C.3

Decidable Theories

MICHAEL O. RABIN

Contents

HANDBOOK OF MATHEMATICAL LOGIC
Edited by J. Barwise

Survey and basic notions

The study of *decidability* involves trying to establish, for a given mathematical *theory* T, or a given problem P, the existence of a decision algorithm AL which will accomplish the following task. Given a sentence A expressed in the language of T, the algorithm AL will determine whether A is true in T, i.e. whether $A \in T$. In the case of a problem P, given an instance I of the problem P, the algorithm AL will produce the correct answer for this instance. Depending on the problem P, the answer may be "yes" or "no", an integer, etc.

If such an algorithm does exist, then we shall variously say that the *decision problem* of T or P is *solvable*, or that the theory T is *decidable*, or simply that the problem P is solvable. Of AL we shall say that it is a *decision procedure* for T or P. Let us illustrate our concepts by two celebrated decidability results.

Let L be a first-order language appropriate for expressing statements about planar Euclidean Geometry. Thus L has individual variables ranging over points; two ternary predicates $\boldsymbol{L}(x, y, z)$ and $\boldsymbol{B}(x, y, z)$ to denote colinearity and betweenness; two predicates $\boldsymbol{C}(x, y, z, u, v, w)$ and $\boldsymbol{A}(x, y, z, u, v, w)$ to denote congruence of triangles and congruence of angles (i.e. $\Delta xyz \cong \Delta uvw$ and $\sphericalangle xyz = \sphericalangle uvw$); and a quaternary predicate $\boldsymbol{E}(x, y, u, v)$ to denote equality of length (i.e. $\overline{xy} = \overline{uv}$). The formula

$$\forall xyzuv\, [\boldsymbol{A}(x, y, v, z, y, v) \wedge \boldsymbol{A}(x, z, u, y, z, u) \wedge \boldsymbol{B}(x, y, u) \wedge \boldsymbol{B}(x, z, v) \wedge \boldsymbol{E}(u, z, v, y) \rightarrow \boldsymbol{E}(x, y, x, z)],$$

for example, is an expression in L of the famous high school problem to the effect that if in a triangle the angle bisectors are equal, then the triangle is isosceles.

Tarski [1951] has proved that the theory EG consisting of all sentences of L true in planar Euclidean Geometry, the so-called elementary geometry, is decidable.

Actually Tarski proved a stronger result. Let $\mathfrak{R} = \langle R, +, \cdot \rangle$ be the field of real numbers, then the first-order theory $\mathrm{Th}(\mathfrak{R})$ is decidable. This last result implies the decidability of elementary geometry via the introduction of Cartesian coordinates and the reduction of geometric statements to equivalent algebraic statements.

Our second example is also related to Euclid. The problem GCD consists of finding for pairs a, b of natural numbers their greatest common divisor (a, b). A slight variant of the famous Euclid's algorithm is based on the facts that $(a,0)=a$ and, for $a \leq b$, $(a,b)=(a,b-a)$. A succession of steps of the second type will transform any g.c.d. (a,b) into $(c,0)$, so that $(a,b)=c$. Thus $(27,15)=(12,15)=(12,3)=(9,3)=(6,3)=(3,3)=(3,0)=3$.

The assignment of a precise mathematical meaning to decidability involves the notion of a computable or recursive function. By an appropriate Gödel numbering G, the set of all sentences of a language L is 1–1 mapped onto a recursive subset $S \subseteq \mathbf{N}$ of the set $\mathbf{N}$ of natural numbers. This transforms T into a set $G(T) \subseteq S$. The theory T is, by definition, decidable if and only if $G(T)$ is a recursive set, i.e. its characteristic function f_T is computable. Solving the decision problem of T involves producing a program or algorithm for computing f_T, or at least proving that f_T is computable. The notion of solvability of a problem P is explicated in a similar fashion.

As explained elsewhere in this volume, the theory T is defined to be *undecidable* if f_T is not recursive.

There is a significant methodological difference between the study of decidability and the study of undecidability, and this despite the obvious fact that the two concepts are just the opposite sides of the same coin. Attempts to establish the undecidability of a theory T must presuppose a mathematically precise notion of computable functions. For only after we know what a computable function is, can we prove that f_T is not computable.

On the other hand, the decision problem of a theory T can be solved by exhibiting a decision algorithm AL which is directly recognized and accepted by mathematicians as being an effective computational procedure. Thus, for example, Euclid's algorithm given above, when supplemented by explicit rules for comparison and subtraction of natural numbers, is universally agreed upon as constituting a computational method.

This state of affairs accounts for the historical fact that some decidability results preceded the definition of recursive functions in the mid-Thirties, whereas the first undecidability results only followed the formulation of this definition.

There are three main methods for establishing decidability of theories. The first and oldest is elimination of quantifiers. This method consists of transforming the given sentence A into another sentence B such that $T \vdash A \leftrightarrow B$ and B belongs to a class $\mathcal{K}$ of sentence for the members of which we can directly determine whether they are in T.

The second method is model-theoretic. In its typical form it involves a (recursive) set of axioms AX for the theory T. Model-theoretic methods are employed to either show that AX is complete, in which case T is readily seen to be decidable, or to systematically survey all completions of AX. In the latter case we shall know for each sentence A whether $T \cup \{\neg A\}$ is consistent and consequently whether $T \vdash A$.

In certain cases a combination of the two methods is used. Namely a set $\mathcal{K}$ of sentences of L is judiciously chosen, the fact that for every sentence A of L there exists a sentence $B \in \mathcal{K}$ equivalent to A by T is then established by model-theoretic means. Also the sentences $B \in \mathcal{K}$ true in T are picked out by a survey of models of T.

The third method involves interpretations. Let T_0 be a theory known to be decidable, and let T be any theory. Assume that we have a (computable) map t which transforms or translates each sentence A of the language L into a sentence $t(A)$ of the language L_0 so that $t(A) \in T_0$ if and only if $A \in T$. Under these conditions T is decidable. For in order to determine whether $A \in T$ we just find $t(A)$ and check whether $t(A) \in T_0$. Usually the interpretation t involves model-theoretic considerations. It is shown that models of T can be isomorphically reproduced from models of T_0 by relations definable in L_0. This method was used in RABIN [1969] to establish most of the then known decidability results as well as several new ones.

The method of interpretations is, mutatis-mutandis, also a powerful tool for proofs of undecidability. For if the theory T is known to be undecidable, and is interpretable in T_0 in the manner of the previous paragraph, then T_0 must be undecidable, see RABIN [1965].

The study of decidability should be viewed as a component, or natural outgrowth, of Hilbert's Program for the foundations of mathematics. Hilbert envisioned a codification of the various branches of mathematics by systems of axioms, with an axiomatized logic serving as a common basis for deduction of consequences (theorems) from the axioms. Hilbert hoped that such a formalization would turn the derivation of mathematical results into a mechanical game with strings of symbols. According to Hilbert's plan, this would give us such a comprehensive survey of all formal theorems within any mathematical discipline, that we would be able to demonstrate that no formal statement and its negation are jointly provable, thereby demonstrating the consistency of mathematics. Also implied by Hilbert's Programme is the belief that the process of theorem-proving is mechanizable or, in modern parlance, that mathematical theories are decidable. Failing to implement Hilbert's plan for mathematics as a whole, by proving the consistency and decidability of, say, set theory, researchers turned their attention to more restricted segments of mathematics. Many of the decidability results which we shall discuss later on, such as Presburger's decision method for the theory of addition of natural numbers, were obtained in the Twenties and early Thirties and were motivated by Hilbert's plan.

Gödel's celebrated Incompleteness Theorems and Church's Undecidability result, dating back to the early and mid-Thirties, dashed the hopes for realization of Hilbert's Programme in its original form. Namely, Gödel demonstrated the impossibility of proving the consistency of any appreciable portion of mathematics by the finitist methods advocated by Hilbert. And Church proved that the predicate calculus, as well as the arithmetic of addition and multiplication of natural numbers, are undecidable. These results put into perspective the study of decidability and engendered a considerable body of research into the decidability and undecidability of various mathematical theories.

Only in recent years attention turned to the issue of the computational complexity of solvable decision problems. In the spirit of Hilbert's Programme and of Turing's analysis of computability, it was tacitly assumed that for a theory T proved decidable, the question whether a given sentence is a theorem of T is a trivial one. For one needs only to mechanically apply the decision procedure in order to answer any such question. No creative or intelligent thinking is required for this process. From this point of view, any decidable theory is trivial and uninteresting. Work of Fischer, Meyer, Rabin, and others has caused a reevaluation of this attitude. They have shown that many theories, even though decidable, are from the practical point of view undecidable because any decision algorithm would require a practically impossible number of computation steps. For the arithmetic of addition of natural numbers, proved decidable by Presburger, Fischer and Rabin [1974] have proved that for every decision algorithm AL there exist sentences A of size (i.e. number of symbols) n such that AL requires 2^{2^n} computational steps to decide A. Meyer [1975] has proved even more devastating complexity results for theories such as the theory of linear-order. Fischer and Rabin have also shown that Elementary Geometry has a decision problem which is inherently of exponential complexity. Results such as these cast doubt on the assertion that any theory proved decidable is trivial because its theorems could be checked by a computer program. Computations involving, say, $2^{2^{30}}$ steps cannot be considered as a feasible method for establishing the truth of a mathematical statement.

Are there *any* theories with a practically solvable decision problem?

Startlingly enough the answer to this fundamental question is as yet unknown. It is readily seen that the decision problem of any formalized theory is at least as complex as the decision problem PC of the propositional calculus. Now Cook [1971] has observed that many combinatorial decision problems which have defied attempts at producing an efficient, not exponentially complex, decision procedure, are reducible to PC. This lends credence to the conjecture that PC is exponentially complex. Cook has related this question, via a use of Turing machines, to the question whether non-deterministic computations requiring a number of steps polynomial in the problem size, are always equivalent to ordinary deterministic computations requiring a polynomial number of steps. This so-called $P = NP$ problem is, as of the time of writing of this paper, one of the central open questions in theoretical computer science. It also relates to the older "spectrum problem" concerning models of sentences of the predicate calculus. Details will be given in the text.

Since the study of decidability involves methods from propositional and predicate logic, theory of computable functions, and theory of models, we shall have to rely on these prerequisites in our exposition. In most cases only the rudiments of these subjects will be required for following discussions. The uninitiated reader is urged to consult the relevant chapters of this book for any auxiliary information that he may need.

1. The method of elimination of quantifiers

1.1. Theories and models

The theories dealt with in the study of decidability will present themselves in one of two ways: *axiomatically* or *semantically*, as the set of sentences true in a structure or class of structures. Usually we shall consider first-order theories, i.e. theories expressible in some first-order predicate language. This rule will, however, have some very important exceptions.

Definition 1. Let L be some fixed first-order language and let H be a recursive consistent set of sentences of L. The *theory axiomatized by H* is, by definition, the set Th(H) of all logical consequences of H

$$\mathrm{Th}(H) = \{A \mid H \vdash A\}.$$

The theory Th(H) and the set of axioms H are called *complete* if for every sentence A of L we have $A \in \mathrm{Th}(H)$ or $\neg A \in \mathrm{Th}(H)$.

