UNIVERSITAS COMENIANA
ACTA MATHEMATICA UNIVERSITATIS COMENIANAE
XLVI—XLVII—1985

GENERALIZED PASCAL TRIANGLES. DECIDABILITY RESULTS

IVAN KOREC, Bratislava

Abstract. Generalized Pascal triangles (GPT) are created analogously as the usual Pascal triangle is. However, instead of the addition on the set N of nonnegative integers an arbitrary binary operation on a set A is used. Moreover, the left and the right margins need not be constant but they are formed by using two unary operations on A, and the top is replaced by a finite sequence of elements of A. Even if only finite sets A are considered some questions about GPT are algorithmically unsolvable. For example, it is undecidable whether an element $x \in A$ occurs in a given GPT, or whether x occurs there infinitely many times. These problems are proved to be solvable in some special cases. It is so e.g. if the set of rows of a GPT is a semilinear language or if the left and the right margins are constant and the binary operation is addition modulo a prime or if A consists of at most two elements.

1. Introduction

The present paper is motivated by [1], [2], [3], where the systolic trellis automata are studied as a model for parallel computations with pipelining which is suitable for VLSI technology.

The systolic trellis automata consist of processors (without memory) which are placed in the lattice points of the first quadrant of the plane. We suppose that positive directions of the coordinate axes will be right-down and left-down so that the origin will be the top point of the first quadrant. Every processor obtains the data from its sons (i.e. down neighbours) and sends the output signal (with unit delay) to two its fathers (if the processor is inside) or to its father (if the processor is on the margin). If a word w of the length n ought to be processed, then it is given to the inputs of the horizontal row which contain exatly n processors. The acceptance of w depends on the output of the top processors after n steps. Notice that the infiniteness of a systolic trellis automaton is an abstraction similar to that of a tape of Turing machines.

It is obviously substantial for systolic trellis automata how regularly their processors are placed. An important class of systolic trellis automata is formed by so called regular systolic automata [1]. They will consist of labelled processors (of finitely many kinds). A labelled processor consists of a label and a (proper) processor (which is a pair consisting of an input function and a transition function); the label of a processor in a regular trellis automata is uniquely determined by the labels of its fathers (left and right ones, or only one of them if the other does not exist). Hence a regular trellis automaton can be easily constructed from the top downwards. It suffices to know the label of the top processor and the rules how to determine the labels of other processors from the labels of its fathers. Besides questions about the computational power of regular systolic automata (and just by study of these questions) there also arise some questions about their global and local structures. For example: which (labels of) processors mentioned in the rules do occur in the regular trellis automata and how often they occur; which (labels of) processors occur in their neighbourhood and how they are placed; what is the structure of horizontal rows of a regular trellis automaton. If we study only these structural queastions we may forget the (proper) processors at all, and consider only their labels; this we shall do in all further sections. That is one way how to obtain generalized Pascal triangles. Another way which also motivates the term used is given by their definition.

The above-mentioned questions are algorithmically unsolvable in general. In this paper we shall deal with some special cases when they are solvable. The restrictions will be made either to the rules of forming new labels or to the structure of an initial segment which can be constructed and investigated. Moreover, we shall prove solvability of these problems if only two labels of processors are used. We shall prove this for a more general case when several processors in the top line are allowed.

The two next sections contain terminology and definitions, precise formulation of the algorithmic problems and a proof of their unsolvability. (Abstract symbols are considered instead of labels of processors.) The case considered in the fourth and the fifth sections can be informally called semilinear. The sixth section considers Pascal triangles modulo p; they resemble tree structures of processors (and indeed make it possible to realize some of them). The seventh section applies the obtained results e.g. for the case of two symbols (i.e., that of two named processors). The last section contains some modifications and generalizations; e.g. how to apply the previous results fo groups of symbols (or named processors) which are not contained in one row.

2. Notation and basic definitions

The letters Z, N denote the set of integers and the set of nonnegative integers, respectively. The $\{(x, y) \in Z \times Z;\ x+y \geqq 0\}$ will be denoted by D. Further, let $\mathscr{S}$ be an infinite countable set of symbols which contains a special symbol (a cipher) for every $x \in N$, and some further symbols, e.g. $\$$. These further symbols will be enumerated by (several or all) negative integers, e.g. $\$$ will have the number -1. Finite sequences of elements of $\mathscr{S}$ will be called words, and all usual word and language operations will be used for them and their sets, e.g. concatenation (without any symbol), iteration *, iteration $^+$ without the empty word ε, power as repeated concatenation and boolean set operations union $\cup$, intersection $\cap$ and difference $-$. The set of all subwords of the words of a language L will be denoted by Sub(L). The length of a word w will be denoted by $|w|$, and the cardinality of a set X will be also denoted by $|X|$. All considered alphabets will be (finite nonempty) subsets of $\mathscr{S}$, and base sets of all algebras also will be subsets of $\mathscr{S}$. (An exception will be necessary for uncountable algebras but we shall not need them.) As a rule, the symbol $\$$ will not be used in alphabets and base sets unless it is explicitely mentioned.

Enumeration of elements of $\mathscr{S}$ enables us to speak about recursivity and recursive enumerability not only for languages but e.g. for classes of finite algebras of the same (finite) signature, and for infinite countable algebras. (Codings of such object e.g. into the alphabet $\{0, 1\}$ is possible but it will not be necessary.)

Members of sequences (finite as well as infinite ones) will be usually numbered by all elements of N. Hence a sequence will usually begin with the 0-th member. We shall often need ultimately periodic sequences by the following definition.

2.1. Definition. 1) An infinite sequence $(a_0, a_1, a_2, \ldots)$ will be called ultimately periodic if there are nonnegative integers k, p such that for every integer $x \geqq k$ it holds $a_{x+p} = a_x$.

2) The least integers k, p with the above property will be called pre-period and period of the sequence $(a_0, a_1, a_2, \ldots)$, respectively.

3) The finite sequences

$$(a_0, a_1, a_2, \ldots, a_{k-1}), \qquad (a_k, a_{k+1}, \ldots, a_{k+p-1}),$$

where k, p are as above, will be called the pre-periodical part and the periodical part of $(a_0, a_1, a_2, \ldots)$, respectively.

4) An ultimately periodic sequence will be called periodic if its pre-periodical part is empty (i.e. pre-period is 0).

We shall consider functions defined on some special subsets of the set D. These subsets and some notions concerning them are introduced in the following definition.

2.2. Definition. 1) For every $m, n \in \mathrm{Z}$ we denote

$$\mathrm{D}_{m,n} = \{(x, y) \in \mathrm{D};\ x \geqq m \text{ and } y \geqq -n\}\ . \tag{2.2.1}$$

2) For $\mathrm{M} = \mathrm{D}_{m,n}$ or $\mathrm{M} = \mathrm{D}$ and $k \in \mathrm{Z}$ the sets

$$\{(x, y) \in \mathrm{M};\ x + y = k\}\ , \qquad \{(x, y) \in \mathrm{M};\ x - y = k\}\ ,$$
$$\{(x, y) \in \mathrm{M};\qquad x = k\}\ , \qquad \{(x, y) \in \mathrm{M};\qquad y = k\}$$

will be called the k-th row of M, the k-th column of M, the k-th left diagonal of M and the k-th right diagonal of M, respectively.

3) For $\mathrm{M} = \mathrm{D}_{m,n}$ and $k \in \mathrm{Z}$ the sets

$$\{(x, y) \in \mathrm{M};\ m \leqq x < m + k\}\ , \qquad \{(x, y) \in \mathrm{M};\ -n + k > y \geqq -n\}$$
$$\{(x, y) \in \mathrm{M};\ x + y < k\}$$

will be called the left, the right, and the upper margins of width k of the set M, respectively. Their union will be called the margin of width k of M.

4) The margin of width 1 will be simply called the margin, and analogously for the left, the right, and upper margins. The complement of the margin of M (taken in M, of course) will be called the interior of M.

5) If F is a function with the domain $\mathrm{M} = \mathrm{D}$ or $\mathrm{M} = \mathrm{D}_{m,n}$ then the restriction of F to the k-th row (k-th column etc.) of M will be called the k-th row (k-th column etc.) of F, respectively. The same names will be used for (finite or infinite) sequences created from the appropriate values of F.

In the upper part of *Figure 2.1* there is a function F with the domain $\mathrm{D}_{0,0} = \mathrm{N} \times \mathrm{N}$. The lower part illustrates a function G with the domain $\mathrm{D}_{0,2}$. If we write the function values in this manner (which is usual for the Pascal triangle), then the notions of Definition 2.1 are quite natural. We shall informally use also further geometrical notion, e.g. the $(m - n)$th column of $\mathrm{D}_{m,n}$ will be called the axis of $\mathrm{D}_{m,n}$.

$$
\begin{array}{ccccccccc}
 & & & & F(0,0) & & & & \\
 & & & F(0,1) & & F(1,0) & & & \\
 & & F(0,2) & & F(1,1) & & F(2,0) & & \\
 & F(0,3) & & F(1,2) & & F(2,1) & & F(3,0) & \\
F(0,4) & & F(1,3) & & F(2,2) & & F(3,1) & & F(4,0) \\
\ldots & & \ldots & & \ldots & & \ldots & & \ldots
\end{array}
$$

$$
\begin{array}{ccccccccccc}
 & & & G(0,0) & & G\,1,-1) & & G(2,-2) & & & \\
 & & G(0,1) & & G(1,0) & & G(2,-1) & & G(3,-2) & & \\
 & G(0,2) & & G(1,1) & & G(2,0) & & G(3,-1) & & G(4,-2) & \\
G(0,3) & & G(1,2) & & G(2,1) & & G(3,0) & & G(4,-1) & & G(5,-2) \\
\ldots & & \ldots & & \ldots & & \ldots & & \ldots & & \ldots
\end{array}
$$

Figure 2.1

Algebras will be denoted by letters $\mathscr{A}$, $\mathscr{B}$, ..., and their base sets by the appropriate letters A, B, ... The algebras with the signature (0, 1, 1, 2) will be frequent in what follows; if no signature is explicitly mentioned, an algebra with the signature (0, 1, 1, 2) is considered. Later we shall use the nulary operation (sometimes) to construct the top of a generalized Pascal triangle, the unary operation to construct its left and its right margins, and the binary operation to construct its interior. (Hence the usual Pascal triangle can be obtained from the algebra (N; 1, id, id, +), where $\mathrm{id}(x)=x$ for all $x\in \mathrm{N}$ and + denotes the usual addition.)

We shall sometimes extend functions defined on the sets $\mathrm{D}_{m,n}$ to the whole set D by the value $\mathcal{S}$. In connection with such extensions the following definition will be suitable.

2.3. Definition. Let $\mathscr{A}=(\mathrm{A}, \mathrm{K}, \mathrm{l}, \mathrm{r}, .)$ be an algebra of signature (0, 1, 1, 2) and $\mathcal{S}\notin \mathrm{A}$. We shall say that the binary operation $*$ on the set $\mathrm{A}\cup\{\mathcal{S}\}$ is canonically ordered to $\mathscr{A}$ if for all $x, y\in \mathrm{A}\cup\{\mathcal{S}\}$ it holds:

$$x*y=\begin{array}{ll} \mathcal{S} & \text{if } x=\mathcal{S},\ y=\mathcal{S} \\ \mathrm{l}(y) & \text{if } x=\mathcal{S},\ y\in \mathrm{A} \\ \mathrm{r}(x) & \text{if } x\in \mathrm{A},\ y=\mathcal{S} \\ x\cdot y & \text{if } x\in \mathrm{A},\ y\in \mathrm{A} \end{array} \tag{2.3.1}$$

A binary operation on a finite set A is usually given by its Cayley table. All four operations of an algebra $\mathscr{A}$ will be also given by a table. This table will be divided into the head row, a row for the first unary operation l and $|\mathrm{A}|$ further rows for the binary operation. Analogously it will be divided into $|\mathrm{A}|+2$ columns (only the second unary operation r is used instead of l). The head column will contain ., l and the elements of A; the head row will contain ., r and the elements of A in the same order. Now it is clear how to write the values of ., l, r into the table. The value K will be written into the intersection of the row headed by l with the column headed by r. Notice that if we replace l, r and K in the table by the symbol $\mathcal{S}$ (and the dot . by a star $*$) we obtain the Cayley table of the canonically ordered operation.

It may happen sometimes that some values of the operations of $\mathscr{A}$ will be unsubstantial. In this case the dot will be written at the appropriate place of the Cayley table or another expression for $\mathscr{A}$. It will not mean that we want to consider a partial algebra. We will consider any total algebra which can arise if we arbitrarily replace dots by the elements of A.

2.4. Definition. 1) If $\mathscr{A}=(\mathrm{A}, \mathrm{K}, \mathrm{l}, \mathrm{r}, .)$ is an algebra of signature (0, 1, 1, 2), then the generalized pascal triangle of algebra $\mathscr{A}$ (notation: $\mathrm{GPT}(\mathscr{A})$) is the mapping F of $\mathrm{N}\times\mathrm{N}$ into A defined for all $x, y\in\mathrm{N}$ by the formulae

$F(0, 0)=\mathrm{K}$

$F(0, y+1) = l(F(0, y))$
$F(x+1, 0) = r(F(x, 0))$
$F(x+1, y+1) = F(x, y+1) \cdot F(x+1, y)$

2) Let $\mathscr{A} = (\mathrm{A};\ \mathrm{K},\ \mathrm{l},\ \mathrm{r},\ .)$ be an algebra of signature (0, 1, 1, 2), let $w = a_0 a_1 \ldots a_n \in \mathrm{A}^+$ and let i be an integer. The generalized Pascal triangle of $\mathscr{A}$ with the top w and with the left margin at the i-th left diagonal (notation: GPT($\mathscr{A}$, w, i)) is the mapping G with the domain

$$D_{i,\,n+i} = \{(x, y) \in \mathrm{D};\ x \geqq i \text{ and } y \geqq -(n+i)\}$$

which is defined by the formulae

$G(x, -x) = a_{x-i}$ for $i \leqq x \leqq n+i$
$G(i, y+1) = \mathrm{l}(G(i, y))$ for $y \geqq -i$
$G(x+1,\ -(n+i)) = \mathrm{r}(G(x,\ -(n+i))$ for $x \geqq n+i$
$G(x+1, y+1) = G(x, y+1) \cdot G(x+1, y)$ for $x \geqq i,\ y \geqq -(n+i),$
$x + y + 2 > 0.$

3) For $i = 0$ we shall write GPT($\mathscr{A}$ w) instead of GPT($\mathscr{A}$, w, i); we shall also omit the words "with the left margin on the i-th left diagonal".

