

## WEAK SECOND-ORDER ARITHMETIC AND FINITE AUTOMATA<sup>1)</sup>

By J. RICHARD BÜCHI in Ann Arbor, Michigan

### Introduction

The formalism of regular expressions was introduced by S. C. KLEENE [6] to obtain the following basic theorems.

*Synthesis. To every regular expression  $\mathfrak{E}$  one can effectively obtain a finite automaton  $\mathfrak{A}$  with binary output  $\mathfrak{U}$  such that  $\mathfrak{E}$  denotes the behavior of  $\langle \mathfrak{A}, \mathfrak{U} \rangle$ .*

*Analysis. To every finite automaton  $\mathfrak{A}$  with binary output  $\mathfrak{U}$  one can effectively construct a regular expression  $\mathfrak{E}$  such that the behavior of  $\langle \mathfrak{A}, \mathfrak{U} \rangle$  is denoted by  $\mathfrak{E}$ .*

For simplified expositions of KLEENE's theory see COPI-ELGOT-WRIGHT [4], RABIN-SCOTT [13], and MYHILL [8].

It will be shown here that a more conventional formalism, a weak second-order arithmetic, can be used in place of the formalism of regular expressions. Our theorems 1, 2 section 4 are equivalent to KLEENE's synthesis and analysis theorems. This result is of interest for automata theory because formulas of weak second-order arithmetic seem to be more convenient than regular expressions for formalizing conditions on the behavior of automata. In addition, our synthesis and analysis theorems yield rather complete information on the strength of weak second-order arithmetic (see Section 5), thus providing an example of applying automata theory to logic.

### § 1. Notations and Terminology

The following letters (possibly with subscripts) will be used as syntactic variables with range as indicated,

$x, y, z, t$	denote <i>individual variables</i>
$X, Y, Z$	denote <i>propositional variables</i>
$i, j, r, s, u$	denote <i>monadic predicate variables</i>
$A, B, C, D, \dots$	denote <i>formulas of propositional calculus</i>
$\mathfrak{A}, \mathfrak{B}, \mathfrak{C}, \mathfrak{D}$	denote <i>formulas</i> .

---

<sup>1)</sup> The author wishes to thank Dr. J. B. WRIGHT for many stimulating discussions. Some of the results were announced in the *Notices*, American Mathematical Society. (Decision Problems of Weak Second-Order Arithmetics and Finite Automata. Prelim. Report, Part I, Vol. 5, No. 7, December 1958.) This work was done on contract with the Office of Naval Research, Office of Ordnance Research, Army Signal Corps, and with the assistance of a grant from the National Science Foundation.

Besides the already mentioned (countable) lists of variables our formalism contains the primitive symbols  $\wedge, \vee, \neg, \supset, \equiv, [, ], \forall, \exists, (, ), 0, ' .$  The formulas of propositional calculus (p. f.'s) are obtained in the conventional manner from  $\wedge, \vee$  and propositional variables by means of the connectives and  $[, ]$ . Examples of atomic formulas are  $i(0'), i(x), r_2(t'')$ . Matrices are obtained by replacing all propositional variables in a p. f.  $A$  by atomic formulas. Formulas (of restricted, i. e. monadic, second-order-predicate calculus) are obtained in the conventional manner from matrices by applying individual — and predicate — quantifiers, and propositional connectives. A sentence is a formula without free variables.

A method for indicating occurrences of free variables is explained by the following examples. " $D[X, Y, Z]$ " denotes a p. f. in which the indicated propositional variables but no others may occur. " $D[i(0'), i(x), s(t')]$ " denotes the matrix obtained by making the obvious substitutions in  $D[X, Y, Z]$ , and may be abbreviated as " $D(0, x, t)$ ". " $\mathfrak{C}(i_1, i_2, t)$ " denotes a formula in which the variables  $i_1, i_2, t$  but no others may have free occurrences. We will also abbreviate " $[\sim A]$ " by " $\tilde{A}$ ", and " $[A \wedge B]$ " by " $AB$ ".

We will often deal with  $n$ -tuples of objects. The symbol " $\frown$ " denotes concatenation of  $n$ -tuples. Thus for example " $H_1 \frown H_2 \frown H_1$ " and " $i_1 \frown i_2 \frown i_3 \frown i_4$ " denote respectively a 3-tuple of matrices and a 4-tuple of predicate variables. The second component of the 3-tuple  $a \frown b \frown c$  is  $b$ . The  $n$ -tuple  $2 \frown 2 \frown 2 \dots \frown 2$  will also be denoted by " $n$ ". We next explain the use of a vector-notation which will considerably condense the presentation and make it more comprehensible.

$X^n, Y^n, Z^n$	denote $n$ -tuples of propositional variables
$i^n, j^n, r^n, s^n$	denote $n$ -tuples of monadic predicate variables
$A^n, B^n, \dots$	denote $n$ -tuples of propositional formulas.

For example, " $r^m(t') \equiv H^m[r^m(t), i^n(t)]$ " stands for the  $m$ -tuples of matrices whose components are

$$\begin{aligned} r_1(t') &\equiv H_1[r_1(t), \dots, r_m(t), i_1(t), \dots, i_n(t)], \\ &\vdots \\ r_m(t') &\equiv H_m[r_1(t), \dots, r_m(t), i_1(t), \dots, i_n(t)]. \end{aligned}$$

" $X^n \frown Y$ " stand for the  $(n+1)$ -tuple  $X_1 \frown X_2 \frown \dots \frown X_n \frown Y$ . At some places a notation for a  $n$ -tuple of propositional formulas is used ambiguously to denote the conjunction of the p. f.'s occurring in the  $n$ -tuple. Furthermore the superscripts on notations for  $n$ -tuples are often omitted; in such cases it is clear from the context how they are to be restored.

*Interpretation:* We will not make use of any deductive structure on the syntactic frame. However, propositional formulas and formulas will be interpreted. Because the interpretations are quite conventional we will not state rigorous definitions. Furthermore, we will make ambiguous use of the syntactic notations, using them at times also for reference to the interpretation. The following list will explain additional notations and terminology.

$\wedge, \vee$	truth values false, true
$\mathbf{\Lambda}^n$	the $n$ -tuple $\mathbf{\Lambda} \cap \dots \cap \mathbf{\Lambda}$
predicate	function from natural numbers to $\{\mathbf{\Lambda}, \mathbf{\vee}\}$ (set of natural numbers)
special predicate	predicate which is ultimately false (finite set of natural numbers)
$n$ -predicate	$n$ -tuple of predicates
special $n$ -predicates	$n$ -predicate with special components ( $n$ -predicate which is ultimately $\mathbf{\Lambda}^n$ ).

Further notations are introduced in the various sections.

## §2. Weak second-order arithmetic

We now consider the following interpretation of the primitive symbols of the restricted second-order system described in §1.

$\mathbf{\Lambda}, \mathbf{\vee}$	false, true
$\mathbf{\wedge}, \mathbf{\vee}, \mathbf{\sim}, \mathbf{\supset}, \mathbf{=}$	the usual truth functions
$0, '$	the natural number zero, the successor-function on natural numbers
individual variables	range over natural numbers
predicate variables	range over <i>special</i> predicates (finite sets of natural numbers)
$(\exists x), (\forall x)$	there is a n. n. $x$ , for all n. n.'s $x$
$(\exists i), (\forall i)$	there is a <i>special</i> predicate $i$ , for all <i>special</i> predicates $i$ .

This leads in a conventional manner to a definition of *satisfaction* for formulas, and *truth* for sentences. The resulting interpreted system will be called *weak second-order arithmetic* (W. 2. A). The notation " $\hat{i}\mathfrak{U}(i)$ " will be used for the set of all special predicates  $i$  on natural numbers which satisfy the formula  $\mathfrak{U}(i)$ . The notations " $\hat{x}\mathfrak{U}(x)$ ", " $\hat{i}\mathfrak{U}(i^n)$ " are used similarly. The formula  $\mathfrak{U}(i)$  defines the set  $\hat{i}\mathfrak{U}(i)$  in W. 2. A.  $\mathfrak{U}(i, x)$  is equivalent to  $\mathfrak{B}(i, x)$  if  $\hat{i}\hat{x}\mathfrak{U}(i, x) = \hat{i}\hat{x}\mathfrak{B}(i, x)$ , i. e., if  $(\forall i)(\forall x). \mathfrak{U}(i, x) \equiv \mathfrak{B}(i, x)$  holds in W. 2. A.

Lemma 1. In W. 2. A., to every formula  $\mathfrak{U}(i^n)$  one can effectively construct a formula  $\mathfrak{U}^*(i^n)$  such that  $\hat{i}\mathfrak{U}^*(i^n) = \hat{i}\mathfrak{U}(i^n)$  and  $\mathfrak{U}^*(i^n)$  is of the form

$$(j) \cdot K[j(0)] \wedge (\forall t) B[i(t), j(t), j(t')],$$

whereby  $j$  is a  $r$ -tuple  $j_1 \cap \dots \cap j_r$  of predicate variables and  $(j)$  is a prefix of quantifiers  $(\exists j_r)$  and  $(\forall j_r)$ .

Proof. It is clear that one can construct a formula  $\mathfrak{B}(x)$  such that 0 is the only number satisfying  $\mathfrak{B}(x)$ . Now if 0 occurs in  $\mathfrak{U}(i)$  and  $\mathfrak{D}(i, x)$  is obtained by substituting  $x$  for 0 in  $\mathfrak{U}(i)$ , then

$$\mathfrak{U}(i) \equiv (\exists x) [\mathfrak{B}(x) \wedge \mathfrak{D}(i, x)]$$

holds for  $i$ . Thus we have obtained a formula  $\mathfrak{U}_1(i)$  equivalent to  $\mathfrak{U}(i)$  in which 0 does not occur. The next step is to construct from  $\mathfrak{U}_1(i)$  an equivalent formula  $\mathfrak{U}_2(i)$

in which no iterations of “'” occur and no “'” occurs in argument places of  $i$ . This is easily accomplished by introduction of new predicate variables which are appropriately quantified. Then  $\mathfrak{C}_3(i)$  is obtained by passing to prenex form.

Next we repeatedly apply to  $\mathfrak{C}_3(i)$  identities of the form

$$\begin{aligned} & (\text{prefix 1}) (\exists x) (\text{prefix 2}) A(x, \dots) \\ & \cdot \equiv \cdot (\text{prefix 1}) (\exists j) (\forall x) (\text{prefix 2}) (\exists y) [j(y) \wedge j(x) \supset A(x, \dots)] \end{aligned}$$

and their duals, to obtain a formula  $\mathfrak{C}_4(i)$  equivalent to  $\mathfrak{C}(i)$  which is of the form

$$\mathfrak{C}_4(i): \quad (\text{predicate prefix}) (\text{individual prefix}) [\text{Matrix}].$$

Because the matrices are constructed from monadic predicate variables only we can make use of BEHMANN'S [1] device of moving individual quantifiers into the matrix of  $\mathfrak{C}_4(i)$ . The result is a formula  $\mathfrak{C}_5(i)$  equivalent to  $\mathfrak{C}(i)$ , and of form

$$\mathfrak{C}_5(i): (\text{predicate prefix}) \bigwedge_{\nu} [(\exists t) D_{\nu}(t) \vee (\forall t) E_{\nu,1}(t) \vee \dots \vee (\forall t) E_{\nu,n}(t)].$$

Now we note that predicate quantification is over special predicates, and therefore identities of the form

$$(\forall t) E(t) \cdot \equiv \cdot (\forall j) \cdot \tilde{j}(0) \vee (\exists t) [j(t) \tilde{j}(t') E(t)]$$

hold in W. 2. A. If we accordingly replace each constituent  $(\exists t) E_{\nu,\mu}(t)$  in  $\mathfrak{C}_5(i)$ , making use of different  $j$ 's, and then move the newly introduced predicate-quantifiers into the prefix, the result is a formula  $\mathfrak{C}_6(i)$  equivalent to  $\mathfrak{C}(i)$  and of the form,

$$\mathfrak{C}_6(i): \quad (\text{predicate prefix}) \bigwedge_{\nu} [A_{\nu}(0) \vee (\exists t) B_{\nu}(t)].$$

Now we remark again that predicate quantification is over special predicates, and therefore identities of the form

$$A(0) \vee (\exists t) B(t) \cdot \equiv \cdot (\exists j) \cdot [\tilde{A}(0) \supset j(0)] \wedge (\forall t) [j(t) \tilde{j}(t') \supset B(t)]$$

hold in W. 2. A. Accordingly we replace each  $A_{\nu}(0) \vee (\exists t) B_{\nu}(t)$  in  $\mathfrak{C}_6(i)$ , making use of different  $j$ 's, and then move the new predicate quantifiers into the prefix. The result is a formula  $\mathfrak{C}^*(i)$  equivalent to  $\mathfrak{C}(i)$  and of the form required in lemma 1.

### § 3. Finite automata

We will now define automata as syntactic entities. We refer to § 6 for motivation. In § 6, our concept also is compared with other definitions of “finite automaton”.

**Definition 1.** A (finite)  $n/m$ -automaton with input  $X^n$  and transit  $Y^m$  is a  $2m$ -tuple  $E^m \frown H^m [Y^m, X^n]$  of propositional formulas. (Note that no propositional variables occur in  $E^m$ .) A (binary) output of a  $n/m$ -automaton with transit  $Y^m$  is a propositional formula  $U[Y^m]$ .

The transition-recursion of an automaton  $E^m \frown H^m$  is the  $2m$ -tuple of matrices

$$\begin{aligned} r^m(0) & \equiv E^m, \\ r^m(t') & \equiv H^m[r^m(t), i^n(t)], \end{aligned}$$

it defines recursively a functional which will be denoted by  $r = \zeta(E, H, i)$ .

