LA-10492-MS

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36.

...thematics Explain Natural Intelligence?

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

Edited by Kyle Wheeler, Group C-2
Prepared by Yvonne Martinez, Group C-3

LA-10492-MS

UC-32
Issued: July 1985

Can Mathematics Explain Natural Intelligence?

Jan Mycielski*

*Consultant at Los Alamos. Professor, Department of Mathematics, University of Colorado, Boulder, CO 80309.

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

[redacted]

2. THE REAL BRAIN. I shall not talk about studies of the human brain except for the following few facts and contentions (for more information and references see [8, 12, 22]).

The neurons fire along their *axons* (output fibers) and send their signals to *dendrites* (input fibers), cell bodies, and axons of other neurons. The connections between different neurons are called *synapses*. The individual shots of a neuron are brief signals of the same magnitude (all-or-nothing rule). The frequency of those signals multiplied by the fraction of each of them passing through a synapse is believed to be the only information that a neuron transmits to its addressee. For a given neuron these frequencies may vary from a few to more than a thousand per second. This frequency depends on the messages received by the dendrites and the cell body, whereas the fractions of signals transmitted depends also on the messages received by the axon; different branches of the axon may receive messages from different neurons.

---

*This is the content of some lectures that I gave at Los Alamos in August 1984.

The firing of certain neurons into the dendrites or the cell bodies of a neuron N diminishes the rate of firing of N. Others excite their addressees. There are two types of neurons, the excitatory and inhibitory ones. As mentioned above, some inhibitory neurons fire into the branches of some axons diminishing the magnitude of individual shots delivered by those branches.

It is thought that the frequency of firing of a neuron N is a linear function (with a time lag) whose arguments are the sizes of individual shots times their frequencies, which are delivered to the dendrites and the body of N. The coefficients and constant terms of these functions can be positive or negative corresponding to excitatory or inhibitory synapses. Perhaps the input vectors are also normalized to some extent, so that the output frequencies remain within some fixed intervals. The absolute values of the coefficients may be determined by the sizes and other properties of the synapses. It is conjectured that these coefficients change in time and that the matrix of these coefficients constitutes the memory.

The tissues of the brain are repetitive in a sense. For example, the cerebral cortex is a large organ that has to be packed in the skull in a folded way. But the fibers of most neurons in the cortex are either perpendicular to the cortical layers or parallel to them. The dendrites of many neurons in the cortex of the cerebellum are organized in planes like the veins of a leaf, and these are stacked parallel to each other, whereas fibers of other types of neurons run through these stacks in the direction perpendicular to the leaves.

Other parts of the brain also display some regularities.

3. AN OPTIMISTIC REMARK. In principle we should be able to understand the mechanism of the brain. Unlike the fundamental problems of cosmology or of the structure of matter, where one explores a reality that appears to be infinite, here one has to explain only an engineering marvel created in finite time by natural selection. The regularities mentioned above suggest that there is only a moderate number of basic "ideas" that are repeated millions of times in the brain. If we understood them the problem would be solved, but it is still difficult to predict the complexity of a satisfactory solution.

Perhaps the point of view expressed in §1 and more mathematical imagination may yield conjectures amenable to experiments of physiologists and to computer simulation. Several authors share my optimism that some kind of mathematical models of the brain will prove important, see, for example, [7, 11, 13, 15, 17]. But we are not yet able to judge how important are the models presented in this report.

4. PREDICTORS THAT DO NOT LEARN. A well-developed branch of applied mathematics, "time series analysis," is essentially a study of predictors with a fixed algorithm. Let me recall the main idea.

If $f(x)$ is a polynomial of degree less than n, then

$$\Delta_h^n f(x) = 0 \ , \tag{4-1}$$

where, for any function g, $\Delta_h g(x) = g(x) - g(x-h)$, and $\Delta_h^{n+1} = \Delta_h \Delta_h^n$. Now, $\Delta_h^n g(x) = \sum_{k=0}^{n} (-1)^k \binom{n}{k} g(x - kh)$. Hence (4-1) suggests the following predictor. Given $g(x), g(x-h), \cdots, g(x-(n-1)h)$, we predict $g(x+h)$ by the formula

$$\hat{g}(x+h) = \sum_{k=0}^{n-1} (-1)^k \binom{n}{k+1} g(x-kh) \ .$$

This is exact, that is $\hat{g}(x+h) = g(x+h)$, if g is a polynomial of degree less than n.

It is possible that a mechanism for the computation of some simple cases of this formula, such as

$$\hat{g}(x+h) = 3g(x) - 3g(x-h) + g(x-2h) \ ,$$

which is exact for all quadratic or linear polynomials, exists in the brain.

5. LINEAR PREDICTORS THAT LEARN. I present here an algorithm A, which is folklore, but I have not been able to locate it (nor its main property Theorem 5.1) in the literature.

