

stichting
mathematisch
centrum



AFDELING INFORMATICA

IW 18/74

APRIL

P.M.B. VITÁNYI
ON THE SIZE OF DOL LANGUAGES

Prepublication

2e boerhaavestraat 49 amsterdam

BIBLIOTHEEK MATHEMATISCH CENTRUM
AMSTERDAM

Printed at the Mathematical Centre, 49, 2e Boerhaavestraat, Amsterdam.

The Mathematical Centre, founded the 11-th of February 1946, is a non-profit institution aiming at the promotion of pure mathematics and its applications. It is sponsored by the Netherlands Government through the Netherlands Organization for the Advancement of Pure Research (Z.W.O), by the Municipality of Amsterdam, by the University of Amsterdam, by the Free University at Amsterdam, and by industries.

AMS (MOS) subject classification scheme (1970): 68A30, 94A30
ACM - Computing Reviews - category: 5.22, 5.23

ON THE SIZE OF DOL LANGUAGES *)

PAUL M.B. VITÁNYI

ABSTRACT

Languages generated by monogenic (i.e. deterministic) context independent Lindenmayer systems (DOL systems) are investigated. Necessary and sufficient conditions are established under which the language generated by a DOL system is finite. Thus, sharp bounds on the cardinality of such a language are obtained. A feasible solution for the membership problem is given. The problems are solved of what is the minimum sized alphabet over which there is a DOL language of cardinality n and, conversely, what is the maximum sized finite DOL language over an alphabet of m letters. This in turn provides us with some number theoretic functions, interesting in their own right, of which several properties, interrelations and asymptotic approximations are derived.

1. INTRODUCTION

Lindenmayer systems are a class of parallel rewriting systems introduced by Lindenmayer [59,60] as a model for the developmental growth of filamentous organisms. A Lindenmayer system consists of an initial filament, symbolized by a string of letters, and the subsequent stages of development are obtained by rewriting all letters in a string simultaneously at each time step. It is called deterministic if the system is essentially monogenic, i.e. each string has a unique successor. It is called context independent if the rewriting of a letter does not depend on its neighbors.

The study of Lindenmayer systems and the languages they generate has gone a long way since its original biological motivation. It has found its own place within the

*) To appear in *Topics in L systems*, G. Rozenberg & A. Salomaa (eds.), Lecture Notes in Computer Science, Springer Verlag, Berlin etc., 1974.

body of formal language theory by the growing interest in parallel processes and the different notions, problems and techniques particular to this field, see e.g. Herman and Rozenberg [45]. For instance, the notion of generating languages by monogenic re-writing systems is altogether foreign to the usual generating grammar approach since there such a language would either be empty or contain one element. It is in this direction that our present investigations take place. We shall be concerned with deterministic context independent Lindenmayer systems and the languages they generate. This family of languages has been studied in detail, e.g. with respect to its place in the Chomsky hierarchy [91,117], (anti)closure properties [91,102], and the growth of word length [111,15,75,98,116]. The membership problem for DOL languages has been solved affirmatively in [14] where a gigantic upper bound on the size of such a language is given in case it is finite.

The present paper consists of two parts. In the first part we establish, by a simple combinatorial argument, necessary and sufficient conditions (with respect to the production rules) under which the language generated by a deterministic context independent Lindenmayer system is finite. These conditions yield sharp bounds on the size of such a language depending on the size of the alphabet and the production rules. Furthermore, a feasible decision procedure for the membership problem is provided and we solve the problems of what is the minimum sized alphabet over which there is a deterministic context independent Lindenmayer language of size n and, conversely, what is the maximum sized finite deterministic context independent Lindenmayer language over an alphabet of m letters. The solutions to these last two problems provide us with some number theoretic functions, interesting in their own right, which form the object of study in the second part of our paper. We derive several properties, interrelations and asymptotic approximations to these functions.

2. FINITE DETERMINISTIC CONTEXT INDEPENDENT LINDENMAYER LANGUAGES

We assume that the standard terminology of formal language theory is familiar. We customarily use, with or without indices, i, j, k, n, m, p, q to range over the set of natural numbers $\mathbb{N} = \{0, 1, 2, \dots\}$; a, b, c, d to range over an alphabet W ; v, w to range over W^* , i.e. the set of all words over W including the *empty word* λ . A *deterministic context independent Lindenmayer system (DOL system)* is a triple $G = \langle W, \delta, w \rangle$ where W is a finite non-empty *alphabet*, δ is a total mapping from W into W^* called the *set of production rules*, and $w \in WW^*$ is the *axiom*. We extend δ to W^* by defining $\delta(\lambda) = \lambda$ and $\delta(a_1 a_2 \dots a_n) = \delta(a_1) \delta(a_2) \dots \delta(a_n)$. (I.e. δ is a homomorphism on W^* .) δ^i is the composition of i copies of δ and is inductively defined by $\delta^0(v) = v$ and $\delta^i(v) = \delta(\delta^{i-1}(v))$ for $i > 0$. The *DOL language* generated by G is $L(G) = \{\delta^i(w) \mid i \geq 0\}$. A letter $a \in W$ is *mortal* ($a \in M$) iff $\delta^i(a) = \lambda$ for some i ; *vital* ($a \in V$) iff $a \notin M$; *recursive* ($a \in R$) iff $\delta^i(a) \in W^* \{a\} W^*$ for some $i > 0$; *monorecursive* ($a \in MR$) iff $\delta^i(a) \in$

$\in M^*\{a\}M^*$ for some $i > 0$. Clearly, if $a \in M, R, MR$ then there is an i as above such that $i \leq \#M, \#R, \#MR$, respectively, where $\#Z$ denotes the cardinality of a set Z .

