THE BELL SYSTEM

Technical Journal

DEVOTED TO THE SCIENTIFIC AND ENGINEERING ASPECTS OF ELECTRICAL COMMUNICATION

Volume 47 April 1968 Number 4

THE BELL SYSTEM TECHNICAL JOURNAL

THE BELL SYSTEM TECHNICAL JOURNAL is published ten times a year by the American Telephone and Telegraph Company, B. S. Gilmer, President, C. E. Wampler, Vice President and Secretary, J. J. Scanlon, Vice President and Treasurer. Checks for subscriptions should be made payable to American Telephone and Telegraph Company and should be addressed to the Treasury Department, Room 2312C, 195 Broadway, New York, N. Y. 10007. Subscriptions $7.00 per year; single copies $1.25 each. Foreign postage $1.00 per year; 15 cents per copy. Printed in U.S.A.

THE BELL SYSTEM
TECHNICAL JOURNAL

Volume 47 April 1968 Number 4

Some Theorems and Procedures for Loop-Free Routing in Directed Communication Networks

By R. MAGNANI

(Manuscript received November 15, 1967)

This paper examines two general methods for specifying directed routing patterns in communication networks. Hierarchical routing, as currently used in the toll network, is such a directed routing pattern. However, it is only one member of a large set of possible routing strategies that can be realized by storing, in each office, a list of outgoing trunk groups and an order-of-choice for these groups for each received call address. The general class of routing strategies is defined by this method of realization, subject to the constraint that routing patterns be loop-free. The paper discusses procedures for generating loop-free patterns, for detecting whether or not a given pattern is loop-free, and for specifying "good" patterns from the large number which are realizable.

I. INTRODUCTION

The fundamental problem which besets people concerned with the design of communication networks is how to provide a network which is, at once:

(*i*) Of sufficient routing capability to allow any two users to be connected with a high probability of success.

(*ii*) Economical in its use of transmission facilities and switching centers.

(*iii*) Capable of surviving extensive natural or man-made damage.

(*iv*) Adaptable to changing traffic patterns and overload situations.

(*v*) Capable of being engineered and implemented in small sections, over a period of years, by many different people.

This is a problem of such complexity that, with the present state of the theory, it must be attacked piecemeal.

This paper deals with a small but important section of the problem. It considers some of the topological properties of communication networks and examines a class of alternate routing strategies from a general point of view. Our purpose is to state rules which will allow the orderly production of routing patterns, for arbitrary networks, by a computer. We approach the problem in three stages:

(*i*) Several simple rules are stated for producing "loop-free" routing patterns.

(*ii*) The rules are "generalized" to allow the proof of some theorems about the extent of their application.

(*iii*) A more limited but practical statement of the rules is presented followed by several heuristic procedures, based on these rules, which can be used to specify "good" routing patterns from the large number which can be generated.

II. BACKGROUND

What is fundamental to the routing process as it is practiced today in the telephone network? Certainly one of the answers is that each office stores a list of outgoing trunks and an order of choice for those trunks for each possible call address that can be received. We can think of the aggregate of these lists as constituting a "routing map" which has been distributed among many offices. The "map" which is currently stored in the telephone network realizes the hierarchical routing plan.

But suppose we wish to examine strategies which do not require a "hierarchy" of offices? Is there a way to realize a general class of routing maps which will allow offices to be of *equal* rank and which can be implemented in the same fashion as the current hierarchical plan? The answer is that such a class of routing maps does exist and that, indeed, the present hierarchical map is simply one member of the set. To see this, consider the simple network of four offices shown in Fig. 1. This may be an entire network or some subset of a much larger network with the connecting trunk groups omitted. For the

present, we will assume the trunk groups shown are all two-way (that is, contain trunks that can be seized from either end) and that routing between the offices is subject to the following constraints:

(*i*) No routing control information is passed between offices.

(*ii*) Offices do not check for shuttle.*

The resulting network is a fair approximation to a subset of the present day commercial network.† Routing between offices is accomplished as follows :

(*i*) Each office is assigned a unique address such as NNX or NPA-NNX.

(*ii*) Upon receiving a call request, an office checks to see if it is the destination office. If it is, the call is counted as a success (although in practice the called subscriber's line must still be checked). If not, the office consults a routing table and, on the basis of the received NNX, selects an outgoing trunk group.

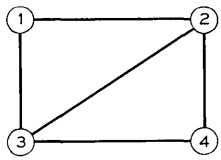


Fig. 1 — A four-office network.

(*iii*) If a trunk in the group is available, it is seized and the called address is passed over it to the next office. At this point we return to step (*ii*).

(*iv*) If no trunks are available, the office again consults its routing table, to find an alternate trunk group, and returns to step *iii*. The process continues until all alternate trunk groups have been tried.

(*v*) If no trunks are available in any of the alternate trunk groups, the call is blocked, and the caller is so notified.

For a particular network and destination office, this process may be conveniently summarized on the graph which represents the network; for routing to a particular office, we assign each trunk group in the network a direction and an order of choice out of the office in which it originates. For example, if office 4 were the destination office in the

* Shuttle refers to routing a call out over the same trunk group on which the call arrived.

† With one-way trunk groups omitted.

network of Fig. 1, we might summarize routing to this office by the use of Fig. 2.

Here we are to understand that calls must route in the directions shown and that trunk groups are to be selected in the given order (beginning with 1). The routes between office 1 and office 4 can easily be seen to be: 1–2–4, 1–3–4, and 1–3–2–4, where the numbers represent *office* numbers and the routes are listed in the order they will occur. Similarly, offices 2 and 3 can be seen to have routes 3–4, 3–2–4, and 2–4.

Fig. 2 represents routing from *all offices* to office 4 and will be said to be a *directed routing pattern* to office 4 on the network of Fig. 1. When such a routing strategy is followed, the way in which a call routes from a particular office is independent of the past history of the call; this is characteristic of routing in the present DDD network. To

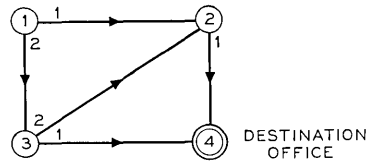


Fig. 2 — A directed routing pattern to office 4.

completely specify routing in the network of Fig. 1, four directed routing patterns are necessary, one with each office as destination. The direction and order of choice assigned to the trunk groups will vary from pattern to pattern depending on the destination office. To see that the hierarchical pattern in use today is of this type, we need simply draw the pattern. (See Fig. 3.)

III. ROUTING PATTERNS

Let us examine some simple rules for constructing such patterns and adopt the standard graph theory terminology: "branch" or "link" for trunk group, and "node" for office.

Because we assume offices do not check for shuttle or looping, we will require the patterns we generate to be loop-free.* A "loop" for our purposes is defined as: a set of branches and nodes (not containing the destination node) constitutes a *loop* if we can select any node

*It has been suggested by J. H. Weber that a small probability of looping may be acceptable if looping can be detected (see Ref. 1).

in the set and, by following the directions of the branches, traverse each branch once to form a path which returns to the selected node (that is, loops must be "directed"). It is not clear in the case of a large network just how one goes about obtaining a loop-free directed routing pattern, particularly if the network is nonplanar. To demonstrate that a systematic procedure is required, we invite you to try to draw a loop-free pattern on the network of Fig. 4a.

A closely related problem is illustrated by Fig. 4b in which we are given a routing pattern and asked to determine whether or not it contains a loop. In this case, the single loop that the network does contain may or may not be obvious to you; however, if the network were much larger, say 40 nodes, a systematic procedure again would be required.

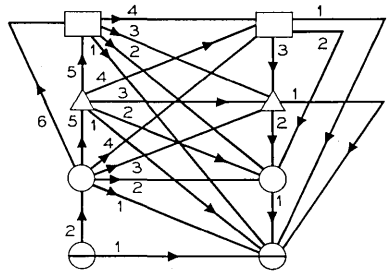


Fig. 3 — Hierarchical directed routing pattern.

Consider the following two rules for generating directed routing patterns in an arbitrary network.

Rule 1—Select any node in the network (usually the destination node) and label all its branches incoming.

Rule 2—Now select a node which has at least one outgoing branch and label all its remaining free branches incoming.* Repeat this rule until all branches in the network have been assigned a direction.

Fig. 5 illustrates the process for a simple 6-node network. The numbers in the node circles represent the order in which rules 1 and 2 are applied, beginning with the heavily circled node that is the destination node for this pattern. Notice that the process is finished in four steps, leaving the two blank nodes with only outgoing branches. We will call nodes of this type (the blank nodes) *originating* nodes, although it is assumed that *calls routing to the destination node can originate at any*

* Free branches are those which have not been given a direction.

node. Nodes with both incoming and outgoing branches will be termed *tandem nodes* (for example, 2, 3, 4). The remaining node, node 1, will be called a *terminating node* and, in this case, it is the destination node for the pattern. Indeed, if the rules remain as stated, there can be only *one* terminating node in any pattern, the destination node. Unless stations are being considered which are multiple-homed, this

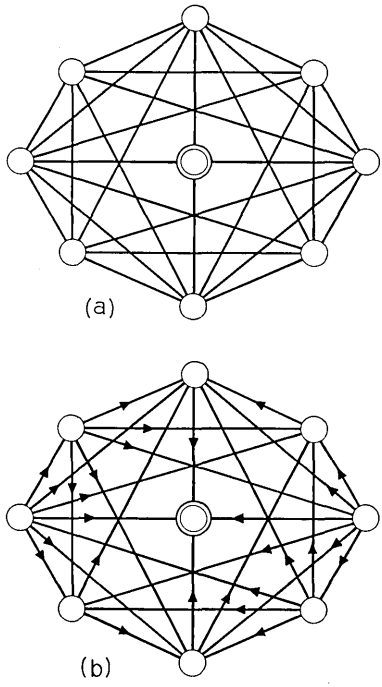


Fig. 4 — A sample pattern.

is not a limitation.* However, to deal with a multiple-homed situation and to allow the proofs of some general theorems, we will remove this restriction by restating rule 1:

Rule 1′ Select any free node in the network and label all its branches incoming.† This rule may be repeated an arbitrary number of times, each application creating a terminating node.

*A station which can be reached from more than one office is said to be multiple-homed. A dual-homed station, for example, can be reached from either of two offices—in this case, we would want both offices (if not connected) to be terminating nodes in the routing pattern.

† A free node is one with no directed branches. Each application of this rule, of course, creates a "trap" for traffic.

Rule 2′ Same as 2.

This revised set of rules will be referred to as *backward production.*

Clearly an analogous process exists in the forward direction and will be called *forward production.*

Rule 1″ Select any free node in the network and label all its branches outgoing. This rule may be repeated arbitrarily, each application creating an originating node.

Rule 2″ Now select a node which has at least one incoming branch and label all its remaining free branches outgoing. Repeat rule 2 until all branches in the network have been assigned a direction.

We show later that backward production is the more useful process for generating telephone network routing patterns.

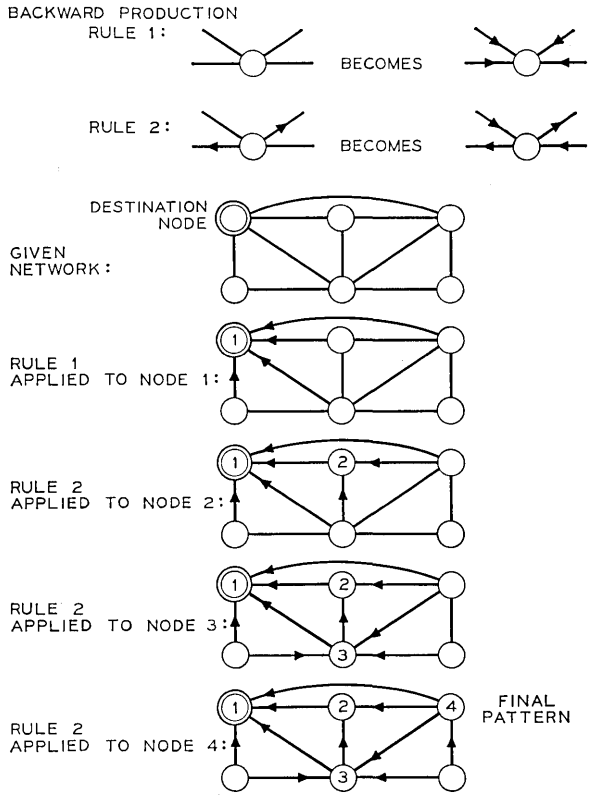


Fig. 5 — Example of use of backward production.

IV. GENERAL THEOREMS

4.1 *Proof of Lemma and Theorems*

We now prove the following lemma and theorems:

*Lemma 1: Any loop-free pattern must contain at least one originating node and at least one terminating node.**

Proof: This can be proven by exhausting the possibilities. Clearly it is not possible to have a network consisting only of originating nodes or only of terminating nodes. The remaining possibilities are:

(*i*) *Only Tandem Nodes*—If A is a tandem node, it must have an outgoing branch to some other node, B. Similarly, B must have an outgoing branch to some node other than A (or we would have a loop), say C. This argument proceeds until all nodes in the network have been considered. The last node to be considered *must* connect to some previous node since it, too, is a tandem node. Such a connection would create a loop.

(*ii*) *Originating Nodes and Tandem Nodes*—If A is a tandem node, its outgoing branches must connect to another tandem node, since no branches can terminate on an originating node. Therefore, the argument presented in *i* can be used here.

(*iii*) *Terminating Nodes and Tandem Nodes*—If A is a tandem node, its incoming branches must originate at another tandem node, say B, since no branches originate at terminating nodes. Since B is also a tandem node, it must have incoming branches from some node other than A (or we would have a loop), say C. Again, this argument proceeds until all nodes in the network have been considered. The last node considered must have an incoming branch from some previously considered tandem node, thus creating a loop.

The remaining two possibilities (terminating the originating nodes only, and all three node types), each contain at least one terminating and one originating node.

Theorem 1: Routing patterns generated by backward production are loop-free.

Proof: Rule 1′ tells us we may create terminating nodes arbitrarily (we must create at least one) in sequential fashion. Since candidates for terminating nodes must have all branches free and these branches

* As this paper was being prepared, the author learned of work by S. L. Hakimi in which Lemma 1 and Theorems 1 and 2 are proved in a more formal fashion. (See Ref. 2.)

are all made incoming, it is not possible to loop through a terminating node, nor can there be branches between terminating nodes. Let A and B be terminating nodes and let C be the first nonterminating node, with a branch to A (and possibly to B), to which we apply rule 2′. Since it is not possible to loop through nodes A and B, we may disregard the branches to these nodes as far as loops are concerned. Then the only branches which can be members of a loop are the remaining (all-free) branches on C. But rule 2′ tells us to make all these branches incoming. Therefore, the only way to loop through node C is to loop through node A (or B). Since it is not possible to loop through A or B, it is not possible to loop through C. Clearly the same argument applies at each stage of the process; the only way to loop through the present node is to loop through some previously considered node, which is not possible. The process ends when all branches have been given a direction. At this point, the originating nodes will be seen to have been created by applying rule 2′ to all the nodes to which they connect. Since it is not possible to loop through originating nodes, the pattern must, indeed, be loop-free.

By a completely analogous proof, it may be shown that the routing patterns generated by forward production are also loop-free.

Now we have two procedures for generating loop-free patterns. The question is: What sort of patterns do they generate? We prove:

Theorem 2: All loop-free routing patterns can be generated by backward production.

Proof: This is proven by induction on Lemma 1. Let N_0 be any arbitrary loop-free routing pattern. Then, by Lemma 1, it must contain at least one terminating node. For generality, assume it contains two, A and B. We will make these nodes evident, leaving a reduced network, N_1. See Fig. 6.

In a blank network (that is, a copy of N_0 without branch assign-

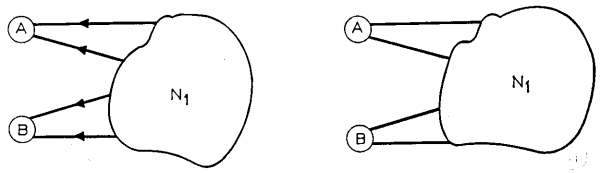


Fig. 6 — Construction of a duplicate of N_0 from a blank network—first stage.

ments), the terminating nodes A and B may be generated by the application of rule 1′.

If we were to remove these nodes and their branches from N_0, we would have left the network N_1, which must also be loop free. (If N_1 is not loop-free, then neither is N_0.) Since N_1 did not contain terminating nodes (all terminating nodes in N_0 were made evident), we must create terminating nodes in the process of discarding nodes A and B and their branches. That is, in the set of nodes to which A and B connect, there must be at least one node which becomes a terminating node if its branches to A and B are removed. In addition, if there is more than one such node, the nodes cannot be connected to each other. (Any such connection would make one of the nodes a nonterminating node.) We will call these terminating nodes, generated by discarding previous nodes and branches, *pseudoterminating*. Let us assume there are two such nodes, C and D, in the network N_1 and place them in evidence. See Fig. 7. In the blank network, the outgoing branches from C and D (and all other nodes) to A and B, were generated when we applied rule 1′ to nodes A and B. If we were now to apply rule 2′ to nodes C and D, in any order, we would generate the nodes C and D in the blank network exactly as they appear in the network N_0.

If we now remove nodes C and D and their branches from N_1, we are left with network N_2, which must also be loop-free. N_2 had contained only tandem and originating nodes (all pseudoterminating nodes in N_1 were made evident). With the removal of branches to C and D, we must therefore create at least one pseudoterminating node in N_2. Let there be two such nodes, E and F, and make them evident. See Fig. 8. In the blank network, the application of rule 1′ to nodes A and B, and of rule 2′ to nodes C and D, assigned all the outgoing branches from nodes E and F (and all other nodes) to nodes A, B, C, and D. Now the application of rule 2′ to nodes E and F in the blank network will properly assign all the incoming branches to nodes E and F, and these nodes will appear as they do in N_0.

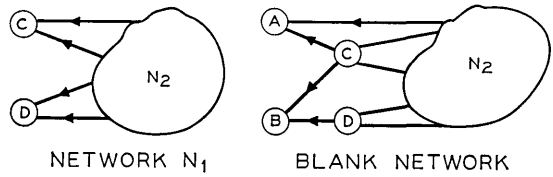


Fig. 7 — Second stage.

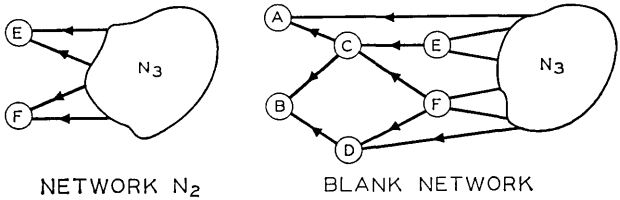


Fig. 8 — Third stage.

This argument may be applied to smaller and smaller networks, each time generating the proper nodes and branch assignments in the blank network. Since the network N_0 is assumed to be finite, the process terminates in K steps with some network, N_K, which contains all the originating nodes in N_0, now isolated (all branches will have been discarded). At this point in our assignments in the blank network, we will find that all branches have been assigned and that we have generated network N_0 by applying rules 1′ and 2′.

Again, an analogous proof will show that all loop-free patterns may be generated by forward production.

It is possible to decide whether or not a given network and routing pattern contains a loop by a procedure which is a variant of that given in theorem 2. (This procedure relies on lemma 1 for its justification.)

Identify all originating and terminating nodes and remove them and all their branches from the graph. Now repeat this step until either:

(*i*) Only branchless nodes remain, or

(*ii*) No originating or terminating nodes can be found. If the network can be reduced to branchless nodes, it is loop free. If not, at the point where no further reduction is possible, the remaining network contains at least one loop.

4.2 *Conclusions from the Theorems*

We can now draw the following conclusions:

(*i*) The class of routing patterns defined by rules 1′ and 2′, or by rules 1″ and 2″, is equal to the class of all loop-free routing patterns.

(*ii*) Therefore, any pattern that can be generated by forward production can also be generated by backward production.

(*iii*) If a pattern is loop free, there is at least one sequence of nodes, to which we can apply backward production, that will generate the

pattern. (A similar statement holds for forward production.) The sequence that will generate the pattern is discoverable by applying the procedure given in theorem 2.

(*iv*) If N_i represents the number of terminating or pseudoterminating nodes made evident in the i^{th} step, the number of ways to generate the pattern N_0, by backward production, is:

$$\text{No. of Ways} = \prod_{i=1}^{K} (N_i\ !)$$

In general, the number of ways to produce a pattern using forward production is unequal to the number of ways using backward production.

V. SYMMETRIC NETWORKS

5.1 *Additional Conclusions*

The theorems and conclusions of Section IV apply equally well to symmetric networks (that is, fully interconnected). In addition, it can be shown that, for symmetric networks, the following is true:

(*i*) There is exactly one originating node and exactly one terminating node in any pattern.

(*ii*) There is only one way to produce a *given* pattern using backward production. (Similarly for forward production.)

(*iii*) If we choose a terminating node and apply backward (or forward) production, we can generate $(N-1)$ distinct loop-free routing patterns to the terminal node. These patterns will all be isomorphic (that is, the same with a relabeling of the nodes). Indeed, there is really only one "abstract" loop-free pattern in a symmetric network, no matter which node is the destination node or what order of choice is applied, since all patterns can be shown to be isomorphic.

5.2 *Routing in Symmetric Networks*

We would now like to examine routing in symmetric networks both as useful in itself and for a bound on routing in incomplete networks. We begin by deriving an expression for the number of K-link routes in a symmetric routing pattern from a given node to the destination node.

Assume we are given an N node symmetric network with the first node as the destination. Also assume forward production is applied to the network, using rule 1 on node N and rule 2 on the nodes $N-1$, $N-2$, $\cdots$ 3, and 2, *in that order*. (Equivalently, we could use backward

production, applying rule 1 to node 1 and rule 2 to nodes 2, 3, 4, $\cdots$ $N - 2$, and $N - 1$, in that order.) Then we may show the following:

Theorem 3: In a symmetric network of N nodes, the number of K-link routes between the Q^{th} node ($2 \leqq Q \leqq N$) and the destination node (node 1) is exactly given by the binomial coefficient $C\binom{Q-2}{K-1}$.

Proof: Consider node N. It will have one branch to the destination node, giving a single one-link path. We may write this as $C\binom{N-2}{0}$. Now, any 2-link path must pass through an intermediate node, of which there are $N - 2$. If we can show that each of the $C\binom{N-2}{1}$ selections of an intermediate node generates exactly one 2-link path, the number of 2-link paths from node N to the terminal node will be $C\binom{N-2}{1}$. Let A be one of the possible intermediate nodes. Since we are using forward production to generate the routing pattern, each successive node considered must have no branches directed toward previously considered nodes and must have exactly one branch directed to each of the nodes not yet considered. Since node A is considered sometime after node N, there must exist a branch from N to A. But A is considered before the destination node (which is last in the process); therefore, there must exist a branch *from A to* the destination node. Thus, there is exactly one 2-link route from node N, through node A, to node 1. This argument is valid for any of the $C\binom{N-2}{1}$ choices for A. Therefore, there are exactly $C\binom{N-2}{1}$ 2-link routes from node N to node 1.

This argument may be generalized for K-link routes. Each K-link route requires $K - 1$ intermediate nodes between node N and node 1. There are $C\binom{N-2}{K-1}$ ways to choose a distinct set of $K - 1$ intermediate nodes. If we let $A_1, A_2, A_3, \cdots A_{K-1}$ be a particular choice of the $K - 1$ nodes, then there must exist exactly one ordering of these nodes which represents the sequence in which rule 2″ was applied. If the ordering is $A_1, A_2, A_3, \cdots A_{K-1}$, node A_1 will have a branch to A_2, which will have a branch to A_3, and so on. Since A_1 is always considered after node N, and node A_{K-1} is always considered before node 1, there will be exactly one K-link path for this choice of $K - 1$

nodes. Since the choice was arbitrary, there are $C\binom{N-2}{K-1}$ K-link routes from node N to node 1.

Now consider node $N-1$. All branches from node N were made outgoing and therefore are of no use to later nodes for the purpose of routing. If we remove node N and its branches from the graph, we are left with an $N-1$ node symmetric network. In this reduced graph, we may consider node $N-1$ as the originating node and node 1 as the destination node. The sequence in which rule 2″ was applied in this graph is the same sequence used in the larger graph. Hence, the argument presented above applies to this network with $N-1$ nodes replacing N nodes. The number of K-link routes is therefore $C\begin{bmatrix}(N-1)-2\\ K-1\end{bmatrix}$. We may proceed to remove node $N-1$ and its branches from the graph, and so on, generating successively smaller symmetric networks and applying the same arguments at each stage. In general, from node Q, the number of K-link routes to the destination is $C\binom{Q-2}{K-1}$, $(2 \leqq Q \leqq N$ and $1 \leqq K \leqq Q-1)$. As a corollary, it can be shown that the *total* number of K-link routes from all nodes in the graph to the destination node is given by $C\binom{N-1}{K}$. That is:

$$C\binom{N-1}{K} = \sum_{Q=2}^{N} C\binom{Q-2}{K-1}$$

where

$$C\binom{Q-2}{K-1} \text{ is zero for } K \geqq Q.$$

It is possible to summarize routing in symmetric networks by using Table I. As an example of the information obtainable from the table, consider a 6-node symmetric network to which we have applied backward production in the order: (rule 1) 1, (rule 2) 2, 3, 4, 5, 6. (See Fig. 9.)