We assume that the input at time t is a nonzero vector $x_t \in H$, where H is a real (or complex) Hilbert space (finite dimensional in all applications that I know). The desired responses are real (or complex) scalars y_t, where $t = 0,1, \ldots$. The states of the memory of the predicting algorithm A are vectors $m_t \in H$. Upon receiving x_t, algorithm A predicts y_t according to the formula

$$\hat{y}_t = \langle m_t, x_t \rangle \tag{5-1}$$

(a conjugate linear functional). The error of this prediction is defined by the formula

$$e_t = \frac{y_t - \hat{y}_t}{\|x_t\|} \tag{5-2}$$

(a relative error). We assume that algorithm A learns e_t immediately after having predicted $\hat{y}_t$ and then updates its memory according to the formula

$$m_{t+1} = \begin{cases} m_t, \text{ if } |e_t| \leq \Theta \ , \\ m_t + e_t \left(1 - \dfrac{\Theta}{|e_t|}\right) \dfrac{x_t}{\|x_t\|}, \quad \text{if } |e_t| > \Theta \ . \end{cases} \tag{5-3}$$

Then m_0 is chosen arbitrarily in H, and Θ is a nonnegative constant called the *threshold of tolerated errors*.

The formula given in (5-3) can be explained in the following way:

$$m_{t+1} = \text{(the unique vector } m \text{ such that } |y_t - \langle m, x_t \rangle| / \|x_t\| \leq \Theta \qquad (5\text{-}3')$$
$$\text{and } \|m - m_t\| \text{ is minimal)} \ .$$

(If we wanted the linear predictor $\langle x_t, m_t \rangle$ instead of its conjugate, we would only have to substitute $\bar{e}_t$ for e_t in (5-3) or $\langle x_t, m \rangle$ for $\langle m, x_t \rangle$ in (5-3').)

The algorithm A is motivated by the following Theorem 5.1.

But we need first a notation:

$$E(m) = \sup_{t=0,1,\ldots} \frac{|y_t - \langle m, x_t \rangle|}{\|x_t\|} \ . \qquad (5\text{-}4)$$

Thus $E(m)$ is a measure of the predictive power of m.

5.1. Theorem. *For every $m \in H$ we have*

(i). *If $E(m) \leq \Theta$, then*

$$\sum_{|e_t| > \Theta} (|e_t| - \Theta)^2 \leq \|m - m_0\|^2 \ . \qquad (5\text{-}5)$$

(ii). *If $E(m) < \Theta$, then*

$$\sum_{|e_t| > \Theta} (|e_t| - \Theta) \leq \frac{\|m - m_0\|^2}{2(\Theta - E(m))} \ .$$

5.2. Problem. Let $E_0 = \inf_{m \in H} E(m)$. In all applications H is finite dimensional, and in this case there exists an m such that $E(m) = E_0$. But, for infinite dimensional H, the following question arises: *does* $\Theta = E_0$ *imply*

$$\lim_{t \to \infty} \sup |e_t| \leq \Theta \ ?$$

We know only that $\Theta = E_0$ does not imply (5-5). In fact, for $m_0 = 0$, $\Theta = 0$, $x_0, x_1, \ldots$ an orthonormal sequence, and $y_t = 1/\log(t+2)$, we have $E_0 = 0$, but $\sum |e_t|^\alpha = \infty$ for every real number α.

5.3. Example. We set $H = R^{\binom{p+q}{p}}$, where R is the real line, and

$$x_t = (x_{t1}^{k_1} \cdots x_{tp}^{k_p} : \ k_1 + \cdots + k_p \leq q) \ .$$

Then m_t is the vector of coefficients of a real polynomial

$$\hat{y}_t = \sum m_{tk_1 \cdots k_p} x_{t1}^{k_1} \cdots x_{tp}^{k_p} \ .$$

This is practical only if the dimension $\binom{p+q}{p}$ is not too large.

5.4. Example. We set $H = C^{2N+1}$, where C is the complex plane, and

$$x_t = (\xi_t^{-N}, \xi_t^{-N+1}, \ldots, 1, \ldots, \xi_t^N) \ ,$$

where $\xi_t \in C$ and $|\xi_t| = 1$ for $t = 0,1, \ldots$. Then m_t is the vector of coefficients of the trigeometric polynomial

$$\hat{y}_t = \sum_{-N}^{N} m_{tk} \xi_t^k \ .$$

(In this example the formula (5-3) is easier to compute than in Example 5.3 since here $\|x_t\| = \sqrt{2N+1}$ for all t.)

