

POLSKA AKADEMIA NAUK, INSTYTUT MATEMATYCZNY

DISSERTATIONES
MATHEMATICAE
(ROZPRAWY MATEMATYCZNE)

KOMITET REDAKCYJNY

KAROL BORSUK redaktor

ANDRZEJ BIAŁYNICKI-BIRUŁA, BOGDAN BOJARSKI,
ZBIGNIEW CIESIELSKI, JERZY ŁOŚ, ANDRZEJ MOSTOWSKI,
ZBIGNIEW SEMADENI, MARCELI STARK, WANDA SZMIELEW

CVIII

DON JENSEN

On local proof restrictions for strong theories

WARSZAWA 1973

PAŃSTWOWE WYDAWNICTWO NAUKOWE

S. 7133



PRINTED IN POLAND

W R O C Ł A W S K A D R U K A R N I A N A U K O W A

161 192.

CONTENTS

Introduction	5
Chapter 1. Preliminaries	7
Chapter 2. Formal arithmetization	12
Chapter 3. Proof restriction functions	18
Chapter 4. Reflexive theories and proof restriction functions	24
Chapter 5. Truth definitions and formalized consistency proofs	30
Chapter 6. Pair proof restriction functions	37
Conclusions	45
Bibliography	46

INTRODUCTION*

Consider classical axiomatic theories, such as Peano arithmetic or Zermelo–Fraenkel set theory, which naturally formalize domains of mathematical reasoning. Frequently, these theories require very complex axiom sets (e.g. infinite, etc.). We consider whether it is possible to obtain a local improvement on this global axiomatization problem for such a theory T : A mapping $a \mapsto F_a$ from sentences a of T to recursively enumerable subtheories of T will be called a *proof restriction function* (for T) provided, for each a , $F_a \vdash a$ in case $T \vdash a$. Furthermore, we should impose appropriate demands on the correspondence (and the F_a) to insure that it is a genuine (and informative) improvement on the global axiom problem for T . For example: at least some of the F_a should be proper subtheories of T ; and producing each F_a should not depend upon *finding* a proof of a .

Chapters 1 and 2 review standard results. Some readers will omit these except for reference concerning our use of notation.

A first observation is that a mapping $a \mapsto F_a$ should be realized by a function f where each $f^{\ulcorner a \urcorner}$ is a *natural description* of the subtheory F_a . (Asking for a *natural* description avoids the intensional problem that the same axiom set will have many non-equivalent descriptions.)

In Chapters 3 and 4 we consider the simplest situation where we ask *each* F_a to be more restricted than T in some *explicitly* described way. For example, in Chapter 3 we consider the situation where we ask each F_a to be explicitly finitely axiomatized while T cannot be. For this to be significant, we note that T should be undecidable, f should be recursive, and have a domain which includes undecided sentences. Under these conditions we also note simple examples of positive and negative solutions concerning the possibility of proof restriction functions. The positive solution realizes our intuition that simple sentences should require only simple axioms in their proofs (if they can be proved).

* Many of the results reported here were obtained while the author was a student of Professor Andrzej Ehrenfeucht at the University of Southern California; these appear in the author's thesis [8].

The preparation of this paper was supported under grants A-5267 and A-5549, National Research Council, Canada.

However, our main theorem (4.4) of Chapter 4 prevents this intuition from being realized in stronger theories. Roughly stated, if T interprets Peano arithmetic then there is no recursive proof restriction function f for T such that $T \vdash \text{Consis}(F_\alpha)$ for each α ⁽¹⁾. (This extends both Gödel's second theorem (on consistency proofs) and also the global axiomatization solution for many theories.) We note some applications and illustrations of this theorem. In many applications (e.g. Peano arithmetic, Zermelo-Fraenkel set theory), it prevents the F_α from being finite, of bounded-depth, etc.

In Chapter 5 we recall classical results concerning truth definitions and formalized consistency proofs. These are essential for the applications of Chapters 4 and 6.

Instead of asking *each* F_α to be more restricted than T , a less demanding possibility for an informative proof restriction function is based on the following intuition: in attempted proofs of a and $\neg a$ we would tend to use different axioms. In Chapter 6 we consider whether we can ask that F_α and $F_{\neg\alpha}$ be *explicitly distinct* for each α . Examples suggest how this should be formalized: we should ask for one sentence δ_α which is provable in exactly one of F_α and $F_{\neg\alpha}$. Our main theorem (6.6) says that this is not possible where f is recursive, the δ_α are given by a recursive function d , and T has a truth definition for all the δ_α . Examples show the necessity of each of these hypotheses. Theorem 6.6 generalizes 4.4 in several directions and has numerous applications of which we note only a few. We conclude that it is difficult to imagine informative proof restriction functions which would not be subject to Theorem 6.6.

⁽¹⁾ The author has noticed that a result similar to our Theorem 4.4 was also reported in [10]. Also, another technique of that paper was useful in one of our applications (6.7).

Chapter I

PRELIMINARIES

In this chapter and the next we summarize classical notions and results basic to our subsequent discussion. These chapters concentrate on logical syntax and its formal arithmetization. We take Shoenfield's text [18] as a standard reference, but use other conventions when convenient.

A. First-order theories. The formal languages and theories which we discuss are classical, finitary, first-order with equality. Furthermore, in our applications, the languages will be countable, and the theories will have recursively enumerable theorem sets. The logical symbols are \neg , \vee , \exists , $=$, and the individual variables v_0, v_1, v_2, \dots (denumerably many). Besides these, each particular *language* L will be determined by a fixed (usually finite) set of non-logical *constant* symbols: predicate symbol(s), and function symbol(s), each with a fixed number n ($n \geq 0$) of argument places. For each language L , we form *expressions*, *terms*, and *formulas* as in [18], pp. 14–18. For each L to obtain the *logical theory* for L we specify *logical axioms* and *logical rules of inference*, again as in [18], p. 21. We denote the logical theory (for any L) by Log . Besides these, a particular *theory* T with language L will be determined by its set of axioms. From these, the *proofs* and *theorems* of T are defined inductively as usual. (Because we are often concerned with particular axiom sets, we find it convenient to distinguish theories by their axiom sets and not only their theorem sets. Of course, some of our results will apply for the more liberal identification of theories.)

B. Abbreviating conventions. The discussion of a formal language is facilitated by the use of abbreviations. We will follow the usual conventions (e.g. [18], pp. 14–18) for the use of parentheses, $\&$, \rightarrow , \leftrightarrow , \neq , and \forall as abbreviations. Small greek letters (α, β, \dots) will be used as syntactical variables for formulas (also terms, etc., when indicated). Similarly, we use the letters x, y, z, \dots to denote arbitrary fixed individual variables (except where noted). A convenient usage is the introduction of informal *abbreviating operators* which, when applied to a particular object (inside or outside the language), denote a particular understood expression of the formal language.

Thus we use “[]” as an abbreviating operator for substitution: if a is a formula and τ is a term, both of language L , then $a[\tau]$ is the formula obtained by replacing in a all *free occurrences* of an understood variable x by τ (changing *bound variables* if necessary). The particular variable x being replaced will usually be clear from the context, especially if we have previously written $a[x]$.

We also use a “vector” notation quite liberally as an abbreviating operator for finite sequences of variables, terms, etc. Thus $a[\vec{\tau}_i]$ indicates simultaneous substitution. What is intended by other usages should be clear from the context. We make some use of $\bigwedge_i (a_i)$ to indicate finite indexed conjunction.

C. Structures and models. We make little use of model theoretic notions. By a *structure for L* we mean a relational set $\mathcal{A} = \langle A, \dots R_i \dots, \dots F_j \dots \rangle$ in the usual sense of [18], pp. 18–20. For a a formula of L we take the usual definition of \mathcal{A} *satisfies* a (or a “is valid in” \mathcal{A}). If theory T has language L , we say that a structure \mathcal{A} (for L) is a *model for T* in case \mathcal{A} satisfies each of the axioms of T . The *theory of a structure \mathcal{A}* is the set of formulas of L satisfied by \mathcal{A} .

D. Recursive functions and arithmetization of syntax. The notions we use from recursion theory are standard (e.g. *recursive, recursively enumerable, Π_1^0 , partial function*, etc.). One of the main facts we utilize is that there are recursive functions which code (as a single number) sequences of natural numbers of arbitrary finite length; and also unambiguous decoding functions.

We generally present formal languages and theories in the format of expressions in a familiar alphabet of symbols. However we could just as well regard the symbols as fixed natural numbers (our languages are countable). And noting the recursive coding functions we may then regard an expression as a single natural number. For a fixed language L we will assume that this is done as in [18], pp. 122–126. If a is an expression of L , then $\ulcorner a \urcorner$ will denote the *code number* of a and φ_n will denote the expression (if any) with code number n . Both formal symbols and effective abbreviating operators may consequently be coded as recursive functions (e.g. a function *neg* so that always $\text{neg} \ulcorner a \urcorner = \ulcorner \neg a \urcorner$). The correspondence between the language and the codes is, of course, in the metatheory. For the definitions of the next section we need only that it is effective. In the next chapter we will develop this somewhat further.

E. Standard notions for theories. We write $T \vdash a$ for “ a is a theorem of T ”. $S \cup T$ will denote the theory whose axioms are the union of those of S and those of T (language union also). If both the language and the theorems of theory T' include those of theory T then T' *extends* T and T

is a *subtheory* of T' . An extension is *conservative* if it does not add new theorems in the smaller language. We say that a theory T is *recursively enumerable* if the set of code numbers of its theorems is recursively enumerable. A theory T is (can be)... axiomatized if the set of code number of its axioms (of a set of theorems which imply all the others) is... (e.g. finite, recursive, etc.). Noting Craig [2], a recursively enumerable theory can always be given a recursive set of axioms. We will frequently adapt this device. As usual, a theory T is *consistent* provided there are formulas of its language which are not theorems of T (see also Chapter 1-J). By the Completeness Theorem, T is consistent iff T has a model \mathcal{A} . T is *decidable* if the set of code numbers of its theorems is recursive. T is *essentially undecidable* provided every consistent extension of T is undecidable (see also Chapter 2-G). Note that the theory of a finite structure is always decidable.

F. Definitional extensions. We follow [18], pp. 57-61. Let a be a formula (all of whose free variables are in \vec{x}) of the language L of T , and suppose p is a (predicate) symbol which is not in L . Then we may give p the *defining axiom* $p\vec{x} \leftrightarrow a[\vec{w}]$. Here a is called the *definiens*. (Function symbols are treated similarly, except that they require *defining conditions*: $T \vdash \exists y (a[\vec{x}, y] \& \forall w (a[\vec{x}, w] \rightarrow w = y))$.) Thus we extend both L and T . If each new symbol of T' is obtained by a given finite sequence of extensions of this sort starting with T , then T' is called a *definitional extension* of T . In this situation, let **Def** denote the theory (in the language of T') whose axioms are all of the defining axioms. Then we may give an effective abbreviating operator **DEF** (i.e. *definitional translation*) from formulas a of the language of T' to formulas of the language of T such that **Def** $\vdash a \leftrightarrow \text{DEF}(a)$. Thus, a definitional extension is a conservative extension, but moreover, the *expressive* power of the theory is not increased. The new symbols serve mainly to emphasize the latent expressive power of the original theory. Note that a scheme of inductive definition is not apparently allowed. However, in certain situations this limitation may be overcome (cf. Chapter 2-C).

G. Arithmetic theories. K will hereafter designate the formal language of simple arithmetic determined by the symbols $0, S, +$. For languages which include K we use the abbreviating operator \bar{n} which operates on natural numbers and denotes the $(n+1)$ st *numeral* constant term: $S \dots (n\text{-times}) \dots S0$. Where σ, τ are terms we write $\sigma \leq \tau$ as an abbreviation for $\exists x (x + \sigma = \tau)$ with x suitably designated; we use $<$ similarly. The theories P, Q , and R will refer to the classical theories with language K as presented in the Tarski, Mostowski, Robinson monograph [21].

Theory **R** consists of five schemata (infinitely many axioms) of numerical equalities and inequalities. Where m and n are arbitrary natural numbers, these schemata are: $\overline{m+n} = \overline{m} + \overline{n}$; $\overline{m \cdot n} = \overline{m} \cdot \overline{n}$; $\overline{m} \neq \overline{n}$ for $m \neq n$; $x \leq \overline{n} \rightarrow (x = 0 \vee x = \overline{1} \vee \dots \vee x = \overline{n})$; and $x \leq \overline{n} \vee \overline{n} \leq x$.

Theory **Q** is an extension of **R** with the following seven axioms: $Sx = Sy \rightarrow x = y$; $0 \neq Sy$; $x \neq 0 \rightarrow \exists y(x = Sy)$; $x + 0 = x$; $x + Sy = S(x + y)$; $x \cdot 0 = 0$; and $x \cdot Sy = (x \cdot y) + x$.

By **P** we will mean Peano arithmetic which has the axioms of **Q** and also each *induction axiom* in the language **K**: $\alpha[0] \& \forall x(\alpha[x] \rightarrow \alpha[Sx]) \rightarrow \alpha[x]$.

H. Theories interpreting arithmetic. Our main discussion will consider a theory **T** with fixed language **L** and its subtheories. **L** will *interpret K* which means that **L** includes the symbols of **K** and also a unary predicate symbol N (see [18], pp. 61–65). We denote the familiar *interpretation* of a formula α of **K** by $\text{INT}(\alpha)$: first form α_N by *relativizing* α to N (replace each part $\exists x\beta$ of α by $\exists x(Nx \& \beta_N)$); then $\text{INT}(\alpha)$ is $\bigwedge_i (N\overline{x}_i \rightarrow \alpha_N$ where \overline{x}_i includes the variables free in α . INT is an effective abbreviating operator and it preserves propositional connectives. We also allow $\mathbf{L} = \mathbf{K}$ where N is absent (and in which case $\text{INT}(\alpha) = \alpha$). Let **Int** denote the theory with language **L** and the axioms: $\exists xNx$, and a closure axiom for each symbol of **K** (e.g. $Nx \rightarrow NSx$, etc.). If **S** is a theory with language **K**, let \mathbf{S}_N denote the theory with language **L**, the axioms of **Int**, and an axiom $\text{INT}(\alpha)$ for each α which is an axiom of **S**.