Lemma 1. Let $G = \langle W, \delta, w \rangle$ be a DOL system. If there is an i and a $b \in R-MR$ such that b is a subword of $\delta^i(w)$ then $L(G)$ is infinite.

Proof. If $b \in R-MR$ then there is a $j \leq \#R$ and a $c \in V$ such that $\delta^j(b) = v_1 b v_2 c v_3$ or $\delta^j(b) = v_1 c v_2 b v_3$. Hence, if $lg_v(v)$ denotes the number of occurrences of vital letters in a word v , we have

$$(1) \quad lg_v(\delta^{i+nj}(w)) \geq lg_v(\delta^{nj}(b)) > n,$$

and $L(G)$ is infinite. \square

Lemma 2. Let $G = \langle W, \delta, w \rangle$ be a DOL system. If there is an $i \geq \#(V-R)$ and a $b \in V-R$ such that b is a subword of $\delta^i(a)$ for some $a \in W$ then there is a $j < i$ and a $c \in R-MR$ such that c is a subword of $\delta^j(a)$.

Proof. There is a sequence of letters a_0, a_1, \dots, a_i such that $a_0 = a$, $a_i = b$ and a_{j+1} is a subword of $\delta(a_j)$ for $0 \leq j < i$. If $b \in V-R$ then $a_j \in V$ for $0 \leq j \leq i$. Since there are at least $\#(V-R)+1$ a_j 's there is one which is recursive and therefore there is a $j_1 < i$ such that $a_{j_1} \in R$. It is easy to see that for a recursive letter d always holds that $\delta^t(d)$ contains a recursive letter as a subword for each t . Therefore, $\delta^{i-j_1}(a_{j_1}) = v_1 d v_2 b v_3$ or $\delta^{i-j_1}(a_{j_1}) = v_1 b v_2 d v_3$ where $d \in R$ and $b \in V-R$. Hence $a_{j_1} \in R-MR$. By taking c equal to a_{j_1} the lemma is proved. \square

Lemma 3. Let $G = \langle W, \delta, w \rangle$ be a DOL system. If $\delta^t(w) \in (MuMR)^*$ for $t = \#(V-R)$ then $L(G)$ is finite.

Proof. Suppose

$$(2) \quad \delta^{\#(V-R)}(w) = v_1 a_1 v_2 a_2 \dots v_n a_n v_{n+1},$$

where $a_1, a_2, \dots, a_n \in MR$ and $v_1, v_2, \dots, v_{n+1} \in M^*$. Now it is easy to see that for each $a_i \in MR$ there is a k_i ($1 \leq k_i \leq \#MR$) and a sequence $a_{i0}, a_{i1}, \dots, a_{ik_i}$ such that $a_{i0} = a_{ik_i} = a_i$, $a_{ij_1} \neq a_{ij_2}$ for $0 \leq j_1 < j_2 < k_i$, and $a_{ij+1} \in MR$ is the only vital letter in $\delta(a_{ij})$, $0 \leq j < k_i$. Also,

$$(3) \quad \delta^{\#M}(b) = \lambda \quad \text{for all } b \in M.$$

Hence, for all $a_i \in MR$ and all $t, t' \geq \#M$ holds

$$(4a) \quad \delta^t(a_i) = \delta^{t'}(a_i) \quad \text{for } t \equiv t' \pmod{k_i},$$

$$(4b) \quad \delta^t(a_i) \neq v_1 \delta^{t'}(a_i) v_2 \quad \text{for } t \not\equiv t' \pmod{k_i}, \quad \text{for all } v_1, v_2 \in W^*.$$

By (2), (3) and (4) we have that for all $t \geq \#(W-R)$ holds:

$$(5) \quad \delta^t(w) = \alpha_{1j_1} \alpha_{2j_2} \cdots \alpha_{nj_n},$$

where $\alpha_{ij_i} = \delta^{j_i}(a_i)$, $j_i \equiv t \pmod{k_i}$ and $\#M \leq j_i < \#M+k_i$, $1 \leq i \leq n$. By (2), (4) and (5):

$$(6a) \quad \delta^t(w) \neq \delta^{t'}(w) \quad \text{for all } t, t' \text{ such that} \\ \#(W-R) \leq t < t' < \#(W-R) + \text{l.c.m.}(k_1, k_2, \dots, k_n);$$

$$(6b) \quad \delta^t(w) = \delta^{t'}(w) \quad \text{for all } t, t' \text{ such that } t, t' \geq \#(W-R) \text{ and} \\ t \equiv t' \pmod{\text{l.c.m.}(k_1, k_2, \dots, k_n)}.$$

Therefore

$$(7) \quad \text{l.c.m.}(k_1, k_2, \dots, k_n) \leq \#L(G) \leq \text{l.c.m.}(k_1, k_2, \dots, k_n) + \#(W-R). \quad \square$$

We are now ready to state the main theorem of this section.