5.5. A generalization of the algorithm A. The above examples suggest a way of circumventing the assumption (made before (5-2)) that $x_t \neq 0$ for all t and extending the applicability of algorithm A. Namely, we can always replace H by $H \oplus C$ (or $H \oplus R$) and apply the map $x \mapsto (x,1)$, which omits (0,0). We may also use the unit sphere in $H \oplus C$, in lieu of the hyperplane $H \times \{1\}$, if we apply the stereographic projection

$$x \mapsto \frac{(4x,\ 4 - \|x\|^2)}{4 + \|x\|^2} \ ;$$

of course this gives applications different from those given by the map $x \mapsto (x,1)$. In general a preparatory map of the space of data into H may be needed, for example, the map $\xi \mapsto (\xi^{-N}, \xi^{-N+1}, \ldots, \xi^N)$ encountered in Example 5.4.

5.6. A hypothetical explanation of the cerebral cortex by means of the algorithm A. The axons bringing information into the cerebral cortex (the afferent axons) run parallel and are interspersed with the axons exporting the information from it (the efferent axons). Hence we can conjecture that the two messages are compared, and their difference causes learning. Of course the efferent signals (frequencies) depend upon the frequencies of a neighborhood of afferent axons of several time steps. The time steps would be measured by the α rhythm of the brain. When the brain thinks rather than watches, the efferent signals could be copied by the afferent neurons. The computation of the efferent frequencies and the modifications of the memory would be similar to those of the algorithm A.

5.7. On the range of applicability of the algorithm A. Even in the case when $E(m) = \infty$ for all $m \in H$, algorithm A may still be useful, for example, if $y_t = < m_t^*, x_t >$, where m_t^* drifts slowly enough in H.

5.8. The least squares algorithm. As it is pointed out in [21], from the point of view that is adopted here, the least squares algorithm is not better than algorithm A. In fact, if $\hat{y}_0 = \langle m_0, x_0 \rangle$ and $x_0 = m - m_0$, then $|e_0| = \|m - m_0\| \pm E(m)$, and so no estimates sharper than those of Theorem 5.1 are possible with the above formula for $\hat{y}_0$. Moreover, the computations for least squares are much more expensive than for the algorithm A.

Proof of Theorem 5.1. We put $x_t^\circ = x_t / \|x_t\|$, $s_t = \|m - m_t\|^2$, $\sigma_t = 1 - \Theta / |e_t|$, and $u_t = (y_t - \langle m, x_t \rangle) / \|x_t\|$. Then, by (5-4), we have

$$|u_t| \leq E(m) \ , \tag{5-6}$$

and, by (5-1) and (5-2),

$$\langle m - m_t, x_t^\circ \rangle = e_t - u_t \ . \tag{5-7}$$

Hence, if $|e_t| > \Theta$,

$$\begin{aligned}
s_{t+1} &= \|(m - m_t) - e_t \sigma_t x_t^\circ\|^2 \qquad \text{(by (5-3))} \\
&= s_t - \sigma_t e_t \langle x_t^\circ, m - m_t \rangle - \sigma_t \bar{e}_t \langle m - m_t, x_t^\circ \rangle + \sigma_t^2 |e_t|^2 \\
&= s_t - \sigma_t e_t (\bar{e}_t - \bar{u}_t) - \sigma_t \bar{e}_t (e_t - u_t) + \sigma_t^2 |e_t|^2 \qquad \text{(by (5-7))} \\
&= s_t - \sigma_t (2 - \sigma_t) |e_t|^2 + 2\sigma_t \mathrm{Re}(e_t \bar{u}_t) \\
&\leq s_t - \sigma_t (2 - \sigma_t) |e_t|^2 + 2\sigma_t |e_t| E(m) \qquad \text{(by (5-6))} \\
&= s_t - \sigma_t |e_t| ((2 - \sigma_t) |e_t| - 2E(m)) \ .
\end{aligned}$$

Therefore,

$$s_{t+1} \leq s_0 - \sum_{|e_k| > \Theta,\ 0 \leq k \leq t} \sigma_k |e_k| ((2 - \sigma_k) |e_k| - 2E(m)) \ .$$

Of course, $s_{t+1} \geq 0$ for all t; hence,

$$\sum_{|e_t| > \Theta} \sigma_t |e_t| \left((2 - \sigma_t) |e_t| - 2E(m)\right) \leq s_0 \ ,$$

which is equivalent to

$$\sum_{|e_t| > \Theta} (|e_t| - \Theta)(|e_t| + \Theta - 2E(m)) \leq \|m - m_0\|^2 \ .$$

Both parts of Theorem 5.1 follow immediately from this inequality.

6. THE PREDICTION OF VECTORS IN THE UNIT SPHERE. The theorem presented in this section was announced in [20].

Notice (compare also **5.6**) that the algorithms of §4 and §5 can be used for predicting vectors in R^n or C^n. Namely, one can apply those algorithms to each coordinate separately. In this section we introduce a more global procedure A^*, although its interest at present is rather theoretical.