THEOREM 1.1. ***S** is a theory with language **K**. Then $\mathbf{S} \vdash \alpha$ iff $\mathbf{S}_N \vdash \text{INT}(\alpha)$.*

This is the familiar Interpretation Theorem of [18], p. 62. We say that **T** *interprets S* if **T** extends \mathbf{S}_N (so the interpretation is always N). For an extension of the operator INT , see Chapter 2–D.

N will denote the class of all formulas of **L** which are $\text{INT}(\alpha)$ for α a formula of **K**. If Φ is a class of formulas of **K** (cf. below) we say that $\text{INT}(\alpha)$ is a Φ *formula of N* exactly as α is a Φ formula of **K**. Other syntactic notions are carried over similarly.

I. Classes of formulas. We take the usual notions for formulas: *atomic* (it has a single predicate symbol or =); *open* (no quantifiers); *sentence* or *closed* (no free variables). An occurrence of a symbol in a formula is called *positive* exactly when it lies in the scope of an even number of \neg symbols (when abbreviations are removed). A formula α is called *existential* if every \exists is positive in α (abbreviations removed); and *universal* if no \exists is positive in α . We denote these two classes respectively by \exists and \forall .

If α is a formula of **K**, then an occurrence of \exists in α is called *bounded* if it appears in the form: $\dots \exists x(x < \tau \& \dots)$, where τ is a term in which x

does not occur. We say that α is *existential unbounded* provided every unbounded \exists is positive in α ; and *universal unbounded* if no unbounded \exists is positive in α . We denote these two classes respectively by \exists^* and \forall^* . These classes contain the \exists , and respectively, \forall classes of formulas of K . We denote the class of \exists^* -sentences and \forall^* -sentences by Γ^* .

We give a subclass of the \exists^* -formulas which is useful for comparing our results with other references. The *RE-formulas* of K are given by the two inductive conditions:

- (i) the formulas $\tau_1 = \tau_2$, $\tau_1 + \tau_2 = \tau_3$, $\tau_1 \cdot \tau_2 = \tau_3$, and their negations are *RE-formulas* exactly when each τ_i is a variable or a numeral;
- (ii) the *RE-formulas* are closed under \vee , $\&$, $\forall x(x < y \rightarrow \dots)$ where x and y are distinct, and $\exists x(\dots)$.

This class corresponds closely to "R-formula" in [18], p. 209 and "BPF" in Feferman's paper [6], p. 54. We denote the class of *RE-sentences* and their negations by Γ^{RE} .

J. Weak ω -consistency. An *RE-formula* α will be designated a *P-recursive formula* when there is an *RE-formula* β such that $P \vdash \alpha \leftrightarrow \neg\beta$. (This is not exactly the same class as the "PR-formulas" of [6], but the rôle is similar. Our class is effectively enumerable and so cannot include a formula "corresponding" to each recursive predicate. See Chapter 2-D for a more intuitive description of this class.) We say that a theory S with language K is *weakly ω -consistent* provided: if α is a *P-recursive formula* with x as its only free variable and $S \vdash \exists x\alpha[x]$, then we do not also have $S \vdash \neg\alpha[\bar{n}]$ for every n .

THEOREM 1.2 (cf. Gödel [7]). *If α is an \exists^* -sentence of K , then $Q \vdash \alpha$ iff α is true. The same holds with R instead of Q . (true: α is satisfied by the standard model \mathcal{N} whose universe is the natural numbers.)*

Remarks. The proof is first given for *RE-sentences* and then extended to \exists^* -sentences (cf. [18], pp. 209–210, Lemmas 1, 2; Lemma 3 gives the \leftarrow direction for *RE-sentences*). The \rightarrow direction is given by the weak ω -consistency of Q which follows from the usual finitary consistency proofs for Q (cf. [18], p. 51).

THEOREM 1.3 (Matijasevič [13], Robinson, Davis, Putnam [3], [4], [17]). *If α is an *RE-formula* of K , then we can effectively find an existential formula $\text{DIOPH}(\alpha)$ of K such that $P \vdash \alpha \leftrightarrow \text{DIOPH}(\alpha)$.*

Remarks. We will take $\text{DIOPH}(\alpha)$ (the *Diophantine translation*) to be a fixed translation, where the essential step is provided as follows: If α is a formula of the form $\forall x(x < y \rightarrow \beta)$ where β is existential, then we may explicitly construct an existential formula γ which is equivalent to α : using [4], Lemma 3 (etc.); [17] (5.3) and (5.4); and [3]. The equivalence of α and γ may be translated into P by a translation of certain of these arguments into recursive analysis.

Chapter 2

FORMAL ARITHMETIZATION

In the last chapter (D) we noted that formal syntax could be coded by natural numbers, and that effective syntactic operations could thereby be coded by recursive functions. In this chapter we review how these recursive functions, etc. may in turn be formally imbedded in arithmetic theories. We will need two aspects of this imbedding. First, we will represent the external notions (function and predicates) in a local or “extensional” way by formulas or defined symbols. Second, we will demand that these definitions be natural or “intensional” in that we be able to prove abstract things about them in sufficiently strong theories. Refer to the excellent discussion in Feferman’s paper [6], pp. 35–39 to contrast these two concepts.

A. Theory M . M will designate a definitional extension of P in which we have present formal symbols for a convenient discussion of some arithmetized metamathematics involving the languages K and L . For a given application we will need only finitely many symbols in M , and most of these will be given in the next few sections. Typically, we will define a given number theoretic function f (external: in italics) by a formal symbol f (internal: in Roman type). Furthermore, we will ask that f *naturally* defines f in the sense of the next section.

B. Representation and natural definition. Let S be a theory in a language which contains K and a a formula in that language. We say that a *weakly represents in S* a set of natural numbers A provided $S \vdash a[\bar{n}]$ iff $n \in A$. We say that a *represents A* (in S) if in addition, $\neg a$ weakly represents $\neg A$ (the complement of A). These notions for functions and predicates on natural numbers are similar. Representation is entirely adequate for some demonstrations such as showing the essential undecidability of R (e.g. [1] and [16]).

We will say that f is a *natural definition* for f if we can prove in M every proposition involving f which we can prove informally about f using only number theoretic methods. Although this cannot be made precise without naming all these properties of f , there is no difficulty in practice, because in defining f we need only duplicate as closely as

possible the way in which f is defined informally. For example, we may give a formula which naturally defines a finite set, simply by listing it:

DEFINITION. Let $A = \{a_1, \dots, a_n\}$ be a finite set of natural numbers. A *listing formula* for A denoted by λ_A is a formula $v_0 = \bar{a}_1 \vee \dots \vee v_0 = \bar{a}_n$ (λ_A may be fixed; λ_\emptyset is $v_0 \neq v_0$).

On the other hand, in the following chapters we will sometimes have formulas which represent (locally) the empty set, but which are not provably equivalent to λ_\emptyset even in very strong theories.

Although we will always ask that a defined symbol f of M should naturally define f , we cannot always ask that f locally *represent* what is intended by f (cf. Chapter 5-C). However, both will usually apply. In contrast, the notion of natural definition is not important in R or Q because we can give very few abstract proofs in those theories anyway.

C. Inductive definitions in M . Because it is a definitional extension, M has (as theorems) all of the induction axioms involving defined symbols. Giving definitions in M can be simplified by a number of devices. Where τ is a term with only \vec{x} free and a is a formula of M with only \vec{x}, y free, then $\tau = \mu y a[\vec{x}, y]$ will be an abbreviation for $\tau = y \leftrightarrow a[\vec{x}, y] \ \& \ \forall z (z < y \rightarrow \neg a[\vec{x}, z])$. Then we can introduce f by the defining axiom $f(\vec{x}) = \mu y a[\vec{x}, y]$ provided we can give the existence condition $M \vdash \exists y a[\vec{x}, y]$ (μ and induction take care of uniqueness).

No induction is required to introduce in M the sequence coding and decoding symbols (cf. [18], pp. 115-117 and 208). We use (with informal meanings): $\langle x_0, \dots, x_{n-1} \rangle$ ("the sequence code number of x_0, \dots, x_{n-1} "; here $\langle \rangle = 0$); $\text{Seq}(y)$ (" y is a sequence code number"); $(y)_z$ ("the $(z+1)$ st member of the sequence with code y "); $\text{lh}(y)$ ("the length of the sequence with code y "); $\text{seg}(y, z)$ ("the sequence code number of the initial segment (of length $z \leq \text{lh}(y)$) of the sequence with code number y ").

Many number theoretic functions are defined by induction, which is not directly allowed by Chapter 1-F. We may overcome this difficulty when the induction is on *finitely* many *previous* values. Suppose that g has already been defined and we want to define f by $f(x) = g(x, f^*(x))$ where f^* will naturally define a function such that

$$f^*(n) = \langle f(0), f(1), \dots, f(n-1) \rangle.$$

In this case we give f^* the defining axiom

$$f^*(x) = \mu w \left(\text{Seq}(w) \ \& \ \text{lh}(w) = x \ \& \ \forall z (z < x \rightarrow (w)_z = g(z, \text{seg}(w, z))) \right).$$

This gives f^* the desired intensional properties, and so we may define f as above, preserving naturality. We may define predicates in an analogous way, and also may use parameters. (However, see Chapter 5-C for the limitations.) We can give successor inductive definitions, etc., simply by letting g (above) involve a suitable decoding of f^* .

D. Recursive- and re-symbols in M . The above technique of imbedding inductively defined functions in M does not apply only to recursive functions. But this is a major application. Certain symbols and formulas in M will be distinguished:

DEFINITION. The symbols of K will be called *recursive-symbols* as will any defined symbol for which the defining axiom has a definiens $a[\vec{x}]$ or $\mu y a[\vec{x}, y]$ where a is an open formula having only previously defined recursive-symbols.

DEFINITION. A formula a of M is called an *existential recursive formula* if a satisfies the conditions of being \exists^* (existential unbounded: Chapter 1-I) except that a may have recursive-symbols and *positive* re-symbols of M . An *re-symbol* of M has a defining axiom for which the definiens is an existential recursive formula.

Our recursive-symbols are the "symbols introduced in a recursive extension" in [18], pp. 206-208. These notions for symbols permit very natural introductions of recursive and recursively enumerable functions and predicates in M . Any *provably* total recursive function or predicate may be naturally defined by a recursive-symbol. Any recursively enumerable predicate may be naturally defined by an re-symbol.

If f is a partial (or not provably total) recursive function we may define f in M by $f(\vec{x}) \simeq y$ where f and \simeq together constitute an re-symbol for the graph of f .

THEOREM 2.1. *We may fix a definitional translation DEF from formulas of M to K such that:*

(i) *If a is an existential recursive formula of M , then $\text{DEF}(a)$ is an RE-formula of K (and $\text{Def} \vdash a \leftrightarrow \text{DEF}(a)$; see Chapter 1-F).*

(ii) *If f is a recursive symbol of M which naturally defines the recursive function f , then $\text{DEF}(f(\vec{x}) = y)$ represents f in R (predicates analogous).*

(iii) *If f, \simeq constitute an re-symbol of M which naturally defines a partial recursive function f , then $\text{DEF}(f(\vec{x}) \simeq y)$ weakly represents f in R ; and represents, if f is total (recursively enumerable predicates are analogous).*

Remarks on proof. The definitional translation DEF on existential recursive formulas is provided by minor modifications of [18],

pp. 209–210 (Lemmas 1 and 2). Then for any formula β , this translation is used on maximal existential recursive subformulas of β (and in some reasonable way on other symbols). Parts (ii) and (iii) may be obtained using (i) and Theorem 1.2. ■

The representation of (ii) also holds for \mathbf{P} because representation is already “frozen” by \mathbf{R} . Weak representation in (iii) also holds for \mathbf{P} but only because \mathbf{P} is weakly ω -consistent. (Otherwise the weakly represented predicate would be larger than intended.) The definitional translations of recursive predicate symbols correspond very closely to the \mathbf{P} -recursive formulas of Chapter 1–J.

With DEF fixed as in Theorem 2.1, we extend the abbreviating operator INT (Chapter 1–H) to formulas a of \mathbf{M} by $\text{INT}(a) = \text{INT}(\text{DEF}(a))$. However to simplify notation we sometimes write symbols of \mathbf{M} directly in \mathbf{L} , but the translation INT is intended.

E. Symbols for arithmetized syntax. We next list some formal symbols of \mathbf{M} for the arithmetized analogs of syntactic notions (along with their informal meanings). These are all recursive-symbols, and we assume that they have been given in some fixed natural way. For a given expression φ (of \mathbf{K} or \mathbf{L}) we will write $\bar{\varphi}$ instead of $\ulcorner \varphi \urcorner$ (the numeral for the expression code number of φ). Here we will also let φ_n denote the expression (of \mathbf{K} or \mathbf{L}) with code number n .

Predicate symbols (a subscript \mathbf{K} or \mathbf{L} will be understood and often omitted): $\text{Term}_{\mathbf{K}}(x)$ (φ_x is a term of \mathbf{K}); $\text{Atomic}(x)$ (φ_x is an atomic formula); $\text{Formula}(x)$ (φ_x is a formula); $\text{Sentence}(x)$ (φ_x is a sentence); $\text{REformula}(x)$ (φ_x is an *RE*-formula of \mathbf{K}).

