are equal. The proof of (21) is similar, so I shall omit the proof.

LEMMA If there exist numbers $p_r \geq i$, $r = 1....n$ such that

$$\alpha_i F(x_1....x_n) = \alpha_i F(\alpha_{p_1} x_1,...\alpha_{p_n} x_n) \tag{25}$$

then for any $q \geq p_r$ for all r

$$\alpha_i F(x_1....x_n) = \alpha_i F(\alpha_q x_1,...\alpha_q x_n) \tag{26}$$

Proof Substitute $\alpha_q x_i$ for x_i in (25).

i.e. $\alpha_i F(\alpha_q x_1,...\alpha_q x_n) = \alpha_i F(\alpha_{p_1} \alpha_q x_1,...\alpha_{p_n} \alpha_q x_n)$

$= \alpha_i F(\alpha_{p_1} x_1,...\alpha_{p_n} x_n)$ from (16)

$= \alpha_i F(x_1....x_n)$ " (25)

We now prove Theorem 2.3.1.

The basic functions are of height 0. So if $q = (\lambda + i)^2$ where $\lambda = 0$, $q = i^2$. So for $i \geq 1$, $q \geq i \geq 1$. So from (13), (14) and (15), if F is a basic function, Theorem 2.3.1 holds.

Suppose that Theorem 2.3.1 holds for any 2 functions f, g then it also holds for $F = f + g$, $f \cdot g$, $f(x) \dot{-} 1$, $1 \dot{-} f(x)$, $\sum_{x<y} f(x)$ and $\prod_{x<y} f(x)$.

The proofs are tediously long but straightforward, so I shall prove the result for f + g to illustrate the method.

Let $F = f + g$. By assumption

$$\alpha_i f(x_1....x_n) = \alpha_i f(\alpha_{q_{f,i}} x_1,...\alpha_{q_{f,i}} x_n) \tag{27}$$

$$\alpha_i g(x_1...x_n) = \alpha_i g(\alpha_{q_{g,i}} x_1,...\alpha_{q_{g,i}} x_n) \tag{28}$$

where $q_{f,i} = (\lambda_f + i)^2$

and $q_{g,i} = (\lambda_g + i)^2$

Now $\alpha_i(f(x_1....x_n) + g(x_1....x_n)) = \alpha_i(\alpha_i f(x_1....x_n) + \alpha_i g(x_i....x_n))$

from (17) (29)

Now $q_{f,i} = (\lambda_f + i)^2 \quad q_{g,i} = (\lambda_g + i)^2$

and $q_{f+g,i} = (\max(\lambda_f, \lambda_g) + i)^2$

So $q_{f+g,i} \geq q_{f,i}$ and $q_{f+g,i} \geq q_{g,i}$ (30)

Using the Lemma, from (27), (28) and (30)

$$\alpha_i f(x_1,\ldots x_n) = \alpha_i f(\alpha_{q_{f+g,i}} x_1,\ldots \alpha_{q_{f+g,i}} x_n)$$

and $$\alpha_i g(x_1,\ldots x_n) = \alpha_i f(\alpha_{q_{f+g,i}} x_1,\ldots \alpha_{q_{f+g,i}} x_n).$$

So from (29)

$$\alpha_i(f(x_1\ldots x_n) + g(x_1\ldots x_n)) = \alpha_i(\alpha_i f(\alpha_{q_{f+g,i}} x_1,\ldots \alpha_{q_{f+g,i}} x_n) + \alpha_i g(\alpha_{q_{f+g,i}} x_1\ldots \alpha_{q_{f+g,i}} x_n))$$

$$= \alpha_i(f(\alpha_{q_{f+g,i}} x_1\ldots) + g(\alpha_{q_{f+g,i}} x_1\ldots))$$

i.e. if $F = f + g$, since $q_{F,i} = q_{f+g,i} = q_{F,i}$

$$\alpha_i F(x_1\ldots x_n) = \alpha_i F(\alpha_{q_{F,i}} x_1,\ldots \alpha_{q_{F,i}} x_n).$$

So if Theorem 2.3.1 holds for f,g, then it holds for $F = f + g$.

Similarly if the Theorem holds for f,g it holds for f.g, $1 \dot{-} f$, $f \dot{-} 1$, $\sum_{y<x} f(y)$ and $\prod_{y<x} (y)$.

So by induction on the structure of a function $F \varepsilon C'$, Theorem 2.3.1 holds for all $F \varepsilon C'$.

Theorem 2.3.2 If $F \varepsilon C'$ the class of formulae $F = 0$ is decidable.

Proof By Theorem 2.3.1, if $q = (\lambda_F + i)^2$

$$\alpha_1 F(x_1,\ldots x_n) = \alpha_1 F(\alpha_q x_1,\ldots \alpha_q x_n) \quad (31)$$

But $F(x_1\ldots x_n) = 0 \quad \alpha_1 F(x \ldots x_n) = 0$ (32)

$\alpha_1 F(\alpha_q x_1,\ldots \alpha_q x_n) = 0$ from (31)

$F(\alpha_q x_1,\ldots \alpha_q x_n) = 0.$

But since for $x \geq q$, $\alpha_q x = q$,

$F(x_1 \ldots x_n) = 0$ $F(\xi_1, \ldots \xi_n) = 0$ for all possible $\xi_i \ldots 0, 1 \ldots q$ and for $i = 1 \ldots n$.

This leaves us to decide a finite number of cases. So $F = 0$ is decidable for $F \varepsilon C'$.

So we have reduced the decision problem to a finite problem. The number $q = (\lambda_F + i)^2$ is clearly not necessarily the smallest possible number of values of the variables that we need try. For a particular function we can find the smallest such number (and this may be different for each variable) by applying α_i to the function and using (13) - (24) in each case using the smallest possible j.

Corollary The function $x \dot{-} y$ cannot be obtained by substitution from the functions of C'.

Proof If it could, C' would contain all the elementary functions and hence be undecidable.

§4. Completeness of Certain Classes of Equations in the Equation Calculus

Theorem 2.4.1 If $F \varepsilon C'$ then the equation

$$F(x_1, \ldots x_n) = 0$$

is provable in R, if and only if, each instance of the equation holds when $x_1 \ldots x_n$ take values in the set $0, 1, \ldots (\lambda_F + 1)^2$.

Proof All the equations (13)-(26) are provable in R.

The proofs are simple but extremely long and I shall omit them. Then one may prove in R, that if $F \varepsilon C'$,

$$\alpha_i F(x_1 \ldots x_n) = \alpha_i F(\alpha_q x_1, \ldots \alpha_q x_n) \qquad (33)$$

where $q = (\lambda_F + i)^2$ by an inductive procedure.

Now if $\xi_i = 0, 1, \ldots q - 1, x_i + q$, using E_3 substituting $x_i = \xi_i$ from (33) with $i = 1$, we get $(q + 1)^n$ different equations

$$\alpha_1 F(\xi_1, \ldots \xi_n) = \alpha_1 F(\alpha_q \xi_1, \ldots \alpha_q \xi_n)$$

But $\alpha_q \xi_i$ is provably equal to some one of $0,1,\ldots q$

So $\alpha_1 F(\alpha_q \xi_1, \ldots \alpha_q \xi_n)$ is provably equal to $\alpha_1 F(\theta_1, \ldots \theta_n)$

for some value $0,1 \ldots q$ of θ_i, $i = 1 \ldots n$.

Suppose now that $F(\theta_1 \ldots \theta_n) = 0$ for $\theta_i = 0,1,\ldots q$, then the

$(q+1)^n$ equations $\alpha_1 F(\xi_1 \ldots \xi_n) = 0$ are provable

i.e. " " $F(\xi_1 \ldots \xi_n) = 0$ " "

So the $(q+1)^{n-1}$ equations

$$F(0,\xi_2,\ldots\xi_n) = 0, \quad F(1,\xi_2,\ldots\xi_n) = 0 \ldots\ldots\ldots F(x_1 + q, \xi_2 \ldots \xi_n) = 0$$

are provable.

Let $G(x_1,\xi_2,\ldots\xi_n) = F(x + (q-1),\xi_2,\ldots\xi_n)$

Then $G(Sx_1,\xi_2,..\xi_n) = F(x_1 + q, \xi_2 \ldots \xi_n) = 0$, and $G(0,\xi_2,\ldots\xi_n) =$

$F(q_1 - 1,\xi_2,\ldots\xi_n) = 0$

Hence by E_3, $G(x,\xi_2,\ldots\xi_n) = 0$

i.e. $F(x_1 + q - 1, \xi_2 \ldots \xi_n) = 0$

Repeating this procedure q times we obtain

$$F(x_1,\xi_2,\ldots\xi_n) = 0$$

Similarly we now prove

$$F(x_1,x_2,\xi_3 \ldots \xi_n) = 0$$

and so on to $F(x_1,x_2 \ldots x_n) = 0$

So if $F(\theta_1,\theta_2,\ldots\theta_n) = 0$ for $\theta_i = 0,1 \ldots q$, $i = 1 \ldots n$ then

$F(x_1 \ldots x_n) = 0$ is provable.

The converse is obtained by substituting particular numbers for x using schema Sb_1.

Corollary The class of equations $F = 0$ where $F \varepsilon C'$ is complete.

For if an equation $F = 0$ holds for all values of its variables $x_1 \ldots x_n$, then it holds for $x_i = 0,1 \ldots q$ where $q = (\lambda_F + 1)^2$, and hence is provable.

§5. Further Functions in C'

We know that the class of equations $F = 0$ for F elementary is un-

decidable, the elementary functions being those obtainable by applying the operations $\sum_{y<z}$, $\prod_{y<x}$, and composition, to the functions S, Z, U_i^n, $x + y$, $x \cdot y$ and $x \dot{-} y$. By omitting $x \dot{-} y$ from our initial functions but retaining the 2 special cases[1] $1 \dot{-} y$, $x \dot{-} 1$, we obtained a decision procedure (Theorem 2.3.2) for the remaining class of equations $F = 0$. It is of course possible that we could have added a large number of special cases of $x \dot{-} y$ to our class of functions without altering our decision procedure . I mention 2 such functions, $x + (y \dot{-} x)$ and $x \dot{-} (x \dot{-} y)$, which represent the maximum and minimum of x,y respectively. That the decision procedure is unchanged is seen from the equations

$$\alpha_i(x + (y \dot{-} x)) = \alpha_i(\alpha_j x + (\alpha_j y \dot{-} \alpha_j x)) \qquad \text{for all } j \geq i$$

and

$$\alpha_i(x \dot{-} (x \dot{-} y)) = \alpha_i(\alpha_j x \dot{-} (\alpha_j x \dot{-} \alpha_j y)) \qquad \text{for all } j \geq i$$

Apart from adding special cases of $x \dot{-} y$ to the initial functions it is worth noting that certain other special cases are already in C'

e.g. $f \dot{-} N = (\ldots((f \dot{-} 1) \dot{-} 1)\ldots.\dot{-}1)$ i.e. performing N successive subtractions (34)

$$SN \dot{-} g = (N \dot{-} g) + (1 \dot{-} (g \dot{-} N)) \qquad (35)$$

$$(N \dot{-} x) \dot{-} y = N \dot{-} (x + y)$$

$$x \dot{-} (N \dot{-} y) = x \cdot (1 \dot{-} (N \dot{-} y)) + ((x + y) \dot{-} N)(1 \dot{-} (1 \dot{-} (N \dot{-} y)))$$

These last 2 cases suggest that as long as one side of $x \dot{-} y$ is bounded, the class of equations $F = 0$ will be decidable.

Furthermore the equations (34) and (35) give rise to the following theorem.

Theorem 2.5.1

If $F \in C'$ the class of equations $F = N$ where N is a constant is

(i) decidable for truth

(ii) is provable in R, if and only if each instance of the equation hold when $x_1 \ldots . x_n$ take values in the set $0,1,\ldots(\lambda_{F+N} + 1)^2$

(iii) is complete

Proof Replace F by $(F \dot{-} N) + (N \dot{-} F)$ in Theorems 2.3.2, 2.4.1 and the corollary to Theorem 2.4.1. The height of $(F \dot{-} N) + (N \dot{-} F)$ = max {height of $F \dot{-} N$, height of $N \dot{-} F$} = max $\{\lambda_{F+N}, \lambda_F\}$ = λ_{F+N}

Subsequent to this work, G. Rousseau proved similar results by a different and interesting method [15]

Chapter 3

§1 Introduction

When a class C of functions is formed by composition from a set of initial functions which includes $x + y$ and $x \dot{-} y$, any identity $G = H$ for $G,H \in C$ can be put in the form $F = 0$ by virtue of the equivalence

$$G = H \leftrightarrow (G \dot{-} H) + (H \dot{-} G) = 0$$

So if the initial functions include $x + y$ and $x \dot{-} y$, we need consider only equations of the form $F = 0$. If $x \dot{-} y$ is not one of the initial functions (as in C') the class of equations $G = H$ is a larger class than the class $F = 0$. So we consider equations $F = G$ and see if the results of chapter 2 extend to this larger class.

The counterpart to Theorem 2.3.2, namely "If $F,G \in C'$, the class of equations $F = G$ is decidable" is, as far as I know, an open question. However, and this is the main result in this chapter, "If $F,G \in C(+, ., 1\dot{-}, \dot{-}1)$ then the class of equations $F = G$ is decidable" i.e. we have had to omit bounded summation $\Sigma_{y<z}$ and bounded multiplication $\Pi_{y<z}$ from the initial functions considered in Theorem 2.3.2.

§2 Decidability for truth

Theorem 3.2.1 The class of equations $F = G$ where $F,G \in C(+,., 1\dot{-}, \dot{-}1)$ is decidable.

Proof The proof is in 2 parts

(i) To prove that $F = G$ is equivalent to a finite conjunction of sentences of the form

$$h = 0 \rightarrow P = Q \qquad (1)$$

where $h \in C(+, ., 1\dot{-}, \dot{-}1)$ and P,Q are polynomials with positive coefficients.

(ii) To show that sentences in the form (1) are decidable.

Proof of (1)

Let the predicate H stand for the equation F = G. We now successively eliminate[1] occurrences of $1 \dot{-} f$, $f \dot{-} 1$ from H, to turn H into a predicate representing the identity of 2 polynomials.

This is done by an inductive procedure starting with occurrences of $1 \dot{-} f$ and $f \dot{-} 1$ where the f itself does not contain any occurrence of functions involving $\dot{-}$. First note that F = G is equivalent to $0 = 0 \to F = G$ or rather

$$0 = 0 \to H$$

Now consider the sentence

$$f_1 = 0 \wedge f_2 = 0 \wedge \cdots \wedge f_k = 0 \wedge 1 \dot{-} f_{k+1} = 0 \wedge \cdots \wedge 1 \dot{-} f_r = 0 \to H' \qquad (2)$$

Suppose H' contains a function $1 \dot{-} f$ where f itself is free of the full recursive difference $\dot{-}$

Then (2) is equivalent to

$$\begin{aligned} &(f_1 = 0 \wedge f_2 = 0 \wedge \cdots \wedge f_k = 0 \wedge 1 \dot{-} f_{k+1} = 0 \wedge \cdots \wedge 1 \dot{-} f_r = 0 \wedge f = 0 \to H'(1)) \\ \wedge\ &(f_1 = 0 \wedge f_2 = 0 \wedge \cdots \wedge f_k = 0 \wedge 1 \dot{-} f_{k+1} = 0 \wedge \cdots \wedge 1 \dot{-} f_r = 0 \wedge 1 \dot{-} f = 0 \to H'(0)) \end{aligned} \qquad (3)$$

and we have eliminated the occurrence of $1 \dot{-} f$ in H'. Similarly if H' contains a function $f \dot{-} 1$ where f itself is free of $\dot{-}$, then (2) is equivalent to

$$\begin{aligned} &(f_1 = 0 \wedge f_2 = 0 \wedge \cdots \wedge f_k = 0 \wedge 1 \dot{-} f_{k+1} = 0 \wedge \cdots \wedge 1 \dot{-} f_r = 0 \wedge f = 0 \to H'(0)) \\ \wedge\ &(f_1 = 0 \wedge f_2 = 0 \wedge \cdots \wedge f_k = 0 \wedge 1 \dot{-} f_{k+1} = 0 \wedge \cdots \wedge 1 \dot{-} f_r = 0 \wedge 1 \dot{-} f = 0 \to H'(f-1)) \end{aligned} \qquad (4)$$

Since if $1 \dot{-} f = 0$, $f \dot{-} 1 = f - 1$ using the equivalence $a_1 = 0 \wedge \cdots \wedge a_p = 0 \leftrightarrow a_1 + \cdots + a_p = 0$, we see that eliminating an occurrence of $1 \dot{-} f$ or $f \dot{-} 1$ from H' leads to

$$\begin{aligned} &h_1 = 0 \to H'(1) \text{ and } h_2 = 0 \to H'(0) \text{ from (3)} \\ &\text{or} \\ &h_1 = 0 \to H'(0) \text{ and } h_2 = 0 \to H'(f-1) \text{ from (4)} \end{aligned} \qquad (5)$$

---

[1] In connection with occurrences of $f \dot{-} 1$, we change $f \dot{-} 1$ to $f - 1$ rather than eliminating it.

respectively, where $h_1, h_2 \in C(+,.,1\dot{-},\dot{-}1)$.