This network will have:

From node 6: one 1-link, four 2-link, six 3-link, four 4-link, and one 5-link route to node 1 (the destination).

From node 5: one 1-link, three 2-link, three 3-link, and one 4-link route to node 1.

And so on, reading node I routes from line I. Reading from $N=6$,

TABLE I—SUMMARY OF ROUTING IN SYMMETRIC NETWORKS

Number of K-Link Routes From Node I to Node 1 (Terminal Node)	Number of Routes											
I	1 Link	2 Link	3 Link	4 Link	5 Link	6 Link	7 Link	8 Link	9 Link	10 Link	11 Link	N
2	1											
3	1	1										2
4	1	2	1									3
5	1	3	3	1								4
6	1	4	6	4	1							5
7	1	5	10	10	5	1						6
8	1	6	15	20	15	6	1					7
9	1	7	21	35	35	21	7	1				8
10	1	8	28	56	70	56	28	8	1			9
11	1	9	36	84	126	126	84	36	9	1		10
12	1	10	45	120	210	252	210	120	45	10	1	11
		1 Link	2 Link	3 Link	4 Link	5 Link	6 Link	7 Link	8 Link	9 Link	10 Link	N
		Total Number of Routes										Total Number of K-Link Routes in a Network of N Nodes

there will be a total of five 1-link routes from all nodes to node 1, a total of ten 2-link routes, and so on.

These numbers represent the maximum numbers of K-link routes in an arbitrary (not necessarily symmetric) 6-node network. This follows from the fact that we may generate the arbitrary network by removing branches (and therefore routes) from the corresponding symmetric network.

VI. HEURISTIC PROCEDURES FOR ARBITRARY NETWORKS

How does one go about choosing a "good" routing pattern to a given terminal node in an arbitrary network? We can begin by making the following observation.

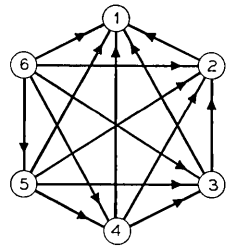


Fig. 9 — Six-node symmetric network.

Since a practical pattern defines routes to *one* node, the destination, it is reasonable to require "good" routing patterns to have only one terminating node type, the destination node. This means we may apply backward production as defined by rules 1 and 2; therefore, we cannot expect to generate *all* loop-free routing patterns (which require rule 1′ in place of rule 1). This is not a limitation unless we are dealing with multiple-homed stations. We will ignore this latter case, although the procedures we discuss can be generalized to deal with multiple homing.

Now, what is meant by a "good" routing pattern? One with the lowest average blocking from all nodes to the destination node? A pattern which minimizes blocking from a *selected* node to the destination? One with the smallest average route length? A pattern with the maximum total number of routes?*

To the author's knowledge no algorithm exists which will guarantee

* There is, of course, the larger problem of designing a network which realizes all of these and is, at the same time, rugged, economical, and so on, as described in the introduction. This is a complex problem; its very statement is difficult and has been the subject of intensive study, (See Refs. 3, 4 and 5.)

any of these criteria. However, it is possible to approach the last two criteria by using a heuristic procedure which will generate patterns with large numbers of short routes, and which also has the virtue of assigning orders of choice to the branches.

6.1 *Generating Patterns with Many Short Routes*

Consider the following method for applying rules 1 and 2 to an arbitrary network:

(*i*) Select the destination node as the first node and apply rule 1, labeling all the branches incoming. Label the originating ends of each of these branches the first choice out of the respective nodes.

(*ii*) Now, in the set of nodes to which the destination node connects (these will be called "predecessors" of the destination node), select any node and apply rule 2, labeling its free branches incoming. Label the originating ends of these branches first choice out of the respective nodes, if possible; or, if a first choice already exists (from step *i*) label the branch second choice.

(*iii*) Continue step *ii*, choosing nodes only from the predecessors of the destination node; each time, label the branches n plus first choice out of the node at the originating end, where n choices already exist. Continue until all the predecessors of the destination node have had rule 2 applied to them.

(*iv*) Consider the set of nodes which has outgoing branches to any node (or nodes) which are predecessors of the destination node.† These may be thought of as second level predecessors of the destination node. Apply rule 2 to these nodes until they have been exhausted (or until you are exhausted, whichever comes first), each time labeling branches the n plus first choice out of the node in which they originate.

(*v*) Identify the third level predecessors of the destination node, and so on. Continue the process until every branch in the graph has a direction and order of choice out of the node from which it originates.

Fig. 10 gives an example of the procedure, which is tedious to describe, but easy to perform.

At this point, we can observe that all the paths from the K^{th} level predecessors of the destination node have at least K links. We prove the following theorem:

† The predecessors of any node (or nodes) can be identified without reference to branch directions. In this procedure, a predecessor of node A is any node connected to node A by an (as yet) undirected branch. If we are seeking the predecessors of a group of nodes, branches between nodes in the group are ignored.

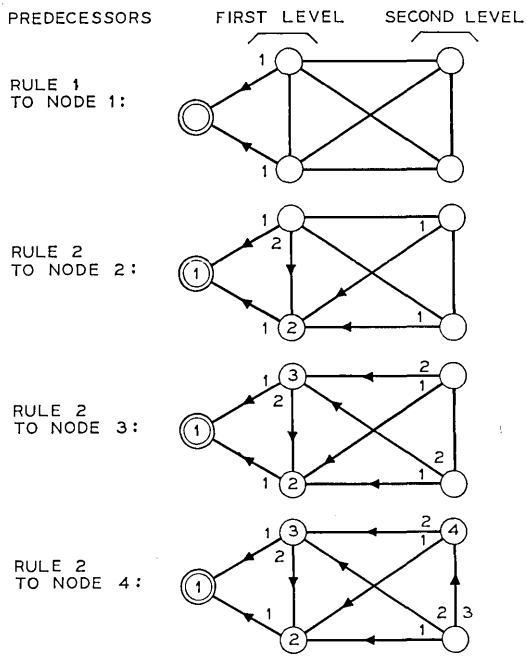


Fig. 10 — Deriving order of choice in the backward production process.

Theorem 4: The given procedure creates the maximum number of 1-link and 2-link routes.

Proof: Clearly, there is no way to create more 1-link paths to the destination node than to label all its branches incoming. Consider the first level predecessors of the destination node. Among their free branches, they may have branches to each other, and branches from second level predecessors.

Each time we label a branch between first level predecessors we create one 2-link path to the destination. This is true regardless of the direction given to the branch. Hence, the number of 2-link paths created this way is fixed, and is exactly the number of branches between first level predecessors. If we now discard the branches between first level predecessors and consider the reduced graph, it is clear that the way to get the maximum number of 2-link paths is to label every free branch, on every first level predecessor, incoming. But this is exactly the effect of the given procedure. Branches that do not connect first level predecessors remain free until rule 2 is applied to the node; then they are all labeled incoming. It follows that the total

number of 2-link paths created is fixed, and is equal to the number of free branches on all first level predecessors after the application of rule 1 to the destination node. The order in which rule 2 is applied to these nodes has no effect on the number of 2-link routes.

It might seem that this theorem can be extended to show that the procedure produces the maximum number of K-link routes ($K > 2$), subject to the fact that K-1 link, K-2 link, . . . , 2-link, and 1-link routes have been maximized. Unfortunately, one need go no higher than 3-link routes to find a counterexample as shown in Fig. 11.

The heuristic procedure can be improved by eliminating the arbitrary choosing of nodes in step *ii* and in later steps. That is, having identified the N^{th} level predecessors of the destination node, we apply rule 2 to these nodes in a particular order.

6.2 *Choosing N^{th} Level Predecessors*

We suggest this revised heuristic procedure for choosing among N^{th} level predecessors:

(*i*) Arrange the graph to show the various level predecessors in stages. An example is Fig. 12, where higher and higher level predecessors are encountered as we progress from left to right.

(*ii*) Direct all branches between stages toward the destination node. (See Fig. 12.)

(*iii*) Now consider the first level predecessors, nodes 2, 3, and 4.

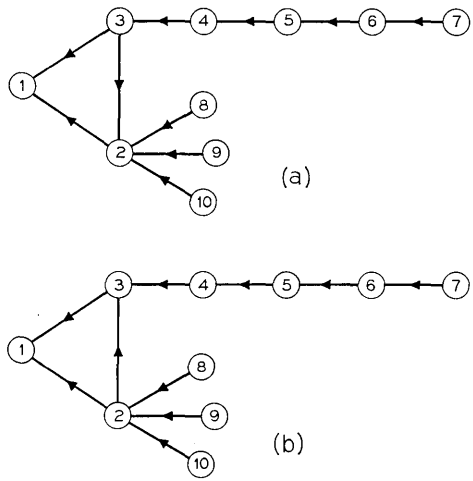


Fig. 11 — Counterexample. (a) Pattern generated by heuristic procedure; number of 3-link routes: 2. (b) Pattern with maximum number of 3-link routes; number of 3-link routes: 4.

For each of these nodes, we compile two figures, the number of routes provided from the node to the destination, and the number of nodes served by this node. Node B is said to be served by node A if there exists at least one directed path from B to A. Node 2, for example, serves nodes 5 and 9, while node 7 serves none. We take the difference of these two numbers (nodes served minus routes provided) and use the resulting number as a measure of the need for additional routes. For Fig. 12b these numbers may be tabulated as follows:

Node	No. Nodes Served	No. Routes Provided	Difference
2	2	1	1
3	3	1	2
4	3	1	2

(*iv*) Now choose the lowest of the difference numbers and apply rule 2 to the corresponding node. In this example, node 2 is the choice and we label all its branches incoming. (Presumably, it needs the least number of additional routes.)

(*v*) Node 2 is now removed from consideration and we may restate the table for nodes 3 and 4, adding the routes picked up by the branches directed into node 2:

Node	No. Nodes Served	No. Routes Provided	Difference
3	3	2	1
4	3	2	1

In this case, we have equality and so choose node 3 arbitrarily. Node 4 is, then, the last node in the process and the result is shown in Fig. 12c.

(*vi*) We now move one stage to the right and consider second level predecessors:

Node	No. Nodes Served	No. Routes Provided	Difference
5	1	1	0
6	2	2	0
7	0	4	−4
8	1	4	−3

This suggests that node 7 is least in need of additional routes and we may label all its branches incoming. Restating the table two more

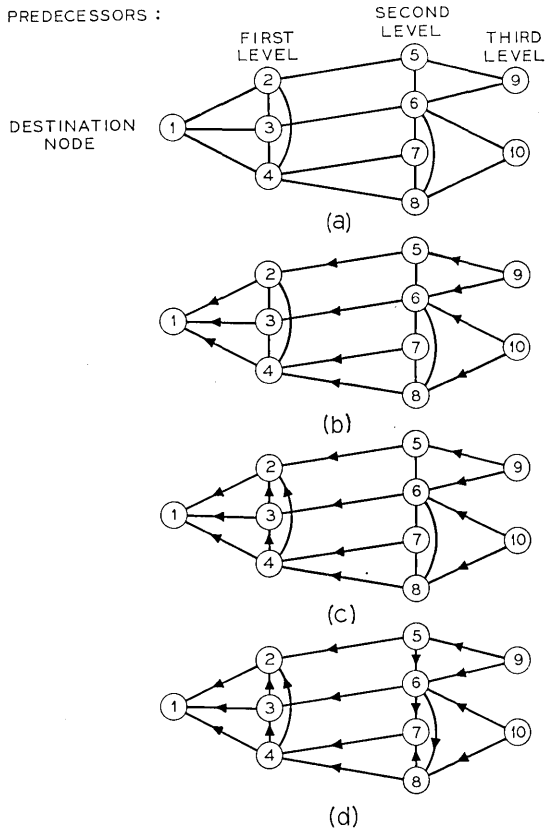


Fig. 12 — Example of heuristic procedure.

times, we obtain node 8 next and, finally, node 6. The result is shown in Fig. 12d.

Node	No. Nodes Served	No. Routes Provided	Difference
5	1	1	0
6	2	6	−4
8	1	8	−7
5	1	1	0
6	2	14	−12

To obtain an order of choice for the branches, we simply apply the heuristic procedure for generating patterns with large numbers of

short routes in node order 1, 2, 3, 4, 7, 8, 6, and 5. (See Section 6.1.) The result, identical to that in Fig. 12d, is shown in Fig. 13.

This method yields routing patterns in which the average path length is short and the total number of routes large. It is, however, not infallible, and counterexamples can be generated—networks in which the process leads to neither a minimum average path length nor a maximum total number of routes.

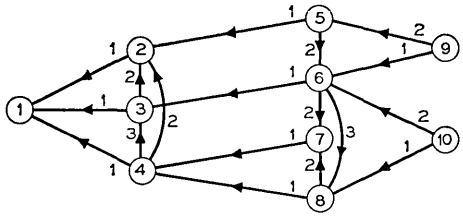


Fig. 13 — The complete routing pattern to node 1. Apply backward production in the order: 1, 2, 3, 4, 7, 8, 6, 5.

VII. SUMMARY AND CONCLUSIONS

This paper discusses methods for generating loop-free directed routing patterns and for detecting the presence of loops in arbitrary patterns. The heuristic procedures suggested seem to yield useful patterns for the size network that can be considered by hand; moreover, they are clearly programmable, thus allowing us to deal with large networks.

The procedures and theorems we present are not addressed to the problem of achieving optimum traffic handling abilities of communication networks. They are, however, a preliminary step to such examinations and, hopefully, present an orderly and useful way of looking at the process of routing as it is currently practiced. These theorems and procedures suggest ways of modifying present routing practices which may be fruitful to explore.

REFERENCES

1. Weber, J. H., unpublished work.
2. Hakimi, S. L., "On the Degrees of the Vertices of a Directed Graph," J. Franklin Inst., *279*, No. 4 (April 1965), pp. 290–308.
3. Wernander, M. A., "Systems Engineering for Communications Networks," talk at IEEE summer general meeting and nuclear radiation effects conference, Toronto, Ont., Canada, June 16–21, 1963.
4. Beneš, V. E., unpublished work.
5. Beneš, V. E., "Programming and Control Problems Arising from Optimal Routing in Telephone Networks," talk at the First International Conference on Programming and Control, USAF Academy, Colorado, April 15–16, 1965.

Diode and Transistor Self-Analogues for Circuit Analysis

By BERNARD T. MURPHY

(Manuscript received September 25, 1967)

A new method of circuit analysis based on the time-scaling of actual circuits has been proposed. Audio-frequency self-analogues of microwave frequency transistors can be constructed using charge control theory, and these accurately model transistor performance in the active region. Scaling of storage times in diodes and transistors requires multiple-lump modeling. The multiple lump model developed by Linvill is reformulated here on the basis of an analogy between charge density in the semiconductor and charge density in the model, rather than between carrier density and voltage. Only two parameters, time constants corresponding to lifetime and a diffusion transit time in the semiconductor, need be specified in the reformulated model. This simplified multiple-lump model should be generally useful for device calculations. We describe a diode self-analogue which is an exact physical realization of the multiple-lump model. Separation of active and saturation region stored charges can be achieved in a transistor self-analogue, so that a single-lump model can be used for the active, and a multiple-lump model for the saturation region stored charges.

I. INTRODUCTION

A new approach to circuit analysis has been poposed which allows high-frequency circuits to be characterized using simple low-frequency models.[1] With this approach, nanosecond diodes and transistors can be slowed down to audio frequencies and interconnected in audio frequency breadboard models of the high-frequency circuits. Thus, high-frequency circuits can be evaluated and optimized with the convenience afforded by low-frequency breadboard techniques.

According to charge-control theory,[2] the frequency and transient responses of diodes and transistors are determined by the charges stored within the devices. Nanosecond diodes and transistors can be slowed down to millisecond models simply by multiplying their stored charge by a factor of 10^6 or some other convenient value. Charge

storage in the devices can be classified broadly as terminal voltage dependent (fixed charge in depletion layers) and terminal current dependent (charge in transit). The former can be multiplied by connecting large capacitors between the device terminals, as described by Levine.[3] The latter can be multiplied by using small resistors as current sensors in series with device terminals, and using the voltage developed across these resistors, suitably amplified, to cause charge storage in capacitors connected to the device.[1] Models thus constructed have given excellent results for transistors operated in their active regions.[4]

The models also give time-scaled storage times when representing diodes or transistors operated in their saturation regions, but the values may be in error by a factor of two or more. One difficulty is that the charge-control model does not provide any means for representing the distribution of charge throughout bulk semiconductor regions. It is shown here that by replacing the storage capacitors in the model by resistance-capacitance networks, an exact physical realization of the multiple-lump Linvill model can be obtained. A second difficulty in the case of the transistor is that the time constants for charge storage in the saturation region and in the active region are different in general. Section 3 describes means for overcoming this difficulty.

II. DIODE SELF-ANALOGUES

2.1 *Charge-Control Self-Analogue*

Fig. 1 shows a simple self-analogue of a diode. The diode itself, D, is its own dc model. Capacitor C_D is used to multiply depletion layer

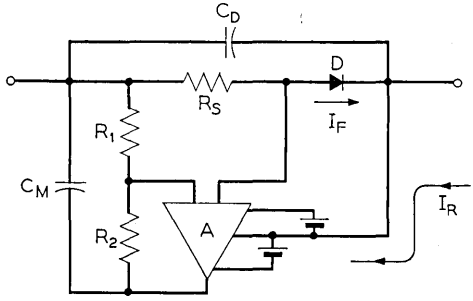


Fig. 1 — Diode self-analogue with scaling of charge storage.

stored charge in D, with,

$$C_D = KC_J \tag{1}$$

in which C_J is the junction capacitance of the diode. C_D would ideally have the same voltage dependence as C_J. (A large area junction might possibly be used, with a battery to avoid forward bias).

Resistor R_S, amplifier A (whose gain is R_2/R_1) and capacitor C_M are used to multiply minority carrier storage effects. Minority carrier charge storage Q_M within the diode itself is given by

$$Q_M = I_F\tau \tag{2}$$

in which I_F is the forward current in the diode and τ is an effective lifetime which will usually be dominated by the bulk minority carrier lifetimes in the P and the N regions. The charge Q'_M stored in C_M in the model is

$$Q'_M = I_F R_S\left(\frac{R_2}{R_1}\right)C_M \tag{3}$$

in which parameters in the model are chosen to give

$$R_S\left(\frac{R_2}{R_1}\right)C_M = K\tau. \tag{4}$$

K is the desired time-scaling factor.

During turn-on and turn-off transients, Q_M in the diode and Q'_M in the model obey the charge control equations

$$\frac{dQ_M}{dt} = I - \frac{Q_M}{\tau} \tag{5a}$$

$$\frac{dQ'_M}{dt} = I - \frac{Q'_M}{K\tau} \tag{5b}$$

in which in which I is the terminal current. In the model, a current I_F which is proportional to Q'_M, flows in the actual diode at all times, maintaining the correct bias voltage on the diode at all times (assuming that series resistance gives negligible voltage drops).

Provided that amplifier A has good common-mode rejection, the diode voltage has negligible influence on the charge stored on C_M. Fig. 2 shows a slightly modified version of the analogue in which the full diode voltage appears across the plates of C_M. In this modified version stored charge on C_M is the analogue of both the depletion

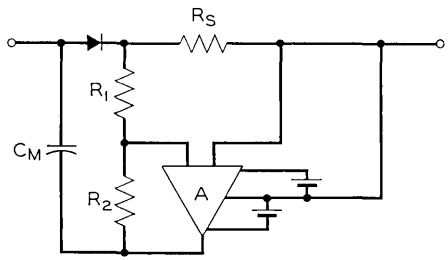


Fig. 2 — Alternative diode self-analogue with scaling of charge storage.

layer stored charge and the minority carrier stored charge; therefore, capacitor C_D has been eliminated. This version cannot be used if non-linear depletion layer effects are to be represented.

The analogue of Fig. 1 is satisfactory whenever the charge control equation (suitably time-scaled) satisfactorily describes the transient behavior of the actual diode. This is true in at least two important types of diode: epitaxial diodes with epitaxial layers which are thin compared with a diffusion length, and diodes formed in integrated circuits using the emitter-base junction of a transistor whose base-collector junction is shorted. In the second type of diode, minority carrier storage is confined to the thin base layer.

2.2 *Multiple Lump Analogues*

Although thin epitaxial layers are generally used for high speed switching diodes, such diodes are often so heavily gold doped that the diffusion length for minority carriers is even less than, or at least comparable with, the epitaxial layer thickness. In this case diffusion delays comparable with diode storage times occur during turn-off. The charge-control equation (5) is not satisfactory then, and the simple model shown in Fig. 1 is inadequate. For example, in the extreme case of a diode formed on a uniformly doped substrate, and for a reverse current equal to the forward current, Kingston's analysis,[5] which includes diffusion delays, gives a storage time $t_s = 0.25\ \tau$, and a fall time $t_f = 0.6\ \tau$, whereas equation (5) gives $t_s = 0.7\ \tau$ and $t_f = 0$.

Diffusion delays can be taken into account by using a multiple-lump Linvill model.[6] Figure 3 shows a diode self-analogue which can be made an exact physical realization of such a model. This self-analogue is justified later by its node equations which are expressed

in terms of the stored charge at each node, rather than the voltage:

$$\text{Node 1.} \qquad I - I_F = I - \frac{Q_1}{rAC} = \frac{Q_1 - Q_2}{RC} + \frac{dQ_1}{dt}$$

$$\text{Node X.} \qquad \frac{Q_{X-1} - Q_X}{RC} = \frac{Q_X - Q_{X+1}}{RC} + \frac{dQ_X}{dt} + Q_X \cdot \frac{G}{C}$$

$$\text{Node N.} \qquad \frac{Q_{N-1} - Q_N}{RC} = \frac{dQ_N}{dt} + Q_N \cdot \frac{G}{C} + \frac{Q_N}{R_S C}.$$

In this case the output of the amplifier is double-ended; each output has the polarity of the input opposite to which it is drawn.

The Linvill model is based on an analogy between carrier density in the diode and voltage in an r-g-c line. The well-known continuity and current equations for the uniformly-doped semiconductor region adjacent to the transition region in a diode are

$$\frac{\partial(N - Ne)}{\partial t} = -\frac{N - Ne}{\tau} - \frac{\partial I/qA}{\partial x} \tag{6}$$

$$\frac{I}{qA} = -D \frac{\partial(N - Ne)}{\partial x} \tag{7}$$

in which

$N - Ne$ = carrier density in excess over equilibrium density
τ = lifetime
I = current
q = particle charge
A = cross sectional area of diode
D = diffusion constant
Drift current is assumed negligible.

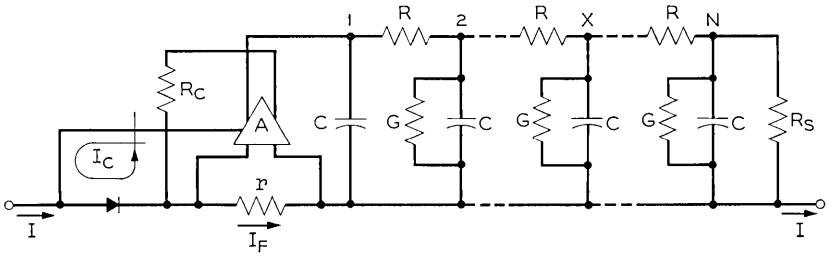


Fig. 3 — Physical realization of Linvill model as a diode self-analogue.

The voltage-current equations for the analogous r-g-c transmission line are

$$\frac{\partial I}{\partial X} = -G'V - C'\frac{\partial V}{\partial t} \tag{6a}$$

$$\frac{\partial V}{\partial X} = -R'I \tag{7a}$$

in which

		Analogue of
V	= voltage	$N - Ne$
I	= current	I
G'	= conductance (combinance) per unit length	qA/τ
C'	= capacitance (storance) per unit length	qA
R'	= resistance (1/diffusance) per unit length.	$1/qAD$

The analogy can be expressed in a simpler, dimensionally-consistent way if the equations are written in terms of charge per unit length, Q', in both cases. Then equations 6 and 7 become

$$\frac{\partial(Q' - Q'e)}{\partial t} = -\frac{Q' - Qe'}{\tau} - \frac{\partial I}{\partial X} \tag{8}$$

$$I = -D\frac{\partial(Q' - Qe')}{\partial X} \tag{9}$$

and Equations 6a and 7a become

$$\frac{\partial I}{\partial X} = -\frac{G'Q'}{C'} - \frac{\partial Q'}{\partial t} \tag{8a}$$

$$\frac{\partial Q'}{\partial X} = -R'C'I. \tag{9a}$$

Analogous quantities are now

Diode	r-g-c line
$Q' - Qe'$	Q'
I	I
τ	C'/G'
D	$1/R'C'$

The length of the r-g-c line is equal to the length of the semiconductor region which it represents. Redundancy caused by introducing area A in Equations 6 and 7 has been removed and only two param-

eters of the r-g-c line need be specified. The elements $1/R$, G, and C correspond to the diffusance, combinance, and storance in the Linvill model.

