SECTION

I

MATHEMATICAL LOGIC

Symposium on Decision Problems

ON A DECISION METHOD IN RESTRICTED SECOND ORDER ARITHMETIC

J. RICHARD BÜCHI

University of Michigan, Ann Arbor, Michigan, U.S.A.

Let SC be the interpreted formalism which makes use of individual variables t, x, y, z, . . . ranging over natural numbers, monadic predicate variables q( ), r( ), s( ), i( ), . . . ranging over arbitrary sets of natural numbers, the individual symbol 0 standing for zero, the function symbol ′ denoting the successor function, propositional connectives, and quantifiers for both types of variables. Thus SC is a fraction of the restricted second order theory of natural numbers, or of the first order theory of real numbers. In fact, if predicates on natural numbers are interpreted as binary expansions of real numbers, it is easy to see that SC is equivalent to the first order theory of [Re, +, Pw, Nn], whereby Re, Pw, Nn are, respectively, the sets of non-negative reals, integral powers of 2, and natural numbers.

The purpose of this paper is to obtain a rather complete understanding of definability in SC, and to outline an effective method for deciding truth

This work was done under a grant from the National Science Foundation to the Logic of Computers Group, and with additional assistance through contracts with the Office of Naval Research, Office of Ordnance Research, and the Army Signal Corps.

of sentences in SC. This answers a problem of A. Tarski's, which was discussed by R. M. Robinson [10].

A *congruence of finite rank* on words is a congruence with finite partition of concatenation; a *multi-periodic set* of words is a union of congruence classes of a congruence of finite rank. These concepts are intimately related to that of a finite automaton (Kleene [5], Myhill [6], Copi, Elgot, and Wright [3]), and turn out to be the key to an investigation of SC. Our results concerning SC may therefore be viewed as an application of the theory of finite automata to logic. In turn, SC arises quite naturally as a condition-language (Church [2]) on finite automata or sequential circuits, and *"sequential calculus"* is an appropriate name for SC. The significance of the decision method for SC is that it provides a method for deciding whether or not the input (**i**)-to-output (**u**) transformation of a proposed circuit $A(\mathbf{i}, \mathbf{r}, \mathbf{u})$ satisfies a condition $C(\mathbf{i}, \mathbf{u})$ stated in SC.

An important role in our theory of SC is played by Lemma 1, the *Sequential Lemma*. This is a combinatorial statement about ω-sequences, which may well be of importance elsewhere. It turns out to be a simple consequence of Ramsey's Theorem A. The usefulness of the "Unendlichkeitslemma" of König (also known as the "fan-theorem" in its intuitionistic version) in related problems of automata theory was first observed by Jesse B. Wright. Because of its affinity to König's lemma the present application of Ramsey's theorem was suggested. The author wishes to thank Dr. Wright for his continued assistance in the work presented here.

1. Notations

i denotes an n-tuple of predicate variables. Expressions like $A[\mathbf{i}(0)]$, $B[\mathbf{i}(t), \mathbf{i}(t')]$ denote propositional formulas in the indicated constituents. Σ_n, Π_n, denote the classes of formulas of SC of the following type:

$$\begin{aligned}
\Sigma_1 &: (\exists \mathbf{r}) \cdot A[\mathbf{r}(0)] \wedge (\forall t) B[\mathbf{i}(t), \mathbf{r}(t), \mathbf{r}(t')] \wedge (\exists t)\, C[\mathbf{r}(t)],\\
\Pi_1 &: (\forall \mathbf{r}) \cdot A[\mathbf{r}(0)] \vee (\exists t)\, B[\mathbf{i}(t), \mathbf{r}(t), \mathbf{r}(t')] \vee (\forall t)\, C[\mathbf{r}(t)],\\
\Sigma_{n+1} &: (\exists \mathbf{r}) \cdot F(\mathbf{i}, \mathbf{r}), \text{ whereby } F \in \Pi_n,\\
\Pi_{n+1} &: (\forall \mathbf{r}) \cdot F(\mathbf{i}, \mathbf{r}), \text{ whereby } F \in \Sigma_n.
\end{aligned}$$

The quantifiers $(\exists t)_x^y A(t)$ for $(\exists t)[x \leqq t < y \wedge A(t)]$, $(\forall t)_x^y A(t)$ for $(\forall t)[x \leqq t < y \supset A(t)]$, $(\exists^\omega t)A(t)$ for $(\forall x)(\exists t)[x < t \wedge A(t)]$, $(\forall_\omega t)A(t)$ for $(\exists x)(\forall t)[x < t \supset A(t)]$, $(\exists j)_\omega A(j)$ for $(\exists j)[(\exists^\omega t) j(t) \wedge A(j)]$ can be defined in SC. The classes Σ_1^ω and Π_1^ω of formulas are defined as follows:

$$\begin{aligned}
\Sigma_1^\omega &: (\exists \mathbf{r}) \cdot A[\mathbf{r}(0)] \wedge (\forall t)\, B[\mathbf{i}(t), \mathbf{r}(t), \mathbf{r}(t')] \wedge (\exists^\omega t)\, C[\mathbf{r}(t)],\\
\Pi_1^\omega &: (\forall \mathbf{r}) \cdot A[\mathbf{r}(0)] \vee (\exists t)\, B[\mathbf{i}(t), \mathbf{r}(t), \mathbf{r}(t')] \vee (\forall_\omega t)\, C[\mathbf{r}(t)].
\end{aligned}$$