By this induction the equation F = G, represented by the predicate H, is equivalent to a finite conjunction of sentences of the form

$$h = 0 \rightarrow H^*$$

or

$h = 0 \rightarrow P^* = Q^*$ where P^*,Q^* are polynomials and $h \in C(+,.,1\dot{-},\dot{-}1)$[1]

So bringing the terms of P^*,Q^* to give P,Q(polynomials with positive coefficients), F = G, for $F,G \in C(+,.,1\dot{-},\dot{-}1)$ is equivalent to a finite conjunction of sentences of the form

$$h = 0 \rightarrow P = Q \tag{6}$$

where $h \in C(+,.,1\dot{-},\dot{-}1)$ and P,Q are polynomials with positive coefficients.

Proof of (ii)

$C(+,.,1\dot{-},\dot{-}1) \subseteq C(+,.,1\dot{-},\dot{-}1,\Sigma,\Pi)$ so by Theorem 2.3.1 if $h \in C(+,.,1\dot{-},\dot{-}1)$

$\alpha_1 h(x_1 \ldots x_n) = \alpha_1 h(\alpha_q x_1 \ldots \alpha_q x_n)$, where $q = (\lambda_h + 1)^2$, λ_h being the height of h as defined in chapter 2.

Furthermore, since $h(x_1 \ldots x_n) = 0 \leftrightarrow \alpha_1 h(x_1 \ldots x_n) = 0$

$$\leftrightarrow \alpha_1 h(\alpha_q x_1 \ldots \alpha_q x_n) = 0$$

$$\leftrightarrow h(\alpha_q x_1 \ldots \alpha_q x_n) = 0$$

Hence (6) is equivalent to

$$h(\alpha_q x_1 \ldots \alpha_q x_n) = 0 \rightarrow P(x_1 \ldots x_n) = Q(x_1 \ldots x_n) \tag{7}$$

Since $\alpha_q x_i \leq q$ for all i, we have reduced the problem of deciding (6) in so far as we need consider the value of h for only a finite number (q) of values of each of its variables.

---

[1] P^*,Q^* may have negative coefficients due to the f - 1 that arises in eliminating $f \dot{-} 1$. However, these negative coefficients can be removed by transferring terms with negative coefficients to the other side. For the same reason h may contain an expression f - 1, but this may be changed to $f \dot{-} 1$ since h = 0 will already contain the condition that $1 \dot{-} f = 0$ and $f \dot{-} 1 = f - 1$.

If $x_i \leqslant q$, $\alpha_q x_i = x_i$ and if $x_i \geqslant q$ then $\alpha_q x_i = q$.

Let $(r_1 \ \ldots \ r_n)$, $r_i \leqslant q$ be such that $h(r_1 \ldots r_n) = 0$.

If $r_i < q$, substitute $x_i = r_i$ in $P = Q$

and if $r_i = q$, substitute $x_i + q$ for x_i in $P = Q$, and call the new polynomials P_1 and Q_1. The truth or falsity of $F \equiv G$ is then equivalent to the truth or falsity of $P_1 \equiv Q_1$ for all the different polynomials $P_1 Q_1$ that arise due to the different sets of values of $r_1 \ldots r_n$ that satisfy $h(r_1 \ldots r_n) = 0$ and the different sentences $h = 0 \rightarrow P = Q$ in the finite conjunction of such sentences as indicated in (4).

So the problem is reduced to deciding whether 2 polynomials P_1, Q_1 are identically equal. But 2 polynomials are equal for all values of their variables if and only if they are the same polynomial, and hence $P_1 \equiv Q_1$ is decidable.

Chapter 4

§1. Introduction:-

As we pointed out in Chapter I the class of equations between elementary functions is undecidable (see p 2). Our aim in Chapters 2 and 3 was to add to the class of equations we could decide, starting from the class of equations between functions formed by composition from the initial functions $x + y$, $x \cdot y$, and $1 \dot{-} x$ (i.e. Goodstein's result). In Chapter 2 we concentrated on equations $F = 0$ and obtained a decision procedure for the class of such equations where F was any function which could be formed from the initial functions $x + y$, $x \cdot y$, $1 \dot{-} x$, $x \dot{-} 1$, by use of bounded summation ($\sum_{x<y} f(x)$), bounded multiplication ($\prod_{x<y} f(x)$) and composition, i.e. all the elementary functions which do not contain the full recursive difference $\dot{-}$, although certain special cases of it can be included.

In Chapter 3 we did the same for equations $F = G$ but here again the use of the full recursive difference could not be included, and furthermore our equations did not contain the bounded summation and bounded multiplication (Σ,Π). This result is closely related to that in Chapter 2 since $F = G$ is equivalent to $F \dot{-} G = 0$ and $G \dot{-} F = 0$. So we had effectively sacrificed the bounded summation and multiplication for the sake of one use of the full recursive difference.

Both in Chapters 2 and 3 the recursive difference is the obstacle to further extensions. There clearly has to be some such function since we know that the class of equations $F = G$ for F,G elementary is undecidable. An obvious alternative line to follow is to start with the recursive difference and to see what initial functions we could add to the recursive difference and retain decidability. In this Chapter we show that addition can be added, i.e. the class of equations $F = G$ where $F,G \in C(+,\dot{-})$ is decidable, but that including addition and multiplication leads to a

decision problem equivalent to Hilbert's 10th Problem.

Having obtained a decision procedure for the class of equations $F = G$ where $F,G \in C(+,\dot{-})$ by a somewhat complex procedure (§2), eventually involving integer linear programming, see Appendix B, it was pointed out to me (and here I must thank Craig McKay, University of East Anglia) that the result I had obtained was a direct consequence of Presburger's algorithm for deciding that fragment of formal arithmetic containing addition but not multiplication [14], and this is explained in §3.

The question naturally arose whether Presburger's algorithm (devised in 1929) is any use in solving the integer linear programming problem (first solved in 1958 by R.E. Gomory [3] [4] and also by A.H. Land and A.G. Doig [12]). The answer is yes. The proof of this and an algorithm for solving the programming problem are in Appendix B.

§2. A Decision Procedure for the Class of Equations $F = G$ where $F,G \in C(+,\dot{-})$

Theorem 4.2.1. The decision problem for equations $F = G$, where $F,G \in C(+,\dot{-})$ is equivalent to the decision problem for the consistency of sets of linear inequalities for non-negative integer values of the variables.

Proof If $G,H \in C(+,\dot{-})$, $G = H$ if and only if $G \dot{-} H = 0$ and $H \dot{-} G = 0$. Thus our problem is equivalent to deciding equations of the form $F = 0$, $F \in C(+,\dot{-})$. For any such equation $F = 0$, we eliminate $\dot{-}$ from F, as in [16], by successive replacement of $F(g \dot{-} h) = 0$ by

$$(g \geq h \wedge F(g - h) = 0) \vee (g < h \wedge F(0) = 0)$$

Thus we obtain finally a formula of the form

$$\left.\begin{array}{l} ((g_1 \geq h_1) \wedge \ldots\ldots \wedge (g_r \geq h_r) \wedge F(g_1 - h_1,\ g_2 - h_2, \ldots_1,\ g_r - h_r) = 0) \\ \vee((g_1 \geq h_1) \wedge \ldots\ldots \wedge (g_r < h_r) \wedge F(g_1 - h_1 \ldots g_{r-1} - h_{r-1},\ 0) = 0) \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ \vee((g_1 < h_1) \wedge \ldots\ldots \wedge (g_r < h_r) \wedge F(0,0,\ldots.0) = 0) \end{array}\right\} (1)$$

Writing this as

$$(I_1 \wedge F_1 = 0) \vee (I_2 \wedge F_2 = 0) \vee \ldots\ldots \vee (I_{2^r} \wedge F_{2^r} = 0) \qquad (2)$$

then $F = 0$ if and only if, for all $s = 1,2\ldots.2^r$, $F_s = 0$ for all the non-negative integer values of the variables that satisfy the preceding set of inequalities I_s.[1]

Consider a particular disjunct $I_t \wedge F_t = 0$. Replace $g \geqslant h$ by $g - h \geqslant 0$ and $g < h$ by $h - g - 1 \geqslant 0$, since we are interested only in integer values of the variables. We can now put each inequality in the form $L \geqslant b$ where L is a homogeneous linear form and b an integer. $F_t = 0$ can be put in the form $L_t = C_t$ where L_t is a homegeneous linear form and C_t a non-negative integer. So the disjunction becomes

$$(L_{1t} \geqslant b_{1t}) \wedge \ldots\ldots\ldots \wedge (L_{rt} \geqslant b_{rt}) \wedge (L_t = C_t) \qquad (3)$$

If for any $t = 1 \ldots 2^r$

$$\left.\begin{array}{l} (L_{1t} \geqslant b_{1t}) \wedge \ldots\ldots\ldots \wedge (L_{rt} \geqslant b_{rt}) \wedge (L_t > C_t) \\ \text{or} \\ (L_{1t} \geqslant b_{1t}) \wedge \ldots\ldots\ldots \wedge (L_{rt} \geqslant b_{rt}) \wedge (L_t < C_t) \end{array}\right\} \qquad (4)$$

are consistent, for non-negative integer values, then $F \neq 0$; if all these sets of inequalities are inconsistent ($t = 1 \ldots 2^r$) then $F = 0$.

Thus if we can decide the consistency of sets of linear inequalities, for non-negative integer values of the variables, we can decide the equation $F = 0$.

Conversely, given any set of linear inequalities (strict or otherwise) we may put them in the form

$$g_1 \geqslant h_1 \ , \ g_2 \geqslant h_2 \ , \ldots\ldots\ldots g_r \geqslant h_r \qquad (5)$$

as above. Each term may be transferred to its positive side and so we

---

1. Any set of values for the variables in F satisfies one and only of the sets of conditions $I_1, I_2, \ldots . I_{2^r}$.

may assume that g_i , h_i are sums of positive terms. Now consider the function F where

$$F = 1 \dot{-} [(h_1 \dot{-} g_1) + (h_2 \dot{-} g_2) + ... + (h_r \dot{-} g_r)]$$

This is a function in $C(+,\dot{-})$ such that if $F = 0$

then $g_i \geq h_i$, $i = 1 .... r$, are not consistent and if $F \neq 0$

then $g_i \geq h_i$, $i = 1 .... r$ are consistent.

So if we can decide $F = 0$ for any $F \varepsilon C(+,\dot{-})$ then we can decide the consistency, for integer values of the variables, of any set of linear inequalities.

Corollary. The class of equations $G = H$ where $G,H \varepsilon C(+,\dot{-})$ is decidable.

Proof. In (3) convert the inequalities to equations by the introduction of slack variables[1]

$$\text{i.e. } L_{1t} - x_{1t} = b_{1t} \text{ , } L_{2t} - x_{2t} = b_{2t} \text{ , ... } L_{mt} - x_{mt} = b_{mt} \qquad (6)$$

where $x_{it} \geq 0$. We now optimise the linear form L_t subject to the above constraints. In [3] and [4], R.E. Gomory gives a finite algorithm for solving such integer linear programming problems. In particular Gomory's method indicates those cases in which the constraints are inconsistent and hence there is no solution[2]. Once we have optimised L_t subject to the constraints in (6) for every $t = 1,2, ...,2^r$, we then know whether the set of inequalities in (3) are consistent or not and hence whether $F = 0$ or not.

---

1. The slack variable for $L_{1t} \geq b_{1t}$ is not only different from that for $L_{2t} \geq b_{1t}$ but is also different for different values for t.
2. Gomory's method uses the simplex and dual simplex algorithms see [3] and [4] and such theoretical problems that can occur in special cases (e.g. degeneracy) have all been resolved (see [4]).

§3. A Decision Procedure using Presburgers Algorithm

We established in the proof of Theorem 4.2.1. that deciding whether or not an equation F = G, $F,G\varepsilon C(+,\dot{-})$, holds is equivalent to deciding whether a statement in the form of (1) holds. We now convert all the inequalities in (1) into equations using slack variables, z_{ij}, i = 1 ... r, j = 1,2, ... 2^r.

Suppose that the variables contained in (1) are $x_1,\ldots x_n$ consider the statement

$$(x_1)(x_2)\ldots.(x_n)(Ez_1)(Ez_2)\ldots.(Ez_{r,2^r})R(x_1\ldots x_n,\ z\ \ldots.z_{r,2^r})$$

where R is such that this expresses the statement that for all values of the variables $x_1,\ldots x_n$, the statement (1) in §2 is true. This is a statement in the formal system D in Hilbert and Bernays Vol. 1, and we simply apply Presburger's algorithm to it to decide its truth or falsity i.e. whether the original equation F = 0 is true or not.

§4. The Class of Equations F = G where $F,G\varepsilon C(+,\dot{-},\cdot)$

We now consider what happens when we add multiplication to the initial functions i.e. is the class of equations F = G, where $F,G\varepsilon C(+,\dot{-},\cdot)$ decidable.

Theorem 4.4.1. The decision problem for the class of equations F = G where $F,G\varepsilon C(+\ \dot{-}\ \cdot)$ is equivalent to deciding the consistency of sets of polynomial inequalities for non-negative integer values of the variables.

Proof Similar to that in Theorem 4.2.1.

Corollary

The existence of a decision procedure for the above-mentioned class of equations would imply that Hilbert's 10th problem were soluble.

Proof Suppose F is any polynomial. Consider the inequalities $F \geqslant 0$, $F \leqslant 0$. From Theorem 4.4.1 if there is a decision procedure for the given class of equations, we can decide the consistency, for integer values of

the variables of any set of inequalities

i.e. we can decide if $F \geq 0$ and $F \leq 0$ are consistent

i.e. " " " " $F = 0$ for any non-negative integer values of the variables.

i.e. Hilbert's 10th problem is soluble.

In fact of course there is no existing solution to Hilbert's 10th problem and the results of Davis, Putnam and Robinson [2] suggest that the problem is insoluble, and hence that class of equations $F = G$ where $F,G \in C(+ \dot{-} \cdot)$ is undecidable.

Chapter 5

§1 Introduction

In previous chapters the results are concerned with classes of equations in Recursive Arithmetic and in some cases, apart from considering whether the class of equations is decidable for truth, we have investigated to find if these equations which are true are provable, in the particular formalisation of Recursive Arithmetic, the Equation Calculus, and hence shown that certain fragments of the Equation Calculus are complete. In these chapters we have referred to equations F = G, when in fact our results are about identities, i.e.

$$(x_1) \ldots (x_n)(F(x_1 \ldots x_n) = G(x_1 \ldots x_n))$$

In this chapter our concern is number theoretic equations for their own sake and without reference to provability in the Equation Calculus or any other formal system. In effect we shall consider the decidability of classes of equations

$$(Q_1x_1) \ldots (Q_nx_n)(F(x_1 \ldots x_n) = G(x_1 \ldots x_n))$$

where Q_ix_i is (Ex_i) or (x_i), for F,G in several different classes of functions, and in particular Q_ix_i being Ex_i for all i.[1] These results will be considered in relation to well-known number theory problems and results like Hilbert's 10th Problem and the undecidability of exponential diophantine equations (the result due to Davis, Putnam and Robinson) [2]

§2 $(Q_1x_1) \ldots (Q_nx_n)(F(x_1 \ldots x_n) = G(x_1 \ldots x_n))$

Which, if any, of our "identity" results can be extended to equations with any quantifier prefix.[1]

---

[1] One is tempted to think that if the class $(x_1) \ldots (x_n)(F = G)$ is decidable then since from $(x)A(x)$ we deduce $(Ex)A(x)$, $(Q_1x_1) \ldots (Q_nx_n)$ is decidable. However if a particular case $(x_1) \ldots (x_n)(F. = G.)$ is decidable and is false, one knows no more about the rest of $(Q_1x_1) \ldots (Q_nx_n)F(x_1, \ldots, x_n)$.