Function symbols: $\text{neg}(x)$ (code of $\neg \varphi_x$); $\text{disj}(x, y)$ (code of $\varphi_x \vee \varphi_y$); $\text{imp}(x, y)$ (code of $\varphi_x \rightarrow \varphi_y$); $\text{conj}(x, y)$ (code of $\varphi_x \& \varphi_y$); $\text{var}(x)$ (code for the $(x+1)$ st variable); $\text{freevariables}(x)$ (the sequence code for the code numbers of variables which occur free in φ_x); $\text{exist}(\text{var}(0), x)$ (code for $\exists v_0 \varphi_x$); $\text{univ}(\text{var}(0), x)$ (code for $\forall v_0 \varphi_x$); $\text{closure}(x)$ (code for the closure of φ_x); $\text{suc}(x)$ (code for $S\varphi_x$ where φ_x is a term of \mathbf{K}); $\text{num}(x)$ (code for the $(x+1)$ st numeral); $\text{equals}(x, y)$ (code for $\varphi_x = \varphi_y$ where φ_x, φ_y are terms); $\text{sub}(x, \text{var}(0), \text{num}(\bar{n}))$ (code number of $\varphi[\bar{n}]$ where φ_x is $\varphi[v_0]$); of course, we allow any arguments which make sense, and also for simultaneous substitution by the “vector” notation: e.g. $\text{sub}(x, \vec{\text{var}}(\vec{i}), \vec{\text{num}}(v_i))$ for “substitute the (v_i+1) st numeral for the $(i+1)$ st variable”; the arrows of course denote appropriate sequence codes.

Particularly note the analogs for special abbreviating operators: $\text{int}(x)$ (code for $\text{INT}(\varphi_x)$); $\text{dioph}(x)$ (code for $\text{DIOPH}(\varphi_x)$, the Diophantine translation of the *RE*-formula φ_x).

F. Proof predicates. We next display how we give natural proof definitions in \mathbf{M} . For this purpose we assume we have the recursive

symbols: $\text{LogicAxiom}(x)$ (φ_x is a logical axiom); $\text{InferenceRule}(x, y, z)$ (φ_z is a consequence of φ_x and φ_y by one of our given rules of logical inference). A particular proof definition will be fixed upon choosing the axiom formula $\alpha[z; v_0]$ of M and the language K or L (we show L):

$$\begin{aligned} \text{Prf}_{L,\alpha}(z; y, x) \leftrightarrow & \text{Seq}(y) \& y \neq 0 \& x = (y)_{\text{lh}(y)-1} \\ & \& \forall w (w < \text{lh}(y) \rightarrow \text{Formula}_L((y)_w) \& (\text{LogicAxiom}((y)_w) \\ & \vee \alpha[z; (y)_w] \vee \exists u \exists v (u < w \& v < w \& \text{InferenceRule}((y)_u, \\ & (y)_v, (y)_w))) \end{aligned}$$

The “extra” parameter z will often be omitted, but it will also be useful in Chapter 4–B (where used, it is set off by “;” for emphasis). We give a few of the particular definitions we will use:

$$\text{Prf}_{L,\sigma}(y, x) \quad \text{as the definition} \quad \text{Prf}_{L,\alpha}(y, x)$$

(or subscript K) where we take $v_0 \neq v_0$ as the axiom formula $\alpha[z; v_0]$ (i.e. logical provability). We will take

$$\text{Prf}_Q(y, x) \quad \text{as the definition} \quad \text{Prf}_{K,\alpha}(y, x)$$

where the axiom formula α is $\lambda_Q[v_0]$ the listing formula for the finite set Q of code numbers for the axioms of theory Q .

As usual, we will use the definitions $\text{Prv}_\alpha(z; x)$ for $\exists y \text{Prf}_\alpha(z; y, x)$ and $\text{Con}_\alpha(z)$ for $\text{Prv}_\alpha(z; 0 \neq 0)$, but as Prf_α is intensionally correct, any other natural definitions for these would be equivalent in M . Notice that Prv_α is an re-symbol whenever α is an existential recursive formula of M .

THEOREM 2.2.

$$M \vdash \forall x (\lambda_{Q_N}[x] \rightarrow \text{Prv}_{L,\sigma}(x)) \rightarrow \forall x (\text{Prv}_Q(x) \rightarrow \text{Prv}_{L,\sigma}(\text{int}(x)))$$

where Q_N is the set of code numbers of the axioms of Q_N .

THEOREM 2.3. Let α be a \exists^* (existential unbounded) formula of K such that all of the free variables of α are included in \vec{v}_i .

Then

$$M \vdash \alpha[\vec{v}_i] \leftrightarrow \text{Prv}_Q(\text{sub}(\vec{a}, \vec{\text{var}}(\vec{i}), \vec{\text{num}}(v_i)))$$

Comment on proofs. Theorem 2.2 is a formalization of Theorem 1.1. This is an example of the typical syntactic proposition which can be formalized in M . Theorem 2.3 is a formalization of Theorem 1.2. (Even though Theorem 2.3 is a schema rather than a single theorem of M , it is possible to produce a single finitely axiomatized subtheory of M in which all of these equivalences can be established.) ■

THEOREM 2.4 (Gödel's Lemma of Self-Reference [7]).

(i) Let $\alpha[v_0]$ be a formula of \mathbf{L} with v_0 as its only free variable; then there is a sentence γ of \mathbf{L} such that

$$\mathbf{R}_N \vdash \gamma \leftrightarrow \alpha[\bar{\gamma}].$$

(ii) Assume instead that $\alpha[v_0]$ is an existential recursive formula of \mathbf{M} (Chapter 2-D); then γ may be chosen an RE -sentence such that

$$\mathbf{R} \vdash \gamma \leftrightarrow \text{DEF}(\alpha[\bar{\gamma}]).$$

(iii) As in (ii), let $\eta[v_0]$ be an existential recursive formula of \mathbf{M} ; then ψ may be chosen as an existential sentence of \mathbf{K} such that

$$\mathbf{P} \vdash \psi \leftrightarrow \text{DEF}(\eta[\bar{\psi}]).$$

Adaptation of proofs. For (i) let $\beta[v_0]$ be $\exists y(\alpha[y] \& \text{INT}(\text{sub}(v_0, \text{var}(0), \text{num}(v_0)) = y))$ and let γ be $\beta[\bar{\beta}]$. For (ii) let $\beta[v_0]$ be $\text{DEF}(\exists y(\alpha[y] \& \text{sub}(v_0, \text{var}(0), \text{num}(v_0)) = y))$ and let γ be $\beta[\bar{\beta}]$. For (iii) we use (ii) where $\alpha[v_0]$ is taken as $\exists y(\eta[y] \& \text{dioph}(v_0) = y)$. Applying (ii) we get γ as an RE -sentence and so take ψ as $\text{DIOPH}(\gamma)$ by Theorem 1.3. And so it follows from that theorem that $\mathbf{P} \vdash \psi \leftrightarrow \gamma$ and also $\mathbf{M} \vdash \text{dioph}(\bar{\gamma}) = \bar{\psi}$. ■

G. Recursive inseparability. We conclude this chapter with a familiar application of the Lemma of Self-Reference. Let Φ be a class of sentences of \mathbf{L} which includes with each sentence its negation. Recall that Φ is called a *recursively inseparable class for theory \mathbf{T}* if there is no recursive set B such that B includes the code numbers of all the formulas of Φ which are also theorems of \mathbf{T} , and $\neg B$ includes the code numbers of all the formulas of Φ which are also refutations of \mathbf{T} (refutations: their negations are theorems). This is, of course, a strong form of essential undecidability for \mathbf{S} .

THEOREM 2.5. *The class Γ^{RE} (RE -sentences and their negations) is a recursively inseparable class for the theory \mathbf{R} . By Theorem 1.1, the analogous statement holds for \mathbf{R}_N .*

Proof. Let B be any recursive set which contains the code numbers of the theorems of \mathbf{R} which are in Γ^{RE} . By Theorem 2.1 (ii) there is an RE -formula $\alpha[v_0]$ which represents $\neg B$ in \mathbf{R} (because $\neg B$ is also recursive). By 2.4 (ii) we get an RE -sentence γ such that $\mathbf{R} \vdash \gamma \leftrightarrow \alpha[\bar{\gamma}]$. If $\ulcorner \gamma \urcorner$ were in $\neg B$, then $\mathbf{R} \vdash \alpha[\bar{\gamma}]$ by representation, and so we would have $\mathbf{R} \vdash \gamma$. But by hypothesis, this says $\ulcorner \gamma \urcorner$ is in B , because γ is also in Γ^{RE} . So $\ulcorner \gamma \urcorner$ is in B and by representation $\mathbf{R} \vdash \neg \alpha[\bar{\gamma}]$ and so $\mathbf{R} \vdash \neg \gamma$. But then $\ulcorner \gamma \urcorner$ is in B at the same time \mathbf{R} refutes γ . ■



Chapter 3

PROOF RESTRICTION FUNCTIONS

We consider the problem of restricting the axiom set for a theory T . However, instead of the customary "global" axiomatization problem we will consider whether we can have certain "local" restrictions. By "local" axiom restrictions we mean correspondences of the following sort: for each sentence a of L (or a part of L) associate a subtheory F_a of T such that a can be proved from F_a if it can be proved from T (regardless of whether or not a can be proved). To make the question about the possibility of such correspondences significant, we should ask that the correspondence be recursive, and that the F_a satisfy certain conditions. In the next two chapters we will consider the situation where we ask each F_a to be considerably more simple than T : If T cannot be axiomatized globally by an \mathcal{F} (e.g. finite, etc.) set of axioms, then we ask each if F_a can be in \mathcal{F} . (Apparently, we might hope to produce such a natural correspondence $a \mapsto F_a$ by studying the syntax of T .) In this chapter we note some simple consequences of these notions.

A. Presenting the subtheories of T . If we wish to speak precisely about correspondences $a \mapsto F_a$ between sentences and theories, we face the immediate problem: We need not only a code for sentences (which we have) but also a code for *theories*. However, because we are interested here only in recursively enumerable theories, there is a very obvious way to do this:

DEFINITION. By a *presentation* of a recursively enumerable theory S with language L , we mean an RE -formula $\sigma[v_0]$ of K with v_0 as its only free variable, such that $\sigma[v_0]$ weakly represents in R the set of code numbers for axioms of S . Furthermore, if the axiom set of S is described and σ is (the definitional translation of) a natural definition for that description (in the sense of Chapter 2-B) then we will say that σ is *natural* for S , as described. We call $\ulcorner \sigma \urcorner$ the *presentation code*.

One should note that even though a theory S is distinguished only by its axiom set, a presentation of S is further distinguished by how that axiom set is described. If S is finitely axiomatized where the axiom code numbers are a finite set \mathcal{S} , then $\lambda_{\mathcal{S}}$ is the natural presentation for S .

B. Proof restrictions.

DEFINITION. We take a *proof restriction for α* (in \mathbf{T}) to be a recursively enumerable subtheory F_α of \mathbf{T} such that if $\mathbf{T} \vdash \alpha$ then $F_\alpha \vdash \alpha$. Noting the remarks above we will ask F_α to be presented by an *RE*-formula σ (and *naturally* presented when F_α is *described*).

Remark. The proofs (if any) of α in F_α need not be the same, or as short as the proofs (if any) in \mathbf{T} .

DEFINITION. A *proof restriction function* (for \mathbf{T}) for a class Φ of sentences of \mathbf{L} is a function f such that for each α in Φ , $f \ulcorner \alpha \urcorner$ is a presentation code $\ulcorner \sigma \urcorner$ of a proof restriction F_α for α . Furthermore, if each value $f \ulcorner \alpha \urcorner$ of f is some $\ulcorner \lambda_S \urcorner$ (a listing formula, cf. Chapter 2-B) for a finite set of axioms or theorems of \mathbf{T} (dependent on α), then we call f a *finite proof restriction function*.

Remarks. We emphasize strongly that we do *not* require the formalized condition “ f is a proof restriction function for Φ ” to be *provable* in \mathbf{T} , even though we could *express* this in \mathbf{T} . (This might be an unreasonable demand because we can often give arguments about the syntax of \mathbf{T} which we cannot formalize in \mathbf{T} : e.g. constructive consistency proofs for \mathbf{P}). We have defined a proof restriction function to be a correspondence between certain code numbers so that it is clear what we mean when we say f is recursive, etc. However, in practice, below, when we produce a proof restriction function we informally describe F_α instead of computing its presentation code.

C. Immediate observations. We are naturally interested in those \mathbf{T} with recursively enumerable sets of axioms (a recursive proof restriction function for all sentences of \mathbf{T} automatically endows \mathbf{T} with such a set of axioms). Where the axioms of \mathbf{T} are recursively enumerable, the constant function which gives for each α the set of axioms of \mathbf{T} is trivially a proof restriction function. Thus if \mathbf{T} is finitely axiomatized it has such a finite proof restriction function. Usually we will not be interested in such trivial functions nor in finitely axiomatized \mathbf{T} . A less trivial but equally uninformative proof restriction function is the following:

THEOREM 3.1. *If \mathbf{T} is a recursively enumerable theory then it has a total recursive proof restriction function whose values weakly represent (in \mathbf{R}) finite sets of axioms (total: for all sentences of \mathbf{L}).*

Proof. Let \mathbf{T} be presented by τ . Then for each sentence α of \mathbf{L} , let $f \ulcorner \alpha \urcorner$ be the code number of the presentation formula:

$$\text{DEF} \left(\exists p \left(\text{Prf}_\tau(p, \bar{a}) \& \forall q (q < p \rightarrow \neg \text{Prf}_\tau(q, \bar{a})) \& \exists z (z < \text{lh}(p) \& (p)_z = v_0 \& \tau[v_0]) \right) \right).$$

This specifies the (finitely many) axioms used in the first proof of α .

If there is no proof, it does not specify any axioms. Clearly, f is a proof restriction function as sought. ■

More informative proof restriction functions should have further requirements on the values of f . It does not help to ask that $f^{\ulcorner a \urcorner}$ should represent a recursive set of axioms (rather than weakly represent a recursively enumerable set). We may modify the presentation formula of 3.1 to satisfy these conditions using Craig-style devices: Instead of specifying an axiom β used in the first (if any) proof p of a , we specify the theorem $\beta \& v_p = v_p$ (actually, we can usually even produce an *axiom* which "codes" p).