Also the following classes of formulas will play an essential role:

$$\begin{aligned}
\Sigma^0 &: (\exists \mathbf{r}) \cdot A[\mathbf{r}(x)] \wedge (\forall t)_x^y B[\mathbf{i}(t), \mathbf{r}(t), \mathbf{r}(t')] \wedge C[\mathbf{r}(y)],\\
\Pi^0 &: (\forall \mathbf{r}) \cdot A[\mathbf{r}(x)] \vee (\exists t)_x^y B[\mathbf{i}(t), \mathbf{r}(t), \mathbf{r}(t')] \vee C[\mathbf{r}(y)].
\end{aligned}$$

These may be called *regular formulas*.

Let $\mathbf{i}$ be a k-tuple of predicates. The 2^k *states of* $\mathbf{i}$ are the k-tuples of truth-values. $\mathbf{i}$ may be viewed as an infinite sequence $\mathbf{i}(0)\mathbf{i}(1)\mathbf{i}(2)\ldots$ of states. The variables $u, v, w, \ldots$ will be used for *words* (i.e., finite sequences) of states; uv denotes the result of juxtaposing the words u and v. A *congruence* is an equivalence relation $u \backsim v$ on words such that $u \backsim v$ implies $uw \backsim vw$ and $wu \backsim wv$; it is of *finite rank* n in case there are n equivalence classes. A set $\mathscr{S}$ of words is *multi-periodic* if $\mathscr{S} = \mathscr{E}_1 \cup \ldots \cup \mathscr{E}_m$, whereby $\mathscr{E}_1, \ldots \mathscr{E}_m$ are some of the congruence classes of a congruence of finite rank.

Note that the value of a regular formula $R(\mathbf{i}, x, y)$ depends only on the word $\mathbf{i}(x)\,\mathbf{i}(x+1)\ldots\mathbf{i}(y-1)$. If $\mathscr{R}$ is the set of all words $\mathbf{i}(0)\,\mathbf{i}(1)\ldots\mathbf{i}(h)$ such that $R(\mathbf{i}, 0, h+1)$, then the formula $R(\mathbf{i}, x, y)$ is said to *determine* the set $\mathscr{R}$ of words. The symbol "Σ^0" will be used also to denote the class of all sets $\mathscr{R}$ of words determined by formulas R in Σ^0. Similarly, the symbol "Σ_1^ω" is used also to denote the class of all sets $\hat{\mathbf{i}}F(\mathbf{i})$ defined by formulas $F(\mathbf{i})$ in Σ_1^ω. Corresponding remarks hold for Π^0, Π_1^ω, Σ_1, Π_1.

2. The Sequential Lemma

The working of the decision-method for SC is based on induction and a rather more sophisticated property of infinity, namely Theorem A of Ramsey [9]. Essential parts of this theorem can actually be formulated in SC, in the form of a surprising assertion about the division of infinite sequences into consecutive finite parts.

LEMMA 1. *Let* $\mathbf{i}$ *be any k-tuple of predicates, and let* $\mathscr{E}_0, \ldots, \mathscr{E}_n$ *be a partition of all words on states of* $\mathbf{i}$ *into finitely many classes. Then there exists a division* $\mathbf{i}(0)\,\mathbf{i}(1)\ldots\mathbf{i}(x_1-1), \mathbf{i}(x_1)\,\mathbf{i}(x_1+1)\ldots\mathbf{i}(x_2-1), \mathbf{i}(x_2)\,\mathbf{i}(x_2+1)\ldots\mathbf{i}(x_3-1), \ldots$ *of* $\mathbf{i}$ *such that all words* $\mathbf{i}(x_p)\,\mathbf{i}(x_p+1)\ldots\mathbf{i}(x_q-1)$ *belong to one and the same of the classes* $\mathscr{E}_0, \ldots, \mathscr{E}_n$.

PROOF. Assume $\mathbf{i}, \mathscr{E}_0, \ldots, \mathscr{E}_n$ are as supposed in Lemma 1. For $0 \leqq c \leqq n$ let P_c consist of all $\{y_1, y_2\}$ such that $y_1 < y_2$ and $\mathbf{i}(y_1)\,\mathbf{i}(y_1+1)\ldots\mathbf{i}(y_2-1) \in \mathscr{E}_c$. Then $P_0, \ldots, P_n$ clearly is a partition of all 2-element sets of natural numbers. By Ramsey's Theorem A it follows that there is an infinite sequence $x_1 < x_2 < x_3 < \ldots$ and a $0 \leqq c \leqq n$ such that $\{x_p, x_q\} \in P_c$ for all $x_p < x_q$. By definition of P_c this yields the conclusion of Lemma 1.