4) A function F will be called a generalized Pascal triangle (abbreviation: GPT), if there is an algebra $\mathscr{A}$, a word $w \in \mathrm{A}^+$ and an integer i such that $F = \mathrm{GPT}(\mathscr{A}, w, i)$.

2.5. Remarks. 1) The first point of the above definition can be considered as a special case of its second point because $\mathrm{GPT}(\mathscr{A}) = \mathrm{GPT}(\mathscr{A}, \mathrm{K})$.

2) For every mapping F of D into $\mathscr{S}$ we denote by $F_{\mathcal{S}}$ its completion to the domain D by the value $\mathcal{S}$. Then for every algebra $\mathscr{A}$ and $w \in \mathrm{A}^+$, $i \in \mathrm{N}$ the function $G_{\mathcal{S}} = \mathrm{GPT}_{\mathcal{S}}(\mathscr{A}, w, i)$ satisfies

$$G_{\mathcal{S}}(x+1, y+1) = G_{\mathcal{S}}(x, y+1) * G_{\mathcal{S}}(x+1, y)\ , \tag{2.5.1}$$

where $*$ denotes the operation canonically ordered to $\mathscr{A}$. The relation between $G_{\mathcal{S}}$ and G is very simple (remember that $\mathcal{S}$ is not a value of G), and therefore we shall sometimes use $G_{\mathcal{S}}$ instead of G (also without explicitly mentioning it). For example, if an initial subword of length m of a row of G which is shorter than m is considered, we add the necessary number of $\mathcal{S}$ at its end. Analogously we add $\mathcal{S}$ at the beginning of final subwords. (In the 6-th section we shall use 0 instead of $\mathcal{S}$.)

4) The change of i in the expression GPT($\mathscr{A}$, w, i) corresponds to the horizontal translation of this GPT, which is usually unsubstantial. (An exception is when the mutual position of several GPT is considered.) Therefore we shall restrict ourselves to the case $i = 0$ if possible.

Some notions from Definition 2.4 and Remark 2.5 are explained by examples in *Figure 2.2c, d, e, f.*

.	r	0	1
1	1	1	1
0	1	0	1
1	0	1	0

a) Algebra $\mathcal{A}$

*	$\mathcal{S}$	0	1
$\mathcal{S}$	$\mathcal{S}$	1	1
0	1	0	1
1	0	1	0

b) The operation * (2. 3. 1)

$$\begin{array}{ccccccccccccc}
 & & & & & & 1 & & & & & & \\
 & & & & & 1 & & 0 & & & & & \\
 & & & & 1 & & 1 & & 1 & & & & \\
 & & & 1 & & 0 & & 0 & & 0 & & & \\
 & & 1 & & 1 & & 0 & & 0 & & 1 & & \\
 & 1 & & 0 & & 1 & & 0 & & 1 & & 0 & \\
1 & & 1 & & 1 & & 1 & & 1 & & 1 & & 1
\end{array}$$

c) GPT($\mathcal{A}$)

$$\begin{array}{ccccccccccccccc}
 & \mathcal{S} & & \mathcal{S} & & \mathcal{S} & & 1 & & \mathcal{S} & & \mathcal{S} & & \mathcal{S} & \\
\mathcal{S} & & \mathcal{S} & & \mathcal{S} & & 1 & & 0 & & \mathcal{S} & & \mathcal{S} & & \mathcal{S} \\
 & \mathcal{S} & & \mathcal{S} & & 1 & & 1 & & 1 & & \mathcal{S} & & \mathcal{S} & \\
\mathcal{S} & & \mathcal{S} & & 1 & & 0 & & 0 & & 0 & & \mathcal{S} & & \mathcal{S} \\
 & \mathcal{S} & & 1 & & 1 & & 0 & & 0 & & 1 & & \mathcal{S} & \\
\mathcal{S} & & 1 & & 0 & & 1 & & 0 & & 1 & & 0 & & \mathcal{S} \\
 & 1 & & 1 & & 1 & & 1 & & 1 & & 1 & & 1 &
\end{array}$$

d) $\mathrm{GPT}_{\mathcal{S}}$ ($\mathcal{A}$)

$$\begin{array}{ccccccccccccc}
 & & & & 0 & & 1 & & 1 & & & & \\
 & & & 1 & & 1 & & 0 & & 0 & & & \\
 & & 1 & & 0 & & 1 & & 0 & & 1 & & \\
 & 1 & & 1 & & 1 & & 1 & & 1 & & 0 & \\
1 & & 0 & & 0 & & 0 & & 0 & & 1 & & 1
\end{array}$$

e) GPT($\mathcal{A}$, 011)

$$\begin{array}{ccccccccccccccccc}
\mathcal{S} & & \mathcal{S} & & \mathcal{S} & & 0 & & 1 & & 1 & & \mathcal{S} & & \mathcal{S} & & \mathcal{S} \\
 & \mathcal{S} & & \mathcal{S} & & 1 & & 1 & & 0 & & 0 & & \mathcal{S} & & \mathcal{S} & \\
\mathcal{S} & & \mathcal{S} & & 1 & & 0 & & 1 & & 0 & & 1 & & \mathcal{S} & & \mathcal{S} \\
 & \mathcal{S} & & 1 & & 1 & & 1 & & 1 & & 1 & & 0 & & \mathcal{S} & \\
\mathcal{S} & & 1 & & 0 & & 0 & & 0 & & 0 & & 1 & & 1 & & \mathcal{S}
\end{array}$$

f) $\mathrm{GPT}_{\mathcal{S}}$($\mathcal{A}$, 011)

Figure 2.2

2.6. Definition. 1) The set of rows of GPT($\mathcal{A}$), GPT($\mathcal{A}$, w), GPT($\mathcal{A}$, w, i) will be denoted R($\mathcal{A}$), R($\mathcal{A}$, w), R($\mathcal{A}$, w, i), respectively. (The rows are enumerated according to Definition 2.2.)

2) Let $\mathcal{A}=(A; K, l, r, .)$ be an algebra and $*$ be the binary operation (2.3.1) canonically ordered to $\mathcal{A}$. Then by next or next$_{\mathcal{A}}$ if $\mathcal{A}$ must be stressed we denote the mapping of $(A\cup\{\mathcal{S}\})^+$ into $(A\cup\{\mathcal{S}\})^*$ defined for all $a_0, a_1, \ldots, a_n \in A\cup\{\mathcal{S}\}$ by the formula

$$\operatorname{next}(a_0a_1 \ldots a_n)=b_0b_1 \ldots b_{n-1}\,, \tag{2.6.1}$$

where

$$b_i=a_i * a_{i+1} \text{ for all } i=0, 1, \ldots, n-1\,.$$

An application of the function next and its powers (as repeated compositions) gives the next lemma and the theorem about ultimate periodicity of margins of GPT for finite algebras.

2.7. Lemma. For every algebra $\mathcal{A}=(A; K, l, r, .)$, $w\in A^+$ and integers $m\geqq 0$, $i\geqq 0$ denote:

r_i the i-th row of GPT($\mathcal{A}$, w),
$x_{i,m}$ the initial subword of length m of r_i (or $r_i\mathcal{S}^m$),
$y_{i,m}$ the final subword of length m of r_i (or $\mathcal{S}^m r_i$).

Then for all i, $k\in N$ it holds

$$r_{i+1} =\operatorname{next}(\mathcal{S}r_i\mathcal{S}), \qquad r_{i+k} =\operatorname{next}^k(\mathcal{S}^k r_i\mathcal{S}^k) \tag{2.7.1}$$

$$x_{i+1,m}=\operatorname{next}(\mathcal{S}x_{i,m}), \qquad x_{i+k,m}=\operatorname{next}^k(\mathcal{S}^k x_{i,m}) \tag{2.7.2}$$

$$y_{i+1,m}=\operatorname{next}(y_{i,m}\mathcal{S}), \qquad y_{i+k,m}=\operatorname{next}^k(y_{i,m}\mathcal{S}^k) \tag{2.7.3}$$

Proof. The definition of next immediately implies the left formulae. The right formulae are for $k=0$ trivial (e.g. $r_i=r_i$) and for positive k they can be proved by induction from the left formulae.

2.8. Theorem. Let $\mathcal{A}=(A; K, l, r, .)$ be a finite algebra, $w\in A^+$, let m be a positive integer and let for every $i\in N$ the words $x_i=x_{i,m}$, $y_i=y_{i,m}$ be defined as in Lemma 2.7. Then the sequences

$$(x_0, x_1, x_2, \ldots), \quad (y_0, y_1, y_2, \ldots)$$

are ultimately periodic. The period of every of them is at most $|A|^m$, and the sum of the period and the pre-period is at most $|A|^m+\max(0, m-|w|)$.

Proof. Consider for example the left sequence. It starts with $\max(0, m-|w|)$

members containing $\$$, and further it contains only the elements of A^n. Hence it contains at most

$$n = |A|^m + \max(0, m - |w|)$$

different members. Therefore for some $q \leqq n$ the member x_q is equal to a member x_s, $s < q$. Let q be the smallest possible and denote $p = q - s$. Then $x_{p+s} = x_s$, and by (2.7.2) $x_{p+k} = x_p$ for all $k \geqq s$. The integers p, s are the period and the pre-period of the sequence $(x_0, x_1, x_2, \ldots)$, and obviously $p + s \leqq n$. It remains to prove $p \leqq |A|^m$. If $p > n$, then x_s contains the symbol $\$$ and then infinitely many members x_i contain $\$$, which is a contradiction.

A similar theorem, with another proof, and with another bounds for the pre-period (initial index) is contained in [7].

3. Formulation of the decision problems

Now we shall formulate a theorem about algorithmic undecidability of some problems concerning generalized Pascal triangles. We shall give a wider list of these problems so that we can use it later. The theorem will deal with the class of all generalized Pascal triangles of finite algebras. If we restrict the class of considered GPT, some of the problems can be algoritmically decidable. For example, we can restrict the cardinalities of algebras, or the length of the tops of GPT. We can also require that the algebras fulfil some identities or other conditions. In further sections we shall consider some of such suitable restrictions.

The following mnemonic is used in the notation of problems: o means "at least one", i means "infinitely many", a means "almost all" (i.e. with finitely many exceptions), e means "every", s means "symbol", and w means "word". We shall say that a word u belongs to GPT($\mathscr{A}$, w) if u is a subword of a row of GPT($\mathscr{A}$, w), i.e. if $u \in \mathrm{Sub}(\mathrm{R}(\mathscr{A}, w))$.

3.1. Theorem. There is no algorithm which for every finite algebra $\mathscr{A} = (A; K; l, r, .)$, every symbol $x \in A$, every $B \subseteq A$ and every $u, w \in A^+$ determines:

(P.1/oss) whether the symbol x occurs in GPT($\mathscr{A}$);
(P.1/ows) whether the word u occurs in GPT($\mathscr{A}$);
(P.1/osw) whether the symbol x occurs in GPT($\mathscr{A}$, w);
(P.1/oww) whether the word u occurs in GPT($\mathscr{A}$, w);
(P.1/iss) whether x occurs in infinitely many rows of GPT($\mathscr{A}$);
(P.1/iws), (P.1/isw), (P.1/iww) analogical to (P.1/iss);
(P.1/ass) whether x occurs in almost all rows of GPT($\mathscr{A}$);
(P.1/aws), (P.1/asw), (P.1/aww) analogical to (P.1/ass);
(P.1/esw) whether x occurs in every row of GPT($\mathscr{A}$, w);

(P.1/eww) whether u occurs in every row of GPT($\mathscr{A}$, w);
(P.2/oss) whether at least one row of GPT($\mathscr{A}$) belongs to x^*;
(P.2/ows) whether at least one row of GPT($\mathscr{A}$) belongs to u^*;
(P.2/osw) whether at least one row of GPT($\mathscr{A}$, w) belongs to x^*;
(P.2/oww) whether at least one row of GPT($\mathscr{A}$, w) belongs to u^*;
(P.2/iss) whether infinitely many rows of GPT($\mathscr{A}$) belong to x^*;
(P.2/iws), (P.2/isw), (P.2/iww) analogical to (P.2/iss);
(P.2/ass) whether almost all rows of GPT($\mathscr{A}$) bolong to x^*;
(P.2/aws), (P.2/asw), (P.2/aww) analogical to (P.2/ass);
(P.3/oss) whether every symbol $y \in \mathrm{B}$ occurs in GPT($\mathscr{A}$);
(P.3/ows) whether every word $v \in \mathrm{B}^+$ occurs in GPT($\mathscr{A}$);
(P.3/osw) whether every symbol $y \in \mathrm{B}$ occurs in GPT($\mathscr{A}$, w);
(P.3/oww) whether every word $v \in \mathrm{B}^+$ occurs in GPT($\mathscr{A}$, w);
(P.3/iss) whether every symbol $y \in \mathrm{B}$ occurs in infinitely many lines of GPT($\mathscr{A}$);
(P.3/iws), (P.3/isw), (P.3/iww) analogical to (P.3/iss);
(P.3/ass) whether every symbol $y \in \mathrm{B}$ occurs in almost all lines of GPT($\mathscr{A}$);
(P.3/aws), (P.3/asw), (P.3/aww) analogical to (P.3/ass);
(P.3/esw) whether every symbol $y \in \mathrm{B}$ occurs in all lines of GPT($\mathscr{A}$, w).