THEOREM 1. *If the theory T is axiomatizable and complete then T is decidable.*

PROOF. Let H be an axiomatization of T. The sequences $S_i = (A_{i1}, A_{i2}, \ldots, A_{in_i})$, which are formal proofs from the axioms H, can be effectively enumerated. This can be done, for example, by enumerating *all* finite sequences (words) on the alphabet of L and deleting those sequences which are not proofs. The last members A_{in_i} of the proofs enumerated run through all statements A which are provable from H, i.e. through all elements of $\mathrm{Th}(H) = T$.

Thus, the theorems of T can be effectively (computationally) enumerated in a sequence $S = (A_1, A_2, \ldots)$. Let now A be any sentence of L. Start enumerating S and for each A_n obtained check whether $A_n = A$ or whether $A_n = \neg A$. Since T is complete, one of the two alternatives must eventually occur, at which time we shall know whether $A \in T$ or $A \notin T$. □

People seeing the above argument for the first time often encounter some difficulty in convincing themselves that the proposed procedure is a valid computational process for deciding T. This is because an essential feature of a computation is that we are sure it will terminate, while here when given A we have no a-priori bound on the number of steps required before the computation terminates. However, the fact that T is complete ensures that the computation will terminate by either A or $\neg A$ being encountered in the enumeration. This constitutes a proof of termination of the algorithm. In fact, the number-of-steps function is thereby shown to be a calculable function, albeit not of the commonly encountered variety such as n^n.

The idea underlying Theorem 1 can be extended to cases where the theory T is not complete.

THEOREM 2. *Let T be axiomatizable and assume that there exists a recursive (computable) sequence $A_1, A_2, \ldots,$ of sentences satisfying the following conditions.*

(1) *$T \cup \{A_n\}$ is consistent for every n.*

(2) *Every completion $T \subseteq T_1$ of T has a (not necessarily recursive) set of axioms $B = \{B_1, \ldots, B_k, \ldots\}$ such that $T_1 = \mathrm{Th}(B)$, and for every k there*

exists an n for which $T \vdash A_n \leftrightarrow B_1 \wedge \cdots \wedge B_k$. *Conditions* (1)–(2) *imply that T is decidable.*

PROOF. Let H be a recursive axiomatization for T. By a process resembling dovetailing we can computationally enumerate in one sequence $D_1, D_2, \ldots$, the logical consequences of all the sets $H_n = H \cup \{A_n\}$, $n = 1, 2, \ldots$. Namely, start with an effective enumeration of the consequences of H_1 as in the proof of Theorem 1. When the first consequence $H_1 \vdash D_1$ is encountered, start enumerating consequences of H_2 until the first one, D_2, is obtained. Now return to generate the second consequence, call it D_3, of H_1. Then return to H_2 to obtain $H_2 \vdash D_4$, and next obtain the first consequence, call it D_5, of H_3; and so on.

Again dovetailing, effectively enumerate the above sequence $D_1, \ldots, D_n, \ldots$, and also the sequence $E_1, \ldots, E_n, \ldots$, of all consequences of H. The second sequence is, in fact, an enumeration of $T = \mathrm{Th}(H)$. Thus, if $A \in T$ then A will appear among the E_i's. We claim that if $A \notin T$, then $\neg A$ will appear among the D_i's. Thus the double enumeration will computationally yield an answer to the question whether $A \in T$.

To establish the claim, note that if $A \notin T$, then $T \cup \{\neg A\}$ is consistent. Hence it is a subset of a complete theory $T \cup \{\neg A\} \subseteq T_1$. Let $B = \{B_1, B_2, \ldots\}$ be the set of axioms for T_1 mentioned in the assumptions on the sequence $A_1, A_2, \ldots$. Then $B \vdash \neg A$ and consequently for some integer k, $B_1 \wedge \cdots \wedge B_k \vdash \neg A$. By our assumptions there exits an n so that $T \vdash A_n \leftrightarrow B_1 \wedge \cdots \wedge B_k$. Hence $T \cup \{A_n\} \vdash \neg A$, and $\neg A$ appears in the sequence $D_1, D_2, \ldots$. □

Theorem 2 is used to establish decidability in cases where T is not complete and yet we can somehow survey all possible completions of T.

DEFINITION 2. Let $\mathfrak{A} = \langle A, R_1, R_2, \ldots \rangle$ be a structure and L a first-order language appropriate for $\mathfrak{A}$; that is, L has a symbol P_i corresponding to any relation or function R_i of $\mathfrak{A}$. The *theory* $\mathrm{Th}(\mathfrak{A})$ of $\mathfrak{A}$ is the set of sentences of L true in $\mathfrak{A}$

$$\mathrm{Th}(\mathfrak{A}) = \{B \mid \mathfrak{A} \models B\}.$$

If $\mathscr{K}$ is a class of structures all of the same type, and L is a language appropriate to one and hence all structures in $\mathscr{K}$ then by definition,

$$\mathrm{Th}(\mathscr{K}) = \bigcap_{\mathfrak{A} \in \mathscr{K}} \mathrm{Th}(\mathfrak{A}).$$

We shall now give examples of theories defined by axioms as well as examples of theories defined by models, i.e. defined *semantically.*

EXAMPLE 1. Let L be a language with just one binary predicate symbol $\leq$ (greater or equal than). Consider the set of axioms OR:

1. $$\forall x\, \forall y\, [x \leq y \wedge y \leq x \to x = y],$$

2. $$\forall x\, \forall y\, [x \leq y \vee y \leq x],$$

3. $$\forall x\, \forall y\, \forall z\, [x \leq y \wedge y \leq z \to x \leq z].$$

Any model $\langle A, \leq \rangle \models \mathrm{OR}$ is a totally ordered set.

Let us introduce the abbreviation $x < y$ to stand for $x \leq y \wedge \neg\, x = y$.

EXAMPLE 2. Consider the axioms DO obtained by adding to OR the axioms

4. $$\exists x\, \forall y\, [y \leq x \to x = y],$$

5. $$\forall x\, \exists y\, [x < y \wedge \forall z\, [x < z \to y \leq z]],$$

6. $$\forall x\, \forall y\, [y < x \to \exists z\, \forall w\, [z < x \wedge [z < w \to x \leq w]]].$$

Any model $\langle A, \leq \rangle \models \mathrm{DO}$ is a totally ordered set which has a first element, every element has a unique immediate successor, and every element except the first element has a unique predecessor. Such orders will be called *discrete orders.*

Denoting, as usual, $\omega = \{0, 1, 2, \ldots\}$ we see that $\mathfrak{N} = \langle \omega, \leq \rangle \models \mathrm{DO}$. If we denote by ω^* the reverse order-type of ω (i.e. $0 > 1 > 2 > \cdots$), then every ordered set which is a model of DO has order type $\omega + (\omega^* + \omega)\lambda$, where λ is any order type.

EXAMPLE 3. Consider the class $\mathcal{K}_{\mathrm{uf}}$ of all structures $\langle A, f \rangle$ where f is a unary function $f : A \to A$. $\mathrm{Th}(\mathcal{K}_{\mathrm{uf}})$ is the theory of a unary function.

EXAMPLE 4. Let $\langle \omega, + \rangle$ be the structure of the integers with addition. $\mathrm{Th}(\langle \omega, + \rangle) = \mathrm{PAR}$ will be called Presburger's arithmetic or the theory of addition of natural numbers.

1.2. Elimination of quantifiers for discrete orders

Thus far we have seen examples of axiomatically defined theories and of semantically defined theories. We also gave two principles for establishing the decidability of axiomatized theories. We shall now illustrate the method of elimination of quantifiers by proving the decidability of $\mathrm{Th}(\langle\omega, \leq\rangle)$, a result due to C.H. Langford in 1927.

THEOREM 3. $\mathrm{Th}(\mathfrak{N})$ *is decidable.*

PROOF. Let us enrich the structure $\mathfrak{N}$ by making 0 a distinguished element and adding the successor function $S(x) = x + 1$ (the element next to x in the ordering). Call the resulting structure $\mathfrak{N}_1 = \langle\omega, 0, \leq, S\rangle$ and denote the corresponding language by L_1. We shall decide $\mathrm{Th}(\mathfrak{N}_1)$, from which the decidability of $\mathrm{Th}(\mathfrak{N})$ follows.

Let us use the abbreviation $\boldsymbol{S}^n(t)$ to denote n applications of S to the term t, thus $\boldsymbol{S}^3(\boldsymbol{x}) = \boldsymbol{S}(\boldsymbol{S}(\boldsymbol{S}(\boldsymbol{x})))$. In particular, $\boldsymbol{S}^0(\boldsymbol{y}) = \boldsymbol{y}$. The terms of the language are $\mathbf{0}, \boldsymbol{x}, \boldsymbol{y}, \ldots, \boldsymbol{S}^n(\mathbf{0}), \boldsymbol{S}^n(\boldsymbol{x}), \ldots, 1 \leq n$.

The class $\mathscr{R}$ of formulas to which we shall reduce every formula of L_1 will be the following

(1) $t_1 = t_2$, $t_1 < t_2$, where t_1, t_2 are terms;

(2) formulas which are disjunction of conjunctions of formulas in (1). For example $[\boldsymbol{S}^3(\mathbf{0}) = \boldsymbol{x} \wedge \boldsymbol{S}(\boldsymbol{y}) < \boldsymbol{z}] \vee \boldsymbol{S}^5(\boldsymbol{z}) < \boldsymbol{S}^3(\boldsymbol{y})$ is in $\mathscr{R}$.

We shall show that every formula A of L_1 is equivalent in $\mathfrak{N}_1$ to a formula $B \in \mathscr{R}$. Our proof will also provide a method for effectively transforming A into B.

The statement that two formulas C and D are equivalent in $\mathfrak{N}_1$ means that $\mathfrak{N}_1 \models C \leftrightarrow D$. We shall make assertions concerning equivalence of formulas leaving verification to the reader.

Let A be an open, i.e. quantifier-free formula. Express all other propositional connectives by means of $\vee$, $\wedge$, $\neg$. Move all occurrences of $\neg$ next to the atomic formulas, using rules such as $\neg[C \vee D] \equiv \neg C \wedge \neg D$. Drop all occurrences of double-negation $\neg\neg$. Replace constituents of the form $t_1 \leq t_2$ by $t_1 = t_2 \vee t_1 < t_2$, $\neg t_1 = t_2$ by $t_1 < t_2 \vee t_2 < t_1$, and $\neg t_1 < t_2$ by $t_1 = t_2 \vee t_2 < t_1$. Finally, by use of the distributive law for $\wedge$ and $\vee$, transform the formula into a formula in $\mathscr{R}$. Thus we see that repeated applications of the above rules will transform any open formula into an equivalent formula in $\mathscr{R}$.

Assume now that A_1 has the form $\exists y\,(C_1 \vee \cdots \vee C_n]$ where each C_i is a conjunction of formulas $t_1 = t_2$ or $t_1 < t_2$. We have $A_1 \equiv \exists y C_1 \vee \cdots \vee \exists y C_n$. Thus it suffices to give rules for transforming a formula of the form

$$\begin{aligned} A_1 = \exists \boldsymbol{y}\,[t_1 = t_2 \wedge \cdots \wedge t_{2i-1} = t_{2i} \wedge \\ t_{2i+1} < t_{2i+2} \wedge \cdots \wedge t_{2k-1} < t_{2k}]. \end{aligned}$$

The reader can work out for himself how to treat the case that some equation or inequality in A_1 is of the form $S^m(y) = S^j(y)$ or $S^m(y) < S^j(y)$. Thus we may assume that in each equation or inequality in A_1 at most one side is of the form $S^j(y)$.