Fig. 4 shows the lumped version of the r-g-c line. Its node equations are:

Node 1. $$I = \frac{Q_1 - Q_2}{RC} + \frac{dQ_1}{dt} + \frac{Q_1 G}{C}$$

Node X. $$\frac{Q_{X-1} - Q_X}{RC} = \frac{Q_X - Q_{X+1}}{RC} + \frac{dQ_X}{dt} + \frac{Q_X G}{C}$$

Node N. $$\frac{Q_{N-1} - Q_N}{RC} = \frac{dQ_N}{dt} + \frac{Q_N G}{C} + \frac{Q_N}{R_S C}$$

These are expressed in terms of charge stored on the capacitors connected to each node rather than node voltages. It may be assumed that the line and diode have been cut into equal lengths δx, so that

$$C/G = \tau \tag{10}$$

$$RC = \delta x^2 R'C' = \delta x^2/D. \tag{11}$$

The RC product in equation 11 is the analogue of a diffusion transit time between sections of the diode.

Resistor R_s can be used to represent a surface with recombination velocity v_s (or a collector junction in which carriers travel with scatter limited velocity v_s). Then R_s is given by

$$\frac{\delta x}{R_s C} = v_s \ . \tag{12}$$

The node equations for the lumped network are identical with the node equations for the diode self-analogue represented by Fig. 3, provided $rA = 1/G$. The charge distribution in the self-analogue is then the same as that in the Linvill model under the same boundary

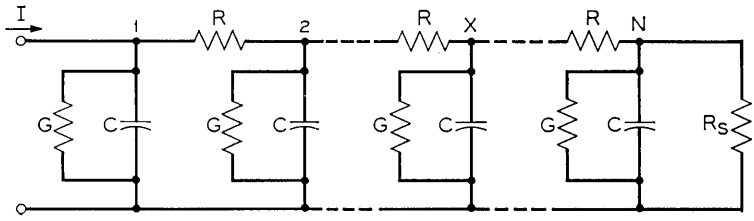


Fig. 4 — Multiple-lump model.

conditions. If the self-analogue is to operate at low frequencies, appropriate scaling factors are needed. We discuss an example in the appendix and in Section 2.3. It remains to be shown that the correct diode voltage is maintained at all times.

In the actual diode, the applied junction voltage is

$$V_J = \frac{kT}{q} \ln \frac{Q_1}{Qe} \tag{13}$$

in which

Q_1 = charge in lump closest to the junction
Q_e = equilibrium charge in that lump.

In the self-analogue, neglecting internal resistance, and assuming that r is chosen to be small, the analogue diode voltage is:

$$V_{JA} = \frac{kT}{q} \ln \left[\frac{I_F + I_C}{I_{\text{sat}}} + 1 \right].$$

With I_C being the current so designated in Fig. 3,

$$V_{JA} = \frac{kT}{q} \ln \left[\frac{I_F\left(1 + \frac{rA}{R_C}\right)}{I_{\text{sat}}} + 1 \right]$$

$$= \frac{kT}{q} \ln \left[\frac{\left(1 + \frac{rA}{R_C}\right) \frac{Q_1}{rAC}}{I_{\text{sat}}} + 1 \right]. \tag{14}$$

Bearing in mind that Q in the model is the analogue of $Q - Qe$ in the diode, (13) and (14) are identical if

$$R_C = rA \Big/ \left(\frac{rACI_{\text{sat}}}{Qe} - 1 \right). \tag{15}$$

Rather than evaluate (15) directly, it is easier to proceed as follows. If R_C is chosen correctly for one condition, (13) and (14) show that the diode voltage will be correct under all conditions. Under dc conditions it is required that the full current I should flow in the diode. The current through R_C should therefore replace that lost through the dc resistance of the rGC network, and R_C should be set equal to this dc resistance.

2.3 *Numerical Example*

Figure 5b shows a two-lump self-analogue of the typical diode shown in Figure 5a. Numerical values for the parameters are derived

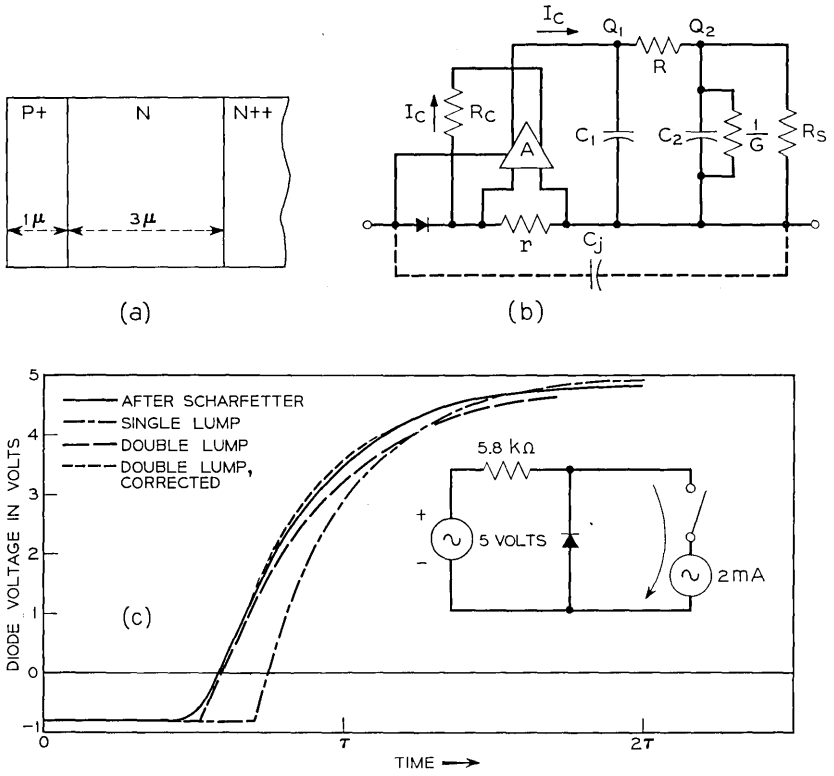


Fig. 5 — (a) Typical epitaxial diode. (b) Model of epitaxial diode. (c) Diode circuit and response curves.

in the appendix, where the behavior of the two-lump analogue under transient conditions also is described. Figure 5c shows the behavior of the actual diode, a single-lump and a double-lump model in the circuit shown inset. This circuit gives equal forward and initial reverse currents, and 5 volts reverse bias when the diode is switched off.

The solution for the diode was provided by D. L. Scharfetter from an exact solution of the semiconductor equations using the well-tested procedures that he and Gummel developed.[7] The single-lump model gives only a rough approximation to the actual diode and cannot be adjusted to give reasonable agreement because of its incorrect storage time. The two-lump model, as first calculated from the diode parameters is still not in good agreement, but a 20 per cent reduction in the assigned value of the junction capacitance gives ex-

cellent results. This agreement is, in fact, better than might be expected and results from compensating errors.

Transient calculations for the three-lump model give a storage time of 0.44τ. For the infinite lump model the storage time is presumably shorter. However, current-dependent stored charge in the "depletion" region was found by Scharfetter to be a significant proportion of the total. Also, the "depletion" layer capacitance is very large in the storage regions. Both effects lead to a longer storage time, and compensate for the over-estimation resulting from representing the epitaxial layer by only two lumps.

III. TRANSISTOR SELF-ANALOGUES

3.1 *Transistor Operated in Active Region*

Figure 6 shows the simplest way of time-scaling a transistor. Charge stored on C_M is the analogue of control charge stored in the transistor. Capacitors connected between B and C, and between B and E can obviously be used to represent fixed depletion layer capacitance. They have been omitted from the diagram for the sake of simplicity and will not be discussed further.

The analogy holds even under base-widening conditions[8] and in saturation provided that the control charge recombines everywhere within the transistor according to a single lifetime. In that case,

$$I_B = \frac{Q}{\tau_Q} \tag{16}$$

in which

I_B = base current
Q = in-transit control charge
τ_Q = lifetime.

In the analogue r, R_1, R_2 and C_M should satisfy

$$r\left(\frac{R_2}{R_1}\right)C_M = K\tau_Q \, . \tag{17}$$

The distribution of controlled and control charge in high-frequency, double-diffused transistors is quite complicated and does not lend itself well to separation into "base" stored charge, "collector" stored charge, or even into "current" controlled charge and "voltage" controlled charge. This can be seen from the results of numerical analysis of charge distribution in such transistors, as given by Gummel.[7]

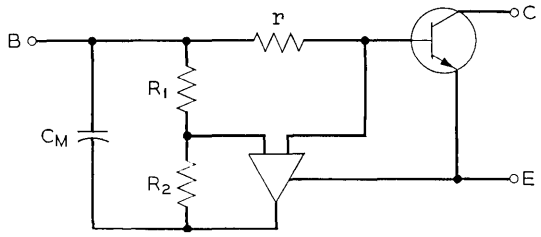


Fig. 6 — Transistor self-analogue.

To a first approximation however, the "current dependent" control charge in an npn transistor consists of (*i*) a component of base charge Q_B associated with electrons in transit through the base, and (*ii*) a component of base charge Q_c associated with charge in transit through the collector transition region. Recombination current associated with Q_c is very small, and equation 16 is not applicable in the sence described above. However, since

$$Q_B = I_B \tau_B \,, \tag{18}$$

and

$$Q_c = I_c \frac{t_c}{2} = I_B h_{FE} \frac{t_c}{2} \tag{19}$$

in which

τ_B = effective base lifetime
I_c = collector current
t_c = transit time through collector depletion layer, equation 16 is still applicable provided that τ_Q is interpreted as

$$\tau_Q = \tau_B + \frac{h_{FE} t_c}{2}. \tag{20}$$

In spite of the difficulties in modeling described above, the simple model illustrated in Fig. 6 has been shown to give a remarkably accurate representation of transistor operation in the active region.[4]

The multiple-lump Linvill model cannot be used to represent diffusion delay in the base of the transistor in the same way that it was used for the diode. Suppose lump 1 represents the base section closest to the emitter, lump N that closest to the collector. Emitter junction voltage should be related to the charge stored on lump 1 if the effects of emitter transition region storage on high frequency

response is to be correctly reproduced; collector current should be related to charge stored on lump N, and the two requirements conflict with one another. Two-pole representation of the transistor can be obtained if necessary, however, using two amplifiers as Fig. 7 shows.[1]

3.2 *Transistor Operated in Saturation Region*

When transistors operate in their saturation regions, excess control charge is stored in the device. Excess control charge is also stored in the model because of the excess base current. But equation 16 is not generally valid in this region because the majority of the excess control charge in double-diffused transistors is stored in the collector region in which the lifetime ordinarily differs from that of the base. Also, because the collector region is much thicker than the base, a multiple lump model is usually needed to represent the charge distributed throughout the collector region even though a single lump model is satisfactory for the base.

If the primary problem is that the collector life-time τ_c is not equal to τ_B, then the model shown in Fig. 8 can be used, in which

$$R' \cdot \frac{R_2}{R_1} \cdot C_M = \tau_Q \tag{21}$$

$$R'' \cdot \frac{R_2}{R_1} \cdot C_M = \tau_c \, . \tag{22}$$

In this model, two time constants are obtained with a single operational amplifier. The simplicity of the model is, however, achieved at the cost of loss of accuracy in the dc collector voltage in saturation.

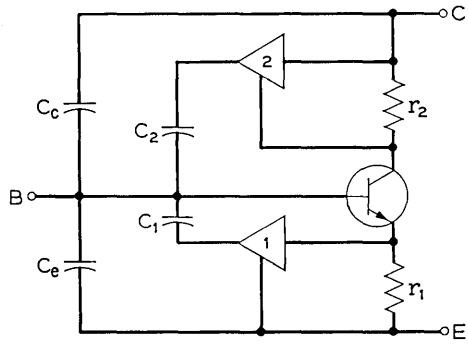


Fig. 7.— Two-pole transistor self-analogue.

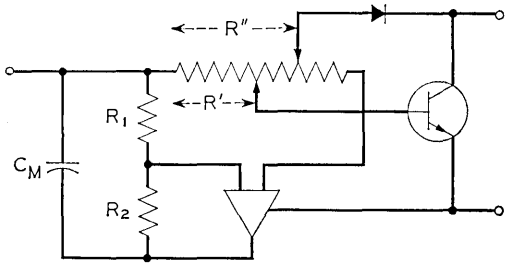


Fig. 8 — Transistor self-analogue with different time constants for active and saturation region storages.

As in the case of diodes, charge distribution in the collector layer of epitaxial transistors in saturation is likely to be troublesome. For example, an npn transistor which gives a storage time of 5 ns (measured with $I_{BF} = I_{BR} = I_C$) has an effective lifetime of 7 ns according to charge control theory. This, however, gives a diffusion length of about 3 μm (with $D = 10$ cm^2/sec), which is about equal to typical epitaxial layer thicknesses, and is inconsistent with the assumption of charge-control theory that charge is stored close to the junctions. Digital programs for circuit analysis commonly use the Ebers-Moll model,[10] which uses a similar assumption. Predicted storage times are too long and current fall times are too short in this situation. A multiple-lump model similar to that proposed for the diode is needed for more accurate representation of storage and fall time. In this case, it is necessary to represent simultaneously (*i*) active region storage with a single-lump model and (*ii*) saturation region storage with a multiple-lump model.

In the model shown in Fig. 9, active and saturation region storages are separated by the following means. The combination R_1 , A_1 , C_1 represents active region storage, R_2 , A_2 and its associated $R - G - C$ network represent saturation region storage. Active region storage which would otherwise occur because of base current flowing in R_2 is cancelled by feedback via R_4 . Thus, the feedback current I_f in R_4 , which flows only in R_2' because of the ground connection of amplifier A_1 , is

$$I_f = \left(\frac{I_e R_1 A_1}{R_4}\right) \tag{23}$$

in which I_e = emitter current.

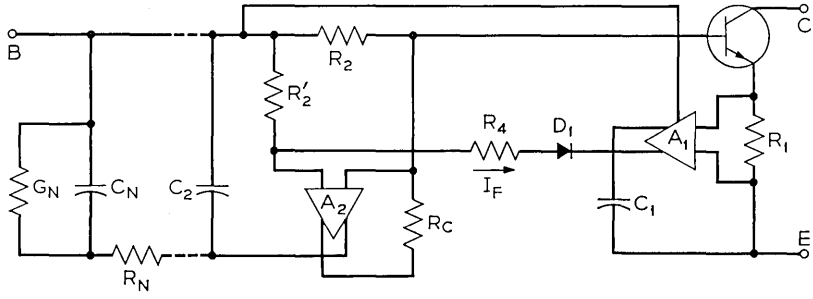


Fig. 9 — Separation of active and saturation region storage.

Diode D_1 is chosen to compensate for the forward bias voltage across the base-emitter junction of the transistor, and voltages across R_1, R_2, and R_2' are designed to be negligible. Then if $R_4/R_1A_1 = h_{FE} + 1$ the desired cancellation will be achieved. Cancellation can only be partial since h_{FE} depends on i_e.

CONCLUSIONS

A new technique of circuit analysis, by which high-frequency circuits can be evaluated and optimized using simple audio frequency breadboard techniques has been demonstrated. It is based on the use of audio frequency self-analogues of diodes and transistors, which can be formed by multiplying the stored charge in the devices by some suitable factor such as 10.[6] Self-analogues based on charge-control models can satisfactorily represent (*i*) the base region of a transistor and (*ii*) diodes and transistors formed in epitaxial layers which are thin compared with a diffusion length. The latter condition is not satisfied by most switching diodes and transistors. Self-analogues can be constructed which are exact physical realizations of the multiple-lump Linvill model. These can be used to represent diodes and transistors formed on epitaxial layers of arbitrary thickness.

APPENDIX

Specific Design Example

Figure 5a shows an epitaxial diode with typical dimensions. Other parameters used in this section are:

Lifetime, $\tau = 3$ns
Diffusion constant, $D = 10$ cm^2/sec

Epitaxial layer doping $= 10^{17}/\text{cm}^3$
Surface concentration of diffused layer $= 10^{20}/\text{cm}^3$.

The effective recombination velocity at the epitaxial interface can be expected to be low because outdiffusion from the substrate creates a built-in field which keeps minority carriers away from the interface. 1000 cm per second is used as an illustration.

Figure 5b shows the two-lump self-analogue of the diode. We will assume that each of the lumps represents half of the epitaxial layer, so that $C_1 = C_2$. With a scaling factor of 10^6:

$$\frac{C_1}{G} = K\tau = 10^6 \times 3 \times 10^{-9} = 3 \text{ ms}$$

$$RC_1 = K\frac{\delta \times 2}{D} = 10^6 \times \frac{(1.5 \times 10^{-4})^2}{10} = 2.25 \text{ ms}.$$

Both r and A can be arbitrarily chosen. A gain of 100 and resistance of a few ohms, say 5Ω, are convenient values to use in practice. Since $rA = 1/G$, this gives $1/G = 500\Omega$, and $C_1 = 3ms$ $G = 6\ \mu F$, and $R = 2.25ms/C_1 = 275\Omega$. Finally, equation 12 with a scaled value for saturation velocity gives $R_S = 25{,}000\Omega$, which is too high to have a significant effect on the model and will be neglected.

Current-dependent charge storage will also occur in the p-layer and the depletion layer. The p-layer is typically diffused and a charge stored in it will lie close to the junction. Both effects could therefore be represented by an additional capacitor in parallel with C_1, with rA reduced in value to $\gamma\ r\ A$, in which γ is the efficiency of injection into the n-layer. This complication was not introduced into the present model.

The behavior of the two lump model was calculated assuming a steady-state forward-bias current I_f followed instantaneously at $t = o$ by a reverse bias current I_r. Solutions for Q_1 and Q_2 during storage time t_s are then:

$$Q_1 = -I_r\tau\left(\frac{GR+1}{GR+2}\right) + \frac{\tau}{2}(I_r + I_f)e^{-t/\tau} + \frac{GR\tau}{2(GR+2)}(I_r + I_f)\exp\left(-\frac{GR+2}{GR}\frac{t}{\tau}\right)$$

$$Q_2 = -\frac{I_r\tau}{GR+2} + \frac{\tau}{2}(I_r + I_f) - \frac{GR\tau}{2(GR+2)}(I_r + I_f)\exp\left(-\frac{GR+2}{GR}\frac{t}{\tau}\right).$$

Storage time ts is defined as the time at which Q_1 goes to zero. After T_s:

$$Q_1 = 0$$

$$Q_2 = Q_2(t_s) \exp\left(-\frac{GR+1}{GR}\cdot\frac{t}{\tau}\right).$$

The behavior of the model inset in Figure 5c was compared with that of an actual diode in the following way. The response of the diode was obtained from an exact computer solution of the semiconductor equation using the procedures developed by Gummel and Scharfetter.[7] This solution is given in Figure 5c. The area of the diode, 3.33×10^{-6} square centimeters, corresponds to a current density of 300Å per square centimeter.

Using the Lawrence-Warner[10] curves, the average junction capacitance, defined as total charge per total voltage, in the range from 0 to 5.8 volts is 0.153pF. Using the equations and numerical values for the two-lump model given previously, this leads to the double-lump curve in Figure 5c. A 20 per cent reduction of C_j to obtain best fit led to the corrected double-lump curve. The figure shows the solutions for a single lump model for comparison. These results are discussed in Section 2.2.

REFERENCES

1. Gummel, H. K., and Murphy, B. T., "Circuit Analysis by Quasi-Analogue Computation," Proc. IEEE, *55* (October 1967), pp. 1758–62.
2. Sparks, J. J., and Beaufoy, R., "The Junction Transistor as a Charge Control Device," ATEJ, *13* (October 1957), pp. 310–327.
3. Levine, R., unpublished work.
4. Angelo, E. J., Jr., Logan, J., and Sussman, K. W., "The Separation Technique—A Method for Simulating Transistors to Aid Circuit Design," to be published IEEE Trans. Elec. Computers, February 1968; also: Balaban, P., and Logan, J., "Analog Computer Simulation of Semiconductor Circuits," to be presented at Spring Joint Computer Conference, Atlantic City, N. J., April 1968.
5. Kingston, R. H., "Switching Time in Junctions Diodes and Junction Transistors," Proc. IRE, *42,* (May 1954), pp. 829–834.
6. Linvill, J. G., "Lumped Models of Transistors and Diodes," Proc. IRE, *46* (June 1958), pp. 1141–1152; also: Linvill, J. G. and Wunderlin, W., "Transient Response of Junction Diodes," IEEE Trans. Circuit Theory, *10* (June 1963), pp. 191–197.
7. Gummel, H. K., "A Self-Consistent Iterative Scheme for One-Dimensional Steady State Transistor Calculations, IEEE Trans. Elec. Device, *ED 11* (October 1964), pp. 455–465. Also: Scharfetter, D. L. and Gummel, H. K., unpublished work.
8. Kirk, C. T. "A Theory of Transistor Cutoff Frequency Falloff at High Current Densities," IRE Trans. Elec. Devices, *ED 9* (March 1962), pp. 164–174.
9. Ebers, J. J., and Moll, J. L., "Large Signal Behavior of Junction Transistors," Proc. IRE, *42* (Dec. 1954), pp. 1761–1772.
10. Lawrence, H., and Warner, R. M., Jr., "Diffused Junction Depletion Layer Calculations, B.S.T.J., *39* (March 1960), pp. 389–404.

Nonlinear Distortion in Feedback Systems

By J. M. HOLTZMAN

(Manuscript received November 21, 1967)

We give a method for determining the distortion effect of a nonlinearity in a feedback loop.

I. INTRODUCTION

Desoer gives an interesting analysis of distortion resulting from a nonlinearity of the form $v + \epsilon v^m$ (m an odd integer) in a feedback loop.[1] Sandberg considers virtually the same problem for nonlinearities with upper and lower bounds on the slope.[2] On page 2546 of his work, Sandberg suggests that Desoer's result may be sharpened. Our purpose is to show how a small modification of Desoer's analysis might give this sharpening and extend its applicability.

Desoer's method is to find conditions for a particular mapping to be a contraction in a ball. The method presented in another work is particularly suited to that problem and will be applied in this paper.[3] The problem of distortion in nonlinear systems is also considered in References 4 and 5 among other papers.

II. NOTATION AND PRELIMINARIES

The feedback loop (with unity feedback for simplicity) is assumed to be described by

$$y = NL(r - y) \tag{1}$$

where the input r and output y are in some Banach space. L and N are linear and nonlinear operators, respectively, mapping the Banach space into itself. We need not, at this point, specify which Banach space we are working in. Rather, we refer the reader to Reference 2 for details on two Banach spaces of interest for analysis of nonlinear feedback loops.* In particular, Reference 2 shows how to evaluate

* It must be verified that the Banach space (or an appropriate subset) is mapped into itself by the nonlinearity. In particular, nonlinearities such as described by polynomials do not map L_2 into itself.

the norm of the linear operator when it is defined by a convolution operation or by a transfer function (frequency response).

III. THE PROBLEM OF DISTORTION

Suppose that

$$N(x) = x + \epsilon P(x). \tag{2}$$

Then the loop is linear if $\epsilon = 0$ and it is of interest to determine how the loop response differs for a nonzero ϵ. This difference is called the distortion. On the other hand, we might consider some fixed $|\epsilon| > 0$ and determine how small the input r must be in order that the distortion is sufficiently small. This latter question assumes that $P(x)$ is of an order less than x as $x \to 0$.

The following manipulation is convenient for this problem. From equations 1 and 2 we have

$$y = L(r - y) + \epsilon P[L(r - y)]. \tag{3}$$

If we assume that $(I+L)$ has a bounded inverse where I is the identity map,* we obtain

$$y = (I + L)^{-1}Lr + (I + L)^{-1}\epsilon P[L(r - y)].$$

Then, if z is the response of the linearized loop,

$$z = (I + L)^{-1}Lr.$$

And if ξ represents the distortion,

$$\xi = y - z,$$

we have

$$\begin{aligned} \xi &= \epsilon(I + L)^{-1}P[z - L\xi] \\ &\equiv M(\xi). \end{aligned}$$

We are thus interested in finding a fixed point of the operation $M(\xi)$. In particular, how large is ξ? To solve this problem, we use a convenient modification of the contraction mapping fixed point theorem.

IV. THE CONTRACTION MAPPING THEOREM

Let X be a complete metric space (with metric d) containing the closed set Ω and let F map Ω into itself. F is a contraction mapping if there is an $\alpha \in [0, 1)$ such that

$$d[F(x), F(x')] \leqq \alpha d(x, x') \qquad (x, x' \in \Omega).$$

* For conditions for the existence of this bounded inverse, see Reference 2, especially p. 2538.

