ZING

SYSTEMS

1962

Edited By:

MARSHALL C. YOVITS, Office of Naval Research

GEORGE T. JACOBI, Armour Research Foundation

GORDON D. GOLDSTEIN, Office of Naval Research

SPARTAN BOOKS

6411 CHILLUM PLACE, N. W. • WASHINGTON 12, D. C.

ON PROBABILISTIC PUSH-DOWN STORAGES

M. P. Schützenberger
Department of Preventive Medicine
Harvard Medical School
Boston, Massachusetts

1. INTRODUCTION

The aim of this note is to describe certain elementary problems of random walks arising in the study of a restricted family of finite automata with a push-down storage.

For any set $X = \{x\}$ we shall denote by $F(X)$ the set of all finite strings of elements of X and we shall refer to $F(X)$ as the *free monoid generated by X* (the operation being, as usual, the concatenation).

If X is considered as the input alphabet of an automaton $\mathcal{S}$ consisting of sets of states $S = \{s\}$ with transition function $(SxX) \to S$, we shall use the following notations:

$$\text{if } f = x_{i_1} x_{i_2} \ldots x_{i_m} \in F(X) \text{ and } s \in S,$$

$$s.f = s_{i_m} \text{ is defined inductively}$$

$$\text{by } s_{i_o} = s,\ s_{i_1} = (s_{i_o}, x_{i_1}), \ldots,$$

$$(s_{i_j}, x_{i_{j+1}}) = s_{i_{j+1}}, \text{ and } s.f = s \text{ if } f = e \text{ (the empty word).}$$

$$\phi_S f = \phi_S f' \text{ if and only if for all } s \in S,\ s.f = s.f'.$$

By definition a finite automaton with push-down storage $\mathcal{a}$ is given by:

(1) a finite input alphabet X
(2) two finite sets of states S and T; two distinguished elements $s_o \in S$, $t_o \in T$; a distinguished subset $\bar{S}$ of S.
(3) an internal alphabet X; a word $g_o \in F(Z)$; a finite subset $\bar{G}$ of $F(Z)$.

(4) the following mappings:
$\sigma : (S,T,X) \to S$;
$\chi : (S,T,X) \to (T,T)$ (the family of all sets of pairs of elements of T);
$\alpha : (S,T,X) \to F(Z)$;
$(T,Z) \to T$.

The unbounded part of the memory of $\mathfrak{A}$ consists of a tape on which words in the alphabet Z can be written or erased.

If $f = x_{i_1} x_{i_2} \dots x_{i_n} \dots$ is an input sequence, the automaton starts in the initial state (s_o, t_o, g_o) where g_o means that the word g_o is already stored in the memory of $\mathfrak{A}$.

The letters $x_{i_1}, x_{i_2} \dots x_{i_n} \dots$ are read sequentially and, for each of them, the following cycle of operations is performed:

If the state of $\mathfrak{A}$ is $(s,t,g) \in (S,T,F(Z))$ and the incoming input letter is $x_{i_n} = x$:

(1) the state s is changed to $\sigma(s,t,x) = s'$;
(2) the machine searches if there exists a factorisation $g = g'g''$ of the stored word g such that $(t_o g', t_o g'') \in \chi(s,t,x)$.
If at least one such factorisation exists and if the one for which the degree (= length) of g'' is minimal is $g'.g''$, the word g'' is erased from the memory. If no such factorisation exists, nothing is erased and $g' = g$.
(3) the word $\alpha(s,t,x) = g''$ is written in the memory to the right of g'.
(4) the state t is changed to $t_o.g'g'''$ and the cycle is completed.

If after completing the cycle corresponding to the n-th letter of the input sequence $\bar{f} = x_{i_1} x_{i_2} \dots x_{i_n} \dots$ the state is s and the stored word is g, we shall write $f_n = x_{i_1} x_{i_2} \dots x_{i_n} \in F(X)$, $s = \sigma(f_n)$, $g = \alpha(f_n)$. By definition, $f_n \in \bar{K}$ if $(\sigma(f_n), \alpha(f_n)) \in (\bar{S}, \bar{G})$ and $f_n \in K$ if $f_n \in \bar{K}$, and for $n' < f_n' \notin \bar{K}$. In accordance with usual terminology, K may be called the set of the words accepted by $\mathfrak{A}$.