For any terms t_1, t_2, and $1 \leq n$, $\mathfrak{N}_1 \models t_1 < t_2 \leftrightarrow S^n(t_1) < S^n(t_2)$ and similarly for $t_1 = t_2$. By applying to both sides of all equations and inequalities in A_1 appropriate powers S^j, we transform A_1 into an equivalent formula in which all occurrences of y are of the form $S^m(y)$, with the same $1 < m$. Let us assume A_1 already has this property. "Eliminate" $\exists y$ from A_1 by: (1) dropping $\exists y$ from A_1; (2) for each conjunct $S^m(y) = t_j$ add a conjunct $S^{m-1}(\mathbf{0}) < t_j$; (3) for each $S^m(y) < t_j$ add $S^m(\mathbf{0}) < t_j$; (4) if any equation $S^m(y) = t_j$ occurs in A_1, replace all occurrences of $S^m(y)$ in A_1 by t_j; (5) assuming that no such equations occur and that all inequalities are of the form $S^m(y) < t_j$, or all are of the form $t_j < S^m(y)$, drop all conjuncts involving $S^m(y)$; (6) lastly, if no equation involving $S^m(y)$ occurs but inequalities of both types do appear, then for every pair $t_j < S^m(y)$ and $S^m(y) < t_p$ add a conjunct $S(t_j) < t_p$, and later drop all conjuncts involving $S^m(y)$. It is clear that steps (1)–(6) transform A_1 into an equivalent formula $B \in \mathcal{R}$ which does not contain y.

Let A be any formula of L_1. Since $\forall x F \equiv \neg \exists x \neg F$, we may assume that A contains just existential quantifiers, say n in number. Let $\exists y D$ be an innermost occurrence of $\exists$, i.e. D is open. Transform D into a disjunction of conjunctions as explained before. Then $\exists y D \equiv A_1$ where A_1 has the form treated above. Distribute $\exists y$ over the disjunctions, and treat each disjunct by steps (1)–(6). By these transformations $\exists y D$ is replaced in A by an equivalent open formula, and A is transformed into an equivalent formula with $n - 1$ quantifiers. Repeating this process n times, A will be effectively transformed into a $B \in \mathcal{R}$.

Finally, if A was a sentence, then the transformed formula is a sentence, hence a propositional combination of formulas $S^m(\mathbf{0}) = S^i(\mathbf{0})$ or $S^m(\mathbf{0}) < S^i(\mathbf{0})$. The truth or falsehood of such a sentence can be directly ascertained. Thus we have a decision procedure for $\mathrm{Th}(\mathfrak{N}_1)$ and hence for $\mathrm{Th}(\mathfrak{N})$. □

Let us observe that we could have avoided the passage from $\mathfrak{N}$ to $\mathfrak{N}_1$, and this because the relation $y = S^n(x)$ is definable by an appropriate formula $D_n(x, y)$ in $\mathfrak{N}$. This enables us to translate all basic formulas $t_1 < t_2$ and $t_1 = t_2$ into formulas of N, and thus get a reduction class of formulas of $\mathfrak{N}$.

Note that if we carry out the quantifier-elimination procedure in $\mathfrak{N}$ then we actually do not get rid of all quantifiers. Rather, we "hide" them in the formulas $D_n(x, y)$. But this still yields the desired results. In general, the elimination of quantifiers method should be construed in this broader sense.

The same decision procedure applies to Th(DO). The only modification required is that the various statements concerning equivalence of formulas which were mathematically verified for $\mathfrak{R}_1$, must now be derived as formal consequences of the axioms DO. Following this route, we not only decide Th(DO) but also show that Th(DO) = Th($\mathfrak{R}$), i.e. Th(DO) is complete.

1.3. Presburger's arithmetic

The theory PAR was decided by PRESBURGER [1929] using the method of elimination of quantifiers. By an appropriate formula $\boldsymbol{x} <_n \boldsymbol{y}$ we can express, for each fixed n, the relation $x < y \wedge x \equiv y \pmod{n}$. Thus, for example, $\boldsymbol{x} <_3 \boldsymbol{y}$ is $\exists \boldsymbol{z}\, [\neg \boldsymbol{z} = \mathbf{0} \wedge \boldsymbol{x} + \boldsymbol{z} + \boldsymbol{z} + \boldsymbol{z} = \boldsymbol{y}]$. Enrich the structure $\langle \omega, + \rangle$ by making 0, 1 into distinguished elements. The terms of the language are now $\mathbf{0}, \mathbf{1}, \boldsymbol{x}, \boldsymbol{y}, \ldots$, and all expressions which are sums of these, e.g. $\boldsymbol{x} + \boldsymbol{z} + \boldsymbol{z} + \mathbf{1} + \mathbf{1} + \mathbf{1}$ abbreviated by $\boldsymbol{x} + 2\boldsymbol{z} + \mathbf{3}$.

The reduction set $\mathcal{K}$ will consist of all formulas obtained from the basic formulas $t_1 = t_2$, $t_1 <_n t_2$ (t_1, t_2 are terms, n is an integer) by conjunctions and disjunctions. The proof, while not trivial, is not too hard and follows the lines of the proof of 1.2.

1.4 Theory of real numbers

This is perhaps the best known application of the elimination of quantifiers method. We consider the field of real numbers $\mathfrak{R} = \langle R, 0, 1, +, \cdot, \leq \rangle$ as an ordered field. Instead of giving Tarski's original decision procedure we shall outline the algorithm of COHEN [1969] using the formulation in Monk's thesis.

We introduce, on a provisional basis, certain algebraic-like functions. Let $P(x_1, \ldots, x_n) \in Z[x_1, \ldots, x_n]$ be a polynomial with integral coefficients, $d_n(P)$ be its degree in x_n, and let $\eta \in R^{n-1}$. Define $P_\eta(x_n) = P(\eta_1, \ldots, \eta_{n-1}, x_n)$. For $1 \leq i \leq d_n(P)$ define $f_{P,i}(\eta)$ to equal 0 if $P_\eta \equiv 0$ or P has no real roots, otherwise the i-th real root of $P_\eta = 0$ if there are at least i such roots, otherwise the largest real root. This makes $f_{P,i}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$ a term which denotes the function $f_{P,i} : R^{n-1} \to R$. Call such terms algebraic functions.

A polynomial relation is a Boolean combination of atomic formulas of the form $0 \leq P$, where $P \in Z[x_1, \ldots, x_n]$. An algebraic relation is a Boolean combination of polynomial relations and formulas of the forms, $\mathbf{0} \leq P[\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}, f_1(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n-1}))$, or $f_1 \leq f_2$, where $P \in Z[x_1, \ldots, x_n]$ and f_1, f_2 are algebraic functions. With each algebraic relation a *rank* is associated in such a way that the rank of a polynomial relation is 0.

The procedure for the elimination of quantifiers will reduce any formula of the original language to an open formula (i.e. polynomial relation) and this by passing temporarily through the more general algebraic relations.

The proof involves two lemmas. The first lemma asserts that a formula $\exists x_n A$, where A is a polynomial relation, is equivalent to an algebraic relation B. The second lemma states that an algebraic relation B of rank $1 \leq k$ is equivalent to an algebraic relation of rank at most $k-1$. This lemma implies by induction that every algebraic relation is equivalent to a polynomial relation. Taken together, these facts ensure that every formula is equivalent to an open formula.

The proofs of the lemmas make use of Rolle's theorem to the effect that between every two consecutive roots of $p(x)=0$ there lies a root of $p'(x)=0$. If $p(x)$ is a polynomial, then the location of the roots of $p(x)=0$ can be determined, with sufficient accuracy for our purposes, from the location of roots of $p'(x)=0$ and the values $p(-\infty)$, $p(+\infty)$. Denote, for $P \in Z[x_1, \ldots, x_n]$, $P' = \partial P/\partial x_n$. It turns out, roughly speaking, that statements involving $f_{P,i}$, i.e. statements about the i-th root of $P=0$, can be transformed into statements involving the terms $f_{P',j}$. Within the framework of our notion of rank, this entails rank reduction which is the key point in the proof of the second lemma.

1.5. Other theories

Let us briefly mention some additional theories which we have shown decidable by the method of elimination of quantifiers.

Let $\mathbf{Q}$ be the rational numbers, $\eta = \langle \mathbf{Q}, \leq \rangle$ their order-type. $\mathrm{Th}(\eta)$ is decidable.

Let ALC be the class of algebraically closed fields, then Th(ALC) is decidable. Here every formula is equivalent to an open formula, a fact that can be established, for example, by employing the classical algebraic elimination theory.

If BA is the class of all Boolean algebras, then Th(BA) is decidable. This result is due to Tarski [1949] and is somewhat difficult.

2. Model theoretic methods

2.1. Categoricity, completeness and decidability

A theory T is called *categorical* in cardinality α if all models $\mathfrak{A} \models T$ of cardinality $c(\mathfrak{A}) = \alpha$ are pairwise isomorphic. By the cardinality of $\mathfrak{A}$ we mean the cardinality of the domain of $\mathfrak{A}$. The following simple observation is due to R. Vaught. It is assumed that T is countable.

THEOREM 4. *If the theory T has no finite models and is categorical in some (necessarily infinite) cardinality α then T is complete.*

PROOF. Assume, by way of contradiction, that T is not complete. Then there exists a sentence S in the language of T such that $T_1 = T \cup \{S\}$ and $T_2 = T \cup \{\neg S\}$ are consistent. Hence there are countable (i.e. finite or denumerable) models $T_1 \models \mathfrak{A}_1$, $T_2 \models \mathfrak{A}_2$. Since T has no finite models, $c(\mathfrak{A}_1) = c(\mathfrak{A}_2) = \omega$. If $\alpha = \omega$, then $\mathfrak{A}_1 \approx \mathfrak{A}_2$, but $\mathfrak{A}_1 \models S$ and $\mathfrak{A}_2 \models \neg S$, a contradiction. Otherwise there exist, by the Skolem–Tarski–Vaught theorem, elementary extensions $\mathfrak{A}_1 < \mathfrak{B}_1$, $\mathfrak{A}_2 < \mathfrak{B}_2$ so that $c(\mathfrak{B}_1) = c(\mathfrak{B}_2) = \alpha$. Again $\mathfrak{B}_1 \approx \mathfrak{B}_2$, $\mathfrak{B}_1 \models S$ and $\mathfrak{B}_2 \models \neg S$. □

The stipulation that T has no finite models is essential. Consider the theory E of just equality $=$. E is categorical in every cardinality. Yet $E \cup \{\forall x\, \forall y\, [x = y]\}$, as well as $E \cup \{\exists x\, \exists y\, [\neg x = y]\}$, are consistent.