3.2. Remarks. 1) The notation system of Theorem 3.1 allows us to formulate also some further problems, e.g.:

(P.1/ews) whether x occurs in all rows of GPT($\mathscr{A}$);
(P.2/ess) whether all rows of GPT($\mathscr{A}$, w) bolong to x^*;
(P.3/eww) whether every word $v \in \mathrm{B}^+$ occurs in all rows of GPT($\mathscr{A}$, w).

However, these problems are algorithmically solvable, as we shall see later.

In what follows we shall refer to all 48 problems which can be generated by the notation system of Theorem 3.1.

2) The mentioned problems can be considered not only for finite algebras but also for some infinite algebras (e.g. recursive ones). In the present paper we shall do it only if the generalized problems do not become more complicated. E.g., generalized Pascal triangles with infinite many symbols will not be considered at all.

3) Pascal triangles with finitely many symbols are recursive (independently of the properties of the algebras they are created from). Such GPT are determined by their sufficiently wide upper margins (which are finite). However, when an upper margin of a GPT is given it is not clear whether it is sufficiently wide or not. (The set of those upper margins which are not sufficiently wide is recursively enumerable. However, it is not clear whether it is recursive.)

Sketch of proof of Theorem 3.1. The proof is based on a simulation of computation of a Turing machine T by creating a generalized Pascal triangle. One step of the computation will be simulated by creating several (e.g. eight) further rows of the generalized Pascal triangle. Only the middle part of GPT is used for the simulation (it is the part filled in by X in the upper part of *Figure 3.1*). Therefore after halting of the Turing machine T it is possible to remove all traces of simulation and to arrange that all sufficiently futher rows have a requested form (e.g. that they consist only of zeros). The halting of T is signalized by a symbol (from the group) Y in a row of GPT. The creating of further rows is displayed in the middle of *Figure 3.1*.

In the table in the lower part of Figure 3.1 the letters X, Y represent groups of symbols. The symbols of the group X are used at first to create an initial configuration of a (universal) Turing machine T. Then they are used to simulate the appropriate computation of T. The symbols of the group Y are used only when T halts. Then they arrange the form of all sufficiently farther rows of GPT. For example, if we want that all these rows consist of zeros if suffices that the group Y consists of 0 only (and $0 \notin X$). If we need a more complicated form of further rows, the group Y can simulate another suitable Turing machine.

```
                         1
                       2   2
                     3   3   3
                   0   1   1   0
                 0   2   X   2   0
               0   3   X   X   3   0
             0   0   1   X   1   0   0
           0   0   2   X   X   2   0   0
         0   0   3   X   X   X   3   0   0
       0   0   0   1   X   X   1   0   0   0
     0   0   0   2   X   X   X   2   0   0   0
   0   0   0   3   X   X   X   X   3   0   0   0
 0   0   0   0   1   X   X   X   1   0   0   0   0
0   0   0   0   2   X   X   X   X   2   0   0   0   0
```

```
          0   0   0   2   X   X   X   2   0   0   0
        0   0   0   3   X   Y   X   X   3   0   0   0
      0   0   0   0   1   Y   Y   X   1   0   0   0   0
    0   0   0   0   2   Y   Y   Y   X   2   0   0   0   0
  0   0   0   0   3   Y   Y   Y   Y   X   3   0   0   0   0
0   0   0   0   0   Y   Y   Y   Y   Y   1   0   0   0   0   0
```

.	r	0	1	2	3	X	Y
1	1	0	2	3	0	.	.
0	0	0	2	3	0	.	0, Y
1	2	2	X	.	.	X	Y
2	3	3	.	3	.	X	Y
3	0	0	.	.	1	Y	Y
X	.	.	X	X	1	X, Y	Y
Y	.	0, Y	Y	Y	Y	Y	Y

Figure 3.1

The *Table 3.2* contains a more detailed information about the problems (P.k/XYZ), $k \in \{1, 2, 3\}$, $X \in \{o, i, a, e\}$, $Y, Z \in \{s, w\}$. The left part consists of information about the GPT of all finite algebras. The right part deals with GPT of algebras consisting of at most n elements, where n is sufficiently large, $n \geqq n_0$. The constant n_0 (not necessarily the smallest possible) can be determined by a detailed analysis of the proof of Theorem 3.1, but we do not look for it.

Table 3.2

Class of GPT:		*a* finite				$\lvert a\rvert \leqq n$ $(n \geqq n_0)$			
Problem:		ss	sw	ws	ww	ss	sw	ws	ww
P. 1	o	U	U	U	U	S	U	U	U
	i	U	U	U	U	S	U	U	U
	a	U	U	U	U	S	U	U	U
	e	S, n	U	NO	U	S, n	U	NO	U
P. 2	o	U	U	U	U	S	U		U
	i	U	U	U	U	S	U		U
	a	U	U	NO	NO	S	U	NO	NO
	e	S, n	S, n	NO	NO	S, n	S, n	NO	NO
P. 3	o	U	U	U	U	S	U	U	U
	i	U	U	U	U	S	U	U	U
	a	U	U	U	U	S	U	U	U
	e	NO	U	NO	NO	NO	U	NO	NO

Attention: $\lvert u\rvert \geqq 2$ and $\lvert B\rvert \geqq 2$ is assumed.

U ... unsolvable	S ... solvable
NO ... always „no“	YES ... always „yes“
n ... more often „no“	y ... more often „yes“

3.3. Remarks. 1) In the *Table 3.2* the letter U denotes algorithmic unsolvability and the letter S algorithmic solvability. The words YES, NO mean that the answer is always "yes" or "no". (Hence they also mean algorithmic solvability; maybe, some S in the table can also be replaced by NO or YES. Notice that YES does not occur in *Table 3.2* but it will occur in similar tables below.) If beside S the letter y or n is written it means that more typical or more usual answer is "yes" or "no". (The criteria for writing y or n depend on the problem, and they also can be a little bit subjective.)

2) If the problem (P.1/oss) is algorithmically unsolvable for a class of GPT, then the problems (P.1/osw), (P.1/ows), (P.1/oww) are automatically also algorithmically unsolvable because (P.1/oss) can be reduced to each of them. (Roughly speaking, it is contained in them.) However, *Table 3.2* is filled up for $|u| \geqq 2$, $|\mathrm{B}| \geqq 2$. This manner allows us to read a more detailed information from the table. E.g. the answer U for (P.1/oww) in the table means a little more than the sentence "(P.1/oww) is unsolvable" in Theorem 3.1. (The trivial reduction described above does not work in *Table 3.2.*) On the other hand, we must be careful in reading *Table 3.2.* For example, the problem (P.2/aww) is unsolvable (despite NO in the table) because U is in the table for (P.2/asw); (P.2/aww) is unsolvable for $|u| = 1$.

We do not prove the answers U in *Table 3.2* (it is partially done in the sketch of proof of Theorem 3.1) but we prove all other answers.

The word u, $|u| \geqq 2$ cannot occur in the 0-th row of GPT($\mathscr{A}$) which has the length 1. Therefore NO occurs in the window for (P.1/ews). The lengths of the words from u^* are multiplies of $|u| \geqq 2$. However, every generalized Pascal triangle contains infinitely many lines the lengths of which are not multiples of $|u|$. Therefore NO occurs in the windows for (P.2/XwZ), $X \in \{\mathrm{a}, \mathrm{e}\}$, $Z \in \{\mathrm{w}, \mathrm{s}\}$. Every line of every GPT contains only finitely many subwords, and B^+ is infinite. Therefore NO occurs in the windows for (P.3/ews), (P.3/eww). The 0-th row of GPT($\mathscr{A}$) contains only one symbol and $|\mathrm{B}| \geqq 2$; therefore NO occurs in the window for (P.3/ess).

Consider now the problem (P.1/ess). If for an algebra $\mathscr{A} = (\mathrm{A}; \mathrm{K}, \mathrm{l}, \mathrm{r}, .)$ and $x \in \mathrm{A}$ the answer is "yes", then

$$\mathrm{K} = x \text{ and } (\mathrm{l}(\mathrm{K}) = \mathrm{K} \text{ or } \mathrm{r}(\mathrm{K}) = \mathrm{K}) . \tag{3.T.1}$$

Conversely, if $\mathrm{K} = x$ and for example $\mathrm{l}(\mathrm{K}) = \mathrm{K}$, then the left margin consists of x only, and hence the answer is "yes". The condition (3.T.1) can be easily verified, and hence S must be in the window. It cannot be replaced by NO or YES because there are pairs ($\mathscr{A}$, x) satisfying (3.T.1) as well as those not satisfying (3.T.1). However, for A fixed, $|\mathrm{A}| \geqq 4$ there are more pairs ($\mathscr{A}$, x) which do not satisfy (3.T.1). Therefore n is also written in the window for (P.1/ess).

Consider now the problem (P.2/esw). If for an algebra $\mathscr{A} = (\mathrm{A}; \mathrm{K}, \mathrm{l}, \mathrm{r}, .)$ and for a word $u \in \mathrm{A}^+$ the answer is "yes" then

$$u \in K^+ \text{ and } l(K)=K \text{ and } r(K)=K \text{ and } K \cdot K = K. \tag{3.T.2}$$

Conversely, if (3.T.2) holds, then the answer is "yes". The condition (3.T.2) obviously can be verified, and hence we have S in the window for (P.2/esw) (and also for (P.2/ess) because the assumption $|u| \geqq 2$ is not substantially used). For every fixed A, m, $|A| \geqq 2$, $m \geqq 1$ there are pairs $(\mathscr{A}, u)$, $|u| = m$ which satisfy (3.T.2) but there are more pairs $(\mathscr{A}, u)$, $|u| = m$ not satisfying this condition. Therefore S cannot be replaced by NO or YES, and n is written beside S.

It remains to explain S for the problems (P.k/Xss), $k \in \{1, 2, 3\}$, $X \in \{o, i, a\}$ for $|\mathscr{A}| \leqq n$, where $n \geqq 1$ is fixed (the condition $n \geqq n_0$ is necessary only for the answers U). The considered class contains only finitely many nonisomorphic algebras, and hence the problems (P.k/Xss) contain only finitely many instantions in essential. Therefore they can be solved using a table of answers for these instantions. (However, we have no algorithm which constructs the appropriate table for arbitrary given $n \geqq 1$.)

4. Simple semilinear languages

In this section simple semilinear languages are introduced. They form a subclass of the class of semilinear languages (not necessarily context-free ones) which are studied e.g. in [4] and [6]. We shall show that if the language R($\mathscr{A}$) or R($\mathscr{A}$, w) is simple semilinear, all problems (P.k/XYZ) from Theorem 3.1 and Remark 3.2.1 are decidable. The assumption that $\mathscr{A}$ is finite can be sometimes weakened or omitted at all. However, as it will be seen later, only a seeming generalization arises in this way. Indeed, if R($\mathscr{A}$, w) (or R($\mathscr{A}$)) is a language (i.e. if it is a subset of some B^* for a finite set B), then there is a finite algebra $\mathscr{B}$ such that GPT($\mathscr{A}$, w) = GPT($\mathscr{B}$, w). ($\mathscr{B}$ need not be a subalgebra of $\mathscr{A}$.)

4.1. Definition. 1) A language L will be called a simple linear language of degree at most k if there are words

$$u_0, u_1, v_1, u_2, v_2, \ldots, v_k, u_k \tag{4.1.1}$$

such that

$$L = \{u_0 v_1^i u_1 v_2^i u_2 \ldots v_k^i u_k \,;\, i \geqq 1\} \tag{4.1.2}$$

2) A language L is said to be a simple semilinear language (abbreviation: SSL language) of degree at most k if L is the union of finitely many pairwise disjoint simple linear languages of degree at most k.

3) A language L is a simple (semi-)linear language of degree k if L is a simple (semi-)linear language of degree at most k, and L is not a simple (semi-)linear

language of degree at most $k-1$. A language L is a simple (semi-)linear language if L is a simple (semi-)linear language of degree at most k for some $k \in \mathrm{N}$.

4) The finite sequence (4.1.1) will be called a generating sequence for L if (4.1.2) holds. A generating system for a simple semilinear language L is a set which consists of generating sequences for a finite set M of pairwise disjoint simple semilinear languages such that L is the union of M. We say that a simple semilinear language is given if a generating system for L is given.

Every simple linear language is nonempty but an SSL language can be empty. If (4.1.2) is a simple linear language of degree k, then the words $v_1, v_2, \ldots, v_k$ are nonempty. L is an SSL language of degree 0 if and only if L is finite. If L is an SSL of degree at most 1 (or at most 2) then L is regular (or context-free). The converse does not hold in general but see Theorem 4.7.

The language L from (4.1.2) is a subset of $u_0^* v_1^* u_1^* \ldots v_k^* u_k^*$. We can order a subset M of N^{2k+1} to the language L by the method of the 5-th chapter of [4]. Then the set M is linear; in the notation of [4] we have $\mathrm{M}=\mathrm{L}(c; p)$ where $c=(1, 1, \ldots, 1) \in \mathrm{N}^{2k+1}$ and $p=(0, 1, 0, 1, \ldots, 1, 0) \in \mathrm{N}^{2k+1}$. Hence simple (semi-)linear languages form a subclass of the class of those bounded languages to which (semi-)linear subsets of sets N^n, $n=1, 2, 3 \ldots$ correspond. To every SSL language a finite union of sets of the form $\mathrm{L}(c; p)$ corresponds, and vice versa.(Notice that simple linear languages need not be linear or metalinear in the sense of [4].)