The contraction mapping theorem (Reference 6, p. 627) states that if F is a contraction mapping then there is a unique $x^* \in \Omega$ such that $x^* = F(x^*)$, that is, x^* is a fixed point of the operation F. Also, x^* is the limit of a sequence $\{x_n\}$ where

$$x_{n+1} = F(x_n)$$

and x_0 is any element of Ω.

One aspect of using the above theorem is finding the appropriate set Ω mapped into itself. Often, the contraction mapping theorem is used when Ω is the whole space, that is, F is globally Lipschitzian. The analysis of Reference 1 may be viewed as a method of determined a ball about the origin such that a mapping is a contraction in that ball. The general problem of simultaneously trying to determine a set mapped into itself such that a mapping is contraction on that set is discussed in Reference 3. The following simple theorem from Reference 3 is useful in this direction.

Theorem: Let B be a Banach space. F maps B into itself and $x_0 \in B$. It is assumed that

(*i*) *F has a derivative at all $x \in B$*

(*ii*) *There is a nondecreasing function g such that if $x \in B$, then*

$$|| F'(x) || \leqq g(|| x - x_0 ||)$$

(*iii*) *There is an $\alpha \in [0, 1)$ such that*

$$g\left(\frac{k}{1 - \alpha}\right) \leqq \alpha$$

where

$$k \geqq || F(x_0) - x_0 ||.$$

Then there is a unique $x^* \in \Omega$ such that

$$x^* = F(x^*)$$

where

$$\Omega = \left\{x: x \in B, || x - x_0 || \leqq \frac{k}{1 - \alpha}\right\}.$$

Remarks: See chapter XVII of Reference 6 for a general discussion of differentiation in Banach spaces.

It is often a straightforward matter to find an appropriate function g as we shall see in the distortion problem of this paper.

V. SOLUTION OF THE DISTORTION PROBLEM

To apply the preceding theorem to the distortion problem of Section III, we first find $M'(\xi)$, then a nondecreasing g such that

$$|| M'(\xi) || \leqq g(|| \xi - \xi_0 ||) = g(|| \xi ||) \qquad (\xi_0 = 0). \tag{4}$$

And with $P(0) = 0$ (for simplicity),

$$\begin{aligned} || M(\xi_0) - \xi_0 || &= || \epsilon(I + L)^{-1}P(z) || \\ &\leqq k. \end{aligned} \tag{5}$$

We must finally find an α $\epsilon[0.1)$ such that

$$g\left(\frac{k}{1 - \alpha}\right) \leqq \alpha. \tag{6}$$

With $(I+L)^{-1}$ and L both assumed to be bounded linear operators, we have

$$|| M'(\xi) || = || \epsilon(I + L)^{-1}P'(z - L\xi)L || \tag{7}$$

(assuming that P has a Fréchet derivative). It should be clear that our method of analysis is not restricted to nonlinearities described by functions of the form of $v + \epsilon v^m$ as used in Reference 1. For the case of the space of continuous real valued functions with the sup norm* and

$$P(x) = x^m \qquad m \text{ an integer} > 1 \text{ (not necessarily odd)} \tag{8}$$

we have that

$$P'(x) = mx^{m-1}. \tag{9}$$

Then, using equations 7 and 9,

$$\begin{aligned} || M'(\xi) || &\leqq | \epsilon | \cdot || (I + L)^{-1} || \cdot m \cdot || (z - L\xi)^{m-1} || \cdot || L || \\ &\leqq m | \epsilon | \cdot || (I + L)^{-1} || (|| z || + || L || \cdot || \xi ||)^{m-1} || L || \\ &\equiv g(|| \xi - \xi_0 ||) \\ &= g(|| \xi ||) \qquad (\xi_0 = 0). \end{aligned} \tag{10}$$

Now, using equations 5 and 6, we obtain

$$m | \epsilon | \cdot || (I + L)^{-1} || \cdot || L || \left(|| z || + \frac{|| L || \cdot | \epsilon | \cdot || (I+L)^{-1} || \cdot || z ||^m}{1 - \alpha} \right)^{m-1} \leqq \alpha. \tag{11}$$

* What might be considered to be a disadvantage of using this space is that the norms of the linear operator are expressed in terms of impulse responses rather than frequency responses.

Condition (11) could be used in several ways. For a fixed ϵ, we could determine how small $\| z \|$ has to be in order for there to be an $\alpha \in [0, 1)$ satisfying (11) and thus get a bound on the distortion $\| \xi \|$. The emphasis in Reference 1 is in determining how small ϵ should be with linearized outputs satisfying $\| z \| < 1$ and the distortion $\| \xi \| < \frac{1}{2}$ in order for the method of successive approximations to converge at a given rate ($\alpha = \frac{1}{4}$). The discussion on page 2546 of Sandberg's article[2] assumes the following conditions (in our notation):

$$\begin{aligned} & m = 3 \\ & \| L \| = 100 \\ & \| (I + L)^{-1} \| = 2. \end{aligned} \tag{12}$$

Then, (11) becomes

$$600 \, | \epsilon | \left(1 + \frac{200 \, | \epsilon |}{3/4}\right)^2 \leqq \frac{1}{4}. \tag{13}$$

If $|\epsilon|$ is less than about 1/2900 (actually a little larger, then (13) is satisfied. Then, the distortion ξ satisfies

$$\begin{aligned} \| \xi \| &\leqq \frac{k}{1 - \alpha} \\ &\leqq | \epsilon | \, \tfrac{4}{3} \, \| (I + L)^{-1} \| \cdot \| z \|^m \\ &\leqq \tfrac{8}{3} \, | \epsilon |. \end{aligned} \tag{14}$$

The bound obtained using equation 25 of Desoer's article[1] is $| \epsilon | \leqq 1/(2150 \cdot 2900)$, a substantially smaller bound.

VI. CONCLUSION

Notice that since we do not require the mapping to be a contraction in the whole space, we only get uniqueness in Ω, the ball of radius $k/(1-\alpha)$. However, the result may be strengthened by also seeking the largest contraction constant α, satisfying condition *iii* of the theorem. Then the fixed point is also unique in the larger ball. On the other hand, uniqueness information might be available from another source (for example, a property of a differential equation).

We notice that the existence of derivatives in the theorem may actually be relaxed if there is a nondecreasing function suitably bounding Lipschitz constants. We also mention the possibility of using transformations to facilitate the application of the result.

The following may be helpful in visualizing the application of the contraction mapping theorem.* Assume that condition *iii* of the theorem of Section IV is satisfied with equality, that is,

$$g\left(\frac{k}{1-\alpha}\right) = \alpha.$$

The radius of the ball Ω is $k/(1-\alpha)$. Letting

$$r = \frac{k}{1-\alpha}$$

the condition is seen to be

$$g(r) = 1 - \frac{k}{r}$$

which Fig. 1 shows pictorially.

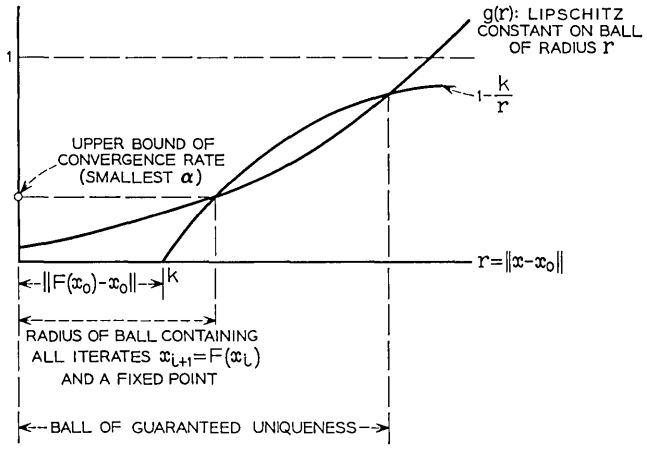


Fig. 1 — Contraction mapping theorem.

VII. ACKNOWLEDGEMENT

We greatly appreciate the comments of H. S. Witsenhausen.

* Suggested by H. S. Witsenhausen.

REFERENCES

1. Desoer, C. A., "Nonlinear Distortion in Feedback Amplifiers," IRE Trans. Circuit Theory, *CT-9,* (March 1962), pp. 2–6.
2. Sandberg, I. W., "Signal Distortion in Nonlinear Feedback Systems," B.S.T.J., *42,* No. 6 (November 1963), pp. 2533–2550.
3. Holtzman, J. M., "The Use of the Contraction Mapping Theorem with Derivatives in a Banach Space," Quart. Appl. Math., (to be published).
4. Zames, G., "Functional Analysis Applied to Nonlinear Feedback Systems," IEEE. Trans. Circuit Theory, *CT-10* (September 1963), pp. 392–404.
5. Anderson, D. R. and Leon, B. J., "Nonlinear Distortion and Truncation Errors in Frequency Converters and Parametric Amplifiers," IEEE Trans. Circuit Theory, *CT-12* (September 1965), pp. 314–321.
6. Kantorovich, L. V. and Akilov, G. P., *Functional Analysis in Normed Spaces,* trans. D. E. Brown, ed. A. P. Robertson, New York: The MacMillan Co., 1964.

Numerical Integration of Systems of Stiff Nonlinear Differential Equations

By I. W. SANDBERG and H. SHICHMAN

(Manuscript received November 29, 1967)

In connection with the design of transistor circuits, for example, it is frequently necessary to obtain a numerical solution of a system of nonlinear ordinary differential equations. In some cases, these equations possess a property that leads to intolerable computational requirements relative to the use of standard predictor-corrector techniques or general linear multipoint formulas of open type.

Here we describe an alternative approach which has been used to solve some practical problems by permitting dramatic step-size increases (for example, a factor of 10^4). The approach is developed in a way which provides some detailed understanding of why it is useful.

I. INTRODUCTION

In connection with the design of transistor circuits, for example, it is often necessary to obtain a numerical solution of a system of nonlinear differential equations

$$\dot{x} + f(x, t) = 0, \qquad t \geqq 0, \qquad [x(0) = x_0] \tag{1}$$

in which x and $f(x, \cdot)$ are N-vector-valued functions of t. The simplest numerical-integration formula which can be in principle used for this purpose is Euler's formula:

$$y_{n+1} = y_n + hy'_n, \qquad n \geqq 0 \tag{2}$$

in which h, a positive number, is the step size; $y_0 = x_0$;

$$y'_n = -f(y_n, nh) \quad \text{for} \quad n \geqq 0;$$

and y_n is of course the approximation to $x(nh)$ for $n \geqq 1$.

It is frequently the case that $f(x, \cdot)$ possess a property that leads to computational requirements consistent with the use of (2) that are intolerable. To see clearly how this situation can arise suppose that

the solution of (1) is desired over some finite interval $[0, \tau]$, and consider the very special case in which $f(x, t) = Ax$ with A an $N \times N$ matrix possessing distinct eigenvalues $\{a_i\}$ all of which have positive real parts. Then using the fact there exists a nonsingular transformation T such that

$$A = TDT^{-1}, \qquad D = \text{diag}\,(a_1, a_2, \cdots, a_N) \tag{3}$$

we have

$$y_{n+1} = T(1_N - hD)T^{-1}y_n, \qquad n \geqq 0, \qquad [y_0 = x_0] \tag{4}$$

in which 1_N is the identity matrix of order N. From (4)

$$y_k = T(1_N - hD)^k T^{-1}x_0, \qquad k \geqq 0. \tag{5}$$

Since

$$x(kh) = Te^{-Dkh}T^{-1}x_0, \qquad k \geqq 0 \tag{6}$$

it is evident that the numerical solution is "acceptable" if h is so small that $(1 - ha_i)^k$ is an "acceptable" approximation to $e^{-a_i kh}$ for all i and all values of k for which $0 \leqq kh \leqq \tau$. On the other hand if for some value of i

$$| 1 - ha_i | = 1, \quad \text{or} \quad | 1 - ha_i | > 1$$

then for at least one initial condition vector x_0, $\{|| y_k ||\}_0^\infty$ ($||\cdot||$ denotes the usual Euclidian norm) does not approach zero as $k \to \infty$ or is unbounded, respectively [that is, (2) is numerically unstable]. Therefore if the sequence $\{y_k\}$ defined by (4) is to be a good approximation to the samples of the solution of (1) with $f(x, t) = Ax$, it is certainly necessary that

$$| 1 - ha_i | < 1 \quad \text{for all } i. \tag{7}$$

Moreover, in order to fully determine the character of the solution of the differential equation, it is reasonable to assume that τ, the length of the interval over which the solution is desired, is proportional (by some factor c such as 3 or 10) to the reciprocal of $\min_i \text{Re}(a_i)$ (that is, proportional to the largest time constant of the system). Thus in addition to (7) we have

$$\tau = c[\min_i \text{Re}\,(a_i)]^{-1}. \tag{8}$$

A lower bound on the number of evaluations of (2) necessary to compute the solution is τ/h where h satisfies (7). If all of the a_i are

real, the smallest lower bound is simply

$$\tfrac{1}{2}c \frac{\max_i (a_i)}{\min_i (a_i)}. \tag{9}$$

It is a simple matter to give examples of, for instance, positive-element linear RC networks governed by a state equation of the form $\dot{x} + Ax = 0$ for which the bound (9) can be made arbitrarily large by choosing the value of one capacitor to be arbitrarily small. Thus, from the practical viewpoint, computation based on (2) can be impossible as a result of the presence of parasitic circuit elements that have no really significant effect on the circuit performance! It is not surprising therefore that a more complex and pressing problem of the same type arises in connection with the numerical solution of the nonlinear differential equations of transistor circuits, as a result of, for example, the parasitic capacitors associated with the models of transistors. For many practical circuits of this type, computation time estimates, based upon use of (2) and a modern high-speed computer, are about 1000 hours.

The well-known basic problem described above arises not only in connection of the use of (2), but (as can easily be shown) is encountered also in attempts to use more general integration formulas of open type[1, 2]

$$y_{n+1} = \sum_{k=0}^{p} a_k y_{n-k} + h \sum_{k=0}^{p} b_k y'_{n-k} , \tag{10}$$

or predictor-corrector techniques[1, 2] such as

$$\begin{aligned} y_{n+1}^{(p)} &= y_{n-1} + 2hy'_n \\ y_{n+1}^{(c)} &= y_n + \tfrac{1}{2}h(y'_n + y'^{(p)}_{n+1}). \end{aligned} \tag{11}$$

The fundamental difficulty associated with the integration of "stiff equations" results from the restrictions that must be imposed on h in order to insure numerical stability.

The purpose of this paper is to consider the properties of alternative numerical methods for obtaining solutions of equations of the form (1). Our principal objective is to present some analytical results that shed some light on the properties of a class of numerical-integration techniques that have been used to solve practical transistor circuit problems by permitting dramatic step-size increases (for example, a factor of 10^4) relative to the methods defined by (10) and (11).

More explicitly, attention is focused on "large-h algorithms" based

on, or derived from, the standard formula of closed type

$$y_{n+1} = y_n + hy'_{n+1} \tag{12}$$

which is a special case of the general multipoint formula

$$y_{n+1} = \sum_{k=0}^{p} a_k y_{n-k} + h \sum_{k=-1}^{p} b_k y'_{n-k} \tag{13}$$

with $b_{-1} \neq 0$. There is an extensive body of information concerning (12) in the numerical-analysis literature only for the case in which h is "sufficiently small."

II. INTEGRATION FORMULA

If we use the numerical-integration formula

$$y_{n+1} = y_n + hy'_{n+1} \tag{12}$$

in an attempt to compute the solution of (1), then y_{n+1} is defined implicitly in terms of y_n through

$$y_{n+1} + hf[y_{n+1}, (n+1)h] = y_n, \qquad n \geqq 0, \qquad [y_0 = x_0]. \tag{13}$$

For the special case considered in Section I, in which $f(x, t) = Ax$ and $A = TDT^{-1}$, we have

$$y_{n+1} = T(1_N + hD)^{-1}T^{-1}y_n, \qquad n \geqq 0 \tag{14}$$

and to the extent that $(1 + ha_i)^{-1}$ is a good approximation to $e^{-a_i h}$, (13) generates an acceptable numerical solution of the differential equation. More explicitly (13) generates the *exact* solution of the differential equation

$$\dot{x} + \tilde{A}x = 0, \qquad t \geqq 0, \qquad [x(0) = x_0] \tag{15}$$

in which $\tilde{A} = T\tilde{D}T^{-1}$ and $e^{-\tilde{D}h} = (1_N + hD)^{-1}$.

Let us suppose now that all of the a_i are real and that ha_i is very small relative to unity for i belonging to a proper subset $\mathcal{S}$ of $\mathfrak{N} \triangleq \{1, 2, \cdots, N\}$, and that ha_i is very large relative to unity for i belong to the complement $\bar{\mathcal{S}}$ of $\mathcal{S}$ with respect to $\mathfrak{N}$. Then for all $i \in \mathcal{S}$, $\tilde{a}_i$, the ith element of $\tilde{D}$ is very nearly a_i, while for all $i \in \bar{\mathcal{S}}$, $\tilde{a}_i < a_i$ and $\tilde{a}_i$ is very much larger than all of the $\tilde{a}_i$ for which $i \in \mathcal{S}$.

In other words, roughly speaking, (13) generates a solution to a differential equation governing a system similar to that governed by $\dot{x} + Ax = 0$; the former system has virtually the same low-frequency performance and less pronounced high-frequency performance. To

look at the situation in still another way, in using (13) we are able to (*i*) break away from an extremely restrictive requirement on h for numerical stability, such as (7), and (*ii*) trade step-size for accuracy of high-frequency solution components.

The simple heuristic argument given above suggests that the use of (12) can lead to a considerable increase in permissible step sizes for a class of nonlinear transistor circuit problems in which typically the Jacobian matrix $\partial f(x, t)/\partial x$ of $f(x, t)$ along the solution of (1) possesses only real eigenvalues which are widely separated. This argument is supported by a proposition, proved in Section IV, which is concerned with the case in which there exists a constant $m > 0$ such that (with $\langle \cdot , \cdot \rangle$ denoting the usual inner product)

$$\langle y, f(y, nh) - f(0, nh) \rangle \geqq m \| y \|^2 \tag{16}$$

for all $n \geqq 0$ and all y. If this condition is satisfied for all $h > 0$, which for the scalar case is true if

$$\frac{\partial f(y, t)}{\partial y} \geqq m$$

for all t and all y, if $\| f(0, t) \| \to 0$ as $t \to \infty$ or if $\| f(0, t) \|$ is uniformly bounded on $[0, \infty)$, then (as can easily be shown) $\| x(t) \| \to 0$ as $t \to \infty$ or $\| x(t) \|$ is uniformly bounded on $[0, \infty)$, respectively. The Proposition asserts that if (16) is met and y_{n+1} is defined for $n \geqq 0$ by (13) ,then

$$\| y_n \| \leqq (1 + mh)^{-n} \| x_0 \| + \sum_{k=0}^{n-1} (1 + mh)^{-(k+1)} \| hf[0, (n - k)h] \|$$

for all $n \geqq 1$, which implies that (13) is numerically stable for all h in the sense that for all h, $\| f(0, nh) \| \to 0$ as $n \to \infty$ implies that $y_n \to 0$ as $n \to \infty$ and $\{\| f(0, nh) \|\}_0^\infty$ bounded implies that $\{y_n\}_0^\infty$ is bounded.

Although the result stated above does not provide quantitative information concerning the errors incurred in using (13), it does show under a reasonable assumption concerning $f(x,t)$ that unlike all formulas (10) of open type and unlike predictor-corrector methods such as (11), (13) defines for any step size a sequence $\{y_n\}$ which is consistent with either or both of two possible basic properties of the true solution.

The discussion above does not take into account the fact that at each step errors are inevitably introduced in solving the equation

$$y_{n+1} + hf[y_{n+1}, (n + 1)h] = y_n \tag{17}$$

for y_{n+1}. Consider the result of using the iteration scheme

$$y_{n+1}^{(k+1)} = y_n - hf[y_{n+1}^{(k)}, (n+1)h], \qquad y_{n+1}^{(0)} = y_n$$

which is the usual method described [1, 2] in connection with the theory of integration formulas of closed type. For the linear case [that is, for $f(x, t) = Ax$],

$$\begin{aligned} y_{n+1}^{(k)} &= \sum_{i=0}^{k} (-hA)^i y_n \\ &= T \sum_{i=0}^{k} (-hD)^i T^{-1} y_n . \end{aligned} \tag{18}$$

Therefore, if $\tilde{y}_1$ denotes the approximation to y_1 computed from y_0 after k_1 iterations, and if $\tilde{y}_2$ denotes the approximation to y_2 computed from $\tilde{y}_1$ after k_2 iterations and so forth, then

$$\tilde{y}_K = T\Theta_{k_K}\Theta_{k_{K-1}} \cdots \Theta_{k_2}\Theta_{k_1}T^{-1}y_0$$

in which

$$\Theta_{k_p} = \operatorname{diag}\left(\sum_{i=0}^{k_p} (-ha_1)^i, \cdots, \sum_{i=0}^{k_p} (-ha_N)^i\right).$$

Since (assuming now that all of the a_i are real)

$$\left| \sum_{i=0}^{k_p} (-ha_i)^i \right| > 1 \tag{19}$$

provided that $ha_i > 2$ and $k_p \geqq 1$, if $ha_i > 2$ for some i, then $\| \tilde{y}_k \| \to \infty$ as $k \to \infty$ for some initial condition y_0, independent of the sequence $k_1, k_2, \cdots$. Therefore the usual iteration method will reintroduce the numerical instability for insufficiently small h which it is our objective to avoid.*

Let us consider now a different and more general approach of solving (17) for y_{n+1}. Assume that there exists a positive constant l such that $f(y, nh)$ satisfies the Lipshitz condition

$$\| f(y_1, nh) - f(y_2, nh) \| \leqq l \| y_1 - y_2 \|$$

for all $n \geqq 0$ and all y_1 and y_2. Suppose also that the smallest eigenvalue of the symmetric part of the Jacobian matrix $\partial f(y, nh)/\partial y$ of

* Similar instability results for the nonlinear case can be proved. But since this is hardly surprising, we shall not consider the matter further.

$f(y, nh)$ is bounded from below by m, a positive constant, for all y and all n.

Ideally, we would like to determine the sequence $\{y_n\}_0^\infty$ defined by

$$y_{n+1} + hf[y_{n+1}, (n+1)h] = y_n, \qquad n \geqq 0.$$

Suppose that we determine instead a sequence $\{\tilde{y}_n\}_0^\infty$ such that $\tilde{y}_0 = y_0$

$$\| \tilde{y}_n - y_n^* \| \leqq \epsilon$$

and

$$y_{n+1}^* + hf[y_{n+1}^*, (n+1)h] = \tilde{y}_n$$

for $n \geq 0$ in which ϵ is an arbitrary positive constant independent of n. In other words, suppose that at each step the *local* error in solving for y_{n+1} is at most ϵ. Then, according to Theorem 1 (Section IV)

$$\| \tilde{y}_n - y_n \| \leqq \epsilon(1 + hl)(1 + hm)^{-1} \sum_{k=0}^{n-1} (1 + hm)^{-k}$$

for all $n \geq 1$, which of course implies the uniform bound

$$\| \tilde{y}_n - y_n \| \leqq \epsilon(1 + hl)(hm)^{-1}, \qquad n \geqq 1. \tag{20}$$

Our assumption concerning $\partial f(y,nh)/\partial y$ implies that the condition

$$\langle y, f(y, nh) - f(0, nh) \rangle \geqq m \| y \|^2$$

of the Proposition is met. Thus it follows from the Proposition and (20) that if the local error in solving for y_{n+1} is held to within ϵ at each step, then the algorithm is numerically stable for all h in the sense that for all h (*i*) $\{\| f(0, nh) \|\}_0^\infty$ bounded implies that $\{\tilde{y}_n\}_0^\infty$ is bounded, and (*ii*) $\| f(0, nh \| \to 0$ as $n \to \infty$ implies that for any $\delta > 0$ there exists an n_0 such that $\| \tilde{y}_n \| \leqq \epsilon(1 + hl)(hm)^{-1} + \delta$ for all $n \geqq n_0$.

The combination of this stability result and the heuristic argument of Section I strongly suggests that the following approach should permit the use of considerably increased step sizes with acceptable accuracy, for many of the "widely-separated eigenvalue" problems described earlier. Referring to (17), solve for y_{n+1} at each step using, say, the Newton-Raphson technique;* iterate until some norm of

* After the work reported here had been completed, A. N. Willson, Jr. brought to our attention a preprint of a paper by R. Willoughby and several of his colleagues at IBM, in which an approach of this type is suggested. The preprint does not contain the principal results of this paper, the material of Section IV.

the difference between the last two iterates is not greater than some small prescribed constant.

In particular, notice that for $f(x, t) = Ax$, this approach, using the Newton-Raphson iteration procedure, reduces to the use of the formula $y_{n+1} = (1_N + hA)^{-1}y_n$ (that is, to equation 14).