Perhaps the simplest application is proving that $\mathrm{Th}(\eta)$ is decidable. Consider the axioms DNO consisting of the axioms for total-order together with

$$\forall x\, \forall y\, \exists z\, [x < y \rightarrow x < z < y]$$

$$\forall x\, \exists y\, \exists z\, [z < x < y].$$

Every model $\langle A, \leq \rangle \models \mathrm{DNO}$ is a totally and densely ordered set without a first or last element. By Cantor's characterization of the order type $\eta = \langle \mathbf{Q}, \leq \rangle$, if $c(A) = \omega$ then $\eta \approx \langle A, \leq \rangle$. Consequently, Th(DNO) is complete and hence, by Theorem 1, decidable. Now $\eta \models \mathrm{DNO}$ so that $\mathrm{Th(DNO)} \subseteq \mathrm{Th}(\eta)$, but Th(DNO) is complete so that Th(DNO) $= \mathrm{Th}(\eta)$. □

2.2. Algebraically closed fields

Next we consider the theory ALC of algebraically closed fields. This theory can be axiomatized in a language L having symbols $\mathbf{0}, \mathbf{1}, +, \cdot$, by writing the usual field axioms and adding a sequence of axioms A_n, $n = 1, 2, \ldots,$

$$A_n = \forall y_0 \cdots y_{n-1}\, \exists x\, [y_0 + y_1 x + \cdots + y_{n-1} x^{n-1} + x^n = \mathbf{0}].$$

The axiom A_n is written using some obvious abbreviations.

The axioms ALC are not complete because the characteristic of the field has not been specified. This can be done by adding, for a prime p, an axiom $C_p = p \cdot \mathbf{1} = \mathbf{0}$, where the left-hand side abbreviates a sum of terms $\mathbf{1}$. To obtain axioms for characteristic 0 put $C_0 = \{\neg C_2, \neg C_3, \ldots\}$.

We shall now show that $T_p = \text{ALC} \cup \{C_p\}$ is complete for any prime p, and $T_0 = \text{ALC} \cup C_0$ is also complete. Let $F \models T_p$, where p is a prime or $p = 0$, be an algebraically-closed field of characteristic p and cardinality $\omega < c(F) = \alpha$. Let $P \subset F$ be the prime-field in F, then $P = Z/pZ$ for $p \neq 0$, and $P = Q$ if $p = 0$. By Steinitz's structure theorem for algebraically-closed fields, there exists a set $X \subseteq F$ of elements algebraically independent over P so that $F \supset P(X) \supset P$ is the algebraic closure of $P(X)$. The isomorphism type of F depends just on P (i.e. p) and $c(X)$, the so-called degree of transcendence of F. Now if $\omega < c(F)$ then $c(X) = c(F) = \alpha$. Hence for all p, T_p is categorical in every non-countable cardinality $\omega < \alpha$.

By use of Theorem 1, this implies that for each $p \geq 0$, the theory T_p of algebraically-closed fields of characteristic P is decidable.

Since $\text{ALC} \cup C_0$, $\text{ALC} \cup \{C_p\}$, p prime, gives an enumeration of all completions of ALC, it follows from Theorem 2 that Th(ALC), the theory of algebraically-closed fields is decidable.

2.3. Real-closed fields

Tarski's result concerning the decidability of the theory of the field of real numbers can also be achieved by model theoretic methods. We first need a set of axioms for the field of real numbers. These were provided by Artin and Schreier in their famous study of real-closed fields.

Consider a language L_1 which, like L of Section 2.2, has $\mathbf{0}, \mathbf{1}, +, \cdot$, but in addition has a greater or equal symbol $\leq$. The axioms RLC consist of the following: the field axioms, axioms stating that $\leq$ is a total order, and furthermore

$$\forall x\, \forall y\, \forall z\, [x \leq y \wedge \mathbf{0} \leq z \rightarrow xz \leq yz],$$

$$\forall x\, \forall y\, \forall z\, [x \leq y \rightarrow x + z \leq y + z],$$

$$\forall x\, \exists y\, [\mathbf{0} \leq x \rightarrow y^2 = x],$$

$$A_n \quad \text{for } n = 1, 3, 5, \ldots .$$

Here, as in 2.2, A_n is the statement that the n-th degree polynomial equation has a root.

The field of real numbers is a model of RLC but by no means the only model. Any ordered field $\langle F, 0, 1, +, \cdot, \leq \rangle \models \text{RLC}$ will be called (*ordered*) *real-closed*. Th(RLC) is not categorical in any power, so that Vaught's test cannot be used. Completeness, and consequently decidability, are proved by means of Robinson's concept of model completeness.

A theory T is *model complete* if for any two models $T \models \mathfrak{A}$, $T \models \mathfrak{B}$ such that $\mathfrak{A} \subseteq \mathfrak{B}$, we have $\mathfrak{A} < \mathfrak{B}$ ($\mathfrak{B}$ is an elementary extension of $\mathfrak{A}$).

ROBINSON [1956] gave a test for model completeness. From this test the model completeness of RLC can be deduced. The proof, while not very hard, does require some effort. Alternatively one can use the following test.

T is model complete if for any two models $\mathfrak{A} \subseteq \mathfrak{B}$ of T there exists an elementary extension $\mathfrak{A} < \mathfrak{C}$ such that $\mathfrak{B} \subseteq \mathfrak{C}$. Combining this with algebraic properties of real-closed fields and with the method of ultraproducts we can get a somewhat different proof for the model completeness of Th(RLC).

Now a model complete theory T need not be complete. However, if T is model complete and has a *prime model* P which is, up to isomorphism, included as a submodel in every model of T, then T is complete. It is readily seen that under these conditions every two models of T are elementarily equivalent.

The theory Th(RLC) has a prime model. Namely, every real-closed field F is of characteristic 0 and hence contains the field Q of rational numbers. It therefore also contains an isomorphic copy of the field of real-algebraic numbers and this is the common prime field. Consequently Th(RLC) is complete and decidable. The field $\mathfrak{R}$ of real numbers satisfies $\mathfrak{R} \models$ RLC, hence Th($\mathfrak{R}$) = Th(RLC) and is decidable.

It should be remarked that this approach to the decidability of the field of real numbers is, despite the difference in methods, not too different from the classical elimination of quantifiers method. At the bottom of the proof of model completeness lies the fact that if two ordered fields F_1, F_2 are isomorphic (the mapping preserves also the order) then their real-closures are isomorphic. This is proved by using information concerning the location of roots of equations. The analysis involved is not too different from the examination of the location of roots in the elimination of quantifiers method. On the other hand, one can claim that the uniqueness of the real-closure is a basic algebraic result established on its own right. In the model-theoretic proof of the decidability of Th(RLC) we are thus quoting a standard result, and from this point of view the proof is more elementary.

2.4. Theory of DO revisited

By way of illustrating how model-theoretic methods are useful for establishing decidability even in the absence of categoricity, let us reexamine Th(DO).

In 1.1, Example 2, we observed that every model of DO has the order-type $\omega + (\omega^* + \omega)\lambda$. We shall show that every countable model

$\mathfrak{A} = \{A, \le\rangle \models \mathrm{DO}$ has an elementary extension $\mathfrak{A} \prec \mathfrak{B}$ for which the λ is η. This will imply $\mathrm{Th}(\mathfrak{A}) = \mathrm{Th}(\mathfrak{B})$, hence every two countable models of DO are elementarily equivalent, and hence Th(DO) is complete and decidable.

Define an equivalence relation E on any model $\mathfrak{A} \models \mathrm{DO}$. $xEy \equiv c(\{z \mid x < z < y \vee y < z < x\}) < \omega$, i.e. the number of elements between x and y is finite. We see that the equivalence classes with respect to E are the "blocks" ω, and each $\omega^* + \omega$, in the order type $\omega + (\omega^* + \omega)\lambda$ of $\mathfrak{A}$. Consider any two blocks (equivalence classes) B_1, B_2 of $\mathfrak{A}$, without loss of generality let $x < y$ for all $x \in B_1$, $y \in B_2$. It is consistent with all the elementary statements about $\mathfrak{A}$, i.e. with the complete diagram $\mathrm{CD}(\mathfrak{A})$ of $\mathfrak{A}$, to assume the existence of an element c such that $x < c < y$ for all $x \in B_1$, $y \in B_2$. Thus $\mathrm{CD}(\mathfrak{A})$ and all these inequalities have a countable model $\mathfrak{A}_1$ which is an elementary extension $\mathfrak{A} \prec \mathfrak{A}_1$ of $\mathfrak{A}$. In this $\mathfrak{A}_1$ the block B of c lies between B_1 and B_2. Similarly we can construct an extension with a block above B_2. Because the extension $\mathfrak{A} \prec \mathfrak{A}_1$ is countable, it is possible to construct a tower of elementary extensions $\mathfrak{A} \prec \mathfrak{A}_1 \prec \cdots$, so that each $\mathfrak{A}_n$ is countable and for each pair B_1, B_2 of blocks of each $\mathfrak{A}_n$ there exists an $n < k$ and a block B of $\mathfrak{A}_k$ situated between B_1 and B_2, and similarly for a block above B_2. Let $\mathfrak{B} = \bigcup_{n<\omega} \mathfrak{A}_n$, then $\mathfrak{A} \prec \mathfrak{B}$, $\mathfrak{B}$ is countable, and the blocks of $\mathfrak{B}$ are densely ordered. Thus the order type of $\mathfrak{B}$ is $\omega + (\omega^* + \omega)\eta$, which completes the proof.

2.5. Further results

Many additional important results we obtained by model-theoretic methods, sometimes in combination with the method of elimination of quantifiers. Let us mention without proofs a few outstanding theorems. The proofs usually involve deep mathematical results concerning the structures in question, making for an interesting combination of standard mathematics and logic.

THEOREM 5 (AX and KOCHEN [1965a, 1965b, 1966]). *The theory of p-adic fields is decidable.*

This result laid to rest a long-standing conjecture that the only decidable fields are the real-closed and the algebraically closed fields.

THEOREM 6 (AX [1968]). *The theory of the class of all finite fields is decidable.*

THEOREM 7 (SZMIELEW [1954]). *The theory of commutative groups is decidable.*

THEOREM 8 (EHRENFEUCHT [1959]). *The theory of linearly* (*totally*) *ordered sets is decidable.*

This result was announced in an abstract. The first full proof to be published is due to LÄUCHLI and LEONARD [1966]. As will be seen in the next chapter, this result is an easy consequence of the method in RABIN [1969].

Finally we shall quote a special case of more general results relating models to decidability.

Let $\mathfrak{A} = \langle A, f_0, \ldots, f_i, \ldots \rangle_{i<\alpha}$ and $\mathfrak{B} = \langle B, g_0, \ldots, g_i, \ldots \rangle_{i<\alpha}$, $\alpha \leq \omega$, be similar algebras of the same type, i.e. f_i and g_i are n_i-ary operations on $\mathfrak{A}$ and $\mathfrak{B}$ respectively. The direct product $\mathfrak{A} \times \mathfrak{B}$ is defined in the obvious manner.

THEOREM 9. *If* $\mathrm{Th}(\mathfrak{A})$ *and* $\mathrm{Th}(\mathfrak{B})$ *are decidable so is* $\mathrm{Th}(\mathfrak{A} \times \mathfrak{B})$.

This is but a special case of a general study of the first-order properties of products of structures and classes of structures initiated by MOSTOWSKI [1952] and developed by FEFERMAN and VAUGHT [1959]. We have restricted ourselves to a very special case in order to avoid the elaborate definitions appearing in the general theory.