Theorem 1. Let $G = \langle W, \delta, w \rangle$ be a DOL system. $L(G)$ is finite iff $\delta^t(w) \in (MUMR)^*$ for $t = \#(V-R)$.

Proof. "If". By lemma 3.

"Only if".

Case 1. $\delta^t(w) \in W^*(R-MR)W^*$. By lemma 1 $L(G)$ is infinite.

Case 2. $\delta^t(w) \in W^*(V-R)W^*$ for $t = \#(V-R)$. By lemma 2 there is a $t' < t$ such that $\delta^{t'}(w) \in W^*(R-MR)W^*$, and therefore case 1 holds and $L(G)$ is infinite.

Hence, if $\delta^t(w) \in W^*(V-MR)W^*$ for $t = \#(V-R)$ then $L(G)$ is infinite, i.e. if $L(G)$ is finite then $\delta^t(w) \in (MUMR)^*$ for $t = \#(V-R)$. \square

From the previous lemmas and the theorem we can derive some interesting corollaries.

Corollary 1. $L(G)$ is finite iff $\delta^t(w) \in (MUMR)^*$ for all $t \geq \#(V-R)$.

Corollary 2. A DOL language is finite iff all recursive letters which are accessible from the axiom (i.e. which occur in words in the language) are monorecursive.

Since all letters which can be derived from a certain letter (or word) are derived within $\#W$ steps, it is easy to determine whether a letter is mortal, vital, recursive, monorecursive. The quickest way is to determine subsequently M , V , R and MR .

Corollary 3. There is an algorithm to determine whether the language generated by a DOL system is finite or not. (Hint: determine M,V,R and MR and apply theorem 1 or corollary 2.)

Next we consider the *membership problem*: given a DOL system $G = \langle W, \delta, w \rangle$ and a word $v \in W^*$, decide whether or not v is in $L(G)$. (Equivalently, is there an i such that $\delta^i(w) = v$). Now assume that $L(G)$ is finite and

$$\delta^{\#(W-R)}(w) = v_1 a_1 v_2 a_2 \dots v_n a_n v_{n+1},$$

where $a_1, a_2, \dots, a_n \in MR$ and $v_1, v_2, \dots, v_{n+1} \in M^*$. Assume further that $v = \alpha_{1j_1} \alpha_{2j_2} \dots \alpha_{nj_n}$ where $\alpha_{ij_i} = \delta^{j_i}(a_i)$ for some j_i such that $\#M \leq j_i < \#M+k_i$, $1 \leq i \leq n$. By

(4b) $\delta^{j_i}(a_i) \neq v_1 \delta^{j_i'}(a_i) v_2$ for all j_i, j_i' such that $\#M \leq j_i < j_i' < \#M+k_i$ and all $v_1, v_2 \in W^*$, $1 \leq i \leq n$. Therefore, the parse of v (if it exists) is unique and can be executed easily from left to right given $\delta^t(a_i)$ for all t and i , $\#M \leq t < \#M+k_i$, $1 \leq i \leq n$. Since by (4a) $\delta^t(a_i) = \delta^{t'}(a_i)$ for all $t, t' \geq \#M$ such that $t \equiv t' \pmod{k_i}$ the problem can now be restated as follows: is there a positive integer u such that $u \equiv (j_i - \#M) \pmod{k_i}$, $1 \leq i \leq n$. The solution is well known.

Lemma 4. (Chinese remainder theorem¹). Let k_1, k_2, \dots, k_n be positive integers and let t_1, t_2, \dots, t_n be any integers. There is exactly one integer u which satisfies the conditions

$$\begin{aligned} 0 \leq u < \text{l.c.m.}(k_1, k_2, \dots, k_n), \\ u \equiv t_i \pmod{k_i} \quad (1 \leq i \leq n) \\ \text{iff } t_i \equiv t_j \pmod{\text{g.c.d.}(k_i, k_j)} \quad (1 \leq i < j \leq n). \end{aligned}$$

There is no integer $u \equiv t_i \pmod{k_i}$, ($1 \leq i \leq n$), if not $t_i \equiv t_j \pmod{\text{g.c.d.}(k_i, k_j)}$, ($1 \leq i < j \leq n$).

Therefore, if u exists then $v = \delta^{\#(W-R)+u}(w)$ and $v \neq \delta^t(w)$ for all $t \geq \#(W-R)$ otherwise. If a parse of v as mentioned is not possible then by (5) $v \neq \delta^t(w)$ for all $t \geq \#(W-R)$. Hence we have

Theorem 2. There is an algorithm which solves the membership problem for DOL languages.

Proof. The proof consists in giving an outline of the algorithm.

(i) Determine whether $L(G)$ is finite or not (corollary 3). If $L(G)$ is infinite then generate successively $w, \delta(w), \delta^2(w), \dots$ and compare each $\delta^i(w)$ with v . Is

¹ See e.g. Knuth, D. *Seminumerical algorithms*. Addison-Wesley, Reading (Mass.) (1969), 256.