The *output-recursion* of an automaton with output  $E^m \wedge H^m \wedge U$ , is obtained by adding the matrix  $u(t) \equiv U[r(t)]$  to the transition-recursion. It defines recursively a functional which will be denoted by  $u = \psi(E, H, U, i)$ .

The "input to output" functional  $u = \psi(E, H, U, i)$  might be defined to be the behavior of the automaton with output. However, this is inconvenient for establishing relations to W. 2. A., because in general  $u = \psi(i)$  will not be special even for special  $i$ . We therefore will deal with a special sort of output only. It will be seen in § 6 that this is not essentially a restriction.

**Definition 2.**  $U[Y^m]$  is a *special output* of the automaton  $E^m \wedge H^m$  if  $U[Y] \equiv U[H[Y, \Lambda]]$ .

It is easy to see that in case  $U$  is special the output recursion defines an operator  $\psi$  such that  $\psi(i)$  or  $\sim\psi(i)$  is special whenever  $i$  is special. This makes the operator definition of behavior manageable in W. 2. A. However, it is more convenient to work with the set  $\beta$  consisting of all special  $i$  for which  $\sim\psi(i)$  is special, rather than with  $\psi$  directly. This leads to the following definition of behavior, which is closely related to KLEENE's [6] (see § 7).

**Definition 3.** The *behavior*  $\beta(E^m, H^m, U)$  of a  $n/m$ -automaton with special output is the set of all special  $n$ -predicates  $i^n$  for which the predicate  $u = \psi(E, H, U, i)$ , determined by the output-recursion, is ultimately true, i. e.,  $(\exists x) (\forall t)_x^\infty u(t)$ .

By the *length* of a special  $n$ -predicate  $i^n$  we mean the smallest number  $x$  such that  $i^n(t) \equiv \Lambda^n$  for  $t > x$ . By definitions 2, 3 one easily proves,

(\*) If  $U$  is a special output of the automaton  $E^m \wedge H^m$ , and if  $i^n$  is a  $n$ -predicate of length  $l$ , then  $i \in \beta(E, H, U) \equiv U[r(l+1)]$  whereby  $r = \zeta(E, H)$  is given by the transition-recursion of  $E \wedge H$ .

**Lemma 2.** If  $U$  is a special output of the  $n/m$ -automaton  $E^m \wedge H^m$  then so is  $\sim U$ . Furthermore the behaviors  $\beta(E, H, U)$  and  $\beta(E, H, \sim U)$  are complementary subsets of the set of all special  $n$ -predicates.

**Proof.** That  $\sim U$  again is special follows directly by definition 2. That the behaviors of  $U$  and  $\sim U$  are complementary is best seen by referring to (\*).

**Definition 4.** An  $n/m$ -automaton with output,  $E^m \wedge H^m \wedge U$ , is in *expanded form* if  $H^m$  and  $U$  are of the form

$$\begin{aligned} H^m[Y_1, \dots, Y_m, X^n] &: Y_1 K_1^m[X^n] \vee \dots \vee Y_m K_m^m[X^n], \\ U[Y_1, \dots, Y_m] &: Y_1 \vee Y_2 \vee \dots \vee Y_k. \end{aligned}$$

**Lemma 3.** To every  $n/m$ -automaton  $E^m \wedge H^m \wedge U$  one can construct a  $n/k$ -automaton  $G^k \wedge L^k \wedge W$  which is in expanded form, and such that if  $U$  is special output of  $E^m \wedge H^m$  then  $W$  is special output of  $G^k \wedge L^k$  and the behavior  $\beta(G, L, W)$  is equal to  $\beta(E, H, U)$ .

**Proof.** We indicate the construction of  $G, L, W$  in case  $m = 2^2$ , and  $U[Y_1, Y_2] \equiv Y_1$ . The given automaton then consists of p. f.'s  $E_1, E_2, H_1[Y_1, Y_2, X^n], H_2[Y_1, Y_2, X^n]$ .

The first step in the construction is to obtain disjunctive forms for  $H_1$ ,  $H_2$ , and  $U$ . Let this be,

$$(1) \quad \begin{cases} H_1[Y_1, Y_2, X] \equiv Y_1 Y_2 A_{11}[X] \vee Y_1 \tilde{Y}_2 A_{12}[X] \vee \\ \quad \vee \tilde{Y}_1 Y_2 A_{13}[X] \vee \tilde{Y}_1 \tilde{Y}_2 A_{14}[X], \\ H_2[Y_1, Y_2, X] \equiv Y_1 Y_2 A_{21}[X] \vee Y_1 \tilde{Y}_2 A_{22}[X] \vee \\ \quad \vee \tilde{Y}_1 Y_2 A_{23}[X] \vee \tilde{Y}_1 \tilde{Y}_2 A_{24}[X], \end{cases}$$

$$(2) \quad U[Y_1, Y_2] \equiv Y_1 Y_2 \vee Y_1 \tilde{Y}_2$$

Now we let  $k = 2 \wedge 2 \wedge 2 \wedge 2$  and construct  $G^k$ ,  $H^k$ ,  $W$  as follows,

$$(3) \quad \begin{cases} G_1 : E_1 E_2, & G_3 : \tilde{E}_1 E_2, \\ G_2 : E_1 \tilde{E}_2, & G_4 : \tilde{E}_1 \tilde{E}_2, \end{cases}$$

$$(4) \quad \begin{cases} L_1[Z_1, Z_2, Z_3, Z_4, X] : Z_1 A_{11}[X] A_{21}[X] \vee Z_2 A_{12}[X] A_{22}[X] \vee \\ \quad \vee Z_3 A_{13}[X] A_{23}[X] \vee Z_4 A_{14}[X] A_{24}[X], \\ L_2[Z_1, Z_2, Z_3, Z_4, X] : Z_1 A_{11}[X] \tilde{A}_{21}[X] \vee Z_2 A_{12}[X] \tilde{A}_{22}[X] \vee \\ \quad \vee Z_3 A_{13}[X] \tilde{A}_{23}[X] \vee Z_4 A_{14}[X] \tilde{A}_{24}[X], \\ L_3[Z_1, Z_2, Z_3, Z_4, X] : Z_1 \tilde{A}_{11}[X] A_{21}[X] \vee Z_2 \tilde{A}_{12}[X] A_{22}[X] \vee \\ \quad \vee Z_3 \tilde{A}_{13}[X] A_{23}[X] \vee Z_4 \tilde{A}_{14}[X] A_{24}[X], \\ L_4[Z_1, Z_2, Z_3, Z_4, X] : Z_1 \tilde{A}_{11}[X] \tilde{A}_{21}[X] \vee Z_2 \tilde{A}_{12}[X] \tilde{A}_{22}[X] \vee \\ \quad \vee Z_3 \tilde{A}_{13}[X] \tilde{A}_{23}[X] \vee Z_4 \tilde{A}_{14}[X] \tilde{A}_{24}[X], \end{cases}$$

$$(5) \quad W[Z_1, Z_2, Z_3, Z_4] : Z_1 \vee Z_2.$$

By definition 4, and (3), (4), (5) it is clear that  $G \wedge L \wedge W$  is in expanded form. Next we obtain from (1) the identities

$$\tilde{H}_v[Y_1, Y_2, X] \equiv Y_1 Y_2 \tilde{A}_{v1}[X] \vee Y_1 \tilde{Y}_2 \tilde{A}_{v2}[X] \vee Y_1 \tilde{Y}_2 \tilde{A}_{v3}[X] \vee \tilde{Y}_1 \tilde{Y}_2 \tilde{A}_{v4}[X].$$

Together with (1) and (4) this yields,

$$(6) \quad \begin{cases} L_1[Y_1 Y_2, Y_1 \tilde{Y}_2, \tilde{Y}_1 Y_2, \tilde{Y}_1 \tilde{Y}_2, X] \equiv H_1[Y_1, Y_2, X] H_2[Y_1, Y_2, X], \\ L_2[Y_1 Y_2, Y_1 \tilde{Y}_2, \tilde{Y}_1 Y_2, \tilde{Y}_1 \tilde{Y}_2, X] \equiv H_1[Y_1, Y_2, X] \tilde{H}_2[Y_1, Y_2, X], \\ L_3[Y_1 Y_2, Y_1 \tilde{Y}_2, \tilde{Y}_1 Y_2, \tilde{Y}_1 \tilde{Y}_2, X] \equiv \tilde{H}_1[Y_1, Y_2, X] H_2[Y_1, Y_2, X], \\ L_4[Y_1 Y_2, Y_1 \tilde{Y}_2, \tilde{Y}_1 Y_2, \tilde{Y}_1 \tilde{Y}_2, X] \equiv \tilde{H}_1[Y_1, Y_2, X] \tilde{H}_2[Y_1, Y_2, X] \end{cases}$$

and from (2) and (5) we get,

$$(7) \quad W[Y_1 Y_2, Y_1 \tilde{Y}_2, \tilde{Y}_1 Y_2, \tilde{Y}_1 \tilde{Y}_2] \equiv U[Y_1 Y_2].$$

Now assume that  $U$  is a special output of  $E \wedge H$ . Let  $C[Y_1, Y_2]$  stand for  $Y_1 Y_2 \wedge Y_1 \tilde{Y}_2 \wedge \tilde{Y}_1 Y_2 \wedge \tilde{Y}_1 \tilde{Y}_2$ . Then using in order (7), definition 2, (7), (6) one obtains

$W[C[Y_1, Y_2]] \equiv W[L[C[Y_1, Y_2], \Lambda]]$ . This means that  $W[Z] \equiv W[L[Z, \Lambda]]$  holds for the values  $\vee \wedge \wedge \wedge \wedge \wedge$ ,  $\wedge \vee \wedge \wedge \wedge \wedge$ ,  $\wedge \wedge \vee \wedge \wedge \wedge$ , and  $\wedge \wedge \wedge \wedge \vee$  of  $Z$ . Because of the particular form (5) of  $W$  the restriction of the range of  $Z$  can be omitted, which by definition 2 means that  $W$  is special output of  $G \wedge L$ .

Next let  $i$  be any  $n$ -predicate and let  $r_1 \wedge r_2 = \zeta(E, H, i)$ ,  $s_1 \wedge s_2 \wedge s_3 \wedge s_4 = \zeta(G, L, i)$ , so that

$$(8) \quad \begin{cases} r(0) \equiv E, \\ r(t') \equiv H[r(t), i(t)], \end{cases} \quad (9) \quad \begin{cases} s(0) \equiv G, \\ s(t') \equiv L[s(t), i(t)]. \end{cases}$$

Using (3), (6), (8), (9) one shows by an induction on  $t$  that  $s_1(t) \equiv r_1(t) r_2(t)$ ,  $s_2(t) \equiv r_1(t) \bar{r}_2(t)$ ,  $s_3(t) \equiv \bar{r}_1(t) r_2(t)$ ,  $s_4(t) \equiv \bar{r}_1(t) \bar{r}_2(t)$  hold for all  $t$ . Because of (7) this yields,  $(\forall t) \cdot W[s(t)] \equiv U[r(t)]$ . By definition 3 it follows that  $\beta(G, L, W) = \beta(K, H, U)$ .

**Lemma 4.** Let  $E^m \wedge H^m$  be an  $n^2/m$ -atomaton in expanded form and let  $L^m$  be defined by,  $L^m[Y^m, X^n] : H^m[Y^m, X^n \vee] \vee H^m[Y^m, X^n \wedge]$ .

(a) If  $i^n \wedge j$  is any  $n^2$ -predicate and if  $r^m = \zeta(E, H, i^n \wedge j)$ , and  $s^m = \zeta(E, L, i^n)$ , then  $(t) [r(t) \supset s(t)]$ .

b) If  $i^n$  is any  $n$ -predicate, and  $s^m = \zeta(E, L, i^n)$ , and  $s_v(x') \equiv \vee$ , then there exists a special predicate  $j$  of length  $\leq x$  such that also  $r_v(x') \equiv \vee$ , in case  $r^m = \zeta(E, H, i^n \wedge j)$ .

**Proof.** By assumption  $H^n$  is of form

$$(1) \quad H^n[Y^m, X^n \wedge Z] : Y_1 K_1^m[X^n \wedge Z] \vee \dots \vee Y_m K_m^m[X^n \wedge Z].$$

Now let  $i^n \wedge j$  be any  $n^2$ -predicate and let  $r = \zeta(E, H, i^n \wedge j)$ ,  $s = \zeta(E, L, i)$ . Then by definition 3 and the construction of  $L$ ,

$$(2) \quad \begin{cases} r(0) \equiv E, \\ r(t') \equiv H[r(t), i(t) \wedge j(t)], \end{cases}$$

$$(3) \quad \begin{cases} s(0) \equiv E, \\ s(t') \equiv H[s(t), i(t) \vee] \vee H[s(t), i(t) \wedge] \end{cases}$$

hold for all  $t$ . Making use of (1), (2), (3) it follows by induction on  $t$  that  $(\forall t) [r(t) \supset s(t)]$ . Thus part (a) of the lemma is established.