The method of products is a powerful tool for obtaining new decidability results from known ones. The following example is due to Mostowski. We know that $\mathrm{Th}(\langle \omega, + \rangle)$ is decidable, see Section 1.3. From the theory of general products it follows that the direct sum $\mathfrak{P}$ of ω copies of $\langle \omega, + \rangle$ has a decidable theory. The domain of P consists of all ω-length vectors

$$v = (n_0, \ldots, n_k, 0, 0, \ldots), \qquad n_i \in \omega, \quad k = 0, 1, \ldots,$$

and the operation is component-wise addition. Define a mapping $\phi(v) = p_0^{n_0} \cdots p_k^{n_k}$, where p_i is the $(i+1)$-th prime. The mapping ϕ is an isomorphism $\phi : \mathfrak{P} \to \langle \omega - \{0\}, \cdot \rangle = \mathfrak{M}$ onto the multiplicative semi-group of integers. Hence $\mathrm{Th}(\mathfrak{M})$ is decidable, a result due to Skolem.

3. The method of interpretations

3.1. Semantic interpretations

We shall start by outlining what is meant by obtaining a structure $\mathfrak{A} = \langle A, R \rangle$ from a structure $\mathfrak{B} = \langle B, S_1, S_2, \ldots \rangle$ by means of definable relations. We need some preliminary notions. Let L be the language of $\mathfrak{A}$

and L_1 be the language of $\mathfrak{B}$. Let $D(x, y, \dots)$ be a formula of L_1 with x as a free variable but possibly containing other free variables $y, \dots$, which will play the role of parameters. Abbreviate $D(x, y, \dots)$ by $D(x)$ or even D.

If F is a formula of L_1, then the formula F^D obtained from F by *relativizing* all quantifiers of F to D, is defined inductively on the structure of F by the following rules. If F is quantifier-free then $F^D = F$. If $F = E \vee G$ or $F = \neg E$ then $D^D = E^D \vee G^D$ or $F^D = \neg E^D$, respectively. The crucial cases are $F = \exists u\, G$ and $F = \forall u\, G$:

$$(\exists u\, G)^D = \exists u\, [D(u) \wedge G^D], \qquad (\forall u\, G)^D = \forall u\, [D(u) \rightarrow G^D].$$

Here $D(u)$ means $D(u, y, \dots)$, i.e. u substituted for x in D. Note that in order to correctly effect the relativization, we must sometimes alphabetically change the names of certain variables in F in order to avoid binding a variable other than x, which is free in D.

Let $b \in B, \dots$, be a sequence of values in B for the parameters $y, \dots$, of D. Define $C = \{a \mid \mathfrak{B} \models D(a, b, \dots)\}$. This is the domain defined by D and the specialization $y = b, \dots$ of the parameters. The subset $C \subseteq B$ induces a substructure $\mathfrak{C} = \langle C, S_1 \mid C, S_2 \mid C, \dots \rangle$ of $\mathfrak{B}$.

LEMMA. *Let* $F(z_1, \dots, z_n)$ *be a formula of* L_1, *and let* D, B, $b \in B, \dots$, *and* C *be as above, then for* $c_1, \dots, c_n \in C$

$$\mathfrak{B} \models F^D(c_1, \dots, c_n) \quad \textit{iff} \quad \mathfrak{C} \models F(c_1, \dots, c_n).$$

Thus the effect of relativization is to convert satisfaction of the formula F in $\mathfrak{B}$ to satisfaction in the substructure $\mathfrak{C}$.

A somewhat more complex construction is the following. For the sake of the simplicity of the notation, let us restrict ourselves to a formula $D(x)$ of L_1 containing just the free variable x (and no parameters) and a formula $E(u, v)$ with two free variables.

DEFINITION 3. The structure $\mathfrak{B}(D, E) = \langle C, R \rangle$ *induced in* $\mathfrak{B}$ by $D(x)$ and $E(u, v)$ has, by definition, the domain $C = \{c \mid B \models D(c)\}$ and the binary relation $R \subseteq C \times C$, $R = \{(b, c) \mid b, c \in C, \mathfrak{B} \models E(b, c)\}$.

Let now $F(z_1, \dots, z_n)$ be a formula in a language with a binary predicate symbol P and define $F^{D,E}$ to be the formula obtained from F by *first* forming F^D and *then* replacing in F^D all atomic formulas $P(z_1, z_2)$ by $E(z_1, z_2)$. Note that the quantifiers in $E(u, v)$ are not being relativized to D. The following lemma is actually a corollary of the previous lemma. It relates satisfaction in the induced structure to satisfaction in $\mathfrak{B}$.

LEMMA. *For* $c_1, \dots, c_n \in C$,

$$\mathfrak{B}(D,E) \models F(c_1,\dots,c_n) \quad \textit{iff} \quad \mathfrak{B} \models F^{D,E}(c_1,\dots,c_n).$$

The application to decidability rests on the following theorem taken from RABIN [1965].

Let T and T_1 be theories in the languages L and L_1, respectively, and let $\mathcal{K}$ and $\mathcal{K}_1$ be classes of structures such that $T = \mathrm{Th}(\mathcal{K})$, $T_1 = \mathrm{Th}(\mathcal{K}_1)$. Assume that L has the predicate symbols $\boldsymbol{P}_0, \dots, \boldsymbol{P}_k$ and no operation symbols.

THEOREM 10. *Let $D(x, y, \dots)$ be a formula of* L_1, *and $E = (E_0, \dots, E_k)$ be a sequence of formulas so that if $\boldsymbol{P}_i$ is n_i-ary then E_i has n_i free variables, $0 \le i \le k$.*

Assume that (1) *For all $\mathfrak{B} \in \mathcal{K}_1$ and all values $\boldsymbol{y} = b, \dots$ of the parameters of D, $\mathfrak{B}(D,E) \models T$.* (2) *For every $\mathfrak{A} = \langle A, R_0, \dots, R_k \rangle \in \mathcal{K}$, there exists a model $\mathfrak{B} \in \mathcal{K}_1$ and a specialization $\boldsymbol{y} = b \in B, \dots$ such that for this specialization $\mathfrak{A} \approx B(D,E)$.*

Under these conditions, if T_1 is decidable then so is T. Conversely, if T is undecidable then so is T_1.

PROOF. Let F be a *sentence* of L, put $G = \forall \boldsymbol{y} \cdots F^{D,E}$ where the universal quantification is over all the parameter-variables in $D(\boldsymbol{x}, \boldsymbol{y}, \dots)$ (these variables are free in F, if F did contain quantifiers). By the second lemma, for any $\mathfrak{B} \in \mathcal{K}_1$ and specialization $\boldsymbol{y} = b \in B \cdots$

$$(*) \qquad \mathfrak{B} \models F^{D,E}(b,\dots) \quad \textit{iff} \quad \mathfrak{B}(D,E) \models F.$$

Let now $F \in T$. Condition (1) implies $\mathfrak{B}(D,E) \models F$ for any $\mathfrak{B} \in \mathcal{K}_1$, $\boldsymbol{y} = b, \dots$. Hence the left side of $(*)$ holds, hence $\mathfrak{B} \models G$. But $\mathfrak{B} \in \mathcal{K}_1$ was arbitrary, hence $G \in \mathrm{Th}(\mathcal{K}_1) = T_1$.

Next assume $G \in T_1$. Let $\mathfrak{A} \in \mathcal{K}$, then, by (2), for some $B \in \mathcal{K}$ and $\boldsymbol{y} = b, \dots, \mathfrak{A} \approx \mathfrak{B}(D,E)$. We have $\mathfrak{B} \models G$, hence $\mathfrak{B} \models F^{D,E}(b,\dots)$. Therefore, by $(*)$, $\mathfrak{B}(D,E) \models F$, hence $\mathfrak{A} \models F$; consequently $\mathcal{K} \models F$ and $F \in T$.

Let T_1 be decidable and F be any sentence of L. Form G; since $G \in T_1$ iff $F \in T$, we can determine whether $F \in T$. □

Remark. It is readily seen that if T is finitely axiomatizable then condition (1) can be dispensed with by modifying the construction of G.

We have stated Theorem 10 for first-order languages. With appropriate changes it also holds if L_1 or even both L and L_1 are second-order languages. The only case of second-order languages which is of any interest from the point of view of decidability is that of monadic second-order languages which have variables ranging over *subsets* of the domain but no variables ranging over relations on the domain. This is because once we have a variable ranging over, say, binary relations, the theory of all true sentences of the second-order language is undecidable.

Assume that L_1 has set variables $\boldsymbol{A}, \boldsymbol{B}, \ldots$. Then the relativizing formula $D(\boldsymbol{x})$ may be of the form $\boldsymbol{x} \in \boldsymbol{A}$ and $\boldsymbol{A}$ will be a parameter in $F^{D,E}$. If L has set variables then they must also be relativized to D by rules such as

$$(\forall \boldsymbol{A}\, F)^D = \forall \boldsymbol{A}\ [\forall \boldsymbol{x}\ [\boldsymbol{x} \in \boldsymbol{A} \rightarrow D(\boldsymbol{x})] \rightarrow F^D].$$

With these natural modifications, Theorem 10 holds for monadic second-order languages.

3.2. Decidable second-order theories

Let us consider monadic second-order languages L_1 which have set variables $\boldsymbol{A}, \boldsymbol{B}, \ldots$, and the $\in$ relation. To be appropriate for a structure $\mathfrak{A} = \langle A, R\rangle$ where R is, say, a binary relation, L_1 must also have a binary predicate symbol $\boldsymbol{P}$. For such a language L_1 the *(monadic) second-order* theory $\mathrm{Th}_2(\mathfrak{A})$ of $\mathfrak{A}$ is the set of all sentences of L_1 true in $\mathfrak{A}$. Similarly we define $\mathrm{Th}_2(\mathcal{K})$ for a class $\mathcal{K}$ of similar structures by $\mathrm{Th}_2(\mathcal{K}) = \bigcap_{\mathfrak{A}\in\mathcal{K}} \mathrm{Th}_2(\mathcal{K})$.

The first significant results of second-order decidability, beyond the decidability of just pure monadic second-order logic, deal with the decidability of $\mathrm{Th}_2(\langle\omega, S\rangle)$ where $S(x) = x + 1$ is the successor function.

THEOREM 11 (BÜCHI [1962]). $\mathrm{Th}_2(\langle\omega, S\rangle)$ *is decidable.*

This result was actually preceded by a weaker version. Consider a *weak* monadic second-order language LW which has set variables $\boldsymbol{\alpha}, \boldsymbol{\beta}, \ldots$ which are restricted to range over *finite* subsets of the domain. The theory of a structure $\mathfrak{A}$ in the language LW, will be called the *weak* (monadic) second-order theory of $\mathfrak{A}$ and denoted by $\mathrm{Th}_w(\mathfrak{A})$.

BÜCHI [1960] and ELGOT [1961] have proved that $\mathrm{Th}_w(\langle\omega, S\rangle)$ is decidable.

For future reference, denote $\mathrm{Th}_2(\omega, S) = \mathrm{S1S}$ (the second-order theory of one successor function) and $\mathrm{Th}_w(\langle\omega, S\rangle) = \mathrm{WS1S}$.

We cannot enter into details of these decidability proofs. Let us just say that they utilize methods and results of automata theory. The case of WS1S employs concepts and results from theory of automata operating on finite sequences and is very simple and transparent. The proof of decidability of S1S required a new notion of an automaton operating on an infinite ω-sequence.

3.3. The tree theorem

Most of the proofs of decidability by interpretations involve the *Tree Theorem* due to Rabin [1969].