$\delta^i(w) \neq v$ for all $i < t_0$ and $\delta^{t_0}(w)$ contains more occurrences of vital letters than does v then $v \notin L(G)$. By (1) $t_0 \leq \#V(l_{\mathcal{G}_V}(v) - l_{\mathcal{G}_V}(w) + 1)$.

(ii) $L(G)$ is finite. Generate successively $w, \delta(w), \dots, \delta^{\#(W-R)}(w)$ and compare each $\delta^i(w)$ with v . Is $\delta^i(w) \neq v$ for all i such that $0 \leq i \leq \#(W-R)$ then try to parse v as discussed above. Is the parse successful then apply the Chinese remainder theorem. Depending on whether or not an integer u , as stated in the theorem, exists v does or does not belong to $L(G)$. If the parse is not successful then $v \notin L(G)$. \square

The decision procedure for the membership problem for DOL languages we gave above is unusual under mathematical decision procedures in that it is *feasible*, i.e. gives answers to reasonable questions within a reasonable time², as testified by an ALGOL 60 implementation, Vitányi [114]. Of course, if $L(G)$ is finite we can test for membership by generating the whole of $L(G)$. But as will appear from the next corollary and the asymptotic approximations in section 4, even for a modest alphabet of, say, a hundred letters, this may turn out not to be feasible.

By the inequality (7) we can easily determine the cardinality of a finite DOL language.

Example. Let $G = \langle \{a, a_1, a_2, a_3, b_1, b_2, c_1, c_2, c_3\}, \{\delta(a) = c_1 a_1 b_1 c_3, \delta(a_1) = c_1 a_2, \delta(a_2) = c_2 a_3, \delta(a_3) = c_3 c_3 a_1, \delta(b_1) = c_1 b_2 c_1, \delta(b_2) = c_2 b_1 c_1, \delta(c_1) = c_2 c_3, \delta(c_2) = c_3, \delta(c_3) = \lambda\}, c_1 a c_3 a_1 b_2 c_2 \rangle$.

Then: $M = \{c_1, c_2, c_3\}$,
 $V = \{a, a_1, a_2, a_3, b_1, b_2\}$,
 $R = MR = \{a_1, a_2, a_3, b_1, b_2\}$.

Since a does not occur in a value of δ , $\delta^{\#(V-R)}(c_1 a c_3 a_1 b_2 c_2) \in (MUMR)^*$: $L(G)$ is finite. The different *periods* k_1, k_2 are 2 and 3. Therefore, by (7)

$$\text{l.c.m.}(2,3) \leq \#L(G) \leq \text{l.c.m.}(2,3) + 9-5,$$

or,

$$6 \leq \#L(G) \leq 10.$$

By writing out $L(G)$ we see that $\#L(G) = 10$.

From the inequality (7) we obtain the following corollary (see also [114]) which forms the basis of the sequel.

² See e.g. Parikh, R. *Existence and feasibility in arithmetic*, J.Symb.Logic. 34 (1971), 494 - 508.

Corollary 4. (i) Let $P: \mathbb{N} \rightarrow \mathbb{N}$ be defined as follows. $P(m)$ is the largest natural number n which is the least common multiple of k_1, k_2, \dots, k_q , for all possible partitions of m into $q = 1, 2, \dots, m$ positive integral summands, plus the number of summands equal to 1. By (7) $P(m)$ is the maximum cardinality of a finite DOL language over an alphabet of m letters.

(ii) Let $S: \mathbb{N} \rightarrow \mathbb{N}$ be defined as follows. $S(n)$ is the smallest natural number m such that there exists a partition of m into positive integral summands k_1, k_2, \dots, k_q , $q \leq m$, and $\text{l.c.m.}(k_1, k_2, \dots, k_q) + \#\{i | k_i = 1\} = n$. By (7) $S(n)$ is the minimum cardinality of an alphabet over which there is a DOL language of cardinality n .

The remainder of the paper will be concerned with the investigation of the number theoretic functions S , P and some variants. Thus we derive lower bounds on the size of the alphabet as a function S of the size of a finite DOL language over such an alphabet, and upper bounds on the size of a finite DOL language as a function P of the size of the alphabet.

3. FUNCTIONS WHICH RELATE SIZE OF LANGUAGE WITH SIZE OF ALPHABET

The number theoretic functions S and P of corollary 4 have a much broader setting than just their connection with DOL systems. Imagine a process which starts by counting until some number d and then initializes some number q of periodic counters. Then $S(n)$ and $P(m)$ have a natural interpretation as the smallest number of states needed to generate a prescribed number n of distinguishable configurations and the largest number of distinguishable configurations which can be generated by using a prescribed number m different states, respectively. If we have the additional restriction $d = 0$ then, in the latter case, we ask in effect for the maximum order of a permutation of the m -th degree. (The order of a permutation of the m -th degree is the exponent of the smallest power of a permutation on m elements which is equal to the identity permutation.) Already Landau³ investigated the maximum order $f(m)$ of a permutation of a given degree m . I.e. $f: \mathbb{N} \rightarrow \mathbb{N}$ where $f(m)$ is defined as the maximum of the least common multiple of k_1, k_2, \dots, k_q for all possible partitions of m into $q = 1, 2, \dots, m$ positive integral summands. We shall return to this connection with Landau's work in section 4.