The input vectors x_t and the vectors to be predicted y_t are on the unit sphere S^{n-1} in R^n, $t=0,1,\ldots$. The memory states of A^* are rotations M_t of S^{n-1}; that is, $M_t \in SO_n$. The algorithm A^* predicts according to the formula

$$\hat{y}_t = M_t x_t \quad . \tag{6-1}$$

(All vectors are treated as column vectors.) The error ρ_t of the prediction at time t is defined to be the angle between $\hat{y}_t$ and y_t (understood to be in the interval $[0, \pi]$); that is,

$$\rho_t = \arccos \langle \hat{y}_t, y_t \rangle \quad . \tag{6-2}$$

The algorithm A^* updates its memory states as follows:

$$M_0 = I = \text{the unit matrix} \quad ,$$

$$M_{t+1} = R_t M_t \quad , \tag{6-3}$$

where, in the case $\hat{y}_t \neq -y_t$, R_t is the *minimal rotation* such that

$$R_t \hat{y}_t = y_t \quad ;$$

that is, the rotation that does not move vectors orthogonal to both $\hat{y}_t$ and y_t, and, if $\hat{y}_t = -y_t$, then $R_t = I$.

[Of course R_t can be effectively computed in terms of $\hat{y}_t$ and y_t. Namely, treating $\hat{y}_t$ and y_t as column vectors, if $\hat{y}_t \neq -y_t$,

$$R_t = I + (y_t - \hat{y}_t)\hat{y}_t^T - (\hat{y}_t^{y_t} + y_t^{\hat{y}_t})(y_t^{\hat{y}_t})^T \quad , \tag{6-4}$$

where the superscript T denotes transposition and, for any $u, v \in S^{n-1}$, $u \neq \pm v$,

$$u^v = (u - \langle u,v \rangle v) / \| u - \langle u,v \rangle v \| \quad ,$$

and if $u = v$, then $u^v = 0$.]

The algorithm A^* is motivated by the following Theorem 6.1.

We assume that there exists a rotation N (unknown to A^*) *without the eigenvalue* -1 such that

$$y_t = Nx_t \text{ for } t = 0,1, \ldots . \tag{6-5}$$

6.1. Theorem. *Under the above assumptions* $\hat{y}_t \neq -y_t$ *for all* t, *and*

$$\Sigma_{t=0}^{\infty} \rho_t^2 < \infty . \tag{6-6}$$

6.2. Example. The algorithm A^* does not yield $\Sigma_0^\infty \rho_t^\alpha < \infty$ for any $\alpha < 2$. In fact, let N be a rotation of R^3 with rotation angle $\pi/2$. Then, for every sequence of positive reals ρ_t such that $\Sigma_0^\infty \rho_t^2 < \pi/4$, there exists a sequence of points $x_0, x_1, \ldots$ on S^2 such that, assuming (5-5), A^* yields precisely the errors ρ_t. Of course, if $\rho_t = c/(\sqrt{t+1}\log(t+2))$ with small enough c, then the above condition is satisfied, but $\Sigma \rho_t^\alpha = \infty$ for all $\alpha < 2$.

6.3. Problem. The assumptions (6-3) and (6-5) correspond to the case of Theorem 5.1 in which $\Theta = E(m) = 0$. Can one refine Theorem 6.1 in the style of Theorem 5.1?

Proof of Theorem 6.1. We need some notations and a lemma. For any $M \in SO_n$ we denote by $\measuredangle M$ the maximum angle φ in the interval $[0,\pi]$ such that $e^{i\varphi}$ is an eigenvalue of M. In other words

$$\cos(\measuredangle M) = \min_{x \in S^{n-1}} \langle Mx, x \rangle, \text{ and } \sin(\measuredangle M) \geq 0 .$$

In particular, for $\hat{y}_t \neq -y_t$, by the definition of R_t, $\measuredangle R_t = \rho_t$.

6.4. Observation. *If* Q *is a two-dimensional linear subspace of* R^n *and* $y \in R^n$, *then there exists a* $q_0 \in Q$, $q_0 \neq 0$, *which is orthogonal to* y.

Proof. If some $q \in Q$ is not orthogonal to y, then $\langle q, y \rangle$ and $\langle -q, y \rangle$ are of opposite signs. Hence, by continuity, there exists a $q_0 \in Q$ with $q_0 \neq 0$ and $\langle q_0, y \rangle = 0$.