The technique described above has provided a significant reduction in total computation time for several types of practical problems. It was used, for example, to solve the system of differential equations governing the circuit of Fig. 1, an oscillator designed to supply a 1 kc signal. The 16 G Western Electric 100 Mc. silicon transistor of Fig. 1 was represented by a charge-control model (see Section 6.2, pp. 556–557 of Koehler[3]) using two nonlinear charge-controlled voltage sources, with the result that the system of equations for the circuit is of order 5.

Motivated by the fact that the local-truncation error for formula (12) is $\frac{1}{2}h^2\ddot{x}(\xi)$ for some $\xi \in [nh, (n + 1)h]$, the following method was used (for this problem as well as for others) to control the step size. Let e denote the largest of the magnitudes of the elements of the vector of second differences associated with the most recently computed point. If $e \in [\frac{1}{4}\bar{e}, \bar{e}]$ (for this problem $\bar{e}$ was taken to be 10^{-4}), then the point is accepted; if $e > \bar{e}$, then the point is rejected and the calculation is repeated with h replaced with $\frac{1}{2}h$. If $e < \frac{1}{4}\bar{e}$, then the point is accepted and h is replaced by $2h$ in the computation of the next point. Average step-size increases of about 10^4 (relative to, for example, the use of a forth-order predictor-corrector method) were obtained for this problem (see Fig. 2).

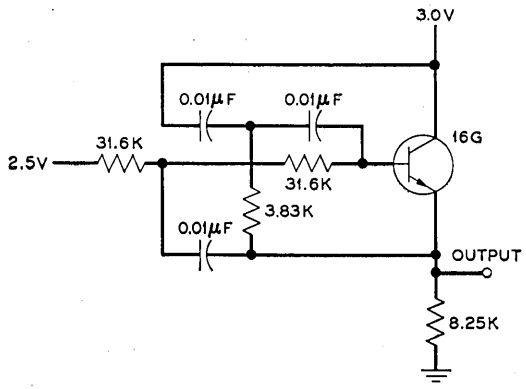


Fig. 1 — One-kilocycle oscillator using a 16G "100 megacycle" transistor.

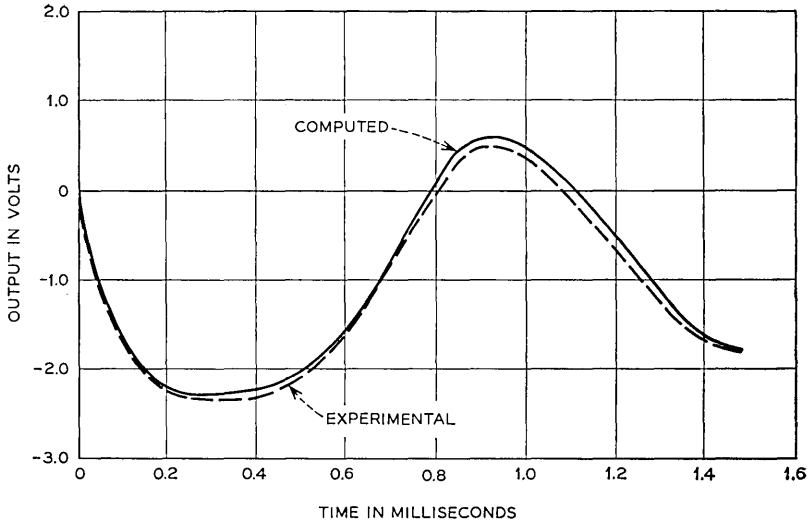


Fig. 2 — Comparison of computed and experimental response of the oscillator shown in Fig. 1.

III. AN EXPLICIT INTEGRATION FORMULA

Of particular interest in connection with the approach described above is the numerical-integration formula

$$y_{n+1} = y_n - \{1_N + hf'[y_n, (n+1)h]\}^{-1} hf[y_n, (n+1)h], \qquad n \geqq 0, \quad [y_0 = x_0] \tag{21}$$

which is obtained from

$$Y_{n+1} + hf[Y_{n+1}, (n+1)h] = Y_n \tag{22}$$

by replacing Y_n by y_n and using as the approximation y_{n+1} to Y_{n+1} the result obtained by using one step of the Newton-Raphson iteration scheme with y_n the initial point. That is, with

$$\begin{aligned} Q(z) &= z + hf[z, (n+1)h] - y_n , \\ y_{n+1} &= y_n - [Q'(z)|_{z=y_n}]^{-1} Q(z)|_{z=y_n} . \end{aligned} \tag{23}$$

In spite of its relative simplicity, it has been found that formula (21) is useful for solving problems of the type that we have been considering. For the problem of Fig. 1, it has led to an average step size increase of about 10^3.

In view of the simplicity of formula (21), and especially in view of the fact that y_{n+1} is defined explicitly in terms of y_n, it deserves special consideration.

Theorem 2 (Section IV) asserts that for any $h > 0$ there exist positive constants k_1 and k_2 such that $k_1 < 1$ and

$$|| y_n || \leqq k_1^n || y_0 || + hk_2 \sum_{k=0}^{n-1} k_1^k || f[0, (n-k)h] || \tag{24}$$

for $n \geqq 1$, provided that the Jacobian matrix $\partial f(y, nh)/\partial y$ satisfies certain conditions. For the scalar case, these conditions reduce to:

(*i*) there exist positive constants k and m such that

$$m \leqq \frac{\partial f(y, nh)}{\partial y} \leqq k$$

for all y and all $n \geq 1$

(*ii*)
$$2\frac{\partial f(y, nh)}{\partial y} - \frac{\partial f(\alpha y, nh)}{\partial y} \geqq 0$$

for all y, $n \geqq 1$ and $\alpha \in [0, 1]$.

Clearly, under these conditions, $y_n \to 0$ as $n \to \infty$ if $f(0, nh) \to 0$ as $n \to \infty$ and $\{y_n\}$ is bounded if $\{||f(0, nh)||\}$ is bounded.

The function $f(y, nh)$ of Fig. 3 is one for which conditions (*i*) and (*ii*) are clearly met. If condition (*ii*) is not met, then (24) need not follow. To show this, consider, for example, the function of Fig. 4

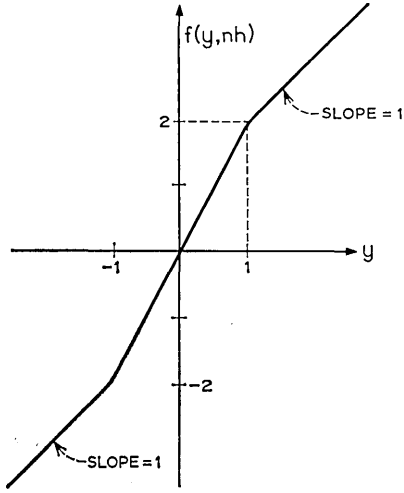


Fig. 3 — Definition of $f(y, nh)$ for all n.

which meets condition (i), but not condition (ii). We have from (21):

$$h = 1 \quad \text{and} \quad y_0 = 1 \quad \text{imply that} \quad y_1 = -1$$

and

$$h = 1 \quad \text{and} \quad y_1 = -1 \quad \text{imply that} \quad y_2 = 1$$

from which it is clear that for this function $y_n = (-1)^n$ if $h = 1$ and $y_0 = 1$, which of course implies [here $f(0, nh) = 0$ for all n] that (24) is not satisfied. Thus we see that if condition (ii) is not met, then (24) need not follow.

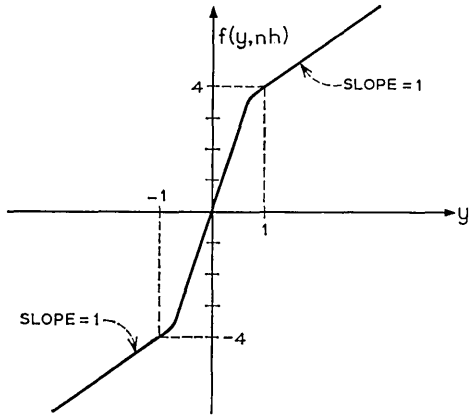


Fig. 4 — Alternate definition of $f(y, nh)$ for all n.

IV. PROPOSITION AND THEOREMS*

Proposition: If $\{y_n\}$ satisfies

$$y_{n+1} + hf[y_{n+1}, (n + 1)h] = y_n, \qquad n \geqq 0$$

and if there exists an $m > 0$ such that

$$\langle y, f(y, nh) - f(0, nh) \rangle \geqq m \, || \, y \, ||^2, \qquad n \geqq 0$$

for all real y, then

$$|| \, y_n \, || \leqq (1 + mh)^{-n} \, || \, y_0 \, || + \sum_{k=0}^{n-1} (1 + mh)^{-k} \, || \, hf[0, (n - k)h] \, ||$$

for $n \geq 1$.

* Throughout this section, $||\cdot||$ denotes the usual Euclidean Norm and $\langle \cdot, \cdot \rangle$ denotes the corresponding usual scalar product.

Proof: Clearly,

$$\begin{aligned}\langle y_{n+1} , y_n - hf[0, (n+1)h]\rangle &= \langle y_{n+1} , y_{n+1} + hf[y_{n+1} , (n+1)h] - hf[0, (n+1)h]\rangle \\ &\geqq (1 + mh) \,||\, y_{n+1} \,||^2,\end{aligned}$$

and, by the Schwarz inequality,

$$\begin{aligned}\langle y_{n+1} , y_n - hf[0, (n+1)h]\rangle \leqq{}& ||\, y_{n+1} \,||\cdot||\, y_n \,|| \\ &+ ||\, y_{n+1} \,||\cdot||\, hf[0, (n+1)h] \,||.\end{aligned}$$

Thus

$$||\, y_{n+1} \,|| \leqq (1 + mh)^{-1} \,||\, y_n \,|| + (1 + mh)^{-1} \,||\, hf[0, (n+1)h] \,||$$

from which we have

$$||\, y_n \,|| \leqq (1 + mh)^{-n} \,||\, y_0 \,|| + \sum_{k=0}^{n-1} (1 + mh)^{-(k+1)} \,||\, hf[0, (n-k)h] \,||$$

for $n \geqq 1$, which completes the proof.

Definition: Let $\lambda(y, nh)$ denote the smallest eigenvalue of the symmetric part of $\partial f(y, nh)/\partial y$.

Theorem 1: Suppose that there exists a constant m such that $\lambda(y, nh) \geqq m > 0$ for all $n \geqq 0$ and all y, and that there exists a constant l such that

$$||\, f(y_1 , nh) - f(y_2 , nh) \,|| \leqq l \,||\, y_1 - y_2 \,||$$

for all $n \geqq 0$ and all y_1 and y_2. If $\{y_n\}$ satisfies

$$y_{n+1} + hf[y_{n+1} , (n+1)h] = y_n , \qquad n \geqq 0$$

if, with ϵ a positive constant, $\{\tilde{y}_n\}$ satisfies

$$||\, \tilde{y}_n - y_n^* \,|| \leqq \epsilon \quad \text{for} \quad n \geqq 0 \quad \text{with}$$
$$y_{n+1}^* + hf(y_{n+1}^* , (n+1)h) = \tilde{y}_n$$

then

$$\begin{aligned}||\, \tilde{y}_n - y_n \,|| \leqq{}& (1 + hm)^{-n} \,||\, \tilde{y}_0 - y_0 \,|| \\ &+ (1 + hm)^{-1}(1 + hl)\,\epsilon \sum_{k=0}^{n-1} (1 + hm)^{-k}\end{aligned}$$

for $n \geqq 1$.

Proof: We have for $n \geqq 0$:

$$\tilde{y}_{n+1} + hf[\tilde{y}_{n+1} + (y^*_{n+1} - \tilde{y}_{n+1}), (n+1)h] = \tilde{y}_n + (\tilde{y}_{n+1} - y^*_{n+1})$$

and

$$y_{n+1} + hf[y_{n+1} + (y^*_{n+1} - \tilde{y}_{n+1}), (n+1)h]$$
$$= y_n + hf[y_{n+1} + (y^*_{n+1} - \tilde{y}_{n+1}), (n+1)h] - hf[y_{n+1}, (n+1)h].$$

Therefore

$$\begin{aligned}\tilde{y}_{n+1} - y_{n+1} &+ hf[\tilde{y}_{n+1} + (y^*_{n+1} - \tilde{y}_{n+1}), (n+1)h] \\ &- hf[y_{n+1} + (y^*_{n+1} - \tilde{y}_{n+1}), (n+1)h] \\ &= \tilde{y}_n - y_n + (\tilde{y}_{n+1} - y^*_{n+1}) + hf(y_{n+1}, (n+1)h) \\ &- hf[y_{n+1} + (y^*_{n+1} - \tilde{y}_{n+1}), (n+1)h]. \qquad (25)\end{aligned}$$

With f'_s the symmetric part of $\partial f(y, nh)/\partial y$, the inner-product of $(\tilde{y}_{n+1} - y_{n+1})$ with the left side of (25) is

$$|| \tilde{y}_{n+1} - y_{n+1} ||^2 + h\Big\langle \tilde{y}_{n+1} - y_{n+1}, \int_0^1 f'_s\{\alpha[\tilde{y}_{n+1} + (y^*_{n+1} - \tilde{y}_{n+1})]$$
$$+ (1-\alpha)[y_{n+1} + (y^*_{n+1} - y_{n+1})], (n+1)h\}\, d\alpha(\tilde{y}_{n+1} - y_{n+1})\Big\rangle, \qquad (26)$$

since

$$f[\tilde{y}_{n+1} + (y^*_{n+1} - \tilde{y}_{n+1}), (n+1)h] - f[y_{n+1} + (y^*_{n+1} - \tilde{y}_{n+1}), (n+1)h]$$
$$= \int_0^1 \frac{\partial f[y, (n+1)h]}{\partial y}\bigg|_{y=\alpha[\ ]+(1-\alpha)[\ ]} d\alpha(\tilde{y}_{n+1} - y_{n+1}).$$

Expression (26) is bounded from below by

$$(1 + hm) || \tilde{y}_{n+1} - y_{n+1} ||^2.$$

By the Schwarz inequality, the inner-product of $(\tilde{y}_{n+1} - y_{n+1})$ with the right side of (25) is bounded from above by

$$|| \tilde{y}_{n+1} - y_{n+1} || \cdot || \tilde{y}_n - y_n ||$$
$$+ || \tilde{y}_{n+1} - y_{n+1} || \cdot || \tilde{y}_{n+1} - y^*_{n+1} || + || \tilde{y}_{n+1} - y_{n+1} ||$$
$$\cdot || hf[y_{n+1}, (n+1)h] - hf[y_{n+1} + (y^*_{n+1} - \tilde{y}_{n+1}), (n+1)h] ||,$$

which is bounded from above by

$$|| \tilde{y}_{n+1} - y_{n+1} || \cdot || \tilde{y}_n - y_n || + ||\tilde{y}_{n+1} - y_{n+1} || (\epsilon + hl\epsilon).$$

Thus,

$$|| \tilde{y}_{n+1} - y_{n+1} || \leqq (1 + hm)^{-1} || \tilde{y}_n - y_n || + (1 + hm)^{-1}(1 + hl)\epsilon,$$

from which it follows that

$$|| \tilde{y}_n - y_n || \leqq (1 + hm)^{-n} || \tilde{y}_0 - y_0 || + (1 + hm)^{-1}(1 + hl)\,\epsilon \sum_{k=0}^{n-1} (1 + hm)^{-k}$$

for all $n \geqq 1$.

Theorem 2: If $\{y_n\}$ satisfies

$$y_{n+1} = y_n - \{1_N + hf'[y_n, (n + 1)h]\}^{-1} hf[y_n, (n + 1)h]$$

for $n \geqq 0$, *if*

(*i*) *there exists a constant* $k < \infty$ *such that*

$$\left|\left| \frac{\partial f(y, nh)}{\partial y} \right|\right| \leqq k$$

for all $n \geqq 1$ *and all* y

(*ii*) *there exists a constant* $m > 0$ *such that* $\lambda(y, nh) \geqq m$ *for all* $n \geqq 1$ *and all* y

(*iii*) *with* $F \triangleq hf'[y, (n + 1)h]$ *and* $F_\alpha \triangleq hf'[\alpha y, (n + 1)h]$, *the symmetric part of* $\{(2F - F_\alpha)F_\alpha^t\}$ *is* nonnegative definite for all* y, *all* n, *and all* $\alpha \in [0, 1]$,

then there exist positive constants k_1 *and* k_2 *such that* $k_1 < 1$ *and*

$$|| y_n || \leqq k_1^n || y_0 || + hk_2 \sum_{k=0}^{n-1} k_1^k || f[0, (n - k)h] || \quad \text{for all} \quad n \geqq 1.$$

Proof: We have

$$y_{n+1} = y_n - \{1_N + hf'[y_n, (n+1)h]\}^{-1}\{hf[y_n, (n+1)h] - hf[0, (n+1)h]\} - \{1_N + hf'[y_n, (n + 1)h]\}^{-1} hf[0, (n + 1)h];$$

hence

$$|| y_{n+1} || \leqq \left|\left| 1_N - (1_N + F)^{-1} \int_0^1 F_\alpha \, d\alpha \right|\right| \cdot || y_n || + || (1_N + F)^{-1} || \cdot || hf[0, (n + 1)h] || \qquad (27)$$

* The superscript t denotes matrix transposition.

with the understanding that F and F_α are evaluated at $y = y_n$, since

$$hf[y_n\ ,\ (n+1)h] - hf[0,\ (n+1)h] = \int_0^1 hf'[\alpha y_n\ ,\ (n+1)h]\ d\alpha\ y_n\ .$$

We now prove that there exists $k_1 \in (0, 1)$ such that

$$\left|\left| 1_N - (1_N + F)^{-1} \int_0^1 F_\alpha\ d\alpha \right|\right| \leqq k_1$$

for all n and all y_n.

From condition (*iii*), with V an arbitrary N-vector,

$$\langle (2F^t - F_\alpha^t)V,\ F_\alpha^t V\rangle \geqq 0$$

or

$$\langle 2F^t V,\ F_\alpha^t V\rangle - \langle F_\alpha^t V,\ F_\alpha^t V\rangle \geqq 0$$

which implies that

$$||\ F_\alpha^t V\ ||^2 - 2\langle F^t V,\ F_\alpha^t V\rangle + ||\ F^t V\ ||^2 \leqq ||\ F^t V\ ||^2$$

or

$$||\ (F^t - F_\alpha^t)V\ ||^2 \leqq ||\ F^t V\ ||^2.$$

In view of conditions (*i*) and (*ii*), it is evident that there exists a $\xi \in (0, 1)$ such that

$$-2\langle F_\alpha^t V,\ V\rangle \leqq -(1-\xi)\ ||\ V\ ||^2 - 2(1-\xi)\langle F^t V,\ V\rangle - (1-\xi)\ ||\ F^t V\ ||^2$$

for all α, n, y_n, and V. Therefore

$$||\ (F^t - F_\alpha^t)V\ ||^2 - ||F^t V\ ||^2 - 2\langle F_\alpha^t V,\ V\rangle$$
$$\leqq -(1-\xi)\ ||\ V\ ||^2 - 2(1-\xi)\langle F^t V,\ V\rangle - (1-\xi)\ ||\ F^t V\ ||^2$$

which is the same as

$$||\ V\ ||^2 + ||\ (F^t - F_\alpha^t)V\ ||^2 + 2\langle (F^t - F_\alpha^t)V,\ V\rangle$$
$$\leqq \xi\ ||\ V\ ||^2 + 2\xi\langle F^t V,\ V\rangle + \xi\ ||\ F^t V\ ||^2$$

or

$$||\ (1_N + F^t - F_\alpha^t)V\ ||^2 \leqq \xi\ ||\ (1_N + F^t)V\ ||^2.$$

With $U = (1_N + F^t)V$, we have

$$||\ (1_N + F^t - F_\alpha^t)(1_N + F^t)^{-1}U\ ||^2 \leqq \xi\ ||\ U\ ||^2. \tag{28}$$

Since (28) is satisfied for all U,

$$|| (1_N + F^t - F_\alpha^t)(1_N + F^t)^{-1} || \leqq k_1$$

with $k_1 = (\xi)^{\frac{1}{2}}$. However,

$$|| (1_N + F^t - F_\alpha^t)(1_N + F^t)^{-1} || = || (1_N + F)^{-1}(1_N + F - F_\alpha)||,$$

and

$$\left|\left| 1_N - (1_N + F)^{-1} \int_0^1 F_\alpha \, d\alpha \right|\right|$$

$$\leqq \int_0^1 || (1_N + F)^{-1}(1_N + F - F_\alpha) || \, d\alpha \leqq k_1 .$$

Consider now $||(1_N+F^{-1}||$. Since for any V

$$|| (1_N + F)V ||^2 = || V ||^2 + 2\langle FV, V\rangle + || FV ||^2 \geqq (1 + 2hm) || V ||^2,$$

it follows at once that

$$|| (1_N + F)^{-1} || \leqq (1 + 2hm)^{-\frac{1}{2}}.$$

Thus with $k_2 = (1+2hm)^{-\frac{1}{2}}$

$$|| y_{n+1} || \leqq k_1 || y_n || + k_2 || hf[0, (n + 1)h] ||$$

from which we obtain the bound on $||y_n||$ stated in the theorem.

V. FINAL REMARKS

The algorithm described in this paper is a marriage of two standard techniques, the use of a well-known closed-type numerical-integration formula and the Newton-Raphson iteration procedure. It is clear that the approach is of use in connection with a certain class of practical problems, and, what is of at least as much importance, we have some detailed understanding of why the algorithm is useful.

It is also clear that some natural generalizations and extensions of the approach, such as using different closed-type formulas* or different methods of solving systems of nonlinear equations, will lead to more efficient techniques. Finally, since there are several alternate approaches available which are also of use in certain cases (see Pope,

* For example, for the trapezoidal rule $y_{n+1} = y_n + \frac{1}{2}h(y_n' + y_{n+1}')$ and for $f(x, t) = Ax$, we have $y_{n+1} = T\Xi T^{-1}y_n$, in which Ξ = diag $[(2 - ha_1)(2 + ha_1)^{-1}, \ldots, (2 - ha_N)(2 + ha_N)^{-1}]$($T$ and the a_i are defined in Section I). In view of the relation between the local-truncation errors of the trapezoidal rule and formula (12), this suggests that for nonlinear problems the trapezoidal rule should permit larger step sizes for the same accuracy *when* the "fast components" of the solution have decayed to a very low level.

for example)[4] much work directed toward the comparison of available methods is needed.

REFERENCES

1. Hamming, R. W., *Numerical Methods for Scientists and Engineers,* McGraw-Hill Book Co., New York (1962).
2. Ralston, A., *A First Course in Numerical Analysis,* McGraw-Hill Book Co., New York (1965).
3. Koehler, D., "The Charge-Control Concept in the Form of Equivalent Circuits Representing a Link Between Classic Large Signal Diode and Transistor Models," B.S.T.J., *46,* No. 3 (March 1967), pp. 523–576.
4. Pope, D. A., "An Exponential Method of Numerical Integration of Ordinary Differential Equations," Commun. ACM, *6* (August 1963), pp. 491–493.

Additional References

Branin, F. H., "Computer Methods of Network Analysis," Proc. IEEE, *55,* No. 11 (November 1967), pp. 1787–1801.

Certaine, J., "The Solution of Ordinary Differential Equations with Large Time Constants," chapter 9 of *Mathematical Methods for Digital Computers,* ed. A. Ralston and H. S. Wilf, New York: John Wiley and Sons, 1960.

Dahlquist, C. G., "A Special Stability Problem for Linear Multistep Methods," BIT, *3,* No. 1 (1963), pp. 27–43.

Rosenbrock, H. H., "Some General Implicit Processes for the Numerical Solution of Differential Equations," Computer J., *5,* (January 1963), pp. 329–330.

An Upper Bound on the Zero-Crossing Distribution*

By NICHOLAS A. STRAKHOV and LUDWIK KURZ†

Let $Q(T)$ equal the probability that a random process, $x(t)$, does not cross the zero axis in a given interval of length T. A family of upper bounds for $Q(T)$ is derived with only weak restrictions imposed on $x(t)$ and it is shown that for gaussian random processes only one member of the family provides useful formulae. Specific results are obtained for $x(t)$ representing a number of interesting random processes.

I. INTRODUCTION

Let $Q(T)$ equal the probability that a random process, $x(t)$, does not cross the zero axis in a given interval of length T. The problem of determining $Q(T)$ (and related functions) has important applications in communications theory and has been investigated by many authors.[1–6] Reference 5 gives an extensive bibliography of most of the related work on this subject prior to 1962. Despite all this effort, $Q(T)$ is known only when $x(t)$ is a simple nongaussian process (such as a process whose zero-crossings obey the Poisson distribution) or a stationary gaussian zero-mean process with one of four explicit correlation functions.[5, 6] Most of the rest of the results obtained are either approximate or form upper or lower bounds.[5]

In this paper, we develop a whole family of upper bounds on $Q(T)$. For computational purposes, however, only one member of the family has been found to provide useful results for most cases of interest.