According to corollary 4,

$$(8) \quad S(n) = \min \left\{ \sum_{i=1}^q k_i + d \mid \text{l.c.m.}(k_1, k_2, \dots, k_q) + d = n \right\},$$

³ Landau, E. *Über die Maximalordnung der Permutationen gegebenen Grades*, Archiv der Math. und Phys., Dritte Reihe, 5 (1903), 92 - 103.

$$(9) \quad P(n) = \max\{\text{l.c.m.}(k_1, k_2, \dots, k_q) + d \mid \sum_{i=1}^q k_i + d = n\}.$$

For the smallest values of n we find:

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14
S(n)	1	2	3	4	5	5	6	7	8	7	8	7	8	9
P(n)	1	2	3	4	6	7	12	15	20	30	31	60	61	84

For instance,

$$S(14) = 2+7 = 4+3+2 = 9 \quad \text{since } 14 = 2*7 = 4*3 + 2.$$

$$P(14) = 2*2*3*7 = 4*3*7 = 84 \quad \text{since } 14 = 2+2+3+7 = 4+3+7.$$

Hence, the corresponding representations of S(n) and P(n) in k_1, k_2, \dots, k_q, d are not unique. Clearly, in (8) and (9) the $\bar{k}_1, \bar{k}_2, \dots, \bar{k}_q$ for which the extrema are reached for a given n will be relatively prime. Suppose we can factorize a \bar{k}_i , $1 \leq i \leq q$, into two relatively prime factors \bar{k}_{i1} and \bar{k}_{i2} :

$$\bar{k}_i = \bar{k}_{i1} * \bar{k}_{i2}, \quad \bar{k}_{i1} > 1, \quad \bar{k}_{i2} > 1.$$

Then

$$\bar{k}_i - (\bar{k}_{i1} + \bar{k}_{i2}) = \bar{k}_{i1} * \bar{k}_{i2} - (\bar{k}_{i1} + \bar{k}_{i2}) = (\bar{k}_{i1} - 1)(\bar{k}_{i2} - 1) - 1 \geq 0.$$

Therefore, it suffices to look for $\bar{k}_1, \bar{k}_2, \dots, \bar{k}_q$ which are powers of distinct primes.

Hence we replace (8) and (9) by

$$(10) \quad S(n) = \min\{\Sigma p^\alpha + d \mid \Pi p^\alpha + d = n\},$$

$$(11) \quad P(n) = \max\{\Pi p^\alpha + d \mid \Sigma p^\alpha + d = n\},$$

where p denotes some prime. To obtain a canonical representation for S(n) and P(n) we take the representation with the smallest d for which the extrema are reached. By the unique factorization property of the natural numbers this representation will be unique. Additionally we define

$$(12) \quad S'(n) = \min\{\Sigma p^\alpha + d \mid \Pi p^\alpha + d \geq n\},$$

$$(13) \quad P'(n) = \max\{\Pi p^\alpha + d \mid \Sigma p^\alpha + d \leq n\}.$$

(Then $S^f(n)$ is the number of letters in the smallest alphabet over which there is a

finite DOL language of at least cardinality n and $P'(n)$ is the cardinality of the largest finite DOL language over an alphabet of at most n letters.) It is convenient to introduce also

$$(14) \quad s(n,d) = \sum p^\alpha + d \quad \text{such that } \prod p^\alpha = n-d,$$

since by the unique factorization property $s(n,d)$ is found immediately; and we see that

$$(15) \quad S(n) = \min\{s(n,d) \mid 0 \leq d \leq n\}.$$

The first 2000 values of $S(n)$ were determined by computer and showed a quite erratic behavior. E.g. $S(1971) = 61$, $S(1972) = 50$, $S(1973) = 51$ and $S(2000) = 39$. (Østerby⁴ contains a detailed computer analysis of $S(n)$ for $1 \leq n \leq 5 \cdot 10^{11}$. Furthermore, $S'(n)$ and $P(n)$ are computed for a large number of values. He considers e.g. the question in how many different ways $S(n)$ can be obtained from n .)

Now let us take a closer look at the general behavior and interrelations of our functions. It is at once apparent that, since $P(n+1) \geq P(n)+1$ for all n , P is strictly increasing and therefore $P' = P$. $S(n+1) \leq S(n)+1$ and $S(8) = S(10) = 7$ while $S(9) = 8$. Therefore, S is not monotonic. By its definition S' is monotonic increasing and $S'(n) \leq S(n)$ for all n . A crude approximation gives us (for $n > 1$):

$$(16a) \quad P(n) < n^n;$$

$$(16b) \quad S(n)^{S(n)} > n;$$

$$(16c) \quad S'(n)^{S'(n)} > n.$$

From (16b) and (16c) it follows that $S(n) \rightarrow \infty$ and $S'(n) \rightarrow \infty$ for $n \rightarrow \infty$. In section 4 we shall derive asymptotic approximations for P , S' and $\inf S$; it will appear that these functions are intimately related to the distribution of the prime numbers. We use the notation $f(x) \sim g(x)$ for $f(x)$ is asymptotic to $g(x)$, i.e. $\lim_{x \rightarrow \infty} f(x)/g(x) = 1$. It is well known⁵ that the number of primes $\pi(x)$ not exceeding x is asymptotic to $x/\log x$: $\pi(x) \sim x/\log x$. Furthermore, the i -th prime p_i is asymptotic to $i \log i$: $p_i \sim i \log i$. It then follows from (16a) that $e^{\log P(n)} \leq e^{n \log n}$ and therefore $\log P(n) \leq n \log n \sim p_n$. Since $S'(n)^{S'(n)} \geq n$, similarly $\log n \leq S'(n) \log S'(n)$.