6.5. Lemma. *Let* $R, B \in SO_n$, $y \in S^{n-1}$, *and* R *be the minimal rotation such that* $RBy = y$, *then*

$$\measuredangle(RB) \leq \measuredangle B .$$

Proof. If $By = y$, then $RB = B$; if $By = -y$, then $\measuredangle(B) = \pi$. In both cases the lemma is obvious, so let us assume that $By \neq y$ and $By \neq -y$. We have $\measuredangle(RB) = \measuredangle(BR)$ and, by the supposition, $BR(By) = By$. By the normal form theorem there exists a plane Q invariant under the rotation BR such that

$$\measuredangle(BR) = \arccos < q, BRq >$$

for every $q \in Q \bigcap S^{n-1}$. Of course, Q is orthogonal to By. By **6.4** we can choose $q_0 \in Q \bigcap S^{n-1}$ such that q_0 is also orthogonal to y. Then, since R is minimal, $Rq_0 = q_0$. So we have $\measuredangle(RB) = \measuredangle(BR) = \arccos < q_0, BRq_0 > = \arccos < q_0, Bq_0 > \leq \measuredangle B$.

Querry. Suppose $z, y \in S^{n-1}$, and $R \in SO_n$ are such that $Rz = y$ and

$$\forall B \in SO_n \; [By = z \rightarrow \measuredangle(RB) \leq \measuredangle B] \; .$$

Must R be minimal? (This question was raised by H. J. Keisler.*)

Returning to the proof of Theorem 6.1, we will study the sequence $B_t = M_t N^{-1}$. By (6-3) and (6-5) we have $B_0 = N^{-1}$, $B_{t+1} = R_t B_t$, and $R_t B_t y_t = y_t$. Thus B_0 does not have the eigenvalue -1 and, by Lemma 6.5, $\measuredangle B_t \leq \measuredangle N = \pi - \epsilon$ for some $\epsilon > 0$ and all t. Hence, by (6-1) and (6-5), $\hat{y}_t = B_t y_t \neq -y_t$ and the first part of Theorem 6.1 is proved.

For every t we choose coordinates in R^n in such a way that

$$y_t = \begin{pmatrix} 1 \\ 0 \\ 0 \\ . \\ . \\ . \\ 0 \end{pmatrix} \quad \textit{and} \quad B_t y_t = \begin{pmatrix} \cos \rho_t \\ \sin \rho_t \\ 0 \\ . \\ . \\ . \\ 0 \end{pmatrix} .$$

It follows that

$$R_t = \begin{pmatrix} \cos \rho_t & \sin \rho_t & 0 & \cdots & 0 \\ -\sin \rho_t & \cos \rho_t & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 1 \end{pmatrix} , \quad B_t = \begin{pmatrix} \cos \rho_t & -a_t \sin \rho_t & a_{13} & \cdots & a_{1n} \\ \sin \rho_t & a_t \cos \rho_t & a_{23} & \cdots & a_{2n} \\ 0 & a_{32} & a_{33} & \cdots & a_{3n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix} ,$$

where $|a_t| \leq 1$, and

*H. J. Keisler, University of Wisconsin, Madison, 1985.

$$B_{t+1} = R_t B_t = \begin{pmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & a_t & a_{23}\cos\rho_t - a_{13}\sin\rho_t, & \cdots, & a_{2n}\cos\rho_t - a_{1n}\sin\rho_t \\ 0 & a_{32} & a_{33} & \cdots & a_{3n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & a_{n2} & a_{n3} & \cdots & a_{nn} \end{pmatrix} .$$

Hence

$$tr(B_{t+1}) - tr(B_t) = (1 + a_t)(1 - \cos\rho_t) ,$$

and

$$tr(B_{t+1}) - tr(B_0) = \sum_{k=0}^{t} (1 + a_k)(1 - \cos\rho_k) . \tag{6-7}$$

Notice that $a_t \neq -1$ since otherwise B_{t+1} would have the eigenvalue -1, contrary to $\sphericalangle B_{t+1} < \pi$.

Since we have also $\sphericalangle B_t \leq \pi - \epsilon$ for all t and since a_t is a continuous function of B_t and y_t, by a simple compactness argument, there exists a $\delta > 0$ such that $a_t > -1 + \delta$ for all t. Hence, by (6-7),

$$n - tr(N^{-1}) \geq \delta \sum_{t=0}^{\infty} (1 - \cos\rho_t) .$$

Since $2\pi^{-2}\rho_t^2 \leq 1 - \cos\rho_t$, we get (6-6).

7. LEARNING TO PREDICT SEQUENCES OF SYMBOLS. I will describe here an algorithm $A^{\sim}$, which was inspired by the work of D. R. Morrison [19], and an idea for creating alphabets that is due to A. Ehrenfeucht [presented here with his kind permission]. Recently a number of papers studying algorithms related to Morrisons' and mine were written [1, 2, 3, 4, 5, 6, 9, 10, 16, 18, 23, 24].

Reflecting upon the functioning of intelligence (rote learning) notice that: One memorizes certain sequences of events. Then, when faced with a sequence that is a proper initial segment of one that has been memorized, one predicts the future. Observe the speed with which prediction occurs, seemingly unobstructed by the size of the set of memorized sequences.