Theorem 5.2.1 The class of equations

$$(Q_1x_1)\ldots(Q_nx_n)(F(x_1\ldots x_n) = 0) \tag{1}$$

where $F\varepsilon C'$ and Q_ix_i is (Ex_i) or (x_i), is decidable

Proof

$$(Q_1x_1)\ldots(Q_nx_n)\left[(F(x_1\ldots x_n) = 0)\leftrightarrow(\alpha_iF(x_1\ldots x_n) = 0)\right] \tag{2}$$

and since $F\varepsilon C'$, if $q = (\lambda_F + i)^2$

$$\alpha_iF(x_1\ldots x_n) = \alpha_iF(\alpha_qx_1,\ldots\alpha_qx_n) \tag{3}$$

Hence from (2)

$$(Q_1x_1).....(Q_nx_n)\left[(F(x_1\ldots x_n) = 0) \leftrightarrow (\alpha_iF(x_1\ldots x_n) = 0)\right] \tag{4}$$

and from (3)

$$(Q_1x_1).....(Q_nx_n)\left[(F(x_1\ldots x_n) = 0) \leftrightarrow (\alpha_iF(\alpha_qx_1,\ldots\alpha_qx_n) = 0)\right] \tag{5}$$

Applying (2) to (5)

$$(Q_1x_1).....(Q_nx_n)\left[(F(x_1\ldots x_n) = 0 \leftrightarrow F(\alpha_qx_1,\ldots\alpha_qx_n) = 0\right] \tag{6}$$

But since $\alpha_qx_i \leq q$, hence

$$(Q_1x_1).....(Qx_n)(F(x_1....x_n) = 0) \leftrightarrow \cancel{(Q_1x_1)\ldots(Q_nx_n)}F(\xi_1,\ldots\xi_n) = 0 \tag{7}$$

where $\xi_i = 0,..,q$ for all i

Hence deciding the right hand side is a finite procedure consisting of at most $(q + 1)^n$ substitutions for $\xi_1....\xi_n$

i.e. the right hand side of (7) is decidable, and hence

$$(Q_1x_1)....(Q_nx_n)(F(x_1\ldots x_n) = 0)$$

where $F\varepsilon C'$, and Q_i for all i = 1...n is a quantifier, either existential or universal, is also decidable.

Theorem 5.2.2 The class of formulae

$$(Q_1x_1)....(Q_nx_n)(F(x_1....x_n) = G(x_1\ldots x_n)) \tag{8}$$

for any $F,G\varepsilon C(+,.)$ is undecidable.

Proof Consider the set of wffs in Peano's Arithmetic (System S as described In Mendelson P103). Every wff in S is equivalent to a formula in the form of (8). For, given any wff in S, put it in prenex normal form. The terms

are equations between polynomials with positive coefficients, though the terms may have negation signs on them. These are eliminated by replacing an expression $F \neq G$ by the equivalent expression $(Ex)(F+1+x = G)_v(Ey)(G+1+y = F)$ where x,y do not appear elsewhere in the formula. (Ex) and (Ey) may then be taken into the quantifiers at the beginning of the formula, giving a formula in the form $(Q_1x_1)\ldots(Q_nx_n)A$ where A is a disjunctive normal form without negation signs.

Each disjunct may now be converted to one equation as follows

$$[f_{il} = g_{il} \wedge \ldots\ldots \wedge f_{ir} = g_{ir}] \equiv [(f_{il} - g_{il})^2 + \ldots\ldots + (f_{ir} - g_{ir})^2] = 0$$
$$\equiv \left[\sum_{j=1..r}(f_{ij}^2 + g_{ij}^2) = 2\sum_{j=1..r} f_{ij}g_{ij}\right] \qquad (9)$$

So each disjunct can be replaced by one equation between polynomials, $f_i=g_i$ say. Then A is equivalent to

$$(f_1 = g_1)_v\ldots\ldots{}_v(f_r = g_r)$$

and

$$[(f_1 = g_1)_v\ldots\ldots{}_v(f_r = g_r)] \equiv [(f_1 - g_1).(f_2 - g_2). \ldots .(f_r - g_r) = 0]$$
$$\equiv (F = G) \qquad (10)$$

where F,G are polynomials, arranged so that all their coefficients are positive.

So to every formula in Peano's Arithmetic there corresponds a formula in the form of (8). So if the class of formulae in the form of (8) were decidable then Peano's Arithmetic would also be decidable. Since it is not, then the class of formulae of form (8) is undecidable.

Theorem 5.2.3 The class of formulae

$$(Q_1x_1)\ldots(Q_nx_n)(F(x_1\ldots x_n) = G(x_1\ldots x_n)) \qquad (11)$$

where $F,G \in C(+,\dot{-})$ is decidable.

Proof Any occurrence of the function $\dot{-}$ in F or G or both is eliminated by use of the following type of equivalence

$$(Q_1x_1)\ldots(Q_nx_n)(F(x\dot{-}y) = G) \leftrightarrow (Q_1x_1)\ldots(Q_nx_n)(Ez)((y+z=x \wedge F(z) = G) \vee F(0) = G) \qquad (12)$$

where z is different from $x_1 \ldots x_n$.

We can now prove by induction on the structure of F and G using (12) that

$$(Q_1x_1)\ldots(Q_nx_n)(F = G) \leftrightarrow (Q_1x_1)\ldots(Q_nx_n)(Q_{n+1}x_{n+1})\ldots(Q_{n+m}x_{n+m})N \qquad (13)$$

where N is a disjunctive normal form of equations A = B where A,B are functions obtained in applying (12) to F,G respectively and $(Q_{n+1}x_{n+1})$ to $(Q_{n+m}x_{n+m})$ arise in the elimination of the recursive difference using L. The equations A = B which form the terms or atomic formulae of M are (or may be put in the form of) equations between linear forms, with all the coefficients positive. So M can be thought of as a statement in Hilbert and Bernay's system of formal arithmetic, system D [8] and hence is decidable by Presburger's algorithm [14]. Hence the original set of equations (11) is decidable.

So if one considers the class of equations F = G, prefixed by any sequence of quantifiers, knowing that if F,G are elementary that the class is undecidable, we try restricting the class of functions. However, even when the functions are restricted to those built by composition from addition and multiplication, the class of equations $(Q_1x_1)\ldots(Q_nx_n)(F = G)$ is undecidable (Theorem 5.2.2). A different result is obtained if multiplication is replaced by recursive difference (Theorem 5.2.3)

If equations F = 0 are considered, the decidability of identities F = 0 for $F \in C'$ is shown in Theorem 2.4.1 and holds also when any sequence of quantifiers precedes F = 0 (Theorem 5.2.1)

§3 Existence Results

First we define an exponential diophantine function to be any function built by composition from the addition, multiplication and exponential functions. (This class of functions is labelled C(+,., exp). It was shown in [2] that the class of formulae $(Ex_1)\ldots Ex_n)(F(x_1\ldots x_n) = G(x_1\ldots x_n))$ where $F,G \in C(+,., \exp)$, is undecidable. Since $x^y = \prod_{t \leq y}(x)$

$C(+,.,\exp) \subseteq C(+,.,\Pi)$

Theorem 5.3.1 The class of formulae $(Ex_1 \ldots Ex_n)(F(x_1 \ldots x_n) = G(x_1 \ldots x_n))$ where $F,G \in C(+,.,\Pi)$ is undecidable, and hence the larger class $(Q_1x_1)\ldots.(Q_nx_n)(F(x_1 \ldots x_n) = G(x_1 \ldots x_n))$ where F,G are elementary, is undecidable. Hence we have an alternative proof of the result given in chapter 1, that the class of formulae

$$F(x_1 \ldots x_n) = 0$$

where F is any elementary function, is undecidable.[1]

The question now is how far we have to reduce the class of equations until we get a decidability result. We first prove an interesting preliminary theorem.

Theorem 5.3.2 If $C_1 \supseteq C(+,.)$ and $C_2 = C_1 \cup \{\dot{-}\}$ then the class of formulae $(Ex_1)\ldots(Ex_n)(F(x_1 \ldots x_n) = G(x_1 \ldots x_n))$ for $F,G \in C_1$ is decidable if and only if this class of equations is decidable for $F,G \in C_2$.

(i.e. if a class of functions contains + and ., the presence of $\dot{-}$ makes no difference to the decidability of equations with only the existential quantifier in the prefix).

Proof Any occurrence of $\dot{-}$ in an equation may be eliminated by use of the following type of equivalence

$$(Ex_1)\ldots(Ex_n)(F(f\dot{-}g) = G) \leftrightarrow (Ex_1)\ldots(Ex_n)(Ex_{n+1})((g+x_{n+1}=f) \wedge F(x_{n+1}=G) \vee (F(0) = G)) \qquad (14)$$

where x_{n+1} is distinct from x_i, $i = 1\ldots.n$.

By repeated application of such equivalences the original equation can be reduced to the form

$$(Ex_1)\ldots(Ex_n)\ldots(Ex_{n+m})M(x_1 \ldots x_n \ldots x_{n+m}) \qquad (15)$$

where $M(x_1 \ldots x_n \ldots x_{n+m})$ is a disjunctive normal form of equations between functions belonging to C_1.

---

[1] Since the elementary functions include $\dot{-}$, the class of equations F = G is the same as the class F = 0.

$$\text{i.e. } M(x_1 \ldots x_{n+m}) = \bigvee_{i=1}^{s} \bigwedge_{j=1}^{r} (f_{ij} = g_{ij}) \leftrightarrow \bigvee_{i=1}^{s} \left(\sum_{j=1}^{r} (f_{ij} - g_{ij})^2 = 0 \right)$$

$$\leftrightarrow \left(\prod_{i=1}^{s} \sum_{j=1}^{r} (f_{ij} - g_{ij})^2 \right) = 0$$

$$\leftrightarrow F' = G'$$

where F', G' do not contain the recursive difference $\dot{-}$, and hence $F', G' \varepsilon C_1$.

If $F, G \varepsilon C_1$ then a fortiori they belong to C_2[1].

Theorem 5.3.3 If $C_1 \supseteq C(+\ .)$ and $C_2 = C_1 \cup \{\dot{-}\}$ then the class of formulae $(Q_1 x_1) \ldots (Q_n x_n)(F(x_1 \ldots x_n) = G(x_1 \ldots x_n))$ for $F, G \varepsilon C_1$ is decidable if and only if the formulae $(Q_1 x_1) \ldots (Q_n x_n)(F(x_1 \ldots x_n) = G(x_1 \ldots x_n))$ for $F, G \varepsilon C_2$ is decidable.

Proof Almost identical to that of 5.3.2, replace (Ex_i) by $(Q_i x_i)$ for all $i = 1 \ldots n$.

Corollary to Theorem 5.3.2 If $F, G \varepsilon C(+, ., \dot{-})$ the class of equations

$$(Ex_1) \ldots (Ex_n)(F(x_1 \ldots x_n) = G(x_1 \ldots x_n)) \tag{16}$$

is decidable if and only if the class

$$(Ex_1) \ldots (Ex_n)(F(x_1 \ldots x_n) = G(x_1 \ldots x_n) \tag{17}$$

is decidable for $F, G \varepsilon C(+\ .)$

The decidability of the latter class of equations is simply Hilbert's 10th problem, which remains unsolved.

---

[1] It might be thought that since $(x_1 \ldots x_n)(F = G)$ for $F, G \varepsilon C(+\ .)$ is a decidable set of equations then $(x_1 \ldots x_n)(F = G)$ for $F, G \varepsilon C(+, ., \dot{-})$ must be decidable. This is not the case as we know from Theorem 4.4.1 and the explanation is that although decidability of $(Ex_1) \ldots (Ex_n)\ F = G$ for $F, G \varepsilon C(+\ .)$ led to decidability for $F, G \varepsilon C(+, ., \dot{-})$ that is because only existential quantifiers arise in the proof.

APPENDIX A

THE SUBSTITUTION SCHEMA IN RECURSIVE ARITHMETIC

In his paper Logic Free Formalisations of Recursive Arithmetic [5] and subsequently in his book, Recursive Number Theory [6] R.L. Goodstein presents a formalisation of primitive recursive arithmetic in which the only axioms are explicit and recursive function definitions, and the rules of inference are the schemata

$$\frac{F(x) = G(x)}{F(A) = G(A)} \qquad (Sb_1)$$

$$\frac{A = B}{F(A) = F(B)} \qquad (Sb_2)$$

$$\frac{\begin{array}{c}A = B \\ A = C\end{array}}{B = C} \qquad (T)$$

where F(x), G(x) are recursive functions and A,B,C are recursive terms, and the primitive recursive uniqueness rule

$$\frac{F(Sx) = H(x,F(x))}{F(x) = H^{x}F(0)} \qquad (U)$$

where the iterative function $H^{x}t$ is defined by the primitive recursion $H^{0}t = t$, $H^{Sx}t = H(x,H^{x}t)$; in U, F may contain additional parameters.

In the same paper it is shown that the schema U may be replaced by

$$\frac{F(0) = 0 \quad F(Sx) = F(x)}{F(x) = 0} \qquad (E)$$

if we take as axioms

$$a + (b \dot{-} a) = b + (a \dot{-} b) \qquad (A)$$

and, in place of the introductory equations for the predecessor function,

$$Sa \dot{-} Sb = a \dot{-} b \qquad (P)$$

This system is referred to as R_1.

The purpose of this paper is to present another formalisation, R*, which also weakens U and yet avoids taking A as an axiom.

The rules of inference of R* are Sb_1, Sb_2, T and

$$\frac{F(Sx) = F(x)}{F(x) = F(0)} \qquad (E_1)$$

$$F(0) = G(0) \qquad (E_3)\dagger$$

$$\frac{F(Sx) = G(Sx)}{F(x) = G(x)}$$

In place of the recursive definitions of addition we have the axioms

$a + 0 = a$ (A_1) $\qquad\qquad$ $a + (b + c) = (a + b) + c$ (A_2)

and for subtraction, we have the recursive definitions of predecessor and difference

$0 \dot{-} 1 = 0$ (S_1); $Sa \dot{-} 1 = a$ (S_2); $a \dot{-} 0 = a$ (S_3); $a \dot{-} Sb = (a \dot{-} b) \dot{-} 1$ (S_4);

and the axiom

$(a \dot{-} b) \dot{-} 1 = (a \dot{-} 1) \dot{-} b$ (S_5)

We have also the recursive definition of multiplication

$a \cdot 0 = 0$ (M_1) $\qquad\qquad$ $a \cdot Sb = a \cdot b + a$ (M_2)

Exactly as in [6] we may prove the following results

$$\frac{A = B}{B = A} \qquad (K)$$

and $Sa \dot{-} Sb = a \dot{-} b$, $a \dot{-} a = 0$, $0 \dot{-} a = 0$, $(a + b) \dot{-} b = a$, $(a + n) \dot{-} (b + n) = a \dot{-} b$, $n \dot{-} (b + n) = 0$.

We now derive the schema,

$$\frac{F(Sx) = SF(x)}{F(x) = F(0) + x} \qquad (U_2)$$

(I am indebted to R.L. Goodstein for the following proof). Write $G(x) = F(0) + x$, then $G(Sx) = SG(x)$ and $G(0) = F(0)$. Using these two results and $F(Sx) = SF(x)$ we deduce $F(x) = G(x)$ for if $L(x) = F(x \dot{-} 1) +$

---

†Retaining the notation of [6].

$\{1 \dot{-} (1 \dot{-} x)\}$, then $L(0) = F(0)$ and $L(Sx) = SF(x) = F(Sx)$ so that, by E_2, $L(x) = F(x)$.