Considering semilinear sets associated to SSL languages by the method of [4] the following theorem can be proved.

4.2. Theorem. Let L_1, L_2 be languages, x, y be words and k, m, n be nonnegative integers. Then:

1) If L_1 is an SSL language (of degree k) and the symmetric difference $\mathrm{L}_1 \dot{-} \mathrm{L}_2$ is finite then L_2 is an SSL language (of degree k).

2) If L_1 is a simple (semi-)linear language (of degree k) then $x\mathrm{L}_1 y$ is a simple (semi-)linear language (of degree k).

3) If L_1 is an SSL language (of degree k) and

$$\mathrm{L}_2=\{w ; \exists x, y(|x|=m, |y|=n \text{ and } xwy \in \mathrm{L}_1)\},$$

then L_2 is an SSL language (of degree k).

4) The reflection of a simple (semi-)linear language (of degree k) is a simple (semi-)linear language (of degree k).

5) The union, the intersection, the difference, and the symmetric difference of SSL languages (of degree at most k) are SSL languages (of degree at most k).

6) If L_1 is a simple (semi-)linear language (of degree $k \geqq 1$) then $\mathrm{L}_2= \{w \in \mathrm{L}_1 ; |w| \geqq n\}$ is a simple (semi-)linear language (of degree k).

4.3. Remark. The points 1), 3), 5) of the above theorem do not hold for simple linear languages. If, for example, (4.1.2) is infinite (i.e. $v_i \neq \varepsilon$ for some i), and we delete $u_0v_1^3u_1v_2^3u_2 \dots v_k^3u_k$ from it, then the obtained language is nèither simple linear nor a disjoint union of two simple linear languages.

4.4. Lemma. Let $\mathscr{A}$ be (arbitrary) algebra, k be a positive integer and $w \in A^+$. Let

$$u_0, u_1, \dots, u_k \in A^*, \quad v_1, \dots, v_k \in A^+,$$

$$i \geqq |v_1| + \dots + |v_k| + 2 \cdot \max(|v_1|, \dots, |v_k|)$$

and let for $j = i$, $j = i+1$ the word

$$u_0v_1^ju_1v_2^ju_2 \dots v_j^ju_k \tag{4.4.1}$$

belong to $R(\mathscr{A}, w)$. Then for every $j \geqq i$ the word (4.4.1) belongs to $R(\mathscr{A}, w)$.

Proof. For this proof denote (4.4.1) by r_j. The assumption is r_i, $r_{i+1} \in R(\mathscr{A}, w)$. We prove $r_{i+2} \in R(\mathscr{A}, w)$; then an easy induction gives the lemma.

Denote $m = |v_1| + \dots + |v_k|$ and consider the finite sequence

$$w_0, w_1, w_2, \dots, w_s \tag{4.4.2}$$

of all subwords of the length $m+1$ of the word $\mathcal{S}^m r_i \mathcal{S}^m$. Then $s = |r_i| + m = |r_{i+1}|$, and Lemma 2.7 implies that for every $x = 0, 1, \dots, s$ the x-th symbol of r_{i+1} is $\text{next}^m(w_x)$. Denote by U_0 the segment of (4.4.2) consisting of the first $|u_0|$ members, V_1 the segment consisting of the further $|v_1^{i+1}|$ members, U_1 the segment consisting of the further $|u_1|$ members, etc. up to the segment U_k which consists of last $|u_k|$ members of (4.4.2). Thus the whole sequence is divided into the segments

$$U_0, V_1, U_1, V_2, U_2, \dots, V_k, U_k$$

so that the function next^m always gives u_x, v_x^{i+1} from the segment U_x, V_x, respectively.

For every $x = 0, 1, \dots, k$ the segment U_x contains only such subwords of $\mathcal{S}^m r_i \mathcal{S}^m$ which intersect the word u_x. (The occurence of u_x clear from (4.4.1) is understood; its position is clear even if u_x is empty.)

Every segment V_x, $x = 1, \dots, k$, contains a subsegment of length at least $2 \cdot |v_x|$ which contains only subwords of v_x^i (from r_i), and gives only a subword of v_x^{i+1} (from r_{i+1}). Therefore V_x can be divided into three segments A_x, B_x, C_x so that B_x contains only subwords of v_x^i, and gives an occurrence of v_x in v_x^{i+1}. So the whole sequence (4.4.2) is divided into segments

$$U_0, A_1, B_1, C_1, U_1, A_2, B_2, C_2, U_2, \dots, A_k, B_k, C_k, U_k\,.$$

Let us form the finite sequence $w_0', w_1', \dots, w_r'$ from r_{i+1} analogously as we have formed (4.4.2) from r_i. With respect to the structure of r_{i+1} the sequence $w_0', w_1', \dots, w_r'$ can be divided into the segments

$$U_0, A_1, B_1, B_1, C_1, U_1, A_2, B_2, B_2, C_2, U_2, \dots, A_k, B_k, B_k, C_k, U_k\,.$$

The last sequence gives (using the function next^m) the word which arises from r_{i+1} if for every $x=1, \ldots, k$ we replace an occurrence of v_x by the word v_x^2. So we obviously obtain the word r_{i+2}, which completes the proof.

4.5. Lemma. Let $\mathscr{A}$ be an (arbitrary) algebra, k be a positive integer, and $w \in \mathrm{A}^+$. Let a row of GPT($\mathscr{A}$, w) be expressed in the form

$$u_0 v_1^i u_1 v_2^i u_2 \ldots v_k^i u_k$$

for some integer $i \geqq 2$ and some words

$$u_0, u_1, \ldots, u_k \in \mathrm{A}^*, \quad v_1, \ldots, v_k \in \mathrm{A}^+ .$$

Then the next line can be expressed in the form

$$x_0 y_1^{i-1} x_1 y_2^{i-1} x_2 \ldots y_k^{i-1} x_k$$

so that $|x_{s-1}| = |u_{s-1}| + |v_s|$, $|y_s| = |v_s|$ for $s=1, \ldots, k$ and $|x_k| = |u_k| + 1$.

Proof. For every $s=1, \ldots, k$ denote the last symbol of v_s by d_s. Then the words $x_0, y_1, x_1, \ldots, y_k, x_k$ are (uniquely) determined by the formulae

$$\begin{gathered} x_0 = \text{next}(\$ u_0 v_1), \quad x_k = \text{next}(d_k u_k \$) \\ y_s = \text{next}(d_s v_s) \quad \text{for} \quad s=1, \ldots, k, \\ x_s = \text{next}(d_{s-1} u_s v_s) \quad \text{for} \quad s=1, \ldots, k=1. \end{gathered}$$

4.6. Theorem. For an (arbitrary) algebra $\mathscr{A}$, every $w \in \mathrm{A}^+$ and every positive integer s the following conditions are equivalent:

(i) The set R($\mathscr{A}$, w) of all rows of GPT($\mathscr{A}$, w) is a simple semilinear language of degree at most s.

(ii) The set R($\mathscr{A}$, w) contains an infinite simple linear language of degree at most s.

(iii) There are positive integers $k \leqq s$, words

$$u_0, u_1, \ldots, u_k \in \mathrm{A}^*, \quad v_1, \ldots, v_k \in \mathrm{A}^+ \tag{4.6.1}$$

and an integer i such that

$$i \geqq 2 \cdot (|v_1| + \ldots + |v_k|) + 2 \cdot \max(|v_1|, \ldots, |v_k|) \tag{4.6.2}$$

and both words

$$u_0 v_1^i u_1 v_2^i u_2 \ldots v_k^i u_k, \quad u_0 v_1^{i+1} u_1 v_2^{i+1} u_2 \ldots v_k^{i+1} u_k \tag{4.6.3}$$

belong to R($\mathscr{A}$, w).

Proof. The implications (i)→(ii), (ii)→(iii) are obvious. Now let (iii) hold. We shall prove that R($\mathscr{A}$, w) is an SSL language of degree at most k. Let $n = |v_1| + \ldots + |v_k|$, and let $t, t+n$ be the numbers of the rows (4.6.3) of GPT($\mathscr{A}$, w). Consider n pairs of rows of GPT($\mathscr{A}$, w), where the m-th pair

($m=0, 1, \ldots, n-1$) consists of the $(t+m)$-th row and the $(t+m+n)$-th row of GPT($\mathscr{A}$, w). Lemma 4.5 implies that for every $m=0, \ldots, n-1$ the m-th pair is of the form

$$x_0 y_1^j x_1 y_2^j x_2 \ldots y_k^j x_k,\ x_0 y_1^{j+1} x_1 y_2^{j+1} x_2 \ldots y_k^{j+1} x_k$$

where

$$|y_1|=|v_1|, \ldots, |y_k|=|v_k|,\ j=i-m$$

and hence

$$j \geqq |y_1|+\ldots+|y_k|+2\cdot\max(|y_1|, \ldots, |y_k|)\,.$$

Then Lemma 4.4 implies that

$$L_m=\{x_0 y_1^r x_1 y_2^r x_2 \ldots y_k^r x_k;\ r \geqq j\} \subseteq R(\mathscr{A}, w)\,.$$

Denote by Y the set of t first lines of GPT($\mathscr{A}$, w). Then it holds

$$R(\mathscr{A}, w)=Y \cup L_0 \cup L_1 \cup \ldots \cup L_{n-1}\,. \tag{4.6.4}$$

Indeed, the inclusion $\supseteq$ is obvious. To show the converse, realize that for every $q \geqq |w|$ every side of (4.6.4) contains exactly one word of length q. Similar argument also shows that Y, L_0, L_1, …, L_{m-1} are pairwise disjoint. Hence R($\mathscr{A}$, w) is an SSL language of degree at most k, and since $k \leqq s$, we have proved (i).

From Theorem 4.6 and well known results about context-free and regular languages we can obtain the following consequence.

4.7. Theorem. For an (arbitrary) algebra $\mathscr{A}$ and every $w \in A^+$:

1) The set R($\mathscr{A}$, w) is a regular language if and only if R($\mathscr{A}$, w) is an SSL language of degree 1.

2) The set R($\mathscr{A}$, w) is a context-free language if and only if R($\mathscr{A}$, w) is an SSL language of degree 1 or 2.

4.8. Theorem. There is an algorithm which for every recursive algebra $\mathscr{A}$ and every $w \in A^+$ gives a generating system for the set R($\mathscr{A}$, w) of all lines of GPT($\mathscr{A}$, w) provided R($\mathscr{A}$, w) is simple semilinear language (and which does not halt when R($\mathscr{A}$, w) is not an SSL language).

Proof. Let the function $F=$ GPT($\mathscr{A}$, w) be recursive, and we know how to compute the values of F. (Only for this reason we need the recursivity of $\mathscr{A}$.) The algorithm will form (in an arbitrary order) all pairs of rows of $\mathscr{A}$. For every pair it will find out whether it can be represented in the form (4.6.3) for some words (4.6.1) and integers k, i which satisfy (4.6.2). (It is a finite process for every pair of rows.) As soon as such a test is positive, the algorithm will stop generating the pairs of rows, and will follow the proof of Theorem 4.6 to obtain (4.6.4).

4.9. Remark. The above algorithm can be modified so that for every given recursive algebra $\mathscr{A}$, $w \in A^+$ and positive integers m it will find out whether

$R(\mathscr{A}, w)$ is a disjoint union of m simple linear languages. (Hence it will always stop.) However, neither for $m = 1$ and finite algebras it can be modified so that it will find out whether $R(\mathscr{A}, w)$ is a disjoint union of m simple linear languages and a finite set.

4.10. Theorem. For every problem $(P.k/XYZ)$, $k \in \{1, 2, 3\}$, $X \in \{o, i, a, e\}$, $Y, Z \in \{w, s\}$ there is an algorithm which solves this problem for every recursive algebra $\mathscr{A}$, $x \in A$, $u, v \in A^+$ and recursive set $B \subseteq A$ provided $R(\mathscr{A}, w)$ (if $Z = w$), or $R(\mathscr{A})$ (if $Z = s$) is a simple semilinear language (and which does not halt when $R(\mathscr{A}, w)$ or $R(\mathscr{A})$ is not an SSL language).

Proof. Consider the cases $Z = w$ which are more general. The appropriate algorithm will look for the generating system for $R(\mathscr{A}, w)$ at first. (If $R(\mathscr{A}, w)$ is not an SSL language the algorithm will not finish this stage, hence it will not halt.) If the generating system is found, then A can be replaced by a finite set $A_1 \subseteq A$, and B by a finite set B_1 (provided $k = 3$, of course).

Further, we shall assume that a representation (4.6.4) for $R(\mathscr{A}, w)$ is given, and that $A_1 = A$, $B_1 = B$. The problem $(P.k/XYZ)$ will be reduced to some problems concerning infinite simple linear languages $L_0, L_1, \ldots, L_{n-1}$ and a finite set Y. As a representant of infinite semilinear languages we consider below the language L from (4.1.2); we assume that $v_1, \ldots, v_k$ are nonempty.

Each of the problems $(P.1/Xww)$ can be reduced to several problems concerning a finite set Y, and to the problems whether u is a subword of at least one word, infinitely many words, almost all words, or all words of the languages L_m.

Table 4.1

Class of GPT:		$R(\mathscr{A})$ or $R(\mathscr{A}, w)$ is an SSL language			
Problem:		ss	sw	ws	ww
P. 1	o	S	S	S	S
	i	S	S	S	S
	a	S	S	S	S
	e	S	S	NO	S
P. 2	o	S	S	S	S
	i	S	S	S	S
	a	S	S	NO	NO
	e	S	S	NO	NO
P. 3	o	S	S	NO	NO
	i	S	S	NO	NO
	a	S	S	NO	NO
	e	S	S	NO	NO
Attention: $\lvert u \rvert \geqq 2$ and $\lvert B \rvert \geqq 2$ is assumed.					