II. DERIVATION OF AN UPPER BOUND ON Q(T)

Consider the transformation

$$z_T = \frac{1}{T}\int_0^T \operatorname{sgn}\,[x(t)]\,dt \tag{1}$$

* An abbreviated version of this paper was presented at the Fifth Allerton Conference on System and Circuit Theory, Monticello, Ill., October 4, 1967.

† New York University.

where $x(t)$ is a sample function of a stochastic process,* T is a fixed observation interval, and z_T is a random variable defined by the stochastic integral (1). The function $sgn[x(t)]$ is defined as

$$\text{sgn}\,[x] = \begin{cases} +1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0. \end{cases}$$

Since z_T is a random variable, it has a cumulative distribution function, $P(z_T)$, associated with it. From (1), two properties of $P(z_T)$ are immediately apparent, regardless of the statistics governing $x(t)$:

(*i*)
$$P(z_T) = 0 \quad \text{for} \quad z_T < -1$$

and

$$P(z_T) = 1 \quad \text{for} \quad z_T > 1 \tag{2}$$

(*ii*)
$$\lim_{\epsilon \to 0} [P(1 + \epsilon) - P(1 - \epsilon)] = Q_U(T) \tag{3}$$

and

$$\lim_{\epsilon \to 0} [P(-1 + \epsilon) - P(-1 - \epsilon)] = Q_L(T) \tag{4}$$

where

$$Q_U(T) = \text{Prob}\,\{x(t) \geqq 0 \quad \text{for} \quad 0 \leqq t \leqq T\} \tag{5}$$

and

$$Q_L(T) = \text{Prob}\,\{x(t) \leqq 0 \quad \text{for} \quad 0 \leqq t \leqq T\}. \tag{6}$$

Obviously, $Q(T)$ as defined previously is related to the last two quantities by

$$Q(T) = Q_U(T) + Q_L(T). \tag{7}$$

If $x(t)$ is a symmetric† process, then

$$Q_U(T) = Q_L(T) = \tfrac{1}{2}Q(T). \tag{8}$$

As a consequence of properties (*i*) and (*ii*), $P(z_T)$ can be represented by

$$P(z_T) = G(z_T) + Q_L(T)u(z_T + 1) + Q_U(T)u(z_T - 1) \tag{9}$$

* Throughout this paper, we assume that almost all sample functions of the stochastic pocess are continuous. Thus, (1), (5), and (6) are well defined.

† The stochastic process $x(t)$ will be called symmetric if the probability measures that govern it also govern the process $-x(t)$.

where $G(z_T)$ is continuous at $z_T = \pm 1$ and

$$u(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geqq 0. \end{cases}$$

We assume throughout that the function $G(z_T)$ is not identically equal to zero. If it were, then it would be easy to show that $Q(T)$ is known exactly; that is, $Q(T) = 1$.

Next, consider the even-order moments of $P(z_T)$, denoted by the Stieltjes integral

$$q_{2k} = \int_{-1}^{1} z_T^{2k}\, dP(z_T) \qquad k = 0, 1, 2, \cdots . \tag{10}$$

By substituting (9) into (10) one obtains

$$q_{2k} = \int_{-1}^{1} z_T^{2k}\, dG(z_T) + Q_U(T) + Q_L(T) \qquad k = 0, 1, 2, \cdots . \tag{11}$$

Neglecting the first term in the right side of (11) (which is always positive) and taking (7) into account leads to a family of upper bounds for $Q(T)$ expressed by

$$Q(T) \leqq q_{2k} \qquad k = 0, 1, 2, \cdots . \tag{12}$$

For $k = 0$, (12) reduces to the obvious result

$$Q(T) \leqq 1.$$

Before discussing the usefuless of the inequality (12), an expression for the moments will be derived.

From its definition, (10), q_{2k} can be expressed as

$$q_{2k} = E\{z_T^{2k}\}$$

where $E\{\cdot\}$ denotes the expected value of the quantity enclosed in braces. Substitution for z_T from (1) results in

$$q_{2k} = \frac{1}{T^{2k}} E\left\{\int_0^T \int_0^T \cdots \int_0^T y(t_1)y(t_2) \cdots y(t_{2k})\, dt_1\, dt_2 \cdots dt_{2k}\right\} \tag{13}$$

where

$$y(t_i) = \operatorname{sgn}[x(t_i)] \qquad i = 1, 2, \cdots, 2k.$$

Interchanging the order of integration and expectation yields

$$q_{2k} = \frac{1}{T^{2k}} \int_0^T \int_0^T \cdots \int_0^T R(t_1, t_2, \cdots, t_{2k})\, dt_1\, dt_2 \cdots dt_{2k} \tag{14}$$

where

$$R(t_1, t_2, \cdots, t_{2k}) = E\{y(t_1)y(t_2) \cdots y(t_{2k})\}. \tag{15}$$

We now make some remarks concerning the ordering of the family of inequalities expressed by (12).

Denote the first term in the right side of (11) by s_{2k}, or

$$s_{2k} = \int_{-1}^{1} z_T^{2k}\, dG(z_T). \tag{16}$$

We next establish that $s_{2k} > s_{2k+2}$ $(k = 0, 1, 2, \ldots)$ which in turn establishes

$$1 > q_2 > q_4 \cdots > Q(T). \tag{17}$$

The former inequality follows directly from

$$\begin{aligned} s_{2k+2} &= \int_{-1}^{1} z^2 z^{2k}\, dG(z_T) \\ &\leqq \int_{-1}^{1} z^{2k}\, dG(z_T) \\ &= s_{2k} \end{aligned}$$

with equality if, and only if, $G(z_T)$ is of the form

$$G(z_T) = Au(z_T + 1) + Bu(z_T - 1). \tag{18}$$

Since $G(z_T)$ is continuous at $z_T = \pm 1$, equality is not possible and therefore

$$s_{2k+2} < s_{2k}$$

which, together with (11), establishes (17). We next establish the readily proven fact that

$$\lim_{k\to\infty} s_{2k} < \epsilon, \qquad \epsilon > 0 \tag{19}$$

and therefore,

$$\lim_{k\to\infty} q_{2k} = Q(T).$$

We begin by choosing an $\alpha(k_0) > 0$ such that

$$\int_{-1}^{-1+\alpha(k_0)} z_T^{2k_0}\, dG(z_T) < \frac{\epsilon}{2}, \qquad \epsilon > 0.$$

This can always be done because $G(z_T)$ is continuous at $z_T = -1$. Using the definition of s_{2k},(16), obvious symmetry properties, and

the fact that

$$\int_{-1}^{-1+\alpha(k_0)} z_T^{2k}\, dG(z_T) < \int_{-1}^{-1+\alpha(k_0)} z_T^{2k_0}\, dG(z_T)$$

for each $k > k_0$, it follows immediately that

$$\lim_{k\to\infty} \int_{-1}^{1} z_T^{2k}\, dG(z_T) < \epsilon + \lim_{k\to\infty} \int_{-1+\alpha(k_0)}^{1-\alpha(k_0)} z_T^{2k}\, dG(z_T).$$

Since the sequence of functions $\{z_T^{2k}\}$, $k = 0, 1, 2, \cdots$ is uniformly convergent to zero on the interval $[-1 + \alpha(k_0), 1 - \alpha(k_0)]$, the limit and the integral may be interchanged yielding (19).

In light of (17) and (19), it appears that (12) should be evaluated for as large a value of k as possible. For the special case when $x(t)$ is a stationary gaussian random process (assumed to be zero-mean without loss of generality), it does not seem to be possible to evaluate q_{2k} for $k > 1$ as evidenced by the following discussion.

As shown by McFadden,[7] the quantity $R(t_1, t_2, \cdots, t_n)$, defined in (15), is equal to the sum of some simple terms plus a quantity $P_n(\mathbf{r})$, which is defined as

$$P_n(\mathbf{r}) = (2\pi)^{-n/2} |\mathbf{r}|^{-\frac{1}{2}} \int_0^\infty dx_1 \cdots \int_0^\infty dx_n \exp\left[-\frac{1}{2} \sum_{i,j}^{n} r_{ij}^{-1} x_i x_j\right]$$

where $\mathbf{r}$ is a covariance matrix with elements

$r_{ij} = r(t_i - t_j) = E\{x(t_i)x(t_j)\}, \qquad i, j = 1, \cdots, n$

$|\mathbf{r}|$ is the determinant of $\mathbf{r}$

$\sum_{i,j}^{n} r_{ij}^{-1} x_i x_j$ is the quadratic form associated with the inverse of $\mathbf{r}$

and

$$x_i = x(t_i), \qquad i = 1, 2, \cdots, n$$

In other words, $P_n(\mathbf{r})$ is the probability that the n jointly distributed gaussian random variables, $x(t_i)$ $(i = 1, \cdots, n)$ are all positive.

As discussed by McFadden,[7] and even more thoroughly by Slepian,[5] expressions for $P_n(\mathbf{r})$ have not been obtained in terms of elementary functions for $n > 3$. Because of this fact, it seems unlikely that an expression for (14) with $k > 1$ can be obtained for a general gaussian process, $x(t)$. It should be pointed out, however, that an expression for q_4 has been derived[8] for $\rho(\tau) = \exp(-|\tau|)$, but without first evaluating $P_4(\mathbf{r})$. This result is not included because, for this correlation function, $Q(T)$ is known exactly.[5]

III. APPLICATION TO GAUSSIAN RANDOM PROCESS

Assume that $x(t)$ in (1) is a stationary, zero-mean, gaussian random process, normalized so that $\rho(0) = 1$ where $\rho(\tau) = E\{x(t)x(t+\tau)\}$. The relationship (14) will be evaluated for the case $k = 1$, that is, for

$$q_2 = \frac{1}{T^2}\int_0^T \int_0^T R(t_1 , t_2)\, dt_1\, dt_2 \tag{20}$$

where

$$R(t_1 , t_2) = E\{\operatorname{sgn} x(t_1) \operatorname{sgn} x(t_2)\}. \tag{21}$$

The latter expression has been evaluated by many authors (see page 58 of Lawson and Uhlenbeck's book,[9] for example) and the result is

$$R(t_1 , t_2) = \frac{2}{\pi}\sin^{-1}[\rho(t_1 - t_2)]. \tag{22}$$

Substituting (22) into (20) and making the obvious simplifications in integration results in

$$q_2 = \frac{4}{\pi}\int_0^1 (1 - u)\sin^{-1}[\rho(Tu)]\, du,$$

or, in light of (12),

$$Q(T) \leqq \frac{4}{\pi}\int_0^1 (1 - u)\sin^{-1}[\rho(Tu)]\, du. \tag{23}$$

This result has been obtained by Slepian,[5] who states it as Theorem 5.* Slepian's proof, however, is long and complicated, as opposed to the simplicity of the proof given here. Furthermore, extensions to other cases can be obtained using the new method.

IV. APPLICATION TO SINE WAVE PLUS GAUSSIAN RANDOM PROCESS

We now turn to applying (12) with $k = 1$ to the case where

$$x(t) = w(t) + A\cos(2\pi ft + \varphi) \tag{24}$$

where

$w(t)$ is a stationary, zero-mean, gaussian random process with normalized correlation function, $\rho(\tau)$,
φ is a random phase constant uniformly distributed on $[0, 2\pi]$,
f is the sine-wave frequency, and
A is the sine-wave amplitude.

* Notice that Slepian's $P[T, r(\tau)]$ equals one half of our $Q(T)$.

As derived in the next subsection, the result obtained is

$$Q(T) \leqq q_2^{(1)} + q_2^{(2)} \tag{25}$$

where

$$q_2^{(2)} = 2\int_0^1 (1 - u)H(uT)\,du \tag{26}$$

with

$$\begin{aligned} H(uT) = \frac{2}{\pi}\int_0^{\sin^{-1}\rho(uT)} &\exp\left\{-\frac{A^2}{2}\frac{1 - \sin\theta\cos 2\pi bu}{\cos^2\theta}\right\} \\ &\cdot I_o\left\{\frac{A^2}{2}\frac{\sin\theta - \cos 2\pi bu}{\cos^2\theta}\right\} d\theta \\ b &= fT \end{aligned} \tag{27}$$

and $I_o(x)$ = modified Bessel function of the first kind, zero order.

The expression given below for $q_2^{(1)}$ is approximate, except for $T = k/f$, $k = 0, 1, 2, \cdots$ where it is exact and consequently the function is most accurate in a neighborhood of these points. In addition, the accuracy of the approximation improves as A increases. For small A, where the approximation is least accurate, (26) dominates $q_2^{(1)}$ and so very little error results in the upper bound, (25) for all values of A. The expression for $q_2^{(1)}$ is given by

$$q_2^{(1)} \cong \frac{2}{b^2} \times \begin{cases} \displaystyle\int_0^{b-n} (b - n - v)S(v)\,dv, & n \leqq b < n + \tfrac{1}{4} \\ \displaystyle -\int_{n+\frac{1}{2}-b}^{\frac{1}{4}} (b - n - \tfrac{1}{2} + v)S(v)\,dv & \\ \quad \displaystyle + \int_0^{\frac{1}{4}} (b - n - v)S(v)\,dv, & n + \tfrac{1}{4} \leqq b < n + \tfrac{1}{2} \\ \displaystyle -\int_0^{b-n-\frac{1}{2}} (b - n - \tfrac{1}{2} - v)S(v)\,dv & \\ \quad \displaystyle + 2\int_0^{\frac{1}{4}} (\tfrac{1}{4} - v)S(v)\,dv, & n + \tfrac{1}{2} \leqq b < n + \tfrac{3}{4} \\ \displaystyle \int_{n+1-b}^{\frac{1}{4}} (b - n - 1 + v)S(v)\,dv & \\ \quad \displaystyle - \int_0^{\frac{1}{4}} (b - n - 1 + v)S(v)\,dv, & n + \tfrac{3}{4} \leqq b < n + 1 \end{cases} \tag{28}$$

where

$$n = 0, 1, 2, \cdots$$

$$S(v) = (1 - 4v) - \frac{e^{-A^2/2}}{\pi} G(v) \tag{29}$$

$$G(v) = \int_0^{\pi - 2\pi v} e^{-K(v)\cos x}\, dx - \int_0^{2\pi v} e^{K(v)\cos x}\, dx \tag{30}$$

and

$$K(v) = \frac{A^2}{2} \cos 2\pi v. \tag{31}$$

While (28) appears to be a formidable equation, it turns out to be easily computed, partly because $S(v)$ does not depend on b and hence needs only to be computed once for each value of A. The expression (26), on the other hand, turns out to be time-consuming to compute, particularly for large values of b.

4.1 *Derivation of Upper Bound, Given by (25)*

The expression for q_2, (14), with $x(t)$ specified by (24) is

$$q_2 = \frac{1}{T^2} \int_0^T \int_0^T R(t_1, t_2)\, dt_1\, dt_2 \tag{32}$$

with $R(t_1, t_2)$ given by (15). Notice that the expectation in this case ranges over the three random variables, $w(t_1)$, $w(t_2)$ and φ. For convenience, define

$$R(t_1, t_2) = \frac{1}{2\pi} \int_{-\pi}^{\pi} r(t_1, t_2)\, d\varphi \tag{33}$$

where

$$r(t_1, t_2) = E\{\operatorname{sgn}[w_1 + a_1] \operatorname{sgn}[w_2 + a_2]\} \tag{34}$$

and

$$w_i = w(t_i)$$

$$a_i = A \cos(2\pi f t_i + \varphi) \qquad i = 1, 2.$$

The latter expectation is with respect to w_1 and w_2 only. Writing out (34) in terms of the definition of $E\{\cdot\}$ results in

$$r(t_1, t_2) = \frac{1}{2\pi[1 - \rho^2(\tau)]^{\frac{1}{2}}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \operatorname{sgn}[w_1 + a_1] \operatorname{sgn}[w_2 + a_2] \cdot \exp\left\{-\frac{w_1^2 + w_2^2 - 2\rho(\tau) w_1 w_2}{2[1 - \rho^2(\tau)]}\right\} dw_1\, dw_2 \tag{35}$$

where $\tau = t_2 - t_1$.

Applying Price's theorem,[10] to this equation results in

$$\frac{\partial r(t_1 , t_2)}{\partial \rho(\tau)} = \frac{2}{\pi[1 - \rho^2(\tau)]^{\frac{1}{2}}} \exp\left\{-\frac{a_1^2 + a_2^2 - 2\rho(\tau)a_1a_2}{2[1 - \rho^2(\tau)]}\right\}. \tag{36}$$

Integrating (36) and applying the appropriate boundary condition yields

$$r(t_1 , t_2) = r(t_1 , t_2)|_{\rho(\tau)=0} + \int_0^{\rho(\tau)} \frac{\partial r(t_1 , t_2)}{\partial \rho(\tau)}\, d\rho$$

or,

$$r(t_1 , t_2) = 4 \operatorname{erf}(a_1) \operatorname{erf}(a_2) + \frac{2}{\pi}\int_0^{\rho(\tau)} \frac{1}{(1 - \alpha^2)^{\frac{1}{2}}} \exp\left\{-\frac{a_1^2 + a_2^2 - 2\alpha a_1a_2}{2[1 - \alpha^2]}\right\} d\alpha \tag{37}$$

where

$$\operatorname{erf}(x) = \frac{1}{(2\pi)^{\frac{1}{2}}}\int_0^x e^{-y^2/2}\, dy.$$

As a result of the natural separation of (37) into the sum of two quantities, define

$$q_2 = q_2^{(1)} + q_2^{(2)} \tag{38}$$

where, by substituting (33) and (37) into (32), the terms in (38) may be defined as

$$q_2^{(1)} = \frac{2}{\pi T^2}\int_0^T \int_0^T \int_{-\pi}^{\pi} \operatorname{erf}(a_1) \operatorname{erf}(a_2)\, d\varphi\, dt_1\, dt_2 \tag{39}$$

$$q_2^{(2)} = \frac{1}{\pi^2 T^2}\int_0^T \int_0^T \int_{-\pi}^{\pi} \int_0^{\rho(\tau)} \frac{1}{(1 - \alpha^2)^{\frac{1}{2}}} \cdot \exp\left\{-\frac{a_1^2 + a_2^2 - 2\alpha a_1a_2}{1 - \alpha^2}\right\} d\alpha\, d\varphi\, dt_1\, dt_2 . \tag{40}$$

The detailed steps of simplifying (39) and (40) are relegated to Appendices A and B, respectively. In Appendix A we discuss the nature of the approximation made in arriving at (28).

Before applying the results just obtained to specific situations, a power series representation for (32) will be given. The series may be derived from (39) and (40) by expanding the integrands of these functions in their respective Taylor series, evaluating the resulting terms and adding the expansions for (39) and (40) together. This

procedure results in

$$q_2 = \frac{4}{\pi}\left\{\int_0^1 (1-u)\sin^{-1}\rho(Tu)\,du + \frac{A^2}{2}\int_0^1 \frac{(1-u)[\cos 2\pi fTu - \rho(Tu)]}{\sqrt{1-\rho^2(u)}}\,du + 0(A^4)\right\}.$$

4.2 *Numerical Results and Comparisons*

We evaluated the inequality (25) with the aid of a digital computer. For the first case considered, $\rho(\tau) = e^{-|\tau|}$, $f = 2$ Hz, T ranged between 0 and 3.5 seconds and A, the sine-wave amplitude, was either 1 or 10. The results of these computations are plotted in Fig. 1.

Let us first discuss the $A = 10$ case. The quantity $q_2^{(1)}(T)$ exhibits a damped oscillatory behavior much like a plot of $[\sin(\pi fT)/(\pi fT)]^2$ while the quantity $q_2^{(2)}(T)$ decays toward zero quite smoothly. For

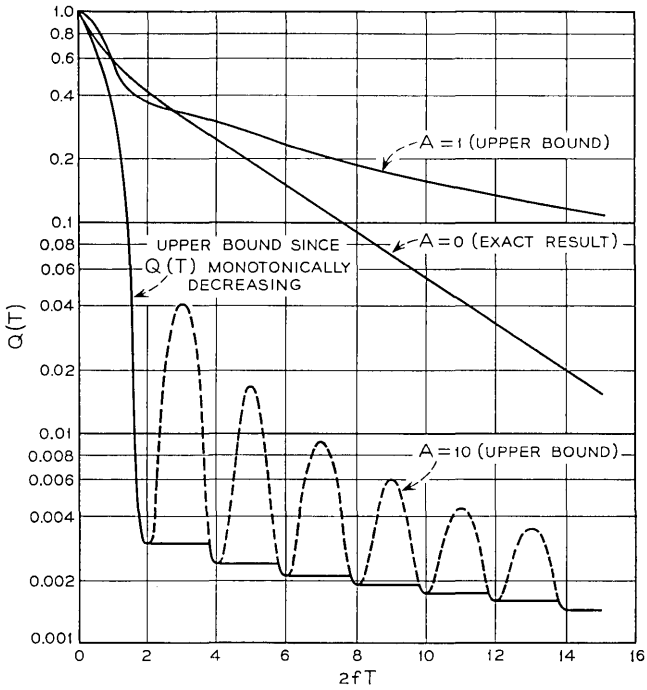


Fig. 1 — Upper bound of $Q(T)$ for zero crossings of sine wave plus gaussian noise.

large A, the former term dominates except at its zeros located at $T = k/f$, $k = 1, 2, \cdots$. The values of $q_2(T)$ are shown as dotted lines on Fig. 1. Since $Q(T)$ is a monotonically decreasing function, its upper bound can be constructed by drawing horizontal lines between the local minima of $q_2(T)$ and the first intersection of this line with $q_2(T)$ to the right of the minimum. This accounts for the step-like curve drawn in Fig. 1 representing the upper bound for $Q(T)$ with $A = 10$.

The curve representing $A = 1$, while not exhibiting as fast a decay as the curve for $A = 10$, shows some interesting features. As contrasted to the last case, the $q_2^{(2)}(T)$ term dominates the $q_2^{(1)}(T)$ term and consequently much of the oscillatory behavior noted earlier has disappeared.

Another interesting observation can be made when the $A = 1$ curve is compared with the $A = 0$ curve (gaussian noise alone) for which $Q(T)$ is known to equal $(2/\pi) \sin^{-1} (e^{-T})$ when $\rho(\tau) = e^{-|\tau|}$. (See Reference 5.) Notice that for $0 < T < 0.25$ the $A = 1$ curve lies above the $A = 0$ curve while for $0.25 < T < 0.75$ the reverse is true.

This result can be explained by recalling that $T = 0.25$ represents one-half the period of $\cos(4\pi t)$. For intervals shorter than this, the sine wave is not likely to cross zero and the effect is to cause fewer zero crossings than would be obtained if the sine wave were absent. Conversely, for time intervals longer than one-half the period ($T = 0.25$ in this case), the sine wave is sure to cross zero and therefore tend to increase the number of zero crossings over the noise-alone case.

As a result of this observation, it seems reasonable to conjecture that for T greater than one half the sine-wave period $Q(T)$, for noise alone, also forms an upper bound to $Q(T)$ for the sum of a sine wave plus noise.

Additional calculations were made for comparison with Cobb's previously reported approximate results.[1] The quantity that Cobb derived is an approximate expression for the probability distribution function of zero-crossing intervals, denoted by $P_0(T)$. Rice gives the relationship between $Q(T)$ and $P_0(T)$ in Reference 4 as

$$Q(T) = 1 - 2\nu T + 2\nu \int_0^T \int_0^x P_0(t)\, dt\, dx \tag{41}$$

where ν = expected number of zero crossings of (24).

As observed in Fig. 1 and 2 of Reference 1,

$$\nu \cong f$$

for the large sine-wave amplitudes where Cobb's approximation is valid. Thus,

$$Q(T) \cong 1 - 2fT + \int_0^{2fT} \int_0^{v} P_0(s)\, ds\, dy. \tag{42}$$

Cobb shows in equation 52 of Reference 1 that

$$P_0(s) \cong \frac{1}{(2\pi\sigma)^{\frac{1}{2}}} \exp\left[- \frac{(s-1)^2}{\sigma^2} \right] \tag{43}$$

where

$$\sigma = \frac{[2(1 + \rho_1)]^{\frac{1}{2}}}{\pi A}$$

$$\rho_1 = \rho(2fT).$$

The approximation (43) is only valid for $\sigma \ll 1$.

Substituting (43) into (42), we obtain

$$Q(T) \cong (1 - 2fT)\left[0.5 + \operatorname{erf}\left(\frac{1 - 2fT}{\sigma} \right) \right] + \frac{\sigma}{(2\pi)^{\frac{1}{2}}} \left\{ \exp\left[- \frac{1}{2} \left(\frac{1 - 2fT}{\sigma} \right)^2 \right] - \exp\left(- \frac{1}{2\sigma^2} \right) \right\}. \tag{44}$$

As in Reference 1, set

$$\rho(\tau) = \frac{\sin \tau}{\tau} \tag{45}$$

$$A = 3 \tag{46}$$

$$2\pi f = 0.875 \text{ rad/sec}. \tag{47}$$

Figure 2 compares the approximate solution based on Cobb's results (44), and our upper bound (25). For $2fT < 1$, the approximate solution is somewhat smaller than the upper bound. For $2fT > 1$, the approximation becomes negative and therefore of little interest while the upper bound gradually approaches zero as T increases.