Let $T = \{0, 1\}^*$ be the set of all finite *words* (sequences) $x = x_1x_2 \cdots x_n$, $x_i \in \{0, 1\}$ on the alphabet $\{0, 1\}$. The empty sequence Λ is also in $\{0, 1\}^*$. The set T can also be interpreted as the infinite binary tree (see Fig. 1). Arbitrarily assigning 0 to *left* and 1 to *right*, the correspondence between node of the tree and T is as follows. The root corresponds to Λ; the right successor (son) corresponds to 1, and the left successor to 0; the left successor of 1 is 10, etc.

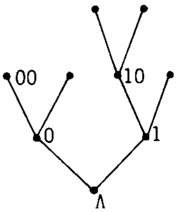

Fig. 1.

Thus we have on T the two *successor functions* $r_0(x) = x0$, $r_1(x) = x1$, $x \in T$. Let L_1 be an appropriate monadic second-order language having operation symbols r_0, r_1. The set of all sentences of L_1 true in $\langle T, r_0, r_1\rangle$ will be denoted, as usual, by $\mathrm{Th}_2(\langle T, r_0, r_1\rangle)$.

THEOREM 12 (Tree Decidability Theorem). *The second-order theory of two successor functions* $\mathrm{S2S} = \mathrm{Th}_2(\langle T, r_0, r_1\rangle)$ *is decidable.*

The proof of this theorem requires a far-reaching extension of the theory of automata to cover the case of a finite automaton operating on an infinite tree. One interesting feature of the proof is that even though we want to establish certain facts concerning finite objects, namely the finite automata, transfinite induction over ordinals up to the first uncountable cardinal ω_1 is used in an essential way.

3.4. Presburger's arithmetic revisited

Büchi and Elgot used the decidability of WS1S to prove the decidability of PAR.

The key idea is to use finite sets $\alpha \subset \omega$ to code integers. Let χ_α be the characteristic function of α. Define

$$n(\alpha) = 1 \cdot \chi_\alpha(0) + 2 \cdot \chi_\alpha(1) + \cdots + 2^x \cdot \chi_\alpha(x) + \cdots .$$

We shall construct a formula $A(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ in the language of WS1S which will be true in $\langle \omega, S \rangle$ for $\alpha, \beta, \gamma \in \omega$ if and only if $n(\alpha) + n(\beta) = n(\gamma)$. This is done by considering the sequence χ_δ of carriers in the addition of $n(\beta)$ to $n(\alpha)$ as binary numbers.

$$\begin{aligned} A(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma}) = \exists \boldsymbol{\delta}\, \forall \boldsymbol{x}\, [\neg \boldsymbol{0} \in \boldsymbol{\delta} \wedge \\ [\boldsymbol{S}(\boldsymbol{x}) \in \boldsymbol{\delta} \leftrightarrow \boldsymbol{x} \in \boldsymbol{\alpha} \wedge \boldsymbol{x} \in \boldsymbol{\beta} \vee \boldsymbol{x} \in \boldsymbol{\alpha} \wedge \\ \boldsymbol{x} \in \boldsymbol{\delta} \vee \boldsymbol{x} \in \boldsymbol{\beta} \wedge \boldsymbol{x} \in \boldsymbol{\delta}| \wedge [\boldsymbol{x} \in \boldsymbol{\gamma} \leftrightarrow \\ \boldsymbol{x} \in \boldsymbol{\alpha} \wedge \boldsymbol{x} \in \boldsymbol{\beta} \wedge \boldsymbol{x} \in \boldsymbol{\delta} \vee \\ \boldsymbol{x} \in \boldsymbol{\alpha} \wedge \sim \boldsymbol{x} \in \boldsymbol{\beta} \wedge \sim \boldsymbol{x} \in \boldsymbol{\delta} \cdots]]. \end{aligned}$$

By systematic use of $A(\boldsymbol{\alpha}, \boldsymbol{\beta}, \boldsymbol{\gamma})$ to replace $\boldsymbol{a} + \boldsymbol{b} = \boldsymbol{c}$ in a sentence F of PAR, a sentence G in the language of WS1S is obtained. We have $F \in \mathrm{Th}(\langle \omega, + \rangle)$ iff $G \in \mathrm{WS1S}$ so that we can decide whether F is true.

3.5. Second-order theory of linear order

Let $K_{\leq}^{\omega}$ be the class of all totally ordered sets $\langle A, \leq \rangle \models \mathrm{OR}$ with a countable domain, $c(A) \leq \omega$.

THEOREM 13 (RABIN [1969]). $\mathrm{Th}_2(K_{\leq}^{\omega})$ *is decidable.*

PROOF. We can define on T the partial-order $x \leq y$, x is an initial segment of y, by a formula $\boldsymbol{x} \leq \boldsymbol{y}$. Namely,

$$\boldsymbol{x} \leq \boldsymbol{y} = \forall \boldsymbol{A}\, [\boldsymbol{x} \in \boldsymbol{A} \wedge \forall \boldsymbol{z}\, [\boldsymbol{z} \in \boldsymbol{A} \rightarrow \boldsymbol{r}_0(\boldsymbol{z}) \in \boldsymbol{A} \wedge \boldsymbol{r}_1(\boldsymbol{z}) \in \boldsymbol{A}] \rightarrow \boldsymbol{y} \in \boldsymbol{A}].$$

Also the lexicographic order $\boldsymbol{x} \preccurlyeq \boldsymbol{y}$ is definable

$$\boldsymbol{x} \preccurlyeq \boldsymbol{y} = \boldsymbol{x} \leq \boldsymbol{y} \vee \exists \boldsymbol{z}\, [\boldsymbol{r}_0(\boldsymbol{z}) \leq \boldsymbol{x} \wedge \boldsymbol{r}_1(\boldsymbol{z}) \leq \boldsymbol{y}].$$

The ordered set $\langle \{x1 \mid x \in T\}, \preccurlyeq \rangle$ has order type η. Therefore for every countable ordered set $\langle A, \leq \rangle$ there exists a set $\bar{A} \subseteq T$ such that $\langle A, \leq \rangle \approx \langle \bar{A}, \preccurlyeq \rangle$. Using the relativizing formula $D(\boldsymbol{x}, \boldsymbol{A}) = \boldsymbol{x} \in \boldsymbol{A}$ and replacing $\leq$

by $\preccurlyeq$, we see that $\mathrm{Th}_2(K^{\omega}_{\leq})$ can be interpreted in S2S in the manner of Theorem 9. Hence $\mathrm{Th}_2(K^{\omega}_{\leq})$ is decidable. □

As simple corollaries we get difficult classical results.

COROLLARY. Th(OR), *the theory of linearly ordered sets, is decidable.*

The Skolem–Löwenheim theorem implies that every $\langle A, \leq\rangle \models \mathrm{OR}$ is elementarily equivalent to a $B \in K^{\omega}_{\leq}$. Hence $\mathrm{Th(OR)} = \mathrm{Th}(K^{\omega}_{\leq})$. The latter theory is, of course, decidable.

The monadic second-order language is sufficiently powerful to express the fact that a set is well-ordered. Namely, the sentence

$$W = \forall \boldsymbol{A}\, \forall \boldsymbol{x}\, \exists y\, \forall z\, [\boldsymbol{x} \in \boldsymbol{A} \rightarrow y \in \boldsymbol{A} \wedge [z \in \boldsymbol{A} \rightarrow y \leq z]]$$

has the property that a linearly ordered set $\mathfrak{A}$ satisfies $\mathfrak{A} \models W$ if and only if $\mathfrak{A}$ is well-ordered. This immediately leads to

COROLLARY. *The monadic second-order theory of countable well-ordered sets is decidable.*

PROOF. For any sentence F we have $K^{\omega}_{\leq} \models W \rightarrow F$ if and only if F is true in all countable well-ordered sets. □

Every well-ordered set $\langle B, \leq\rangle$ has a countable elementary submodel $\mathfrak{A} = \langle A, \leq\rangle \prec \langle B, \leq\rangle$. Hence the first-order theory of well-ordered sets is the same as the first-order theory of countable well-ordered sets which is decidable by the previous corollary. Thus the first-order theory of well-ordered sets is decidable, a result due to TARSKI and MOSTOWSKI [1949].

3.6. Decidability in topology

It is possible to define in S2S the notion of a *path* $A \subseteq T$ going from the root to infinity

$$\mathrm{Path}(\boldsymbol{A}) = \boldsymbol{\Lambda} \in \boldsymbol{A} \wedge \forall \boldsymbol{x}\, \forall \boldsymbol{y}\, [\boldsymbol{x} \in \boldsymbol{A} \rightarrow [\boldsymbol{r}_0(\boldsymbol{x}) \in \boldsymbol{A} \vee \boldsymbol{r}_1(\boldsymbol{x}) \in \boldsymbol{A}] \wedge [\boldsymbol{y} \leq \boldsymbol{x} \rightarrow \boldsymbol{y} \in \boldsymbol{A}] \wedge \neg\, [\boldsymbol{r}_0(\boldsymbol{x}) \in \boldsymbol{A} \wedge \boldsymbol{r}_1(\boldsymbol{x}) \in \boldsymbol{A}]].$$

Consider $\{0,1\}^{\omega}$ with the usual Tychonoff product topology. This is the well-known Cantor Discontinuum CD. For every point $p : \{0,1\} \rightarrow \omega$, the set $A \subset T$ of all finite initial segments of p, is a path of T and this is a one-to-one correspondence. We can also reproduce in S2S the topology of CD.

Let $B \subseteq T$ be a set which is a union of paths, then the set of paths $A \subseteq B$ is a closed subset of CD, and this again is a one-to-one correspondence. Define

$$\mathrm{CL}(\boldsymbol{B}) = \forall \boldsymbol{x}\, [\boldsymbol{x} \in \boldsymbol{B} \rightarrow \exists \boldsymbol{A}\, [\mathrm{Path}(\boldsymbol{A}) \wedge \boldsymbol{A} \subseteq \boldsymbol{B} \wedge \boldsymbol{x} \in \boldsymbol{A}]].$$

THEOREM 14 (RABIN [1969]). *The monadic second-order theory of* CD, *with the set variables restricted to range over closed sets, is decidable.*

PROOF. Let F be any sentence in the language of CD. Relativize all individual variables to Path($\boldsymbol{X}$) and all (closed) set variables to CL($\boldsymbol{B}$), replace all formulas $\boldsymbol{x} \in \boldsymbol{B}$ of $\boldsymbol{F}$ by $\boldsymbol{X} \subseteq \boldsymbol{B}$. The resulting sentence E is true in S2S if and only if $\mathrm{CD} \models F$. □

With slight changes, accounting for the fact that two different sequences $p, q \in \{0,1\}^{\omega}$ may represent the same element of the real-line segment $[0,1]$, the above proof may be modified to cover the case of $[0,1]$.

3.7. Boolean algebras

The following theorem settles the decidability of Th(BA) — the elementary theory of Boolean algebras and much more.