I propose to formalize this as follows. Let Σ be a finite ordered alphabet, and $S_{0,\infty} = (a_0, a_1, \ldots)$, an infinite sequence of letters from Σ. We denote by

$$S_{i,k} = (a_i, a_{i+1}, \ldots, a_{i+k-1})$$

segments of $S_{0,\infty}$. The segment obtained by omitting the first term of a segment S is denoted

S'. A natural way to predict a_t on the basis of $S_{0,t}$ is the following. We find the longest segment $S_{i,t-i}$ $(0 < i \leq t)$ which occurs more than once in $S_{0,t}$. We find also the largest $j < i$ such that $S_{j,t-i} = S_{i,t-i}$ and we put

$$\hat{a}_t = a_{j+t-i} \ ,$$

that is, $\hat{a}_t$ is the successor of $S_{j,t-i}$ in $S_{0,t}$. Since the search for these objects may be too long (when t is very large), we define an algorithm $A^{\sim}$ which also constructs a memory M_t that facilitates this search. (But we shall not obey the requirement made above that j is the largest, since this would induce an uninteresting complication.)

The state M_t is the set of segments $S_{i,k}$ such that $i + k \leq t$, $S'_{i,k}$ occurs only once as a segment of $S_{0,t}$ and, for each i, k is the least number for which this holds. There is one exception, however: the segment $S_{i,k}$ with the largest i satisfying the above conditions is replaced by the segment $S_{i,t-i}$. We call it *the last segment* (it is a final segment of $S_{0,t}$). Moreover, we assume that M_t is ordered lexicographically and that the last segment is marked, to be immediately accessible.

Given some segment S that occurs in $S_{0,t}$, M_t allows one to find the continuations of all occurrences of S in $S_{0,t}$. One finds all $S_{i,k}$ of M_t such that S is an initial segment of $S_{i,k}$ or $S_{i,k}$ is an initial segment of S. Then one locates all $S_{j,l}$ of M_t such that $S'_{i,k}$ are their initial segments, etc. So if S appears only once in $S_{0,t}$, one gets its continuation in $S_{0,t}$. If S appears more times, one gets continuations of all those occurrences.

The algorithm $A^{\sim}$ computes $\hat{a}_t$ in the following way. It looks at the last segment S and finds segments $S_{i,k}$ in M_t such that S'' is initial in $S_{i,k}$. By the definition of S at least one such $S_{i,k}$ exists, and all of them must be longer than S''. Then $\hat{a}_t$ is the first letter of Σ that follows the initial part S'' of such $S_{i,k}$'s.

Using the above information it is easy (but tedious) to define the remaining part of the algorithm $A^{\sim}$, namely the method for constructing M_{t+1} from M_t and a_t. We shall not describe this part, but notice that, like the expected length of the computation of $\hat{a}_t$, the expected length of the computation of M_{t+1} will be a moderate function of $\log t$. Thus we observe that algorithm $A^{\sim}$ is not easily obstructed by long texts.

7.1. Observation. (i). *If the sequence $S_{0,\infty}$ is eventually periodic, then, for all large enough t, $\hat{a}_t = a_t$.*

(ii). *If $\hat{a}_t = a_t$ then M_{t+1} differs from M_t only by the addition of a_t at the end of the last segment.*

Proof. (i). If $S_{0,\infty}$ is eventually periodic, then every segment $S_{i,k}$ with large enough i equals a segment $S_{j,k}$ which overlaps with the first occurrence of the period. This yields (i).

(ii). If S is the last segment of M_t, then S'' occurs more than once in $S_{0,t}$, and so does $S''a_t$ in $S_{0,t+1}$. Hence Sa_t is the last segment of M_{t+1}.

It appears that $A^{\sim}$ is a good prediction algorithm for the letters (punctuation marks and blank included) of an ordinary long enough English text. Also one can "think" by means of $A^{\sim}$. Namely, having accumulated a sizeable M_t, one produces $\hat{a}_t$, $\hat{a}_{t+1}$, ... This can make an interesting pseudotext, a kind of echo to S_{0t}. (We could vary the choices of the letters $\hat{a}_t$ by dropping the requirement tht we choose always the first letter in Σ that was available. We could use instead some probability distribution over Σ or some other idea. This will give more originality to the "story" $\hat{a}_t$, $\hat{a}_{t+1}$, . . .).

One feature of common texts may appear excessively arbitrary: the usual alphabet. A. Ehrenfeucht proposes the following change. One finds the frequency of pairs of consecutive letters in a long English text. Then one marks the places between the letters with the frequencies of the pair at that place. In this way one gets a sequence of real numbers. One divides the text in all those places where that sequence has its local maxima. Experiments show that the mean of the lengths of the resulting parts is about four letters. (Some statistical investigations are still going on.)

