April 31, 1970

2426

D. Discretionary Research: Uniform Permutation of Sequences, *L. Kleinrock*

1. Introduction

In SPS 37-62, Vol. III, pp. 75–79, a simple sequence permuter for disguising speech was described. It was pointed out that a single stage of this simple coder–decoder spread the samples only over three adjacent positions; in order to increase the range of permutation, a concatenation of many stages of the simple coder was considered. In this article, the behavior of this system is analyzed and experimental results showing performance of the permutation algorithm are given.

A new network structure (namely, a ternary tree) for connecting these simple coders is then analyzed, and experimental results are also given for that case. As predicted, the performance of the tree structure is far superior to that of the concatenation.

2. Performance of the Simple Coders

From SPS 37-62, Vol. III, a *simple coder* is defined as an $n = 2$ sequence-permutation coder (for which it was shown that the decoder is essentially the same as the encoder). Recall that this device generates a "key" sequence $\{k_i\}$ where $k_i \in \{1, 2\}$. This key sequence directs a sequence of speech samples $\{X_i\}$ into one of two cells of a register $P = (P_2, P_1)$. Whenever $k_i = j$, then X_i is placed in P_j ($j = 1, 2$), and the previous occupant of that cell is "bumped out," creating a permuted sequence $\{Y_i\}$. Also recall that the key sequence has the property that in every string of three key symbols, both 1 and 2 must each appear at least once. This last is accomplished by using a four-stage shift-register $S = (S_4, S_3, S_2, S_1)$ that contains two 1's and two 2's, which are circulated in one of two cyclic fashions depending upon a pseudorandom sequence $\{c_i\}$ ($c_i \in \{0, 1\}$). If $c_i = 0$, then the sequence in S is shifted one position to the right and the contents of S_4 is replaced by the contents of S_1 (denoted $S_1 \rightarrow S_4$); if $c_i = 1$, then the three rightmost stages are shifted one position to the right and $S_1 \rightarrow S_3$, $S_4 \rightarrow S_4$. The key symbol k_i is taken as the contents of S_1 at the ith shift. This gives rise to the state diagram given in Fig. 1 where the directed branches are labelled with two symbols, the first being c_i and the second being the k_i generated in passing out of the indicated state. Within each state, the contents of $S = (S_4, S_3, S_2, S_1)$ and a state number are given. Note that the k_i generated from a state is independent of c_i.

Let us now analyze the behavior of the simple coder. We assume that adjacent symbols in $\{c_i\}$ are independent

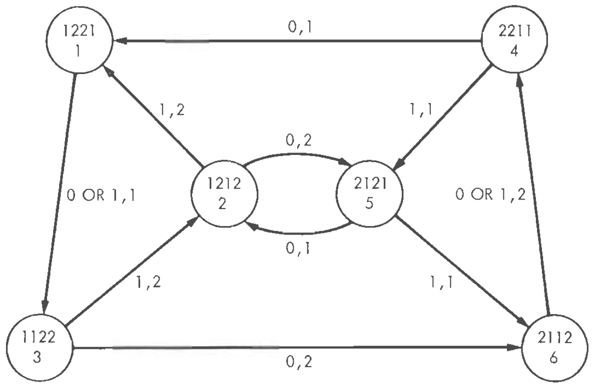


Fig. 1. State diagram for the simple coder

and that $c_i = 0$ with probability ½. From this, we may construct the following probability transition matrix $H = (h_{jk})$ for this six-state Markov process where $h_{jk} = P$ [next state is k | present state is j] as follows:

$$H = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ \tfrac{1}{2} & 0 & 0 & 0 & \tfrac{1}{2} & 0 \\ 0 & \tfrac{1}{2} & 0 & 0 & 0 & \tfrac{1}{2} \\ \tfrac{1}{2} & 0 & 0 & 0 & \tfrac{1}{2} & 0 \\ 0 & \tfrac{1}{2} & 0 & 0 & 0 & \tfrac{1}{2} \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \tag{1}$$

Note that this is a doubly stochastic matrix (rows as well as columns each sum to unity). Let $\boldsymbol{\pi} = (\pi_1, \cdots, \pi_6)$ be the probability vector of stationary state probabilities where

$$\pi_m = \lim_{n \to \infty} P\,[\text{process is in } m\text{th state on } n\text{th step}] \tag{2}$$

As is well known, we solve for $\boldsymbol{\pi}$ from the eigenvector equation

$$\boldsymbol{\pi} = \boldsymbol{\pi} H \tag{3}$$

Also, since H is doubly stochastic, the solution is immediately that

$$\pi_m = \frac{1}{6} \qquad 1 \leq m \leq 6 \tag{4}$$

(where, of course,

$$\sum_m \pi_m = 1$$

has been used). Note that states 1, 4, and 5 give $k_i = 1$ whereas states 2, 3, and 6 give $k_i = 2$.

2426

Assuming that the phase error is small so that the linear approximation holds, we get (Ref. 2)

$$P_E = \frac{1}{\pi I_0(u)} \int_0^{\pi} \exp(u \cos\phi)\, \mathrm{erfc}\left(\sqrt{\frac{2\rho_L \cos^2\theta}{\delta}} \cos\phi\right) d\phi \tag{22}$$

where

$$u = \frac{1}{\sigma_\phi^2} = \frac{\rho_L}{1 + K \sec^2\theta} \tag{23}$$

Also, $W_a = \lambda R_a$, where λ depends on the rate of the code, and we can take $\lambda = 6$ for the uncoded case. Therefore,

$$K = \frac{2\lambda\delta}{\rho_L}$$

and

$$u = \frac{\rho_L}{1 + (2\lambda\delta/\rho_L \cos^2\theta)} \tag{24}$$

Thus, given ρ_L, θ, and δ we can find P_E from Eq. (22). Conversely, given ρ_L, θ, and P_E we can find the maximum data rate that yields error probability $\leq P_E$.

For large ρ_L we have the asymptotic result

$$\frac{\delta}{\rho_L} \xrightarrow[\rho_L \to \infty]{} \frac{2\cos^2\theta}{[\mathrm{erfc}^{-1}(P_E)]^2} \tag{25}$$

which is identical to the asymptotic value of the Interplex system without carrier suppression (*Section B*). In general, the suppressed carrier method yields higher rates. The advantage is more significant for comparable rates in the two-channel system. Results for the extreme case of $\alpha = 0\ (\theta = 0)$ and $\alpha = 1\ (\theta = \pi/4)$ are given in Fig. 3, together with the results for optimum non-suppressed carrier system.