Having to guess *which* finite set of axioms, as required by a finite proof restriction function, would be more informative. (The function of 3.1 presents finite subtheories, but they are not explicit as would be demanded in a finite proof restriction function.) In the remainder of this chapter we consider finite proof restriction functions. The following simple facts are collected together:

THEOREM 3.2. (i) *If T is consistent and has a recursively enumerable set of axioms then T has a partial recursive finite proof restriction function for the set of theorems and negations of theorems of T .*

(ii) *If T is decidable then it has a recursive finite proof restriction function for all sentences.*

(iii) *If T is a definitional extension of T' , then a proof restriction function f' for T' naturally expands to one for T .*

(iv) *If T interprets Q and T is weakly ω -consistent, then T has a recursive finite proof restriction function for the RE-sentences of N .*

Proof sketch: (i) We can list the theorems of T . If a is listed then $f^{\ulcorner a \urcorner}$ is computed as the list of axioms used in the first proof of a . At the same time $f^{\ulcorner \neg a \urcorner}$ is set equal to the empty set.

(ii) Apply an obvious modification of the argument for (i).

(iii) We translate a back into the language of T' and get $f^{\ulcorner \text{DEF}(a) \urcorner}$. We combine this with the list of the defining axioms and defining conditions upon which the new symbols of a depend.

(iv) The sought function is the constant function which gives every RE-formula $\text{INT}(a)$ the axioms of Q_N . To show that Q_N is a proof restriction for each $\text{INT}(a)$ we must make some prenex operations on a (move all \exists quantifiers in front, contract, etc.) to get the matrix to be a P -recursive formula where we can use weak ω -consistency. For details, combine techniques from [18], pp. 151–152 and 207–210. ■

The utility of (iii) is that if L interprets K , it is often the case that T is a definitional extension of a theory T' (e.g. set theory) in which N and the symbols of K must be defined. Thus later when we obtain negative results about proof restriction functions for T we use (iii) to infer that

they extend back to the original theory T' . Conclusion (ii) points out that the notion of proof restriction function is intended for undecidable theories. Conclusions (i) and (iv) suggest that we ask about proof restriction functions with (total or recursive) domains extending beyond RE -sentences, and beyond theorems and refutations. The next result also suggests that we ask the proof restriction function to be recursive.

THEOREM 3.3. *If T has a recursively enumerable set of axioms, then T has a total Π_1^0 finite proof restriction function (total: for all sentences of L).*

Proof. $l(k)$ which lists the k th conjunction of axioms of T (i.e. $\ulcorner \alpha_0 \& \dots \& \alpha_{k-1} \urcorner$) is recursive and so is its inverse l^{-1} on the range of l . Define a recursive function

$$h(n, k) = l \left(\mu j_{j \leq k} \left(\exists p_{p \leq j} \text{Prf}_{\text{Log}}(p, \text{imp}(l(k), n)) \right); \text{otherwise } j = 0 \right).$$

Thus, for n fixed, $h(n, k)$ stabilizes upon finding a proof of φ_n with all its succeeding values being the axioms used in that proof. If there is no proof, then $h(n, k)$ remains always 0. Thus we can define a total proof restriction function f with a Π_1^0 graph as follows:

$$f(n) = m \leftrightarrow \forall k (k \geq l^{-1}(m) \rightarrow h(n, k) = m). \blacksquare$$

As might be expected, some infinitely axiomatized theories will fail to have recursive finite proof restriction functions because their finitely axiomatized subtheories are "much more simple" than the whole theory:

LEMMA 3.4. *Let Φ be a recursively inseparable class of sentences for a consistent theory T . Furthermore, suppose that with each finitely axiomatized subtheory S of T we can effectively associate both a particular consistent extension S' of S and a decision procedure for S' . Then T fails to have a recursive finite proof restriction function for Φ .*

Proof. Supposing that T had such a proof restriction function f we could produce a recursive separation $B, \neg B$ of Φ :

For a and $\neg a$ in Φ we form the union of the subtheories presented by $f^{\ulcorner a \urcorner}$ and $f^{\ulcorner \neg a \urcorner}$, which we call S . We then obtain the decidable extension S' of S as specified by the hypotheses. We decide a and $\neg a$ in S' . If either is provable in S' we put that sentence in B and the other in $\neg B$. Otherwise we can do anything. Thus B is recursive.

For a in Φ , if $T \vdash a$, then $S \vdash a$ by assumption on f , so $S' \vdash a$ and thus a is in B and $\neg a$ in $\neg B$. Thus $B, \neg B$ is a recursive separation of Φ . \blacksquare

THEOREM 3.5. *The theory R fails to have a recursive finite proof restriction function for the class Γ^{RE} (RE sentences and their negations).*

Proof. By Theorem 2.5, Γ^{RE} is a recursively inseparable class for R . We now show that R satisfies the other hypothesis of the lemma;

We will assume that S is composed of finitely many axioms of R (if S were composed of theorems of R , replace S by axioms from which these theorems were proven). Let \bar{n} be the largest numeral which appears in S .

Let $\mathcal{A} = \langle A, 0_a, S_a, +_a, \cdot_a \rangle$ be the structure for K which is the initial segment of the natural numbers $\{0, 1, \dots, n\}$ with the other symbols interpreted normally, except we put $S_a n = n$ and similarly let the operations $+_a$ and \cdot_a "pile-up" on n . It is easy to check that \mathcal{A} satisfies the axioms of S . Let S' be the theory of all formulas of K satisfied by \mathcal{A} . S' extends S and is decidable (complete) because it is the theory of a finite structure. ■

D. Example of a proof restriction function. In contrast to the above, there are non-finitely axiomatized recursively inseparable theories which have recursive finite proof restriction functions. We give a simple example of such a theory.

Here we let U denote the theory with language $L = K \cup \{N\}$ (N is a unary predicate). The axioms of U include:

- (i) Q_N (the axioms of Q_N);
- (ii) annihilation axioms for the function symbols on $\neg N$ (i.e. $\neg N x \rightarrow x + y = 0$, etc.); and
- (iii) for each positive integer k an axiom η_k which asserts the existence of at least k distinct elements in $\neg N$:

$$\exists x_1 \dots \exists x_k \left(\bigwedge_{1 \leq i \leq k} (\neg N x_i) \& \bigwedge_{i < j \leq k} (x_i \neq x_j) \right).$$

In addition we let U_k denote the finitely axiomatized subtheory of U consisting of (i), (ii), and the single axiom η_k .

For the purposes of this example we define: the *variable depth* of a (formula of L) is the maximum number k of distinct variables of a . If \mathcal{A} is a model of Q , let \mathcal{A}_k here denote the unique (up to isomorphism) structure for L obtained by adding a set B of k new distinct elements to the universe A of \mathcal{A} (B to be interpreted as $\neg N$) and then interpreting the other symbols of L either in duplication of \mathcal{A} or else the axioms of U_k .

LEMMA 3.6. *If a has variable depth $\leq k$ and $U \vdash a$ and \mathcal{A} is a model for Q , then a holds in \mathcal{A}_k .*

Idea. The axioms of Q_N say nothing about $\neg N$ and the annihilation axioms (ii) prohibit there from being any significant definable relations amongst elements of $\neg N$ or between elements of N and $\neg N$. a is equivalent to the disjunction of all possible relativizations of its quantifiers to N and $\neg N$, at worst they are all relativized to $\neg N$. Without giving details, these imply that a cannot depend on any fact which requires more than k elements in $\neg N$. ■

THEOREM 3.7. (i) U is not finitely axiomatizable.

(ii) Let f be the function which associates with each a of L the subtheory U_k where k is the variable depth of a . f is a finite proof restriction function for U .

Proof. For the first assertion, if S were a finite set of axioms for U , they would have some maximum variable depth k and so by the lemma, they would all hold in any \mathcal{A}_k . But η_{k+1} does not hold in any \mathcal{A}_k , so S could not axiomatize U , considering the Completeness Theorem [18], p. 43. For the second assertion, if a has variable depth k and $U \vdash a$ then a holds in any \mathcal{A}_j (with $j \geq k$) by the Lemma. But noting the Completeness Theorem, these structures characterize the theorems of U_k . ■

In fact, no consistent extension of U can be finitely axiomatized without adding new symbols because the annihilation axioms (ii) prevent us from saying anything about $\neg N$ with those symbols available. In contrast to Lemma 3.3, it is worth noting that each of the finitely axiomatized U_k in this example are themselves recursively inseparable. This simple example is not surprising because the source of the recursive inseparability and the non-finite axiomatizability are kept separate. However, it does indicate that for a recursively inseparable theory T , a negative solution concerning certain proof restriction functions will not follow *simply* from the assumption that T cannot be *globally* axiomatized in a certain way. On the other hand, both of these are often consequences of the same situation, as we shall see again in the next chapter.

Chapter 4

REFLEXIVE THEORIES AND PROOF RESTRICTION FUNCTIONS

If a theory T is quite strong in the sense that T interprets P , then it *makes sense* for T to discuss formalized syntax, because we can present many formalized notions in T naturally and often unambiguously. Thus we can express properly the consistency of a subtheory S of T in this formalized manner as a sentence of L . Furthermore, we can often even *prove* this sentence *in* T . If we can prove in T these sentences for a large number of subtheories S of T , then we refer to T as “reflexive”. In this chapter we will see that reflexivity for T yields negative results about recursive proof restriction functions for T . In the next chapter we will recall some classical results by which reflexivity is established for various theories.

A. Reflexivity. Where S is a recursively enumerable theory with language L , we express in M the proofs of S by $\text{Prf}_{L,\sigma}(p, x)$ where σ is a presentation of S . This and other symbols of M are then correctly expressed in L through the abbreviating operator INT (which is already extended to M according to Chapter 2-D).

DEFINITIONS. The *reflection set* for T is the set of all “ σ ” such that “ σ ” is the presentation code of a subtheory of T and also $T \vdash \text{INT}(\text{Con}_{L,\sigma})$. Where C is a set of presentation codes, we say T is *C-reflexive* if the reflection set for T includes C . (For example, if C is the set of “ λ_S ” for presentations of all finite subtheories S of T , then we say *finitely reflexive* for C -reflexive.)

B. Proof predicate for arbitrary subtheories. We will need to discuss, formally, the provability of arbitrary recursively enumerable subtheories of T . In the following proof definition (definition in M) we use the free parameter z to range over presentations of theories in language L :

DEFINITION. $\text{Prf}_{RE}(z; p, x)$ is the definition $\text{Prf}_{L,\alpha}(z; p, x)$ where for the axiom formula $\alpha[z; v_0]$ we take:

$$\text{REformula}(z) \& \text{freevariables}(z) = \langle \text{var}(0) \rangle \& \text{Prv}_{\mathcal{Q}}(\text{sub}(z, \text{var}(0), \text{num}(v_0))).$$

Notice that we may take Prf_{RE} as an re-symbol of M because of the form of $\alpha[z; v_0]$. We next note that this definition is “correct”:

LEMMA 4.1. *If $\sigma[v_0]$ is a presentation code then*

$$\mathbf{M} \vdash \text{Prf}_{\mathbf{L},\sigma}(p, x) \leftrightarrow \text{Prf}_{RE}(\bar{\sigma}; p, x).$$

Proof. Theorem 2.3 gives $\mathbf{M} \vdash \sigma[v_0] \leftrightarrow \text{Prv}_{\mathcal{Q}}(\text{sub}(\bar{\sigma}, \text{var}(0), \text{num}(v_0)))$ and so the result follows directly by replacing equivalent parts in the definitions. ■

We are thinking of a function f as being a proposed proof restriction function. We are going to apply the Lemma of Self-Reference to produce a sentence which says: “I am not provable in the subtheory assigned to me by f ”. This involves a simultaneous double diagonalization: one on the sentence, and the other on the theory (needing the second of these motivated giving the definition $\text{Prf}_{RE}(z; p, x)$ with parameter z). As f might only be partial (or not *provably* total) we introduce f as a natural definition in \mathbf{M} by $f(x) \simeq y$. However, we will also write a particular value as $f(\bar{\gamma})$ outright.

LEMMA 4.2. *Suppose that \mathbf{T} interprets \mathbf{P} and that f is a recursive function such that for every universal sentence a of \mathbf{N} , $f^{\ulcorner a \urcorner}$ is defined and is the presentation code of a theory which interprets \mathcal{Q} . Then there is a universal sentence γ of \mathbf{N} and σ such that*

$$\mathbf{T} \vdash \text{INT}(f(\bar{\gamma}) \simeq \bar{\sigma}) \& (\text{INT}(\text{Con}_{\mathbf{L},\sigma}) \leftrightarrow \gamma).$$

Proof. By Theorem 2.4 (iii) – the Lemma of Self-Reference – we obtain an existential sentence β of \mathbf{K} such that

$$(1) \quad \mathbf{M} \vdash \beta \leftrightarrow \exists y (f(\text{neg}(\text{int}(\bar{\beta}))) \simeq y \& \text{Prv}_{RE}(y; \text{neg}(\text{int}(\bar{\beta}))).$$

Let γ be $\neg \text{INT}(\beta)$. So γ is a universal sentence of \mathbf{N} . Also let σ be the presentation formula such that $f^{\ulcorner \gamma \urcorner} = \ulcorner \sigma \urcorner$. Then by representability of INT and f by their formal definitions

$$(2) \quad \mathbf{M} \vdash \text{neg}(\text{int}(\bar{\beta})) = \bar{\gamma},$$

and

$$(3) \quad \mathbf{M} \vdash f(\bar{\gamma}) \simeq \bar{\sigma}.$$

Then using (2), (3), and Lemma 4.1 we convert (1) into

$$(4) \quad \mathbf{M} \vdash \beta \leftrightarrow \text{Prv}_{\mathbf{L},\sigma}(\bar{\gamma}).$$

Now the theory presented by σ interprets \mathcal{Q} , and (because \mathcal{Q} is finite) this fact may be *proven*. So σ satisfies the formal hypothesis in Theorem 2.2 and so $\mathbf{M} \vdash \text{Prv}_{\mathcal{Q}}(x) \rightarrow \text{Prv}_{\mathbf{L},\sigma}(\text{int}(x))$. And noting that β is an existential sentence of \mathbf{K} we have $\mathbf{M} \vdash \beta \rightarrow \text{Prv}_{\mathcal{Q}}(\bar{\beta})$ by Theorem 2.3. Combining these last two observations we have

$$(5) \quad \mathbf{M} \vdash \beta \rightarrow \text{Prv}_{\mathbf{L},\sigma}(\text{int}(\bar{\beta})).$$

But then, combining (2), (4), (5) and noting that the definitions Prv and Con are natural we have

$$(6) \quad M \vdash \beta \rightarrow \neg \text{Con}_{L,\sigma}.$$

The other direction is trivial and so we have

$$(7) \quad M \vdash \text{Con}_{L,\sigma} \leftrightarrow \neg \beta.$$

We now pass through the interpretation into L , combining (3) and (7) and we obtain:

$$T \vdash \text{INT}(f(\bar{\gamma}) \simeq \bar{\sigma}) \& (\text{INT}(\text{Con}_{L,\sigma}) \leftrightarrow \gamma). \blacksquare$$

LEMMA 4.3. *Assume the conditions of Lemma 4.2 and also that T is consistent. Let γ and σ be as obtained in Lemma 4.2 and suppose also that σ presents a subtheory S of T . Then γ is not a theorem of S .*

Proof. Suppose that $S \vdash \gamma$. Then $M \vdash \text{Prv}_{L,\sigma}(\bar{\gamma})$ because the symbol $\text{Prf}_{L,\sigma}$ represents the proof predicate for S . But then noting (4) of Lemma 4.2 (the "meaning" of β) we have $M \vdash \beta$. We then pass into L through the interpretation and this becomes

$$(8) \quad T \vdash \text{INT}(\beta).$$

But also, if $S \vdash \gamma$ then so does its extension T . That is,

$$(9) \quad T \vdash \neg \text{INT}(\beta),$$

which is the same thing. But (8) and (9) cannot hold together if T is consistent. ■

DEFINITION. Where C is a set of presentation codes (language L) we will say that C is Q -closed provided whenever $\ulcorner \varrho \urcorner$ belongs to C then $\ulcorner \varrho \vee \lambda_{Q_N} \urcorner$ belongs to C where Q_N is the set of code numbers for the axioms of Q_N .