Next let  $i$  be any  $n$ -predicate, let  $s = \zeta(E, L, i)$ , and suppose that  $s_v(x') \equiv \vee$ . Then by definition 3 the predicates  $i$  and  $s$  satisfy (3) for all  $t$ , so that by (1) we obtain

$$(4) \quad \begin{cases} s_\varrho(0) \equiv E_\varrho, \\ s_\varrho(t') \equiv \bigvee_{\substack{v=1, \dots, m \\ Y = \Lambda, \vee}} s_v(t) K_{\varrho, v}[i(t') \wedge Y], \end{cases} \quad \varrho = 1, \dots, m$$

Now we stepwise choose a sequence of truth values  $j(x), \dots, j(0)$  and a sequence of indices  $\nu_{x+1}, \dots, \nu_0$  according to the following specifications. I. Let  $\nu_{x+1}$  be  $\nu$ . II. If  $\nu_{y+1}$  has already been obtained choose  $\nu_y, j(y)$  such that  $s_{\nu_y}(y)$  and  $K_{\nu_{y+1}, \nu_y}[i^n(y) \wedge j(y)]$  hold. That these choices are possible clearly follows by the assumption  $s_v(x) \equiv \vee$  and (4). Let now the values  $r(0), \dots, r(x')$  be defined

from  $i(0), \dots, i(x), j(0), \dots, j(x)$  by the recursions (2), so that by (1),  $r_e(0) \equiv E_e$ ;  $r_e(t') \equiv \bigvee_{\mu=1, \dots, m} r_\mu(t) K_{e, \mu}[i(t) \wedge j(t)]$  for  $t \leq x$ . Then by using II one stepwise obtains  $r_{v_0}(0), K_{v_1, v_0}[i(0) \wedge j(0)], \dots, r_{v_x}(x), K_{v_{x+1}, v_x}[i(x) \wedge j(x)], r_{v_{x+1}}(x+1)$ . Therefore by I,  $r_v(x') \equiv V$ . Thus by extending the sequence  $j(0), \dots, j(x)$  letting  $j(t) \equiv \Lambda$  for  $t > x$ , we obtain a special predicate  $j$  such that  $j$  is of length  $\leq x$ , and  $r_v(x') \equiv V$  in case  $r = \zeta(E, H, i^n)$ . Therefore also part (b) of the lemma is established.

**Lemma 5.** *For every  $n/m$ -automaton with output  $E^m \wedge H^m \wedge U$  one can construct a number  $h$  and an output  $W$  such that*

- (a)  $U[Y] \supset W[Y]$ ,
- (b)  $Z_1 \equiv Y \wedge Z_2 \equiv H[Z_1, \Lambda] \wedge \dots \wedge Z_h \equiv H[Z_{h-1}, \Lambda] \supset U[Z_1] \vee \dots \vee U[Z_h]$ ,
- (c) *If  $U[Y] \supset U[H[Y, \Lambda]]$  then  $W$  is special output of  $E \wedge H$ .*

**Proof.** The construction of  $h$  and  $W[Y^m]$  from  $H^m[Y^m, X^m]$  and  $U[Y]$  proceeds by the following rules,

I. Let  $U_0[Y]$  be  $V$ , let  $U_1[Y]$  be  $U[Y]$ .

II. If  $U_k[Y] \equiv U_{k+1}[Y]$  is not tautologous,

let  $U_{k+2}[Y]$  be  $U_{k+1}[Y] \vee U_{k+1}[H[Y, \Lambda]]$ .

III. If  $U_k[Y] \equiv U_{k+1}[Y]$  is tautologous, let  $h$  be  $k$ , let  $W$  be  $U_k$ , and stop.

From this construction it is clear that  $U_0[Y] \supset U_1[Y]$ ,  $U_1[Y] \supset U_2[Y]$ ,  $\dots$ . Because there are only  $a = 2^{(2^m)}$  truth-functions on  $m$  arguments the construction must reach a stage  $k \leq a$  such that  $U_k$  and  $U_{k+1}$  denote the same truth function, i. e., such that  $U_k[Y] \equiv U_{k+1}[Y]$ . The next step in the construction then clearly must use the stoprule III. Thus the construction always ends in  $k \leq a$  steps, yielding an  $h$  and a  $W$ .

Suppose now for example that  $h$  turns out to be 3, so that  $W$  is  $U_3$  and  $U_3 \equiv U_4$ . Letting  $K[Y]$  stand for  $H[Y, \Lambda]$  it follows from the construction,

$$(1) \quad U_2[Y] \equiv U[Y] \vee U[K[Y]],$$

and

$$(2) \quad U_3[Y] \equiv U_2[Y] \vee U_2[K[Y]],$$

and

$$(3) \quad U_4[Y] \equiv U_3[Y] \vee U_3[K[Y]].$$

By (1),  $U[Y] \supset U_2[Y]$  and by (2),  $U_2[Y] \supset U_3[Y]$ . Therefore  $U[Y] \supset U_3[Y]$ . Because  $W$  is  $U_3$  this established part (a) of the lemma.

Next suppose  $Z_1 \equiv Y$ ,  $Z_2 \equiv K[Z_1]$ ,  $Z_3 \equiv K[Z_2]$ , and  $U_3[Y]$ . Then by (2),  $U_2[Z_1] \vee U_2[Z_2]$ . This yields by (1),  $U[Z_1] \vee U[Z_2] \vee U[Z_3]$ . Because  $U_3$  is  $W$  and  $K[\cdot]$  is  $H[\cdot, \Lambda]$  this established part (b) of the lemma.

Now suppose  $U[Y] \supset U[K[Y]]$ . Then by (1),  $U_2[Y] \supset U[K[Y]]$ , and also by (1),  $U[K[Y]] \supset U_2[K[Y]]$ . Therefore  $U_2[Y] \supset U_2[K[Y]]$ . From this, by using (2),  $U_3[Y] \supset U_3[K[Y]]$  is similarly obtained. But also  $U_3[K[Y]] \supset U_3[Y]$ , because of (3) and  $U_3 \equiv U_4$ . Thus,  $U_3[Y] \equiv U_3[K[Y]]$ . Because  $U_3$  is  $W$  and  $K[Y]$  is



$H[Y, \mathbf{A}]$  this means that  $W$  is a special output of  $E \smallfrown H$ . This establishes part (c) of the lemma.

**Lemma 6.** *To every  $n \smallfrown 2/m$ -automaton with special output  $E^m \smallfrown H^m \smallfrown U$  in expanded form one can construct a  $n/m$ -automaton with special output  $E^m \smallfrown L^m \smallfrown W$  such that for every special  $n$ -predicate  $i^n$ ,  $i \in \beta(G, L, W) \equiv (\exists j) [i \smallfrown j \in \beta(E, H, U)]$  and  $E \smallfrown L \smallfrown W$  is again in expanded form.*

**Proof.** The construction of  $L^m, W$  is as follows,

$$(1) \quad L[Y, X] : H[Y, X \smallfrown V] \vee H[Y, X \smallfrown \Lambda],$$

$$(2) \quad h, W[Y] : \text{apply construction of lemma 4 to } E \smallfrown L \smallfrown U.$$

Assume now that  $E \smallfrown H \smallfrown U$  is in expanded form and  $U$  is special. Then by definitions 2, 4,

$$(3) \quad U[Y] : Y_1 \vee Y_2 \vee \dots \vee Y_k,$$

$$(4) \quad U[Y] \equiv U[H[Y, \mathbf{A} \smallfrown \Lambda]].$$

By using (4), (3), (1), (3) in this order one shows  $U[Y] \supset U[L, [Y, \mathbf{A}]]$ . Therefore, by (2) and lemma 5 (c),  $W$  is a special output of  $E \smallfrown L$ .

Next assume  $i \smallfrown j \in \beta(E, H, U)$ , and let  $r = \zeta(E, H, i \smallfrown j)$  and  $s = \zeta(E, L, i)$ . Then by definition 3,  $(\exists x) (\forall t)_x^\infty U[r(t)]$ , and by (1), lemma 4(a),  $(\forall t) [r(t) \supset s(t)]$ . Therefore it follows by (3) that  $(\exists x) (\forall t)_x^\infty U[s(t)]$ . By (2) and lemma 5(a) this yields  $(\exists x) (\forall t)_x^\infty W[s(t)]$ . Therefore, by definition 3,  $i \in \beta(E, L, W)$ . Thus we have shown that  $(\exists j) [i \smallfrown j \in \beta(E, H, U)] \supset i \in \beta(E, L, W)$ .

Assume now that  $i \in \beta(E, L, W)$ , and let  $s = \zeta(E, L, i)$ . Then if  $l$  is the length of  $i$  it follows by (\*) that  $W[s(l+1)]$ . Because of (3) and lemma 5(b) this implies that  $U[s(l+1)] \vee \dots \vee U[s(l+h+1)]$ , and by (3), there are  $v, p \leq h$  such that  $s_v(l+p+1) \equiv V$ . Because  $E \smallfrown H$  is in expanded form and because of (1) we therefore can apply lemma 4(b) to conclude that there is a  $j$  of length  $\leq l+p$  such that  $r_v(l+p+1) \equiv V$ , for  $r = \zeta(E, H, i \smallfrown j)$ . Using (3) we obtain  $U[r(l+p+1)]$ . Now observe that  $i(t) \equiv \mathbf{A}$ , and  $j(t) \equiv \mathbf{A}$ , for  $t > l+p$ . Because  $U$  is special output of  $E \smallfrown H$  it therefore follows that  $(\forall t)_{l+p+1}^\infty U[r(t)]$ . By definition 3 this means that  $i \smallfrown j \in \beta(E, H, U)$ . Thus we have concluded the proof of lemma 6 by showing that  $i \in \beta(E, L, W) \supset (\exists j) [i \smallfrown j \in \beta(E, H, U)]$ .

#### §4. Analysis and synthesis

We begin by establishing a synthesis result for formulas of W. 2. A. which do not contain predicate-quantifiers. Using the lemmas of §2 and §3 one then easily extends the result.

**Lemma 7.** *To every formula  $\mathfrak{U}(i^n)$  of the form  $K[i(0)] \wedge (\forall t) B[i(t), i(t')]$  one can construct a  $n/m$ -automaton  $E^m \smallfrown H^m$  and a special output  $U[Y^m]$  such that  $\beta(E, H, U) = i \mathfrak{U}(i)$ .*

**Proof.** We first determine whether or not  $B[\mathbf{A}, \mathbf{A}]$ . In case  $\tilde{B}[\mathbf{A}, \mathbf{A}]$ , we take for  $E \smallfrown H$  any automaton and for  $U$  the output  $\mathbf{A}$ . Then clearly  $U$  is special output and  $\beta(E, H, U) = i \mathfrak{U}(i)$  are both empty. Thus in this case Lemma 7 is established.

Let us next consider the case  $B[\mathbf{A}, \mathbf{A}]$ . Then we take  $m$  to be  $n + 2$ , and  $\mathbf{E}^m$  to be  $\mathbf{V}^m$ .  $\mathbf{H}^m$  we define as follows.

$$\begin{aligned} H_\nu[\mathbf{Y}^n \cap Z_1 \cap Z_2, \mathbf{X}^n] &: X_\nu, & \text{for } \nu = 1, \dots, n \\ H_{n+1}[\mathbf{Y}^n \cap Z_1 \cap Z_2, \mathbf{X}^n] &: \Lambda, \\ H_{n+2}[\mathbf{Y}^n \cap Z_1 \cap Z_2, \mathbf{X}^n] &: Z_1 K[\mathbf{X}^n] \vee \tilde{Z}_1 Z_2 B[\mathbf{Y}^n, \mathbf{X}^n]. \end{aligned}$$

As output we choose the formula

$$U[\mathbf{Y}^n \cap Z_1 \cap Z_2] : Z_1 K[\mathbf{A}^n] \vee \tilde{Z}_1 Z_2 B[\mathbf{Y}^n, \mathbf{A}^n].$$

Noting that  $B[\mathbf{A}, \mathbf{A}]$ , and using definition 2, one easily checks that  $U$  is a special output of  $\mathbf{E} \cap \mathbf{H}$ . Furthermore, the transition-recursion of  $\mathbf{E} \cap \mathbf{H}$  is clearly equivalent to the recursion

$$\begin{aligned} r_\nu(0) &\equiv \mathbf{V}, & \nu = 1, \dots, n + 2 \\ r_{n+2}(1) &\equiv K[\mathbf{i}(0)], \\ r_\nu(t') &\equiv i_\nu(t), & \nu = 1, \dots, n \\ r_{n+1}(t') &\equiv \Lambda, \\ r_{n+2}(t'') &\equiv r_{n+2}(t') B[\mathbf{i}(t), \mathbf{i}(t')] \end{aligned}$$

and therefore the operator  $\mathbf{r}^m = \zeta(\mathbf{E}, \mathbf{H}, \mathbf{i}^n)$  can also be defined by

$$\begin{aligned} r_\nu(0) &\equiv \mathbf{V}, & \nu = 1, \dots, n + 2 \\ r(t') &\equiv i_\nu(t), & \nu = 1, \dots, n \\ r_{n+1}(t') &\equiv \Lambda, \\ r_{n+2}(t') &\equiv K[\mathbf{i}(0)] \wedge (\forall x)_0^{t'-1} B[\mathbf{i}(x), \mathbf{i}(x')], \end{aligned}$$

Consequently, the output-operator  $u = \psi(\mathbf{E}, \mathbf{H}, U)$  is defined by

$$\begin{aligned} u(0) &\equiv K[\mathbf{A}], \\ u(t') &\equiv K[\mathbf{i}(0)] \wedge (\forall x)_0^{t'-1} B[\mathbf{i}(x), \mathbf{i}(x')] \wedge B[\mathbf{i}(t), \mathbf{A}]. \end{aligned}$$

Because  $B[\mathbf{A}, \mathbf{A}]$  it follows that for any special  $\mathbf{i}$ ,  $u = \psi(\mathbf{E}, \mathbf{H}, U)$  is ultimately  $\mathbf{V}$  just in case  $K[\mathbf{i}(0)] \wedge (\forall t) B[\mathbf{i}(t), \mathbf{i}(t')]$ . I. e., by definition 3,  $\beta(\mathbf{E}, \mathbf{H}, U) = \hat{\mathbf{i}} \mathfrak{U}(\mathbf{i})$ .