With respect to the probabilities we shall always assume that there exists a finite n, a fixed nxn matrix p and a representation μ of F(X) by nxn matrices such that (1) Tr p = 1; (2) for all $f \in F(X)$ $\Sigma \{\mathrm{Tr}\ p\mu fx : x \in X\} = \mathrm{Tr}\ p\mu f \geq 0$. Then, for each $f \in F(X)$, Tr $p\mu f$ can be interpreted as the probability measure of the set of all infinite input sequences which begin with f.

In particular, the case of n = 1 corresponds to the hypothesis that the letters of X are produced independently with constant probabilities $\bar{\mu}x = \mathrm{Tr}\ p\mu x$.

If there exists a one-to-one correspondence between the letters of X and the states of a Markov chain with initial probabilities $Pr(x_i)$ and transitions $Pr(x_i \mid x_j)$, one can take h equal to twice the number n' of elements of X and define for each $x_i \in X$ the matrix μx_i by

$$(\mu x_i)_{j,k} = Pr(x_i \mid x_j) \text{ if } k = i,\ 1 \leq i,\ j \leq n';$$

$$= Pr(x_i) \text{ if } k = i,\ j = n' + 1;$$

$$= 0 \text{ otherwise.}$$

2. EXAMPLES

Since these definitions are quite restrictive, it may be worthwhile to indicate how they relate to more familiar structures.

(1) Let $\alpha(s,t,x)$ be identically the empty word e. So, only the finite part $\mathcal{S}$ plays a role and $\mathfrak{A}$ is a conventional one way one tape automaton (Rabin and Scott). The family of the sets $\overline{K}$ corresponding to these automata is usually called the family $\mathfrak{R}$ of *regular events* (Kleene).

(2) Let Z consist of a single letter. The word $\alpha(f_n) = z^m$ stored in the memory can be identified with the non-negative integer m and $\mathfrak{A}$ can be considered as a finite automaton with a (unbounded) counter. Clearly, given an integer valued function β of X, it is possible to choose S, T, $(S,X) \rightarrow S$, etc., so that for each $f_n = x_{i_1} x_{i_2} \ldots x_{i_m}$:

(i) $\alpha(f_n) = z^{s_n}$ where s_n is the absolute value of the cumulative sum $\beta x_{i_1} + \beta x_{i_2} + \ldots + \beta x_{i_m} = \beta f_n$.

(ii) the sign of βf_n can be determined from the knowledge of $\sigma(f_n)$.

The associated probabilistic problem is that of the elementary discrete one dimensional random walk.

(3) Let $X = \{x_{\pm i}\}$, $Z = \{z_{\pm i}\}$ $(1 \leq i \leq N)$ and assume that the automaton $\mathfrak{A}$ is constructed so that if the stored word $g = \alpha(f_{n-1})$ ends with the letter z_i and if the n-th input letter is x_j then the cycle consists of the following single operations:

if $i = -j$, z_i is erased,

if $i \neq -j$, x_i is written to the right of g.

If g is the empty word z_i is written on the tape.

Assuming that the initial word g_o is the empty word, the word $\alpha(f_n)$ stored after reading the input word f_n can be considered as being obtained by replacing the x_i's by the z_i's and by erasing every pair of consecutive letters with opposite indices.

For instance if $\bar{f} = x_1 x_2 x_3 x_{-3} x_{-4} x_4 x_{-2} \ldots$, the values of $\alpha(f_1), \alpha(f_2), \ldots \alpha(f_7)$ are

$$x_1;\ x_1 x_2;\ x_1 x_2 x_3;\ x_1 x_2;\ x_1 x_2 x_{-4};\ x_1 x_2;\ x_1;\ \ldots$$

Clearly, if g_o is the empty word e, the set $D_N = \{f \in F(Z) : \alpha(f) = e\}$ depends only on N and it is in fact the kernel of the epimorphism γ of F(X) onto the free group generated by $\{x_i\}$ $(1 \leq i \leq N)$ that satisfies identically $1 = \gamma x_i \gamma x_{-1}$.

The theory of the associated random walk is due to Kesten.