THEOREM 4.4. *Assume that T is a consistent theory which interprets P , and that C is a Q -closed set of presentation codes. Let g be a recursive function such that for every universal sentence α of N , $g\ulcorner \alpha \urcorner$ is defined and is in C . If T is C -reflexive, then g is not a proof restriction function for T ⁽²⁾.*

Proof. We define a function f with the same domain as g : If $g\ulcorner \alpha \urcorner = \ulcorner \varrho \urcorner$ we set $f\ulcorner \alpha \urcorner = \ulcorner \varrho \vee \lambda_{Q_N} \urcorner$. By hypothesis, the values of f are still in C and if g were a proof restriction function f would be also. We show that f is not a proof restriction function.

By construction, when $f\ulcorner \alpha \urcorner$ is defined it presents a theory which interprets Q . If we also assume that T is C -reflexive, then T and f satisfy the hypotheses of both lemmas. Let γ , σ , and S be as in the lemmas.

⁽²⁾ The author has found that a similar result was also noted in [10], p. 133 (Theorem 19).

Because $f^{\ulcorner \gamma \urcorner} = \ulcorner \sigma \urcorner$ is in the reflection set for T , S is a subtheory of T and $T \vdash \text{INT}(\text{Con}_{L, \sigma})$. Combining this with Lemma 4.2 we obtain $T \vdash \gamma$. Yet by Lemma 4.3 we must conclude that γ is not a theorem of S . But these mean that f has *failed* as a proof restriction function at the particular universal sentence γ of N . ■

Remarks. Theorem 4.4 is a proper generalization of both Gödel's Second Underivability Theorem (on consistency proofs) and also the negative global axiomatization result for many theories. To see the first, let T be naturally presented by τ , and let g be the function which is constantly $\ulcorner \tau \urcorner$. Obviously g is a proof restriction function, and so $\ulcorner \tau \urcorner$ is not in the reflection set for T . To see the second, if ϱ presents some proposed axiomatization of T and $\ulcorner \varrho \urcorner$ is in the reflection set for T , then the function which is constantly $\ulcorner \varrho \urcorner$ could not be a proof restriction function, and so the axioms presented by ϱ would not suffice for T .

It is also worth noting that for the theorem to hold, we did not require the S presented by $f^{\ulcorner \gamma \urcorner}$ to be a subtheory of T , but only that $S \vdash \alpha$ implies $T \vdash \alpha$ (the reverse implication of the proof restriction requirement). To see this check Lemma 4.3. Thus Theorem 4.4 holds for a broader notion of proof restriction function.

C. Applications. In the remainder of this chapter we will note some applications of Theorem 4.4 and also an exception to a stronger assertion. We will depart from our normal linear development and present the applications now. Then in Chapter 5 we will recall the classical results upon which these applications depend.

DEFINITION. We will call T a *Peano theory* if L contains K and T extends Q and also has as a theorem, the induction axiom $\alpha[0] \& \forall x (\alpha[x] \rightarrow \alpha[Sx]) \rightarrow \alpha[x]$ for each formula α of the language L .

This notion automatically includes every extension T of P *provided* it has the same language K as P . It also includes the definitional extensions of such T . But furthermore it can also include theories with symbols which are not even consistently definable in P (see example Chapter 5-D).

THEOREM 4.5. *No consistent Peano theory T has a recursive finite proof restriction function for the universal sentences of K .*

Proof. We anticipate Theorem 5.7 which says that T is finitely-reflexive. The set of λ presentations for finite subtheories of T is clearly Q -closed. Thus the result follows from Theorem 4.4. ■

We mention also a typical application to set theory. By **ZF** we will mean Zermelo–Fraenkel set theory with language L consisting only of the binary predicate ϵ . By L_N we mean the definitional extension of L which is the usual interpretation of K (i.e. $Nx \leftrightarrow x \in \omega$, etc.).

THEOREM 4.6. *If T is a consistent extension of **ZF** with language L , then T fails to have a recursive finite proof restriction function.*

Proof. We anticipate Theorem 5.4 which says that T is finitely reflexive, and conclude that the same holds for its definitional extension T_N . Given a function f on codes for sentences of L we construct g for L_N in the obvious way noting Theorem 3.1 (iii). We may then apply Theorem 4.4. In fact g and thus f would fail at the interpretation of the sentence γ . ■

D. Quantifier depth. We next obtain a negative result about the possibility of certain proof restrictions whose values are the presentation codes of non-finitely axiomatized subtheories – namely those of bounded quantifier depth. This result will apply to P .

DEFINITION. By the *depth* of a formula a of L (or K) we mean the measure defined inductively by:

- (i) $depth \ulcorner a \urcorner = 0$ if a is atomic,
- (ii) $depth \ulcorner \neg a \urcorner = depth \ulcorner a \urcorner$, $depth \ulcorner a \vee \beta \urcorner = \max(depth \ulcorner a \urcorner, depth \ulcorner \beta \urcorner)$,
- (iii) $depth \ulcorner \exists x a \urcorner = depth \ulcorner a \urcorner + 1$.

(There are more liberal definitions of depth: cf. [10], p. 113; however, this one is sufficient to illustrate the situation.)

By T_n we will mean the subtheory of T consisting exactly of the axioms of T whose closures have $depth \leq n$. If T is naturally presented by an RE -formula τ of K , then T_n may also be presented naturally. Let $depth(x) = y$ be the natural definition in M for *depth*.

DEFINITION. Where τ is a natural presentation of T , let τ_n denote the presentation formula:

$$\tau[v_0] \& \text{DEF}(\text{depth}(\text{closure}(v_0)) \leq \bar{n}).$$

We will say that T is *depth-reflexive* if each τ_n is in the reflection set for T . We will call proof restriction function *depth-bounding* if its values are presentations $\ulcorner \sigma \urcorner$ such that $T \vdash \text{INT}(\sigma \rightarrow \tau_n)$ for some n (depending on σ).

THEOREM 4.7. *Assume that T , presented by τ , is depth reflexive. Then T does not have a depth-bounding, recursive proof restriction function defined on the universal sentences of N .*

Proof. Immediate from Theorem 4.4.

DEFINITION. The *natural presentation* of P is the formula π denoting:

$$\lambda_Q[v_0] \vee \text{DEF}(\exists x \exists y (x < v_0 \& y < v_0 \& \text{Form}_K(x) \& v_0 = \text{imp}(\text{conj}(\text{sub}(x, \text{var}(y), \text{num}(0)), \text{univ}(\text{var}(y), \text{imp}(x, \text{sub}(x, \text{var}(y), \text{suc}(\text{var}(y)))))), x))).$$

(This is an exact duplication of an induction axiom where x codes $a[v_y]$, the inducted formula.)

COROLLARY 4.8. *Theory P , presented by π , fails to have a proof restriction function of the type indicated by Theorem 4.7.*

This will follow from Theorem 5.8. The result of Theorem 4.7 also applies to various theories axiomatized by principles of transfinite induction or other constructive principles. It does *not*, however, pass so easily to consistent extensions as did Theorem 4.5, as we shall see next.

E. Example of a depth-bounding proof restriction function. We illustrate several things with the following simple example. The theory T will have language K and be the conservative extension of P which is recursively axiomatized, by the device of Craig [2], as follows: a is an axiom of T if it is of the form

$$\beta \& v_m = v_m \quad \text{where} \quad \text{Prf}_P(m, \ulcorner \beta \urcorner)$$

(that is $P \vdash \beta$ with proof code m). T will have the depth-bounding total recursive proof restriction given as follows: $f \ulcorner \beta \urcorner$ codes σ , the presentation

$$\text{DEF}(\exists y (v_0 = \text{conj}(\bar{\beta}, \overline{v_y = v_y}) \& \text{Prf}_\tau(y, \bar{\beta}))).$$

Clearly, f is a total recursive proof restriction function. It is depth-bounding because $M \vdash \sigma \rightarrow \tau_n$ where $f \ulcorner \beta \urcorner = \ulcorner \sigma \urcorner$, $\text{depth} \ulcorner \beta \urcorner = n$, and τ is the natural presentation of T . F_β is even presented recursively. If we required y to code the first proof of β in giving σ , then F_β could have *at most one* axiom (none of course, if β were not a theorem of P). So even though each F_β will be finite we could not list it naturally by a λ presentation (we cannot guess which, if any, y).

Furthermore the situations for T concerning global and local axiomatization are different. T cannot be globally axiomatized by axioms of bounded depth (e.g. apply [10], p. 108, Theorem 4). However, as we have seen, it is locally axiomatized with bounded depth.

Chapter 5

TRUTH DEFINITIONS AND FORMALIZED CONSISTENCY PROOFS

In this chapter we digress to summarize some classical results concerning truth definitions and formalized consistency proofs. These results illuminate the range of application of our conclusions of the last chapter and also those of the next chapter. (We have avoided the discussion of reflection principles, e.g. [10], although in more complicated applications they would provide a convenient intermediary between the truth definitions and the consistency proofs.) Formalized truth definitions are patterned after the Tarski [20] definition of truth, except that they are *formalized* in the sense that they speak about the *code* numbers of formulas (this automatically implies that they cannot hold for *all* formulas of a language). In set theory we can conveniently formalize the usual truth definition for any structure which is a *set*. On the other hand, we cannot speak about infinite structures in number theory. Never-the-less, Mostowski [15] discovered how to circumvent the formalization of structures and still obtain a formal truth definition⁽³⁾. Here we will use a notion of truth definition which permits both applications. As truth definitions provide a connection between formulas and their code numbers, they often yield formalized consistency proofs.

A. Truth definitions. In much of the following we will often omit the abbreviating operator INT to simplify notation. No confusion should result as it is obvious where it should be placed (i.e. in front of symbols and formulas of K or M). We are concerned here with subtheories of T , but the notions given generally do not require this.

DEFINITION. Assume L interprets K and let Ψ be a class (usually effective) of sentences of L . We call a formula $v[x]$ of L a *truth definition for Ψ with basis T* in case for all α in Ψ we have

$$(i) \quad T \vdash \alpha \leftrightarrow v[\bar{\alpha}].$$

DEFINITION. Suppose that $v[x]$ satisfies the above and also that $\psi[x]$ is a natural definition for (the code numbers of sentences of) Ψ ,

⁽³⁾ Mostowski [15], p. 147 refers to these as "models of the third kind".

given in M or L . Then we will call v *standard in T* in case the following three conditions are satisfied:

- (ii) $T \vdash \psi[x] \rightarrow (\psi[\text{neg}(x)] \& (v[\text{neg}(x)] \leftrightarrow \neg v[x]))$,
- (iii) $T \vdash (\psi[x] \& \psi[y]) \rightarrow (\psi[\text{disj}(x, y)] \& (v[\text{disj}(x, y)] \leftrightarrow v[x] \vee v[y]))$,
- (iv) $T \vdash (\psi[x] \& \text{Prv}_{\text{Log}}(x)) \rightarrow v[x]$.

DEFINITION. Suppose that T and $v[x]$ satisfy all of the above. Furthermore suppose that theory S , presented by σ , is such that (the closures of) all of the axioms of S are in \mathcal{V} . We say that T *models σ with v* in case:

- (v) $T \vdash \sigma[x] \rightarrow v[\text{closure}(x)]$.

If (i) holds, the condition (v) automatically implies that S is a subtheory of T . Conversely, if S is a *finite* subtheory of T presented by λ_S , then condition (i) implies (v).

Below we will use the natural definition in M : $\text{conj}_\sigma(y) = z$ (φ_σ is the y th conjunction: $\alpha_0 \& \dots \& \alpha_{y-1}$ of axioms presented by σ).