**Theorem 1 (Synthesis).** *For every formula  $\mathfrak{U}(\mathbf{i}^n)$  of W. 2. A. one can construct a  $n/m$ -automaton  $\mathbf{E}^m \cap \mathbf{H}^m$  with special output  $U[\mathbf{Y}^m]$  such that  $\beta(\mathbf{E}, \mathbf{H}, U) = \hat{\mathbf{i}} \mathfrak{U}(\mathbf{i})$ , i. e., such that the behavior of  $\mathbf{E} \cap \mathbf{H} \cap U$  is just the set of special  $n$ -predicates which satisfy  $\mathfrak{U}(\mathbf{i})$ .*

**Proof.** Using Lemma 1 we first construct the formula  $\mathfrak{U}^*(\mathbf{i})$  equivalent to  $\mathfrak{U}(\mathbf{i})$ . Let us for example assume that  $\mathfrak{U}^*(\mathbf{i})$  is as follows,  $\mathfrak{U}^*(\mathbf{i})$ :

$$(\forall \mathbf{r}) (\exists \mathbf{s}) K[\mathbf{r}(0), \mathbf{s}(0)] \wedge (\forall t) B[\mathbf{i}(t), \mathbf{r}(t), \mathbf{s}(t), \mathbf{r}(t'), \mathbf{s}(t')].$$

Next we use the construction of Lemma 7 to obtain an automaton  $\mathbf{E}_1 \cap \mathbf{H}_1$  with special output  $U_1$  such that for special  $\mathbf{i}, \mathbf{r}, \mathbf{s}$ ,

$$(1) \quad \mathbf{i} \cap \mathbf{r} \cap \mathbf{s} \in \beta(\mathbf{E}_1, \mathbf{H}_1, U) \cdot \equiv \cdot K(0) \wedge (\forall t) B(t).$$

Using Lemma 3 we next construct  $E_2 \wedge H_2 \wedge U_2$  in expanded form and such that

$$(2) \quad \beta(E_2, H_2, U_2) = \beta(E_1, H_1, U_1).$$

By repeated application of the construction of Lemma 6 starting with  $E_2 \wedge H_2 \wedge U_2$  one obtains  $E_3 \wedge H_3 \wedge U_3$  such that for special  $i, r$ ,

$$i \wedge r \in \beta(E_3, H_3, U_3) \cdot \equiv \cdot (\exists s) [i \wedge r \wedge s \in \beta(E_2, H_2, U_2)]$$

and therefore by Lemma 2,

$$(3) \quad i \wedge r \in \beta(E_3, H_3, \tilde{U}_3) \equiv (\exists s) [i \wedge r \wedge s \in \beta(E_2, H_2, U_2)].$$

Because the complementation destroys the expanded form we are forced to use Lemma 3 again to obtain  $E_4 \wedge H_4 \wedge U_4$  in expanded form and such that

$$(4) \quad \beta(E_4, H_4, U_4) = \beta(E_3, H_3, \tilde{U}_3).$$

By repeatedly applying Lemma 6 we next construct  $E_5 \wedge H_5 \wedge U_5$  such that for every special  $i$ ,  $i \in \beta(E_5, H_5, U_5) \cdot \equiv \cdot (\exists r) \cdot i \wedge r \in \beta(E_4, H_4, U_4)$  and therefore by Lemma 2,

$$(5) \quad i \in \beta(E_5, H_5, \tilde{U}_5) \cdot \equiv \cdot \sim (\exists r) i \wedge r \in \beta(E_4, H_4, U_4).$$

From (1), ..., (5) it clearly follows that  $i \in \beta(E_5, H_5, U_5) \cdot \equiv \cdot \mathfrak{G}^*(i)$ . Because  $\mathfrak{G}^*$  is equivalent to  $\mathfrak{G}$  this shows that the behavior of  $E_5 \wedge H_5 \wedge U_5$  is  $\hat{i} \mathfrak{G}(i)$ .

We next obtain a rather strong converse to Theorem 1.

**Theorem 2 (Analysis).** *To every  $n/m$ -automaton with special output  $E^m \wedge H^m \wedge U$  one can construct a formula  $\mathfrak{G}(i^n)$  of W. 2. A. such that  $\hat{i} \mathfrak{G}(i) = \beta(E, H, U)$ , and such that furthermore  $\mathfrak{G}(i)$  is of the form  $(\exists j^p) \cdot K[j(0)] \wedge (\forall t) B[i(t), j(t), j(t')]$ .*

**Proof.** By definition 3 it is clear that for every special  $i$ ,

$$i \in \beta(E, H, U) \cdot \equiv \cdot (\exists r) [r(0) \equiv E \wedge (\forall t) [r(t') \equiv H[r(t), i(t)]] \wedge (\exists x) (\forall t)_x^\infty U[r(t)]].$$

However, the range of  $(\exists r)$  in this formula may not be restricted to special predicates. On the other hand, because  $U$  is assumed to be special, the formula may be slightly modified so that the range of  $(\exists r)$  can be restricted to special predicates. Namely, it is clear that for any special  $i$ ,

$$(1) \quad i \in \beta(E, H, U) \cdot \equiv \cdot (\exists r) (\exists x) \cdot r(0) \equiv E \wedge (\forall t)_x^0 [r(t') \equiv H(t)] \\ \wedge (\forall t)_x^\infty [i(t) \equiv \mathbf{A} \wedge U[r(t)]].$$

It remains to change this definition of  $\beta$  in W. 2. A. to one of the simple form required in Theorem 2. This is accomplished by using the following device for changing the individual quantification  $(\exists x)$  to a quantification  $(\exists j)$  over restricted predicates,

$$(2) \quad i \in \beta(E, H, U) \cdot \equiv \cdot (\exists r) (\exists j) \cdot (\forall t) [j(t') \supset j(t)] \wedge r(0) \equiv E \\ \wedge (\forall t) [j(t) \supset [r(t') \equiv H(t)]] \wedge (\forall t) [\tilde{j}(t) \supset [i(t) \equiv \mathbf{A} \wedge U[r(t)]]].$$

To see that (2) is correct in case  $(\exists r) (\exists j)$  is interpreted as ranging over special predicates we observe that the right sides of (1) and (2) are equivalent because of an obvious one-to-one relationship between numbers  $x$  and restricted predicates  $j$  satisfying  $(\forall t) [j(t') \supset j(t)]$ .

Thus the right side of (2) is a formula  $\mathfrak{G}(i)$  as required in Theorem 2.

## §5. Definability in W. 2. A.

We will use the notation " $[0, x + 1, \exists_0 i]$ " to denote W. 2. A. This is intended to indicate that we are dealing with an interpreted system which besides first-order quantification over natural numbers contains 0, the function  $x + 1$ , and quantification over special predicates. Similar notations for other interpreted systems, all containing first-order quantification over natural numbers, are used below. " $\exists_0 \zeta$ " indicates quantification over eventually constant monadic functions from natural numbers to natural numbers.

The following systems are known to be very strong in the sense that all recursively enumerable predicates are definable in each.

$$\begin{aligned} [0, =, x + 1, \exists_0 \zeta] & \quad \text{GÖDEL [5],} \\ [0, =, x + y, x \cdot y] & \quad \text{GÖDEL [5],} \\ [0, x + 1, 2x, \exists_0 i] & \quad \text{ROBINSON [14].} \end{aligned}$$

(In fact these systems are equivalent in the sense that the same predicates on natural numbers can be defined in each.) In contrast we will now show that  $[0, x + 1, \exists_0 i]$  is much weaker, in particular the only monadic predicates on natural numbers definable in W. 2. A. are those which are ultimately periodic.

**Definition 5.** A formula of W. 2. A. is said to be in *normal form* if it is of the following type,  $\mathfrak{G}(i^n, x_1, \dots, x_p)$ :

$$(\exists j^m) [K[j(0)] \wedge (\forall t) B[i(t), j(t), j(t')] \wedge A_1[j(x_1)] \wedge \dots \wedge A_p[j(x_p)]] .$$

**Theorem 3.** For every formula  $\mathfrak{G}(i^n, x_1, \dots, x_p)$  of W. 2. A. one can construct an equivalent formula  $\mathfrak{G}^*(i^n, x_1, \dots, x_p)$  which is in normal form, i. e., every predicate on numbers and special predicates definable in W. 2. A. is definable by a formula in normal form.

**Proof.** Suppose first that  $\mathfrak{U}(i^n)$  is a formula without free individual variables. Then by Theorem 1 one can construct an automaton  $E^{\wedge}H$  with special output  $U$  such that  $\beta(E, H, U) = \hat{i}\mathfrak{U}(i)$ . Next, by Theorem 2 one can construct a formula  $\mathfrak{U}^*(i)$  in normal form such that  $\hat{i}\mathfrak{U}^*(i) = \beta(E, H, U)$ . It follows,  $\hat{i}\mathfrak{U}(i) = \hat{i}\mathfrak{U}^*(i)$ . Thus,

(1) If  $\mathfrak{U}(i)$  does not contain free individual variables one can construct a normal  $\mathfrak{U}^*(i)$  equivalent to  $\mathfrak{U}(i)$ .

Let us next start with a formula  $\mathfrak{G}$  which contains free individual variables, say for example  $\mathfrak{G}(i, x_1, x_2)$ . Let  $\mathfrak{U}$  be defined thus,  $\mathfrak{U}(i, s_1, s_2)$ :

$$\begin{aligned} s_1(0) \wedge s_2(0) \wedge (\forall t) [s_1(t') \supset s_1(t)] \wedge (\forall t) [s_2(t') \supset s_2(t)] \wedge \\ \wedge (\forall t_1 t_2) [s_1(t_1) \tilde{s}_1(t'_1) s_2(t_2) \tilde{s}_2(t'_2) \supset \mathfrak{G}(i, t_1, t_2)] . \end{aligned}$$

It is then easy to see that  $\mathfrak{U}(i, s_1, s_2) \supset \mathfrak{G}(i, x_1, x_2)$ , in case  $x_1 + 1, x_2 + 1$  are respectively the length of the special predicates  $s_1, s_2$ . Restating this we obtain

$$\mathfrak{G}(i, x_1, x_2) \equiv (\exists s_1 s_2) [\mathfrak{U}(i, s_1, s_2) \wedge s_1(x_1) \tilde{s}_1(x'_1) \wedge s_2(x_2) \tilde{s}_2(x'_2)] ,$$

and therefore,

$$(2) \mathfrak{U}(i, x_1, x_2) \cdot \cdot \equiv \cdot (\exists s_1 s_2 r_1 r_2) \cdot \mathfrak{U}(i, s_1, s_2) \wedge (\forall t) [r_1(t) \equiv s_1(t')] \wedge \\ \wedge (\forall t) [r_2(t) \equiv s_2(t')] \wedge s_1(x_1) \tilde{r}_1(x_1) \wedge s_2(x_2) \tilde{r}_2(x_2).$$

Finally we use (1) to obtain  $\mathfrak{U}^*(i, s_1, s_2)$  in normal form and equivalent to  $\mathfrak{U}$ . Replacing in the right side of (2)  $\mathfrak{U}$  by  $\mathfrak{U}^*$  and performing some obvious shifts of quantifiers will then yield a  $\mathfrak{U}^*$  equivalent to  $\mathfrak{U}$  such that  $\mathfrak{U}^*$  is in normal form.

**Corollary 1.** *There is a procedure for deciding whether or not a sentence of W. 2. A. is true.<sup>1)</sup>*

**Proof.** Because of Theorem 3 it is sufficient to indicate a procedure which decides the truth of sentences  $\mathfrak{U}$  of form  $(\exists j) \cdot K[j(0)] \wedge (\forall t) B[j(t), j(t')]$ . We may furthermore assume that  $B[\mathbf{A}, \mathbf{A}]$ , because otherwise the sentence  $\mathfrak{U}$  is clearly false in W. 2. A.  $\mathfrak{U}$  then is equivalent to the assertion,

(1) There is an  $x$  and a sequence of states  $Y_0, \dots, Y_x$  such that  $K[Y_0], B[Y_0, Y_1], \dots, B[Y_{x-1}, Y_x], Y_x \equiv \mathbf{A}$ .

Note that in case  $x \geq k = \text{number of states of } j$  there must occur a repetition in the sequence  $Y_0, Y_1, \dots, Y_x$ , say  $Y_y \equiv Y_z$  for some  $0 \leq y < z \leq x$ . Then if  $Y_0, \dots, Y_y, \dots, Y_z, \dots, Y_x$  is a  $B$ -sequence, the shorter sequence  $Y_0, \dots, Y_y, Y_{z+1}, \dots, Y_x$  is still a  $B$ -sequence. Consequently the assertion (1) is equivalent to,

(2) There is an  $x \leq k = \text{number of states of } j$ , and there is a sequence of states  $Y_0, \dots, Y_x$  such that  $K[Y_0], B[Y_0, Y_1], \dots, B[Y_{x-1}, Y_x], Y_x \equiv \mathbf{A}$ .

Because there are only a finite number of sequences  $Y_0, \dots, Y_x, x \leq k$ , it is clear that one can effectively check whether or not (2) holds. Because  $\mathfrak{U}$  is equivalent to (1), and (1) is equivalent to (2), this establishes Corollary 1.