3. CONTEXT FREE LANGUAGES

Let us consider after N. Chomsky two alphabets $\Xi = \{\xi\}$, $X = \{x\}$ and a finite collection G of pairs (ξ_i, f_i) where $\xi_i \in \Xi$ and where f_i belongs to the free monoid generated by the union of X and Ξ, but is not the empty word nor an element of Ξ. Taking an initial subset Ξ' of Ξ, we consider the least subset $L'_G \in F(X \cup \Xi)$ that satisfies the two conditions:

(1) $\Xi' \in L'_G$,

(2) if $f = f' \xi_i f'' \in L'_G$ and $(\xi_i, f_i) \in G$, then $f' f_i f'' \in L'_G$.

We denote by L_G the subset of L'_G which consists of the words containing only letter from X. L_G is the "context free language" generated by the grammar G. In a loose way, L'_G may be described as the set of all words which can be obtained starting from Ξ' by an arbitrary number of applications of the *rewriting rules* $\xi_i \rightarrow f_i$.

This formal construction is due to Post [12], but its special importance comes from its rediscovery by N. Chomsky who has founded upon it a general theory of natural languages.

As a matter of interest it must be mentioned that S. Ginsburg has recently observed that artificial programming languages like Algol are also context free languages. (Of course this remark should not be understood as implying that actual human languages are nothing more than glorified versions of Algol. Indeed, in Chomsky's theory the context-free level is only an initial germ out from which the true language emerges by the interplay of the higher structures ruled by the so called "transformations.")

As an example, let $\Xi = \{\xi\}$, $X = \{a,b\}$ and $G = \{(\xi, a), (\xi, b\xi\xi)\}$. The starting letter ξ can be replaced by a or by $b\xi\xi$. Since $a \in F(X)$, $a \in L$; in $b\xi\xi$, the first or the second ξ can be replaced by a or by $b\xi\xi$, giving the words $ba\xi$, $bb\xi\xi\xi$, $b\xi a$, baa, $bb\xi\xi a$, $b\xi b\xi\xi$, $bab\xi\xi$, $bb\xi\xi b\xi\xi$, ... on which the process has to be repeated, etc. Thus, one finds that a, baa, bbaaa, babaa, form the set of the words of L of degree of most 5. In fact a word f belongs to L if and only if the number of a's contained in it is equal to 1 plus the number of b's and if any proper left factor of f contains at most as many a's as b's. Thus, L corresponds to the set of the well formed formula in the so-called parenthesis free notation.

An other less obvious property is enjoyed by L: in some well defined sense every word is produced in an essentially unique manner. In other terms there is a unique manner of inserting brackets in any word $f \in L$ so that these brackets indicate how the word has been produced (for instance, (b(a(bba))).

This possibility does not necessarily exist for an arbitrary context free language. Hence for a given grammar G one may find it advisable to attach an integer n_f to every word $f \in F(X)$ in such a way that:

$n_f = 0$ if $f \notin L_G$ and, if $f \in L_G$. n_f = the number of essentially different ways in which f is produced.

Clearly, it is a desirable quality of an artificial language that $n_f \leq 1$ since, otherwise, there would be strings which would admit several possible interpretations. Unfortunately as shown by Bar-Hillel, Shamir and Perles the question to decide if $n_f \leq 1$ identically for an arbitrary context free language is recursively unsolvable.

Returning to the definitions introduced previously we can state the following property:

For any context free language L one can find:

a regular event K,
a natural number N,
a mapping $\phi : X \rightarrow F(X)$,
such that $L = \{\phi f : f \in K \cap D_N\}$

where, as usual ϕf denotes the word obtained when replacing in f every letter x_i by the (eventually empty) word ϕx_i.

Moreover, the construction implies that every word is produced the same number of times by both processes. In fact the number n_f defined rather loosely above can be more accurately defined as the number of words $f \in K \cap D_N$ such that $f' = \phi f$.

Reciprocally given any finite automaton with push-down storage one can find a set Ξ and a grammar $\underline{G}$ such that the corresponding language L is precisely equal to $\overline{K}$ (or to K). Moreover the construction shows that every word is produced at most once so that the correspondence between the two processes is really one-to-one. (Cf. Chomsky 1962 a)

4. GENERATING FUNCTIONS

The construction carried out in the last section can be given another interpretation which will be best explained in the case of the language L described above.

Let us assume that the successive letters are produced randomly according to the hypothesis of section 1 and introduce a new auxiliary variable t.