3. Finite Automata, Multi-periodic Sets, and Σ^0-formulas

The following methods and results are borrowed from the theory of finite automata, and play a very essential role in the study of SC. The reader is referred to Büchi [1], where some of the details are carried out in similar form, and where further references to the mathematical literature on finite automata are given. The basic result is

LEMMA 2. *The following are equivalent conditions on a set* $\mathscr{R}$ *of words:*

(*a*) $\mathscr{R}$ *is determined by a formula* $F(\mathbf{i}, x, y)$ *of* Σ^0.

(*b*) *There is a "finite automata recursion"* $\mathbf{r}(0) \equiv \mathbf{I}$, $\mathbf{r}(t') \equiv \mathbf{J}[\mathbf{i}(t), \mathbf{r}(t)]$, *and an "output"* $U[\mathbf{r}(t)]$ *such that a word* $\mathbf{i}(0)\,\mathbf{i}(1)\ldots\mathbf{i}(x-1)$ *belongs to* $\mathscr{R}$ *just in case the recursion yields an* $\mathbf{r}(x)$ *such that* $U[\mathbf{r}(x)]$ *holds.*

The implication $a \rightarrow b$ is shown in essence by Myhill's [6] "subset-construction"; nearly in the present form the details are in [1] Lemma 7. The implication $b \rightarrow a$ is trivial, $(\exists \mathbf{r}) \cdot \mathbf{r}(x) \equiv \mathbf{I} \wedge (\forall t)_x^y[\mathbf{r}(t') \equiv \mathbf{J}(t)] \wedge U[\mathbf{r}(y)]$ clearly determines $\mathscr{R}$.

The set $\mathscr{R}$ defined by (b) is sometimes called the *behavior* of $[\mathbf{I}, \mathbf{J}, U]$. In this terminology Lemma 2 says that Σ^0 is exactly the class of all behaviors of finite automata with outputs. It is easy to see that the class of behaviors is closed under disjunction and complementation. For example, if $\mathscr{R}$ is the behavior of $[\mathbf{I}, \mathbf{J}, U]$, then clearly $\tilde{\mathscr{R}}$ is the behavior of $[\mathbf{I}, \mathbf{J}, \tilde{U}]$. Therefore by Lemma 2,

LEMMA 3. *If the formulas* $R(\mathbf{i}, x, y)$, $S(\mathbf{i}, x, y)$ *determine* Σ^0*-sets of words, then so do the formulas* $R(\mathbf{i}, x, y) \wedge S(\mathbf{i}, x, y)$, $R(\mathbf{i}, x, y) \vee S(\mathbf{i}, x, y)$, *and* $\sim R(\mathbf{i}, x, y)$.

Suppose next that $R(\mathbf{i}, x, y)$ is the Σ^0-formula $(\exists \mathbf{r}) \cdot K(x) \wedge (\forall t)_x^y H(t) \wedge L(y)$. Then clearly $(\exists z)_x^y R(\mathbf{i}, x, y)$ is equivalent to $(\exists s\mathbf{r}) \cdot s(x) \wedge (\forall t)_x^y[(s(t') \supset s(t)) \wedge (s(t)\tilde{s}(t') \supset K(t)) \wedge (\tilde{s}(t) \supset H(t))] \wedge [\tilde{s}(y)L(y)]$, which is again in Σ^0. Therefore by Lemma 3,

LEMMA 4. *If the formula* $R(\mathbf{i}, x, y)$ *determines a* Σ^0*-set of words, then so do the formulas* $(\exists z)_x^y R(\mathbf{i}, x, y)$ *and* $(\forall z)_x^y R(\mathbf{i}, x, y)$.

Suppose again that $R(\mathbf{i}, x, y)$ is a Σ^0-formula. By Lemma 2 it follows that

$$(1) \qquad R(\mathbf{i}, 0, y) .=. (\exists \mathbf{r}) \cdot \mathbf{r}(0) \equiv \mathbf{I} \wedge (\forall t)[\mathbf{r}(t') \equiv \mathbf{J}(t)] \wedge U(y)$$

for properly chosen matrices $\mathbf{I}$, $\mathbf{J}[\mathbf{i}(t), \mathbf{r}(t)]$, and $U[\mathbf{r}(y)]$. It clearly follows that

$$(2) \qquad R(\mathbf{i}, 0, y) .\equiv. (\forall \mathbf{r}) \cdot [\mathbf{r}(0) \equiv \mathbf{I} \wedge (\forall t)[\mathbf{r}(t') \equiv \mathbf{J}(t)]] \supset U(y).$$

By (1) it follows that $(\exists y)\ R(\mathbf{i}, 0, y)$ is equivalent to a Σ_1-formula. By (2) it follows that $(\forall y)\ R(\mathbf{i}, 0, y)$ is equivalent to $(\forall \mathbf{r}) \cdot [\mathbf{r}(0) \equiv \mathbf{I} \wedge (\forall t)[\mathbf{r}(t') \equiv \mathbf{J}(t)]] \supset (\forall t)\ U(t)$, and therefore to $(\exists \mathbf{r}) \cdot \mathbf{r}(0) \equiv \mathbf{I} \wedge (\forall t)[\mathbf{r}(t') \equiv \mathbf{J}(t)] U(t)$ Thus,

LEMMA 5. *If* $R(\mathbf{i}, x, y)$ *determines a* Σ^0*-set of words, then* $(\exists t)\ R(\mathbf{i}, 0, t)$ *is equivalent to a* Σ_1*-formula, and* $(\forall t)\ R(\mathbf{i}, 0, t)$ *is equivalent to a* Σ_1*-formula of type* $(\exists \mathbf{r}) \cdot K(0) \wedge (\forall t)H(t)$.