**Corollary 2.** *Every formula  $\mathfrak{U}(x)$  of W. 2. A. defines an ultimately periodic set  $\hat{x} \mathfrak{U}(x)$  of natural numbers, i. e., there are numbers  $l$  (phase) and  $p$  (period) such that  $(\forall t) [\mathfrak{U}(l + t + p) \equiv \mathfrak{U}(l + t)]$ .*

*Every ultimately periodic set of natural numbers is definable in W. 2. A. by a formula  $\mathfrak{U}(x)$ .*

**Proof.** By theorem 3 we may assume that  $\mathfrak{U}(x)$  is of the form  $\mathfrak{U}(x)$ :

$$(\exists j) \cdot K[j(0)] \wedge (\forall t) B[j(t), j(t')] \wedge A[j(x)].$$

Now, let  $Y_1, \dots, Y_a$  be those states  $Y$  of  $j$  for which  $A[Y]$ , and for  $v = 1, \dots, a$  let,  $\mathfrak{U}_v(x)$  stand for  $(\exists j) [K(0) \wedge (\forall t) B(t) \wedge j(x) \equiv Y_v]$ . Then clearly,

$$\mathfrak{U}(x) \equiv \mathfrak{U}_1(x) \vee \dots \vee \mathfrak{U}_a(x),$$

$$\mathfrak{U}_v(x) \equiv (\exists j) [K(0) \wedge (\forall t)_0^x B(t) \wedge j(x) \equiv Y_v] \wedge (\exists j) [j(0) \equiv Y_v \wedge (\forall t) B(t)]$$

hold in W. 2. A. Therefore, if  $Y_1, \dots, Y_b$  ( $b \leq a$ ) are those states among  $Y_1, \dots, Y_a$  for which  $(\exists j) [j(0) \equiv Y \wedge (\forall t) B(t)]$ , then

$$\mathfrak{U}(x) \equiv (\exists j) \cdot K(0) \wedge (\forall t)_0^x B(t) \wedge [j(x) \equiv Y_1 \vee \dots \vee j(x) \equiv Y_b]$$

holds in W. 2. A.

<sup>1)</sup> As R. L. VAUGHT remarks, this result can be obtained from a theorem of A. EHRENFUCHT; see ROBINSON [14].

Next let  $k$  = number of states of  $j$ , and let  $Y_1, \dots, Y_b, \dots, Y_k$  be the states of  $j$ , and let  $r_1, \dots, r_k$  be defined by the recursion

$$(1) \quad \begin{cases} r_v(0) = K[Y_v], & v = 1, \dots, k \\ r_v(t') = r_1(t) B[Y_1, Y_v] \vee \dots \vee r_k(t) B[Y_k, Y_v]. & v = 1, \dots, k \end{cases}$$

One then easily shows that

$$(\exists j) [K(0) \wedge (\forall t)_0^x B(t) \wedge [j(x) \equiv Y_1 \vee \dots \vee j(x) \equiv Y_b]]$$

holds if and only if  $[r_1(x) \vee \dots \vee r_b(x)]$ . Consequently,  $\hat{x}\mathfrak{U}(x) = \hat{x}[r_1(x) \vee \dots \vee r_b(x)]$ . Now  $r$  has  $2^k$  states, therefore a repetition must occur in  $r(0), \dots, r(2^k)$ , say  $r(l) \equiv r(l+p)$  whereby  $l+p \leq 2^k$  and  $0 < p$ . By (1) it then follows that  $r(l+t) \equiv r(l+p+t)$ , for all  $t$ , so that  $\hat{x}[r_1(x) \vee \dots \vee r_b(x)] = x\mathfrak{U}(x)$  is ultimately periodic with phase  $l$  and period  $p$ .

The second part of Corollary 2 is best shown by first obtaining definitions in W. 2. A. of the relations  $x = y$ ,  $x < y$ ,  $x \equiv y \pmod{p}$ , for fixed  $p$ .

Note also that the selection of  $Y_1, \dots, Y_b$  from  $Y_1, \dots, Y_a$  in the proof of Corollary 2 can be effectively made (by Corollary 1). As a result one can effectively find the phase and period of the set  $x\mathfrak{U}(x)$ .

By using similar methods to those employed in the proof of Corollary 2 one shows that every relation  $R(x, y)$  definable in W. 2. A. must be of the form

$$R(x, y) \equiv \bigvee_{v=1, \dots, a} [x \leq y \wedge A_v(x) \wedge B_v(y-x)] \vee \bigvee_{v=1, \dots, c} [y \leq x \wedge C_v(y) \wedge D_v(x-y)]$$

whereby  $A_v, B_v, C_v$ , are ultimately periodic. In particular  $y = f(x)$  is definable in W. 2. A. if and only if it is ultimately periodic, i. e., satisfies  $f(l+x+p) = f(l+x) + q$ , for some  $l, p, q$  (compare this with ROBINSON's [14] result on  $[0, x+1, 2x, \exists_0 i]$ ). However, the following result seems more informative.

**Corollary 3.** *If  $R(x, y)$  well-orders a subset of natural numbers and is definable by a formula  $\mathfrak{U}(x, y)$  of W. 2. A., then the type  $\alpha$  of  $R$  is less than  $\omega^2$ . Conversely, if  $\alpha$  is an ordinal less than  $\omega^2$  one can find a formula  $\mathfrak{U}(x, y)$  of W. 2. A. such that  $\hat{x}\hat{y}\mathfrak{U}(x, y)$  is an  $\alpha$ -well-ordering of all natural numbers.*

**Proof.** Suppose  $R(x, y)$  is a well-ordering of natural numbers of type  $\alpha$ , and for  $\eta \leq \alpha$  let  $A_\eta$  be the initial segment relative to  $R$  of type  $\eta$ . Then from a definition  $\mathfrak{U}(x, y)$  of  $R$  in W. 2. A. one can easily obtain definitions of the sets  $A_\eta$ , for  $\eta < \omega^2$ ; so that by Corollary 2 the sets  $A_\eta$ ,  $\eta < \omega^2$  are ultimately periodic. Now it is easy to see that there is no strictly increasing  $\omega^2$ -sequence of ultimately periodic sets of natural numbers. Consequently if  $R(x, y)$  is definable in W. 2. A. then its type  $\alpha$  must be less than  $\omega^2$ . To indicate a proof of the second part of Corollary 3, let us consider the case  $\alpha = \omega 3 + 2$ . Because one can define  $x \equiv y \pmod{3}$  and  $x < y$ , and  $x = y$  in W. 2. A., it is clear that one can also define a relation  $R(x, y)$  which well-orders the natural numbers in sequence, 6, 9, 12, ..., 1, 4, 7, ..., 2, 5, 8, ..., 0, 3.

It seems clear that one could use Theorem 3 also to investigate the nature of sets  $\mathfrak{U}(i)$  definable in W. 2. A. by formulas with free predicate variables. However, a rather concrete characterization of these sets of finite sets is given by Theorems 1 and 2, namely

**Corollary 4.** *The sets of  $n$ -tuples of finite sets definable in W. 2. A. are exactly the behaviors of special outputs of finite automata with  $n$ -ary input.*

We will terminate this section with the presentation of a first-order theory which is equivalent to W. 2. A. in a very strong sense. For this purpose note that

$$f(i) = \sum_{i(x)} (2^x)$$

defines a one-to-one mapping  $f$  from all special predicates (finite sets) of natural numbers onto all natural numbers.  $f(i) = y$  simply means that for  $l = \text{length of } i$ ,  $i(0) i(1) \dots i(l)$  is the binary expansion of  $y$ . This mapping  $f$  induces a natural one-to-one correspondence between relations on special predicates and relations on natural numbers, let us say that  $R(i_1, \dots, i_n)$  and  $S(x_1, \dots, x_n)$  are *adjoined* to each other in case  $R(i_1, \dots, i_n) \equiv S(f(i_1), \dots, f(i_n))$ , for all special  $i_1, \dots, i_n$ . Let furthermore " $Pw_2(x)$ " stand for " $x$  is a power of 2".

**Theorem 4.** *The first-order theory  $[=, +, Pw_2]$  is equivalent to the weak second-order theory  $[0, ', \exists_0 i]$  in the sense that a relation  $S(x_1, \dots, x_n)$  on natural numbers is definable in  $[=, +, Pw_2]$  if and only if its adjoined relation  $R(i_1, \dots, i_n)$  is definable in  $[0, ', \exists_0 i]$ . Moreover, from a definition  $\mathfrak{U}(x_1, \dots, x_n)$  of  $S$  one can effectively obtain a definition  $\mathfrak{D}(i_1, \dots, i_n)$  of  $R$ , and conversely.*

**Proof.** Let  $S(i, j, s)$  be the adjoined to  $x + y = z$ , i. e., " $S(i, j, s)$ " stands for " $f(i) + f(j) = f(s)$ ". Then by formalizing the procedure of adding natural numbers in binary expansion one obtains the following definition of  $S$  in the weak second-order theory  $[0, ', \exists_0 i]$ ,

$$S(i, j, s) : (\exists r) \cdot \tilde{r}(0) \wedge (\forall t) [r(t) \equiv [r(t) i(t) \vee r(t) j(t) \vee i(t) j(t)]] \\ \wedge (\forall t) [s(t) \equiv [r(t) \nabla i(t) \nabla j(t)]]$$

whereby " $X \nabla Y$ " stands for " $X \tilde{Y} \vee Y \tilde{X}$ ". Furthermore, if  $U(i)$  is the adjoined to  $Pw_2(x)$ ; then  $U(i) : (\exists x) (\forall t) [i(t) \equiv (t = x)]$  is a definition of  $U$  in  $[0, ', \exists_0 i]$ . To prove Theorem 4 in one direction it therefore is sufficient to indicate a translation of formulas  $\mathfrak{U}$  of  $[=, +, Pw_2]$  into formulas  $\mathfrak{U}^*$  of  $[0, ', U, S, \exists_0 i]$  such that  $\mathfrak{U}^*(i_1, \dots, i_n)$  and  $\mathfrak{U}(x_1, \dots, x_n)$  define adjoined relations. Because every formula  $\mathfrak{U}$  of  $[=, +, Pw_2]$  is easily changed to an equivalent one in which no iteration of  $+$  occurs, the following specifications yield the required translation.

1.  $[x_\nu + x_\mu = x_\rho]^*$  is  $S(i_\nu, i_\mu, i_\rho)$ ,
2.  $[Pw(x_\nu)]^*$  is  $U(i_\nu)$ ,
3.  $[\mathfrak{U} \wedge \mathfrak{D}]^*$  is  $\mathfrak{U}^* \wedge \mathfrak{D}^*$ ,
4.  $[\sim \mathfrak{U}]^*$  is  $\sim \mathfrak{U}^*$ ,
5.  $[(\exists x_\nu) \mathfrak{U}]^*$  is  $(\exists i_\nu) \mathfrak{U}^*$ .

To prove Theorem 4 in the other direction we note that

$$E(x, y) : Pw_2(x) \wedge (\exists uv) [(y = u + x + v) \wedge (u < v) \wedge [v = 0 \vee 2x \leq v]]$$

is a definition in  $[=, +, Pw_2]$  of “ $x$  is a power of 2, and  $x$  occurs in the representation of  $y$  as a sum of powers of 2”. It therefore is sufficient to indicate a translation of formulas  $\mathfrak{C}$  of  $[0, ', \exists_0 i]$  into formulas  $\mathfrak{C}^\circ$  of  $[=, +, Pw_2, E]$  such that  $\mathfrak{C}^\circ(x_1, \dots, x_n)$  and  $\mathfrak{C}(i_1, \dots, i_n)$  define adjointed relations. If one notes that  $i(x)$  means the same as  $E(x, f(i))$ , and that every formula  $\mathfrak{C}(i_1, \dots, i_n)$  can be changed to an equivalent one in which 0 does not occur and  $'$  only occurs in parts of form  $(\forall t)[j(t) \equiv i(t')]$ , then it is clear that the following specifications yield such a translation. We single out the individual variable  $t$  and divide the remaining ones in  $x_1, x_2, \dots; y_1, y_2, \dots$ . In  $[0, ', \exists_0 i]$  we make use of  $y_1, y_2, \dots$ , and  $t$  only (while it is intended that in  $[=, +, Pw_2, E]$  the range of the  $y$ 's is restricted to  $Pw_2$ ).

1.  $[i_\nu(y_\mu)]^\circ$  is  $E(y_\mu, x_\nu)$ ,
2.  $[(\forall t)[i_\mu(t) \equiv i_\nu(t')]]^\circ$  is  $[x_\nu = x_\mu + x_\mu \vee x_\nu = x_\mu + x_\mu + 1]$ ,
3.  $[\mathfrak{C} \wedge \mathfrak{D}]^\circ$  is  $\mathfrak{C}^\circ \wedge \mathfrak{D}^\circ$ ,
4.  $[\sim \mathfrak{C}]^\circ$  is  $\sim \mathfrak{C}^\circ$ ,
5.  $[(\exists i_\nu) \mathfrak{C}]^\circ$  is  $(\exists x_\nu) \mathfrak{C}^\circ$ ,
6.  $[(\exists y_\nu) \mathfrak{C}]^\circ$  is  $(\exists y_\nu) [Pw_2(y_\nu) \wedge \mathfrak{C}^\circ]$ .

By Theorem 4, Corollary 1, Corollary 4 one clearly obtains,

**Corollary 5.** *The first-order theory  $[=, +, Pw_2]$  is decidable. A relation  $R(x_1, \dots, x_n)$  is definable in this theory if and only if its adjointed  $S(i_1, \dots, i_n)$  is the behavior of a finite automaton with special output.*

One might attempt to prove Corollary 5 directly by extending PRESBURGER's [10] method for  $[=, +]$ . Corollary 1 would then follow by Theorem 4.