References

1. Costas, J. P., "Synchronous Communication," *Proc. IRE*, Vol. 44, No. 12, pp. 1713–1718, Dec. 1956.

2. Viterbi, A. J., *Principles of Coherent Communication*, McGraw-Hill Book Co., Inc., New York, 1966.

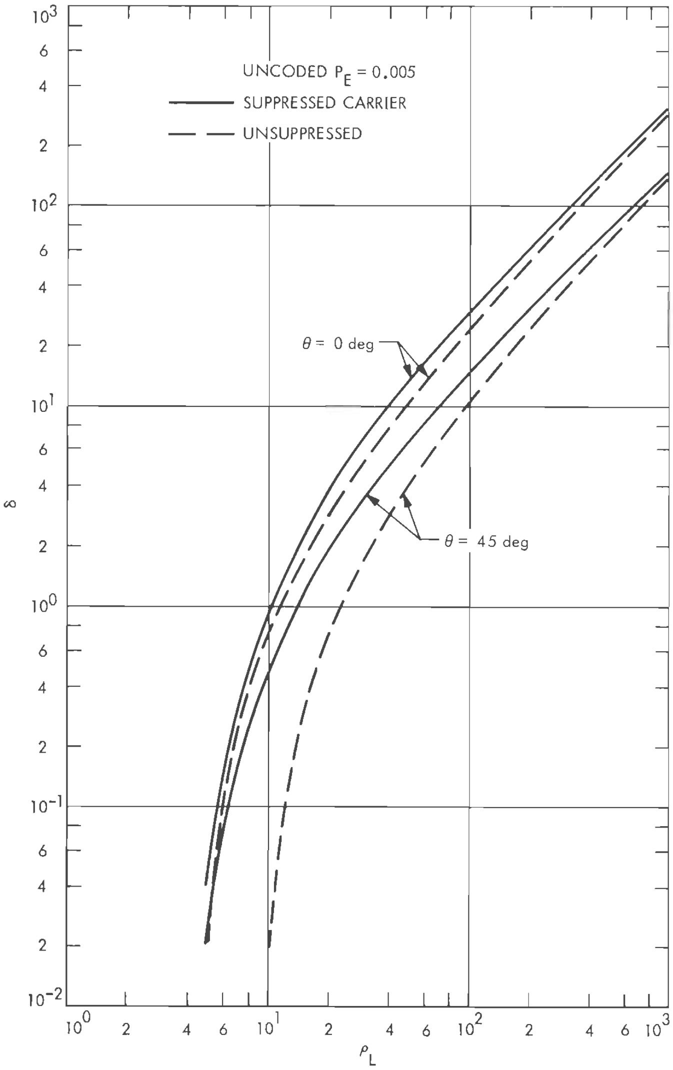


Fig. 3. Rate/bandwidth vs loop SNR for two-channel Interplex system

We are interested in calculating two probability distributions. The first is for the random variable U_i where

$$U_i = j - i \tag{5}$$

if $Y_j = X_i$. That is, U_i is equal to the relative shift of X_i from its original position (i) in the unpermuted sequence to its new position (j) in the permuted sequence. In SPS 37-62, Vol. III, it was shown that X_i will be "bumped out" of the P register at step $i + d_i - 2$ if k_i is next repeated d_i steps later, where $d_i \in \{1, 2, 3\}$. Thus,

$$U_i = d_i - 2$$

and so $U_i \in \{-1, 0, 1\}$. The second distribution of interest is for the random variable V_i where

$$V_i = n - m \tag{6}$$

if $Y_i = X_m$ and $Y_{i+1} = X_n$. That is, V_i is the difference in positions in the original (unpermuted) sequence of the ith and $(i + 1)$th samples which appear in the permuted sequence. For example, if $\{X_i\}$ permutes into $\{Y_i\}$ as shown in Table 1, then $\{U_i\}$ and $\{V_i\}$ are as given in that table.

Table 1. Example of sequences $\{U_i\}$ and $\{V_i\}$

i	X_i	Y_i	U_i	V_i
1	X_1	X_2	1	−1
2	X_2	X_1	−1	3
3	X_3	X_4	1	−1
4	X_4	X_3	−1	2
5	X_5	X_5	0	1
6	X_6	X_6	0	1
7	X_7	X_7	0	
•	•	•	•	•
•	•	•	•	•
•	•	•	•	•

We now compute $P[U_i = k]$ and $P[V_i = k]$. From the state diagram in Fig. 1, we immediately see that beginning in each state, there is a short *deterministic* sequence of key symbols that are generated independently of $\{c_i\}$. These are listed in Table 2 where it is assumed that we are in state m just prior to step i and generate the key symbol k_i upon leaving that state. Note the duality between states 1 and 6, 2 and 5, and 3 and 4.

Table 2. Deterministic portion of $\{k_i\}$ given state at step i

Initial state m	k_i	k_{i+1}	k_{i+2}	k_{i+3}
1	1	2	2	1
2	2	1	2	
3	2	2		
4	1	1		
5	1	2	1	
6	2	1	1	2

In order to calculate the distributions for U_i and V_i, we only need the distribution of the next four key symbols from a given state. We already know the length-4 sequence (which is deterministic) following states 1 and 6. From state 2, we go to states 1 or 5 (each with probability ½) which, from Table 2, must then give the length-4 sequences (following state 2) as 2122 and 2121, respectively. By duality, state 5 must give 1211 and 1212 (each with probability ½). From state 3, we go to states 2 or 6, which give 2212 and 2211, respectively (each with probability ½). Thus, the probability of length-4 sequences from each state is known (Table 3).

Table 3. Length-4 sequences of key symbols from various states

Initial state	Possible length-4 sequences of key symbols	Conditional probability of sequence
1	1 2 2 1	1
2	2 1 2 2	½
2	2 1 2 1	½
3	2 2 1 2	½
3	2 2 1 1	½
4	1 1 2 2	½
4	1 1 2 1	½
5	1 2 1 2	½
5	1 2 1 1	½
6	2 1 1 2	1

Now to find $P[U_i = k]$, $k = -1, 0, 1$, recall that $U_i = d_i - 2$. From this last and from Table 3 we have

$$\begin{aligned} P[U_i = -1] &= P[d_i = 1] \\ &= P[k_i = k_{i+1}] \\ &= \pi_3 + \pi_4 \\ &= \frac{1}{3} \end{aligned} \tag{7}$$

$$\begin{aligned} P[U_i = 0] &= P[d_i = 2] \\ &= P[k_i = k_{i+2} \neq k_{i+1}] \\ &= \pi_2 + \pi_5 \\ &= \frac{1}{3} \end{aligned} \tag{8}$$

$$\begin{aligned} P[U_i = 1] &= [d_i = 3] \\ &= [k_i = k_{i+3} \neq k_{i+1} = k_{i+2}] \\ &= \pi_1 + \pi_6 \\ &= \frac{1}{3} \end{aligned} \tag{9}$$

Thus, from Eqs. (7), (8), and (9), we have

$$P[U_i = k] = \frac{1}{3} \qquad k = -1, 0, 1 \tag{10}$$

Now we calculate $P[V_i = k]$. We are interested in observing two adjacent symbols in the $\{Y_i\}$ sequence. These may be studied by inspecting Table 3 and observing the samples generated as a result of the third and fourth key symbols in each sequence shown. Recall that if the key sequence is $k_{i-1}, 1, 1, 2$ (or $k_{i-1}, 2, 2, 1$) then k_{i-1} must be 2 (or 1) since each key symbol must occur at least once in each string of three. Thus, from state 1, the key sequence $k_i = 1$, $k_{i+1} = 2$, $k_{i+2} = 2$, $k_{i+3} = 1$ would produce $Y_i = X_{i+1}$, $Y_{i+1} = X_i$ giving $V_i = -1$ from Eq. (6). Similar calculations are possible for the other states, and are summarized in Table 4 in a fashion similar to Table 3.