We make a list of all those parts. Their frequencies decrease rather fast so that those parts can be taken as letters of a "semifinite" alphabet. The algorithm $A^{\sim}$ applied to this kind of spelling of ordinary English texts should be more interesting (although experiments have to begin with a preparation of that alphabet). Preliminary experiments of A. Ehrenfeucht show that those letters are similar to the morphemes of languages.

8. ACKNOWLEDGMENT. I am indebted to the late Stan Ulam for many conversations on the problems discussed here.

REFERENCES

1. Apostolico, A.; "Fast applications of suffix trees," in *Advances in Control*, D. G. Lainiotis and N. S. Tzannes, eds. (D. Reidel Publishing Co., Hingham, Mass., 1980) pp. 558-567.
2. Apostolico, A.; "The myriad virtues of suffix trees," Proceedings of the NATO Advanced Research Workshop on Combinatorial Algorithms on Words, Maratea, Italy (June 18-22, 1984).
3. Blumer, A., J. Blumer, A. Ehrenfeucht, D. Haussler, and R. McConnell; "Linear Size Finite Automata for the Set of all Subwords of a Word: An Outline of Results," *Bull. Euro. Assoc. Theor. Comp. Sci.* 21 (1983) 12-20.
4. Blumer, A., J. Blumer, A. Ehrenfeucht, D. Haussler, and R. McConnell; "Building a Complete Inverted File for a Set of Text Files in Linear Time," *Proc. 16th ACM Symp. Theor. Comp.* (May 1984) 349-358.
5. Blumer, J.; "Correctness and linearity of the on-line directed acyclic word graph algorithm," University of Denver Department of Mathematics and Computer Science Technical Report MS-R-8410.
6. Boyer, R. S. and J. S. Moore: "A Fast String Searching Algorithm," *CACM*, v. 20, no. 10 (October 1977) 762-772.
7. Brindley, G. S., "Nerve net models of plausible size that perform many simple learning tasks," Proc. R. Soc. London B174 (1969), 173-191.

8. Calvin, W. H. and G. A. Ojemann, *Inside the Brain* (Mentor Book, 1980).

9. Chen, M. T. and J. Seiferas; "Efficient and elegant subword-tree construction," Univ. of Rochester 1983-84 C. S. and C. E. Research Review, 10-14.

10. Chen, M. T. and J. Seiferas; "Efficient and elegant subword-tree construction," Proceedings of the NATO Advanced Research Workshop on Combinatorial Algorithms on Words, Maratea, Italy (June 18-22, 1984).

11. Cooper, L. N., "A possible organization of animal memory and learning," Proc. Nobel Symp. on Collective Properties of Physical Systems, Aspenasgarden, Sweden, 1973.

12. Eccles, J. C., *The Understanding of the Brain* (McGraw-Hill, New York, 1973).

13. Ehrenfeucht, A. and J. Mycielski, "Learnable functions," in *Foundational Problems in Special Sciences*, Butts and Hintikka, eds. (D. Reidel Publishing Co., Dordrecht-Holland, 1977) pp. 251-256.

14. Longuett-Higgins, H. C., D. J. Willshaw and O. P. Buneman, "Theories of Associative Recall," Q. R. Biophysics 3, 2 (1970), 223-244.

15. Malsburg, Chr. v. der, "Self-organization of Orientation Sensitive Cells in Striate Cortex," Kybernetik 14 (1973), 85-100.

16. Majster, M. E. and Angelika Reiser; "Efficient On-Line Construction and Correction of Position Trees," *SIAM J. Comput.*, v. 9, no. 4 (November 1980) 785-807.

17. Marr, D., "A Theory of Cerebral Neocortex," Proc. R. Soc. London. B 176 (1970), 161-234.

18. McCreight, Edward M.; "A Space-Economical Suffix Tree Construction Algorithm," *JACM*, v. 23, no. 2 (April 1976) 262-272.

19. Morrison, Donald R.; "PATRICIA - Practical Algorithm To Retrieve Information Coded In Alphanumeric," *JACM*, v. 15, no. 4 (October 1968) 514-534.

20. Mycielski, J., "A Learning Algorithm," Amer. Math. Soc. Abstracts 5 (1984), 405.

21. Mycielski, J., "Linear Dynamic Approximation Theory," J. Approximation Theory 25 (1979), 369-383.

22. Schmidt, R. F., *Fundamentals of Neurophysiology* (Springer-Verlag, Berlin, 1978).

23. Slisenko, A. O., "String-Matching in Real Time," Preprint P-1-77. The Steklov Institute of Mathematics, Leningrad Branch (September 1977) (Russian).

24. Slisenko, A. O., "String Matching in Real Time: Some Properties of the Data Structure," Mathematical Foundations of Computer Science 1978, in *Proceedings, 7th Symposium, Zakopane, Poland, 1978, Lecture Notes in Computer Science 64* (Springer-Verlag, Berlin, 1978) pp. 493-496.