Let  $Q_2(x)$  stand for “ $x$  is a square”. Then by PUTNAM [11], the first-order theory  $[=, +, Q_2]$  is undecidable. It is interesting to compare this with Corollary 5, and one easily extends the results as follows,

**Theorem 5.**

*The first-order theory  $[=, +, P]$  is*

- a) *undecidable in case  $P(x)$  is ultimately a hyper-arithmetic-progression, i. e., in case  $P$  ultimately consists of the values of a polynomial of degree  $\geq 2$ ;*
- b) *decidable in case  $P(x)$  is ultimately a geometric progression.*

To better understand the jump in strength from  $[=, +]$  to  $[=, +, \cdot]$  it would be interesting to know whether there is a recursive predicate  $P(x)$  such that  $[=, +, P]$  is undecidable without admitting definitions for all recursive predicates.



## 6. Other concepts of finite automata

Clearly our concept of finite automata is very closely related to that of CHURCH [3]. What we have called the transition-recursion and output-recursion of an automaton (with output) are restricted recursions in his sense. While CHURCH allows instantaneous action (the input at time  $t$  directly affects the transit and output at the same time  $t$ ), it is just a matter of convenience that we have presented our theory in terms of automata with delayed action only; also our restriction to special outputs is inessential (see Section 7). The behavior-concept used implicitly by CHURCH is that of the operator  $u = \psi(i)$  defined by the output-recursion. We hope to show elsewhere how his synthesis procedure (Case II) can be extended to condition-formulas containing predicate-quantifiers.

Recursions like our output-recursions were first used by BURKS and WRIGHT [2]. Their concept of well-formed logical net is equivalent to our finite automata, except that instantaneous action occurs in logical nets.

In the following discussion it is intended that a  $n/m$ -automaton (whose input has  $n$  binary components) represents in coded form a  $2^n/2^m$ -automaton (whose input has one  $2^n$ -ary component. Let  $\Sigma_2 = \{\Lambda, V\}$ ,  $\Sigma_n = \Sigma_2 \times \cdots \times \Sigma_2$  ( $n$  factors). Under the conventional interpretation of propositional formulas  $E^m$  denotes an element  $\sigma \in \Sigma_m$ , while  $H^m[Y^m, X^n]$  may be interpreted to denote a function  $\tau: \Sigma_m \times \Sigma_n \rightarrow \Sigma_m$ . Thus, with an  $n/m$ -automaton  $E^m \wedge H^m[Y^m, X^n]$  is associated a system  $\langle \Sigma_n, \Sigma_m, \sigma, \tau \rangle$  called its *transition system*. The elements of  $\Sigma_n$  and  $\Sigma_m$  are respectively called *input states* and *transit-states*,  $\sigma$  is the *initial-state* and  $\tau$  the *transit-function* of the automaton. An output  $U[Y^m]$  may be interpreted to denote a subset of transit-states. These remarks are intended to indicate how our notion of finite automata (with output) is related to those used by MOORE [9], MYHILL [8], RABIN and SCOTT [13], and others. While our notion is syntactic these authors take automata to be what we called transition systems (with output set). The latter definition is not correct if the mathematical concept "finite automaton" is to correspond to physical structures. Two physical systems (two syntactic systems) can be quite different structurally (syntactically) and still display the same transition-system. Also from a purely mathematical point of view it is not advisable to identify automata with their transition-structure, as it then becomes impossible to rigorously state existence of algorithms as, for example, in our Theorems 1 and 2. The point is that an algorithm is better thought of as applying to and yielding syntactic entities.

We will now indicate how our theory of behavior can be extended to automata with many inputs, each of which may have any number of binary components. For this purpose it is convenient to extend propositional calculus so as to contain  $\omega$  lists, each list consisting of  $\omega$  propositional variables. A notation like " $H[X^n; Y^k; Z^h]$ " will be used to stand for a formula of (extended) propositional calculus in which the variables  $X, Y, Z$  and only these may occur, and furthermore the variables  $X$  belong to a first list, which precedes a second list to which the variables  $Y$  belong, which precedes a third list to which the variables  $Z$  belong.

**Definition 1'.** A (finite)  $n_1; \dots; n_k/m$ -automaton with inputs  $X_1^{n_1}, \dots, X_k^{n_k}$  and transit  $Y^m$  is a  $2m$ -tuple  $E^m \frown H^m[Y^m; X_1^{n_1}; \dots; X_k^{n_k}]$  of propositional formulas (no variables occur in  $E^m$ ).

A (binary) output of such an automaton is a propositional formula  $U[Y^m]$ , where  $Y^m$  is the transit of the automaton.

It is clear how one extends the notion of transition (output)-recursion, introduced in Section 3, to  $n_1; \dots; n_k/m$ -automata (with output). Also for the extension of the concept of special output we refer to Definition 2 in Section 3.

**Definition 3'.** The behavior  $\beta(E^m, H^m, U)$  of a  $n_1; \dots; n_k/m$ -automaton with special output is the  $k$ -ary relation which holds for the special  $n_1$ -predicate  $i_1, \dots, n_k$ -predicate  $i_k$ , just in case  $(\exists x)(\forall t)_x^\infty U[r(t)]$ , whereby  $r^m = \zeta(E, H, i_1; \dots; i_k)$  is obtained by the transition-recursion of  $E \frown H$ .

To obtain an extension of the analysis and synthesis Theorems 1 and 2, we extend W. 2. A. to contain  $\omega$  lists, each list containing  $\omega$  predicate variables. A notation like " $C(i; j)$ " will be used to denote a formula of (extended) W. 2. A. in which the variables  $i, j$  and only these may have free occurrences, and such that the variables  $i$  all belong to the same list which precedes a list containing all the variables  $j$ . A formula  $\mathfrak{G}(i_1^{n_1}; i_2^{n_2}; i_3^{n_3})$  of W. 2. A. defines a ternary relation  $\hat{i}_1; \hat{i}_2; \hat{i}_3 \mathfrak{G}(i_1; i_2; i_3)$  on special  $n_1$ -predicates,  $n_2$ -predicates,  $n_3$ -predicates.

**Theorem 1' (Synthesis).** To every formula  $\mathfrak{G}(i_1^{n_1}; \dots; i_k^{n_k})$  of (extended) W. 2. A. one can construct a  $n_1; \dots; n_k/m$ -automaton  $E^m \frown H^m$  with special output  $U$  such that the behavior  $\beta(E, H, U)$  is the  $k$ -ary relation  $\hat{i}_1; \dots; \hat{i}_k \mathfrak{G}(i_1; \dots; i_k)$ .

**Theorem 2' (Analysis).** To every  $n_1; \dots; n_k/m$ -automaton  $E^m \frown H^m$  with special output  $U$  one can construct a formula  $\mathfrak{G}(i_1^{n_1}; \dots; i_k^{n_k})$  of W. 2. A. such that the  $k$ -ary relation  $\hat{i}_1; \dots; \hat{i}_k \mathfrak{G}(i_1; \dots; i_k)$  is the behavior  $\beta(E, H, U)$ .

To prove Theorem 1' one at first simply disregards the grouping of the  $q = n_1 \cdot n_2 \cdots n_k$  variables in the formula  $\mathfrak{G}(i_1^{n_1}; \dots; i_k^{n_k})$  and constructs the  $q/m$ -automaton with special output according to Theorem 1. By properly grouping the  $q$  components of the input of this automaton one obtains a  $n_1; \dots; n_k/m$ -automaton with special output as required in Theorem 1'. Similarly one proves Theorem 2'.

It thus appears that the theory of behavior for automata with many inputs does not present any additional difficulties, in case one works with special predicates rather than finite strings of input states. The reason for this is that a  $k$ -tuple  $(i_1^{n_1}; \dots; i_k^{n_k})$  of special  $n$ -predicates may be reinterpreted as one special  $n_1 \cdot n_2 \cdots n_k$ -predicate. In contrast a  $k$ -tuple  $(x_1; \dots; x_k)$  of finite strings of input states can not readily be reinterpreted as one finite string of states. This seems to be the reason that no theory of behavior of automata with many inputs occurs in the literature (see section 7).

Our concept of finite  $n_1; \dots; n_k/m$ -automaton still is somewhat specialized in that the components of each input and the transit are binary (i. e., can take values in  $\Sigma_2 = \{\Lambda, V\}$  only). Because we allow many components to occur in an input

(transit) there is of course the possibility of indirectly dealing with automata whose sets of input states (transit states) have any finite cardinality, by way of binary coding. However, to obtain an entirely satisfactory theory of  $n_1; \dots; n_k/m$ -automata (for any numbers  $n_1, \dots, n_k, m$ ) one would have to replace  $\Sigma_2$  by  $\Sigma_2, \Sigma_3, \dots$  ( $\Sigma_k$  being a set of  $k$  elements), and one would have to set up a calculus  $PC_\omega$  to replace propositional-calculus ( $PC_2$ ), such that  $PC_\omega$  contains variables  $X_k$  ranging over  $\Sigma_k$  and provides names  $H_k[X_{k_1}, \dots, X_{k_a}]$  for all functions  $\tau: \Sigma_{n_1} \times \dots \times \Sigma_{n_a} \rightarrow \Sigma_{k_a}$ . To generalize the results obtained in Section 4 one would extend W. 2. A. to contain variables  $i_k$  ranging over monadic functions from natural numbers to  $\Sigma_k$  which ultimately take the fixed value  $\Lambda_k \in \Sigma_k$ . It seems clear that this program could be carried out, and moreover the form of the presentation in Section 4 would remain essentially unchanged. We did not present our results in this general version because no calculus  $PC_\omega$  is available in the literature.

## 7. Finite strings versus special predicates; regular relations

According to our definition the behavior  $\beta$  of an  $n/m$ -automaton with special output is a set of special  $n$ -predicates (infinite strings of input-states which are ultimately  $\Lambda^n$ ). Simplifying KLEENE's [6] theory of regularity, MYHILL [8], RABIN and SCOTT [13], and COPI, ELGOT and WRIGHT [4] work instead with the set  $\beta_{sg}$  consisting of all finite strings of input states which turn on the output. In this approach the restriction to special outputs is not needed, so that it may seem at first that our concept of behavior is too specialized. However we will see in this section that on the contrary our theory of behavior is more general in that it yields a rather natural extension of the concept of regular set of finite strings to regular relations on finite strings.

We begin by observing that there is a one-to-one mapping  $f$  of all finite strings of elements of  $\Sigma_n$  onto all special  $n$ -predicates  $i^n$  which have the property  $i(l) \equiv \mathbf{V}^n$ , if  $l$  is the length of  $i$ . Namely,

$$i = f(Y_0 \frown \dots \frown Y_h) \equiv (\forall t)_{t < h} [i(t) \equiv Y_t] \wedge [i(h') \equiv \mathbf{V}] \wedge (\forall t)_{t > h'} [i(t) \equiv \Lambda].$$

In particular the empty string corresponds to the predicate  $i(0) \equiv \mathbf{V}$ ,  $i(t') \equiv \Lambda$ . To simplify the presentation we will identify finite strings with the corresponding  $n$ -predicate, i. e.,

(\*\*) A finite  $n$ -string of length  $l$  is a special  $n$ -predicate  $i^n$  of length  $l + 1$  which takes the value  $i(l') \equiv \mathbf{V}^n$ .

The definition of finite string behavior which occurs in the literature can, in our notation, be formulated thus,

**Definition 6.** The *sg-behavior* of a  $n/m$ -automaton  $E \frown H$  with arbitrary output  $U$  is the set  $\beta_{sg}(I, H, U)$  consisting of all  $n$ -strings  $i$  such that  $U[r(l + 1)]$ , if  $l$  is the length of the string  $i$  and  $r = \zeta(I, H, i)$  is the  $m$ -predicate determined by the transition-recursion of  $E \frown H$ .

**Theorem 6.** A set  $\beta$  of  $n$ -strings is the *sg-behavior*  $\beta_{sg}(E, H, U)$  of some  $n/m$ -automaton with arbitrary output if and only if it is the behavior  $\beta(I, G, W)$  of some  $n/q$ -automaton with special output.

Proof. Suppose first that  $\beta = \beta_{\text{sg}}(E, H, U)$ , whereby  $U$  is an arbitrary output of the  $n/m$ -automaton  $E \frown H$ . Let the  $n/m^2$ -automaton  $I \frown G$  and its output  $W$  be defined by,

$$\begin{aligned} I_v : E_v \quad G_v[Y^m \frown Z; X^n] : H_v[Y, X], \quad v = 1, \dots, m \\ I_{m+1} : \Lambda \quad G_{m+1}[Y^m \frown Z; X^n] : [X \equiv \mathbf{V}] U[Y] \vee [X \equiv \mathbf{\Lambda}] \dot{Z}, \\ W[Y^m \frown Z] : Z. \end{aligned}$$

Then clearly  $G_{m+1}[Y \frown Z'; \mathbf{\Lambda}] \equiv Z$ , and therefore  $W[G[Y \frown Z; \mathbf{\Lambda}]] \equiv W[Y \frown Z]$ , which by definition 2 means that  $W$  is special output of  $I \frown G$ . Furthermore note that the transition-recursions of the two automata are,

$$(1) \quad \begin{cases} r(0) \equiv E, \\ r(t') \equiv H[r(t), i(t)], \end{cases} \quad (2) \quad \begin{cases} r(0) \equiv E, \quad s(0) \equiv \Lambda, \\ r(t') \equiv H[r(t), i(t)], \\ r(t') \equiv [i(t) \equiv \mathbf{V}] U[r(t)] \vee [i(t) \equiv \mathbf{\Lambda}] s(t) \end{cases}$$

Now let  $i$  be any special predicate of length  $l$ . If  $i \in \beta(I, G, W)$  then by (\*) of Section 3 and (2),  $W[r(l+1) \frown s(l+1)]$ , i. e.,  $s(l+1)$ . Therefore by (2),  $[i(l) \equiv \mathbf{V}] U[r(l)] \vee [i(l) \equiv \mathbf{\Lambda}] s(l)$ . Because  $l$  is the length of  $i$ ,  $i(l) \equiv \mathbf{\Lambda}$  is not the case, so that  $[i(l) \equiv \mathbf{V}] U[r(l)]$ . Consequently by definition 6,  $i \in \beta_{\text{sg}}(E, H, U)$ . These steps are reversible, so that also  $i \in \beta_{\text{sg}}(E, H, U)$  implies  $i \in \beta(I, G, W)$ . Thus we have constructed an automaton  $I \frown G$  with special output  $W$  such that also  $\beta(I, G, W) = \beta$ . This proves Theorem 6 in one direction.