However, u is a subword of (4.4.1) for some $j \geqq |u|$ if and only if u is a subword of (4.4.1) for all $j \geqq |u|$. Therefore to answer all questions above for $L = L_m$ only finitely many words (4.4.1) for $j \leqq |u|$ must be checked. Hence the problems (P.1/Xww), and also their special cases with s instead of w, are solvable (provided R($\mathscr{A}$, w) is SSL language).

The problems (P.2/XYZ) can be reduced to some problems of emptiness or finiteness of an SSL language by Theorem 4.2; hence they also are solvable.

The problems (P.3/XsZ) can be reduced to the problems (P.1/XYZ), hence they also are solvable in the considered case. For $|B| \geqq 2$ the problems (P.3/XwZ) must be always answered "no" because Sub(R($\mathscr{A}$, w)) is a bounded language and B^+ is not bounded. (This explains six new NO in Table 4.1.) For $|B| = 1$ we have to know that Sub(L) contains B^+ if and only if some v_i, $i = 1, \ldots, k$, belongs to B^+. The reduction is similar to that of problems (P.1/XYZ).

Table 4.1 summarizes the information about the problems (P.k/XYZ) for the case when R($\mathscr{A}$, w) is simple semilinear.

4.11. Remark. If R($\mathscr{A}$, w) is a semilinear language then it is a simple semilinear language because every other semilinear language contains words of the same length. However, R($\mathscr{A}$, w) can be bounded and not semilinear.

5. Embeddings of generalized Pascal triangles

We shall consider only the embeddings based on the translations of the sets $Z \times Z$. The role of embeddings is to reduce some problems concerning GPT to the problems concerning another GPT, which is either simpler or already investigated.

5.1. Definition. Let F, G be mappings of subsets X, Y of the set D into the set $\mathscr{S}$. We shall say that F can be embedded into G if there are a, $b \in Z$ such that for every $(x, y) \in X$ it holds $(x + a, y + b) \in Y$ and

$$F(x, y) = G(x + a, y + b)\,. \tag{5.1.1}$$

Every pair (a, b) with the above property will be called an embedding vector of F into G.

The relation of embedding is obviously a quasi-ordering, i.e. it is reflective and transitive. We shall mainly consider the case when F, G are generalized Pascal triangles. Then $X = D_{m,n}$, $Y = D_{p,q}$ for some integers $m \leqq n$, $p \leqq q$ (see (2.2.1)) which implies $a + b \geqq 0$ (and some other conditions depending on m, n, p, q) for every embedding vector of F into G. Theorems 5.6 and 5.7 will be used if some properties of G ought to be proved from some known properties of F or vice versa. However, also for $F = G$ existence of a nontrivial embedding (i.e. one with non-zero embedding vector) gives a nontrivial information about F and G. This situation is described in Theorem 5.4.

Corollary 5.8 combines these results into the form which often seems to be the most suitable for further application of the above mentioned results. The following theorem shows that arbitrarily wide margins of GPT can be replaced by suitable margins of width 1.

5.2. Theorem. Let $m \leqq n$ be integers and F be a mapping of $\mathrm{D}_{m,n} = \{(x, y) \in \mathrm{D}; x \geqq m, y \geqq -n\}$ into $\mathscr{S}$. Then the following conditions are equivalent:

(i) F can be embedded into a GPT($\mathscr{A}$, w) for a finite algebra $\mathscr{A}$ and a word $w \in \mathrm{A}^+$.

(ii) F can be embedded into GPT($\mathscr{B}$, v, $m-1$) with the embedding vector $(0, 0)$ for a finite algebra $\mathscr{B}$ and a word $v \in \mathrm{B}^+$ (which arises from the 0-th row of F by adding a new initial symbol and a new final symbol).

Proof. The implication (ii)$\rightarrow$(i) is obvious; notice that instead of GPT($\mathscr{A}$, w) also GPT($\mathscr{A}$, w, i) can be considered. Now assume (i). Since $\mathscr{A}$ is finite then the left and the right margins of F are ultimately periodic by Theorem 2.10. Let their periods be p, q and their pre-periods s, t, respectively. Let F be extended to the $(m-1)$-th left diagonal and to the $(-n-1)$-th right diagonal (so the new domain will be $\mathrm{D}_{m-1,n+1}$) as follows. Take $s+p$ new symbols for the left margin and $t+q$ new symbols for the right margin ("new" means that they do not belong to A). Then write these elements into the first $s+p$ or $t+q$ places of the $(m-1)$-th left and $(-n-1)$-th right diagonals, respectively. Then the groups of the last p, or q elements will be periodically repeated on the appropriate diagonals. Let G be the extended function. An algebra $\mathscr{B}$ can be constructed such that $G = \mathrm{GPT}(\mathscr{B}, v, m-1)$, where v is the 0-th row of G. The base set of B will consist of the elements of A and the $p+q+s+t$ new elements on the margins of G. Its binary operation will be an extension of that of $\mathscr{A}$; the new values will be used only for the m-th left and $(-n)$-th right diagonals. The nulary operation of $\mathscr{B}$ is not substantial, and the unary operations serve only for the margins, hence only the values of l for $p+s$ new elements and r for $q+t$ new elements are important.

5.3. Remarks. 1) Theorem 5.2 holds also for infinite algebras $\mathscr{A}$, $\mathscr{B}$ provided that the set $\mathscr{S} - \mathrm{A}$ is infinite (or that $\mathscr{S}$ can be enlarged if necessary). In this case one-to-one sequences ought to be written into the new margins.

2) A question arises whether the margin is necessary at all, i.e. whether the conditions (i), (ii) are equivalent with the condition that F is a GPT. The answer is negative, as Figure 2.2e shows. After removing the left and the right margins (of width 1) there is no suitable operation r.

5.4. Lemma. Let F be a mapping of the set $\mathrm{D}_{m,n}$ into a finite set A. Let all left and right diagonals of F be ultimately periodic, and let F be nontrivially embedded into itself with an embedding vector $(a, b) \neq (0, 0)$. Then the set M of all rows of F is a simple semilinear language of degree at most 2; if $a=0$ or $b=0$, then M is an SSL language of degree 1.

Proof. Denote $c=a+b$ and r_k the k-th row of the function F, x_k its initial subword of length a, and a_k its final subword of length b. (Symbols $\mathcal{S}$ are used for small k if necessary.) The sequences

$$(x_0, x_1, x_2, \ldots), \quad (y_0, y_1, y_2, \ldots)$$

are ultimately periodic. Let p be the least common multiple of their periods, let q be the maximum of their pre-periods and $K=q+pc$. Further we always assume $k>K$. By the assumptions we have

$$\mathrm{r}_k = x_k \mathrm{r}_{k-c} y_k = x_k x_{k-c} \mathrm{r}_{k-2c} y_{k-c} y_k = \ldots = x_k x_{k-c} \ldots x_{k+c-pc} \mathrm{r}_{k-pc} y_{k+c-pc} \ldots y_{k-c} y_k .$$

Denote

$$w_k = x_k x_{k-c} \ldots x_{k+c-pc}, \quad z_k = y_{k+c-pc} \ldots y_{k-c} y_k .$$

Then

$$w_k = w_{k-pc}, \quad z_k = z_{k-pc}, \quad \mathrm{r}_k = w_k \mathrm{r}_{k-pc} z_k .$$

If we replace k by $k+pc$ we obtain

$$\mathrm{r}_{k+pc} = w_{k+pc} \mathrm{r}_k z_{k+pc} = w_k \mathrm{r}_k z_k = w_k^2 \mathrm{r}_{k-pc} z_k^2,$$

and by an easy induction

$$\mathrm{r}_{k+pcs} = w_k^{s+1} \mathrm{r}_{k-pc} z_k^{s+1}$$

If we consider this formula for all k, $K<k\leqq K+pc$, and use the length argument similar to that for (4.6.4) we obtain

$$\mathrm{M} = \{\mathrm{r}_i\,;\, i<K\} \cup \bigcup_{k=K+1}^{K+pc} \{w_k^s \mathrm{r}_{k-pc} z_k^s\,;\, s \geqq 1\} \tag{5.4.1}$$

Hence we have expressed M as the union of a finite set and pc simple linear languages. We can immediately see that M is an SSL of degree at most two. If $a=0$ (or $b=0$) then $|w_k|=0$ (and $|z_k|=0$) for all k, respectively and hence (5.4.1) is an SSL of degree 1.

5.5. Remark. If $a=0$ in the above lemma, then $b\neq 0$, and every left diagonal can be nontrivially embedded into itself. Hence all left diagonals are periodic (and their periods divide b). The right diagonals need not be periodic. However, the set of their preperiods is bounded. Hence if we remove a suitable left margin of F, then the right diagonals of the obtained function G will be periodic. The set of their periods is finite, and hence G can be embedded into itself also with an embedding vector $(a_1, 0)\neq(0, 0)$. Of course, a similar consideration can be made for $b=0$. In both cases we have obtained two linearly independent embedding vectors.

Conversely, let there be two linearly independent embedding vectors (a_1, b_1), (a_2, b_2) for embeddings of F into F, and for example $a_1 b_2 < a_2 b_1$. Then

$$a_1 \cdot (a_1, b_1) - a_1 \cdot (a_2, b_2) = (0, a_2b_1 - a_1b_2)$$
$$b_1 \cdot (a_2, b_2) - b_2 \cdot (a_1, b_1) = (a_2b_1 - a_1b_2, 0)$$

also are embedding vectors of F into F, and the set M is regular.

5.6. Theorem. Let F be a mapping of the set $D_{m,n}$ into a finite set A, and let all left and right diagonals of F be ultimately periodic. Further, let M be the set of all rows of F. Then the following conditions are equivalent:

(i) The set M is a simple semilinear language of degree 1.
(ii) If we remove sufficiently wide margins of F, then the obtained function G can be nontrivially embedded into itself with an embedding vector $(a, b) \neq (0, 0)$, where $a = 0$ or $b = 0$.
(iii) If we remove sufficiently wide margins of F then the obtained function G can be nontrivially embedded into itself with two linearly independent embedding vectors.

Proof. If (ii) holds then the set M_1 of all rows of G is an SSL language of degree 1 by Lemma 5.4. Since M is infinite, it can be obtained from M_1 by the operations from Theorem 4.2, the set M also being an SSL language of degree 1. The equivalence (ii)→(iii) was proved in Remark 5.5. It remains to prove (i)→(ii). Let

$$M = Y \cup \bigcup_{i=1}^{k} \{u_i v_i^s w_i\,;\ s \geqq 1\}$$

where Y is a finite set consisting of the $|Y|$ first rows of M, all $v_1, \ldots, v_k$ are nonempty, and the $k+1$ sets on the right are pairwise disjoint. Let a be the least common multiple of $|v_1|, \ldots, |v_k|$. Then the embedding vector we are looking for can be $(a, 0)$ (also $(0, a)$ is suitable). To obtain the function G, it suffices to remove such left margin of F that the elements of Y and the initial $u_1, \ldots, u_k$ in all other rows vanish. Hence the width

$$\max(|u_1|, \ldots, |u_k|, |Y| + 1 + n - m)$$

is sufficient. (For $(0, a)$ a suitable right margin must be removed, and for both embedding vectors both margins can be necessary.)

Theorem 5.6 can be obviously used for $F = \mathrm{GPT}(\mathcal{A}, w)$, where $\mathcal{A}$ is finite. An exact analogy for the case when M is an SSL language of degree 2 does not hold. To show that, consider the language

$$M = \{0^{3i}\,;\ i \geqq 1\} \cup \{0^{2i}1^{i+1}\,;\ i \geqq 1\} \cup \{0^i 1^{2i+2}\,;\ i \geqq 1\}\,.$$

The corresponding F can be easily constructed. However, F cannot be embedded into itself because the vectors $(2, 1)$ and $(1, 2)$ (suitable for the second and the third parts of M, respectively) are linearly independent. By Lemma 4.5 this bad case cannot take place for GPT($\mathcal{A}$, w). Therefore the following theorem holds.

5.7. Theorem For every finite algebra $\mathcal{A}$ and every $w \in A^+$ the following conditions are equivalent:

(i) The set R($\mathcal{A}$, w) is a simple semilinear language of degree 2.
(ii) If we remove sufficiently wide margins of GPT($\mathcal{A}$, w), then the obtained function G can be nontrivially embedded into itself, and every nontrivial embedding vector has both components positive.
(iii) For G as in the previous point, every two embedding vectors of G into G are linearly dependent.

Combining the above theorems with Theorem 4.10 we obtain:

5.8. Corollary. The problems

$$(\mathrm{P}.k/XYZ),\ k \in \{1, 2, 3\},\ X \in \{\mathrm{o, i, a, e}\},\ Y, Z \in \{\mathrm{s, w}\} \tag{5.8.1}$$

are algorithmically solvable for the class of all generalized Pascal triangles GPT($\mathcal{A}$, w), $\mathcal{A}$ finite, which can be nontrivially embedded into itself after removing sufficiently large margins.

The finiteness of $\mathcal{A}$ can be weakened similarly as in Theorem 4.10. Notice that also for finite algebras this corollary is weaker than Theorem 4.10 because it deals only with SSL of degree at most 2. However, it is more comfortable for applications.

6. Pascal triangles modulo p

In the whole section p will mean a positive integer, most often a prime. We shall investigate generalized Pascal triangles for the following algebras.

6.1. Definition. For every positive integer p let M_p denote the set $\{0, 1, \ldots, p-1\}$ and $\mathcal{M}_p$ the algebra $(M_p;\ 1, \mathrm{l}, \mathrm{r}, +)$ where $\mathrm{l}(x)=\mathrm{r}(x)=x$ for all $x \in M_p$ and $+$ means the addition modulo p. All GPT($\mathcal{M}_p$, w, i) will be called generalized Pascal triangles modulo p.