As a consequence of Lemma 2, one thus obtains a rather clear picture of definability by Σ^0-formulas. However, a further characterization of behaviors is needed for the study of SC.

LEMMA 6. *A set* $\mathscr{R}$ *of words satisfies* (b) *of Lemma 2* (*i.e., is the behavior of some finite automaton with output*) *if and only if it is multi-periodic.*

This fact has been observed by several authors; a proof can be found in Rabin and Scott [8]. By Lemma 2 it follows that Σ^0 consists exactly of the multi-periodic sets of words.

4. Definability by Σ_1^ω-formulas

We will now show that also the class Σ_1^ω, just like Σ^0, is closed under Boolean operations.

LEMMA 7. *If $F_1(\mathbf{i})$ and $F_2(\mathbf{i})$ are Σ_1^ω-formulas, then also $F_1(\mathbf{i}) \vee F_2(\mathbf{i})$ is equivalent to a Σ_1^ω-formula.*

PROOF. For $c = 1, 2$ let $F_c(\mathbf{i})$ be the formula

$$(\exists \mathbf{r}_c) \cdot K_c(0) \wedge (\forall t) H_c(t) \wedge (\exists^\omega t) L_c(t).$$

Then clearly the Σ_1^ω-formula

$$(\exists s\, \mathbf{r}_1 \mathbf{r}_2) \cdot [s(0)\, K_1(0) \vee \tilde{s}(0)\, K_2(0)] \wedge (\forall t)[[s(t) \equiv s(t')] \wedge [s(t) H_1(t) \vee \tilde{s}(t) H_2(t)]] \wedge (\exists^\omega t)[s(t) L_1(t) \vee \tilde{s}(t) L_2(t)]$$

is equivalent to $F_1(\mathbf{i}) \vee F_2(\mathbf{i})$.

That Σ_1^ω also is closed under conjunction follows by

LEMMA 8. *A formula of form $(\exists \mathbf{r}) \cdot K(0) \wedge (\forall t) H(t) \wedge (\exists^\omega t) L_1(t) \wedge (\exists^\omega t) L_2(t)$ is equivalent to a Σ_1^ω-formula.*

PROOF. If the predicate $s(t)$ is defined from $p_1(t)$ and $p_2(t)$ by the recursion $s(0) \equiv F$, $s(t') \equiv [\tilde{s}(t)\, p_1(t) \vee s(t)\, \tilde{p}_2(t)]$, then it is easy to see that $[(\exists^\omega t)\, p_1(t) \wedge (\exists^\omega t)\, p_2(t)] \equiv (\exists^\omega t)\, s(t)$. Using this device with p_1 and p_2 corresponding to L_1 and L_2, one obtains a Σ_1^ω-formula as required in Lemma 8.

Using all previous lemmas, one can now establish the closure of Σ_1^ω under complementation.

LEMMA 9. *To every formula $A(\mathbf{i})$ in Σ_1^ω one can obtain a formula $B(\mathbf{i})$ in Σ_1^ω equivalent to $\sim A(\mathbf{i})$.*

PROOF. Suppose $A(\mathbf{i})$ is in Σ_1^ω, say

$$(1) \qquad A(\mathbf{i}) : (\exists \mathbf{r}) \cdot K[\mathbf{r}(0)] \wedge (\forall t)\, H[\mathbf{i}(t),\ \mathbf{r}(t),\ \mathbf{r}(t')] \wedge (\exists^\omega t)\, L[\mathbf{r}(t)].$$

If V, W are states of $\mathbf{r}$ and if $x = X_0 X_1 \ldots X_h$ is a word of states of $\mathbf{i}$, then define

$$[V, x, W]_1 : \bigvee_{U_1 \ldots U_h} \cdot H[X_0, V, U_1] \wedge H[X_1, U_1, U_2] \wedge H[X_2, U_2, U_3] \wedge \ldots \wedge H[X_h, U_h, W],$$

$$[V, x, W]_2 : \bigvee_{U_1 \ldots U_h} \cdot H[X_0, V, U] \wedge \ldots \wedge H[X_h, U_h, W] \wedge [L[U_1] \vee \ldots \vee L[U_h]].$$

(Read $[\ ]_1$ as "there is an H-transition from V by x to W", and $[\ ]_2$ as "there is an H-transition through L from V by x to W".) Next define the binary relation $\backsim$ on words of states of $\mathbf{i}$:

$$x \backsim y : \bigwedge_{VW} ([V, x, W]_1 \equiv [V, y, W]_1) \wedge \bigwedge_{VW} ([V, x, W]_2 \equiv [V, y, W]_2).$$