Suppose next that  $\beta = \beta(I, H, W)$ , whereby  $I \frown H$  is an  $n/m$ -automaton with special output. Define  $U[Y^m] : W[H[Y^m, \mathbf{V}^n]]$ . Noting that  $\beta$  consists of finite  $n$ -strings only, and using Definition 6 and (\*) of Section 3, one easily shows that  $\beta_{\text{sg}}(I, H, U) = \beta(I, H, W) = \beta$ . This establishes Theorem 6 in the other direction.

Theorem 6 shows that the restriction to special outputs in our analysis and synthesis Theorems 1 and 2 (Section 4) is not a limitation in generality, but rather a natural feature which appears when one works with special predicates instead of finite strings. In fact it now is easy to obtain synthesis and analysis theorems in terms of sg-behavior and without restriction on the outputs. For this purpose we remark that the set of all finite  $n$ -strings can be defined by a formula  $Sg_n(i^n)$  of W. 2. A.,

$$Sg_n(i^n) : (\exists j) \cdot j(0) \wedge (\forall t) [j(t') \supset j(t)] [j(t) \dot{j}(t') \supset i(t) \equiv \mathbf{V}] [\dot{j}(t) \supset i(t) \equiv \mathbf{\Lambda}]$$

and we define a formula  $\mathfrak{G}(i^n)$  to be an sg-formula in case  $(\forall i) \mathfrak{G}(i) \supset Sg_n(i)$  holds in W. 2. A. (by Corollary 1 one can effectively decide whether a formula of W. 2. A. is a sg-formula).

#### Theorem 7.

*Analysis. To every  $n/m$ -automaton  $E \frown H$  with arbitrary output  $U$  one can construct a sg-formula  $\mathfrak{G}(i^n)$  of W. 2. A. such that  $\mathfrak{G}(i)$  defines the sg-behavior  $\beta_{\text{sg}}(E \frown H, U)$ .*

*Synthesis. To every sg-formula  $\mathfrak{G}(i^n)$  of W. 2. A. one can construct a  $n/m$ -automaton  $E \frown H$  with output  $U$  such that  $\beta_{\text{sg}}(E, H, U) = \dot{i} \mathfrak{G}(i)$ .*

Proof. By Theorems 1, 2, and 6. Note that we actually established an effective version of Theorem 6.

In short form Theorem 7 states that the sg-behaviors of  $n/m$ -automata with arbitrary (binary) output are exactly the sets of finite  $n$ -strings definable (by sg-formulas) in W. 2. A. It has been shown that the sg-behaviors are the regular sets of finite strings (references at the beginning of this section). Thus,

Theorem 8. *For any set  $\beta$  of finite  $n$ -strings the following are equivalent conditions,*

- (1)  $\beta$  is the sg-behavior of an  $n/m$ -automaton with arbitrary output.
- (2)  $\beta$  is the behavior of an  $n/m$ -automaton with special output.
- (3)  $\beta$  is definable in W. 2. A. (by a sg-formula).
- (4)  $\beta$  is regular.

In the literature sg-behavior has been studied only for automata with one input. Consequently the concept of regularity has been limited to sets. On the other hand we have seen in Section 6 that our theory of behavior can easily be extended to automata with many inputs. On the basis of Theorems 1', 2' (Section 6) and Theorem 8 the following definition suggests itself as a natural extension of the concept of regularity.

A formula  $\mathfrak{G}(i_1^{n_1}; \dots; i_k^{n_k})$  of (extended) W. 2. A. is called an sg-formula if  $(\forall i_1 \dots i_k) \cdot \mathfrak{G}(i_1; \dots; i_k) \supset Sg_{n_1}(i_1) \dots Sg_{n_k}(i_k)$  holds in W. 2. A.

Definition 7. A relation  $R(i_1^{n_1}; \dots; i_k^{n_k})$  on special  $n_r$ -predicates (finite  $n_r$ -strings) is *regular* if it is definable by a formula (sg-formula)  $\mathfrak{G}(i_1; \dots; i_k)$  of extended W. 2. A.

Theorems 1' and 2' can now be restated in the following short form,

Theorem 8'. *A relation  $R(i_1^{n_1}; \dots; i_k^{n_k})$  on special  $n_r$ -predicates (finite  $n_r$ -strings) is the behavior  $\beta(E, H, U)$  of a  $n_1; \dots; n_k/m$ -automaton with special output, if and only if  $R$  is regular.*

In case one prefers to work with finite strings one could extend Definition 6 to a definition of sg-behavior of many-input-automata with arbitrary output, so as to obtain a corresponding extension of Theorem 6, and Theorem 7.

A binary relation on strings which naturally arises in connection with a  $n/m$ -automaton  $E^m \cap H^m$  is its *equi-response relation*  $i^n \sim j^n \pmod{E \cap H}$ . Two finite strings  $i, j$  of input-states are in this relation if they produce the same transit-state, i. e., if  $r(l') \equiv s(h')$  in case  $r = \zeta(E, H, i)$ ,  $s = \zeta(E, H, j)$ ,  $l = \text{length of string } i$ , and  $h = \text{length of string } j$ . It is easy to construct a  $n; n/q$ -automaton with special output whose behavior is the relation  $i^n \sim j^n \pmod{E \cap H}$ . Therefore, by Theorem 8', the equi-response relation of any  $n/m$ -automaton is a regular relation on finite  $n$ -strings. In  $k$ -ary number theory (i. e., the theory of finite strings on the alphabet  $1_1, \dots, 1_k$ ) the equi-response relations module  $k/q$ -automata take the place of conventional congruence relations in ordinary (1-ary) number theory. Correspond-

dingly the concept of regularity is the natural analogue in  $k$ -ary number theory to ultimate periodicity in ordinary number theory.

Another example of a regular relation on special predicates is the adjoined  $S(i, j, s)$  under binary expansion (see Section 5) to the relation  $x + y = z$ . Moreover Theorem 4 may be restated as follows. A relation  $R(x_1, \dots, x_k)$  is definable in the first-order theory  $[=, +, Pw_2]$  if and only if it is the adjoined under binary expansion to a regular relation  $S(i_1, \dots, i_k)$  on special predicates.

In case one prefers to deal with finite strings the following modified concept of binary expansion is appropriate. If  $a_0, \dots, a_h$  are either 1 or 2 then  $a_0 a_1 \dots a_h$  represents  $a_0 2^0 + a_1 2^1 + \dots + a_h 2^h$ . In particular the empty string represents the number 0. It is easy to see that this yields a one-to-one mapping  $x = g(x)$  from all finite 2-strings to all natural numbers. Furthermore one can modify the proof of Theorem 4 to obtain; a relation  $R(x_1, \dots, x_n)$  is definable in  $[=, +, Pw_2]$  if and only if its adjoined  $S(x_1, \dots, x_n)$  under modified binary expansion is a regular relation on finite 2-strings.

Consequently the following two conditions on a relation  $R(x_1, \dots, x_n)$  on natural numbers are equivalent.

(1) *The adjoined  $S(i_1, \dots, i_k)$  of  $R(x_1, \dots, x_k)$  under binary expansion is a regular relation on special predicates.*

(2) *The adjoined  $S_{sg}(x_1, \dots, x_k)$  of  $R(x_1, \dots, x_k)$  under modified binary expansion is a regular relation on finite 2-strings.*

Let us call such a relation  $R$  a *2-regular relation on natural numbers* so that,

**Theorem 9.** *The relations definable in the first-order theory  $[=, +, Pw_2]$  are exactly the 2-regular relations on natural numbers.*

For appropriate definition of  $k$ -regular relation (via adjoined under (modified)  $k$ -ary expansion) Theorem 9 holds if "2-regular" and " $Pw_2$ " are replaced by " $k$ -regular" and " $Pw_k$ " ( $Pw_k(x)$  means  $x$  is a power of  $k$ ). It thus appears that the first-order theory  $[=, +, Pw_k]$  is suitable for a study of regularity on the alphabet  $1_1, \dots, 1_k$  consisting of  $k$  letters. In this connection the following remarks are of interest.

1. Let  $N_k$  denote the set of all finite strings on the alphabet  $1_1, \dots, 1_k$ , and let the function  $g_k: N_k \rightarrow N_1$  ( $N_1$  = set of natural numbers) be defined by

$$g_k(0) = 0, \\ g_k(1_{\nu_0} 1_{\nu_1} \dots 1_{\nu_h}) = \nu_0 k^0 + \nu_1 k^1 + \dots + \nu_h k^h.$$

I. e.,  $g_k$  is that one-to-one enumeration of  $N_k$  in which every string of length  $h$  precedes every string of length  $l > h$ , while strings of equal length are enumerated in lexicographic order reading from right to left. The inverse  $g_k^{-1}$  yields the modified  $k$ -ary expansion of natural numbers. Let  $+^k$  denote the adjoined to  $+$ , i. e.,

$$x +^k y = g_k^{-1}(g_k(x) + g_k(y)), \quad \text{for } x, y \in N_k,$$

and let  $Pw^k$  denote the adjoined to  $Pw_k$ , i. e.,

$$Pw^k(x) \equiv Pw_k(g_k(x)), \quad \text{for } x \in N_k.$$

We remark that it is easy to set up an algorithm for calculating  $\bar{x} +^k \bar{y}$  (similar to the conventional algorithm for addition in ordinary expansion), and that  $Pw^k(\bar{x})$  simply means  $\bar{x}$  is  $1_1$  or  $r$  is  $1_k$  followed by zero or more occurrences of  $1_{k-1}$ . Clearly  $\langle N_1, +, Pw_k \rangle$  is isomorphic to  $\langle N_k, +^k, Pw^k \rangle$ , so that  $[=, +, Pw_k]$  may be interpreted as first-order theory of either of these systems. Theorem 9 as well as the following remarks ought to be interpreted accordingly, and we will often speak of a relation when we actually mean its adjoined under  $g$ .

2. Let  $x <_k y$  and  $x >_k y$  denote respectively the initial segment relation and terminal segment relation on  $N_k$  (or their adjoined). It is easy to construct a  $k; k/m$ -automaton with special output whose behavior is the relation  $x <_k y$ , so that by Theorem 9 (2 replaced by  $k$ ), this relation is definable in  $[=, +, Pw_k]$ . Thus, by Theorem 4,

*The first-order theory of  $\langle N_k, <_k \rangle$  is decidable.*

In contrast we will show elsewhere that the first-order theory of  $\langle N_k, <_k, >_k \rangle$  is not decidable. Therefore,  $x >_k y$  is not definable in  $[=, +, Pw_k]$ .

3. Let  $x \frown^k y$  denote the operation of concatenation on  $N_k$  (or its adjoined on  $N$ ). It is easy to see that

$$x \frown^k y = x + k^{lg_k(x)} \cdot y$$

whereby  $lg_k(x)$  is the length of the string  $x$ , i. e.,

$$lg_k(x) = a \equiv k^a \leq x + 1 < k^{a+1}.$$

However, there is no definition of  $x \frown^k y$  in  $[=, +, Pw_k]$ , because this theory is decidable while it was shown by QUINE [12] that  $[=, \frown^k]$  is undecidable. It can be shown that  $[=, \frown^k]$  and  $[=, +, \cdot]$  are equivalent in the sense that the primitives of one of these theories are definable in the other.

4. We have shown (see also RABIN and SCOTT [13]) that the relations  $x \sim y$  which are equi-response relations modulo automata with one  $k$ -ary input are exactly those right-congruence relations of the operation  $x \frown^k y$  which divide  $N_k$  into finitely many classes. This emphasizes the analogy (remark following Theorem 8') of these relations to ordinary congruence relations of  $x + y = y \frown^1 y$ . We remark that while  $x \frown^k y$  is not definable in  $[=, +, Pw_k]$ , the equi-response relations are definable.

5. The study of congruences is fundamental in elementary number theory. It would be of great importance for a better understanding of regularity (behavior of finite automata) to develop analogously a theory of those (right, left) congruence relations of  $x \frown^k y$  whose partition is finite.