⁴ Østerby, O. *Prime decompositions with minimum sums*. Univ. of Aarhus, Comp. Sci. Dept. Tech. Rept. DAIMI-PB 19 (1973).

⁵ Hardy, G.H. & Wright, E.M. *An introduction to the theory of numbers*, Oxford University Press (1945), 9 - 10.

By noting⁶ that $x/\log x$ is asymptotic to the function inverse of $x \log x$ we have that $S'(n) \geq g(n)$ for some function $g(n) \sim \frac{\log n}{\log \log n} \sim \pi(\log n)$. Therefore, $S(n) \geq g(n)$ also.

Since P is strictly increasing and $P(6) = 7, P(7) = 12; P: \mathbb{N} \rightarrow \mathbb{N}$ is an injection but no surjection; since $S(n+1) \leq S(n)+1$ and $S'(n+1) \leq S'(n)+1$ for all $n, S(n) \rightarrow \infty$ and $S'(n) \rightarrow \infty$ for $n \rightarrow \infty, S(5) = S'(5) = S(6) = S'(6) = 5; S, S': \mathbb{N} \rightarrow \mathbb{N}$ are surjections but no injections. From the definitions we would expect S and S' to be some kind of an inverse of P . Since P is the maximum size language over an alphabet of n letters, and since P is strictly increasing, an alphabet of size n is the minimum size alphabet over which there is a language of (at least) size $P(n)$. Therefore, if we denote the set of values of P by $A = \{P(i) \mid i \geq 0\}$ we obtain $S(P(n)) = S'(P(n)) = n$ for all $n \in \mathbb{N}$. Hence the restrictions of S and S' to A are the inverse of P :

$$(17) \quad S|_A = S'|_A = P^{-1}.$$

From the definitions we also see that between two consecutive values of P, S' is constant (S' is monotonic, $S'(P(n)) = n$ for all $n, S'(P(n)+1) = n+1$ for all n) and therefore:

$$(18) \quad S'(m) = P^{-1}(n) \quad \text{for all } m, \quad P(P^{-1}(n)-1) < m \leq n,$$

where $n \in A$. Since $S'(n) \leq S(n)$ for all n we have therefore by (17)

$$(19) \quad S(n) = S'(n) = P^{-1}(n) \quad \text{and} \quad S(m) \geq P^{-1}(n),$$

for all $n \in A$ and all $m > P(P^{-1}(n)-1)$.

Therefore, S' is a stepfunction where every step of 1 takes place at a value of P . Furthermore, S' is the greatest monotonic increasing function which is a lower bound on S .

In looking at the function S and trying to distinguish its features we readily notice that if n is a prime or the power of a prime then $S(n) = S(n-1)+1$. The way S is defined, however, does not give us a general method, to find the value of S for a certain argument, better than by trial and error. The following theorem is one of the main results of this section and provides an inductive definition of S .

Theorem 3.

$$S(n) = \begin{cases} n & \text{for } n = 0, 1, 2, 3, 4, 5. \\ \min\{S(n-1)+1, s(n, 0)\} & \text{for } n > 5. \end{cases}$$

⁶ Hardy & Wright. Op. cit. 9-10.

Proof. By induction on n . The theorem holds for $n = 0, 1, 2, 3, 4, 5$. Suppose the theorem is true for all $n \leq m$. Since

$$S(m+1) = \min\{s(m+1, d) \mid 0 \leq d \leq m+1\},$$

and

$$s(m'+1, d') = s(m', d'-1) + 1,$$

for all m' and all d' such that $0 < d' \leq m'+1$, we have

$$S(m+1) = \min\{S(m)+1, s(m+1, 0)\}. \square$$

The following corollary of theorem 3 is also stated by Østerby⁷ and gives a recursive definition of $S(n)$. By theorem 3 we have for all n :

$$S(n) = \min\{s(n, 0), s(n-1, 0)+1, \dots, s(1, 0)+n-1, n\}.$$

Since for all k such that $n \geq k > S(n)$ holds $S(n) < s(n-k, 0)+k$, we have:

Corollary 5. $S(n) = \min\{s(n, 0), s(n-1, 0)+1, \dots, s(n-S(n), 0)+S(n)\}.$

Hence we only have to compute $s(n, d)$, i.e. the sum of the highest powers of primes in the factorization of $n-d$, for $d = 0, 1, \dots, k_0$ where k_0 is the minimum of the previously computed values of $s(n, d)+d$.