Hence GPT($\mathcal{M}_p$) can be obtained from the usual Pascal triangle if all its elements are reduced modulo p. If we want to extend GPT($\mathcal{M}_p$) or GPT($\mathcal{M}_p$, w) to the whole set D, we shall use the value 0 instead of $\mathcal{S}$ which was used in the previous sections. In this way (and only in this way) all diagonals will be periodic, and not only ultimately periodic.

6.2. Definition. For every positive integer p and every integer i we denote by F_p^i the mapping of D into M_p defined by the formula

$$F_p^i(x, y) = \begin{cases} 0 & \text{if } x+y=0, x \neq i \\ 1 & \text{if } x=i, y=-i \\ F_p^i(x-1, y)+F_p^i(x, y-1) & \text{if } x+y>0 \end{cases} \quad (6.2.1)$$

The upper index i will be sometimes omitted if $i=0$.

Hence $F_p = \mathrm{GPT}_0(\mathcal{M}_p)$ and $F_p^i = \mathrm{GPT}_0(\mathcal{M}_p, 1, i)$ for all i. The role of the functions F_p^i for GPT modulo p is explained in the next theorem.

6.3. Theorem. For every positive integer p the set

$$\{\mathrm{GPT}_0(\mathcal{M}_p, w, i);\ w \in M_p^+, i \in Z\} \quad (6.3.1)$$

of generalized Pascal triangles modulo p (which are extended to D by the value 0) is a modul (if p is a prime, then a vector space) over the ring (resp. the field) $(M_p;\ +, .)$ of residue classes modulo p. A base of this modul (resp. vector space) is the set $\{F_p^i;\ i \in Z\}$, and for every $a_0, a_1, \ldots, a_n \in M_p$, $k \in Z$ it holds

$$\mathrm{GPT}_0(\mathcal{M}_p, a_0a_1 \ldots a_n, k) = \sum_{i=0}^{n} a_i \cdot F_p^{i+k}. \quad (6.3.2)$$

Proof. All mappings of D into M_p obviously form a modul (resp. a vector space) over $(M_p;\ +, .)$. A mapping f of D into M_p belongs to the set (6.3.1) if and only if it has only finitely many nonzero values in the 0-th rows and for all $x, y \in Z$, $x+y>0$

$$F(x, y) = F(x-1, y) + F(x, y-1)\,.$$

Both these conditions are preserved by linear combinations, (6.3.1) is nonempty, and hence (6.3.1) is a modul. The proof of (6.3.2) is trivial; it must be verified in the 0-th row at first. It remains to show that the functions F_p, $i \in Z$ are linearly independent, which is also easy. The values in the 0-th row must be again considered.

6.4. Remark. Both $\mathrm{GPT}_0(\mathcal{M}_p, w, i)$ and $\mathrm{GPT}(\mathcal{M}_p, w, i)$ will be called generalized Pascal triangles modulo p; they must be distinguished from the context. Analogously $+$ is used for addition modulo p as well as for the usual addition of integers.

6.5. Lemma. For every prime p, for every integers x, y and for every nonnegative integers k the function $F_p = \mathrm{GPT}_0(\mathcal{M}_p, 1, 0)$ satisfies the conditions:

$$p \mid x+y \text{ and } p \nmid x \text{ imply } F_p(x, y) = 0 \quad (6.5.1)$$

$$F_p(p \cdot x, p \cdot y) = F_p(x, y) \quad (6.5.2)$$

$$p^k \mid x+y \text{ and } p^k \nmid x \text{ imply } F_p(x, y) = 0 \quad (6.5.3)$$

$$F_p(p^k \cdot x, p^k \cdot y) = F_p(x, y) \quad (6.5.4)$$

Sketch of proof. (6.5.1) is trivial for $x<0$ or $y<0$. If $x\geqq 0$, $y\geqq 0$ then (6.5.1) is a consequence of the property of binomial coefficients

$$p \,\Big|\, \binom{p}{x} \quad \text{for all } x=1, \ldots, p-1 .$$

(6.5.2) can be proved from (6.5.1) by induction. Since only zeros occur between $F_p(p\cdot x-p, p\cdot y)$, $F_p(p\cdot x, p\cdot x-p)$ we have

$$\begin{aligned} F_p(p\cdot x, p\cdot y) &= F_p(p\cdot x-p, p\cdot y)+F_p(p\cdot x, p\cdot y-p)= \\ &= F_p(x-1, y)+F_p(x, y-1)=F_p(x, y) . \end{aligned}$$

(6.5.3) and (6.5.4) can be proved by induction from (6.5.1) and (6.5.2).

6.6. Remark. The formulae from Lemma 6.5 can be extended also to the functions F_p^i. For example

$$p^k \mid x+y \text{ and } p^k \nmid x \text{ imply } F_p^i(x+i, y-i)=0$$

$$F_p^i(p^k\cdot x+i, p^k\cdot y-i)=F_p^i(x+i, y-i)$$

We shall refer Lemma 6.5 also for the function F_p^i.

6.7. Definition. For every prime p we denote by $M_{p,i}$ the set of all multiples of the i-th row of $\mathrm{GPT}(\mathcal{M}_p)$.

For example, the field $(M_2;+,.)$ contains only one non-zero element 1, and hence

$$M_{2,0}=\{0, 1\},\ M_{2,1}=\{00, 11\},\ M_{2,2}=\{000, 101\},\ M_{2,3}=\{0000, 1111\}, \ldots$$

For $p=3$ we obtain

$$M_{3,0}=\{0, 1, 2\},\ M_{3,1}=\{00, 11, 22\},\ M_{3,2}=\{000, 121,\ 212\}, \ldots$$

These sets will be useful in the formulation of the next theorem about the structure of the rows of $\mathrm{GPT}(\mathcal{M}_p)$.

6.8. Theorem. For every prime p and all $i=0, 1, \ldots, p-1$

$$|M_{p,i}|=p,\quad M_{p,i}\subseteq\{0^{i+1}\}\cup(M_p-\{0\})^{i+1} \tag{6.8.1}$$

The left formula holds for all $i\in N$. Further, denote by r_i the i-th row of $\mathrm{GPT}(\mathcal{M}_p)$ and $q=p^k$ for some integer $k\geqq 1$. Then

$$R(\mathcal{M}_p)\subseteq\bigcup_{i=0}^{q-1}(\{r_i\}\cup r_i 0^{q-1-i}(M_{p,i}0^{q-1-i})^* r_i) \tag{6.8.2}$$

$$\mathrm{Sub}(R(\mathcal{M}_p))\subseteq\bigcup_{i=0}^{q-1}\mathrm{Sub}((M_{p,i}0^{q-1-i})^*) \tag{6.8.3}$$

Proof. The left formula (6.8.1) and the inclusion $M_{p,i}\subseteq M_p^{i+1}$ are obvious from the definitions of $M_{p,i}$. Then the right formula (6.8.1) for $i<p$ follows from the property of binomial coefficients:

$$p \nmid \binom{i}{k} \quad \text{for } 0 \leqq k \leqq i < p,\ p \text{ a prime}.$$

To prove (6.8.2) consider the $(q \cdot x + i)$-th row of $F_p = \mathrm{GPT}(\mathcal{M}_p)$. $0 \leqq i < q$. We may assume $x \geqq 1$. Lemma 6.5 implies that the $q \cdot x$-th row of F_p contains at most $x+1$ nonzero symbols at the 0-th, q-th, ..., $q \cdot x$-th positions. For every symbol $t \neq 0$ in the $q \cdot x$-th row (and also from 0 if it is at one of the mentioned positions) the t-multiple of r_i arises in the $(r \cdot x + i)$-th row. These multiples of r_i will be divided by the groups of $q-1-i$ zeros. Since the $q \cdot x$-th row beins and end with the symbol 1, the $(q \cdot x + i)$-th row begins and end with the subword r_i. Therefore it belongs to the right side of (6.8.2). The formula (6.8.3) is an easy consequence of (6.8.2); notice that $r_i \in M_{p,i}$.

6.9. Lemma. Let p be a prime, $w \in M_p^+ - 0^+$ and let k be the smallest integer such that $|w| \leqq p^k + 1$.Then w occurs in $\mathrm{GPT}(\mathcal{M}_p)$ if and only if w occurs (as a subword) in one of the first p^{k+2} rows of $\mathrm{GPT}(\mathcal{M}_p)$.

Proof. We shall use the induction with respect to k. At first we notice that all nonzero symbols $v \in M_p - \{0\}$ can be found in the first p rows of $\mathrm{GPT}(\mathcal{M}_p)$; the symbol v occurs in the v-th row.

Now assume that a word $w \in M_p^+ - 0^+$, $|w| \geqq 2$ occurs in the $(p \cdot x + i)$-th row of $\mathrm{GPT}(\mathcal{M}_p)$, $0 \leqq i < p$, and consider the segment of the $p \cdot x$-th row on which the occurrence of w depends. The segment has the length at most $|w| + p - 1$, and by Lemma 6.5 it can be obtained from a word $y \in M_p^+ - 0^+$ in the x-th row so that the word 0^{p-1} is put between every two neighbour members of y, and maybe shorter groups of zeros are put at the beginning and the end. Hence for $k>0$ we have

$$|y| \leqq \left\lceil \frac{|w| + p - 1}{p} \right\rceil \leqq \left\lceil \frac{p^k + 1 + p - 1}{p} \right\rceil = p^{k-1} + 1$$

and for $k=0$ it holds $|y| = 1$. In both cases we may use the inductive assumption that y occurs in the first p^{k+1} lines of $\mathrm{GPT}(\mathcal{M}_p)$, i.e. $x < p^{k+1}$. Then w occurs in the first p^{k+2} rows because

$$p \cdot x + i \leqq p \cdot (p^{k+1} - 1) + (p - 1) = p^{k+2} - 1 .$$

6.10. Theorem. Let p be a prime, let $k \geqq n$ be positive integers, $u, w \in M_p^+ - 0^+$ and $|w| \leqq p^n$, $|u| \leqq p^k - p^n + 2$. Then the following conditions are equivalent:

(i) u occurs in $\mathrm{GPT}(\mathcal{M}_p, w)$;
(ii) u occurs in $\mathrm{GPT}(\mathcal{M}_p, w)$ infinitely many times;
(iii) u occurs (as a subword) in one of the first p^{k+2} lines of $\mathrm{GPT}(\mathcal{M}_p, w)$.

Proof. Since the left diagonals of $G=\mathrm{GPT}(\mathcal{M}_p, w)$, are periodical (i) and (ii) are equivalent. (iii) obviously implies (i). It remains to prove that (i) implies (iii).

Theorem 6.8 and (6.3.2) imply that for every x the $p^k \cdot x$-th row of G consists of multiples of w which are divided by the groups of $p^k - |w|$ zeros. The sequence of coefficients of these multiplies coincides with the x-th row of F_p.

Now assume that u occurs in the y-th row of G for the first time, and $y = p^n \cdot x + i$ for some $0 \leqq i < p^n$. If $x = 0$ then obviously $y < p^{k+2}$; let $x \geqq 1$. The segment of the $p^n \cdot x$-th row, which is important for the occurrence of u, has length at most $|u| + p^n - 1$. This segment intersects at most

$$\left\lceil \frac{(|u| + p^n - 1) + p^n - 1}{p^n} \right\rceil \leqq \frac{(p^k + 1) + p^n - 1}{p^n} = p^{k-n} + 1$$

multiples of w, which estimates also the length of the corresponding subword of the x-th line of F_p. Then Lemma 6.9 gives $x < p^{k-n+2}$, and hence

$$y = p^n \cdot x + i \leqq p^n \cdot (p^{k-n+2} - 1) + (p^n - 1) < p^{k+2}$$

which completes the proof.

6.11. Theorem. For the class of algebras $\{\mathcal{M}_p\ ; p \text{ prime}\}$ all problems $(\mathrm{P}.k/XYZ), k \in \{1, 2, 3\}$, $X \in \{\mathrm{o, i, a, e}\}$, $Y, Z \in \{\mathrm{w, s}\}$ are decidable.

Proof. We shall make the necessary consideration for Table 6.1 simultaneously with the proof of the above theorem. Let $\mathcal{A} = \mathcal{M}_p$, p a prime, $x \in \mathrm{M}_p$, $\mathrm{B} \subseteq \mathrm{M}_p$, and $u, w \in \mathrm{M}_p^+$.

Table 6.1

Class of GPT:		$\mathcal{M}_p$, p a prime			
Problem:		ss	sw	ws	ww
P. 1	o	YES	S, y	S	S
	i	YES	S, y	S	S
	a	S, n	S	NO	S
	e	S, n	S	NO	S
P. 2	o	S	S	S, n	S
	i	S, n$^+$	S	S, n$^+$	S
	a	NO	S	NO	NO
	e	NO	S	NO	NO
P. 3	o	YES	S, y	NO	NO
	i	YES	S, y	NO	NO
	a	NO	S	NO	NO
	e	NO	S	NO	NO
$^+$ NO for $p > 2$					
Attention: $\lvert u\rvert \geqslant 2$ and $\lvert B\rvert \geqslant 2$ is assumed.					