Since $\pi_m = 1/6$ for $m = 1, 2, \cdots, 6$, we may then calculate $P[V_i = k]$ from

$$P[V_i = k] = \sum_{m=1}^{6} P[V_i = k|m]\, \pi_m \tag{11}$$

where $P[V_i = k|m]$ is given as the last column in Table 4 for all non-zero terms. Thus,

$$P[V_i = k] = \begin{cases} \frac{1}{3} & k = -1 \\ \frac{1}{6} & k = 1 \\ \frac{1}{3} & k = 2 \\ \frac{1}{6} & k = 3 \\ 0 & \text{otherwise} \end{cases} \tag{12}$$

Equations (10) and (12) give the distributions for U_i and V_i, respectively. Of more importance is the fact that U_i is *uniformly* distributed over $\{-1, 0, 1\}$. Figure 2 shows the experimentally obtained histograms for U_i (Fig. 2a) and V_i (Fig. 2b) where a sequence of 7500 samples X_i was used. This figure corresponds to our calculations in Eqs. (10) and (12).

3. The Concatenation of Simple Coders

In order to increase the span of samples over which we permute, it was suggested in SPS 37-62, Vol. III that we form a concatenation of M simple coders as shown in

Table 4. Calculation of V_i

Initial state	k_i	k_{i+1}	k_{i+2}	k_{i+3}	Y_i	Y_{i+1}	V_i	Conditional probability
1	1	2	2	1	X_{i+1}	X_i	-1	1
2	2	1	2	2	X_i	X_{i+2}	2	½
2	2	1	2	1	X_i	X_{i+1}	1	½
3	2	2	1	2	X_{i-1}	X_{i+1}	2	½
3	2	2	1	1	X_{i-1}	X_{i+2}	3	½
4	1	1	2	2	X_{i-1}	X_{i+2}	3	½
4	1	1	2	1	X_{i-1}	X_{i+1}	2	½
5	1	2	1	2	X_i	X_{i+1}	1	½
5	1	2	1	1	X_i	X_{i+2}	2	½
6	2	1	1	2	X_{i+1}	X_i	-1	1

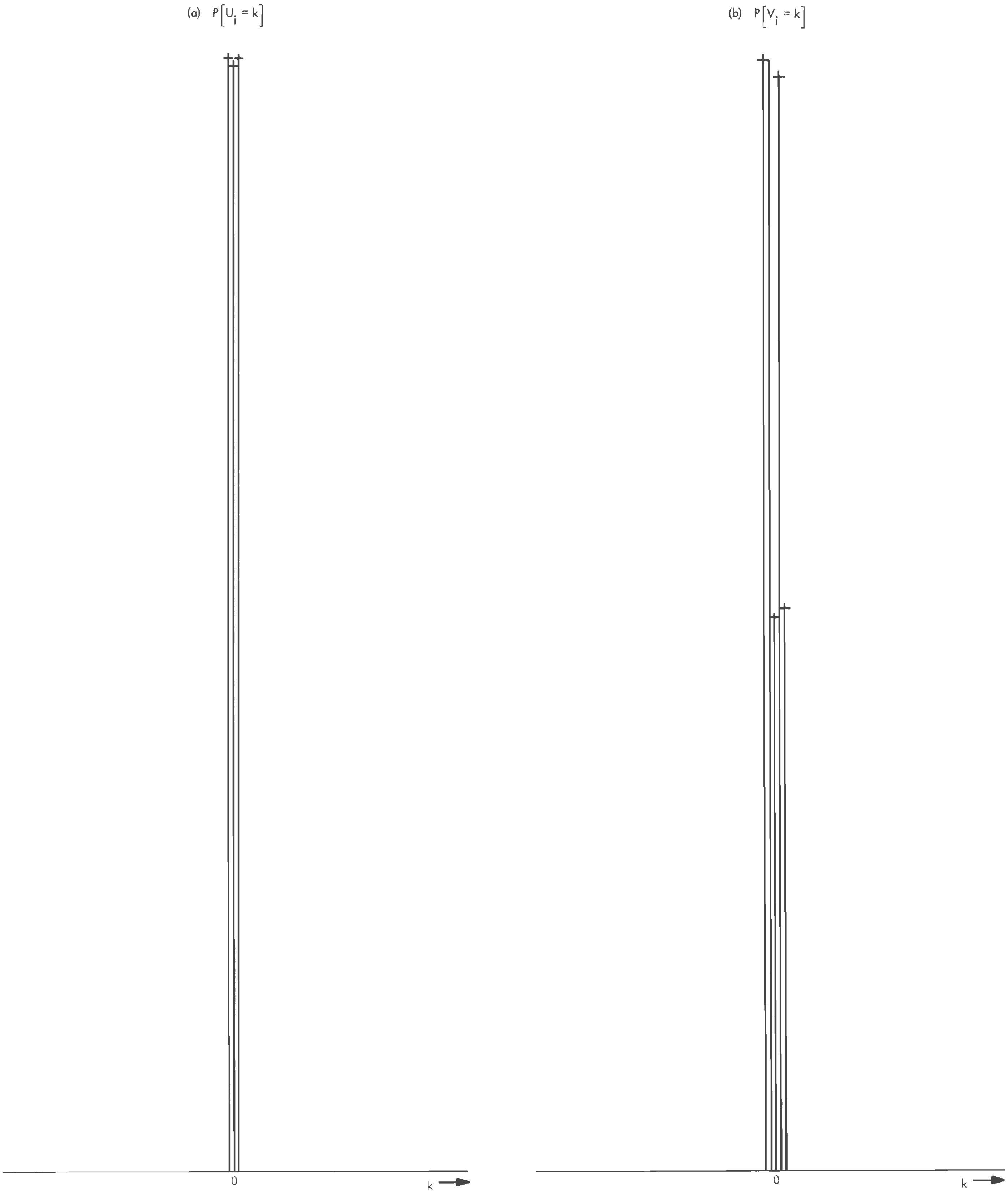


Fig. 2. Histograms for U_i and V_i

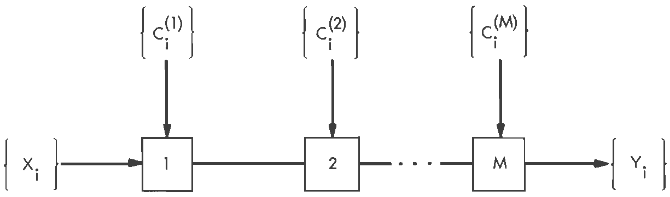


Fig. 3. *M*-level encoder concatenation

Fig. 3. In this configuration, the mth stage uses the pseudorandom sequence $\{C_i^{(m)}\}$ to generate the mth key sequence $\{k_i^{(m)}\}$. The decoder is again a chain of encoders in the reverse order such that the mth decoder key sequence, $\{K_i^{(m)}\} = \{k_i^{(M-m+1)}\}$ (SPS 37-62, Vol. III).