THEOREM 5.1. *Assume T interprets P and that S is a subtheory of T presented by σ . If T models σ with v then $T \vdash \text{INT}(\text{Con}_{L,\sigma})$.*

Proof. By (v) and the definition of conj_σ we get

$$(1) \quad T \vdash v[\text{conj}_\sigma(y)].$$

Assuming that $0 = 0$ is in \mathcal{V} (although this is not important), we apply (i) and (ii) to get

$$(2) \quad T \vdash 0 = 0 \leftrightarrow v[\overline{0 = 0}] \quad \text{and so} \quad T \vdash \neg v[\overline{0 \neq 0}].$$

Using the propositional rules (ii) and (iii), we combine (1) and (2) to yield

$$(3) \quad T \vdash \neg v[\text{imp}(\text{conj}_\sigma(y), \overline{0 \neq 0})].$$

By (iv) this yields

$$(4) \quad T \vdash \neg \text{Prv}_{\text{Log}}(\text{imp}(\text{conj}_\sigma(y), \overline{0 \neq 0})).$$

Now in M we can *formalize* the Deduction Theorem ([18], p. 33)

$$(5) \quad M \vdash \text{Prv}_{L,\sigma}(x) \rightarrow \exists y \text{Prv}_{\text{Log}}(\text{imp}(\text{conj}_\sigma(y), x)).$$

Thus, (4) and (5) yield

$$T \vdash \neg \text{Prv}_{L,\sigma}(\overline{0 \neq 0})$$

as sought. ■

B. Set theory. We first summarize an application to set theory as these are the most natural. A convenient source of further applications is Lévy [12]. We can define and formalize model theoretic notions in set theory. Thus, for example, in a definitional extension ZF' of Zermelo–Fraenkel set theory we can formalize “structure \mathcal{A} satisfies α ”

by the definition $\text{sat}(a, \bar{a})$ in a natural manner (here a would be the defined constant for the set which is the structure \mathcal{A}).

LEMMA 5.2 (Montague [14]; Reflection Principle for Set Theory). *Let S be a finite set of sentences of the language of ZF . Then there is a defined constant a such that $ZF' \vdash \alpha \leftrightarrow \text{sat}(a, \bar{a})$ for every α in S .*

Sketch. An analog of the Löwenheim–Skolem theorem gives us the constant a such that $ZF' \vdash \alpha \leftrightarrow \text{REL}_a(\alpha)$ for each α in S (where REL_a is the abbreviating operator which relativizes α to a : analogously to our Chapter 1–H). Because sat is natural, we can easily (but tediously) establish $ZF' \vdash \text{REL}_a(\beta) \leftrightarrow \text{sat}(a, \bar{\beta})$ for any β . ■

Now let T be an extension of ZF with the same language. Let S be a finite set of theorems of T . Let λ_S list the codes of sentences in S , and let a be as in Lemma 5.2. $\psi[x]$ will be a natural definition of the code numbers of the propositional closure of S . Let T' denote the definitional extension of T with the symbols sat , a , etc.

LEMMA 5.3. *Let T be an extension of ZF as indicated, and S a finite set of theorems of T . Then T' models λ_S with $\text{sat}(a, x)$.*

Sketch. We check the conditions (i)–(v) in the definition above. The equivalences of (ii) and (iii) hold for all x (not just for x such that $\psi[x]$) as does the implication of (iv). This is because sat is just the natural formalization of the model theoretic notion. (i) holds for all α in S by Lemma 5.2. But because S is finite (i) implies condition (v). ■

THEOREM 5.4 (Wang [22]; Montague [14]). *If T is an extension of ZF with the same language, then T is finitely reflexive.*

Clear from Theorem 5.1 and Lemma 5.3. This of course extends to definitional extensions.

C. Number theory. Generally, in order to give a truth definition (satisfaction) we proceed inductively once we have defined truth for atomic instances (variable free) in a structure \mathcal{A} . In set theory we can formalize this for any language and its structure \mathcal{A} if \mathcal{A} is a set. This is not generally possible in number theory. However, Mostowski [15] noticed that we could formalize a truth definition for variable-free atomic formulas of K (the language of number theory), where the structure (unmentioned but available in the syntax) was that of the numerals.

A variable-free term of K , of course, represents a number. Now, our original coding was effective. So given the code number of a term we can compute the number that term represents. Such a decoding function is recursive and thus we introduce it in M as a recursive symbol:

$$\text{decode}(x) = y \text{ (} y \text{ is the number denoted by the variable-free term with code number } x \text{).}$$

So for example $\text{decode}(\ulcorner \bar{3} + (\bar{2} \cdot \bar{7}) \urcorner) = 17$.

We now give a family of symbols of \mathcal{M} which are formalized truth definitions for the sentences of \mathcal{K} (compare [11], p. 104):

$$\begin{aligned} V_0(x) \leftrightarrow & \text{Sentence}(x) \ \& \ \text{depth}(x) = 0 \\ & \& \ ((\text{Atomic}(x) \ \& \ \exists y \exists z (y < x \ \& \ z < x \ \& \ \text{Term}(y) \ \& \ \text{Term}(z) \\ & \& \ x = \text{equals}(y, z) \ \& \ \text{decode}(y) = \text{decode}(z))) \\ & \vee \exists y (y < x \ \& \ x = \text{neg}(y) \ \& \ \neg V_0(y)) \\ & \vee \exists y \exists z (y < x \ \& \ z < x \ \& \ x = \text{disj}(y, z) \ \& \ (V_0(y) \vee V_0(z)))) \end{aligned}$$

and for each n individually

$$\begin{aligned} V_{n+1}(x) \leftrightarrow & \text{Sentence}(x) \ \& \ ((\text{depth}(x) \leq \bar{n} \ \& \ V_n(x)) \vee (\text{depth}(x) = \overline{n+1} \\ & \& \ (\exists y \exists z (y < x \ \& \ z < x \ \& \ x = \text{exist}(\text{var}(y), z) \\ & \& \ \exists w (V_n(\text{sub}(z, \text{var}(y), \text{num}(w)))))) \\ & \vee \exists y (y < x \ \& \ x = \text{neg}(y) \ \& \ \neg V_{n+1}(y)) \\ & \vee \exists y \exists z (y < x \ \& \ z < x \ \& \ x = \text{disj}(y, z) \ \& \ (V_{n+1}(y) \vee V_{n+1}(z))))). \end{aligned}$$

Remarks. By Chapter 2-C, each of the V_n is a proper definition in \mathcal{M} : it depends on only *finitely* many of its own previous values. On the other hand, the subscript n may not be moved into the definition because the definition of V_{n+1} depends on *infinitely* many values of V_n . Thus each of the V_n is a separate definition. V_0 is a recursive symbol, but none of the other V_n are even re-symbols. Each of the V_n naturally defines the corresponding “truth set” in the sense that we can prove many *intensional* things⁽⁴⁾ involving the V_n . On the other hand, for $n > 0$, none of the V_n represent the corresponding truth sets, because those sets are not recursive (but rather Δ_{n+1}^0 for the standard structure \mathcal{N}).

We note a technical lemma which will be useful in the next chapter.

LEMMA 5.5. *Let $\alpha[\vec{v}_i]$ be a formula of \mathcal{K} of depth $\leq n$ and such that the \vec{v}_i include all of the free variables of α . Then*

$$\mathbf{R} \vdash \alpha[\vec{v}_i] \leftrightarrow \text{DEF}(V_n(\text{sub}(\bar{\alpha}, \vec{\text{var}}(\vec{v}), \vec{\text{num}}(v_i))))).$$

Thus, the formula $\text{DEF}(V_n(x))$ of \mathcal{K} is a truth definition with basis \mathbf{R} , for the class of sentences of depth $\leq n$.

Proof sketch. The only remarkable thing is that these equivalences can be established in \mathbf{R} . But this is not so remarkable once we notice the following principle: In the lengthy inductive demonstration, whenever we need to establish an implication involving variables, that

⁽⁴⁾ The definitions V_n and conditions (i)–(v) for truth definitions correspond to the “uniform reflection principle”: $\forall x (\text{Prv}_\sigma(\text{sub}(\bar{\alpha}, \text{num}(x))) \rightarrow \alpha[x])$ (cf. [10], p. 104).

implication is *already* present in the *natural* representing formulas for the various definitions (e.g. $\text{DEF}(\text{neg}(x) = y)$ will actually include the condition $x < y$, and similarly for the other cases that subexpressions have smaller code numbers).

Induction is on n , and for each n on the structure of α . For example, when $n = 0$, we treat the four simple atomic cases first (e.g. α is $Sv_0 = v_1$ etc.). Then we build up to more complicated terms (this is tedious as checking the proper natural representing formula for the decoding function, etc.). For $n = k + 1$ consider $\exists v_0 \beta[v_0, \vec{v}_i]$. By hypothesis, the equivalence holds with $\beta[v_0, \vec{v}_i]$ and V_k , so checking the definitional translation of V_{k+1} we obtain the equivalence with $\exists v_0 \beta[v_0, \vec{v}_i]$ and V_{k+1} . ■

It is sometimes overlooked, but Mostowski showed that for any Peano theory T we can give a family of truth definitions V'_n analogous to the V_n . If f is a new function symbol of T then we can still *define* a decoding function for terms of T , using the fact that Skolem functions involving f are definable in T because all induction axioms involving f are present (the new decode' might no longer be recursive, but we do get $T \vdash \text{decode}'(\text{sub}(fv_0, \text{var}(0), \text{num}(v_0))) = fv_0$. A new predicate symbol $W(x, y)$ is easily handled by adding a new case in the definition of V'_0 (similar to $\text{equals}(y, z)$):

$$x = \text{predicatew}(y, z) \ \& \ W(\text{decode}'(x), \text{decode}'(y))$$

(where $\text{predicatew}(y, z)$ defines "code of $W(\varphi_x, \varphi_y)$ " where φ_x, φ_y are terms of T).

LEMMA 5.6 (Kreisel–Wang [11]). *Let T be a Peano theory. Then $\text{DEF}(V'_n(x))$ is a standard truth definition in T for the class of sentences of T of depth $\leq n$.*

Proof. Condition (i) is easy to establish along the lines of 5.5. Conditions (ii) and (iii) are immediate from the definitions V'_n .

To establish (iv) we must first formalize in T the analog of a consequence of Herbrand's Theorem (cf. [18], p. 54) which says: any logical proof of a sentence of depth $\leq n$ may be replaced by a proof of the same such that every formula in the proof has depth $\leq n$. We can do this in T because T extends P and the proof of Herbrand's Theorem is finitary.

We then formalize in T the proof that logical deduction for formulas of depth $\leq n$, preserves validity (according to V'_n). And this is condition (iv) for a standard truth definition. This is established by a natural induction on the length of proofs. However, it is worth remarking that the induction axioms of T used will involve $\text{DEF}(V'_n(x))$ and so will themselves have depth $> n$. ■

THEOREM 5.7 (Mostowski [15]). *Every Peano theory is finitely reflexive.*

Proof. For subtheory S , presented by λ_S , let n be the maximum depth of axioms of S . So $T \vdash \text{DEF}(V'_n(\bar{a}))$ for each axiom a of S (using only condition (i) of Lemma 5.6). This gives us condition (v), and we have the other conditions by Lemma 5.6. So by Theorem 5.1, $T \vdash \text{INT}(\text{Con}_{\mathcal{L}, \lambda_S})$. ■

So far, condition (v) has been immediate from (i). But this has been only because the subtheories in question are finite. Notice for the example of Chapter 4–E, that (v) would not hold for σ presenting the axioms of T of depth = 0. Otherwise T could establish its own consistency.

THEOREM 5.8 (Gentzen, Ackermann; cf. [19], [11]). *P , presented by π , is depth-reflexive* (for π etc. see Chapter 4–D).

Proof. We consider the subtheory of P of bounded depth which is presented by π_n . In view of Lemma 5.6 we need only establish (v). We argue abstractly, but to simplify notation, a will code the induction axiom on formula with code b . We will also omit the inducted variable of b in writing sub, and univ. As a further simplification, we will suppose that b does not have any other free variables (so that $M \vdash \text{Sentence}(a)$).

- (1) $M \vdash \pi_n(a) \ \& \ \neg V_n(a) \rightarrow V_n(\text{sub}(b, \text{num}(0)))$
 $\ \ \ \ \ \& \ V_n(\text{univ}(\text{imp}(b, \text{sub}(b, \text{suc}(\text{var}(0))))) \ \& \ \neg V_n(\text{univ}(b)),$
- (2) $M \vdash V_n(\text{univ}(b)) \ \& \ V_n(\text{sub}(b, \text{num}(0)))$
 $\ \ \ \ \ \rightarrow \exists y (V_n(\text{sub}(b, \text{num}(y))) \ \& \ \neg V_n(\text{sub}(b, \text{suc}(\text{num}(y)))).$

Here (1) follows from the structure of a and the definition of V_n while (2) follows by induction in M on the definition of V_n . The definition of V_n and (1) and (2) then quickly yield condition (v). So P models π_n with $\text{DEF}(V_n(x))$ and Theorem 5.1 yields the result. ■

Remarks. The introduction of the V_n obscures the constructive nature of the original proofs. Indeed, to establish reflexivity for T we do not need constructive consistency proofs if T embodies convenient nonconstructive principles. Notice that combining the techniques above we see that any extension of P by finitely many axioms is also depth-reflexive (e.g. $P \cup \{\text{DEF}(\text{Con}_n)\}$). This is because the same V_n are used in Lemma 5.5 and Theorem 5.8.

D. A finite axiomatization of P . As a convenient illustration of truth definitions we adapt an example by Kleene [9] of a finite axiomatization of P with a new symbol. Theory T has language K and also a new binary predicate symbol $W(y, x)$ and the two axioms

$$W(0, x) \leftrightarrow \text{DEF}(\dots \text{right-hand side of definition of } V_0 \text{ in last section} \\ \text{but with } W(0,) \text{ replacing } V_0 \text{ everywhere } \dots),$$

$W(y+1, x) \leftrightarrow \text{DEF}(\dots \text{right-hand side of schema for } V_{n+1} \text{ but with } W(y,) \text{ and } W(y+1,) \text{ respectively replacing } V_n \text{ and } V_{n+1} \text{ everywhere} \dots)$.

We also give T the axioms of Q and the single axiom

$$\text{DEF}(\forall x(\pi[x] \rightarrow \exists y W(y, \text{closure}(x))))).$$

(For π , see Chapter 4-D. DEF removes only the defined symbols of M , and has no effect on W .) Using Lemma 5.5, we see that T extends P . In fact T is a conservative extension of P (this can be shown in various ways). Lemma 5.6 cannot be established in T for $W(y, x)$ with y free because T does not have induction on formulas involving W . As W is a truth definition for K it cannot even be consistently defined in an extension of P with language K .