We can associate with L the usual generating function: $r' = \Sigma \{t^{|f|} \Pr f : f \in L\}$ where the notation $|f|$ represents the degree (length) of f and, according to hypothesis, the probability Pr f is equal to Tr μf.

However, for the sake of convenience, let us consider another generating function: $r = \Sigma \{t^{|f|} \mu f : f \in L\}$ from which r' can be deduced by the operation Tr. As it is well known r satisfies the algebraic matrix equation $r = t\mu a + t\mu b r^2$ which simply expresses that any word $f \in L$ which is not the word a has a unique factorisation $f = bf'f''$ where both f' and f'' belong to L.

Thus, in particular if dim $\mu = 1$ we obtain the classical formula

$$2r = 1 - (1 - 4t^2 \bar{\mu} a \bar{\mu} b)^{1/2}$$

in which the right member can be expanded in a Taylor series converging for $|t|$ small enough.

If dim $\mu = N > 1$, this straight-forward method is not possible because the equation is one in non-commutative variables. However, for given μa and μb the matrix μr can be computed from the system of n^2 algebraic equation which results from the identification of the entries in both members.

Now clearly the same remark is valid for the context free language produced by any grammar G. Let us consider the letters $(\xi_1, \ldots \xi_m)$ of Ξ as "unknown" and the letters $x \in X$ as (non commutative) coefficients. Then to the grammar G it corresponds in a one-to-one manner the system of equations

$$\xi_i = \Sigma \{f_i : (\xi_i, f_i) \in G\} \qquad (i = 1, 2, \ldots\ldots)$$

A simple discussion shows that because of our hypothesis on G such a system has always a formal solution $(r_1, r_2, \ldots\ldots, r_m)$ in which each of the components r_i is a formal power series in the variables of X (and coefficients in the semi-ring of non-negative integer). In fact, for any word f the coefficient of f in r_i is exactly n_f, the number of times f is produced by G when the initial letter is ξ_i.

Hence, if G is such that $n_f \leq 1$ identically, and if we define the representation $\bar{\mu}$ by the condition that for each $x \in X$, $\bar{\mu} x = t\mu x$, the image of r_i by $\bar{\mu}$ is precisely the desired generating function. As observed before this is the case for the sets K (or $\bar{K}$) defined by a finite automaton with a push-down storage.

For instance let us consider the set D_N introduced earlier. Clearly, D_N is the disjoint union of the sets $D_{N,i}$ $(-N \leq i \leq N)$ consisting of all words from D_N which begin with the letter x_i. Now is $f \in D_{N,i}$ the hypothesis that the first return to an empty memory occurs at the last letter of f implies that $f = x_i f' x_{-i}$ where f' is a word (possibly empty) of which the memory contains only x_i. Hence, introducing unknown ξ_i $(-N \leq i \leq N)$ in one-to-one correspondence with the letters x_i we have:

$$\xi_i = x_i (1 + \xi_{-i} - \Sigma \xi_i)^{-1} x_{-i}.$$

This can be brought to polynomial form by replacing each equation by the pair:

$$\xi_i = x_i x_{-i} + x_i \bar{\xi}'_i x_{-i}$$

$$\bar{\xi}'_i = \bar{\xi}_i \bar{\xi}'_i - \xi_i \text{ where } \bar{\xi}_i = \sum_{j \neq -i} \xi_{i_j} .$$

For the case of an arbitrary finite automaton with push-down storage, the equations are slightly more complicated but their obtention is a quite straightforward matter [18].

5. PROBABILITIES OF ABSORPTION

Given a finite automaton with push-down storage A, let us call probability of absorption π_A the probability measures of the set of all infinite input sequences which have at least one left factor belonging to K. From the remarks of the last section, it follows instantly that π_A is an *algebraic function* of the entries of the matrices μx. This result is essentially due to Kesten. It is worthwhile to contrast it with the fact that for finite automata (without unbounded memory) the corresponding probability is always a rational function of the entries of μ.

In the opposite direction, probabilities attached to a more general type of unbounded memory usually fail to be algebraic. For instance, let A and A' be two finite automata with counter (i.e., let the internal alphabets of A and of A; consist of a single letter). Simple computation shows that the generating function of the set $K \cap K'$ may have a logarithmic singularity, hence it may be an (elementary) transcendant function [17]. From an analytic point of view, this corresponds to the well known fact that the Hadamard product of two algebraic functions is not necessarily algebraic.