If m is the number of states of $\mathbf{r}$, then clearly $\backsim$ is the intersection of $m^2 + m^2$ dichotomies. Therefore, $\backsim$ is an equivalence relation of finite

rank $a \leqq 2^{2m^2}$. Furthermore, using the definitions of $[\ ]_1$ and $[\ ]_2$, one obtains, $\sim$ is a congruence relation on words. By Lemmas 2 and 6 it therefore follows that one can find formulas $E_1(\mathbf{i}, x, y), \ldots, E_a(\mathbf{i}, x, y)$ such that (2) $E_1, \ldots, E_a$ are Σ^0-formulas, and (3) $E_1, \ldots E_a$ determine the congruence classes of $\sim$.

Next one applies Lemma 1 to the partition $E_1, \ldots, E_a$. It follows that for any $\mathbf{i}$

$$(4)\quad (\exists s)_\omega(\forall y)(\forall x)_0^y[s(x)s(y) \supset E_1(\mathbf{i}, x, y)] \vee \ldots \vee (\exists s)_\omega(\forall y)(\forall x)_0^y[s(x)s(y) \supset E_a(\mathbf{i}, x, y)].$$

If one defines for $1 \leqq c, d \leqq a$,

$$F_{c,d}(\mathbf{i}) : (\exists s)_\omega \cdot (\exists x)[s(x) \wedge E_c(\mathbf{i}, 0, x)] \wedge (\forall y)(\forall x)_0^y[s(x)s(y) \supset E_d(\mathbf{i}, x, y)],$$

then clearly each disjunct of (4) is equivalent to a disjunction of $F_{c,d}$'s. Therefore,

$$(5)\qquad \bigvee_{1 \leqq c,d \leqq a} F_{c,d}(\mathbf{i})$$

holds for all $\mathbf{i}$.

Suppose now that $F_{c,d}(\mathbf{i}) \wedge F_{c,d}(\mathbf{j})$. Then, by definition of $F_{c,d}$ and by (3) there are $x_1 < x_2 < x_3 < \ldots$ and $y_1 < y_2 < y_3 \ldots$ such that

$$\mathbf{i}(0) \ldots \mathbf{i}(x_1 - 1) \sim \mathbf{j}(0) \ldots \mathbf{j}(y_1 - 1),$$
$$\mathbf{i}(x_p) \ldots \mathbf{i}(x_{p+1} - 1) \sim \mathbf{j}(y_p) \ldots \mathbf{j}(y_{p+1} - 1), \qquad p = 1, 2, 3 \ldots .$$

By definition of $\sim$ and (1) it therefore follows that $A(\mathbf{i}) \equiv A(\mathbf{j})$. Thus if $F_{c,d}(\mathbf{i}) \wedge F_{c,d}(\mathbf{j})$, then $A(\mathbf{i}) \equiv A(\mathbf{j})$. Or restating this result,

$$(6)\qquad (\forall \mathbf{i})[F_{c,d}(\mathbf{i}) \supset A(\mathbf{i})] \vee (\forall \mathbf{i})[F_{c,d}(\mathbf{i}) \supset \sim A(\mathbf{i})], \qquad \text{for } 1 \leqq c,d \leqq a.$$

If now one defines the set Φ of pairs (c,d) by

$$(7)\qquad \Phi(c,d) \equiv \sim (\exists \mathbf{j})[A(\mathbf{j}) \wedge F_{c,d}(\mathbf{j})], \qquad \text{for } 1 \leqq c,d \leqq a,$$

then it follows by (5) and (6) that,

$$(8)\qquad \sim A(\mathbf{i}) \equiv \bigvee_{\Phi(c,d)} F_{c,d}(\mathbf{i}).$$

By (2), definition of $F_{c,d}$, and Lemmas 3, 4, 5 it follows that $F_{c,d}$ is of form

$$F_{c,d}(\mathbf{i}) \equiv (\exists s \mathbf{p} \mathbf{q}) \cdot I(0) \wedge (\forall t)\, J(t) \wedge (\exists t)\, M(t) \wedge (\exists^\omega t)\, s(t)$$

for some matrices $I[\mathbf{p}(0), \mathbf{q}(0)]$, $J[\mathbf{i}(t), s(t), \mathbf{q}(t), \mathbf{p}(t), \mathbf{q}(t'), \mathbf{p}(t')]$, $M[\mathbf{q}(t)]$. Note that $(\exists t)\, M(t) \wedge (\exists^\omega t)\, s(t)$ may be replaced by $(\exists^\omega t)[(\exists x)_0^t M(x) \wedge s(t)]$. Furthermore, $(\exists x)_0^t M(x)$ may be replaced by $r(t)$, if $r(0) \equiv F$, $r(t') \equiv [r(t) \vee M(t)]$ are conjoined to $I(0)$ and $J(t)$, respectively, and $(\exists r)$ is added to the prefix. Therefore, each $F_{c,d}(\mathbf{i})$ is equivalent to a Σ_1^ω-formula. By (8) and Lemma 7 it follows that $\sim A(\mathbf{i})$ is equivalent to a Σ_1^ω-formula, which concludes the proof of Lemma 9.