6. Every ultimately periodic set of natural numbers is definable in  $[=, +]$ . By Theorem 9 (extension to  $k$ ) it follows,

*Every ultimately periodic (i. e., 1-regular) set of natural numbers is  $k$ -regular, for every  $k$ .*

7. Let  $A_{n,k}(x)$  stand for  $lg_k(x) \equiv 0 \pmod{n}$ . It then is easy to see that  $A_{n,k}(x)$  is  $k$ -regular, and therefore definable in  $[=, +, Pw_k]$ . Because

$$Pw_{k^n}(x) \equiv [Pw_k(x) \wedge A_{n,k}(x)] \vee [x = 1]$$

it follows that  $Pw_{k^n}$  is definable in  $[=, +, Pw_k]$ . Conversely, for  $n \geq 2$ ,  $Pw_k$  is definable in  $[=, +, Pw_{k^n}]$ , because

$$Pw_k(x) \equiv Pw_{k^n}(x) \vee Pw_{k^n}(kx) \vee \dots \vee Pw_{k^n}(k^{n-1}x).$$

Thus, for any  $k$  and  $n \neq 0$  the theories  $[=, +, Pw_k]$  and  $[=, +, Pw_{k^n}]$  are equivalent with respect to definability. By theorem 9 (2 replaced by  $k$ ) it follows,

*For any  $k$  and  $n \neq 0$ , the  $k^n$ -regular relations on natural numbers are exactly the  $k$ -regular relations.*

8. Suppose that  $n \geq k$  and  $Pw_n$  is  $k$ -regular. Let  $A(x)$  stand for

$$(\exists t)[k^x \leq n^t + 1 < k^{x+1}].$$

Then  $A$  is the set of lengths of elements of  $N_k$  which belong to the  $k$ -adjoined to  $Pw_n$ . Therefore, because  $Pw_n$  is  $k$ -regular it follows that  $A$  must be ultimately periodic. Let  $h$  = phase,  $p$  = period of  $A$ , and let  $q$  be the number of elements of  $A$  occurring in an interval  $(x, x + p - 1)$ , where  $x > h$ . Take  $a > h$  such that  $A(a)$ . Then also  $A(a + p)$ ,  $A(a + 2p)$ ,  $\dots$ ,  $A(a + \nu p)$ ,  $\dots$ , i. e., there are  $b, b_1, b_2, \dots$  such that

$$(1) \quad k^a \leq n^b + 1 < k^{a+1}, \quad k^{a+\nu p} \leq n^{b_\nu} + 1 < k^{a+\nu p+1}, \quad \nu = 1, 2, 3, \dots$$

Note that there are exactly  $q \leq p$  members  $a, a_2, \dots, a_q$  of  $A$  occurring in the interval  $(a, a + p - 1)$ , and that, because  $n \geq k$ , no  $n^{t_1} \neq n^{t_2}$  can have the same length  $x(n^{t_1} + 1$  and  $n^{t_2} + 1$  cannot belong to the same interval  $(k^x, k^{x+1} - 1)$ . It follows that  $a, a_2, \dots, a_q, a + p$  are in order the lengths of  $n^b, n^{b+1}, \dots, n^{b+q-1}, n^{b+q} = n^{b_1}$ . Therefore  $b_1 = b + q$ , and similarly one shows  $b_\nu = b + \nu q$ ,  $\nu = 1, 2, 3, \dots$ . It follows by (1) that

$$\frac{k^{a+\nu p} - 1}{k^{a+1}} < n^{\nu q} < \frac{k^{a+\nu p+1}}{k^a - 1}, \quad \nu = 1, 2, 3, \dots$$

Therefore,

$$\sqrt[\nu]{k^{\nu p} - k^{-a}} \cdot \sqrt[\nu]{k^{-1}} < n^q < \sqrt[\nu]{\frac{k^{a+1}}{k^a - 1}} k^p, \quad \nu = 1, 2, 3, \dots$$

Letting  $\nu$  approach infinity this clearly yields  $n^q = k^p$ .

Thus we have shown that for  $n \leq k$ , if  $Pw_n$  is  $k$ -regular then  $n^q = k^p$  for some  $p \neq 0$  and  $q \neq 0$ . This assertion extends to all  $n \geq 2$ ; because if  $n \geq 2$  there is an  $a$  such that  $n^a \geq k$ , and by 7 if  $Pw_n$  is  $k$ -regular then also  $Pw_{n^a}$  is  $k$ -regular, so that there are  $p \neq 0$  and  $q \neq 0$  such that  $n^{aq} = k^p$ . Consequently, if one defines

(a)  $x \approx y : (\exists uv)[u \neq 0 \wedge v \neq 0 \wedge x^u = y^v]$  one obtains by 7.

(b) If  $n \geq 2$  then  $Pw_n$  is  $k$ -regular (definable in  $[=, +, Pw_k]$ ) if and only if  $n \approx k$ .

(c) The  $k$ -regular relations are exactly the  $n$ -regular relations if and only if  $n \approx k$ .

(d) If  $n \geq 2$  and  $Pw_n$  is  $k$ -regular (definable in  $[=, +, Pw_k]$ ) then also  $Pw_k$  is  $n$ -regular (definable in  $[=, +, Pw_n]$ ), and  $k$ -regularity and  $n$ -regularity have the same meaning.



These facts can, of course, be stated in terms of sequences. For example, the one-to-one mapping  $g_n^{-1}g_k$  of  $N_k$  takes regular relations on  $N_k$  onto regular relations on  $N_n$  just in case  $n \approx k$ .

To better understand the equivalence relation  $\approx$ , let us say that a number  $x$  is *simple* if  $[x = 0 \vee x = 1 \vee \sim(\exists uv)(v \neq 1 \wedge x = u^v)]$ , and let  $spl(y)$  stand for that simple number  $x$  of which  $y$  is a power. Then by elementary number theoretic considerations (prime-decomposition) one shows,

$$(e) \quad x \approx u \cdot \equiv \cdot spl(x) = spl(y).$$

Furthermore, the simple numbers constitute a system of representatives for  $\approx$ , and the equivalence class of a simple number  $x$  is  $\{x^n \mid n \neq 0\}$ .

Because of (b), (c), (d) it follows that it is sufficient to deal with  $k$ -regularity for simple  $k$ . The theory  $[=, +, Pw_n]$  is equivalent to  $[=, +, Pw_k]$ , for  $k = spl(n)$ . (Note that a number  $k$  is simple just in case g.c.d.  $(a_1, \dots, a_n) = 1$  if  $k = p_1^{a_1} \cdots p_n^{a_n}$  is the prime-decomposition of  $k$ .)

By (b) it follows that  $Pw_2$  is not 3-regular. This seems to mean that there is no finite automaton with binary input and ternary output which translates binary expansions of numbers into the corresponding ternary expansions.

## 8. Remarks on feedback

The essentially new element added to pure switching circuits to obtain sequential circuits is feedback. The presence of feedback in our finite automata is reflected in the recursiveness of the definition of the input to transit operation  $r = \psi(i)$  of a finite automaton. In contrast, for a pure switching circuit this operation is introduced by explicit definition. The relationship of feedback to recursions was first clearly realized by BURKS and WRIGHT [2] and KLEENE [6], and is emphasized by CHURCH's contribution to automata theory [3].

A formalism, sufficiently rich to provide notations for all behaviors of finite automata, must contain an operator corresponding to feedback. In KLEENE's [6] formalism of regular expressions, this operator is  $*$ . COPI, ELGOT, and WRIGHT [4] and MYHILL [8], in their modified version of KLEENE's theory use a related monadic operator  $*$ . In W. 2. A.  $= [0, ', \exists_0 i]$  it is the *finite-set-quantifier*  $\exists_0 i$  which corresponds to feedback. This is best shown by indicating how the primitive operators  $\cup, \cdot$ , and  $*$  on regular sets can be obtained in W. 2. A.

For this purpose to every  $Sg_n$ -formula  $\mathfrak{U}(i)$  of W. 2. A. we define the formulas  $\mathfrak{U}_x^y(i)$  and  $\mathfrak{U}_x^\omega(i)$  as follows. First obtain a normal form of  $\mathfrak{U}(i)$ , say

$$\mathfrak{U}(i): (\exists j) \cdot K[j(0)] \wedge (\forall t) B[i(t), j(t), j(t')].$$

Then define

$$\mathfrak{U}_x^y(i): (\exists j) \cdot K[j(x)] \wedge (\forall t)_x^y B(t) \wedge B[\mathbf{V}, j(y), j(y')] \wedge (\forall t)_y^\omega B[\mathbf{A}, j(t), j(t')],$$

$$\mathfrak{U}_x^\omega(i): (\exists j) \cdot K[j(x)] \wedge (\forall t)_x^\omega B[i(t), j(t), j(t')].$$

Next for any  $Sg_n$ -formulas  $\mathfrak{C}(i)$ ,  $\mathfrak{D}(i)$  of W. 2. A. we define  $\mathfrak{C} \cup \mathfrak{D}$ ,  $\mathfrak{C} \cdot \mathfrak{D}$ ,  $\mathfrak{C}^*$  by

$$[\mathfrak{C} \cup \mathfrak{D}](i) : \mathfrak{C}(i) \vee \mathfrak{D}(i),$$

$$[\mathfrak{C} \cdot \mathfrak{D}](i) : (\exists y) \cdot \mathfrak{C}_0^y(i) \wedge \mathfrak{D}_y^w(i),$$

$$[\mathfrak{C}^*](i) : (\exists s) \cdot (\forall xy)[s(x)s(y)(\forall t)_x^y \tilde{s}(t) \supset \mathfrak{C}_x^y(i)] \wedge (\forall x)[s(x)(\forall t)_x^w \tilde{s}(t) \supset \mathfrak{C}_x^w(i)].$$

It is easy to see that these operators on  $Sg_n$ -formulas of W. 2. A. define the operations  $\cup$ ,  $\cdot$ ,  $*$  on sets of finite  $n$ -strings. Note that  $\cdot$  corresponds to individual quantification, while the feedback-operation  $*$  is reflected as a finite-set quantification.

### 9. An unsolved problem

In essence our main result says that in  $[0, ', \exists_0 i]$  one can define only very simple recursive operators  $u = \psi(i)$ , namely, input to output operators of finite automata. In contrast  $[0, ', 2x, \exists_0 i]$  contains definitions for all recursive operators  $u = \psi(i)$  (see ROBINSON [14]).

**Problem.** *Is the weak second-order theory  $[0, 2x + 1, 2x + 2, \exists_0 i]$  decidable? Which are the recursive operators definable in this theory?*

These questions are of interest for automata theory, because via the enumeration of finite strings on two letters  $1_1, 1_2$  (see § 7, 1). The theory  $[0, 2x + 1, 2x + 2, \exists_0 i]$  is equal to the theory  $[0, 1_1 \wedge x, 1_2 \wedge x, \exists_0 i]$  on the set  $N_2$  of finite strings. Let  $\exists_{00} i$  denote quantification over all finite subsets of  $N_k$  which in addition are chains with respect to the terminal segment relation on  $N_k$ . Then by a straightforward extension of the methods used in this paper one obtains:

**Theorem 10.** *The "very-weak" second-order theory  $[0, 1_1 \wedge x, \dots, 1_k \wedge x, \exists_{00} i]$  is decidable. All regular sets, and only regular sets, on the alphabet  $1_1, \dots, 1_k$  can be defined by formulas  $\mathfrak{C}(x)$  of this theory.*

### Bibliography

- [1] BEHMANN, HEINRICH, Beiträge zur Algebra der Logik, insbesondere zum Entscheidungsproblem. Math. Ann. **86**, 163—229 (1922).
- [2] BURKS, A. W., and WRIGHT, J. B., Theory of Logical Nets. Proc. IRE, **41**, 1357—1365 (1953).
- [3] CHURCH, ALONZO, Application of Recursive Arithmetic to the Problem of Circuit Synthesis, Proceedings of the Cornell Logic Conference, Cornell University, 1957. Also see „Application of Recursive Arithmetic in the Theory of Computing and Automata”, Notes for a summer course, The University of Michigan, 1959.
- [4] COPI, I. M., ELGOT, C. C., and WRIGHT, J. B., Realization of Events by Logical Nets. Journal Ass. Comp. Mach. **5**, pp. 181—196 (1958).
- [5] GÖDEL, KURT, On undecidable propositions of formal mathematical systems. Notes by S. C. KLEENE and BARKLEY ROSSER on lectures at the Institute for Advanced Study, 1934. Mimeographed, Princeton, N. J., 30 pp.
- [6] KLEENE, S. C., Representation of Events in Nerve Nets and Finite Automata. Automata Studies, Princeton University Press, 1956, 3—41.

- [7] MEDVEDEV, I. T., On a Class of Events Representable in a Finite Automaton. MIT Lincoln Laboratory Group Report, 34—73, translated from the Russian by J. SCHORR-KON, June 30, 1958.
- [8] MYHILL, JOHN, Finite Automata and Representation of Events. WADC Report TR 57—624, Fundamental Concepts in the Theory of Systems, October, 1957, 112—137.
- [9] MOORE, E. F., Gedanken-Experiments on Sequential Machines. Automata Studies, Princeton, 1956, 129—153.
- [10] PRESBURGER, M., Über die Vollständigkeit eines gewissen Systems der Arithmetik ganzer Zahlen, in welchem die Addition als einzige Operation hervortritt. Comptes-rendus du I Congres des Mathematiciens des Pays Slavs, Warsaw, 1930, 92—101, 395.
- [11] PUTNAM, H., Decidability and Essential Undecidability. Journ. Symb. Log. **22**, 39—54. (1957)
- [12] QUINE, W. V., Mathematical Logic. Harvard Univ. Press, Cambridge, 1947.
- [13] RABIN, M., and SCOTT, D., Finite Automata and Their Decision Problems. IBM Journal, April, 1959, 114—125.
- [14] ROBINSON, R. M., Restricted Set-Theoretical Definitions in Arithmetic. Proc. Am. Math. Soc. **9**, 238—242 (1958).
- [15] SKOLEM, THORALF, Untersuchungen über die Axiome des Klassenkalküls und über Produktions- und Summationsprobleme, welche gewisse Klassen von Aussagen betreffen. Skrifter utgit av Videnskapsselskapet i Kristiania, I. Mat.-nat. kl. 1919, No. 3, 37 pp.

(Eingegangen am 14. September 1959)