The analogue of theorem 3 for P is

$$P(n) = \begin{cases} n & \text{for } n = 0, 1, 2, 3, 4 \\ \max\{P(n-1)+1, \max\{m \mid s(m, 0) = n\}\} & \text{for } n > 4. \end{cases}$$

This does not help us very much, essentially because although the factorization of a natural number is unique, a partition is not. If we assume that the following conjecture by Landau⁸ is true, viz. $P(\sum_{i=1}^k p_i) = \prod_{i=1}^k p_i$ for all k , then since P is strictly increasing we can slightly limit the number of m 's which have to be investigated:

$$P(n) = \begin{cases} n & \text{for } n = 0, 1, 2, 3, 4 \\ \max\{P(n-1)+1, \max\{m \mid s(m, 0) = n \text{ and } \prod_{i=1}^k p_i < m \leq \prod_{i=1}^{k+1} p_i\}\} & \text{for } \sum_{i=1}^k p_i < n \leq \sum_{i=1}^{k+1} p_i, \end{cases}$$

⁷ Østerby, Op. cit.

⁸ Landau, Op. cit.

where we denote by p_i the i -th prime and $p_1 = 2$.

4. ASYMPTOTIC APPROXIMATIONS

In this section we investigate the asymptotic behavior of our functions. Landau⁹ proves that for $f(n) = \max\{\prod p^\alpha \mid \sum p^\alpha \leq n\}$:

$$(20) \quad \log f(n) \sim \sqrt{n \log n}.$$

Theorem 4. $\log P(n) \sim \sqrt{n \log n}$.

Proof. By (20) $\log f(n) \sim \sqrt{n \log n}$, i.e.

$$\lim_{n \rightarrow \infty} \frac{\log f(n)}{\sqrt{n \log n}} = 1.$$

Also,

$$\lim_{n \rightarrow \infty} \frac{\log (f(n)+n)}{\sqrt{n \log n}} = 1 + \lim_{n \rightarrow \infty} \frac{\log (1+n/f(n))}{\sqrt{n \log n}} = 1.$$

Since by (11) and the definition of $f(n)$ we have:

$$f(n) \leq P(n) < f(n)+n, \text{ i.e., } \log f(n) \leq \log P(n) < \log (f(n)+n),$$

and we proved above that

$$\log f(n) \sim \log (f(n)+n) \sim \sqrt{n \log n},$$

we have

$$\log P(n) \sim \sqrt{n \log n}. \quad \square$$

Corollary 6. $\log P(n) \sim \sqrt{p_n}$, where p_n is the n -th prime.

Theorem 5. $S'(n) \sim \frac{\log^2 n}{\log \log^2 n}$.

Proof. If $\log y = \sqrt{x \log x}$, then $\log^2 y = x \log x$ and

$$\log \log^2 y = \log x + \log \log x \sim \log x.$$

⁹ Landau, Op. cit.

Since

$$x = \frac{\log^2 y}{\log x} \text{ we have } x \sim \frac{\log^2 y}{\log \log^2 y}.$$

By this argument and since $\log P(m) \sim \sqrt{m \log m}$ it follows:

$$m \sim \frac{\log^2 P(m)}{\log \log^2 P(m)}$$

or

$$P^{-1}(n) \sim \frac{\log^2 n}{\log \log^2 n} \text{ for } n \in \{P(i) \mid i \geq 0\}.$$

Denote $\log^2 n / \log \log^2 n$ by $h(n)$. By (18) $S'(n) \sim h(n)$ for n in the range of P . This cannot tell us anything about the sup $S'(n)$ since the restriction to special values of n can only yield a lower bound but not an upper bound. According to (18), however, we have for all pairs of consecutive values of P , say n_1, n_2 :

$$S'(n_1) \leq S'(m) \leq S'(n_2) = S'(n_1) + 1, \quad n_1 \leq m \leq n_2.$$

Since h is strictly increasing,

$$\begin{aligned} \lim_{m \rightarrow \infty} S'(m)/h(m) &\geq \lim_{m \rightarrow \infty} S'(m)/h(n_2) \\ &\geq \lim_{m \rightarrow \infty} (S'(n_2) - 1)/h(n_2) \\ &= \lim_{n_2 \rightarrow \infty} (S'(n_2)/h(n_2) - 1/h(n_2)) \\ &= 1 - \lim_{n_2 \rightarrow \infty} 1/h(n_2) = 1. \end{aligned}$$

Analogous we prove that $\lim_{m \rightarrow \infty} S'(m)/h(m) \leq 1$, and therefore $S'(m) \sim h(m)$ for all $m \in \mathbb{N}$. \square

Corollary 7. $S'(n) \sim \pi(\log^2 n)$.

The greatest monotonic increasing function which is a lower bound on S is $S'(n) \sim h(n)$. Therefore:

Corollary 8. $\inf S(n) \sim \frac{\log^2 n}{\log \log^2 n}$.

Because of theorem 3 $\inf S(n) \sim \inf s(n,0)$ and we have:

Corollary 9. The greatest monotonic increasing function which is a lower bound on the sum of the greatest powers of primes in the factorization of n , i.e. $s(n,0)$, is asymptotic to $h(n)$. Hence:

$$\inf s(n,0) \sim \frac{\log^2 n}{\log \log^2 n}.$$

As is to be expected, this lower bound is reached for the special sequence of values $n = \prod_{i=1}^k p_i$, $k \in \mathbb{N}$.