THEOREM 15 (RABIN [1969]). *Let* $\mathfrak{B}_{\omega}$ *be the free Boolean algebra on* ω *generators and let* LI *be a second-order language appropriate for Boolean algebras with set variables ranging over ideals of the algebras.* $\mathrm{Th}_{\mathrm{I}}(\mathfrak{B}_{\omega})$, *the theory of* $\mathfrak{B}_{\omega}$ *in the language* LI, *is decidable.*

PROOF. The set of all closed-and-open (clopen) subsets of CD is a Boolean algebra isomorphic to $\mathfrak{B}_{\omega}$, hence we can identify it with $\mathfrak{B}_{\omega}$. If $I \subseteq \mathfrak{B}_{\omega}$ is any ideal then $U(I) = \bigcup_{S \in I} S$ is an open set $U(I) \subseteq \mathrm{CD}$. The sets $U(I)$ run through all open sets of CD and the correspondence is 1–1. Furthermore, for $S \in \mathfrak{B}_{\omega}$, $S \subseteq U(I)$ if and only if $S \in I$.

Every sentence of LI can therefore be translated into a sentence about CD by relativizing the individual variables to set variables ranging over clopen subsets of CD, relativizing the ideal variables to variables ranging over open sets (i.e. complements of closed sets) of CD, and replacing $\boldsymbol{x} \in \boldsymbol{I}$ by $\boldsymbol{X} \subseteq \boldsymbol{I}$. The transformed sentence is true in CD if and only if the original sentence is true in $\mathrm{Th}_{\mathrm{I}}(B_{\omega})$, and the former question is decidable. □

Let now B be any countable Boolean algebra. Then there exists an ideal $J \subseteq \mathfrak{B}_{\omega}$ so that $\mathfrak{B} \approx B_{\omega}/J$, and the ideals $I \subseteq \mathfrak{B}$ are in a natural 1–1 correspondence with the ideal $J \subseteq J_1 \subseteq \mathfrak{B}_\omega$. By the method of interpretations, this will yield the following theorem. Denote by BA^ω the class of all countable Boolean algebras.

THEOREM 16. $\mathrm{Th}_\mathrm{I}(\mathrm{BA}_\omega)$, *the theory in the language* LI *of all countable Boolean algebras, is decidable.*

If we restrict ourselves to sentences $G = \forall \boldsymbol{I}_1 \cdots \forall \boldsymbol{I}_n F(\boldsymbol{I}_1, \ldots, \boldsymbol{I}_n)$, where F is a formula without any quantification over ideals, then $G \in \mathrm{Th}_\mathrm{I}(\mathrm{BA}^\omega)$ iff $F(I_1, \ldots, I_n)$ is true in every countable Boolean algebra $\mathfrak{B} = \langle B, U, \cap, ', I_1, I_2, \ldots \rangle$, where $I_1, I_2, \ldots,$ are ideals of B. Using the Skolem–Löwenheim theorem we immediately get:

THEOREM 17 (RABIN [1969]). *The elementary theory of all Boolean algebras with a sequence of distinguished ideals is decidable.*

This theorem considerably strengthens the result of ERSHOV [1964] which asserts the decidability of Boolean algebras with a distinguished prime ideal (ultra-filter).

3.8. Non-classical logics

Thus far all the results presented in this paper dealt with theories formalized within classical logic. Many extensions and modifications of classical logic appear in the literature. These may take the form of rejection of certain axioms of classical logic, the intuitionistic logic is an example, or the addition of logical operators or connectives, as is being done in modal or tense logics. Important philosophical considerations and attitudes towards the foundations of mathematics and logic motivate the introduction and study of these systems.

While the decidability of the classical propositional calculus is trivial, the decidability of these fragments or enrichments (by addition of logical operators) of even the propositional logic is in most cases far from obvious. The method of interpretations turned out to be a powerful tool for settling almost all the decidability questions in this field. We shall illustrate this by two examples. The interested reader should consult the article of GABBAY [1975], which is the source for these examples, and where many other results and references are to be found.

The class of *modal propositional logics* to be considered here has, besides the usual propositional connectives, the operator $\Box$ which is intended to express *necessity*, so that $\Box A$ should be construed to mean: necessarily A.

A basic axiom system C-2, has all the theorems and rules of classical logic, and in addition the axioms and rules of inference

$$\Box[A \to B] \to [\Box A \to \Box B],$$

$$\text{from} \quad \vdash A \to B \quad \text{to infer} \vdash \Box A \to \Box B.$$

The system K is obtained from C-2 by adding the rule

$$\text{from} \quad \vdash A \quad \text{to infer} \vdash \Box A.$$

The system T is K plus the axioms $\Box A \to A$. And the system $S4$ is T plus the axioms $\Box A \to \Box\Box A$.

There are many other extensions of C-2. The particular axioms are chosen by the various authors on the basis of their beliefs as to what the correct properties of □ should be.

Let us now describe the intuitionistic *tense* logic J_t. This system will have, besides the connectives $\to$, $\vee$, $\wedge$, $\mathfrak{f}$ (denoting falsehood), the operators G and H. For a formula A, GA reads: "A will always be true", and HA reads: "A was always true". The formula $\neg A$ abbreviates $A \to \mathfrak{f}$.

The axioms and rules of inference for J_t will be those for the intuitionistic propositional logic (including modus ponens), and in addition

$$G[A \to B] \to [GA \to GB],$$

$$H[A \to B] \to [HA \to HB],$$

$$A \vee G \neg HA, \qquad A \vee H \neg GA,$$

$$\text{from} \vdash A \quad \text{to infer} \quad \vdash GA \text{ and} \vdash HA.$$

Kripke, Gabbay, and others, gave for many non-classical logics systems of semantics based on trees and valuations on trees. A formula would then be provable if it has a certain property under all possible valuations or interpretations. The detailed definition of interpretations would, of course, depend on the system of axioms in question.

It turns out that these tree-semantics are expressible in S2S and variants thereof. This makes it possible to derive a multitude of positive decidability results from the Tree Theorem.

4. Complexity of decision procedures

4.1. Turing machine computations

As remarked in the introduction, many results concerning lower bounds on the complexity of *solvable* decision problems appeared in recent years.

In particular almost all the theories discussed in this chapter were shown to have decision problems which do not admit of any simple decision procedure.

Since our aim will be to show that *every* decision procedure for a theory T is complex, we must settle on a definite formulation for algorithms and a definite convention for counting computational steps. We shall choose Turing-machine algorithms as our decision procedures, and the execution of an atomic instruction will count as a basic step.

The results will be of the form that for any decision procedure P for the theory T in question, there are sentences A of size n (i.e. written by use of n symbols) for which P will require at least $f(cn)$ steps to produce an answer as to whether $A \in T$ or not. The function $f(n)$ will be at least exponential 2^n, and c will be a fixed number $0 < c$. Because of the exponential nature of the results, it will make little difference which model of algorithms and computations is chosen. Computation-times in different models for the same algorithm differ by at most a polynomial transformation.

The method for obtaining these inherent complexity results rests on the following observation.

Let us transcribe programs for Turing machines in a uniform standard way by sequences $P \in \{0, 1\}^*$. For any word x define $l(x)$ to be the *length* of x, i.e. the number of symbols in x. In particular, for an integer written in binary notation, $l(n)$ is the number of digits in n.

Let T be a theory in the language L, and $f(k)$ be a function satisfying the following conditions. There exists a constant $0 < d$ so that for every program P and integer n, there exists a sentence $S(n, P)$ of L satisfying:

(i) $l(S(n, P)) \leq d(l(n) + l(P))$,

(ii) $S(n, P) \in T$ if and only if a computation by the program P on input n (viewed as a 0–1 sequence) halts in fewer than $f(l(n))$ steps.

(iii) The formula $S(n, P)$ can be effectively computed from n, P in fewer than $g(l(n) + l(P))$ steps, where $g(k)$ is a fixed polynomial.

If $f(k)$ is a function growing at least at exponential rate, then under the above conditions there exists a constant $0 < c$ so that for infinitely many n there exists a sentence F of L, $l(F) = n$, for which P requires at least $f(cn)$ steps to decide whether $F \in T$.

The proof of the above statement is by a familiar diagonalization argument. One asssumes, by way of contradiction, that there exists a decision procedure P for T which requires for every sentence F, fewer than $f(cl(F))$ steps to decide whether $F \in T$. If $c = 1/2d$ then, by use of the sentence $S(n, P)$, one can construct a Turing machine which stops on an input n_0 if and only if it does not stop on that input.

4.2. The theory WS1S

MEYER [1975] has proved that the decision problem of WS1S is of a very high inherent complexity.

Define a function $F(n, m)$ by

$$F(n, 1) = 2^n, \qquad F(n, m+1) = 2^{F(n,m)}, \qquad m = 1, 2, \dots .$$

If $0 < d$, then $f(n) = F(n, [dn])$ is a function which is an exponentiation by a linear stack of 2's.

THEOREM 18. *There exists a constant* $0 < d$ *so that for the function* $f(n) = F(n, [dn])$, *and every algorithm P for solving the decision problem of* WS1S, *there exist infinitely many formulas A so that P requires more than* $f(l(A))$ *steps to decide whether* $A \in \text{WS1S}$.

PROOF (outline). We have available in the language of WS1S variables $\boldsymbol{\alpha}, \boldsymbol{\beta}, \dots$, ranging over finite subsets of ω. A pair of subsets α, β can be used to code a sequence $p \in \{0, 1\}^*$, $l(p) = c(\alpha)$. If $\alpha = \{i_0 < i_1 < \cdots < i_{k-1}\}$, then $i_j \in \beta$ if and only if $p(j) = 1$. For a fixed α and variable β, the pairs (α, β) will run through codes of all sequences p such that $l(p) = c(\alpha)$.

For the above α and $x \in \alpha$, $y \in \alpha$, we shall say that x and y are d apart in α if $x = i_j$, $y = i_{j+d}$, for some $j \leq k - 1 - d$.

One can now show that for every n there exist two formulas $A_n(\boldsymbol{\alpha})$ and $D_n(\boldsymbol{\alpha}, \boldsymbol{x}, \boldsymbol{y})$ of WS1S which are of length $0(n)$ and have the following properties. $A_n(\boldsymbol{\alpha})$ implies that α has a certain structure and is at least of cardinality $(F(n, n))^2$. For sets α for which $F_n(\boldsymbol{\alpha})$ holds, $D_n(\boldsymbol{\alpha}, \boldsymbol{x}, \boldsymbol{y})$ means that x and y are at distance $F(n, n)$ apart in $\boldsymbol{\alpha}$.

Suppose that we have a Turing machine computation with fewer than $F(n, n)$ steps. Then the machine-head will never be farther than $F(n, n)$ squares away from the starting square. We can assume without loss of generality that the machine never crosses to the left of the starting square.

We can string out in order, from left to right, the complete descriptions of the stretch of the first (leftmost) $F(n, n)$ squares after the execution of each of the machine-instructions. This will be a sequence of length at most $(F(n, n))^2$. This sequence can be coded by use of an $\alpha \subset \omega$ which satisfies $A_n(\boldsymbol{\alpha})$, and additional sets $\beta_0, \beta_1, \dots, \beta_k$. The pair (α, β_0) will code the tape-contents, and $((\alpha, \beta_1), \dots, (\alpha, \beta_k))$ will code the sequence of head locations and machine states.

The formula $D_n(\alpha, x, y)$ will serve as a "ruler" measuring off stretches of length $F(n, n)$. Together with the (definable) order on ω, it will enable us to express the fact that two consecutive stretches of the sequence coded by $(\alpha, \beta_0, \ldots, \beta_k)$ are related by an execution of a single Turing machine instruction.