Therefore

$$F(x) = F(x \dot{-} 1) + \{1 \dot{-} (1 \dot{-} x)\}$$

Let $\phi(n,x)$ be defined by

$$\phi(0, x) = 0 \qquad \phi(Sn,x) = \{1 \dot{-} (1 \dot{-} (x \dot{-} n))\} + \phi(n,x)$$

then

$$F(x \dot{-} n) + \phi(n,x) = \{F(x \dot{-} Sn) + [1 \dot{-} (1 \dot{-} (x \dot{-} n))]\} + \phi(n,x)$$

$$= F(x \dot{-} Sn) + \phi(Sn,x)$$

Using E_1

$$F(x \dot{-} n) + \phi(n,x) = F(x \dot{-} 0) + \phi(0,x) = F(x)$$

Whence taking $n = x$

$$F(0) + \phi(x,x) = F(x)$$

Similarly

$$G(0) + \phi(x,x) = G(x)$$

Hence

$$F(x) = G(x)$$

$$F(x) = F(0) + x.$$

We now use U_2 to prove

$$0 + a = a.$$

Write $F(a) = a$, then $F(Sa) = SF(a)$. Hence using U_2 and K, $0 + a = a$. Similarly using U_2 we may prove $a + Sb = Sa + b$, $a + b = b + a$, $(a + b) \dot{-} a = b$ and exactly as in [6]

$$a + (b \dot{-} a) = b + (a \dot{-} b).$$

Now from E_1, E follows immediately and hence we have postulated or derived all the axioms and rules of inference of system R_1, given in [6]. Hence the sufficiency of R* for the construction of primitive recursive arithmetic follows from the sufficiency of R_1, which is proved in [6]. In fact we can reduce the axiom system R* by postulating only certain

special cases of Sb_2. The special cases are

$$\frac{A = B}{x+A = x+B} \quad (Sb_{21}) \qquad \frac{A = B}{A \dot{-} x = B \dot{-} x} \quad (Sb_{22}) \qquad \frac{A = B}{x \dot{-} A = x \dot{-} B} \quad (Sb_{23})$$

$$\frac{A = B}{F(A) = F(B)} \quad (Sb_{24})$$

where in Sb_{24} A = B is restricted to one of the initial equations A_1, A_2, S_1, S_2, S_3, S_4, S_5, M_1, M_2, or is any recursive or explicit function definition. For $F(0 \dot{-} Sx)) = F((0 \dot{-} x) \dot{-} 1) = F((0 \dot{-} 1) \dot{-} x) = F(0 \dot{-} x)$ using Sb_{24} for the equations $a \dot{-} Sb = (a \dot{-} b) \dot{-} 1$, $(a \dot{-} b) \dot{-} 1 = (a \dot{-} 1) \dot{-} b$, $0 \dot{-} 1 = 0$, and Sb, to substitute 0 for x and x for b. Now writing $G(x) = F(0 \dot{-} x)$, we have proved $G(Sx) = G(x)$ and hence from U_1, $G(x) = G(0)$

therefore

$$F(0 \dot{-} x) = F(0) \qquad 0.1$$

Similarly $F(Sx \dot{-} Sx) = F((Sx \dot{-} x) \dot{-} 1) = F((Sx \dot{-} 1) \dot{-} x) = F(x \dot{-} x)$ and hence by U_1

$$F(x \dot{-} x) = F(0). \qquad 0.2$$

The proofs of the results in the first part of this paper up to and including the proof of $a + (b \dot{-} a) = b + (a \dot{-} b)$ use only the above special cases of Sb_2 and 0.1 and 0.2.

Using $a + b = b + a$ and Sb_{21} we have

$$\frac{A = B}{A+x = B+x} \qquad (Sb_{25})$$

Following the proof, as given in [6], of the sufficiency of R_1 (and therefore of R*, since in R* we have derived or postulated all the axioms and rules of R_1), we may derive the schema

$$\frac{A \dot{-} B = 0,\ B \dot{-} A = 0}{A = B} \qquad (A)$$

The schema

$$\frac{A = B}{Ax = Bx} \qquad (Sb_{26})$$

is now proved as follows

Using Sb_{24}, $A.Sx \dot{-} B.Sx = A.Sx \dot{-} B.x + B = A.x + A \dot{-} B.x + B$. Assuming $A = B$, from Sb_{21}, $z + A = z + B$, and hence from Sb_1, $B.x + A = B.x + B$. Therefore using Sb_{23}, $z \dot{-} (B.x + B) = z \dot{-} (B.x + A)$ and hence from Sb_1 $(A.x + A) \dot{-} (B.x + B) = (A.x + A) \dot{-} (B.x + A)$; but from a previous result $(A.x + A) \dot{-} (B.x + A) = A.x \dot{-} B.x$

Hence

$$A.Sx \dot{-} B.Sx = A.x \dot{-} B.x$$

Using E_1,

$$A.x \dot{-} B.x = 0.$$

Similarly

$$B.x \dot{-} A.x = 0.$$

Hence, by A.

$$A.x = B.x.$$

Exactly as in [6], we may now prove $Sa.b = a.b + b$, $0.a = 0$ and $a.b = b.a$. The schema

$$\frac{A = B}{xA = xB} \qquad (Sb_{27})$$

follows from $a.b = b.a$ and Sb_{26}.

Apart from the special cases of Sb_2 which are axioms or have been derived the only application of Sb_2 in the proof of the sufficiency of R^* occurs in the proof of the substitution theorem, in the form

$$\frac{x + (y \dot{-} x) = y + (x \dot{-} y)}{F(x + (y \dot{-} x)) = F(y + (x \dot{-} y))}$$

I shall give an alternative proof of the substitution theorem which avoids use of this result.

THE SUBSTITUTION THEOREM

$$x = y \rightarrow F(x) = F(y)$$

All primitive recursive functions can be obtained by substitution and recursion according to the schema $F(0) = 0$ $F(Sx) = H(F(x))$, from the initial functions $u + v$, $u \dot{-} v$, $Rt(u)$, where $Rt(0) = 0$, $Rt(Sx) =$

$Rt(x) + [1 \dot{-} p(x,Rt(x))]$ and $p(x,y) = (Sy)^2 \dot{-} Sx$.

It suffices therefore to prove that the substitution theorem holds for these initial functions and is preserved under substitution and the given recursion. From the original proof of the substitution theorem given in [6], we have

$$(1 \dot{-} |x,y|)F(x + (y \dot{-} x)) = (1 \dot{-} |x,y|)F(x)$$

$$(1 \dot{-} |x,y|)F(y + (x \dot{-} y)) = (1 \dot{-} |x,y|)F(y)$$

In the case of $F(z) = z + a$ we have

$$(1 \dot{-} |x,y|)((x + (y \dot{-} x)) + a) = (1 \dot{-} |x,y|)(x + a)$$

$$(1 \dot{-} |x,y|)((y + (x \dot{-} y)) + a) = (1 \dot{-} |x,y|)(y + a)$$

But from Sb_{25} and $x + (y \dot{-} x) = y + (x \dot{-} y)$, $[x + (y \dot{-} x)] + a = [y + (x \dot{-} y)] + a$ and hence from Sb_{27}

$$(1 \dot{-} |x,y|)[(x + (y \dot{-} x)) + a] = (1 \dot{-} |x,y|)[(y + (x \dot{-} y)) + a]$$

Hence

$$(1 \dot{-} |x,y|)(x + a) = (1 \dot{-} |x,y|)(y + a)$$

Thus we have derived the substitution for the function $F(z) = z + a$.

In the way, using Sb_{21}, Sb_{22}, Sb_{23}, Sb_{25}, Sb_{27}, we may obtain the substitution theorem for the initial functions $u + v$, $u \dot{-} v$, $u.v$.

In the following proof of the substitution theorem for the function $Rt(x)$, I shall use theorems of the propositional calculus, which may easily be proved by deriving their corresponding equations in recursive arithmetic. The theorems concerned are

$$(x = x') \rightarrow (Sx = Sx') \tag{1}$$

$$(y = y') \rightarrow (Sy)^2 = (Sy')^2 \tag{2}$$

$$((x = x') \,\&\, (y = y')) \rightarrow (Sy)^2 \dot{-} Sx = (Sy')^2 \dot{-} Sx' \tag{3}$$

$$(x=x') \,\&\, (Rt(x) = Rt(x')) \rightarrow p(x,Rt(x)) = p(x',Rt(x')) \tag{4}$$

$$(x=x') \,\&\, (Rt(x) = Rt(x')) \rightarrow Rt(x) + (1 - p(x,Rt(x))) = Rt(x')+(1-p(x', Rt(x')) \tag{5}$$

$$(x=x') \,\&\, (Rt(x) = Rt(x')) \rightarrow Rt(Sx) = Rt(Sx') \tag{6}$$

We now prove

$((x = x') \to Rt(x) = Rt(x')) \to (Sx = Sx' \to Rt(Sx) = Rt(Sx'))$

with a,b,c standing for $|x,x'|$ (and hence for $|Sx,Sx'|$, $|Rt(x), Rt(x')|$, $|Rt(Sx), Rt(x')|$ respectively), we require to prove

$$(1 \dot{-} (1 \dot{-} a)b)(1 \dot{-} a)c = 0. \tag{7}$$

From (6)

$$(1 \dot{-} (a + b))c = 0.$$

so that

$$(1 \dot{-} (a + b))(1 \dot{-} a)c = 0.$$

Hence

$$(1 \dot{-} a)c \dot{-} b(1 \dot{-} a)c = 0$$

because $a(1 \dot{-} a) = 0$. Therefore

$$(1 \dot{-} a)c(1 \dot{-} b) = 0. \tag{8}$$

Hence

$$\begin{aligned}(1 \dot{-} (1 \dot{-} a)b)(1 \dot{-} a)c &= (1 \dot{-} a)c \dot{-} (1 \dot{-} a)(1 \dot{-} a)bc \\ &= (1 \dot{-} a)c \dot{-} (1 \dot{-} a)bc \\ &= (1 \dot{-} a)c(1 \dot{-} b) \\ &= 0\end{aligned} \tag{9}$$

from (8). Therefore

$$(x = x' \to Rt(x) = Rt(x')) \to (Sx = Sx' \to Rt(Sx) = Rt(Sx')) \tag{10}$$

Now define $P(x,x') = (1 \dot{-} |x,x'|)\left|Rt(x), Rt(x')\right|$

Then, from (9),

$$P(x,x') = 0 \to P(Sx, Sx') = 0.$$

But, from E_3,

$$P(x,0) = (1 \dot{-} x)|Rt(x), Rt(0)| = 0.$$

Similarly

$$P(0,x') = 0.$$

Hence, by I_2,

$$P(x,x') = 0.$$

$$(x = x') \to \{Rt(x) = Rtx'\}$$

We have now proved the substitution theorem for all the initial functions.

Now suppose the substitution theorem holds for the particular functions f,g, i.e.

$$x = y \to f(x) = f(y) \tag{11}$$

and

$$x = y \to g(x) = g(y). \tag{12}$$

From Sb_1 and (11) we have

$$g(x) = g(y) \to f(g(x)) = f(g(y)). \tag{13}$$

We now use the schema

$$\frac{p \to q \qquad q \to r}{p \to r} \tag{14}$$

which may be proved by a consideration of the corresponding equations in recursive arithmetic.

Hence from (12), (13),

$$x = y \to f(g(x)) = f(g(y))$$

i.e. the substitution theorem is preserved under composition.

Now consider $\phi(x)$ defined by the recursion $\phi(0) = 0$, $\phi(Sx) = H(\phi(x))$ and suppose the substitution theorem holds for H.

Define $P(x,y) = (1 \dot{-} |x,y|)|\phi(x), \phi(y)|$. Then, using E_3

$$P(x,0) = (1 \dot{-} x)|\phi(x) , \phi(0)| = 0 \tag{15}$$

and

$$P(0,Sy) = 0. \tag{16}$$

We now derive the result

$$(a = a') \to \{(b = b') \to (a = b \to a' = b')\}.$$

As we observed above

$$(1 \dot{-} |x,y|)F(x + (y \dot{-} x)) = (1 \dot{-} |x,y|)F(x)$$

and so with $F(x) = |x,t|$

$$(1 \dot{-} |x,y|)|x + (y \dot{-} x),t| = (1 \dot{-} |x,y|)\ x,t|.$$

Similarly

$$(1 \dot{-} |x,y|)|y + (x \dot{-} y),\ t| = (1 \dot{-} |x,y|)|y,t|$$

Using $x + (y \dot{-} x) = y + (x \dot{-} y)$, and the given special cases of Sb_2 we obtain

$$(1 \dot{-} |x,y|)|x + (y \dot{-} x),\ t| = (1 \dot{-} |x,y|)|y + (x \dot{-} y),t|.$$

Hence

$$(1 \dot{-} |x,y|)|x,t| = (1 \dot{-} |x,y|)|y,t| \tag{17}$$

Now using (17) and rearranging factors

$$(1 \dot{-} |a,a'|)(1 \dot{-} |b,b'|)(1 \dot{-} |a,b|)|a',b'| = (1 \dot{-} |b,b'|)(1 \dot{-} |a,a'|)$$
$$(1 \dot{-} |a,b|)|a,b|$$
$$= 0.$$

Hence

$$a = a' \to \{b = b' \to (a = b \to a' = b')\}. \tag{18}$$

Replacing a,a',b,b' by $H(\phi(x))$, $\phi(Sx)$, $H(\phi(y))$, $\phi(Sy)$ respectively

$$H(\phi(x)) = \phi(Sx) \to \{H(\phi(y)) = \phi(Sy) \to (H(\phi(x)) = H(\phi(y)) \to \phi(Sx) = \phi(Sy))\}$$

From the definition of ϕ, using modus ponens twice

$$H(\phi(x)) = H(\phi(y)) \to \phi(Sx) = \phi(Sy)$$

Using the substitution theorem for H,

$$\phi(x) = \phi(y) \to H(\phi(x)) = H(\phi(y))$$

and hence by schema (14)

$$\phi(x) = \phi(y) \to \phi(Sx) = \phi(Sy). \tag{19}$$

We now prove

$$P(x,y) = 0 \to P(Sx,Sy) = 0. \tag{20}$$

With a, b, c standing for $|x,y|$, $|\phi(x), \phi(y)|$, $|\phi(Sx), \phi(Sy)|$ respectively there is represented by the equation

$$(1 \dot{-} (1 \dot{-} a)b)(1 \dot{-} a)c = 0 \tag{21}$$

With f(a) standing for the left hand side, $f(Sa) = 0$ and $f(0) = (1 \dot{-} b)c = 0$ from (19) and hence, using E_3 $f(a) = 0$.

Now using I_2 with conditions satisfied by (15), (16), (20), we obtain

$x = y \rightarrow \phi(x) = \phi(y)$

Hence the substitution theorem is preserved under the given recursion and thus it holds for all recursive functions.

APPENDIX B

A Solution to the Integer Linear Programming Problem.

§1. Introduction The General Integer Linear Programming Problem, namely, optimising a linear function subject to non-negative solutions of a set of simultaneous linear inequalities, was one of the major unsolved problems in the theory of linear programming until 1958, when solutions were produced by R. Gomory [3], and A.H. Land and A.G. Doig.[12] Yet a technique for solving this problem was available in 1929 in a result in the field of mathematical logic due to M. Presburger. Of course it must be said that the subject of Linear Programming did not exist in 1929, and so, in this context, Presburger's result was a solution to a non-existent problem. I shall explain briefly in §2 why this much earlier result does solve the programming problem, and in §3, describe in detail an algorithm derived from Presburger's result.

§2. Presburger's Algorithm

Presburger [14] provided a decision procedure for the formal system of arithmetic referred to as System D in Hilbert & Bernays Volume 1 [8]. System D is the 1st order theory with one predicate, equality, one constant 0, and the two functions, S (the successor function), and the addition function. By applying the function S repeatedly to the constant 0, we have all the natural numbers in the system, and applying the addition function to the variables and constants we obtain as terms of the system, linear forms in any number of variables, with positive coefficients and constants. The atomic formulae of the system are the expressions s = t where s,t are terms, and hence the atomic formulae are just linear equations with the variables and constants on their positive side.