Lemma 5. $\sum_{i=1}^k p_i \sim \frac{\log^2 n}{\log \log^2 n}$, where $n = \prod_{i=1}^k p_i$ and $k \in \mathbb{N}$.

Proof. The number of factors in a factorization of a natural number n is denoted by $\omega(n)$. According to Hardy & Wright¹⁰

$$\omega(n) \sim \frac{\log n}{\log \log n}.$$

Therefore, $\sum_{i=1}^k p_i \sim \sum_{i=1}^{\omega(n)} i \log i$. Bounding this discrete summation on both sides by an integral we obtain:

$$\int_1^{\omega(n)} i \log i \, di \leq \sum_{i=1}^{\omega(n)} i \log i \leq \int_2^{\omega(n)+1} i \log i \, di,$$

$$\frac{1}{2} \left[i^2 \log i - i^2/2 \right]_1^{\omega(n)} \leq \sum_{i=1}^{\omega(n)} i \log i \leq \frac{1}{2} \left[i^2 \log i - i^2/2 \right]_2^{\omega(n)+1},$$

$$\frac{1}{2} (\omega(n))^2 (\log \omega(n) - \frac{1}{2}) + \frac{1}{2} \leq \sum_{i=1}^{\omega(n)} i \log i \leq \frac{1}{2} ((\omega(n)+1)^2 (\log(\omega(n)+1) - \frac{1}{2}) - 4 \log 2 + 2).$$

Hence if $n \rightarrow \infty$ through this particular series of values we have

$$\begin{aligned} \sum_{i=1}^k p_i &\sim \frac{1}{2} (\omega(n))^2 \log \omega(n) - \omega(n)^2/2 \\ &\sim \frac{1}{2} \omega(n)^2 \log \omega(n) \\ &\sim \frac{\log^2(n) (\log \log n - \log \log \log n)}{2 (\log \log n)^2} \\ &\sim \frac{\log^2 n}{2 \log \log n} = \frac{\log^2 n}{\log \log^2 n}. \quad \square \end{aligned}$$

¹⁰ Hardy & Wright, Op. cit., 355.

A numerical verification shows:

$$\begin{aligned} (2+3+5)/(\log^2 (2*3*5)/\log \log^2 (2*3*5)) &\approx 0.47 \\ (2+3+\dots+17)/(\log^2 (2*3*\dots*17)/\log \log^2 (2*3*\dots*17)) &\approx 0.58 \\ (2+3+\dots+97)/(\log^2 (2*3*\dots*97)/\log \log^2 (2*3*\dots*97)) &\approx 0.75 \\ (2+3+\dots+173)/(\log^2 (2*3*\dots*173)/\log \log^2 (2*3*\dots*173)) &\approx 0.79. \end{aligned}$$

Resuming the results of this section we have:

$$\log P(n) \sim \sqrt{n \log n} \sim \sqrt{p_n} ;$$

$$S'(n) \sim \inf S(n) \sim \inf s(n,0) \sim \frac{\log^2 n}{\log \log^2 n} \sim \pi(\log^2 n);$$

and, furthermore,

$$s(n,0) \sim \frac{\log^2 n}{\log \log^2 n} ,$$

for $n \rightarrow \infty$ through the particular series of values $n = \prod_{i=1}^k p_i$.

Acknowledgement. I thank O. Østerby and D. Wood for critical comments.

REFERENCES

- [14] P. Doucet, *On the membership question in some Lindenmayer systems*, Indag. Math., 34 (1972) 45-52.
- [15] P. Doucet, *The growth of word length in DOL systems*, University of Aarhus, Computer Science Department, technical report no. 15, 1973, 83-94.
- [45] G.T. Herman & G. Rozenberg, *Developmental systems and languages*, to be published by North-Holland Publishing Company.
- [59] A. Lindenmayer, *Mathematical models for cellular interactions in development, Part I*, J. Theoret. Biol., 18 (1968) 280-299.
- [60] A. Lindenmayer, *Mathematical models for cellular interactions in development, Part II*, J. Theoret. Biol., 18 (1968) 300-315.
- [75] A. Paz & A. Salomaa, *Integral sequential word function and growth equivalence of Lindenmayer systems*, Information and Control, 23 (1973) 313-343.
- [91] G. Rozenberg & P. Doucet, *On OL-languages*, Information and Control, 19 (1971) 302-318.
- [98] A. Salomaa, *On exponential growth in Lindenmayer systems*, Indag. Math., 35 (1973) 23-30.
- [102] A. Salomaa, "*Lindenmayer systems: parallel rewriting without terminals*", Paragraph 13 in Part II of "*Formal languages*", Academic Press, May, 1973.
- [111] A.L. Szilard, *Growth functions of Lindenmayer systems*, University of Western Ontario, Computer Science Department, Technical Report no. 4, 1971.
- [114] P. Vitányi, *DOL-languages and a feasible solution for a word problem*, Technical report no. MR 138/72, Mathematisch Centrum, Amsterdam, 1972.
- [116] P. Vitányi, *Structure of growth in Lindenmayer systems*, Indag. Math., 35 (1973) 247-253.
- [117] P. Vitányi, *A note on nonrecursive and deterministic Lindenmayer languages*, Technical report no. IW 11/73, Mathematisch Centrum, Amsterdam, 1973.