We may analyze the mixing behavior of this chain of M simple coders as follows: From Eq. (10) we know that $U_i = -1, 0, 1$, each with probability ⅓. If we take the permuted output from stage 1 of Fig. 3 and feed it into a second independent stage as shown, the output sequence $\{Y_i^{(2)}\}$ from the second stage will be a permutation of the first permuted sequence. Define $U_i^{(2)}$ as

$$U_i^{(2)} = j - i \tag{13}$$

if $Y_j^{(2)} = X_i$, and so $U_i^{(2)}$ is the relative shift of X_i from its original position, i, in the unpermuted sequence to its new position, j, in the twice permuted sequence. Similarly, we define

$$U_i^{(m)} = j - i \tag{14}$$

if $Y_j^{(m)} = X_i$ where $\{Y_i^{(m)}\}$ is the m-times permuted output sequence from the mth stage.

Since all the permutations are independent of each other (i.e., $\{C_i^{(m)}\}$ are independent pseudorandom sequences), then the random variable $U_i^{(2)}$ is merely the sum of two independent random shifts giving

$$P[U_i^{(2)} = k] = \sum_{j=-1}^{1} P_1[U_i = j]\, P_2[U_{i+j} = k - j] \tag{15}$$

where $P_\alpha[U_i = j]$ is the probability distribution for the relative shift of position to the ith input sample in the αth stage ($\alpha = 1, 2, \cdots, M$). Clearly, the summation in Eq. (15) is the convolution of the distribution $P[U_i = k]$, which is itself evaluated at the kth position. Let us denote convolution by $*$, giving

$$P[U_i^{(2)} = k] = P[U_i = k_1] * P[U_1 = k_2] \tag{16}$$

Clearly, also

$$P[U_i^{(m)} = k] = \overbrace{P[U_i = k_1] * P[U_i = k_2] * \cdots * P[U_i = k_m]}^{m\text{ times}} \tag{17}$$

since the $U_i^{(m)}$ is the sum of m independent random shifts. The author knows no simple form for expressing the probability in Eq. (17). One may define the generating function for $P[U_i^{(m)} = k]$ as

$$G_m(Z) \equiv \sum_{k=-m}^{m} P[U_i^{(m)} = k]\, Z^k \tag{18}$$

We may then use the usual properties of generating functions for sums of independent random variables to get

$$G_m(Z) = [G_1(Z)]^m \tag{19}$$

From Eqs. (10) and (18), we then obtain

$$G_1(Z) = \frac{Z^{-1} + 1 + Z}{3} \tag{20}$$

and, thus, from Eq. (19),

$$G_m(Z) = \left(\frac{1 + Z + Z^2}{3Z}\right)^m \tag{21}$$

From Eq. (18), the value for $P[U_i^{(m)} = k]$ may be obtained as the coefficient of Z^k $(k = 0, \pm1, \cdots, \pm m)$ in Eq. (21). We may also write

$$G_m(Z) = \left(\frac{1}{3}\right)^m \sum_{i=0}^{m} \sum_{j=0}^{i} \binom{m}{i} \binom{i}{j} Z^{m-i-j} \tag{22}$$

SPS 37-64-3 37 70

Inversion of Eq. (22) has not been carried out. However, we may compute $P\,[U_i^{(m)} = k]$ from the following Pascal-like triangle:

						1						
					1	1	1					
				1	2	3	2	1				
			1	3	6	7	6	3	1			
		1	4	10	16	19	16	10	4	1		
	1	5	15	30	45	51	45	30	15	5	1	
1	6	21	50	90	126	141	126	90	50	21	6	1

This triangle is created by making an element equal to the sum of its *three* immediate neighbors in the row above it. Thus, to find $P\,[U_i^{(m)} = k]$, we first locate the kth element (from the center vertical column) in the mth row; the top element of the triangle corresponds to $n = k = 0$. We then take this element and divide by 3^m. For example,

$$P\,[U_i^{(5)} = -4] = \frac{5}{3^5} = \frac{5}{243}$$

Obviously, however, the distribution $P\,[U_i^{(m)} = k]$ is tending to the gaussian distribution in the limit as $m \to \infty$. Therefore, the cumulative distribution can easily be approximated. In any case, however, we have *not* achieved the performance we were seeking; namely, a *uniform* permutation over a large number of shifts. We have obtained a near-gaussian shape that does not spread fast enough as M increases (it spreads like $\sqrt{M}$ instead of like M). Empirical results from simulation are given in Fig. 4, where we show the histograms for $U_i^{(m)}$ for $m = 2, 3, 5, 10$, and 30 ($m = 1$ is given in Fig. 2). Also shown are the histograms for $V_i^{(m)}$ where

$$V_i^{(m)} = n - p \tag{23}$$

if $Y_i^{(m)} = X_p$ and $Y_{i+1}^{(m)} = X_n$ with the same interpretation as for V_i in Eq. (6). (The computation of $P\,[V_i^{(m)} = k]$ is very difficult and is not discussed further.) Note the rapid convergence to gaussian for both distributions.

4. An Improved Configuration—the Ternary Tree

The concatenation of M simple coders leads to a permutation that is not distributed uniformly over the range $[-M, M]$, as was seen in *Subsection 3.*

We now consider the ternary tree configuration shown in Fig. 5. This figure shows the case $M = 13$ where each square box corresponds to a simple coder whose properties are discussed in *Subsection 2.* The interpretations of the connections is as follows: Consider Coder 10; the three inputs to this coder come from the outputs of coders 1, 2, and 3. The connection notation means that coder 10 receives an input from coder 1, followed by an input from coder 2, followed by an input from coder 3 (this sequence is repeated indefinitely). The input sequence

$$\{X_1, X_2, X_3, \cdots,\}$$

is separated into nine sequences:

$$\{X_1, X_{10}, X_{19}, \cdots,\}, \{X_2, X_{11}, X_{20}, \cdots,\}, \cdots, \{X_9, X_{18}, X_{27}, \cdots,\}$$

In general, we have A tiers of coders where the ath tier contains 3^{a-1} simple coders ($a = 1, 2, \cdots, A$). In this case, we have

$$M = \frac{3^A - 1}{2} \tag{24}$$

The input stream is separated into 3^{A-1} sequences. Figure 5 shows the case $A = 3$, $M = 13$.

As for decoding, we merely create a matching decoding ternary tree in the reverse configuration. As an example, consider the two-tier ($A = 2$) case shown in Fig. 6. Here, the input sequence $\{X_i\}$ is passed through a splitting box σ that creates 3 streams (3^{2-1}) as shown. The coder creates the permuted output sequence $\{Y_i\}$. The decoder accepts the sequence $\{Y_i\}$ as input after transmission over some channel and passes it through the reverse tree decoder. This produces three output streams that are then passed through the merging box μ to recreate the original sequence $\{X_i\}$. The pseudorandom sequence $\{C_i^{(m)}\}$ for coder box m is also used for decoder box m'.