Geometrically it is one way of interpreting the essentially deeper character of the two dimensional random walk over the one dimensional absorption problems.

A more directly probabilistic implication of these remarks can be obtained when it is assumed that $\overline{S}$ (the distinguished final set of states) is $\{s_o\}$ (the distinguished initial state) and that, similarily, $\overline{G} = \{g_o\}$ = the empty word. Then, K defines the support of a regular event in Feller's terminology and the process is indeed a recurrent one if the letters x_i are produced independently with constant positive probabilities. Under this hypothesis the character persistent or transient of the recurrent event depends only on the analytic nature of the singularity of the generating function that is nearest to 0.

Thus, if A is a strictly finite automaton, the recurrent event, if persistent, has necessarily a finite mean recurrence time. If A uses in nontrivial fashion its unbounded memory, the event has always an infinite mean recurrence time and it can be persistent only if A is in fact an automaton with a counter and if the probabilities themselves satisfy a certain equation. Again these results go back to Kesten. However, the techniques described here allow generalisation from the case of a free group to that of an extension of a free group by a finite monoid.

ACKNOWLEDGEMENT

Acknowledgement is made to the Commonwealth Fund for the grant in support of the visiting professorship of biomathematics in the Department of Preventive Medicine at Harvard Medical School.

REFERENCES

1. Bar-Hillel, Y., Perles, M., and Shamir, E.: On Formal Properties of Simple Phase Structure Grammars. Techn. Report No. 4, Applied Logic Branch, The Hebrew University of Jerusalem, July 1960.
2. Birkeland, R.: Sur la Convergence des Developments qui Expriment les Racines de l'Equation Algebrique Generale. C. R. Acad. Sciences, *171:* 1370-1372, 1920; *172:* 309-311, 1921.
3. Chomsky, N.: A Note on Phrase Structure Grammars. Information and Control, *2:* 393-395, 1959.
4. Chomsky, N.: On Certain Formal Properties of Grammar. Information and Control, *2:* 137-167, 1959.
5. Chomsky, N.: Context Free Grammars and Push-Down Storage. Quarterly Progress Reports, No. 65, Research Laboratory of Electronics, M.I.T., 1962.

6. Ginsburg, S.: Some Remarks on Abstract Machines. Trans. Am. Math. Soc., *96*: 400-444, 1960.
7. Ginsburg, S., and Rose, G. F.: Operations which Preserve Definability in Languages. Technical Memorandum. System Development Corporation, Santa Monica (Calif.), 1961.
8. Ginsburg, S., and Rice, H. G.: Two Families of Languages Related to Algol. Technical Memorandum. System Development Corporation, Santa Monica (Calif.), 1961.
9. Kesten, M.: Symmetric Random Walks on Groups. Trans. Am. Math. Soc., *92:* 336-354, 1959.
10. Kleene, S. C.: Representation of Events in Nerve Nets and Finite Automata. Automata Studies, Princeton University Press, pp. 3-41, 1956.
11. Parikh, R. J.: Language Generating Devices. Quarterly Progress Report, No. 60, Research Laboratory of Electronics, M.I.T., pp. 199-212, January 1961.
12. Post, E.: A Variant of a Recursively Unsolvable Problem. Bull. Am. Math. Soc., *52*: 264-268, 1946.
13. Rabin, M. O., and Scott, D.: Finite Automata and Their Decision Problems. I.B.M. Journal of Research, *3:* 115-125, 1959.
14. Raney, G. N.: Functional Composition Patterns and Power-Series Reversion. Trans. Am. Math. Soc., *94:* 441-451, 1960.
15. Redei, L.: Die Verallgemeinerung der Schreierschen Erweiterungs Theorie. Act. Sci. Math., Szeyed *14:* 252-273, 1952.
16. Scheinberg, S.: Note on the Boolean Properties of Context Free Languages. Information and Control, *3:* 372-375, 1960.
17. Schützenberger, M. P.: Un Probleme de la Theorie des Automates. Seminaire Dubriel Pisot (Paris), December 1959.
18. Schützenberger, M. P.: On a Family of Formal Power Series. Submitted to Proc. Am. Math Soc.