V. EXTENSIONS TO OTHER CASES

The specific applications discussed should not be considered exhaustive. For example, the case where $x(t)$ is the sum of a sine wave plus gaussian noise could easily be extended to $x(t)$ being the sum of a square wave plus gaussian noise. Although the specific formulae may be more complex, the general result (equations 12 and 14) is still applicable for $x(t)$ nonstationary or nongaussian.

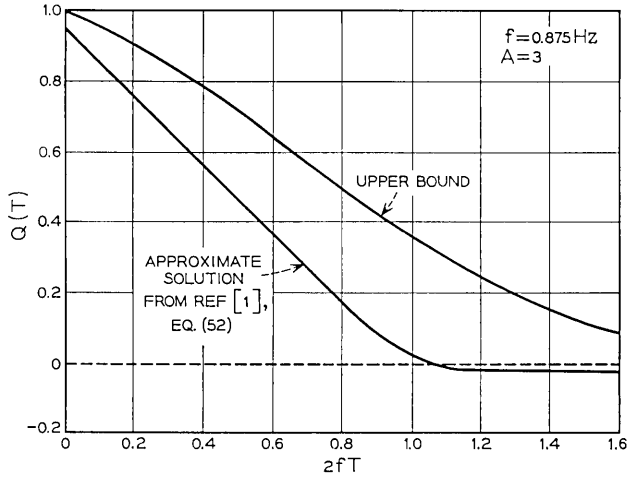


Fig. 2 — Comparison of upper bound and approximate solution of $Q(T)$ for zero crossings of sine wave plus gaussian noise.

In addition, the derivation of the general result can be modified slightly to obtain a useful upper bound on the conditional probability that $x(t)$ does not cross the zero axis for an interval of length T, given that $x(t) = 0$ at the start of the interval. Slepian has intensively investigated this latter probability. See Reference 5 for his discussion.

ACKNOWLEDGMENT

The research reported here was sponsored in part by the Doctoral Support Plan of Bell Telephone Laboratories, while the first-named author was on assignment from Bell Laboratories to New York University; and in part by the National Science Foundation under grant GK-1075. The paper is based on a portion of a dissertation submitted to the University's Graduate Division of The School of Engineering and Science in partial fulfillment of the requirements for the Ph.D. degree.

APPENDIX A

Derivation of $q_2^{(1)}$

We seek a simpler expression for the term

$$q_2^{(1)} = \frac{2}{\pi T^2} \int_0^T \int_0^T \int_{-\pi}^{\pi} \operatorname{erf}(a_1) \operatorname{erf}(a_2)\, d\varphi\, dt_1\, dt_2 \tag{48}$$

which was first encountered in (39) and where

$$a_i = A \cos (2\pi f t_i + \varphi), \qquad i = 1, 2. \tag{49}$$

Substitution of the definition of $\operatorname{erf}(x)$ results in

$$q_2^{(1)} = \frac{1}{\pi^2 T^2} \int_0^T \int_0^T P(t_1 , t_2)\, dt_1\, dt_2 \tag{50}$$

where

$$P(t_1 , t_2) = \int_{-\pi}^{\pi} \int_0^{a_1} \int_0^{a_2} \exp [-\tfrac{1}{2}(x^2 + y^2)]\, dx\, dy\, d\varphi. \tag{51}$$

The first step is to notice the following three easily established properties of (51):

$$P(t_1 , t_2) = P(t_1 - t_2) = P(t_2 - t_1) \tag{52}$$

$$P(\tau) = P\left(\tau + \frac{n}{f}\right), \qquad n = \pm 1, \pm 2, \cdots \tag{53}$$

$$P\left(\tau + \frac{1}{4f}\right) = -P\left(\frac{1}{4f} - \tau\right). \tag{54}$$

As a result of (52), (50) may be written as

$$q_2^{(1)} = \frac{2}{\pi^2 T^2} \int_0^T (T - \tau) P(\tau)\, d\tau. \tag{55}$$

It is a simple matter to demonstrate that, for any function, $H(\tau)$, satisfying the requirements of (52) through (54),

$$\int_{i/f}^{(i+1)/f} (T - \tau) H(\tau)\, d\tau = 0. \tag{56}$$

We next introduce an approximation to (51) that preserves properties (52) through (54). It is important to preserve these properties because, as a result of (56), if they are satisfied, $q_2^{(1)} = 0$ for $T = k/f$, $k = 1, 2, \cdots$; consequently, an approximation satisfying (52) through (54) will be accurate in a vicinity of these values of T. In addition, the three properties permit fast computation of (50).

The approximation chosen is given by

$$\int_0^{a_1} \int_0^{a_2} \exp [-\tfrac{1}{2}(x^2 + y^2)]\, dx\, dy \cong \int_0^{\pi/2} \int_0^{(a_1^2 + a_2^2)^{\frac{1}{2}}} e^{-r^2/2} r\, dr\, d\alpha \tag{57}$$

for sgn $[a_1]$ = sgn $[a_2]$, and

$$\int_0^{a_1}\int_0^{a_2} \exp\left[-\tfrac{1}{2}(x^2+y)\right] dx\,dy \cong -\int_0^{\pi/2}\int_0^{(a_1^2+a_2^2)^{\frac{1}{2}}} e^{-r^2/2} r\,dr\,d\alpha \quad (58)$$

for sgn $[a_1]$ = $-$sgn $[a_2]$.

Essentially the approximation results in deforming the region of integration, as shown in Fig. 3. From this figure, it may be noticed that (57) is in reality an upper bound while (58) is a lower bound. Of course, it is easy to conceive of functions that give an upper bound to (58) and thus result in an upper bound for $q_2^{(1)}$. However, this results in a loss of properties (52) through (54).

Evaluating the integrals appearing in (57) and (58) and then substituting into (51) yields

$$\hat{P}(t_1, t_2) = \frac{\pi}{2}\int_{-\pi}^{\pi} p(\varphi, t_1, t_2)[1 - e^{-\frac{1}{2}(a_1^2+a_2^2)}]\,d\varphi \quad (59)$$

where

$$p(\varphi, t_1, t_2) = \text{sgn}\,[a_1\,a_2] \quad (60)$$

and

$$\hat{P}(t_1, t_2) \cong P(t_1, t_2).$$

Proving that (59) possesses the properties (52) through (54) only requires the use of elementary integration theory and will therefore be omitted. As a result of these properties, (59) may be written as

$$\hat{P}(\tau) = \frac{\pi}{2}\int_{-\pi}^{\pi} p(\theta, \tau)\{1 - e^{-(A^2/2)[\cos^2\theta + \cos^2(\omega\tau+\theta)]}\}\,d\theta \quad (61)$$

where $\omega = 2\pi f$

$$p(\theta, \tau) = \text{sgn}\,[\cos\theta\cos(\omega\tau + \theta)]; \quad (62)$$

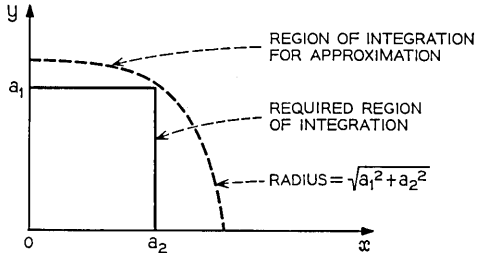


Fig. 3 — Deformation of region of integration for approximation in Appendix A.

furthermore, $\hat{P}(\tau)$ need only be evaluated for the range $0 < \tau < 1/(4f)$. Values beyond this range are related to values within the range by (52) through (54). To evaluate (61), an explicit expression for (62) is required. After studying this latter equation one finds that for

$$0 < \tau < 1/2f$$

$$p(\theta, \tau) = \begin{cases} 1 \quad \text{for} \quad \dfrac{\pi}{2} < \theta \leqq \dfrac{3\pi}{2} - \omega\tau \\ -1 \quad \text{for} \quad \dfrac{\pi}{2} - \omega\tau < \theta \leqq \dfrac{\pi}{2} \\ 1 \quad \text{for} \quad -\dfrac{\pi}{2} < \theta \leqq \dfrac{\pi}{2} - \omega\tau \\ -1 \quad \text{for} \quad -\dfrac{\pi}{2} - \omega\tau < \theta \leqq -\dfrac{\pi}{2} \end{cases} \tag{63}$$

with similar expressions for τ falling in the ranges

$$\frac{k}{2f} \leqq \tau \leqq \frac{k+1}{2f}, \qquad k = 1, 2, \cdots .$$

However, only the expression given by (63) is needed to evaluate (61) in the required range.

Substitution of (63) into (61) results in

$$\hat{P}(\tau) = \frac{\pi}{2}\left[\int_{-\pi/2}^{(\pi/2)-\omega\tau} F(\tau, \theta)\, d\theta + \int_{\pi/2}^{(3\pi/2)-\omega\tau} F(\tau, \theta)\, d\theta - \int_{-(\pi/2)-\omega\tau}^{-\pi/2} F(\tau, \theta)\, d\theta - \int_{(\pi/2)-\omega\tau}^{\pi/2} F(\tau, \theta)\, d\theta\right] \tag{64}$$

where

$$F(\tau, \theta) = 1 - e^{-(A^2/2)[\cos^2\theta + \cos^2(\omega\tau+\theta)]}.$$

With the help of some fundamental trigonometric identities it is easy to show that

$$F(\tau, \theta) = 1 - e^{-A^2/2} e^{-K(\tau)\cos(\omega\tau+2\theta)} \tag{65}$$

where

$$K(\tau) = \frac{A^2}{2}\cos\omega\tau. \tag{66}$$

Substituting (65) into (64) and performing obvious simplifications results in

$$\hat{P}(\tau) = \frac{\pi}{2}\{2\pi - 4\omega\tau - 2e^{-A^2/2}G(\tau)\} \tag{67}$$

where

$$G(\tau) = \int_0^{\pi-\omega\tau} e^{-K(\tau)\cos x}\,dx - \int_0^{\omega\tau} e^{K(\tau)\cos x}\,dx. \tag{68}$$

For the time being assume $n \leqq fT < n + \frac{1}{4}$ where $n = 0, 1, 2, \cdots$. Substituting (67) into (55) gives

$$q_2^{(1)} \cong \frac{2}{T^2}\int_0^T (T - \tau)S(\tau)\,d\tau$$

where

$$S(\tau) = 1 - 4f\tau - \frac{e^{-A^2/2}}{\pi}G(\tau). \tag{69}$$

Using the result (56), the latter equation equals

$$q_2^{(1)} \cong \frac{2}{T^2}\int_{n/f}^T (T - \tau)S(\tau)\,d\tau.$$

Setting $t = \tau - n/f$,

$$q_2^{(1)} \cong \frac{2}{T^2}\int_0^{T-(n/f)} \left(T - \frac{n}{f} - t\right)S\left(t + \frac{n}{f}\right)dt$$

$$= \frac{2}{T^2}\int_0^{T-(n/f)} \left(T - \frac{n}{f} - t\right)S(t)\,dt$$

as a result of property (53). Now substitute $t = v/f$ to obtain

$$q_2^{(1)} \cong \frac{2}{b^2}\int_0^{b-n} (b - n - v)S(v)\,dv \quad \text{for} \quad n \leqq b < n + \tfrac{1}{4} \tag{70}$$

where $b = fT$.

This equation is the same as the first part of the final result stated in (28). The equations defined in (69), (68), and (66) are the same as (29), and (30), and (31), respectively, except for a convenient scale change. The rest of the results stated in (28), for various ranges of T, are derived in a similar manner as (70) was, using relations (52), (53), or (54), as required. Because only straightforward operations are used to obtain these results, they will not be derived.

APPENDIX B

Derivation of $q_2^{(2)}$

The first step in simplifying the expression for $q_2^{(2)}$, as defined in (40), is to interchange the order of integration of the two innermost integrals to yield

$$q_2^{(2)} = \frac{1}{\pi^2 T^2} \int_0^T \int_0^T \int_0^{\rho(\tau)} \frac{1}{(1-\alpha^2)^{\frac{1}{2}}} B(\alpha, t_1, t_2)\, d\alpha\, dt_1\, dt_2 \tag{71}$$

where

$$B(\alpha, t_1, t_2) = \int_{-\pi}^{\pi} \exp\left[-\frac{a_1^2 + a_2^2 - 2\alpha a_1 a_2}{2(1-\alpha^2)}\right] d\varphi. \tag{72}$$

Using the definitions of a_1 and a_2, (34), and some obvious trigonometric identities, it is easy to show that

$$a_1^2 + a_2^2 - 2\alpha a_1 a_2 = A^2\{\cos[\omega(t_1 + t_2) + 2\varphi][\alpha - \cos(\omega\tau)] + [1 - \alpha\cos(\omega\tau)]\}$$

where we have set $\omega = 2\pi f$ for convenience.

Substitution of this relationship into (72) results in

$$B(\alpha, t_1, t_2) = \exp[-J_1(\alpha, \tau)] \cdot \int_{-\pi}^{\pi} \exp\{-J_2(\alpha, \tau)\cos[\omega(t_1 + t_2) + 2\varphi]\}\, d\varphi \tag{73}$$

where

$$J_1(\alpha, \tau) = \frac{A^2}{2}\left[\frac{1 - \alpha\cos(\omega\tau)}{1 - \alpha^2}\right] \tag{74}$$

$$J_2(\alpha, \tau) = \frac{A^2}{2}\left[\frac{\alpha - \cos(\omega\tau)}{1 - \alpha^2}\right]. \tag{75}$$

Setting $\theta = \omega(t_1 + t_2) + 2\varphi$ in (73) and using the periodic properties of the integrand, yields

$$B(\alpha, \tau) = 2\exp[-J_1(\alpha, \tau)] \int_0^{\pi} \exp[-J_2(\alpha, \tau)\cos\theta]\, d\theta.$$

This integral is recognized as an expression for the modified Bessel function of the first kind (see Reference 11, page 181, Equation 4). And so

$$B(\alpha, \tau) = 2\pi I_o[J_2(\alpha, \tau)]\exp[-J_1(\alpha, \tau)] \tag{76}$$

where $I_o(x)$ is the modified Bessel function of the first kind, zero order.

Since (76) is only a function of τ, one may define

$$H(\tau) = \frac{2}{\pi} \int_0^{\rho(\tau)} \frac{1}{\sqrt{1 - \alpha^2}} B(\alpha, \tau)\, d\alpha. \tag{77}$$

Furthermore, it is easy to show that $H(\tau) = H(-\tau)$. Consequently, (71) can be written as

$$q_2^{(2)} = 2 \int_0^1 (1 - u) H(uT)\, du.$$

By setting $\tau = uT$ in (77) and by making the change of variable $\alpha = \sin \theta$, (26) and (27), which are the desired results, follow.

REFERENCES

1. Cobb, S. M., "The Distribution of Intervals Between Zero Crossings of Sine Wave Plus Random Noise and Allied Topics," IEEE Trans. Inform. Theory, *IT-11,* No. 2 (April 1965), pp. 220–231.
2. Newell, G. F., and Rosenblatt, M., "Zero-Crossing Probabilities for Gaussian Stationary Processes," Ann. Math. Stat., *33,* No. 4 (December 1962), pp. 1306–1313.
3. Rice, S. O., "Mathematical Analysis of Random Noise," B.S.T.J., *23,* No. 3 (July 1944), pp. 282–332; and *24,* No. 1 (January 1945), pp 46–156. See especially Sections 3.3 and 3.4.
4. Rice, S. O., "Distribution of the Duration of Fades in Radio Transmission," B.S.T.J., *37,* No. 3 (May 1958), pp. 581–636.
5. Slepian, D., "The One-Sided Barrier Problem for Gaussian Noise," B.S.T.J., *41,* No. 2 (March 1962), pp. 462–501.
6. Wong, E., "Some Results Concerning the Zero-Crossings of Gaussian Noise," SIAM J. Appl. Math., *14,* No. 6 (November 1966), pp. 1246–1254.
7. McFadden, J. A., "Urn Models of Correlation and a Comparison with the Multivariate Normal Integral," Ann. Math. Stat., *26,* No. 3 (September 1955), pp. 478–489.
8. Strakhov, N. A., "A Representation Theory for a Class of Non-Gaussian Distributions with Applications to Digital Data Transmission," New York: Doctoral Dissertation, New York University, October 1967.
9. Lawson, J. L., and Uhlenbeck, G. E., *Threshold Signals,* New York: McGraw-Hill Book Co., Inc., 1950.
10. Price, R., "A Useful Theorem for Nonlinear Devices Having Gaussian Inputs," IRE Trans. Inform. Theory, *IT-4,* No. 2 (June 1958), pp. 69–72.
11. Watson, G. N., *A Treatise on the Theory of Bessel Functions,* 2nd ed., Cambridge, England: University Press, 1962.

Adaptive Redundancy Removal in Data Transmission

By R. W. LUCKY

This paper suggests an adaptive filter, similar to that used in automatic equalization, for use as a predictor in data compression systems. It discusses some of the applications of this adaptive predictor in digital data transmission. In the event of redundant data input to the system the predictor could be used to lower the transmitted power output required for a given error rate or to decrease the error rate while maintaining constant transmitted power. The action of these redundancy-removal and restoration systems is analyzed in simple cases involving Markov inputs.

I. INTRODUCTION

In the design, analysis, and testing of data transmission systems it is invariably assumed that the input digits are identically distributed, independent random variables. However, in many actual systems the input digits may arise from a physical source which imposes significant correlations in the data train. In these cases we know that the entropy of the source is less than when independent digits are presented. Accordingly, we should be able to use the redundancy in the input message to provide, in some sense, more efficient transmission. For example, we could imagine the redundancy being used to decrease bandwidth, to increase speed, to lower probability of error, or to lower average signal power.

Redundancy removal in analog transmission systems was investigated in the early 1950's by Oliver, Kretzmer, Harrison, and Elias[1-4]. Each of these papers relied on the theory of linear prediction as developed by Wiener in the early 1940's.[5] Figure 1 shows the basic idea. It is assumed that the input samples are taken from a stationary time series $\{x_n\}$. These samples are passed through a linear filter whose output $\hat{x}_n$ at time t_n forms a linear prediction of the sample x_n based on all preceding samples. The prediction $\hat{x}_n$ is subtracted from the actual sample x_n and only the error e_n is passed on for further processing and

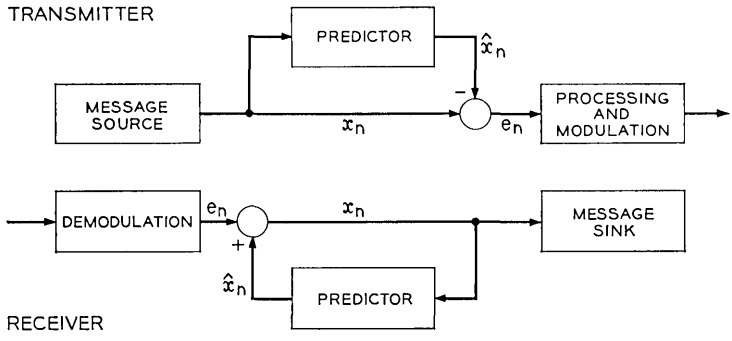


Fig. 1 — Predictive system.

transmission. Since the portion $\{\hat{x}_n\}$ "removed" from the input sequence is a deterministic function of the error sequence, no information has been lost and the original sequence can be reconstructed at the receiver by the feedback loop shown in the figure.

The philosophy of predictive systems has been widely studied for its application in bandwidth compression of telemetry data and of television; for example, see Kortman, Davisson, and O'Neal.[6-8] In these examples the error samples e_k are quantized and transmitted by pcm. Because of redundancy, that is, predictability, in the source data, fewer digits per sample (and consequently less bandwidth) are required for transmitting the error samples than for transmitting the original samples for a given fidelity of reconstruction.

One of the difficulties with these data compression systems is in determining the predictor filter. Although the theory of linear prediction for stationary time series is well known, the practical determination of the statistical properties of the input data and the realization of the corresponding optimum filter are nearly impossible. Generally, an approximate average statistical description is used for the input data and a considerably simplified version of the optimum filter is constructed. Most existing compression schemes appear to use only linear or zero-order extrapolation of the previous sample to form the prediction of the succeeding sample. More complicated and adaptive prediction techniques have been confined to computer-processed data.

In this paper we describe a simply-instrumented adaptive filter for use as a predictor. This filter uses a finite tapped delay line whose coefficients are continually adjusted to provide a least squares prediction of incoming data. The coefficient settings are based on the sta-

tistics of a finite section of the past data (the learning period). As the statistics of the data during this learning period change, the coefficients are changed to provide an updated version of the predictor filter.

Although the most obvious applications of this adaptive predictor would be in the transmission of television or some other very redundant analog signal, we choose here to explore its application in digital data transmission. In the past, little attention seems to have been focused on the use of prediction in digital transmission. Presumably this is because the most effective use of prediction would be in the compression of the analog wave from which the digits are taken.

However, there do exist situations in which the input digital signal is not under the control of the transmission systems designer. This occurs notably in the design of data communications equipment. Although it has been common practice to use redundancy in speech signals to ease transmission system requirements (the TASI system is a dramatic example), nothing similar has been attempted with digital data signals. There would seem to be no compelling reason why any redundancy in digital signals should not be taken advantage of, as long as the error statistics of the output data were not adversely affected by the procedure. After describing a digital redundancy removal and restoration system we shall discuss its possible benefits to the customer and to the transmission plant.

II. SYSTEM DESCRIPTION

Figure 2 shows a digital redundancy removal and restoration scheme. For simplicity we assume that the input digits a_n are binary, although the technique obviously extends to multilevel transmission. The input sequence is passed through a shift-register transversal filter whose tap gains c_k have been adjusted so that the filter output $\hat{a}_n$, where

$$\hat{a}_n = \sum_{k=1}^{N} c_k a_{n-k} , \tag{1}$$

is a linear least squares prediction of a_n. This prediction is subtracted from the actual sample a_n and only the difference e_n is passed to the modulator for transmission. Notice that, although a_n is a binary variable taking on the values ± 1, both $\hat{a}_n$ and e_n are analog. Unless the digits a_n are uncorrelated, the error samples e_n will have smaller variance than the unit variance of the input data. Consequently, a linear modulator

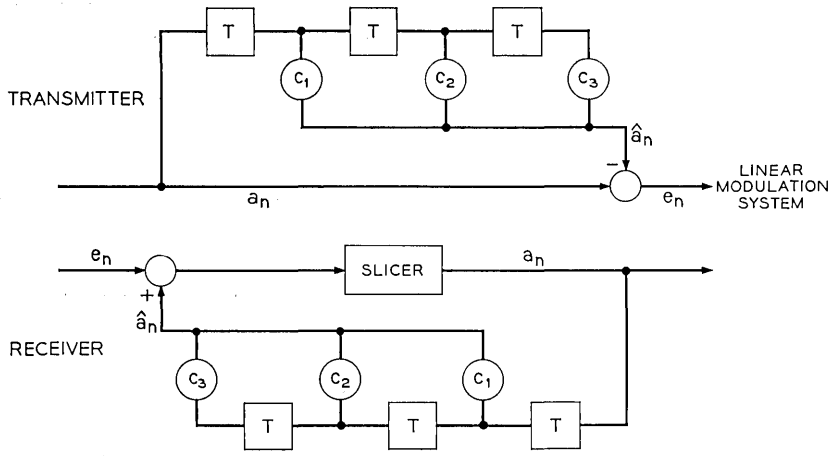


Fig. 2 — Digital redundancy removal and restoration.

will put out less line power in transmitting the error samples than in transmitting the original data.

After demodulation at the receiver, the missing, predictable, component $\hat{a}_n$ must be added to the error sample e_n before slicing, in order to recover a_n. This component is obtained by a bootstrap arrangement wherein the detected symbols are passed through a transversal filter identical to that at the transmitter in order to form the predictions $\hat{a}_n$. The receiver is similar in arrangement to the circuitry used in dc restoration.

There are two relatively simple ways in which this system could be used to improve transmission efficiency. As shown in Figure 2 the system lowers the average transmitted power without appreciably affecting the output data error rate. In this mode of operation any benefit from the data redundancy is used to lower the load requirements on the transmission plant. If many data sets were equipped with such circuitry, the average power handled by the plant would be lowered in a statistical fashion. Some sets, transmitting entirely random data, would require their normal power complement. Others, transmitting redundant data, would require considerably less. Notice that this is exactly the type of effect which now takes place for voice transmission.

As the input data becomes entirely redundant in the limit, the transmitted power goes to zero. In this case the input data consists of a periodic pattern. In spite of the zero-level line signal, the pat-

tern is reconstructed exactly at the receiver (in the absence of noise). Such an eventuality would alleviate the problems now encountered with the transmission of periodic data. These data patterns normally lead to tones, that is, line spectra, in the transmission channel which cause certain overloading and other system malfunctions.