Filling out the details and combining the above ideas, it is possible to show that there exists for WS1S a construction of a formula $S(n, P)$ with the properties enumerated in 4.1. This entails Theorem 18. □

It was independently observed by E. Robertson and by L. Stockmeyer (in his thesis) that a close examination of the full proof of Theorem 18 reveals that it will go through for sentences pertaining to $\langle \omega, \leq \rangle$ which are universal monadic second-order. This means sentences which may contain set quantifiers but these are all $\forall$ quantifiers and appear at the beginning of the sentence. In fact, a single set variable will suffice. A method of direct interpretations will yield from this more detailed result the following theorem due to MEYER [1974, 1975].

THEOREM 19. *The first-order theory* Th(OR) *of linearly ordered sets has inherent complexity* $F(n, [dn])$ *for some* $0 < d$.

The detailed formulation of Theorem 19, as well as of the results in the next subsection, is as in Theorem 18.

An analysis of the automata-theory based decision procedures for WS1S and even S2S shows that they run in time $F(n, [cn])$ for formulas A of size n, for an appropriate $0 < c$. In view of Theorems 18–19 these results are, qualitatively, best possible. There is, of course, the question of the height $[cn]$ of the stack of 2's, but this depends on the notation for the formulas and does not seem to be readily answerable.

4.3. Theories of addition and real-closed fields

For the classical theories $\mathrm{Th}(\langle \omega, + \rangle) = \mathrm{PAR}$, and the theory of the field of real numbers Th(RLC), the inherent complexity results are not as devastating as for WS1S. It does, however, turn out that these theories are at least exponentially complex, and in some cases super-exponentially complex. Thus the contention that the existence of a decision procedure trivializes these theories is not justified.

THEOREM 20 (FISCHER and RABIN [1974]). *There exists a constant* $0 < c$ *so* *that the decision problem of Presburger's arithmetic* PAR *is at least of complexity* $2^{2^{cn}}$.

Proof (outline). We saw that the ability to establish inherent complexity results for a complexity function $f(n)$, rests on the possibility to code within the theory by formulas of size $O(n)$ sequences of length $(f(n))^2$.

There exist in the language of $\langle \omega, + \rangle$ formulas $P_n(x, y, z)$ of size $O(n)$, which are true for any $x, y, z \in \omega$ if and only if $x, y, z < F(n, 3)$ and $x \cdot y = z$. Thus such a formula, which involves only $+$ and is of size $O(n)$, codes the multiplication table up to $2^{2^{2^n}}$. Using integers to code 0–1 sequences, it is now possible to code sequences of length up to $(2^{2^n})^2$ by employing $P_{n+1}(x, y, z)$. □

The lower bound for the complexity of Th(RLC) = Th($\langle R, +, \cdot \rangle$), where R is the field of real numbers, is obtained by considering just $\langle R, + \rangle$.

Theorem 21 (Fischer and Rabin [1974]). Th($\langle R, + \rangle$), *and consequently also* Th(RLC), *are of inherent complexity at least* 2^{cn} *for some* $0 < c$.

The proof is similar to the proof of Theorem 20. In this case it is possible to reproduce (up to isomorphism) by a short formula the multiplication table of integers up to 2^{2^n}.

For the theory of $\mathfrak{M} = \langle \omega, \cdot \rangle$ of the integers under multiplication, the situation is even worse according to a theorem mentioned in Fischer and Rabin [1974], the proof of which will be given in a forthcoming paper of Fischer and Rabin.

Theorem 22. *The theory* Th$\langle \omega, \cdot \rangle$) *of multiplication of natural numbers is of inherent complexity at least* $F(cn, 3)$, *i.e.* $2^{2^{2^{cn}}}$, *for some* $0 < c$.

Algorithms carefully constructed by various researchers strongly suggest that the above lower-bound results are best possible.

4.4. Propositional calculus and $P = NP$

It is customary in the study of abstract computation-models to make a distinction between deterministic and non-deterministic algorithms. When presented with a state-symbol combination, a Turing machine will execute a definite basic computational step (atomic move). The overall course of the computation is, therefore, completely determined by the Turing machine (program), the starting state, and the initial tape-input. The deterministic mode is, of course, a feature of all present-day computers.

The notion of *non-deterministic* computations, or programs, is of fundamental importance in theoretical studies of the properties of algorithms. A non-deterministic program P allows, when presented with a state-symbol combination, the execution of one out of *possibly several* basic moves. For example state q_3, when presented with 1, may call for either $(0, L, 7)$ (erase 1, move to left, go to state q_7) or $(1, R, 15)$. Thus, in a non-deterministic program, to each pair (q, b) where q is a state and b a symbol, there corresponds a set of triplets (c, M, q_i), c is a symbol, $M \in \{L, R\}$, q_i is a state.

When started in state q_0 on an input tape, the program P may be able to go through any one of several sequences of basis steps, i.e. perform different computations on the input. It should be borne in mind that in each particular run a definite unique computation is performed. But several different runs, or *threads*, may be possible.

Let us illustrate the idea by showing that there is a non-deterministic program P which will factor any composite number n in $f(l(n))$ steps where $f(k)$ is a polynomial.

The program P has non-deterministic instructions enabling it, when given input n (in binary notation), to write on the tape any two numbers $1 < b$, $c < n$. As observed before, in any given run, one pair (b, c) will be written. But for every pair, there exists a run producing that pair. After b, c was produced, the program switches to a deterministic mode, calculates $b \cdot c$ and checks whether $n = b \cdot c$. The machine will stop only if the test showed equality.

We may observe the following features. Not every computation by P on n will halt. But if n is indeed composite then there are computations which will halt after a number of computational steps which is polynomial in the size $l(n)$ of the input.

This can be summarized by saying that compositeness of numbers can be non-deterministically recognized in polynomial time.

Consider now the problem of determining whether a propositional formula $F(p_1, \ldots, p_n)$ has a truth-values substitution for the propositional variables, so that F becomes true. This is the *satisfiability* problem for the propositional calculus.

Since $F(p_1, \ldots, p_n)$ is not satisfiable if and only if $F(p_1, \ldots, p_n)$ is a formal theorem of propositional calculus, the satisfiability problem is closely related to the decision problem of PC.

It is again easy to construct a non-deterministic program which will determine whether F is satisfiable by a number of steps which is polynomial in $l(F)$.

Does there exist an ordinary, deterministic, decision procedure for satisfiability which requires time (i.e. number of steps) which is just polynomial in the size of the formula? The polynomial in question may, of course, be faster growing than the polynomial for the non-deterministic program.

The answer to this question is not known. However COOK [1971] has shown that:

THEOREM 23. *If the satisfiability problem of the propositional calculus can be (deterministically) solved in polynomial time, then any problem which can be solved non-deterministically in polynomial time can also be solved in polynomial time by a deterministic algorithm.*

Thus the question whether there exists an efficient (polynomial) algorithm for satisfiability is equivalent to the question whether the class P of algorithms requiring polynomial time is equipotent with the class NP of non-deterministic algorithms requiring polynomial time. This is the celebrated $P = NP$ problem.

The algorithms in NP are very powerful. For example, an isomorphism between two given graphs of size n can be non-deterministically found in polynomial time. Similarly for an Hamiltonian circuit. These are difficult combinatorial-computational problems which defied repeated attempts at simple solutions.

Cook, KARP [1972], and others, found many examples of combinatorial decision problems which are reducible and in a certain sense equivalent to the satisfiability problem. The weight of this evidence may point in the direction that the satisfiability problem, being so powerful, is not of polynomial complexity and hence $P \neq NP$. But this fundamental question is, as yet, unanswered.

References

AX, J.
[1968] The elementary theory of finite fields, *Ann. of Math.*, **88**, 239–271.

AX, J. and S. KOCHEN
[1965a] Diophantine problems over local fields I, *Am. J. Math.* **87**, 605–630.
[1965b] Diophantine problems over local fields II: A complete set of axioms for p-adic number theory, *Am. J. Math.*, **87**, 631–648.

[1966] Diophantine problems over local fields III: Decidable fields, *Ann. of Math.*, **83,** 437–456.

BÜCHI, J.R.

[1960] Weak second order arithmetic and finite automata, *Z. Math. Logik Grundlagen Math.*, **6,** 66–92.

[1962] On a decision method in restricted second order arithmetic, in: *Logic, Methodology and Philosophy of Science*, Proceedings of the 1960 Congress (Stanford Univ. Press, Stanford, CA) pp. 1–11.

COHEN, P.J.

[1969] Decision procedures for real and p-adic fields, *Comm. Pure Appl. Math.*, **22,** 131–151.

COOK, S.A.

[1971] The complexity of theorem proving procedures, Proceedings of the 3rd Annual ACM Symposium on theory of computing, pp. 151–158.

EHRENFEUCHT, A.

[1959] Decidability of the theory of linear ordering relation, *Notices Am. Math. Soc.*, **6,** 268–269.

ELGOT, C.C.

[1961] Decision problems of finite automata design and related arithmetics, *Trans. Am. Math. Soc.*, **98,** 21–51.

ERSHOV, YU. L.

[1964] Decidability of relatively complemented distributive lattices and the theory of filters (in Russian), *Algebra i Logika* **3,** 17–38.

FEFERMAN, S. and R.L. VAUGHT

[1959] The first order properties of products of algebraic systems, *Fund. Math.*, **47,** 57–103.

FISCHER, M.J. and M.O. RABIN

[1974] Super exponential complexity of Presburger's arithmetic, *SIAM–AMS Proceedings* **7,** 27–41.

GABBAY, D.M.

[1975] Decidability results in non-classical logics, Part I, *Ann. Math. Logic*, **8,** 237–295.

KARP, R.M.

[1972] *Reducibility among Cominatorial Problems, Complexity of Computer Computations,* edited by R.E. Miller and J.W. Thatcher (Plenum Press, New York) pp. 85–104.

LAUCHLI, H. and J. LEONARD

[1966] On the elementary theory of linear order, *Fund. Math.*, **59,** 109–116.

MEYER, A.R.

[1975] The inherent complexity of theories of ordered sets, in: *Proceedings of the International Congress of Mathematics,* Vancouver 1974, Vol. 2, Canadian Mathematical Congress, pp. 477–482.

MOSTOWSKI, A.

[1952] On direct products of theories, *J. Symbolic Logic,* **17,** 1–31.

PRESBURGER, M.

[1929] Über die Vollständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt, *Comptes Rendus du Ier Congrès des Mathématiciens des Pays Slaves, Warszawa,* pp. 92–101.

RABIN, M.O.

[1965] A simple method for undecidability proofs and some applications, in: *Logic, Methodology and Philosophy of Science II,* edited by Y. Bar-Hillel (North-Holland, Amsterdam) pp. 58–68.

[1969] Decidability of second order theories and automata on infinite trees, *Trans. Am. Math. Soc.*, **141,** 1–35.

ROBINSON, A.

[1956] *Complete Theories* (North-Holland, Amsterdam).

SZMIELEW, W.

[1954] Elementary properties of Abelian groups, *Fund. Math.*, **41**, 203–271.

TARSKI, A.

[1949] Arithmetical classes and types of Boolean algebras (Preliminary report), *Bull. Am. Math. Soc.*, **55**, 64.

TARSKI, A.

[1951] *A Decision Method for Elementary Ålgebra and Geometry*, 2nd revised ed. (Berkeley and Los Angeles).

TARSKI, A. and A. MOSTOWSKI

[1949] Arithmetical classes and types of well ordered systems, *Bull. Am. Math. Soc.*, **55**, 65.