Thus the system contains the propositional connectives for "not", "and", and "or" and the usual two connectives (Ex) and (x), so the

formulae which Presburger's procedure enables one to decide the truth or falsity of, are conjunctions and disjunctions of linear equations involving the two quantifiers (Ex) and (x). In fact to solve the linear programming problem we only need to solve sentences of the form

$$(Ex_1)(Ex_2).....(Ex_n) \bigwedge_{i=1...m} (A_i(x_1...x_n) = B_i(x_1...x_n))$$

The integer linear programming problem is to optimise a linear form

$$c_1x_1 + c_2x_2 + ... + c_nx_n \qquad (1)$$

subject to non-negative integer solutions of a set of simultaneous inequalities

$$\begin{aligned} &a_{11}x + a_{12}x + ... + a_{1n}x_n \leq b_1 \\ &. \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \\ &. \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \\ &a_{m1}x + a_{m2}x + ... + a_{mn}x_n \leq b_m \end{aligned} \qquad (2)$$

with the condition that

$$x_i \geq 0 \text{ and } x_i \text{ is an integer for } i = 1, ... n \qquad (3)$$

where a_{ij}, b_i are integers, positive, negative or zero.

For the purposes of this part we shall assume that the inequalities have been made into equations by the use of slack variables, and further that each variable and the constants have been transferred to that side on which their coefficients are positive. Thus we replace (2) by equations

$$\begin{aligned} A_1(x_1....x_t) &= B_1(x_1....x_t) \\ \vdots \qquad &\qquad \vdots \\ A_m(x_1....x_t) &= B_m(x_1....x_t) \end{aligned} \qquad (4)$$

where A_i, B_i $i = 1 \ldots m$ are linear, possibly non-homogeneous forms in $x_1....x_t$, $t \geq n$.

If we wish to optimise the linear form (1) let

$$c_1x_1 + ... + c_nx_n = z \qquad (5)$$

and then the optimum value of z subject to (4) and (5) will be the optimum of (1) subject to (4). We now put (5) in the form

$$C_1(x_1x_2\ldots x_nz) = C_2(x_1x_2\ldots x_n,z)$$

where C_1, C_2 are also linear forms with $x_1, x_2 \ldots z$ on their positive side

The expression

$$(Ez)(Ex_t)(Ex_{t-1})\ldots(Ex_1)\left\{\left(\bigwedge_{i=1\ldots m}((A_i(x\ldots x_t) = B_i(x_1\ldots x_t))\right.\right.$$
$$\left.\left.\wedge(C_1(x_1\ldots x_nz) = C_2(x_1\ldots x_nz))\right\}\right. \qquad (6)$$

is then a sentence in Hilbert and Bernays System D and thus we can use Presburger's method for eliminating all the variables until only z and constants remain.

The expression (6) is already in the form in which Presburger's procedure has to put all statements in system D before proceeding with the algorithm i.e. the form $(Ex_n)\ldots(Ex_1)\ A\ (x_1\ldots x_nz)$ where A is quantifier free and in disjunctive normal form. Furthermore we do not need to deal with negation signs as Presburger does. We eliminate equality signs by replacing $a = b$ by $(a < b + 1)\wedge(b < a + 1)$. Let us assume that $r - 1$ variables have been eliminated and that the resulting equivalent statement is in the form

$$(Ez)(Ex_n)\ldots(Ex_r)\left\{\left(\bigwedge_{i\in I_1}(p_ix_r < t_i)\right)\wedge\left(\bigwedge_{i\in I_2}p_ix_r > t_i\right)\wedge\left(\bigwedge_{i\in I_3}p_ix_r \equiv t_i(k_i)\right)\right\} \qquad (7)$$

t_i a function of $x_{r+1}\ldots x_nz$.

I_1 I_2 and I_3 being index sets. The need for congruences will be seen later

Let $p = \text{L.C.M.}\ \{p_i \mid i\in I_1 \cup I_2 \cup I_3\}$. Multiply both sides of each relation by $\frac{p}{p_i}$ to change (7) into

$$(Ez)(Ex_n)\ldots(Ex_r)\left\{\left(\bigwedge_{i\in I_1}(px_r < t_i')\right)\wedge\left(\bigwedge_{i\in I_2}px_r > t_i'\right)\wedge\left(\bigwedge_{i\in I_3}px_r \equiv t_i'(k_i')\right)\right\}$$

where $t_i' = \frac{p}{p_i} \cdot t_i$ and $k_i' = \frac{p}{p_i} \cdot k_i$

We consider the formula,

$$(Ex_r)\left\{\begin{array}{ll} px_r < t_i' & i \in I_1 \\ px_r > t_i' & i \in I_2 \\ px_r \equiv t_i'(k_i') & i \in I_3 \end{array}\right\} \tag{8}$$

Let $y = px_r$, then (8) is equivalent to

$$(Ey)\left\{\begin{array}{ll} y < t_i' & i \in I_1 \\ y > t_i' & i \in I_2 \\ y \equiv t_i'(k_i') & i \in I_3 \\ y \equiv 0(p) & \end{array}\right\} \quad \text{(N.B. this is the stage at which congruences arise)} \tag{9}$$

$$\text{i.e.} \quad (Ey)\left\{\begin{array}{ll} y < t_i' & i \in I_1 \\ y > t_i' & i \in I_2 \\ y \equiv t_i'(k_i') & i \in I_3' \end{array}\right\} \quad I_3' \text{ being } I_3 \text{ enlarged to include } y \equiv 0(p) \tag{10}$$

(10) is then equivalent to

$$\bigvee_{j \in I_2} \bigvee_{r=1,2,\ldots K} \left\{\begin{array}{ll} t_j' + r < t_i' & i \in I_1 \\ t_j' + r > t_i' & i \in I_2 \\ t_j' + r \equiv t_i'(k_i') & i \in I_3' \\ t_\ell' \equiv t_m(k_\ell', k_m') & \end{array}\right. \tag{11}$$

where $K = \text{L.C.M.}\{k_i' | i \in I_3'\}$. The numbers ℓ, m range over all possible values of i , $i \in I_3'$.

For if (11) holds put $y = t_j' + r$ for the appropriate j, r and hence (10) holds.

If $I_2 \neq \phi$, then if (10) holds, then for any set of values of the variables substitute a set of values in (10) and let

$t_j' = \max\{t_i' | i \in I_2\}$ and $t_l' = \min\{t_i' | i \in I_1\}$. From (10) $t_j' \leq y \leq t_l'$

and $y \equiv t_i'(k_i')$, so by the generalised Chinese Remainder Theorem there

exists a number $r = 1,\ldots.K$ such that $t'_j + r \equiv t'_i(k'_i)$ for all $i\varepsilon I'_3$ if and only if $t'_\ell \equiv t'_m \bmod (k'_\ell, k'_m)$ for $\ell, m \varepsilon I'_3$ where (k'_ℓ, k'_m) denotes the g.c.d of k'_ℓ and k'_m.

If $I_2 = \phi$, replace $t'_j = \max\{t'_i | i\varepsilon I_2\}$ by $t'_j = t'_1 - K$.

To eliminate the next variable, x_{r+1}, we use the equivalence

$$(Ex)(A_1 \vee A_2 \vee \ldots \vee A_n) \longleftrightarrow (Ex)A_1 \vee (Ex)A_2 \ldots \vee (Ex)A_n$$

to bring existential quantifers onto each of the disjuncts separately.

Consider the disjunct now in the form

$$(Ex_n)\ldots(Ex_{r+1}) \begin{cases} t'_{j_o} + r_o < t'_i & i \varepsilon I_1 \\ t'_{j_o} + r_o > t'_i & i \varepsilon I_2 \\ t'_{j_o} + r_o \equiv t'_i(k'_i) & i \varepsilon I_3 \\ t'_\ell \equiv t'_m(\ell,m) & \ell, m \varepsilon I'_3 \end{cases} \tag{12}$$

This can now be put in the form

$$(Ex_n)\ldots(Ex_{r+1}) \begin{cases} p_i x_{r+1} < t_i & i \varepsilon I_{12} \\ p_i x_{r+1} > t_i & i \varepsilon I_{22} \\ p_i x_{r+1} \equiv t_i(k_i) & i \varepsilon I_{32} \end{cases} \tag{13}$$

We now repeat the procedure from (7) to eliminate x_{r+1} and so on until only z and constants remain. We than have a number of disjuncts, involving z, each disjunct being a conjunction of sentences of the kind

$$\begin{cases} a_i z > t_i & i \varepsilon I_p \\ a_i z < t_i & i \varepsilon I_q \\ a_i z \equiv t_i(b_i) & i \varepsilon I_s \end{cases} \tag{14}$$

plus a number of parts containing only constants. Clearly a maximum and minimum for z (if they exist) can be found. The congruences are then combined into one congruence using the Generalised Chinese Remainder

Theorem (stated and proved at the end of this appendix). We can then find the value(s) of z that satisfy (14). The parts containing only constants are easily decidable. If a part is false, the whole disjunct is discarded; if true the values of z satisfying (14) then satisfy the whole disjunct.

If we now take the union of the values of z that satisfy each disjunct, this union is the possible values that z and hence $c_1x_1 + c_2x_2 + \ldots + c_nx_n$ take. From this set we then choose the optimum value.

§3. The Derived Algorithm

The algorithm given below is derived from Presburger's. This algorithm, like Presburger's, bears a strong resemblance to the Fourier-Motzkin method of solving sets of linear inequalities, but congruences are introduced to cope with the requirement that the variables have integer values.

We make no claims for the efficiency of the algorithm, merely that it is an alternative solution, and one wholly different in approach from those in current use.

To find the maximum[1] of (1), subject to (2) and (3) suppose x_{n+1} is a variable subject to the condition

$$c_1x_1 + \ldots\ldots + c_nx_n \geq x_{n+1} \qquad (15)$$

Then the maximum of x_{n+1} subject to conditions (2), (3) and (15) will be the same as the maximum of (1) subject to (2) and (3). In fact condition (15) is of the same type as those in (2), so we may consider the more general problem of maximising x_{n+1} subject to the conditions

$$\begin{aligned}
&a_{11}x_1 + \ldots\ldots\ldots\ldots + a_{1n}x_n + a_{1,n+1}\,x_{n+1} \leq b_1\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&a_{m1}x_1 + \ldots\ldots\ldots\ldots a_{mn}x_n + a_{m,n+1}\,x_{n+} \leq b_m\\
&a_{m+1,1}\,x_1 + \ldots\ldots\ldots a_{m+1,n}\,x_n + a_{m+1,n+1}x_{n+1} \leq b_{m+1}
\end{aligned} \qquad (16)$$

1. To minimise, reverse the inequality sign in (15).

and

$$x_i \geq 0 \text{ and } x_i \text{ is an integer for } i = 1, \ldots n \qquad (17)$$

The range of values of x_{n+1} that satisfy (16) and (17) is the set of values of x_{n+1} that make

$$(Ex_n) \ldots (Ex_1)P(x_1 \ldots x_{n+1}) \qquad (18)$$

a true statement, where $P(x_1, \ldots x_{n+1})$ stands for the conjunction of conditions (16) and (17).

The procedure we give below finds a statement in the form

$$s_1(x_{n+1}) \vee S_2(x_{n+1}) \vee \ldots\ldots\ldots S_r(x_{n+1}) \qquad (19)$$

where $s_i(x_{n+1})$ is a conjunction of inequalities and congruences, this statement being equivalent[1] to (18).

On finding the statement (19) we can find the values of x_{n+1} that satisfy it and then choose the maximum.

Having found the maximum value we still have to find the values of $x_1 \ldots x_n$ that give that maximum.

To find a statement of (19) containing only x_{n+1} we eliminate the variables one at a time, finding in each case an equivalent statement with one less variable.

We shall explain the algorithm by showing how to eliminate the first two variables. In the case of the 2nd elimination we need the full

---

1. Equivalent in this sense means that any value of x_{n+1} that makes (19) true also makes (18) true, and vice versa (i.e. for each value of x_{n+1} that makes (18) true there exist values of $x_1 \ldots x_n$ which together with the value for x_{n+1}, make $P(x_1 \ldots x_{n+1})$ true, and any value of x_{n+1} for which there exist values of $x_1 \ldots x_n$ which make $P(x_1 \ldots x_{n+1})$ true, that value of x_{n+1} makes (19) true.

generality of the algorithm, and then the procedure is merely repeated to eliminate the remaining variables[2].

Eliminating x_1: Separate the inequalities (16) into the two types

$$\begin{aligned} a_i x_1 &\leq t_i \qquad i \in I_{11} \\ a_i x_1 &\geq t_i \qquad i \in I_{21} \end{aligned} \tag{20}$$

so that $a_i \geq 0$ for all i. I_{11}, I_{21} are index sets and t_i is a linear function of $x_2, \ldots . x_{n+1}$

Let $a_1 = \text{L.C.M.}\left\{ a_{i1} \mid i \in I_{11} \cup I_{21} \right\}$

(18) is then equivalent to

$$(Ex_n) \ldots . (Ex\ ) \begin{cases} a_1 x_i \leq t_i' & i \in I_{11} \\ a_1 x_i \geq t_i' & i \in I_{21} \\ x_i \geq 0 \text{ for } i = 2 \ldots n,\ x_i \text{ an integer for } i = 1, \ldots n \end{cases} \tag{21}$$

where $t_i' = \dfrac{a_1}{a_i} \cdot t_i$ and I_{21}' is I_{21} enlarged to include $x_1 \geq 0$, or rather $a_1 x_1 \geq 0$

Let $y = a_1 x_1$, (21) is then equivalent to

$$(Ex_n) \ldots (Ex_2)(Ey) \begin{cases} y \leq t_i' & i \in I_{11} \\ y \geq t_i' & i \in I_{21}' \\ y \equiv 0 (a_1) & \\ x_i \geq 0 \text{ and } x_i \text{ an integer for } i = 2 \ldots n \end{cases} \tag{22}$$

---

2. We can eliminate the variables in any order, and the order <u>will</u> affect the length of the algorithm, but for convenience we shall eliminate x_1 first, x_2 second etc.

which is equivalent to

$$(Ex_n) \ldots (Ex_2) \bigvee_{j \varepsilon I'_{21}} \bigvee_{r=0 \ldots a_1-1} \begin{cases} t'_j + r \leqslant t'_i & i \varepsilon I_{11} \\ t'_j + r \geqslant t'_i & i \varepsilon I'_{21} \\ t'_j + r \equiv 0 \ (a_1) & \\ x_i \geqslant 0 \text{ and } x_i \text{ an integer for } i = 2 \ldots n & \end{cases} \tag{23}$$

for, if a set of values for $y, x_2 \ldots x_{n+1}$ satisfy (22), substitute for $x_2 \ldots x_{n+1}$ in (22) and let $t'_j = \max \{t'_i | i \varepsilon I'_{21}\}$ and $t'_1 = \min \{t'_i | i \varepsilon I_{11}\}$. From (22) $t'_j \leqslant y \leqslant t'_1$ and $y \equiv 0(a_1)$. But for some $r = 0,1,\ldots a_1-1$, $t'_j + r \equiv 0(a_1)$ and hence $t'_j \leqslant t'_j + r \leqslant t'_1$ i.e. $t'_j + r \leqslant t'_i$ for $i \varepsilon I_{11}$, $t'_j + r \geqslant t'_i$ for $i \varepsilon I'_{21}$. Hence if a set of values for $y, x_2, \ldots x_{n+1}$ satisfy (22), then these values for $x_2 \ldots x_{n+1}$ satisfy (23).

If a set of values for $x_2 \ldots x_{n+1}$ satisfy (22) let $y = t'_j + r$, then these values for $y, x_2 \ldots x_{n+1}$ satisfy (23).

Thus (22) and (23) are equivalent.

We now apply the rule $(Ex)(A_1 \vee A_2 \ldots \vee A_s) \leftrightarrow (Ex)A_1 \vee (Ex)A_2 \vee \ldots \vee (Ex)A_s$ so (23) is equivalent to

$$\bigvee_{j \varepsilon I'_{21}} \bigvee_{r=0 \ldots a_1-1} (Ex_n) \ldots (Ex_2) \begin{cases} t'_j + r \leqslant t'_i & i \varepsilon I_{11} \\ t'_j + r \geqslant t'_i & i \varepsilon I'_{21} \\ t'_j + r \equiv 0(a_1) & \\ x_i \geqslant 0 \text{ and } x_i \text{ an integer for } i = 2 \ldots n & \end{cases} \tag{24}$$

So (24) is our equivalent statement not containing x_1.