Note that by definition $F_{c,d} = E_c E_d E_d E_d \ldots$, and by (5) and (6) the set A is the finite union of all $F_{c,d}$ such that $\sim \Phi(c, d)$. Furthermore, the sets of words E_c are all multi-periodic. Thus our proof also yields

LEMMA 10. *Let $\mathscr{A}$ be a set of ω-sequences of states definable by a Σ_1^ω-formula. Then $\mathscr{A} = \mathscr{S}_1 \cup \ldots \cup \mathscr{S}_m$, whereby each $\mathscr{S}_k$ is of form $\mathscr{C}\mathscr{D}\mathscr{D}\mathscr{D}\ldots$, for multi-periodic sets $\mathscr{C}$, $\mathscr{D}$ of words.*

This provides a rather clear understanding of Σ_1^ω-definability, because multi-periodic sets of words have been investigated in automata theory.

5. Definability in SC

The following lemma may be proved by methods similar to those in [1] Lemma 1.

LEMMA 11. *To every formula $A(\mathbf{i})$ in* SC *one can obtain an equivalent formula $B(\mathbf{i})$ belonging to some Σ_n.*

Furthermore a Σ_1-formula can be transformed to an equivalent Σ_1^ω-formula (see end of proof of Lemma 9). Repeated application of Lemma 9 now clearly yields an equivalent Σ_1^ω-formula to every Σ_n-formula. Thus we obtain

THEOREM 1. *To every formula $A(\mathbf{i})$ of* SC *there is an equivalent formula $B(\mathbf{i})$ in Σ_1^ω.*

We add the following remarks:

1. Because of Lemma 10 this theorem provides a clear understanding of which relations $A(\mathbf{i}_1, \ldots, \mathbf{i}_n)$ on predicates are definable in SC.

2. It is easy to see that Σ_1^ω- and Σ_2-formulas define the same relations. Therefore, the hierarchy Σ_n, Π_n collapses at $n = 2$. This result cannot be improved much; the set $\mathfrak{U}$ consisting of all infinite i's is definable by a Σ_2-(a Σ_1^ω)formula, but not by a Π_2-formula.

3. Using Theorem 1, one easily shows that also formulas $A(\mathbf{i}, \mathbf{x}_1, \ldots, \mathbf{x}_m)$ of SC, containing free individual variables, have a normal form, namely,

$$(\exists \mathbf{r}) \cdot \mathrm{K}[\mathbf{r}(0)] \wedge (\forall \mathrm{t}) \mathrm{H}[\mathbf{i}(\mathrm{t}), \mathbf{r}(\mathrm{t}), \mathbf{r}(\mathrm{t}')] \wedge (\exists^\omega \mathrm{t}) \mathrm{L}[\mathbf{r}(\mathrm{t})] \wedge \mathrm{U}_1[\mathbf{r}(\mathbf{x}_1)] \wedge \ldots \wedge \mathrm{U}[\mathbf{r}(\mathbf{x}_m)].$$

This yields rather complete information on definability in SC. For example,

4. A conjecture of Robinson [10]: A relation $\mathscr{R}(\mathbf{x}_1, \ldots, \mathbf{x}_m)$ on natural numbers is definable in SC if and only if it is definable in $\mathrm{SC}_{\mathrm{fin}}$, which is like SC except that the variables i, j, r, ... range over finite sets of natural numbers. This follows by remark 3 and methods similar to those in the proof of Lemma 12, Section 6. Similarly, one shows a relation $\mathscr{R}(\mathbf{i}_1, \ldots, \mathbf{i}_m)$ on finite sets of natural numbers is definable in SC if and only if it is definable in $\mathrm{SC}_{\mathrm{fin}}$. For a complete discussion of definability in $\mathrm{SC}_{\mathrm{fin}}$ see Büchi [1].

5. Theorem 1 holds in a stronger version: there is an algorithm which to any formula $A(\mathbf{i})$ in SC yields an equivalent formula $B(\mathbf{i})$ in Σ_1^ω. See next section.

6. A Decision Method for SC

To obtain a method for deciding truth of sentences in SC we need a further lemma, whose proof again is typical for automata theory:

LEMMA 12. *There is an effective method for deciding truth of sentences A in* Σ_1^ω.