But induction axioms involving W make sense. Let T' be the extension of T with all of these axioms. T' is a Peano theory. So Theorems 5.7 and 4.5 apply to T' . And thus T' properly extends T , even in the language K .

Chapter 6

PAIR PROOF RESTRICTION FUNCTIONS

For strong theories, an informal interpretation of the results of Chapter 4 is this: It is too much to ask that *each* value of a proof restriction function presents a subtheory which is substantially (and *explicitly* so) more simple than T . However, there might be other proof restriction functions which are “informative”. In this chapter we consider whether we can ask only that each pair $f^{\ulcorner a \urcorner}$ and $f^{\ulcorner \neg a \urcorner}$ present subtheories which are predictably different.

Notation. In this chapter, when $f^{\ulcorner a \urcorner}$ is a presentation code, F_a will always denote the presented subtheory. Note that F_a and S might have identical axioms even though their presenting formulas are not provably equivalent.

A. Related sentences and proof restrictions. Let f be a proposed proof restriction function. If f were produced by studying syntax, it would be reasonable to expect relationships amongst the values of f for structurally related sentences. But we remark that such relationships might not be simple (e.g. in a Boolean sense): Of course, $F_{a \& \beta}$ should be $F_a \cup F_\beta$. But often we must assign a rich set of axioms to a disjunction where we had already confidently assigned no axioms to each of the disjuncts (e.g. consider P presented by π and the disjunction $\text{DEF}(\ulcorner \neg \text{Con}_\pi \urcorner \vee \nu_\pi$ where ν_π is $\text{DEF}(\ulcorner \neg \text{Prv}_\pi(\bar{\nu}_\pi) \urcorner)$). We will not generally consider complex relationships amongst values of f .

Except, suppose that we were attempting to prove a and also $\neg a$ separately, but at the same time. It would seem reasonable that in these two attempts we would tend to use different axioms — that is, sharing “few” axioms in common. In conversation, Ehrenfeucht had wondered whether this could be realized as an “informative” proof restriction function. It is rather sensitive how we make this question precise. To see how we might do this, we first ask: Can we produce a recursive proof restriction function f so that F_a and $F_{\neg a}$ have an “essentially finite” intersection? (Because the axioms of T are usually not independent, by “essentially finite” intersection, we mean that there is a finite subtheory U_a of T such that whenever $F_a \vdash \beta$ and $F_{\neg a} \vdash \beta$, then $U_a \vdash \beta$.) This

question has an affirmative answer, but by means of an uninformative proof restriction function. We give an example:

THEOREM 6.1. *Assuming \mathbf{P} is consistent, it has a recursive proof restriction function f whose values represent recursive subtheories of \mathbf{P} and such that for each α , \mathcal{Q} is identical to at least one of F_α and $F_{\neg\alpha}$.*

Proof. We need only a minor modification of Theorem 3.1. F_α always includes \mathcal{Q} . In addition, if $\mathbf{P} \vdash \alpha$ where p codes the first proof, and $\beta[0] \& \forall x(\beta[x] \rightarrow \beta[Sx]) \rightarrow \beta[x]$ is an axiom used in that proof, then we give F_α the axiom $\gamma[0] \& \forall x(\gamma[x] \rightarrow \gamma[Sx]) \rightarrow \gamma[x]$ where $\gamma[x]$ is $\beta[x] \& \forall v_p(v_p = v_p)$. These are the only axioms given F_α , and so F_α is recursive. By the consistency of \mathbf{P} , at most one of any pair F_α and $F_{\neg\alpha}$ can have axioms other than those of \mathcal{Q} . ■

Remark. If α is neither provable nor refutable in \mathbf{P} , then F_α and $F_{\neg\alpha}$ are both identical to \mathcal{Q} . However, we could not force a proof restriction function to be more informative simply by demanding that F_α and $F_{\neg\alpha}$ have different theorem sets. To see this, modify 6.1 as follows: Let φ and ψ be two axioms of \mathbf{P} which are independent of each other with respect to \mathcal{Q} . Modify f by always assigning φ to F_α and ψ to $F_{\neg\alpha}$ (identifying $\neg\neg\alpha$ with α , etc.). Then F_α and $F_{\neg\alpha}$ will always have different sets of theorems.

B. Explicitly distinct subtheories. The proof restriction function of Theorem 6.1 is uninformative because it tells us nothing about reasonable attempts to prove α short of finding a proof of α . An informative feature of both listing and depth-bounding functions is that they would require some definite *commitment* which could not always depend on finding a proof. Thus, we might hope to obtain an informative distinction between F_α and $F_{\neg\alpha}$ by asking for some definite, explicit way in which they differ. Such a criterion which we will consider is the following:

(ED 1): Name one theorem δ_α by which F_α and $F_{\neg\alpha}$ differ: that is, δ_α is provable in exactly one of them. (Contrast this with the remark following Theorem 6.1.)

Furthermore, we will generally ask for two additional reasonable conditions:

(ED 2): This distinction should be in the presence of the numerical equalities and inequalities, \mathbf{R}_N (that is, δ_α is provable in exactly one of $F_\alpha \cup \mathbf{R}_N$ or $F_{\neg\alpha} \cup \mathbf{R}_N$); and

(ED 3): The distinguishing sentence δ_α should be "relatively simple".

We will now indicate why it is reasonable to demand (ED 2); in section D of this chapter we will consider (ED 3):

EXAMPLE 6.2. *Assume that \mathbf{T} interprets \mathbf{R} and that the theorems of \mathbf{T} are recursively enumerable. Then we may give \mathbf{T} a recursive proof restriction function f , such that $0 \neq \bar{1}$ is provable in exactly one of each F_α and $F_{\neg\alpha}$.*

Demonstration. Let \mathcal{A} be a fixed one element structure for L (so that \mathcal{A} satisfies $0 = \bar{1}$). If \mathcal{A} satisfies a , let F_a be assigned $a \ \& \ v_k = v_k$ where a is the k th theorem of T in the fixed listing (so it will have no axioms if a is never listed). If \mathcal{A} satisfies $\neg a$ then to F_a we assign $a \ \& \ v_k = v_k$ as before, but we also assign $0 \neq \bar{1}$. Then each F_a is recursive, since satisfaction in a finite structure is effectively determined.

Now suppose that $T \vdash a$ and \mathcal{A} satisfies a . Then $F_a \vdash 0 \neq \bar{1}$ by construction, but $0 \neq \bar{1}$ is not a theorem of F_a : If it were, then $\text{Log} \vdash a \rightarrow 0 \neq \bar{1}$, but this is not so because $0 \neq \bar{1}$ fails in \mathcal{A} . The other cases are similar. ■

Similarly if we asked each F_a and $F_{\neg a}$ to be distinguished in the presence of some finite number of numerical identities (axioms of R_N), these could all be satisfied in some finite structure \mathcal{A} where some other identity which failed in \mathcal{A} could be the distinguishing sentence. Thus, to force a significant distinction between F_a and $F_{\neg a}$ we should ask for condition (ED 2). However, our later results will have a greater range of applications if we ask for a somewhat weaker condition. This weaker condition rules out a pathology of the axiomatization of the F_a in the above example: Suppose that in the example above, T interprets Q and β is the conjunction of the axioms of Q_N . Let γ be $0 \neq \bar{1} \rightarrow \beta$, so F_γ is equivalent to $\{\gamma\}$ because $0 \neq \bar{1}$ fails in \mathcal{A} . But then F_γ fails to extend R_N , but it is deficient in doing this only by *finitely many* (one) axiom of R_N . We wish to prevent this rather unnatural situation.

DEFINITION. Here S is a theory with language L (L interprets K). We say that S satisfies the *R-extension condition* provided either S extends R_N , or else $S \cup \{\varrho\}$ does not extend R_N where ϱ is a conjunction of axioms of R_N .

DEFINITION. Subtheories S_1 and S_2 of T are *explicitly distinguished by sentence δ* (of L) in case δ is provable in exactly one of S_1 or S_2 .

Notation. In the following f and d will always be functions of the appropriate type: f is from sentence codes to presentation codes, and d is from sentence codes to sentence codes. The sentence coded by $d^{\ulcorner a \urcorner}$ will be indicated by δ_a (or simply δ where a is understood).

DEFINITION. We say f, d is a *pair proof restriction function* (for some class Φ of sentences) for T in case the following hold:

(PP 1) f is a proof restriction function (for Φ and T);

(PP 2) for each a , F_a and $F_{\neg a}$ are explicitly distinguished by δ_a (where $d^{\ulcorner a \urcorner} = d^{\ulcorner \neg a \urcorner}$); and

(PP 3) each F_a satisfies the *R-extension condition*. Furthermore, we say f, d is recursive if both f and d are recursive; we omit Φ if Φ is the class of all sentences of L .

If (ED 2) held instead of (PP 2) and (PP 3) we could clearly convert the situation into one in which these held. The utility of using the weaker (PP 2) and (PP 3) is that they apply to many familiar axiomatizations of theories (e.g. each subset of the given axioms of \mathcal{Q} or \mathcal{P} satisfies the \mathcal{R} -extension condition).

It is clear from the remark following Theorem 6.1, that we should ask that d be recursive. We consider where this demand could be met. Consider \mathcal{P} or any weakly ω -consistent theory. Let Γ^* be the class of existential unbounded sentences and (their negations) the universal unbounded sentences. Then \mathcal{P} has a recursive pair distinguishing proof restriction function for Γ^* : If α is existential unbounded give F_α the axioms of \mathcal{R} and $F_{\neg\alpha}$ all the axioms of \mathcal{P} and let d be constantly $\forall x(x \neq Sx)$ (which is not a theorem of \mathcal{R}). Then f, d have the desired properties upon noting Theorem 3.2 (iv). Thus, for weakly ω -consistent theories, a pair proof restriction would become interesting if it extended to more complex sentences than those in Γ^* . It is illuminating to consider how one might try to assign subtheories to an unsettled pair of sentences outside Γ^* (e.g. the twin prime question). Also notice that such an assignment should induce an assignment (better than f above) for their instances which are in Γ^* .

Although it is not immediately obvious, if f, d is a recursive pair proof restriction function for \mathcal{T} , then \mathcal{T} must be consistent. This is because d induces recursive divisions of the sentences of \mathcal{L} (by *division* we mean simply a decomposition into a recursive set and its complement, as opposed to *separation* in the sense of Chapter 2-F).

LEMMA 6.3. *Assume f and d are recursive and conditions (PP 2) and (PP 3) hold. Let e be a recursive function from RE-sentences of \mathcal{N} to sentences of \mathcal{L} . Then e (along with f, d) induces a recursive division $E, \neg E$ of the class Γ^{RE} : where γ is a RE-sentence and $e \ulcorner \gamma \urcorner = \ulcorner \alpha \urcorner$, put $\ulcorner \gamma \urcorner$ in E and $\ulcorner \neg \gamma \urcorner$ in $\neg E$ iff $F_\alpha \vdash \delta_\alpha$ (and thus the reverse if $F_{\neg\alpha} \vdash \delta_\alpha$).*

Proof. The procedure is obvious. Both E and $\neg E$ are recursive. Note that f does not need to be a proof restriction function.

COROLLARY 6.4. *If f, d is a recursive pair proof restriction function for \mathcal{T} , then \mathcal{T} is consistent.*

Proof. Let β denote the conjunction of the axioms of \mathcal{Q}_N , and let $e(x)$ be the function $\text{conj}(\ulcorner \beta \urcorner, \text{neg}(x))$ so that $e \ulcorner \gamma \urcorner = \ulcorner \beta \& \neg \gamma \urcorner$.

Suppose that \mathcal{T} were inconsistent and $\mathcal{R}_N \vdash \gamma$ where γ is an RE-sentence. Then also $\mathcal{T} \vdash \beta \& \neg \gamma$ and so $F_{\beta \& \neg \gamma} \vdash \beta \& \neg \gamma$, because f is a proof restriction function. So even $F_{\beta \& \neg \gamma}$ is inconsistent and thus proves δ . Thus $\ulcorner \gamma \urcorner$ is in E and $\ulcorner \neg \gamma \urcorner$ is in $\neg E$ as specified by the lemma. By a similar argument, the situation is reversed if $\mathcal{R}_N \vdash \neg \gamma$.

But then $E, \neg E$ is a recursive separation of Γ^{RE} for the theory R_N which is impossible by Theorem 2.5. ■

We will consider less trivial division of Γ^{RE} as allowed by the lemma.

LEMMA 6.5. *Assume T interprets Q and that f, d is a recursive pair proof restriction function for T . Assume also that $v[x]$ is a truth definition for the range of d with finite basis B (a subtheory of T).*

Then given a sentence γ of N we may effectively construct a sentence ε of L with the following property: If $R_N \vdash \gamma$ then $F_\varepsilon \vdash \delta_\varepsilon$; and if $R_N \vdash \neg \gamma$ then $F_{\neg\varepsilon} \vdash \delta_\varepsilon$.

We may code this construction by a recursive function e (such that $e \ulcorner \gamma \urcorner = \ulcorner \varepsilon \urcorner$).

Proof. Let β be the conjunction of the axioms in $B \cup Q_N$ (so $T \vdash \beta$). By Theorem 2.4 (i) (Lemma of Self-Reference) we obtain ε effectively (given γ) such that

$$R_N \vdash \varepsilon \leftrightarrow (\gamma \leftrightarrow (\beta \ \& \ v[d(\bar{\varepsilon})])).$$

(Precisely, by $v[d(\bar{\varepsilon})]$ we mean $\exists y(\text{INT}(d(\bar{\varepsilon}) \simeq y) \ \& \ v[y])$ where d, \simeq are the re-symbol for d in M ; however we will write symbols of M in L for convenience.)

Notice that for all $a, T \vdash \delta_a$ by assumption on d ; and furthermore, $T \vdash v[d(\bar{a})]$ because v is a truth definition which includes the range of d .