Eliminating x_2:- we proceed separately with each of the disjuncts of (24). Any disjunct can be put in the form

$$(Ex_n) \ldots (Ex_2) \begin{cases} a_{i2} x_2 \leqslant t_i & i \varepsilon I_{12} \\ a_{i2} x_2 \geqslant t_i & i \varepsilon I_{22} \\ a_{i2} x_2 \equiv t_i(a_1) & i \varepsilon I_{32} \\ x_i \geqslant 0, \text{ and } x_i \text{ an integer for } i = 2 \ldots n & \end{cases} \tag{25}$$

Let a_2 = L.C.M. $\{a_{i2} | i \in I_{21} \cup I_{22} \cup I_{23}\}$ and let $y = a_2 x_2$

Then (25) is equivalent to

$$(Ex_n) \ldots (Ex_3)(Ey) \begin{cases} y \leqslant t_i' & i \in I_{12} \\ y \geqslant t_i' & i \in I_{22}' \\ y \equiv t_i'(a_1') & i \in I_{32} \\ y \equiv 0 \ (a_2) \\ x_i \geqslant 0 \text{ and } x_i \text{ an integer for } i = 3, \ldots n \end{cases} \tag{26}$$

These conditions differ in type from those in (22) because of the presence of 2 congruences rather than 1.

But

$$y \equiv t_i'(a_1') \longleftrightarrow \bigvee_{s=0,1\ldots a_1'-1} (y \equiv s(a_1') \quad t_i' \equiv s(a_1')) \tag{27}$$

substituting for $y \equiv t_i'(a_1')$ (19) can be put in the form

$$\bigvee_{s=0,1,\ldots a_1'-1} (Ex_n) \ldots (Ex_3)(Ey) \begin{cases} y \leqslant t_i' & i \in I_{12} \\ y \geqslant t_i' & i \in I_{22}' \\ y \equiv s(a_1') \\ t_i' \equiv s(a_1') & i \in I_{32} \\ y \equiv 0(a_2) \\ x_i \geqslant 0 \text{ and } x_i \text{ an integer for } i = 3 \ldots n \end{cases} \tag{28}$$

Since $t_i' \equiv s(a_1')$ does not contain y we use the rule $(Ex)(A \wedge B) \longleftrightarrow (Ex)A \wedge B$, where B does not contain x, to write each disjunct of (28) in the form

$$(Ex_n) \ldots (Ex_3)(Ey) \left\{ \begin{array}{l} y \leqslant t_i' \\ y \geqslant t_i' \\ y \equiv s(a_1') \\ y \equiv 0(a_2) \\ x_i \geqslant 0 \text{ and } x_i \text{ an} \\ \qquad \text{integer for} \\ \qquad i = 3 \ldots n \end{array} \right\} \wedge (Ex_n) \ldots (Ex_3)\{t_i \equiv s(a_1')\} \tag{29}$$

$i \in I_{32}$

We now combine the congruences $y \equiv s(a_1')$, $y \equiv 0(a_2)$ into one congruence using The Generalised Chinese Remainder Theorem (stated and proved at the end of this appendix) for the case n = 2.
Hence $y \equiv s(a_1')$ and $y \equiv 0(a_2)$ have a solution if and only if $s \equiv 0 \mod(a_1', a_2)$ and if this is so $y = ks \mod[a_1', a_2]$ where k is determined by a_1', a_2 and can be calculated using the Generalised Chinese Remainder Theorem.

So the part of (29) involving y is equivalent to

$$(Ex_n) \ldots (Ex_3)(Ey) \begin{cases} y \leqslant t_i' & i \in I_{12} \\ y \geqslant t_i' & i \in I_{22}' \\ y \equiv ks & \mod[a_1', a_2] \\ x_i \geqslant 0 \text{ and } x_i \text{ integer for } i = 3 \ldots n & \\ s \equiv 0 & \mod(a_1', a_2) \end{cases} \tag{30}$$

Each value of s which satisfies $s = 0 \mod (a_1', a_2)$ where $s = 0, 1, \ldots a_1' - 1$ gives a set of conditions in the same form as (22) and the elimination of y is carried out in the same way. So a second variable is eliminated and the process is repeated until only x_{n+1} remains. We then have a statement in the form

$$s_1(x_{n+1}) \vee \ldots\ldots \vee s_r(x_{n+1}) \tag{31}$$

where each $s_i(x_{n+1})$ is a conjunction of the form

$$\begin{aligned} a_{i,n+1} x_{n+1} &\leqslant t_i & i &\in I_{1,n+1} \\ a_{i,n+1} x_{n+1} &\geqslant t_i & i &\in I_{2,n+1} \\ a_{i,n+1} x_{n+1} &\equiv t_i(a_i) & i &\in I_{3,n+1} \end{aligned} \tag{32}$$

where the t_i's and a_i's are integers, and this statement is equivalent to the statement (18).

FINDING THE MAXIMUM VALUE FOR x_{n+1} :-

$$\text{Let } a_{n+1} = \text{L.C.M. } \{a_{i,n+1} \mid i \in I_{1,n+1} \cup I_{2,n+1} \cup I_{3,n+1}\}$$

Then (31) is equivalent to

$$a_{n+1}\, x_{n+1} \leq t'_i \qquad i \in I_{1,n+1}$$

$$a_{n+1}\, x_{n+1} \geq t'_i \qquad i \in I_{2,n+1}$$

$$a_{n+1}\, x_{n+1} \equiv t'_i \qquad i \in I_{3,n+1}$$

Let $m = \min \{t'_i \mid i \in I_{1,n+1}\}$ and $M = \max \{t'_i \mid i \in I_{2,n+1}\}$

Then

$$m \leq a_{n+1}\, x_{n+1} \leq M$$

Let $x = a_{n+1}\, x_{n+1}$

Then $m \leq x \leq M$, $x \equiv 0\ (a_{n+1})$ and $x \equiv t'_i\ (a'_i)$ (33)

Combining the 2 congruences, $x \equiv 0\ (a_{n+1})$ and $x \equiv t'_i\ (a'_i)$ if and only if $t'_i \equiv 0 \bmod (a_{n+1}, a'_i)$ and if this is so $x \equiv kt'_i \bmod [a_{n+1}, a'_i]$ where k is a number determined by a_{n+1} and a'_i (see §4. Generalised Chinese Remainder Theorem)

So the range of values of x_{n+1} satisfying (33) is the range of values of $\frac{x}{a_{n+1}}$ for which $x \equiv kt'_i \bmod [a_{n+1}, a'_i]$ and $m \leq x \leq M$

So the possible values of x_{n+1} is the union of the values of x_{n+1} that satisfy one of the disjuncts in (31). Having chosen the maximum value of x_{n+1} we see from (15) that it is also the maximum of $c_1x_1 + \ldots + c_nx_n$

FINDING THE CORRESPONDING VALUES OF $x_1, \ldots x_n$:-

During the elimination of variables, at some time we have a disjunction of sets of conditions like (32), but containing x_{n+1} and x_n. If we substitute the maximum value for x_{n+1} we are left with a set of conditions for x_n from which we can find the possible values for x_n, as

described for x_{n+1}. If there is more than one value for x_n, in the equivalent conditions on x_{n-1}, x_n, x_{n+1} we have to substitute the maximum of x_{n+1} together with each of the possible values of x_n in turn, to find the permissible values of x_{n-1}. This process is continued until we have at least one set of values for $x_1 \ldots x_n$ which when substituted in $c_1x_1 + \ldots . c_nx_n$ give the maximum value for x_{n+1}.

§4. The Generalised Chinese Remainder Theorem

Notation:- Let $(a_1, \ldots a_n)$ denote the g.c.d. of $a_1 \ldots a_n$

Let $[a_1, \ldots a_n]$ denote the L.C.M. of $a_1 \ldots a_n$

Let $A_i = \dfrac{[a_1 \ldots a_n]}{a_i}$

LEMMA If $(A_1, \ldots . A_n) = 1$ then thereexist integers $c_1 \ldots c_n$

such that $1 = c_1A_1 + c_2A_2 + \ldots + c_nA_n$ (34)

For the case n = 2 this result may be proved and the c_1 and c_2 calculated using the Euclidean Algorithm (see Birkhoff and MacLane A Survey of Modern Algebra pp 18, 19).

Generalised Chinese Remainder Theorem

$y \equiv x_i \bmod a_i$, $i = 1 \ldots n$ is equivalent to $x_i \equiv x_j \bmod (a_i, a_j)$

for $i, j = 1 \ldots n$ and $y \equiv \sum_{i=1}^{n} c_iA_ix_i \bmod [a_1 \ldots a_n]$

Proof If $y \equiv x_i \bmod a_i$ then $y = x_i + l_ia_i$ for some l_i

and if $y \equiv x_j \bmod a_j$ then $y = x_j + l_ja_j$ for some l_j

So $x_i - x_j = l_ja_j - l_ia_i$

$\equiv 0 \bmod (a_i, a_j)$

So $x_i \equiv x_j \bmod (a_i, a_j)$

Now $x_j - \sum_{i=1}^{n} c_iA_ix_i = \sum_{i=1}^{n} c_iA_i(x_j - x_i)$ from (34) (35)

But $(a_i,a_j) \mid (x_i - x_j)$ and $[A_i, A_j] \mid [a_1 \ldots a_n]$

But $[a_1 \ldots a_n] = A_i a_i$, $a_i a_j = (a_i,a_j)[a_i,a_j] \mid (x_j - x_i) A_i a_i$

$$a_j \mid A_i (x_j - x_i)$$

$$a_j \mid \sum_{i=1}^{n} c_i A_i (x_j - x_i)$$

So from (35) $x_j - \sum_{i=1}^{n} c_i A_i x_i \equiv 0 \bmod a_j$

i.e. $\sum_{i=1}^{n} c_i A_i x_i \equiv x_j \bmod a_j$ for all j

But $y \equiv x_j \bmod a_j$ for all j

So $y \equiv \sum_{i=1}^{n} c_i A_i x_i \bmod a_j$ for all j

So $y \equiv \sum_{i=1}^{n} c_i A_i x_i \bmod [a_1 \ldots a_n]$

If $y \equiv \sum_{i=1}^{n} c_i A_i x_i \bmod [a_1 \ldots a_n]$ and $x_i \equiv x_j \bmod (a_i,a_j)$

Then $x_i = x_j + l_j (a_i,a_j)$ for all i, j (36)

Replace the x_i's in $\sum_{i=1}^{n} c_i A_i x_i$, except x_j, using (36)

Then $y = \sum_{\substack{i=1 \\ i \neq j}}^{n} c_i A_i (x_j + l_j (a_i,a_j)) + c_j A_j x_j + k [a_1 \ldots a_n]$ for all j

$$= \left(\sum_{i=1}^{n} c_i A_i \right) x_j + \sum_{\substack{i=1 \\ i \neq j}}^{n} c_i \frac{[a_1 \ldots a_n]}{a_i} l_j (a_i,a_j) + k.[a_1 \ldots a_n]$$

for all j

$$= x_j + \sum_{\substack{i=1 \\ i \neq j}}^{n} \frac{c_i l_j}{a_i} \cdot a_i a_j + k [a_1 \ldots a_n] \text{ for all } j$$

$$= x_j + \sum_{\substack{i=1 \\ i \neq j}}^{n} c_i l_j a_j + k \left[a_1 \ldots a_n\right]$$

$$\equiv x_j \bmod a_j \qquad\qquad j = 1 \ldots n$$

In the case n = 2, which is the case we use c_1 and c_2 are easily found using the Euclidean Algorithm.

BIBLIOGRAPHY

1. Birkhoff G. and MacLane S.

A Survey of Modern Algebra. New York 1941

2. Davis M., Putnam H, and Robinson J.

The Decision Problem for Exponential Diophantine Equations. Annals of Math. Vol. 74, 1961 pp 425 - 436.

3. Gomory R.E.

Outline of an Algorithm for Solutions to Linear Programs. Bulletin of the A.M.A. Vol. 64, 1958 pp 275 - 6.

4. Gomory R.E.

An Algorithm for Integer Solutions to Linear Programs. National Acad. Sci. Vol. 53 No. 2, 1965. Page 260.

5. Goodstein R.L.

Logic-Free Formalisation of Recursive Arithmetic. Math. Scand. 2, 1954 pp 247 - 261.

6. Goodstein R.L.

Recursive Number Theory. Amsterdam 1957.

7. Goodstein R.L.

A Decidable Fragment of Recursive Arithmetic. Zeitschr. f. Math. Logik und Grundlagen d. Math. Vol. 9, 1963, pp. 199 - 201.

8. Hilbert D. and Bernays P.

Grundlagen der Mathematik Vol I,II. Berlin 1934, 1939.

9. Kalmar L.

Ein einfaches Beispiel für ein unentscheidbares arithmetisches Problem. Matematikai es fizikai lapok. Vol 50. 1943 pp. 1 - 23.

10. Kleene S.C.

Introduction to Metamathematics. Amsterdam, Gronigen, New York, 1952.

11. Kneebone G.T.

Mathematical Logic and the Foundations of Mathematics. London, New York, 1963.

12. Land A.H. and Doig A.G.

An Automatic Method of Solving Linear Programming Problems with some Integer Variables. Econometrica Vol. 28, 1960.

13. Mendelson E.

Introduction to Mathematical Logic. London, New York 1964.

14. Presburger M.

Uber die Vollstandigkeit etc.

Comptes-rendus du 1 Congres des Mathematiciens des Pays Slaves. Warsaw 1930 pp. 92 - 101.

15. Rousseau G.

A decidable Class of Number Theoretic Equations. Journal London Math. Soc. Vol. 41, 1966, pp. 737 - 741.

16. Shepherdson J.C.

A Non-Standard Model for a free variable fragment of Number Theory. Bull. Acad. Polon. Sér Sci. Math. Astr. Phys. Vol 12 1964 pp 79-86.

Zeitschr. f. math. Logik und Grundlagen d. Math.
Bd. 12, S. 235—239(1966)

A DECIDABLE CLASS OF EQUATIONS IN RECURSIVE ARITHMETIC

by R. L. Goodstein and R. D. Lee in Leicester (England)

1. Let C be the smallest class of functions from ordered sets of natural numbers into natural numbers which contains the initial functions

$$a \dot{-} x, x \dot{-} b, x + y, xy, x + (y \dot{-} x), x \dot{-} (x \dot{-} y)$$

(where a, b are arbitrary constants) and is such that $f \in C$ if there are functions g, h which belong to C and f is one of

$$a \dot{-} g, g \dot{-} b, g + h, gh, g + (h \dot{-} g), g \dot{-} (g \dot{-} h)$$

or

$$f(x_1, x_2, \ldots, x_n) = \sum_{x \leqq x_1} g(x, x_2, \ldots, x_n)$$

or

$$f(x_1, x_2, \ldots, x_n) = \prod_{x \leqq x_1} g(x, x_2, \ldots, x_n).$$

Further let C_k be the subclass of C in which $a \leqq k, b \leqq k$.

2. Our principal result is

Theorem 2. *If $f \in C$ the class of equations*

$$f(x_1, x_2, \ldots, x_n) = c,$$

where c is a constant, is decidable.

3. We start by defining the *height* and *spread* of members of C. Identity functions are of height 0 and spread 0. If f is of height 0 and spread μ and g is of height 0 and spread $\leqq \mu$,

$$f + g, fg, f + (g \dot{-} f), f \dot{-} (f \dot{-} g), a \dot{-} f$$

are of height 0 and spread $\mu + 1$.

If g is of height λ and spread μ then

$$\Sigma g, \Pi g, g \dot{-} b$$

are of height $\lambda + 1$ and spread μ.

If g is of height $\leqq \lambda + 1$ and spread $\leqq \mu$, and h is of height $\leqq \lambda + 1$ and of spread $\leqq \mu$, and if at least one of g, h is of height $\lambda + 1$, and at least one of spread μ, then

$$a \dot{-} g, g + h, gh, g + (h \dot{-} g), g \dot{-} (g \dot{-} h)$$

are of height $\lambda + 1$ and spread $\mu + 1$.

4. In preparation for Theorem 1 below we establish some properties of the function $\alpha_i(x) = i \dot{-} (i \dot{-} x)$ which leaves x unchanged if $x \leqq i$ and maps x on i if $x > i$.