PROOF. Let $C(\mathbf{r})$ be a formula of form $K[\mathbf{r}(0)] \wedge (\forall t)H[\mathbf{r}(t), \mathbf{r}(t')] \wedge (\exists^\omega t)\, L[\mathbf{r}(t)]$. Suppose $\mathbf{r}$ is a k-tuple of predicates such that $C(\mathbf{r})$ holds. Then there are $x_1 < x_2 < \ldots$ such that $L[\mathbf{r}(x_1)], L[\mathbf{r}(x_2)], \ldots$. Because $\mathbf{r}$ has but a finite number of states, there must be a repetition $\mathbf{r}(x_p) = \mathbf{r}(x_q)$ of some state U. Therefore, $(\exists \mathbf{r})\, C(\mathbf{r})$ implies the assertion

(1) There are words $x = X_0X_1 \ldots X_a$ and $y = Y_1Y_2 \ldots Y_b$ of states and a state U such that $L[U]$, and $K[X_0] \wedge H[X_0, X_1] \wedge \ldots \wedge H[X_{a-1}, X_a] \wedge H[X_a, U]$, and $H[U, Y_1] \wedge H[Y_1, Y_2] \wedge \ldots \wedge H[Y_{b-1}, Y_b] \wedge H[Y_b, U]$.

Conversely (1) implies $(\exists \mathbf{r})\, C(\mathbf{r})$, because one has but to let $\mathbf{r} = xUyUyUy\ldots$. Thus, a method (I) which decides, for given propositional formulas K, H, L and given state U, whether or not (1) holds will also be a method for deciding truth of Σ_1^ω-sentences $(\exists \mathbf{r})\, C(\mathbf{r})$. Clearly such a method (I) can be composed from a method (II) which, for given propositional formula $H[X, Y]$ and given states V and W, decides whether or not

(2) There is a word $x = X_1X_2 \ldots X_a$ such that $H[V, X_1] \wedge H[X_1, X_2] \wedge \ldots \wedge H[X_{a-1}, X_a] \wedge H[X_a, W]$.

Let $n = 2^k$ be the number of states, and note that in a word $x = X_1X_2 \ldots X_a$ of length $a > n$ there must occur a repetition $X_p = X_q$, $p < q \leqq a$. Clearly if x satisfies (2), then so does the shorter word $y = X_1X_2 \ldots X_pX_{q+1}X_{q+2} \ldots X_a$. Therefore, to establish whether or not (2) holds, it suffices to check among the finitely many words x of length $\leqq n$. This remark clearly yields a method (II) for (2), whereby Lemma 12 is established.

Lemma 2 is proved in automata theory in a strong effective version. Also the proof of Lemma 6 actually yields the following result:

(a) Let $\backsim$ be a congruence of finite rank on words. Given a method for deciding $x \backsim y$ and a set of representatives $x_1, \ldots, x_a$ of the congruence classes of $\backsim$, one can construct a finite automaton $[\mathbf{I}, \mathbf{J}]$ and outputs $U_1, \ldots, U_a$ such that the congruence class of x_c is equal to the behavior of $[\mathbf{I}, \mathbf{J}, U_c)$. Clearly also Lemmas 3, 4, 5, 7, 8, 11 hold effectively. This leaves only the following two critical steps in the proof of the crucial Lemma 9:

(b) The Σ^0-formulas $E_c(\mathbf{i}, x, y)$, $c = 1, \ldots,$ a can be effectively constructed from $A(\mathbf{i})$.

(c) The relation $\Phi(c, d)$ on the finite set $\{1, \ldots, a\}$ can be effectively constructed from $A(\mathbf{i})$ (so that the disjunction (8) can be effectively obtained).

To prove (b) note that given A the definition of $\backsim$ in the proof of lemma 9 provides us with a method for deciding $x \backsim y$. Because we also have a bound 2^{2m^2} on the rank a of $\backsim$, it is possible to obtain a set of representatives $x_1, \ldots, x_a$ for the congruence classes. By (a) and Lemma 2 the assertion (b) follows.

To prove (c) we refer to the definition (7) of Φ in the proof of Lemma 9. By Lemma 8 one can actually construct a Σ_1^ω-formula equivalent to $A(j) \wedge F_{c,d}(j)$. Lemma 12 therefore provides a method for deciding whether or not $\Phi(c, d)$ holds. This takes care of (c). Thus also Lemma 9 holds effectively.

It now follows that Theorem 1 holds effectively; in particular, to every sentence A in SC one can construct an equivalent sentence in Σ_1^ω. Applying Lemma 12 again, we have

THEOREM 2. *There is an effective method for deciding truth of sentences in* SC.

The strength of this result is best seen by noting some very special cases which occur in the literature and have been obtained by rather divergent methods:

1. The decidability of Σ_2-sentences of SC contains the result of Friedman [4], and implies the existence of various other algorithms of finite automata theory as programmed by Church [2]. It also implies some of the results of Wang [11].

2. In SC one can define $x = y$, $x < y$, $x \equiv y \pmod{k}$ (for $k = 1, 2, \ldots$). The decidability of SC therefore considerably improves a result of Putnam [7].

3. In SC one can define "i is finite". Theorem 2 therefore implies the decidability of SC_{fin}, which was also proved in Büchi [1], and according to Robinson [10] is due to A. Ehrenfeucht.

4. The decidability of the first order theory of [Nn, +, Pw] follows from Theorem 2 and improves the classical result of Presburger.

5. Theorem 2 is closely related to another classical result, namely, the decidability of the monadic predicate calculus of second order, proved first by Th. Skolem and later by H. Behmann. A modified form of Lemma 11 yields a rather simple solution to this problem.