Now assume that $R_N \vdash \gamma$. Then $T \vdash \gamma \ \& \ \beta \ \& \ v[d(\bar{\varepsilon})]$ and so $T \vdash \varepsilon$. So by assumption on $f, F_\varepsilon \vdash \varepsilon$. But then $F_\varepsilon \cup R_N \vdash \beta \ \& \ v[d(\bar{\varepsilon})]$ which means that $F_\varepsilon \cup \{\varrho\} \vdash \beta$ where ϱ is a finite conjunction of axioms of R_N and β already implies all of R_N because it includes the axioms of Q_N . So $F_\varepsilon \cup \{\varrho\}$ extends R_N and so F_ε itself extends R_N because it satisfies the R -extension condition. So $F_\varepsilon \vdash \beta \ \& \ v[d(\bar{\varepsilon})]$ and thus $F_\varepsilon \vdash \delta_\varepsilon$ because β axiomatizes B , the basis for the truth definition.

If $R_N \vdash \neg \gamma$, the argument is analogous noting that

$$R_N \vdash \neg \varepsilon \leftrightarrow (\neg \gamma \leftrightarrow (\beta \ \& \ v[d(\bar{\varepsilon})])). \quad \blacksquare$$

We now give our main theorem. The condition that T interpret Q should not obscure the fact that in most applications, T will be very much stronger than Q .

THEOREM 6.6. *Assume that T interprets Q and that T has a finitely based truth definition for a class of sentences Ψ . Then T does not have any recursive pair proof restriction function f, d where the range of d is in Ψ .*

Proof. Suppose there were such f, d and Ψ . Let v be the truth definition with finite basis B . Then take e as in Lemma 6.5. e is a recursive

function defined on RE -sentences of N (and more). According to Lemma 6.3, e induces a recursive division of I^{RE} :

$$\begin{array}{l} \ulcorner \gamma \urcorner \text{ is in } E, \text{ and} \\ \ulcorner \neg \gamma \urcorner \text{ is in } \neg E \end{array} \quad \text{iff } F_e \vdash \delta_e$$

(where $e \ulcorner \gamma \urcorner = \ulcorner \varepsilon \urcorner$).

Now if $R_N \vdash \gamma$, then $F_e \vdash \delta_e$ by Lemma 6.5, and so $\ulcorner \gamma \urcorner$ is in E . Similarly, the situation is reversed if $R_N \vdash \neg \gamma$. But then $E, \neg E$ is a recursive separation of I^{RE} for R_N , which is impossible by Théorem 2.5. ■

Remarks. Asking in the theorem that the truth definition be finitely based is not a significant restriction. Most of the usual truth definitions for languages with finitely many symbols are finitely based (e.g. by Lemma 5.5, \mathcal{Q} is a basis for the truth definitions V_n of Chapter 5).

Upon examining the proofs of Theorem 2.5 and Lemmas 6.3 and 6.5 we may find a particular pair of sentences $\varepsilon, \neg \varepsilon$ where f, d fails. First obtain the RE -sentence γ by the lemma of self-reference:

$$\begin{aligned} \gamma \leftrightarrow & \text{INT}(\exists p(\text{Prf}_{RE}(f(\text{neg}(e(\bar{\gamma}))); p, d(e(\bar{\gamma}))) \\ & \& \forall q(q < p \rightarrow \neg \text{Prf}_{RE}(f(e(\bar{\gamma})); q, d(e(\bar{\gamma})))))). \end{aligned}$$

Then ε is coded by $e \ulcorner \gamma \urcorner$. The complexity of this ε depends, of course, on B, v , and the complexity of δ_e . In some applications (below) we can see that ε may be fairly simple.

C. Applications. We give only an indication of some of the applications of Theorem 6.6.

Firstly, it can be applied to fairly weak theories (unlike Theorem 4.4). For example, consider \mathcal{Q} . We cannot effectively assign all of the seven axioms to one of each $\alpha, \neg \alpha$ and some six (or fewer) of them to the other. The proof should be clear, noting that these axiom sets satisfy the R -extension condition.

Secondly, Theorem 6.6 permits generalizations of all of the various applications of Theorem 4.4. For those theories, we cannot decide to which of $\alpha, \neg \alpha$ we should assign all of the axioms, while restricting the axiom set of the other (to a finite list, or a depth-bound). In these applications, the distinguishing sentence δ_a would be some $\text{INT}(\text{Con}_{RE}(g(\bar{a})))$ where each $g \ulcorner \alpha \urcorner$ is in the reflection set for T . Furthermore, we may avoid the truth definition of Lemma 6.5 and directly use $\beta \& \text{INT}(\text{Con}_{RE}(g(\bar{\varepsilon})))$ in the construction of ε , where β is the conjunction of the axioms of \mathcal{Q}_N . In these applications, functions f, d would fail at a sentence which is a propositional combination of I^{RE} sentences (as simple as we could hope for).

Thirdly, by using more complex distinguishing sentences we may obtain results which have no analogs in Chapter 4. (Note that the dis-

tinguishing sentences of the paragraph above are \forall^* .) For many theories we can produce a *theorem* ν_n of \mathbf{T} which is not a consequence of any theorems of \mathbf{T} of depth n (cf. [10] and Theorem 6.7 below). For such theories we may apply Theorem 6.6 to see that we cannot decide to which of each α , $\neg\alpha$ we should assign all of the axioms of \mathbf{T} , while assigning to the other all of the *theorems of depth* n (provided n is somewhat less than the depth of α : see Theorem 6.7. However, n need not be fixed using a modification of 6.6.) The set of theorems of depth n is not in the reflection set for \mathbf{T} .

In the above applications of Theorem 6.6, the F_α did not have a local character. But that merely reflects general techniques for distinguishing theories. A more significant intuitive interpretation of Theorem 6.6 is the following: we cannot effectively measure attempts to prove either α or $\neg\alpha$ against a theorem δ (which is more simple than α).

D. Example with complex distinguishing sentences. Condition (ED3) of Section B, above, asked that the distinguishing sentences be “relatively simple.” This condition appears in Theorem 6.6 where we ask that \mathbf{T} has a truth definition for \mathcal{V} which includes the range of d . It seems reasonable that if we could make a distinction between F_α and $F_{\neg\alpha}$ which tells us about attempts to prove α or $\neg\alpha$, then the distinguishing sentence should be simple. We now show that in some instances we may be able to get an uninformative distinction when we relax this demand:

THEOREM 6.7. *There is a theory \mathbf{T} with exactly the same theorems of \mathbf{P} such that \mathbf{T} has a recursive pair proof restriction function.*

Proof. \mathbf{T} is the same theory as in the example of Chapter 4–E. That is, $a \& v_k = v_k$ is an axiom of \mathbf{T} exactly if k codes the first proof of a in \mathbf{P} . Noting the truth definitions of Chapter 5–C, let ν_n be such that:

$$(1) \quad \mathbf{R} \vdash \nu_n \leftrightarrow \text{DEF}(\forall x(V_n(x) \rightarrow \neg \text{Prv}_{\text{Log}}(\text{imp}(x, \bar{v}_n))))$$

(compare [10], Lemma 1). Now ν_n has depth m with $m > n$ (in some constructions of truth definitions, $m = n + 1$, but here it is somewhat larger). Noting Lemma 5.6 and clauses (ii), (iii), and (iv) for a standard truth definition, we have:

$$(2) \quad \mathbf{M} \vdash \exists x(V_n(x) \& \text{Prv}_{\text{Log}}(\text{imp}(x, \bar{v}_n))) \rightarrow V_m(\bar{v}_n).$$

But, combined with (1), this gives us

$$(3) \quad \mathbf{M} \vdash \neg \nu_n \rightarrow V_m(\bar{v}_n), \quad \text{and so} \quad \mathbf{T} \vdash \nu_n.$$

Next designate sentences as either positive or negative depending on the parity of initial \neg symbols (abbreviations removed). Let α be

a positive sentence of depth n . We give F_a and $F_{\neg a}$ recursively as follows (here β denotes the conjunction of the axioms of \mathcal{Q}):

γ is an axiom of F_a iff γ is v_n , or γ is β , or γ is $a \& v_k = v_l$ where k codes the first proof of a in \mathbf{P} ; and

γ is an axiom of $F_{\neg a}$ iff γ is β , or γ is $\neg a \& v_l = v_k$ where l codes the first proof of $\neg a$ in \mathbf{P} .

And we define d by $d \ulcorner a \urcorner = \ulcorner v_n \urcorner$.

To see that v_n explicitly distinguishes F_a and $F_{\neg a}$, we show that it is not a theorem of $F_{\neg a}$: If $\mathbf{P} \vdash \neg a$, then $F_{\neg a}$ is logically equivalent to $\{\beta \& \neg a\}$. And furthermore, since $\neg a$ has depth n , $\mathbf{M} \vdash V_n(\beta \& \neg a)$. Now suppose $F_a \vdash v_n$. Then $\mathbf{Log} \vdash (\beta \& \neg a) \rightarrow v_n$. So $\mathbf{M} \vdash V_n(\beta \& \neg a) \& \text{Prv}_{\mathbf{Log}}(\text{imp}(\beta \& \neg a, v_n))$. But combining this with (1) we get \mathbf{M} (or \mathbf{T}) $\vdash \neg v_n$ which contradicts (3) assuming \mathbf{P} is consistent.

So f, d is a pair proof restriction function for \mathbf{T} . ■

CONCLUSIONS

These results extend easily to applications not developed above. For example, measure the complexity of proofs in finitely axiomatized Gödel–Bernays set theory by the maximum number class constructions upon which an application of the replacement axiom depends. Then by 6.6 we could not restrict one of F_a or $F_{\neg a}$ to an explicit bound in this measure. The applications to **ZF** set theory were more natural because its axioms were “finer” than those of Gödel–Bernays (“finer” with respect to the Tarski–Lindenbaum algebra of that language). On the other hand, examples such as that of 6.7 have axioms which are too fine for convenient application of our Theorems 4.4 and 6.6.

Although we have considered only classical first-order theories, our results can be modified to extend to other finitary deductive systems. Of course, they extend immediately to various deductive type theories as these can be imbedded naturally in first-order systems.

We had accepted at the outset that producing an informative proof restriction function for a strong theory T might require arguments which could not be formalized in T . The results above do not prohibit us from producing proof restriction functions. However, they do suggest that even if we did, we could probably not describe (in an explicit way *expressible* in T) how the F_a should improve on T . (An improvement should relate to proving or refuting a . On the contrary, the distinguishing sentences of 6.7 were chosen to *avoid* saying anything about this.) There are still other possibilities for informative local proof restrictions: one is that a proof restriction function might hold for a large class of “well-behaved” sentences; another is to concentrate not on the explicit strength of the F_a , but rather their efficiency as systems for attempts to prove a .

Bibliography

- [1] A. Church, *An unsolvable problem of elementary number theory*, Amer. J. Math. 58 (1936), pp. 345-363.
- [2] W. Craig, *On axiomatizability within a system*, J. Symbolic Logic 18 (1953), pp. 30-32.
- [3] M. Davis, *An explicit Diophantine definition of the exponential function*, (1971) not yet published.
- [4] M. Davis, H. Putnam, and Julia Robinson, *The decision problem for exponential Diophantine equations*, Ann. of Math. 74 (1961), pp. 425-436.
- [5] A. Ehrenfeucht and S. Feferman, *Representability of recursively enumerable sets in formal theories*, Arch. Math. Logik Grundlagenforsch. 5 (1960), pp. 37-41.
- [6] S. Feferman, *Arithmetization of metamathematics in a general setting*, Fund. Math. 49 (1960), pp. 35-92.
- [7] K. Gödel, *Über formal unentscheidbare Sätze der Principia Mathematica und verwandter Systeme I*, Monatsh. Math. Phys. 38 (1931), pp. 173-198.
- [8] D. Jensen, *Proof restriction functions for infinitely axiomatized theories*, Dissertation, University of Southern California, Los Angeles (1971).
- [9] S. C. Kleene, *Finite axiomatizability of theories in the predicate calculus using additional predicate symbols. Two papers on the predicate calculus*, Mem. Amer. Math. Soc. 10 (1952), pp. 27-68.
- [10] G. Kreisel and A. Lévy, *Reflection principles and their use for establishing the complexity of axiomatic systems*, Zeitschr. Math. Logik 14 (1968), pp. 97-142.
- [11] G. Kreisel and H. Wang, *Some applications of formalized consistency proofs*, Fund. Math. 42 (1955), pp. 101-110; II (Addition to the preceding) Fund. Math. 45 (1958), pp. 334-335.
- [12] A. Lévy, *A hierarchy of formulas in set theory*, Mem. Amer. Math. Soc. 57 (1966).
- [13] Yu. V. Matijasevič, *Diophantine representation of recursively enumerable predicates*, Proceedings of the Second Scandinavian Logic Symposium, Amsterdam 1971.
- [14] R. Montague, *Two contributions to the foundations of set theory*, Logic, Philosophy and Methodology of Science, Proc. 1960 Int. Congress, Stanford (1962), pp. 94-110.
- [15] A. Mostowski, *On models of axiomatic systems*, Fund. Math. 39 (1952), pp. 133-158.
- [16] J. B. Rosser, *Extensions of some theorems of Gödel and Church*, J. Symbolic Logic 1 (1936), pp. 87-91.
- [17] Julia Robinson, *Existential definability in arithmetic*, Trans. Amer. Math. Soc. 72 (1952), pp. 437-449.
- [18] J. Shoenfield, *Mathematical Logic*, Reading 1967.
- [19] M. E. Szabo, ed., *The Collected Papers of Gerhard Gentzen*, Amsterdam 1969.

- [20] A. Tarski, *Der Wahrheitsbegriffe in den formalisierten Sprachen*, *Studia Philos.* 1 (1936), pp. 261–405.
 - [21] A. Tarski, A. Mostowski, and R. Robinson, *Undecidable Theories*, Amsterdam 1953 and 1968.
 - [22] H. Wang, *The non-finitizability of impredicative principles*, *Proc. Nat. Acad. Sci. U. S. A.* 36 (1950), pp. 479–484.
-