If $j \geqq i$ then

$$\alpha_i(\alpha_j x) = \alpha_i x = \alpha_j(\alpha_i x). \tag{4.1}$$

$$\alpha_i(x + y) = \alpha_i(\alpha_j x + \alpha_j y). \tag{4.2}$$

$$\alpha_i(xy) = \alpha_i(\alpha_j x \cdot \alpha_j y). \tag{4.3}$$

$$\alpha_i\{x + (y \dot{-} x)\} = \alpha_i\{\alpha_j x + (\alpha_j y \dot{-} \alpha_j x)\}. \tag{4.4}$$

$$\alpha_i\{x \dot{-} (x \dot{-} y) = \alpha_i\{\alpha_j x \dot{-} (\alpha_j x \dot{-} \alpha_j y)\}. \tag{4.5}$$

If $j \geqq i \geqq a$ then

$$\alpha_i(a \dot{-} x) = \alpha_i(a \dot{-} \alpha_j x). \tag{4.6}$$

We omit the proofs.

If $j \geqq 2i$ and $i \geqq b$ then

$$\alpha_i(x \dot{-} b) = \alpha_i(\alpha_j x \dot{-} b). \tag{4.7}$$

For if $x \leqq j$, $\alpha_j x = x$ and if $x > j$ then $x \dot{-} b > i$ and $\alpha_j(x) \dot{-} b = j \dot{-} b \geqq i$ so that both sides of equation (4.7) have the value i.

If $r \geqq i$, $s \geqq i$ then

$$\alpha_i(x + ry) = \alpha_i(x + sy), \tag{4.8}$$

$$\alpha_i(xy^r) = \alpha_i(xy^s). \tag{4.81}$$

For (4.8) obviously holds for $y = 0$, and with $y \geqq 1$ both arguments exceed i, and (4.81) is obvious for $x = 0$, or $y = 0, 1$; if $x \geqq 1$ and $y \geqq 2$ then both xy^r and xy^s exceed i.

5. We observe next that if there are numbers p_r, $i \leqq p_r \leqq q$, $r = 1, 2, \ldots, n$ such that

$$\alpha_i f(x_1, x_2, \ldots, x_n) = \alpha_i f(\alpha_{p_1} x_1, \alpha_{p_2} x_2, \ldots, \alpha_{p_n} x_n) \tag{5.01}$$

then

$$\alpha_i f(x_1, x_2, \ldots, x_n) = \alpha_i f(\alpha_q x_1, \alpha_q x_2, \ldots, \alpha_q x_n). \tag{5.1}$$

For by (5.01),

$$\begin{aligned} \alpha_i f(\alpha_q x_1, \ldots, \alpha_q x_n) &= \alpha_i f(\alpha_{p_1}\alpha_q x_1, \ldots, \alpha_{p_n}\alpha_q x_n) \\ &= \alpha_i f(\alpha_{p_1} x_1, \ldots, \alpha_{p_n} x_n) \\ &= \alpha_i f(x_1, \ldots, x_n), \quad \text{by (5.01).} \end{aligned}$$

6. If to each i corresponds a $p \geqq i$ such that

$$\alpha_i f(x_1, x_2, \ldots, x_n) = \alpha_i f(\alpha_p x_1, \alpha_p x_2, \ldots, \alpha_p x_n), \tag{6.1}$$

and if

$$F(x_1, x_2, \ldots, x_n) = \sum_{x \leqq x_1} f(x, x_2, \ldots, x_n),$$

then for any $j \geqq p + i$

$$\alpha_i F(x_1, x_2, \ldots, x_n) = \alpha_i F(\alpha_j x_1, \alpha_j x_2, \ldots, \alpha_j x_n). \tag{6.2}$$

For, by (4.2),

$$\alpha_i F = \alpha_i \sum_{x \leqq x_1} \alpha_i f(x, x_2, \ldots, x_n) = \alpha_i \sum_{x \leqq x_1} f(\alpha_p x, \alpha_p x_2, \ldots, \alpha_p x_n) \tag{6.3}$$

and

$$\alpha_i \sum_{x \leqq \alpha_j x_1} f(x, \alpha_j x_2, \ldots, \alpha_j x_n) = \alpha_i \sum_{x \leqq \alpha_j x_1} f(\alpha_p x, \alpha_p x_2, \ldots, \alpha_p x_n). \tag{6.4}$$

Thus if $x_1 \leqq j$, so that $\alpha_j x_1 = x_1$, the sums (6.3) and (6.4) are equal. If $x_1 > j$, writing f_t for $f(t, \alpha_p x_2, \ldots, \alpha_p x_n)$ sum (6.3) equals

$$\alpha_i (f_0 + \cdots + f_{p-1} + f_p + (x_1 - p) f_p) \tag{6.5}$$

and sum (6.4) equals

$$\alpha_i (f_0 + \cdots + f_{p-1} + f_p + (j - p) f_p). \tag{6.6}$$

Since $x_1 - p > i$ and $j - p \geqq i$ it follows by (4.8) that (6.5) and (6.6) are equal which completes the proof of (6.2).

Under the same condition (6.1) if

$$P(x_1, x_2, \ldots, x_n) = \prod_{x \leqq x_1} f(x, x_2, \ldots, x_n)$$

then

$$\alpha_i P(x_1, x_2, \ldots, x_n) = \alpha_i P(\alpha_j x_1, \ldots, \alpha_j x_n). \tag{6.7}$$

The proof of (6.7) is similar to that of (6.2) using (4.81) in place of (4.8).

7. If $b \leqq i$ and if there is a $p_i \geqq i$ such that

$$\alpha_i f(x_1, x_2, \ldots, x_n) = \alpha_i f(\alpha_{p_i} x_1, \ldots, \alpha_{p_i} x_n) \tag{7.1}$$

and if

$$g(x_1, x_2, \ldots, x_n) = f(x_1, x_2, \ldots, x_n) \dot{-} b$$

then

$$\alpha_i g(x_1, \ldots, x_n) = \alpha_i g(\alpha_{p_{2i}} x_1, \alpha_{p_{2i}} x_2, \ldots, \alpha_{p_{2i}} x_n). \tag{7.2}$$

For

$$\begin{aligned} \alpha_i g &= \alpha_i (f \dot{-} b) = \alpha_i (\alpha_{2i} f \dot{-} b) \quad \text{by (4.7)} \\ &= \alpha_i \{f(\alpha_{p_{2i}} x_1, \ldots, \alpha_{p_{2i}} x_n) \dot{-} b\} \quad \text{by (7.1)} \\ &= \alpha_i g(\alpha_{p_{2i}} x_1, \ldots, \alpha_{p_{2i}} x_n). \end{aligned}$$

8. We come now to the result which provides the decision procedure.

Theorem 1. *If $f \in C_k$ and if f is of height λ then for $i \geqq k$, and $q = 2^\lambda i$*

$$\alpha_i f(x_1, x_2, \ldots, x_n) = \alpha_i f(\alpha_q x_1, \alpha_q x_2, \ldots, \alpha_q x_n). \tag{8.1}$$

We consider first the case $\lambda = 0$. If f is of height 0 and spread 0 then (8.1) holds by (4.1). If (8.1) holds for functions of height 0 and spread $\leqq \mu$ then it holds for functions of height 0 and spread $\mu + 1$, by (4.2), (4.3), (4.4), (4.5), (4.6). Thus (8.1)

holds for all f of height 0. If (8.1) holds for functions of height $\leqq \lambda$, and if f is of height $\lambda + 1$ and spread 0 then there is a $g \in C_k$ of height λ such that

$$f = \Sigma g \quad \text{or} \quad f = \Pi g \quad \text{or} \quad f = g \dot{-} b;$$

if

$$f(x_1, x_2, \ldots, x_n) = \sum_{x \leqq x_1} g(x, x_2, x_3, \ldots, x_n)$$

and g satisfies (8.1) with $q = 2^\lambda i$, then by (6.2)

$$\alpha_i f = \alpha_i f(\alpha_j x_1, \alpha_j x_2, \ldots, \alpha_j x_n)$$

with $j \geqq 2^\lambda i + i$, and since $2^{\lambda+1} i \geqq (2^\lambda + 1)\, i$ it follows that f satisfies (8.1) with $q = 2^{\lambda+1} i$.

Similarly if

$$f(x_1, x_2, \ldots, x_n) = \prod_{x \leqq x_1} g(x, x_2, \ldots, x_n)$$

then f satisfies (8.1) by (6.7).

Finally if

$$f(x_1, x_2, \ldots, x_n) = g(x_1, x_2, \ldots, x_n) \dot{-} b$$

then by (7.2)

$$\alpha_i f(x_1, x_2, \ldots, x_n) = \alpha_i f(\alpha_{2q} x_1, \alpha_{2q} x_2, \ldots, \alpha_{2q} x_n)$$

and since $2q = 2^{\lambda+1} i$ this proves that f satisfies (8.1). Thus if Theorem 1 holds for functions of height $\leqq \lambda$, it holds for functions of height $\lambda + 1$ and spread 0. Suppose now the theorem holds for functions of height $\leqq \lambda$ and any spread, and for functions of height $\lambda + 1$ and spread $\leqq \mu$. Let f be of height $\lambda + 1$ and spread $\mu + 1$; then either (1) there is a g of height $\lambda + 1$ and spread $\leqq \mu$, and an h of height $\leqq \lambda + 1$ and spread $\leqq \mu$ (the spread being μ in at least one of the two cases) such that f is one of

$$a \dot{-} g,\ g + h,\ gh,\ g + (h \dot{-} g),\ g \dot{-} (g \dot{-} h)$$

or (2) there is a φ of height λ and spread $\mu + 1$ such that

$$f = \Sigma \varphi \quad \text{or} \quad f = \Pi \varphi \quad \text{or} \quad f = \varphi \dot{-} b.$$

In virtue of the hypothesis that all functions of height $\leqq \lambda$ satisfy the theorem, it follows exactly as in the proof above for functions of spread 0 that f satisfies the theorem in case (2). In case (1) it follows by (4.6), (4.2), (4.3), (4.4), (4.5), and (5.1) that (8.1) holds also for f. By induction over μ it now follows that if Theorem 1 holds for some λ it holds for $\lambda + 1$, and since Theorem 1 has been proved for $\lambda = 0$, this completes the proof of the theorem by induction over the height.

To prove Theorem 2 we observe that if $f \in C$ then $f \in C_k$ for any k at least as great as any of the constants a, b which occur in functions $a \dot{-} x$, $x \dot{-} b$ used in the formation of f. The height λ of f does not exceed the number of times the operations Σ, Π and $x \dot{-} b$ are used in the construction of f.

Then, by Theorem 1, if $q = 2^\lambda k$ then

$$\alpha_k f(x_1, \ldots, x_n) = \alpha_k f(\alpha_q x_1, \alpha_q x_2, \ldots, \alpha_q x_n).$$

Taking $k > c$ it follows that

$$f(x_1, x_2, \ldots, x_n) = c$$

identically if and only if

$$f(x_1, x_2, \ldots, x_n) = c$$

for all values of $x_1, x_2, \ldots, x_n$ in the *finite* set $(0, 1, 2, \ldots, q)$.

9. Since all the properties (4) down to (7.2) of the function $\alpha_i(x)$ are provable in a suitable formalisation $\mathfrak{R}$ of recursive arithmetic (such as the equation calculus) it follows that (8.1) is provable in $\mathfrak{R}$ for each particular function $f \in C$, and therefore

Theorem 3. *If $f \in C$ and if f is of height λ and k exceeds c and all constants built into f then the equation*

$$f(x_1, x_2, \ldots, x_n) = c$$

is provable in $\mathfrak{R}$ if each instance of the equation holds when $x_1, x_2, \ldots, x_n$ take all possible values in the finite set $(0, 1, \ldots, 2^\lambda k)$.

For, if $q = 2^\lambda k$,

$$\alpha_k f(x_1, \ldots, x_n) = \alpha_k f(\alpha_q x_1, \ldots, \alpha_q x_n)$$

is provable in $\mathfrak{R}$ and by hypothesis the right hand side has the value $c < k$ for all substitutions of values $0, 1, 2, \ldots, q, x_i + q$ for x_i, $1 \leqq i \leqq n$, and so all

$$f(\xi_1, \xi_2, \ldots, \xi_n) = c$$

where ξ_i is one of $0, 1, 2, \ldots, q-1, x_i + q$ for all i, $1 \leqq i \leqq n$, whence in turn we readily prove in $\mathfrak{R}$ all the equations

$$f(x_1, \xi_2, \ldots, \xi_n) = c,\ f(x_1, x_2, \xi_3, \ldots, \xi_n) = c, \ldots f(x_1, x_2, \ldots, x_{n-1}, \xi_n) = c$$

and finally $f(x_1, x_2, \ldots, x_n) = c$.

Reference

R. L. Goodstein, A decidable fragment of recursive arithmetic. This Zeitschr. **9** (1963), 199—201.

(Eingegangen am 12. April 1965)

Notre Dame Journal of Formal Logic
Volume VI, Number 3, July 1965

THE SUBSTITUTION SCHEMA IN RECURSIVE ARITHMETIC

R. D. LEE

In his paper *Logic Free Formalisations of Recursive Arithmetic* [1] R. L. Goodstein presents a formalisation of primitive recursive arithmetic in which the only axioms are explicit and recursive function definitions, and the rules of inference are the schemata

$$(\mathbf{Sb}_1) \qquad \frac{F(x) = G(x)}{F(A) = G(A)}$$

$$(\mathbf{Sb}_2) \qquad \frac{A = B}{F(A) = F(B)}$$

$$(\mathbf{T}) \qquad A = B$$

$$\frac{A = C}{B = C}$$

where $F(x)$, $G(x)$ are recursive functions and A,B,C are recursive terms, and the primitive recursive uniqueness rule

$$(\mathbf{U}) \qquad \frac{F(Sx) = H(x,F(x))}{F(x) = H^x F(0)}$$

where the iterative function $H^x t$ is defined by the primitive recursion $H^0 t = t$, $H^{Sx} t = H(x,H^x t)$; in **U**, F may contain additional parameters.

In the same paper it is shown that the schema **U** may be replaced by

$$(\mathbf{E}) \qquad \frac{F(0) = 0 \quad F(Sx) = F(x)}{F(x) = 0}$$

if we take as axioms

$$(A) \qquad a + (b \dot{-} a) = b + (a \dot{-} b)$$

and, in place of the introductory equations for the predecessor function,

$$(P) \qquad Sa \dot{-} Sb = a \dot{-} b$$

This system is referred to as $\mathbf{R}_1$.

Received November 6, 1964

The purpose of this paper is to present another formalisation, **R***, which also weakens **U** and yet avoids taking A as an axiom.

The rules of inference of **R*** are $\mathbf{Sb}_1$, $\mathbf{Sb}_2$, **T** and

$$(\mathbf{E}_1) \qquad \frac{F(Sx) = F(x)}{F(x) = F(0)}$$

$$(\mathbf{E}_3)^{\dagger} \qquad \frac{F(0) = G(0)}{F(x) = G(x)}$$

In place of the recursive definitions of addition we have the axioms

(A_1) $a + 0 = a$ $\qquad$ (A_2) $a + (b + c) = (a + b) + c.$

and for subtraction, we have the recursive definitions of predecessor and difference

(S_1) $0 \dot{-} 1 = 0$; (S_2) $Sa \dot{-} 1 = a$; (S_3) $a \dot{-} 0 = a$; (S_4) $a \dot{-} Sb = (a \dot{-} b) \dot{-} 1$;

and the axiom

(S_5) $(a \dot{-} b) \dot{-} 1 = (a \dot{-} 1) \dot{-} b$

We have also the recursive definition of multiplication

(M_1) $a \cdot 0 = 0$ $\qquad$ (M_2) $a \cdot Sb = a \cdot b + a.$

Exactly as in [1] we may prove the following results

$$(\mathbf{K}) \qquad \frac{A = B}{B = A}$$

and $Sa \dot{-} Sb = a \dot{-} b$, $a \dot{-} a = 0$, $0 \dot{-} a = 0$, $(a + b) \dot{-} b = a$, $(a + n) \dot{-} (b + n) = a \dot{-} b$, $n \dot{-} (b + n) = 0$.