Every symbol $x \in \mathrm{M}_p$ occurs in infinitely many rows of $\mathrm{F}_p \in \mathrm{GPT}(\mathcal{M}_p)$. This gives YES in the windows for (P.k/Xss), $k \in \{1, 3\}$, $X \in \{\mathrm{o}, \mathrm{i}\}$. All NO from Table 3.2 can be transfered. Infinitely many rows of F_p belong to the set 10^+1 which gives NO for (P.2/ass), (P.2/ess). Only the symbol 1 occurs in almost all rows of F_p, and it occurs in all of them. So we obtain NO for (P.3/ass) because $|\mathrm{B}| \geqq 2$ is assumed. (For $|\mathrm{B}| = 1$ the answer is "yes" if and only if $p = 2$ and $\mathrm{B} = \{1\}$.) This argument also proves S (and explains n) for (P.1/ass), (P.1/ess). The set $\mathrm{R}(\mathcal{M}_p) \cap x^*$ is nonempty if and only if $x = 1$; it is infinite only if $p = 2$. This proves S for (P.2/oss), (P.2/iss). Now consider the problem (P.3/Xws). Let y, z be different elements of B. If $yzzyyyz$ occurs in F_p then a word of the form $u0t00u$ ($u \neq 0$, $t \neq 0$) occurs (in the preceding row) which contradicts Theorem 6.8. Thus we obtain three new NO. The problem (P.1/ows) can be solved by Lemma 6.9. Since the left diagonals of F_p are periodical the problem (P.1/iws) is equivalent to (P.1/ows). So we obtain two S in the table. If a word u is contained in almost all lines of F_p, then it must occur in the line of the form $10^n 1$ and in the preceding row that does not contain 0, which contradicts $|u| \geqq 2$. So we obtain NO for (P.1/aws). For the problems (P.2/Xws) assume that a row of F_p is of the form u^k for $|u| \geqq 2$, $k \geqq 2$. Then u is a palindrome because of symmetry of F_p, and it has 1 at the ends. If u contains zero then u^k contains groups without 0 that are twice longer inside than at the ends, which contradicts (6.8.2). Therefore u does not contain zero. Then $p = 2$ because for $p > 2$ the rows of F_p without zero do not contain the word 11. For $p = 2$ we can easily find out that $|u|$ is a power of 2; then (P.2/iws), (P.2/ows) really must be answered "yes". However, for $p > 2$ we have $k = 1$, hence "yes" is seldom for (P.2/ows) and impossible for (P.2/iws).

Theorem 6.10 gives S in Table 6.1 for all eight problems (P.1/XZw); only the trivial cases $w \in 0^+$, $u \in 0^+$ must be considered separately. The problems (P.1/osw), (P.1/isw) have the answer "yes" whenever $w \notin 0^+$. The problems (P.3/Xws) can be reduced to (P.1/Xws), hence they are solvable. The problems (P.3/Xww) must be answered NO for similar reason as (P.3/Xws).

For the problems (P.2/XYw) imagine that the y-th row of $G = \mathrm{GPT}(\mathcal{M}_p, w)$, $w \notin 0^+$ belongs to u^*, and construct the zone of G between its $(y - p^n + 1)$-th and the y-th row (including these rows), where n is the least integer such that $|w| \leqq p^n$. Let the zone be constructed from the left to the right, at each stage a new occurence of u into the y-th row be added, and let the zone up to the appropriate right diagonal of G be filled in.

A necessary condition for the right margin of $\mathcal{A}$ just being constructed is that all rows have the same final subword of the form $z0^m$, $z \neq 0$ as the words u, w have. (This also gives a necessary condition for u, w which can be checked at first. Similarly, u, w must have the same initial subword of the form $0^m z$, $z \neq 0$.) Another necessary condition is that a row of the zone consists only of multiples of w which are divided by the groups of $p^n - |w|$ zeros. If, finally, the appropriate sequence of

```
                0 1 0 1 0 1 1 . .
               0 1 1 1 1 1 2 1 . .
              0 1 2 2 2 2 0 0 2 . .
             0 1 0 1 1 1 2 0 2 1 . .
            0 1 1 1 2 2 0 2 2 0 1 . .
           0 1 2 2 0 1 2 2 1 2 1 0 . .
          0 1 0 1 2 1 0 1 0 0 0 1 0 . .
         0 1 1 1 0 0 1 1 1 0 0 1 1 1 . .
        0 1 2 2 1 0 1 2 2 1 0 1 2 2 1 . .
```

Figure 6.1

coefficients belongs to $R(\mathcal{M}_p)$ then the last row of (the just constructed part of) the zone does belong to $R(\mathcal{M}_p, w)$. Since the process of construction is ultimately periodic, we can decide whether u^k belongs to $R(\mathcal{A}, w)$ for at least one k, infinitely many k, etc. (Sometimes the construction of the zone can be interrupted much sooner than the ultimate periodicity is found. It is so also in Figure 6.1 if e.g. $w = 01221$.) Therefore the problems (P.2/XYw) are algorithmically solvable.

Table 6.2 summarizes the obtained information about generalized Pascal triangles modulo p. Its system is the same as that of *Table 3.2.*

7. Small algebras and other special cases

Several classes of algebras will be given for which all problems (P.k/XYZ) from Theorem 3.2 are decidable. The proof will be either direct applications of the previous sections or at least it will be based on methods given there. Particularly, Corollary 5.8 will be often used.

7.1. Theorem. All the problems (5.8.1) are algorithmically solvable for the class of all finite algebras $\mathcal{A} = (\mathrm{A}; \mathrm{K}, \mathrm{l}, \mathrm{r}, .)$ such that there is a partial ordering $\leqq$ of the set A with the property

$$x \cdot y \geqq x \text{ and } x \cdot y \geqq y \qquad (7.1.2)$$

for all $x, y \in \mathrm{A}$.

Proof. Let a finite algebra $\mathcal{A}$ satisfy (7.1.2) and let $w \in \mathrm{A}^+$. Denote $G = \mathrm{GPT}(\mathcal{A}, w)$.

Let (m, n) belong to the interior of the domain of G, $c = G(m, n)$ and let c be a maximal element of the set of all values $G(x, y)$ for (x, y) for the interior of the domain of G. Then

$$G(m+1, n) = G(m, n) \cdot G(m+1, n-1) \geqq G(m, n) = c$$

and hence $G(m+1, n)=c$. Analogously $G(m, n+1)=c$ and hence by induction

$$G(x, y)=c \quad \text{for all} \quad x \geqq m, y \geqq n .$$

Now we can use Corollary 5.8 (the appropriate embedding vectors are e.g. $(1, 0)$, $(0, 1)$).

7.2. Theorem. All the problems (5.8.1) are algorithmically solvable for the class of all finite algebras $\mathscr{A}=(\mathrm{A}; \mathrm{K}, \mathrm{l}, \mathrm{r}, .)$ in which the binary operation is associative and idempotent, i.e.

$$(x \cdot y) \cdot z = x \cdot (y \cdot z) \quad \text{and} \quad x \cdot x = x \tag{7.2.1}$$

for all $x, y, z \in \mathrm{A}$.

Proof. We shall again use Corollary 5.8. Consider a finite algebra which satisfies (7.2.1) and a word $w \in \mathrm{A}^+$. We may delete an upper margin (of a suitable width) of $G=\mathrm{GPT}(\mathscr{A}, w)$, and so arrange that $|w|>1$ and that the left and the right margin are periodical. Let its periods be p, q, respectively. We shall prove that $(0, p)$ and $(q, 0)$ are embedding vectors for a suitable restriction of G.

Order the elements of the margin M of G from the left to the right and adjoin to every pair (x, y) from the domain $\mathrm{D}_{0, |w|-1}$ of G a segment of M as follows. If (x, y) belongs to the margin of the domain of G then the one-element sequence consisting of $G(x, y)$ is adjoined to (x, y). Otherwise a segment of M which is over the y-th right diagonal and the x-th left diagonal is adjoined to (x, y). For example, if $|w|=3$ then the finite sequence

$$G(0, 1), \; G(0, 0), \; G(1, -1), \; G(2, -2)$$

is adjoined to the pair (2,1) (see Figure 2.1).

Then $G(x, y)$ is the product (in the sense of "." of $\mathscr{A}$) of the segment adjoined to (x, y). We shall prove this statement by induction. For (x, y) from the margin of $\mathrm{D}_{0, |w|-1}$ it is obvious. Otherwise let a finite sequence

$$a_0, a_1, \ldots, a_{n-1}, a_n \tag{7.2.2}$$

be adjoined to (x, y). Then the finite sequences

$$a_0, a_1, \ldots, a_{n-1} \qquad a_1, \ldots, a_{n-1}, a_n$$

are adjoined to $(x-1, y)$, $(x, y-1)$, respectively. Then by (7.2.1) we have

$$\begin{aligned} G(x, y) = G(x-1, y) \cdot G(x, y-1) &= (a_0 \cdot a_1 \cdot \ldots \cdot a_{n-1}) \cdot (a_1 \cdot \ldots \cdot a_{n-1} \cdot a_n) = \\ &= a_0 \cdot (a_1 \cdot \ldots \cdot a_{n-1})^2 \cdot a_n = a_0 \cdot a_1 \cdot \ldots \cdot a_{n-1} \cdot a_n . \end{aligned}$$

Now we can prove that

$$G(x, y+p) = G(x, y) \quad \text{and} \quad G(x+q, y) = G(x, y)$$

for all (x, y) below the p-th right diagonal and the $(q+|w|)$-th left diagonal. Let

(x, y) be such pair, and (7.2.2) be adjoined to it. Then at least p elements of (7.2.2) belong to the left margin of G, and hence

$$a_0,\ a_1,\ \ldots,\ a_{p-1},\ a_0,\ a_1,\ \ldots,\ a_{p-1},\ a_p,\ \ldots,\ a_{n-1},\ a_n$$

is adjoined to $(x, y+p)$. Therefore by (7.2.1)

$$\begin{aligned} G(x, y+p) &= (a_0 \cdot a_1 \cdot \ldots \cdot a_{p-1})^2 \cdot a_p \cdot \ldots \cdot a_{n-1} \cdot a_n = \\ &= (a_0 \cdot a_1 \cdot \ldots \cdot a_{p-1}) \cdot a_p \cdot \ldots \cdot a_{n-1} \cdot a_n = G(x, y)\ . \end{aligned}$$

The other formula can be proved similarly but we can use Corollary 5.8 also without it.

7.3. Theorem. All the problems (5.8.1) are algorithmically solvable for the class of all finite algebras $\mathscr{A} = (\mathrm{A};\ \mathrm{K}, \mathrm{l}, \mathrm{r}, .)$ which satisfy one of the identities:

$$(x \cdot y) \cdot (y \cdot z) = x \tag{7.3.1}$$

$$(x \cdot y) \cdot (y \cdot z) = y \tag{7.3.2}$$

$$(x \cdot y) \cdot (y \cdot z) = z \tag{7.3.3}$$

$$((x \cdot y) \cdot (y \cdot z)) \cdot ((y \cdot z) \cdot (z \cdot t)) = y \cdot z \tag{7.3.4}$$

Proof. We shall use Corollary 5.8 in every case. The appropriate embedding vectors will be (2, 0), (1, 1), (0, 2) and (1, 1), respectively, and the restriction of $\mathrm{G} = \mathrm{GPT}(\mathscr{A}, w)$ to the set $\{(a, b) \in \mathrm{N} \times \mathrm{N};\ a \geqq 2,\ b \geqq 2\}$ will be embedded into itself in all cases.

If (7.3.1) holds we have

$$\begin{aligned} G(a+2, b) &= G(a+1, b) \cdot G(a+2, b-1) = \\ &= (G(a, b) \cdot G(a+1, b-1)) \cdot (G(a+1, b-1) \cdot G(a+2, b-2)) = G(a, b)\ , \end{aligned}$$

and analogously $G(a+1, b+1) = G(a, b)$ for (7.3.2) and $G(a, b+2) = G(a, b)$ for (7.3.3). The computation for (7.3.4) is a little longer, and we shall do it only for $a = 4,\ b = 7$ to obtain shorter formulae.

$$\begin{aligned} G(4, 7) &= G(3, 7) \cdot G(4, 6) = \\ &= (G(2, 7) \cdot G(3, 6)) \cdot ((G(3, 6) \cdot G(4, 5)) = \\ &= ((G(1, 7) \cdot G(2, 6)) \cdot (G(2, 6) \cdot G(3, 5))) \cdot ((G(2, 6) \cdot G(3, 5)) \cdot (G(3, 5) \cdot \\ &\cdot G(4, 4))) = G(2, 6) \cdot G(3, 5) = G(3, 6)\ . \end{aligned}$$

Analogously, $G(a+1, b+1) = G(a, b)$ in general.

The role of the inequalities $a \geqq 2,\ b \geqq 2$ is that all necessary values of G are defined.

To avoid too frequent drowing of Cayley tables we shall determine a binary operation on a finite set A by a linear expression. This expression will be formed so that we fix the order of elements of A and then we shall write all rows of the Cayley table (except the head row). The whole expression will be given into brackets, and the rows will be separated by semicolons from each other. Analogously, a unary operation on A will be determined by its values for argument in the fixed order; they will also be given into brackets. For the sake of uniqueness the value of a nulary operation will also be written in brackets. The fixed order of elements of A will be given in the usual expression of a finite set.

For example, the algebra from Figure 2.2a can be expressed as

$$\mathcal{A} = (\{0, 1\}; (1), (11), (10), (01; 10)).$$

The canonically ordered operation on the set $\{\mathcal{S}, 0, 1\}$ which is displayed in Figure 2.2b can be expressed as $(\mathcal{S}11; 101; 110)$.

7.5. Theorem. For all algebras consisting of at most two elements, all the problems (5.8.1) are algorithmically solvable.

Proof. We may restrict ourselves to the algebras $\mathcal{A}$ which base set is $M_2 = \{0, 1\}$. (One-element algebras can be considered as subalgebras of suitable two-element ones.) We shall consider all sixteen binary operations on M_2.