☆ U.S. GOVERNMENT PRINTING OFFICE 1985—576-034/20113

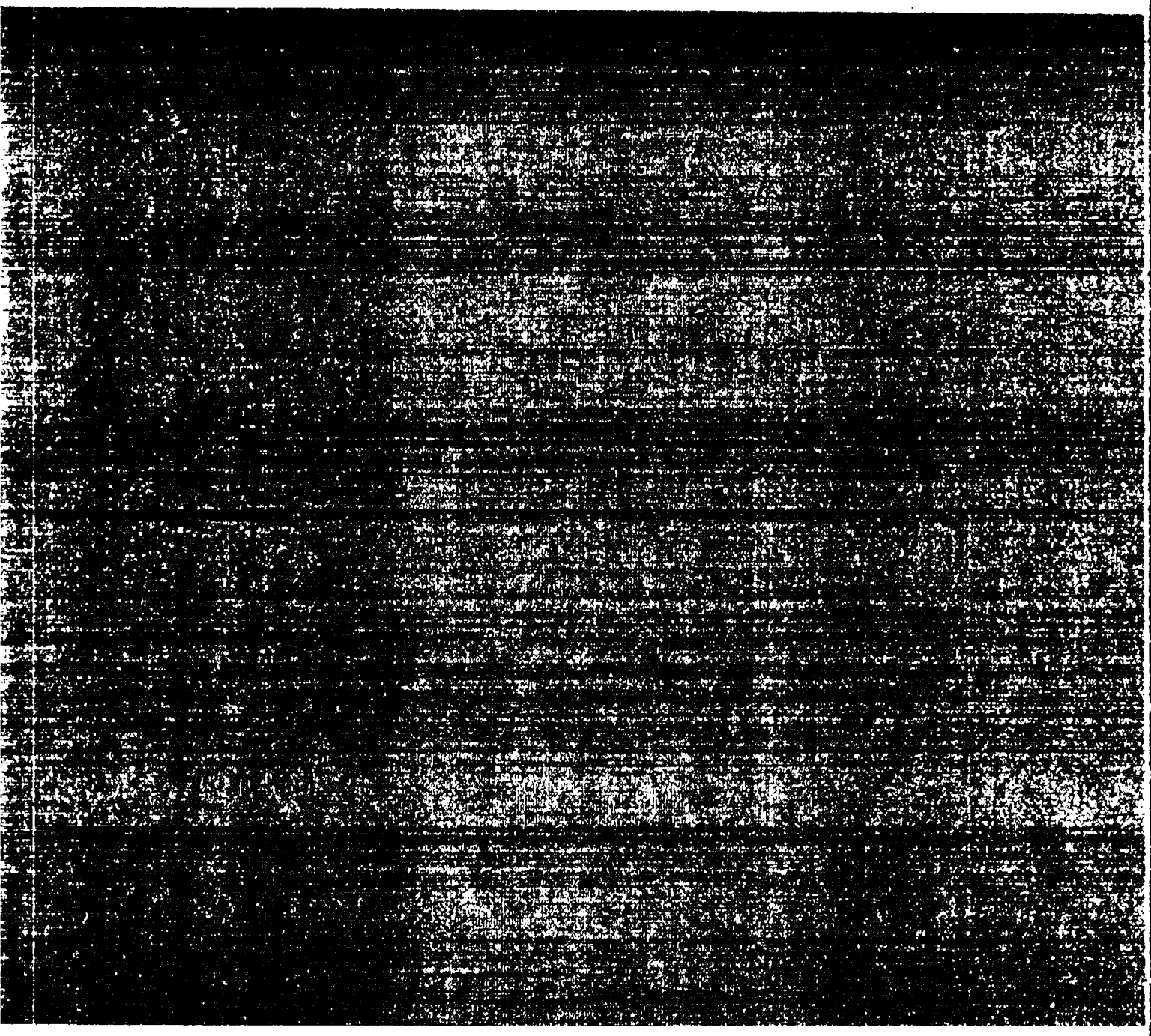
[redacted]

Printed in the United States of America
Available from
National Technical Information Service
US Department of Commerce
5285 Port Royal Road
Springfield, VA 22161

Microfiche (A01)

Page Range	NTIS Price Code	Page Range	NTIS Price Code	Page Range	NTIS Price Code	Page Range	NTIS Price Code
001-025	A02	151-175	A08	301-325	A14	451-475	A20
026-050	A03	176-200	A09	326-350	A15	476-500	A21
051-075	A04	201-225	A10	351-375	A16	501-525	A22
076-100	A05	226-250	A11	376-400	A17	526-550	A23
101-125	A06	251-275	A12	401-425	A18	551-575	A24
126-150	A07	276-300	A13	426-450	A19	576-600	A25
						601-up*	A99

*Contact NTIS for a price quote.



Los Alamos