We now consider the shift $U_i^{(A)}$ defined for the A-tier system as

$$U_i^{(A)} = j - i \tag{25}$$

if $Y_j = X_i$ where the overall output sequence is $\{Y_i\}$. We will show that

$$P\,[U_i^{(A)} = k] = \begin{cases} 3^{-A} & k = 0, \pm 1, \cdots, \pm M \\ 0 & \text{otherwise} \end{cases} \tag{26}$$

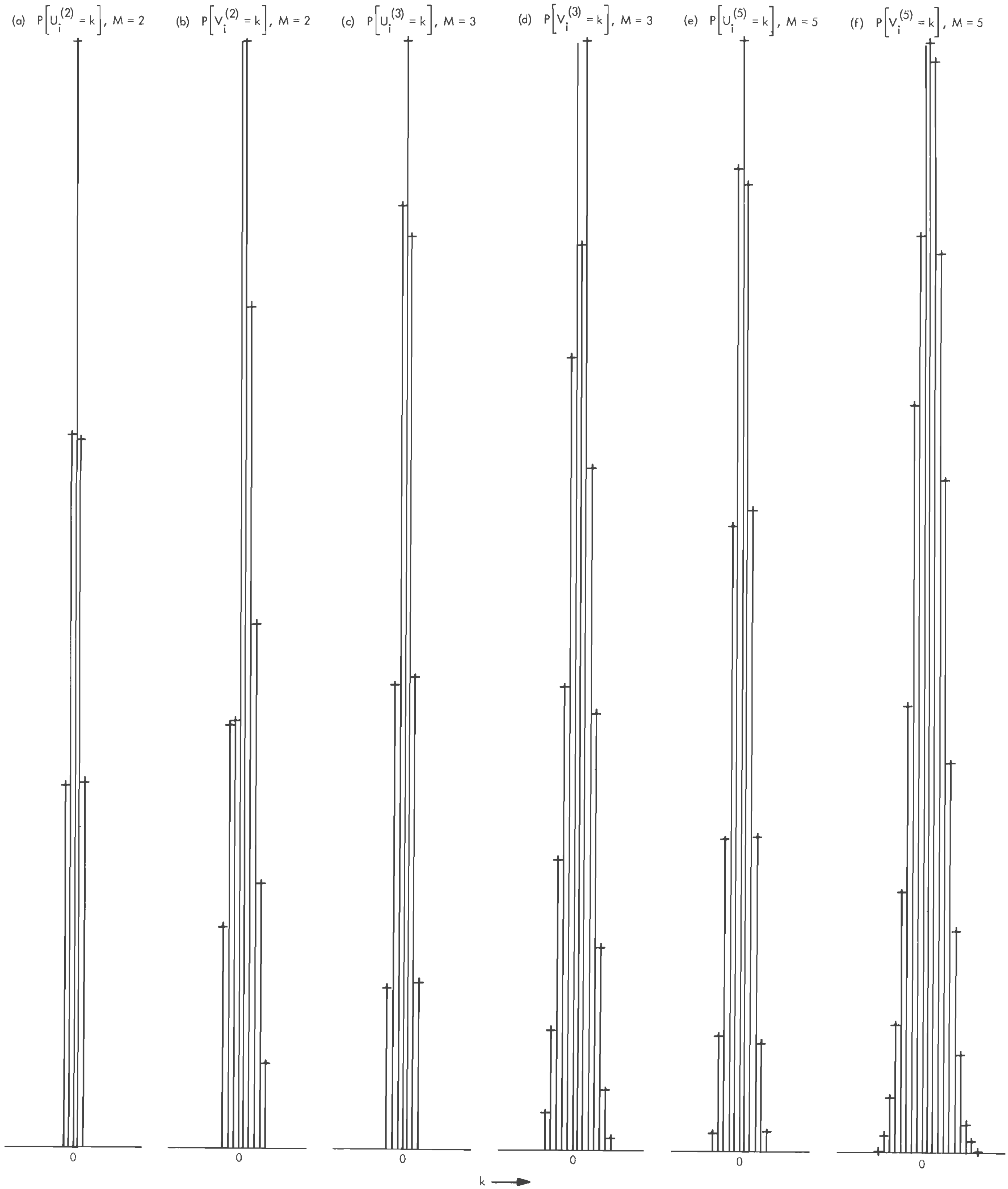


Fig. 4. Histograms for $U_i^{(m)}$ and $V_i^{(m)}$ for $m = 2, 3, 5, 10,$ and 30 (for the concatenated structure)