Consider the operations (01; 10) (i.e. the sum modulo 2) and (10; 01) at first. Since 0, 1 can be interchanged it suffices to consider (01; 10). If GPT($\mathcal{A}$, w) ought to be considered we can arrange that the left and the right margins are periodical; it suffices to delete the first line for that. We also omit the trivial case $w \in 0^+$ (when e.g. Theorem 7.1 can be used). If the left margin is not constant we add a new left margin consisting of only 1; the same we do for the right margin. In this way GPT($\mathcal{A}$, w) is transformed into GPT($\mathcal{M}_2$, w') for some $w' \in M_2^+$. (Hence w' is one of the words v, $1v$, $v1$, $1v1$, *where* v is either the 0-th or the first row of the original GPT($\mathcal{A}$, w).)

The problems (5.8.1) for GPT($\mathcal{A}$, w) usually can be reduced to the appropriate problems for GPT($\mathcal{M}_2$, w'). (The reduction need not mean that the answers are equal.) Only the problems (P.2/Xww) are exceptional. They can be reduced to the problems

(P.2′/oww) find out whether for given u_1, u, u_r, $w' \in M_2^+$ at least one word from $u_1u^*u_r$ belongs to R($\mathcal{M}_2$, w'),

and the appropriately formed problems (P.2′/Xww), $X \in \{i, a, e\}$. (The variants with s instead of w are special cases of the above.) They can be solved in the same way which was used for (P.2/Xww) in the proof of Theorem 6.10.

The operations (10; 00) and (11; 10) satisfy the identity (7.3.4); it suffices to check that for one of them. Notice that already R($\mathcal{A}$) can be an SSL of degree 2 in this case. Of course, Corollary 5.8 can be directly used instead of Theorem 7.3.

For the remaining ten binary operations the sets R($\mathcal{A}$, w) are SSL languages of degree 1. Hence Corollary 5.8 can be used; the appropriate embedding vectors will always be (0, 2) and (2, 0), sometimes also (0, 1) and (1, 0). However, Theorem 7.1 (e.g. for (00; 00), (00; 01)), Theorem 7.2 (e.g. for (00; 01); (00; 11), (01; 01)) and Theorem 7.3 (e.g. (7.3.3) for (10; 10)) also can be used.

8. Concluding remarks

Up to now we have ordered generalized Pascal triangles only to the algebras of signature (0, 1, 1, 2). Now we shall use the notation GPT($\mathcal{A}$, w), GPT($\mathcal{A}$, w, i) also for the algebras of signatures (0, 2), (2) and the notation GPT($\mathcal{A}$) for the algebras of signature (0, 2). The missing unary operations will be replaced by the identical mapping of A into A. The more general case which is considered in the previous sections can be reduced to these special cases by the following theorem.

8.1. Theorem. Let $\mathcal{A}=(A; K, l, r, .)$ be an algebra of signature (0, 1, 1, 2), $\mathcal{S}$a, $B=A\cup\{\mathcal{S}\}$, let $*$ be the operation canonically ordered to $\mathcal{A}$ by the formula (2.3.1) and let $\circ$ be the operation on the set B defined by

$$x\circ y=\begin{cases}K \text{ if } x=\mathcal{S},\ y=\mathcal{S}\\ x*y \text{ otherwise.}\end{cases} \tag{8.1.1}$$

Let $\mathcal{B}_1=(B; K, \circ)$, $\mathcal{B}_2=(B; *)$ be algebras of signatures (0, 2) and (2), respectively. Then:

1) GPT($\mathcal{A}$) can be embedded into GPT($\mathcal{B}_1$) with the embedding vector (1, 1).

2) For every word $w\in A^+$ the function GPT($\mathcal{A}$, w) can be embedded into GPT($\mathcal{B}_2$, $\mathcal{S}w\mathcal{S}$) with the embedding vector $(1, -1)$.

The proof is easy and will be omitted. As an illustrating example consider the algebra $\mathcal{A}$ from Figure 2.2a. The corresponding GPT($\mathcal{B}_1$), GPT($\mathcal{B}_2$, w) are displayed in *Figure 8.1a, b.* Notice that

$$\mathrm{GPT}_{\mathcal{S}}(\mathcal{A}, w)=\mathrm{GPT}_{\mathcal{S}}(\mathcal{B}_2, \mathcal{S}w\mathcal{S}, -1) \tag{8.1.2}$$

but no similar formula holds for $\mathrm{GPT}_{\mathcal{S}}(\mathcal{A})$. Below we shall sometimes write GPT($\mathcal{A}$, w) instead of GPT($\mathcal{B}_2$, w) for arbitrary $w\in\mathcal{S}^*A^+\mathcal{S}^*$.

$$\begin{array}{ccccccccccc}
 & & & & & \mathcal{S} & & & & & \\
 & & & & \mathcal{S} & & \mathcal{S} & & & & \\
 & & & \mathcal{S} & & 1 & & \mathcal{S} & & & \\
 & & \mathcal{S} & & 1 & & 0 & & \mathcal{S} & & \\
 & \mathcal{S} & & 1 & & 1 & & 1 & & \mathcal{S} & \\
\mathcal{S} & & 1 & & 0 & & 0 & & 0 & & \mathcal{S}
\end{array}$$

a) GPT($\mathcal{B}_1$)

$$
\begin{array}{c}
\mathcal{S}\ 0\ 1\ 1\ \mathcal{S}\\
\mathcal{S}\ 1\ 1\ 0\ 0\ \mathcal{S}\\
\mathcal{S}\ 1\ 0\ 1\ 0\ 1\ \mathcal{S}\\
\mathcal{S}\ 1\ 1\ 1\ 1\ 1\ 0\ \mathcal{S}\\
\mathcal{S}\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ \mathcal{S}
\end{array}
$$

b) GPT($\mathcal{B}_2$, $\mathcal{S}011\mathcal{S}$)

$$
\begin{array}{c}
?\ \ ?\ \ F\ \ ?\ \ F\ \ F\ \ ?\ \ ?\\
?\ \ F\ \ ?\ \ ?\ \ F\ \ ?\ \ ?\\
F\ \ ?\ \ ?\ \ ?\ \ F\ \ F\\
?\ \ ?\ \ ?\ \ F\ \ ?\\
?\ \ ?\ \ F\ \ F\\
?\ \ ?\ \ ?\\
?\ \ ?\\
?
\end{array}
$$

c) The minimal triangle for F

Figure 8.1

Theorem 8.1 makes possible to consider GPT for the algebras of the signature (0, 2) or (2) which are, of course, simpler than that of signature (0, 1, 1, 2). However, the reduction enlarges the cardinality of algebras and hence it is disadvantageous for such results as Theorem 7.4. Therefore we have preferred to study a more general case.

For the problem (P.1/Xww) the occurrence of a word w in a GPT is understood as the occurrence in a row of this GPT. Similar problems could be considered also for columns and left and right diagonals. More generally, the following problem can be considered:

(P.1/ofw) For a given algebra $\mathcal{A}$, a word $w \in A^+$ and a mapping F of a finite subset of D into A find out whether the set $\{a+b;\ (a, b)$ is an embedding vector of F into GPT$(\mathcal{A}, w)\}$ is nonempty.

The problems (P.1/ifw), (P.1/afw), (P.1/efw) will be obtained if "is nonempty" is replaced by "is infinite", "contains almost all $n \in N$", and "contains all $n \in N$", respectively.

The problems (P.1/Xww) can be reduced to the problems (P.1/Xfw) so that the word $u = u_0 u_1 \ldots u_n$ is replaced by the function

$$F(x, y) = \begin{cases} u_x & \text{if } 0 \leqq x \leqq n \text{ and } y = -x \\ \text{undefined otherwise} \end{cases}$$

For $X \in \{o, i\}$ the opposite reduction is also possible. To explain it, we define:

8.2. Definition. Let $\mathscr{A}$ be an algebra of signature (0, 1, 1, 2) $w \in \mathcal{S}^* \mathrm{A}^+ \mathcal{S}^*$ and F be a mapping of a finite subset of D into A. We shall say that w is an $\mathscr{A}$-ceiling of F if F can be embedded into the triangle GPT($\mathscr{A}$, w) with the sides in the 0-th row, the 0-th right diagonal and the ($|w|-1$)-th left diagonal so that every of its sides contains at least one element corresponding to an element of F.

All $\mathscr{A}$-ceilings of F have the same length which depends only on its domain, and they can be determined as follows. At first we find the least triangle T with the horizontal side on the top which contains the domain of F. (See Figure 8.1c, where the symbols F show the domain of F.) Then we extend F to a mapping H of T into $\mathrm{A} \cup \{\mathcal{S}\}$ so that a word $w \in \mathcal{S}^* \mathrm{A}^+ \mathcal{S}^*$ arises in the top side and it holds

$$H(x, y) = H(x-1, y) \circ H(x, y-1) \tag{8.2.1}$$

whenever both sides are defined. All words which can be written into the top side of T are $\mathscr{A}$-ceilings of F. (The condition (8.2.1) may cause that no $\mathscr{A}$-ceiling of F exists.) Now we can reduce (P.1/Xfw) to (P.1/Xww), $X \in \{o, i\}$ by the following theorem.

8.3. Theorem. For every algebra $\mathscr{A}$, every word $w \in \mathrm{A}^+$ and every mapping F of a finite subset of D into A the following conditions are equivalent:

(i) The set of all embedding vectors of F into GPT($\mathscr{A}$, w) is nonempty (resp. infinite).

(ii) At least one $\mathscr{A}$-ceiling of F occurs in at least one row (resp. infinitely many rows) of GPT($\mathscr{A}$, $\mathcal{S}^n w \mathcal{S}^n$) where n is the length of the $\mathscr{A}$-ceilings of F.

Hence the problems (P.1/ofw), (P.1/ifw) are solvable e.g. for the class of GPT described in Theorem 4.10. In this case also the problems (P.1/afw), (P.1/efw) are solvable. For the proof we can use that the set

$$\{n \in \mathrm{N};\ u \text{ occurs in the } n\text{-th row of GPT}(\mathscr{A}, w)\}$$

is semilinear provided R($\mathscr{A}$), w) is an SSL language. The mentioned problems are solvable also for all two-element algebras.

REFERENCES

[1] Čulík, K. II.—Gruska, J.—Salomaa, A.: Systolic trellis automata (for VLSI). Research report CS-81-34, December 1981, University of Waterloo, Canada.

[2] Čulík, K. II.—Gruska, J.—Salomaa, A.: Systolic trellis automata: Stability, decidability, and complexity. Research report CS-82-04, January 1982, University of Waterloo, Canada.

[3] Gruska, J.: Systolic automata, in Parallel and distributed systems. (In Slovak). Research report, May 1983, Computing Research Center Bratislava, Czechoslovakia.

[4] Ginsburg, S.: The mathematical theory of context-free languages. McGraw Hill Book Company, 1966.

[5] Hopcroft, J. E.—Ullman, J. D.: Formal languages and their relation to automata. Addison-Wesley Publishing Company, 1969.

[6] Ibarra, O. H.: A note on semilinear sets and bounded-reversal multihead pushdown automata. Technical Report 74-8, May 1974, University of Minnesota, U.S.A.

[7] Ibarra, O. H.—Kim, S. M.—Moran, S.: Trellis automata: Characterizations, speed-up, hierarchy, decision problems. Preprint 1983, University of Minnesota, U.S.A.

Received: 8. 2. 1984

Author's address:
Ivan Korec
Katedra algebry a teórie čísel MFF UK
Mlynská dolina
842 15 Bratislava

РЕЗЮМЕ

ОБОБЩЕННЫЕ ТРЕУГОЛЬНИКИ ПАСКАЛЯ РАЗРЕШИМЫЕ ПРОБЛЕМЫ

Иван Корец, Братислава

В статье определены обобщенные треугольники Паскаля. Они возникают аналогично обыкновенному треугольнику Паскаля, только вместо суммы на множестве N всех натуральных чисел используется любая бинарная операция на некотором множестве A. Более того, элементы левой и правой стороны определяются при помощи двух одноместных операций на A, и в начальной строке находится любая конечная последовательность элементов множества A. Проблема находится-ли данный элемент $x \in A$ (по меньшей мере раз или же бесконечно много раз) в данном обобщенном треугольнике Паскаля G, алгоритмически неразрешима даже если A – конечное множество. В статье исследуются некоторые классы обобщенных треугольников Паскаля, для которых аналогичные проблемы алгоритмически разрешимы. Это имеет место например тогда, когда множество строк G является полулинейным языком или когда бинарная операция есть сумма по простому модулю p, и одноместные операции тривиальны.

SÚHRN

ZOVŠEOBECNENÉ PASCALOVE TROJUHOLNÍKY ROZHODNUTEĽNÉ PROBLÉMY

Ivan Korec, Bratislava

Definujú sa zovšeobecnené Pascalove trojuholníky; tie sa vytvárajú obdobne ako obyčajný Pascalov trojuholník, ale namiesto sčítania (na množine N) sa používa ľubovoľná binárna operácia na nejakej množine A. Naviac, ľavý a pravý okraj nemusia byť konštantné, ale sa vytvárajú pomocou dvoch

unárnych operácií na množine A, a do horného riadku sa zapisuje ľubovoľná konečná postupnosť prvkov A. Aj pre konečnú množinu A sú niektoré jednoduché otázky algoritmicky neriešiteľné. Napríklad je algoritmicky nerozhodnuteľné, či sa v danom zovšeobecnenom Pascalovom trojuholníku G (určenom horným riadkom a operáciami pre tvorbu okrajov a vnútra) nachádza daný prvok $x \in$ A aspoň raz, resp. nekonečne mnoho krát. V tejto práci sa ukazujú špeciálne prípady, keď sú obdobné problémy algoritmicky riešiteľné. Je to tak napríklad vtedy, keď množina všetkých riadkov G je semilineárny jazyk, alebo keď je ľavý i pravý okraj G konštantný a binárna operácia pre vnútro je sčítanie modulo prvočíslo p. Uvedené otázky sú algoritmicky riešiteľné tiež vtedy, keď A je najviac dvojprvková množina.