7. Concluding Remarks: Unsolved Problems

A careful analysis of the decision method for SC would yield a complete axiom system for SC. The most interesting candidate for an axiom schema is that part of Lemma 1 which is used in the proof of Lemma 9, namely,

$$(\mathrm{Ax}) \qquad (\forall \mathbf{i})(\exists s)_\omega \cdot (\forall y)(\forall x)_0^y [s(x)s(y) \supset E(\mathbf{i}, x, y)] \vee (\forall y)(\forall x)_0^y [s(x)s(y) \supset \sim E(\mathbf{i}, x, y)]$$

for any formula $E(\mathbf{i}, x, y)$ in Σ^0.

Such an analysis also shows that the same method yields a decision about whether or not a sentence is true in SC_{per}, which is like SC except that the variables i, j, r, ... range over ultimately periodic sets of natural numbers. In particular it can be seen that (Ax) also holds in SC_{per}. However this is not shown by using Ramsey's theorem; rather one uses the fact that every element c of a finite semi-group has a power c^n which is idempotent. These remarks outline a proof of

THEOREM 3. *A sentence A is true in* SC_{per} *if and only if it is true in* SC.

Using predicates i as binary expansions of real numbers, one obtains as a corollary to Theorems 2 and 3

THEOREM 4. *The first order theories of* [Re, +, Pw, Nn] *and* [Ra, +, Pw, Nn] *are arithmetically equivalent, and decidable.*

Here Re, Ra, Nn, Pw stand for the sets of non-negative reals, rationals, integers, integral powers of 2, respectively.

It is interesting to note that SC becomes undecidable if the function $2x$ is added (Robinson [10]). Also in case monadic predicate quantification is replaced in SC by quantification over monadic functions, all recursive relations become definable (Gödel).

Problem 1. Let SC^2 be like SC, except that the functions $2x+1$ and $2x+2$ are taken as primitives in place of $x+1$. Is SC^2 decidable?

This is of some interest, because the functions $2x+1$ and $2x+2$ can be interpreted as the right-successor functions x1 and x2 on the set of all words on two generators 1 and 2.

Problem 2. Let $SC(\alpha)$ be like SC, except that the domain of individuals is the ordinal α, and the well ordering on α is added as a primitive. Is $SC(\omega^2)$ decidable?

As outlined in the introduction, Theorem 2 may be interpreted as a method for deciding whether or not a given finite automaton satisfies a given condition in SC.

Problem 3. Is there a solvability algorithm for SC, i.e., is there a method which applies to any formula $C(\mathbf{i}, \mathbf{u})$ of SC and decides whether or not there is a finite automata recursion $A(\mathbf{i}, \mathbf{r}, \mathbf{u})$ which satisfies the condition C (i.e., $A(\mathbf{i}, \mathbf{r}, \mathbf{u}) \supset C(\mathbf{i}, \mathbf{u})$)?

REFERENCES

[1] BÜCHI, J. R. "Weak Second Order Arithmetic and Finite Automata", *Zeitschrift für Math. Log. und Grundl. der Math.*, 6 (1960), pp. 66–92.

[2] CHURCH, ALONZO. "Application of Recursive Arithmetic to the Problem of Circuit Synthesis", *Notes of the Summer Institute of Symbolic Logic*, Cornell, 1957, pp. 3–50, and "Application of Recursive Arithmetic in the Theory of Computing and Automata", Notes: *Advanced Theory of the Logical Design of Digital Computers*, U. of Michigan Summer Session, 1959.

[3] COPI, I. M., C. ELGOT, and J. B. WRIGHT. "Realization of Events by Logical Nets", *Journal of the Association for Computing Machinery*, Vol. 5, No. 2, April, 1958.

[4] FRIEDMAN, JOYCE. "Some Results in Church's Restricted Recursive Arithmetic", *Journal of Symbolic Logic*, 22, pp. 337–342 (1957).

[5] KLEENE, S. C. "Representation of Events in Nerve Nets and Finite Automata", *Automata Studies*, Princeton University Press, 1956, pp. 3–41.

[6] MYHILL, JOHN. "Finite Automata and Representation of Events", WADC Report TR 57–624, *Fundamental Concepts in the Theory of Systems*, October 1957, pp. 112–137.

[7] PUTNAM, HILLARY. "Decidability and Essential Undecidability", *Journal of Symbolic Logic*, 22 (1957), pp. 39–54.

[8] RABIN, M., and D. SCOTT. "Finite Automata and their Decision Problems", *IBM Journal*, April 1959, pp. 114–125.

[9] RAMSEY, F. P. "On a Problem of Formal Logic", *Proc. London Math. Soc.*, (2) 30 (1929), pp. 264–286.

[10] ROBINSON, R. M. "Restricted Set-theoretical Definitions in Arithmetic", *Proc. Am. Math. Soc.*, 9 (1958), pp. 238–242.

[11] WANG, HAO. "Circuit Synthesis by Solving Sequential Boolean Equations", *Zeitschrift für Math. Log. und Grundl. der Math.*, 5 (1959), pp. 291–322.