We now derive the schema,

$$(\mathbf{U}_2) \qquad \frac{F(Sx) = SF(x)}{F(x) = F(0) + x}$$

(I am indebted to R. L. Goodstein for the following proof). Write $G(x) = F(0) + x$, then $G(Sx) = SG(x)$ and $G(0) = F(0)$. Using these two results and $F(Sx) = SF(x)$ we deduce $F(x) = G(x)$ for if $L(x) = F(x \dot{-} 1) + \{1 \dot{-} (1 \dot{-} x)\}$, then $L(0) = F(0)$ and $L(Sx) = SF(x) = F(Sx)$ so that, by $\mathbf{E}_6$, $L(x) = F(x)$. Therefore

$$F(x) = F(x \dot{-} 1) + \{1 \dot{-} (1 \dot{-} x)\}$$

Let $\phi(n,x)$ be defined by

$$\phi(0, x) = 0 \qquad \phi(Sn,x) = \{1 \dot{-} (1 \dot{-} (x \dot{-} n))\} + \phi(n,x)$$

then

$$\begin{aligned} F(x \dot{-} n) + \phi(n,x) &= \{F(x \dot{-} Sn) + [1 \dot{-} (1 \dot{-} (x \dot{-} n))]\} + \phi(n,x) \\ &= F(x \dot{-} Sn) + \phi(Sn,x) \end{aligned}$$

---

†Retaining the notation of [1].

Using $\mathbf{E}_1$

$$F(x \dot{-} n) + \phi(n, x) = F(x \dot{-} 0) + \phi(0, x) = F(x)$$

Whence taking $n = x$

$$F(0) + \phi(x,x) = F(x)$$

Similarly

$$G(0) + \phi(x,x) = G(x)$$

Hence

$$F(x) = G(x)$$
$$F(x) = F(0) + x.$$

We now use $\mathbf{U}_2$ to prove

$$0 + a = a.$$

Write $F(a) = a$, then $F(Sa) = SF(a)$. Hence using $\mathbf{U}_2$ and **K**, $0 + a = a$. Similarly using $\mathbf{U}_2$ we may prove $a + Sb = Sa + b$, $a + b = b + a$, $(a + b) \dot{-} a = b$ and exactly as in [1]

$$a + (b \dot{-} a) = b + (a \dot{-} b).$$

Now from $\mathbf{E}_1$, **E** follows immediately and hence we have postulated or derived all the axioms and rules of inference of system $\mathbf{R}_1$, given in [1]. Hence the sufficiency of **R*** for the construction of primitive recursive arithmetic follows from the sufficiency of $\mathbf{R}_1$, which is proved in [1]. In fact we can reduce the axiom system **R*** by postulating only certain *special* cases of $\mathbf{Sb}_2$. The special cases are

$$(\mathbf{Sb}_{21})\ \frac{A = B}{x+A = x+B} \qquad (\mathbf{Sb}_{22})\ \frac{A = B}{A \dot{-} x = B \dot{-} x} \qquad (\mathbf{Sb}_{23})\ \frac{A = B}{x \dot{-} A = x \dot{-} B}$$

$$(\mathbf{Sb}_{24})\ \frac{A = B}{F(A) = F(B)}$$

where in $\mathbf{Sb}_{24}$ $A = B$ is restricted to one of the initial equations $A_1, A_2, S_1, S_2, S_3, S_4, S_5, M_1, M_2$, or is any recursive or explicit function definition. For $F(0 \dot{-} Sx))$ $F((0 \dot{-} x) \dot{-} 1) = F((0 \dot{-} 1) \dot{-} x) = F(0 \dot{-} x)$ using $\mathbf{Sb}_{24}$ for the equations $a \dot{-} Sb = (a \dot{-} b) \dot{-} 1$, $(a \dot{-} b) \dot{-} 1 = (a \dot{-} 1) \dot{-} b$, $0 \dot{-} 1 = 0$, and **Sb**, to substitute 0 for x and x for b. Now writing $G(x) = F(0 \dot{-} x)$, we have proved $G(Sx) = G(x)$ and hence from $\mathbf{U}_1$, $G(x) = G(0)$
therefore

0.1 $$F(0 \dot{-} x) = F(0)$$

Similarly $F(Sx \dot{-} Sx) = F((Sx \dot{-} x) \dot{-} 1) = F((Sx \dot{-} 1) \dot{-} x) = F(x \dot{-} x)$ and hence by $\mathbf{U}_1$

0.2 $$F(x \dot{-} x) = F(0).$$

The proofs of the results in the first part of this paper up to and including the proof of $a + (b \dot{-} a) = b + (a \dot{-} b)$ use only the above special cases of $\mathbf{Sb}_2$ and 0.1 and 0.2.

Using $a + b = b + a$ and $\mathbf{Sb}_{21}$ we have

$$(\mathbf{Sb}_{25}) \qquad \frac{A = B}{A+x = B+x}$$

Following the proof, as given in [1], of the sufficiency of $\mathbf{R}_1$ (and therefore of $\mathbf{R}^*$, since in $\mathbf{R}^*$ we have derived or postulated all the axioms and rules of $\mathbf{R}_1$), we may derive the schema

$$(\mathbf{A}) \qquad \frac{A \dot{-} B = 0,\ B \dot{-} A = 0}{A = B}$$

The schema

$$(\mathbf{Sb}_{26}) \qquad \frac{A = B}{Ax = Bx}$$

is now proved as follows

Using $\mathbf{Sb}_{24}$, $A.Sx \dot{-} B.Sx = A.Sx \dot{-} B.x + B = A.x + A \dot{-} B.x + B$. Assuming $A = B$, from $\mathbf{Sb}_{21}$, $z + A = z + B$, and hence from $\mathbf{Sb}_1$, $B.x + A = B.x + B$. Therefore using $\mathbf{Sb}_{23}$, $z \dot{-} (B.x + B) = z \dot{-} (B.x + A)$ and hence from $\mathbf{Sb}_1$ $(A.x + A) \dot{-} (B.x + B) = (A.x + A) \dot{-} (B.x + A)$; but from a previous result $(A.x + A) \dot{-} (B.x + A) = A.x \dot{-} B.x$
Hence

$$A.Sx \dot{-} B.Sx = A.x \dot{-} B.x$$

Using $\mathbf{E}_1$,

$$A.x \dot{-} B.x = 0.$$

Similarly

$$B.x \dot{-} A.x = 0.$$

Hence, by **A**.

$$A.x = B.x.$$

Exactly as in [1], we may now prove $Sa.b = a.b + b$, $0.a = 0$ and $a.b = b.a$. The schema

$$(\mathbf{Sb}_{27}) \qquad \frac{A = B}{xA = xB}$$

follows from $a.b = b.a$ and $\mathbf{Sb}_{26}$.

Apart from the special cases of $\mathbf{Sb}_2$ which are axioms or have been derived the only application of $\mathbf{Sb}_2$ in the proof of the sufficiency of $\mathbf{R}^*$ occurs in the proof of the substitution theorem, in the form

$$\frac{x + (y \dot{-} x) = y + (x \dot{-} y)}{F(x + (y \dot{-} x)) = F(y + (x \dot{-} y))}$$

I shall give an alternative proof of the substitution theorem which avoids use of this result.

THE SUBSTITUTION THEOREM

$$x = y \rightarrow F(x) = F(y)$$

All primitive recursive functions can be obtained by substitution and recursion according to the schema $F(0) = 0\ F(Sx) = H(F(x))$, from the initial functions $u + v$, $u \dot{-} v$, $Rt(u)$, where $Rt(0) = 0$, $Rt(Sx) = Rt(x) + [1 \dot{-} p(x, Rt(x))]$ and $p(x,y) = (Sy)^2 \dot{-} Sx$.

It suffices therefore to prove that the substitution theorem holds for these initial functions and is preserved under substitution and the given recursion. From the original proof of the substitution theorem given in [1], we have

$$(1 \dot{-} |x,y|)F(x + (y \dot{-} x)) = (1 \dot{-} |x,y|)F(x)$$
$$(1 \dot{-} |x,y|)F(y + (x \dot{-} y)) = (1 \dot{-} |x,y|)F(y)$$

In the case of $F(z) = z + a$ we have

$$(1 \dot{-} |x,y|)((x + (y \dot{-} x)) + a) = (1 \dot{-} |x,y|)(x + a)$$
$$(1 \dot{-} |x,y|)((y + (x \dot{-} y)) + a) = (1 \dot{-} |x,y|)(y + a)$$

But from $\mathbf{Sb}_{25}$ and $x + (y \dot{-} x) = y + (x \dot{-} y)$, $[x + (y \dot{-} x)] + a = [y + (x \dot{-} y)] + a$ and hence from $\mathbf{Sb}_{27}$

$$(1 \dot{-} |x,y|)[(x + (y \dot{-} x)) + a] = (1 \dot{-} |x,y|)[(y + (x \dot{-} y)) + a]$$

Hence

$$(1 \dot{-} |x,y|)(x + a) = (1 \dot{-} |x,y|)(y + a)$$

Thus we have derived the substitution for the function $F(z) = z + a$.

In the way, using $\mathbf{Sb}_{21}$, $\mathbf{Sb}_{22}$, $\mathbf{Sb}_{23}$, $\mathbf{Sb}_{25}$, $\mathbf{Sb}_{27}$, we may obtain the substitution theorem for the initial functions $u + v$, $u \dot{-} v$, $u.v$.

In the following proof of the substitution theorem for the function $Rt(x)$, I shall use theorems of the proportional calculus, which may easily be proved by deriving their corresponding equations in recursive arithmetic. The theorems concerned are

(1) $(x = x') \rightarrow (Sx = Sx')$

(2) $(y = y') \rightarrow (Sy)^2 = (Sy')^2$

(3) $((x = x') \,\&\, (y = y')) \rightarrow (Sy)^2 \dot{-} Sx = (Sy')^2 \dot{-} Sx'$

(4) $(x=x') \,\&\, (Rt(x) = Rt(x')) \rightarrow p(x,Rt(x)) = p(x',Rt(x'))$

(5) $(x=x') \,\&\, (Rt(x) = Rt(x')) \rightarrow Rt(x) + (1 \dot{-} p(x,Rt(x))) = Rt(x') + (1 \dot{-} p(x',Rt(x'))$

(6) $(x=x') \,\&\, (Rt(x) = Rt(x')) \rightarrow Rt(Sx) = Rt(Sx')$

We now prove

$$((x = x') \rightarrow Rt(x) = Rt(x')) \rightarrow (Sx = Sx' \rightarrow Rt(Sx) = Rt(Sx'))$$

with a,b,c standing for $|x,x'|$ (and hence for $|Sx,Sx'|$, $|Rt(x), Rt(x')|$, $|Rt(Sx), Rt(x')|$ respectively), we require to prove

(7) $(1 \dot{-} (1 \dot{-} a)b)(1 \dot{-} a)c = 0$.

From (6)

$$(1 \dot{-} (a + b))\, c = 0.$$

so that

$$(1 \dot{-} (a + b))\, (1 \dot{-} a)\, c = 0.$$

Hence

$$(1 \dot{-} a)c \dot{-} b\,(1 \dot{-} a)c = 0$$

because $a(1 \dot{-} a) = 0$. Therefore

(8) $$(1 \dot{-} a)c\,(1 \dot{-} b) = 0.$$

Hence

(9) $$\begin{aligned}(1 \dot{-} (1 \dot{-} a)b)\,(1 \dot{-} a)c &= (1 \dot{-} a)c \dot{-} (1 \dot{-} a)\,(1 \dot{-} a)bc \\ &= (1 \dot{-} a)c \dot{-} (1 \dot{-} a)bc \\ &= (1 \dot{-} a)c\,(1 \dot{-} b) \\ &= 0\end{aligned}$$

from (8). Therefore

(10) $$(x = x' \rightarrow Rt(x) = Rt(x')) \rightarrow (Sx = Sx' \rightarrow Rt(Sx) = Rt(Sx'))$$

Now define $P(x,x') = (1 \dot{-} |x,x'|)\,\big|\, Rt(x),\, Rt(x')\big|$
Then, from (9),

$$P(x,x') = 0 \rightarrow P(Sx,\, Sx') = 0.$$

But, from E_3,

$$P(x,0) = (1 \dot{-} x)\,|Rt(x),\, Rt(0)| = 0.$$

Similarly

$$P(0,x') = 0.$$

Hence, by I_2,

$$\begin{aligned}&P(x,x') = 0. \\ &(x = x') \rightarrow \{Rt(x) = Rtx'\}\end{aligned}$$

We have now proved the substitution theorem for all the initial functions.

Now suppose the substitution theorem holds for the particular functions f,g, i.e.

(11) $$x = y \rightarrow f(x) = f(y)$$

and

(12) $$x = y \rightarrow g(x) = g(y).$$

From Sb_1 and (11) we have

(13) $$g(x) = g(y) \rightarrow f(g(x)) = f(g(y)).$$

We now use the schema

(14) $\dfrac{\begin{array}{l} p \to q \\ q \to r \end{array}}{p \to r}$

which may be proved by a consideration of the corresponding equations in recursive arithmetic.

Hence from (12), (13),

$$x = y \to f(g(x)) = f(g(y))$$

i.e. the substitution theorem is preserved under composition.

Now consider $\phi(x)$ defined by the recursion $\phi(0) = 0$, $\phi(Sx) = H(\phi(x))$ and suppose the substitution theorem holds for H.

Define $P(x,y) = (1 \dot{-} |x,y|)|\phi(x), \phi(y)|$. Then, using $\mathbf{E}_3$

(15) $P(x,0) = (1 \dot{-} x)|\phi(x), \phi(0)| = 0$

and

(16) $P(0,Sy) = 0.$

We now derive the result

$$(a = a') \to \{(b = b') \to (a = b \to a' = b')\}.$$

As we observed above

$$(1 \dot{-} |x,y|)F(x + (y \dot{-} x)) = (1 \dot{-} |x,y|)F(x)$$

and so with $F(x) = |x,t|$

$$(1 \dot{-} |x,y|)|x + (y \dot{-} x),t| = (1 \dot{-} |x,y|)|x,t|.$$

Similarly

$$(1 \dot{-} |x,y|)|y + (x \dot{-} y), t| = (1 \dot{-} |x,y|)|y,t|$$

Using $x + (y \dot{-} x) = y + (x \dot{-} y)$, and the given special cases of $\mathbf{Sb}_2$ we obtain

$$(1 \dot{-} |x,y|)|x + (y \dot{-} x), t| = (1 \dot{-} |x,y|)|y + (x \dot{-} y),t|.$$

Hence

(17) $(1 \dot{-} |x,y|)|x,t| = (1 \dot{-} |x,y|)|y,t|$

Now using (17) and rearranging factors

$$\begin{aligned}(1 \dot{-} |a,a'|)(1 \dot{-} |b,b'|)(1 \dot{-} |a,b|)|a',b'| &= (1 \dot{-} |b,b'|)(1 \dot{-} |a,a'|) \\ &\quad (1 \dot{-} |a,b|)|a,b| \\ &= 0.\end{aligned}$$

Hence

(18) $a = a' \to \{b = b' \to (a = b \to a' = b')\}.$

Replacing a,a',b,b' by $H(\phi(x)), \phi(Sx), H(\phi(y)), \phi(Sy)$ respectively

$$H(\phi(x)) = \phi(Sx) \to \{H(\phi(y)) = \phi(Sy) \to (H(\phi(x)) = H(\phi(y)) \to \phi(Sx) = \phi(Sy))\}$$

From the definition of ϕ, using modus ponens twice

$$H(\phi(x)) = H(\phi(y)) \rightarrow \phi(Sx) = \phi(Sy)$$

Using the substitution theorem for H,

$$\phi(x) = \phi(y) \rightarrow H(\phi(x)) = H(\phi(y))$$

and hence by schema (14)

(19) $\quad \phi(x) = \phi(y) \rightarrow \phi(Sx) = \phi(Sy).$

We now prove

(20) $\quad P(x,y) = 0 \rightarrow P(Sx,Sy) = 0.$

With a, b, c standing for $|x,y|$, $|\phi(x), \phi(y)|$, $|\phi(Sx), \phi(Sy)|$ respectively there is represented by the equation

(21) $\quad (1 \dot{-} (1 \dot{-} a)b)\,(1 \dot{-} a)c = 0$

With $f(a)$ standing for the left hand side, $f(Sa) = 0$ and $f(0) = (1 \dot{-} b)c = 0$ from (19) and hence, using E_3 $f(a) = 0$.

Now using I_2 with conditions satisfied by (15), (16), (20), we obtain

$$x = y \rightarrow \phi(x) = \phi(y)$$

Hence the substitution theorem is preserved under the given recursion and thus it holds for all recursive functions.

My thanks are due to Professor R. L. Goodstein for help and encouragement in the preparation of this paper.

REFERENCE

[1] R. L. Goodstein, Logic-free formalisation of recursive arithmetic. *Math. Scand*, 2(1954). 247-261.

University of Leicester
Leicester, England