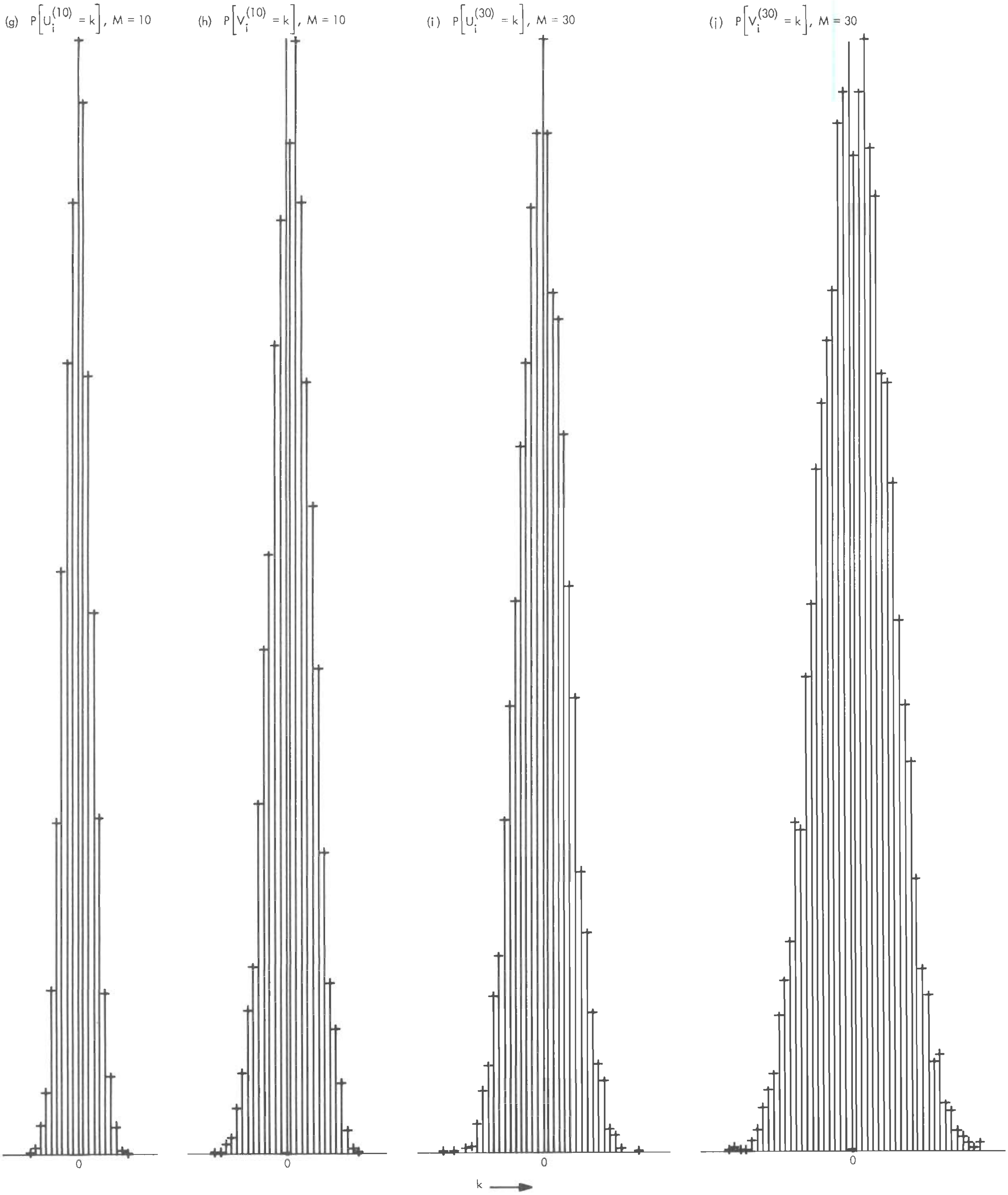

(g) $P\left[U_i^{(10)} = k\right]$, M = 10
(h) $P\left[V_i^{(10)} = k\right]$, M = 10
(i) $P\left[U_i^{(30)} = k\right]$, M = 30
(j) $P\left[V_i^{(30)} = k\right]$, M = 30
0
0
0
0
k

Fig. 4 (contd)

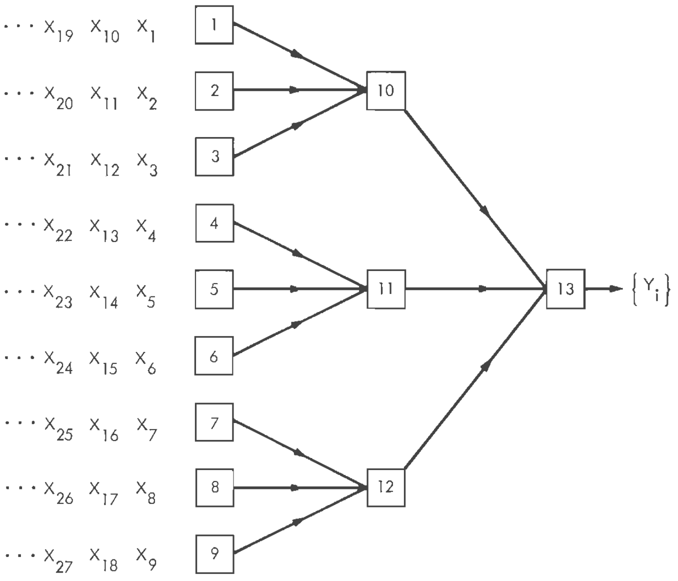


Fig. 5. The ternary tree structure

That is, $U_i^{(A)}$ is *uniformly* distributed over $3^A = 2M + 1$ positions. From Eq. (10), we know for $A = 1$ that

$$P\,[U_i^{(1)} = k] = 3^{-1} \qquad k = -1, 0, 1 \tag{27}$$

Now consider $A = 2$. For this case, we separate the input sequence into three subsequences as shown in Fig. 6. Consider X_8. In passing through coder 2, we have (from Eq. 27) that X_8 will appear in place of X_5, X_8, or X_{11}, each with probability ⅓. Then, when each of the singly permuted subsequences leave the first tier, the elements are interlaced as an input to coder 4. In that case, we have the following: if X_8 had moved to position 5 in passing through coder 2, then (in passing through coder 4) it will move to original positions 4, 5, or 6 (each with probability ⅓). If X_8 had remained in position 8, it will move to positions 7, 8, or 9. If X_8 had moved to position 11, it will now move to positions 10, 11, or 12. Since each of these nine possibilities is equally likely, we have proved Eq. (26) for $A = 2$. Thus, for $A = 3$ in Fig. 5, X_{14} will move to positions 4, 5, 6, 13, 14, 15, 22, 23, or 24 (each with probability 1/9) in passing through the first two tiers as just proven. Now, in a similar fashion, each of those nine possible positions moves is equally likely to three others in passing through the third tier. Thus X_{14} moves to positions $1, 2, 3, \cdots, 26$, or 27 (each with probability 1/27) for $A = 3$. By induction, proof of Eq. (26) follows.

Define $V_i^{(A)}$ for A tiers as

$$V_i^{(A)} = n - p \tag{28}$$

if $Y_i = X_p$ and $Y_{i+1} = X_n$. The computation of $P\,[V_i^{(A)} = k]$ is also difficult in this case and is not carried out.

Figure 7 gives empirical results for the ternary tree. Histograms are shown for $U_i^{(A)}$ and $V_i^{(A)}$ for $A = 2, 3, 4$. Note the essentially uniform distribution for $U_i^{(A)}$, as we had been seeking! The distribution for $V_i^{(A)}$ remains gaussian-like with every third entry of larger value.

5. Conclusions

We have analyzed and experimented with the M-stage concatenation of simple coders for use in speech scrambling. As predicted, the performance is far below that which we require for effective scrambling.

We proposed a new topology, the ternary tree, for more effective scrambling and found that this structure produced the sought-after uniform distribution over $2M + 1$ positions (for M simple coders). The simple coders are such that the decoders are identical to the encoders. Thus, we propose to use this method for permuting sequences of speech samples.

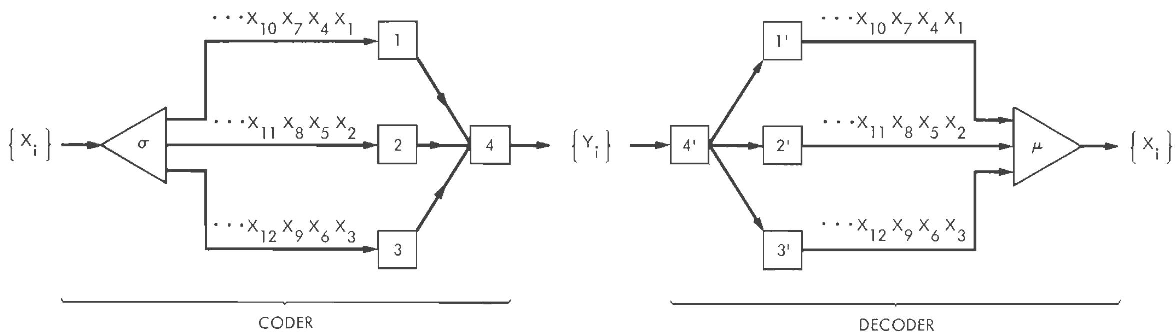


Fig. 6. Coder–decoder for A = 2

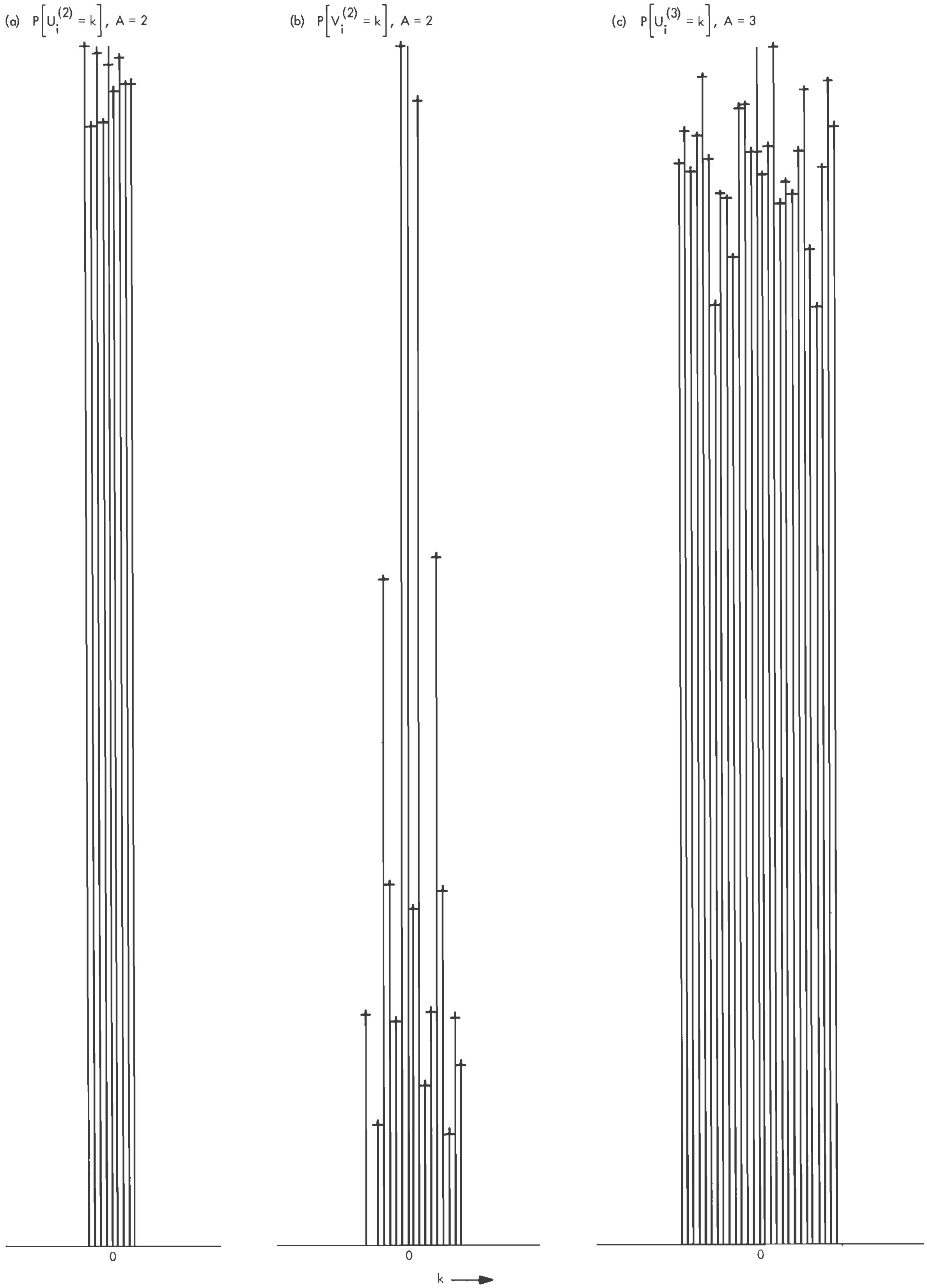
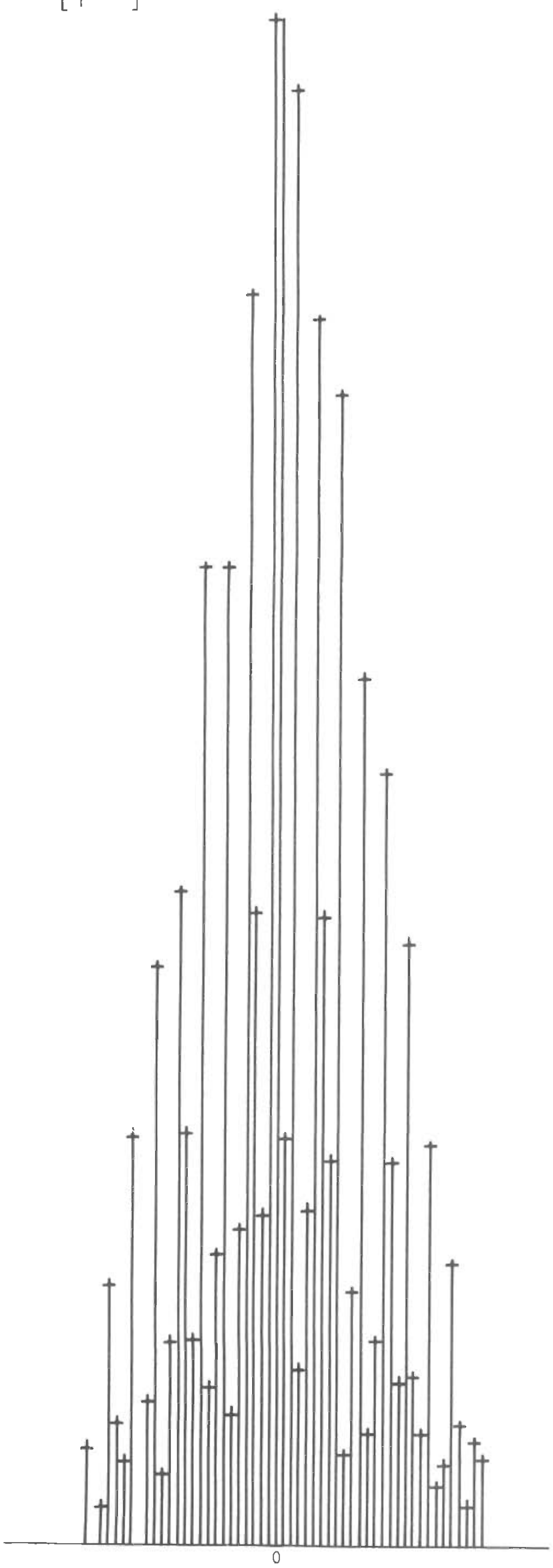


Fig. 7. Histograms for $U_i^{(A)}$ and $V_i^{(A)}$ for $A = 2$, 3, and 4 (for the ternary tree)

(d) $P\left[V_i^{(3)} = k\right]$, A = 3



(e) $P\left[U_i^{(4)} = k\right]$, A = 4

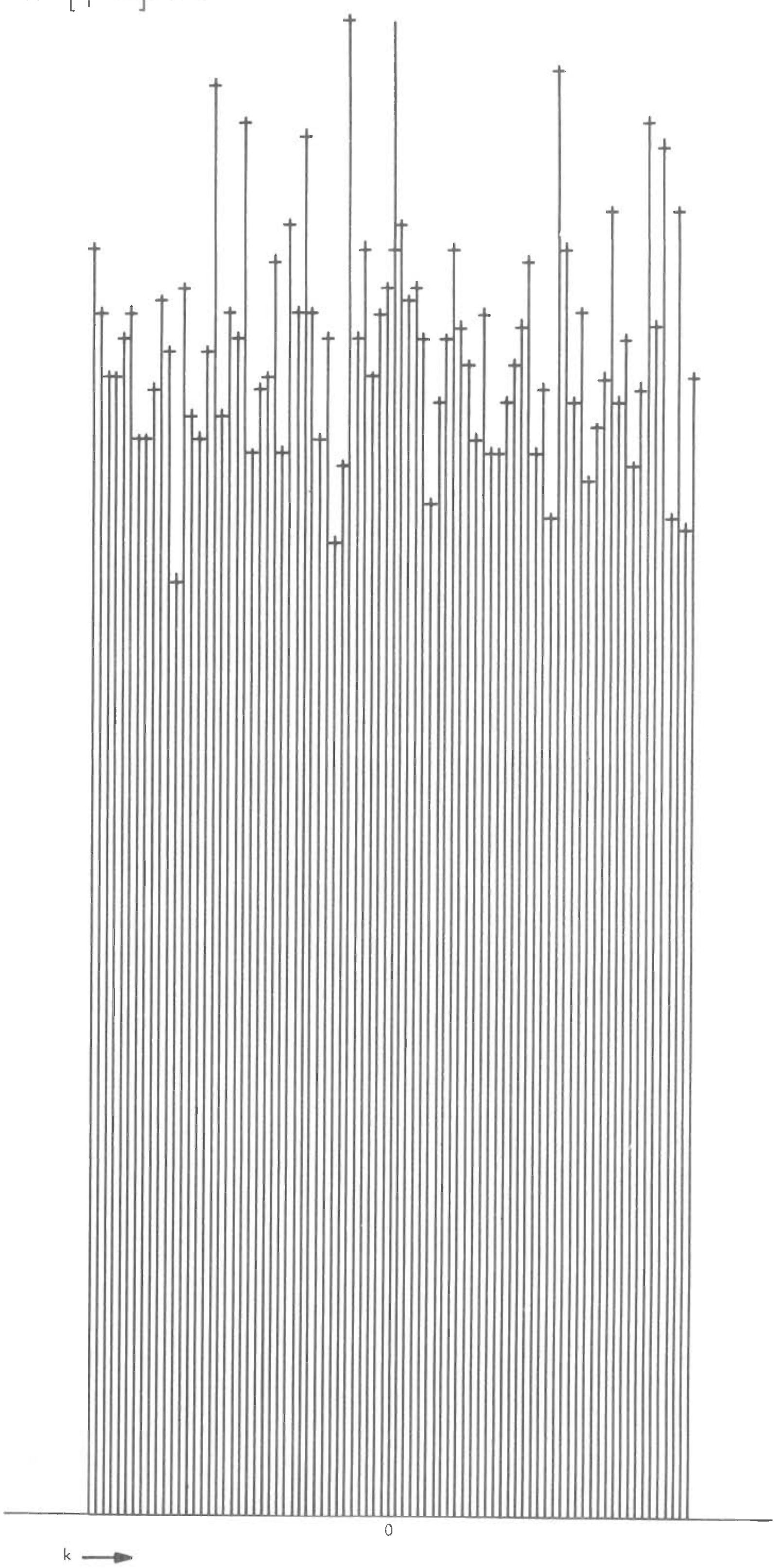


Fig. 7 (contd)

(f) $P\left[V_i^{(4)} = k\right]$, A = 4

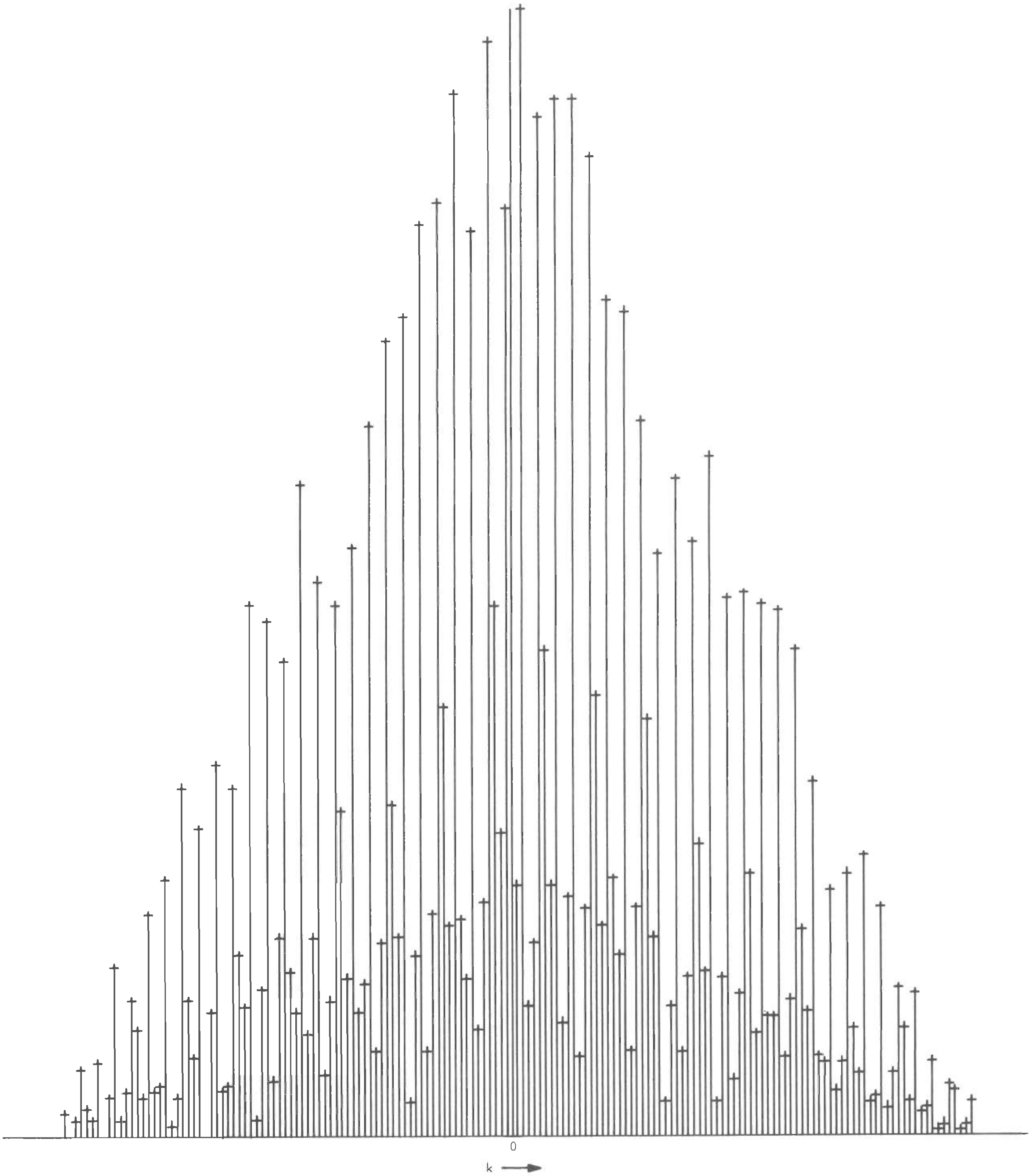


Fig. 7 (contd)