Currently the problem is being treated in wideband transmission by the introduction of digital scramblers.[9] In practice the zero-level transmitted signal would not be a satisfactory solution to the tone problem since some signal strength would be required for synchronizing and timing maintenance. However, proper design of the system could ensure that some minimum signal strength was maintained under all circumstances. For example, a nonlinear element in each predictor could be used to keep the predictions smaller than unity. As long as the same nonlinearity were used in both transmitter and receiver, the data signal would be reconstructed perfectly at the receiver.

The other simple way to use redundancy removal to aid transmission would be to keep the level of transmitted power constant while lowering the probability of error. In this case, compensating gain controls would be placed at the transmitter output and at the receiver input. These controls would be adjusted to keep the transmitted power constant regardless of signal redundancy. During periods of redundancy most of the voltage presented to the slicer at the receiver would come via the feedback predictor and therefore would be noiseless (in the absence of errors). Since the small error signal transmitted would be greatly amplified to keep line power constant, the total noise presented to the slicer after complementary deamplification would be much smaller than in normal transmission. Consequently, the error rate would be diminished during periods of redundant data transmission.

Complementary amplification and deamplification surrounding channel noise introduction are automatically accomplished in transmission over compandored facilities. Normally for these channels we would expect that the error rate would be independent of transmitted power level. In the redundancy removal system, however, this mechanism is defeated by using the noiseless feedback in the detection process.

There are further uses of redundancy removal in data transmission, but they appear to involve more complicated system arrangements. For example, the bit rate and bandwidth of the data signal could be lowered for redundant data. This could be accomplished by slicing

the prediction $\hat{a}_n$ to obtain a closest *digital* prediction and then subtracting $\hat{a}_n$ from a_n in digital form. The resulting error digits could then be processed by run-length encoding to achieve message compression. Of course we would then need a buffer to ensure a constant channel bit rate. We will not discuss this type of system further here.

Thus far we have alluded to the possible benefits of redundancy removal in data transmission. There is also one major drawback—that of error propagation. Since the estimate $\hat{a}_n$ at the receiver depends on the correct reception of all previous data, the compensation at the receiver is perfect only in the absence of errors. When an error occurs, the probability of error in succeeding bits tends to be larger and an error propagating effect occurs. Notice that this effect does not depend on the particular circuit configuration for its existence, but is a philosophical necessity in any redundancy removal operation. We analyze the effect of error propagation in a simple example in Section V. Normally we would not expect the error propagation to increase the entire error rate by more than a small algebraic factor.

III. THE ADAPTIVE PREDICTION FILTER

In the theory of linear prediction developed by Wiener[5] and others it is assumed that the input samples a_n are taken from a stationary time series with known covariance function $R(n)$, where

$$E[a_m a_n] = R(m - n). \tag{2}$$

The power output, which is the mean square prediction error, is

$$P = E[e_n^2] = E\left\{\left(a_n - \sum_{k=1}^{N} c_k a_{n-k}\right)^2\right\}. \tag{3}$$

The coefficients c_k; $k = 1, \ldots N$, which minimize this prediction error, can be obtained by the solution of the N simultaneous equations

$$\sum_{k=1}^{N} c_k R(n - k) = R(n); \qquad n = 1, 2, \cdots, N. \tag{4}$$

In case of an infinite filter ($N = \infty$) the coefficients c_k and the prediction error are given by a method involving factoring of the spectral density $G(f)$ of the input process. Under proper conditions the prediction error P can be expressed in the form

$$P = \exp\left[\int_{-\frac{1}{2}}^{\frac{1}{2}} \log G(f)\, df\right] \tag{5}$$

(See Doob for the mathematical niceties of this result.[10]) Notice that if the input symbols are independent, $G(f) = 1$, $| f | \leq \frac{1}{2}$, and $P = 1$. Since the input power is also unity no gain is achieved by the prediction process. If, on the other hand, $G(f)$ is not flat the prediction error, P is less than unity and power is saved.

While the mathematics of linear prediction for stationary time series serve as a guide to actual system performance, it is clear that the assumptions are philosophically inadmissible. Furthermore, since the data source is outside the designer's control, it would be extremely unlikely that the covariance function would be known in advance. For these reasons, Balakrishnan[11] in 1961 developed a mathematical formulation for a learning or adaptive predictor wherein the form of the prediction operator was dependent solely on the past data and not on any assumptions of stationarity or of prior knowledge of data statistics.

In Balakrishnan's formulation that prediction operator is chosen as optimum at time t_n which works best when applied at times $t_{n-1}, \ldots, t_{n-L}$. Since all past information is available, we could "try out" all possible prediction operators on the previous data and select the operator for which

$$E_n = \sum_{j=1}^{L} [a_{n-j} - \hat{a}_{n-j}]^2 w_j \tag{6}$$

is minimum. The weights w_j could be used to assign a relative importance to each past trial of the predictor.

For our finite linear predictor we have

$$E_n = \sum_{j=1}^{L} \left[a_{n-j} - \sum_{k=1}^{N} c_k a_{n-j-k} \right]^2 w_j \, . \tag{7}$$

In order to develop a physical implementation for this adaptive filter we use a motivation based on a steepest descent approach. The derivatives of the error E_n with respect to the coefficients c_m are

$$\frac{\partial E_n}{\partial c_m} = - \sum_{j=1}^{L} 2w_j \left[a_{n-j} - \sum_{k=1}^{N} c_k a_{n-j-k} \right] a_{n-j-m} \tag{8}$$

$$\frac{\partial E_n}{\partial c_m} = - \sum_{j=1}^{L} 2w_j e_{n-j} a_{n-j-m} \, . \tag{9}$$

Notice that these derivatives can be obtained by passing the product of sample a_{n-m} and the error voltage e_n through a filter with impulse response $\{w_j\}$. Thus we are led to the adaptive filter configuration

shown in Figure 3. This configuration is entirely similar to that currently being used for equalization[12] and for echo suppression.[13, 14]

When the input samples a_n are digital, the circuitry of Figure 3 is quite simple. The delay line becomes a shift register and the multipliers become simple polarity switches. However, the circuit is not limited to digital applications, but could be used in such analog functions as telemetry or television compression systems.

In any event, the response of the system, involving accuracy and settling time as well as stability, is controlled by selection of the smoothing filters $W(\omega)$. Basically these filters must perform an averaging followed by an integration. If the data were stationary and the memory L sufficiently long, the result of averaging the product of the error and sample voltages for the m^{th} tap coefficient would give (see equation 8)

$$y_m(t) \cong E[a_{n-m}e_n] = R(m) - \sum_{k=1}^{N} c_k(t)R(m - k). \tag{10}$$

Then these voltages would be integrated for use as tap coefficients, so that the governing system equations would be

$$\dot{c}_m(t) = A\left[R(m) - \sum_{k=1}^{N} c_k(t)R(m - k)\right] \quad \text{for} \quad m = 1, \cdots, N. \tag{11}$$

This system would be stable for all A, since the covariance matrix, whose nm^{th} entry is $R(n\text{-}m)$, must be positive definite (see Davenport and Root[15]). All voltages $y_m(t)$ would be asymptotically reduced to

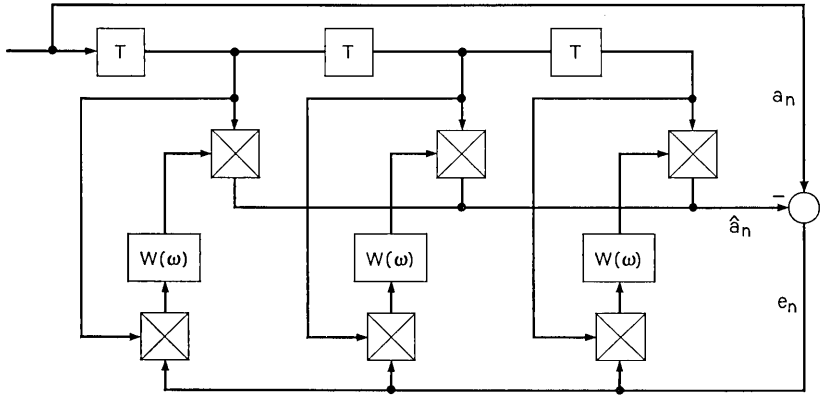


Fig. 3 — Adaptive prediction filter.

zero and the filter coefficients would asymptotically approach those of the optimum (least squares) linear predictor of equation (4).

For nonstationary data and realistic filters $W(\omega)$ the analysis of the nonlinear, multidimensional control system is extremely complicated. Let us study the dynamics of the one-dimensional system formed by using a one-tap predictor as a guide to the behavior of the system.

In order to put this analysis into proper perspective with regard to the system of Figure 2 we should observe that when the input data statistics change abruptly, both transmitter and receiver predictors undergo the same transients. If the predictors are identical, these transients cancel exactly at the receiver summer and no loss in noise margin is suffered. However, the statistics of the transmitted signal are affected by only the transmitter predictor. Therefore, the proper design of the adaptive predictor is crucial to obtaining desirable line power statistics, but not to the performance of the entire system.

IV. THE ONE-TAP TRANSMITTER FOR BINARY DATA

Figure 4 shows a one-tap transmitter with a binary input signal of the form

$$\begin{aligned} s(t) &= \sum_{n=0}^{\infty} a_n r(t - nT) \\ a_n &= \pm 1 \\ r(t) &= \begin{cases} 1 & 0 \leqq t < T \\ 0 & \text{elsewhere} \end{cases} . \end{aligned} \tag{12}$$

The transmitted voltage is given by

$$e(t) = s(t) - c(t)s(t - T) \tag{13}$$

where

$$c(t) = Aw(t)*[s(t - T)e(t)]. \tag{14}$$

Because of the binary nature of the input $s^2(t) = 1$ and thus

$$c(t) = Aw(t)*[s(t)s(t - T) - c(t)]. \tag{15}$$

Let $m(t) = s(t)s(t-T)$; then the Laplace transform solution for $C(s)$ is*

* Some liberty has been taken with the shift-register starting state.

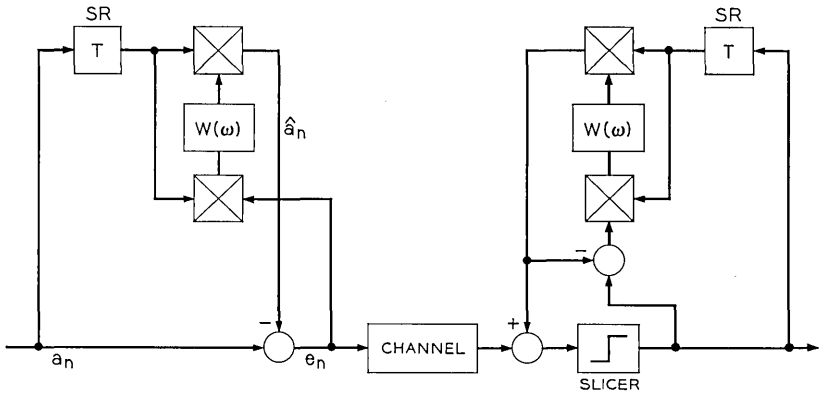


Fig. 4 — One-tap digital system.

$$C(s) = \frac{AW(s)M(s)}{1 + AW(s)}. \tag{16}$$

Now returning to equation (13) we multiply both sides by $s(t-T)$ to obtain

$$e(t)s(t - T) = m(t) - c(t). \tag{17}$$

Combining equations (16) and (17) gives

$$e(t)s(t - T) = m(t)*h(t) \tag{18}$$

where

$$H(s) = \frac{1}{1 + AW(s)}. \tag{19}$$

The output signal itself can be written by again multiplying equation (18) by $s(t-T)$

$$e(t) = s(t - T)[m(t)*h(t)]. \tag{20}$$

Notice that the special properties of binary sequences have been used in arriving at this solution, so that equation (20) does not hold for multilevel or analog input.

Figure 5(a) shows the mathematically equivalent transmitter given by equation (20) as well as its corresponding receiver. Since the second multiplier does not affect the transmitted power in any way, both transmitter and receiver can be simplified by its removal to result in the equivalent represented by Figure 5(b).* This final

* The systems differ in their noise performance, however.

equivalent system is amazingly simple and appears to bear little resemblance to the initial system of Figure 4. It is interesting to observe that, while the initial system was termed "adaptive," no one would seriously consider its equivalent in Figure 5(b) as being adaptive in any sense.

Figure 5(b) has an intriguing interpretation. The input data is first subjected to the nonlinear operation of delay and multiplication. The output of the multiplier is

$$m(t) = \sum_n a_n a_{n-1} r(t - nT). \tag{21}$$

This voltage has a mean value given by $R(1)$ in the stationary case. If the filter $W(\omega)$ has been designed as a low pass filter, then the filter $1/[1 + AW(\omega)]$ in the equivalent circuit is a high pass filter. Thus the dc component of $m(t)$ is removed before transmission and reinserted via a dc restorer at the receiver. In other words, a nonlinear operation on the input signal has converted the correlation into a spectral line which can then be removed by a time invariant linear filter. It would seem that some generalization of this concept should be possible, but as yet none has been found.

The equivalent circuit can be used for design purposes in selecting

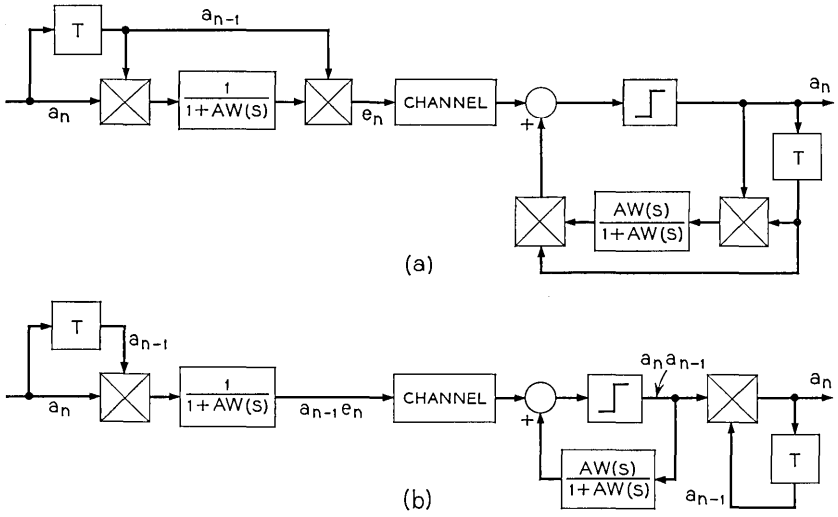


Fig. 5 — Equivalent binary one-tap systems. (a) Equivalent system. (b) Simplified equivalent.

$W(\omega)$, or for calculating line power or transient response. Here are the results of a few straightforward examples.

Example 1

Simple RC filter, dotting pattern input applied at time zero:

$$W(s) = \frac{\alpha}{s + \alpha}; \qquad \alpha = \frac{1}{RC}$$
$$a_n = \begin{cases} +1, & n \text{ even} \\ -1, & n \text{ odd} \end{cases}. \tag{22}$$

A deterministic sequence is to be transmitted. We find that the output of the equivalent circuit is

$$e(t)s(t - T) = -\left[\frac{1}{A + 1} u(t) + \frac{A}{A + 1} e^{-\alpha(A+1)t}\right]. \tag{23}$$

Thus the error voltage transmitted in the original circuit becomes

$$e(t) = \left[\sum_{n=0}^{\infty} (-1)^n r(t - nT)\right]\left[\frac{1}{A + 1} u(t) + \frac{A}{A + 1} e^{-\alpha(A+1)t}\right]. \tag{24}$$

The error voltage does not approach zero because of the lack of an integration in the smoothing filter.

Example 2

Simple RC filter, markov input:

If the input is a first order Markov process the one-tap predictor becomes the optimum linear predictor. (We study this case more thoroughly in the next section.) The covariance function of the input time series is taken to be

$$R(n) = R^{|n|}. \tag{25}$$

Since we now are dealing with a random input, our concern is with the transmitted power level rather than the exact waveform as in the previous example. The transmitted power is the same in Figures 4 and 5b, so we use the simpler structure of the latter diagram for analysis.

When the input Markov process is subjected to delay and multiplication, it can be shown that the resultant symbols $(a_n a_{n-1})$ have mean value R and are uncorrelated. The spectral density of the

multiplier output $m(t)$ is given by

$$S_M(\omega) = R^2\,\delta(\omega) + (1 - R^2)T\,\frac{\sin^2\frac{\omega T}{2}}{\left(\frac{\omega T}{2}\right)^2}. \tag{26}$$

This spectral density can be multiplied by $|H(\omega)|^2$ and integrated to give the transmitted power. The power becomes

$$P = \frac{R^2}{(1+A)^2} + (1 - R^2) \cdot\left\{\frac{1}{(1+A)^2} + \left[1 - \frac{1}{(1+A)^2}\right]\left[\frac{1 - e^{-\alpha(1+A)T}}{\alpha(1+A)T}\right]\right\}. \tag{27}$$

Ideally, of course, this power should be $(1-R^2)$, but the crude RC filter is unable to approximate this result unless the gain is high and the time constant $(1/\alpha)$ is large.

Better results in both examples could be achieved by an improved selection of the filter characteristic $W(\omega)$. We can see from the equivalent circuit that the best choice of $W(\omega)$ makes $1/[1 + AW(\omega)]$ an efficient high pass filter with a transmission zero at $\omega = 0$. Of course this must be compromised with any requirement on the filter response time.

In this section we stress the use of the equivalent circuit as a method of analysis rather than as an implementable system. Clearly, if one were to build a one-tap binary predictor, the circuit of Figure 5(b) would be preferred to that of the original system. However we believe that such a restricted system would not be of great practical interest.

While the implementation of the simple equivalent circuit cannot be extended to wider application, it is hoped that the easy analysis of the simple system conveys some insight into the performance of multiloop systems. This would be particularly true if there were small interaction between taps on the multiloop system. Such a situation would occur if the covariance $R(n)$ decreased rapidly with n.

V. ERROR PROPAGATION

When noise is added in the transmission channel there is some probability of the received digits being incorrectly detected by the slicer. Even though the transmitted power might have been substan-

tially reduced by the redundancy removal, the probability of an initial error is identical to that of a full power system. Once an error has been made, however, the probability of making subsequent errors is increased because of the incorrect symbol being used in redundancy restoration. Thus, errors tend to bunch together in the received data. Besides increasing the average probability of error this error propagation considerably complicates the problems of error control in the entire system.

Error propagation in dc restoration circuits has been examined by Zador, Aaron, and Simon.[16, 17] It appears to be a very complicated problem, in general, which is even more confused by the presence of the adaptive, pattern sensitive filters in the redundancy removal system we are considering here. Therefore, we shall attempt the analysis of only the simplest meaningful theoretical model. Both transmitter and receiver will have one-tap transversal filters as shown in Figure 4. The input data is taken to be a binary first order Markov process, with zero mean and covariance

$$R(n) = R^{|n|}.$$

The transition matrix for this process is:

a_n \ a_{n+1}	$+1$	-1
$+1$	$\frac{1+R}{2}$	$\frac{1-R}{2}$
-1	$\frac{1-R}{2}$	$\frac{1+R}{2}$

The ideal linear predictor for this time series is simply $\hat{a}_n = Ra_{n-1}$ and the average transmitted power using this predictor is $1 - R^2$. Since the ideal predictor uses only a single tap filter, the assumption of single tap filters in the actual system is not particularly restrictive. If additional taps were used, their gains would be small and their effect on error propagation would not be significant.

We will assume that noise samples ξ_k, uncorrelated Gaussian random variables with zero mean and variance σ^2, are added to the transmitted symbols in the channel. We further assume that sufficient smoothing is done at the transmitter so that the tap gain may

be fixed at its optimum value, R. Thus the transmitted samples are

$$e_k = a_k - Ra_{k-1} . \tag{28}$$

Now at the receiver we shall write the received symbols as $\beta_k a_k$. The parameter $\beta_k = \pm 1$ indicates the absence $(+1)$ or the presence (-1) of an error at time t_k. If the tap gain at the receiver is denoted by the parameter c, the detected symbols can be written

$$\beta_k a_k = \operatorname{sgn} [a_k - a_{k-1}(R - c\beta_{k-1}) + \xi_k]. \tag{29}$$

Thus the error parameter β_k is

$$\beta_k = \operatorname{sgn} [1 - a_k a_{k-1}(R - c\beta_{k-1}) + \eta_k] \tag{30}$$

where $\eta_k = \xi_k a_k$ has the same statistical properties as ξ_k. The probability of error at time t_k is the probability that $\beta_k = -1$, which is the probability that η_k is such that the term in brackets is negative.

Now we must turn our attention to the behavior of the receiver tap gain c. If no errors are made, then this gain is identical to the transmitter gain and as $k \to \infty$, $c \to R$. However, because of the presence of errors, the receiver tap gain tends to be different from the transmitter tap gain. At time t_k the output voltage of the multiplier at the receiver is

$$v_k = \beta_k a_k \beta_{k-1} a_{k-1} - c. \tag{31}$$

The random variables v_k are averaged to determine the movement of c. Notice that, since $| \beta_k a_k \beta_{k-1} a_{k-1} | = 1$, the magnitude of c cannot exceed unity except as a transient starting state. This eliminates any possibility of a runaway in c resulting from unusual error patterns.

We assume that the action of the loop at the receiver is to reduce to zero the expectation of the multiplier output voltage at time infinity. Thus

$$E[v_\infty] = 0 = \lim_{k \to \infty} E[\beta_k a_k \beta_{k-1} a_{k-1}] - c_\infty . \tag{32}$$

This type of final behavior would be exhibited by systems in which $W(\omega)$ consisted of a long term averaging followed by an integration. The expectation of the term in brackets in equation (32) depends on c_∞ itself, so in general we end with a fairly complicated equation requiring a trial and error solution for c_∞ . By taking the limit as $k \to \infty$ of the expectation we eliminate the dependence on time and on the initial probability distributions for the random variables involved.

Define a vector random variable $\bar{\alpha}_k = (a_k , \beta_k)$ taking on the four

possible states (+1, +1), (+1, −1), (−1, +1) and (−1, −1), denoted by states 1 through 4, respectively. Because a_k is Markov and since the expression for β_k in equation (30) involves only a_k, a_{k-1}, β_{k-1}, and η_k, we conclude that $\bar{\alpha}$ is also Markov. The four-by-four transition matrix π for $\bar{\alpha}$ has entries p_{ij} which may be calculated from the original transition matrix for the input symbols a_k and from equation (30) for the probabilities of error in various states. Table I lists these transition probabilities. If the 4-entry row vector $\bar{w}^{(k)}$ gives the probabilities of $\bar{\alpha}_k$ assuming each of the four possible states, then

$$\bar{w}^{(k)} = \bar{w}^{(k-1)}\pi. \tag{33}$$

In terms of the initial state distribution $\bar{w}^{(0)}$

$$\bar{w}^{(n)} = \bar{w}^{(0)}\pi^n. \tag{34}$$

For $|R| < 1$ it is clear from standard Markov chain theory (see, for example, Reference 18) that steady-state probabilities exist for

TABLE I—TRANSITION PROBABILITIES FOR $\bar{\alpha}_k = (a_k, \beta_k)$

$$Q(x) = \int_x^\infty \frac{e^{-y^2/2}}{\sqrt{2\pi}}\,dy$$

$$p_{11} = p_{33} = \left(\frac{1+R}{2}\right)\left[1 - Q\left(\frac{1-R+c}{\sigma}\right)\right]$$

$$p_{12} = p_{34} = \left(\frac{1+R}{2}\right)Q\left(\frac{1-R+c}{\sigma}\right)$$

$$p_{13} = p_{31} = \left(\frac{1-R}{2}\right)\left[1 - Q\left(\frac{1+R-c}{\sigma}\right)\right]$$

$$p_{14} = p_{32} = \left(\frac{1-R}{2}\right)Q\left(\frac{1+R-c}{\sigma}\right)$$

$$p_{21} = p_{43} = \left(\frac{1+R}{2}\right)\left[1 - Q\left(\frac{1-R-c}{\sigma}\right)\right]$$

$$p_{22} = p_{44} = \left(\frac{1+R}{2}\right)Q\left(\frac{1-R-c}{\sigma}\right)$$

$$p_{23} = p_{41} = \left(\frac{1-R}{2}\right)\left[1 - Q\left(\frac{1+R+c}{\sigma}\right)\right]$$

$$p_{24} = p_{42} = \left(\frac{1-R}{2}\right)Q\left(\frac{1+R+c}{\sigma}\right)$$

the transition matrix π, that is, $\bar{w}^{(n)}$ approaches a constant vector $\bar{w}$ as $n \to \infty$ independent of $\bar{w}^{(0)}$. The steady-state probabilities of the four possible states can be obtained by the solution of the equations given by

$$\bar{w}\pi = \bar{w}. \tag{35}$$

Some algebraic manipulation yields the probabilities

$$w_1 = P(a_\infty = +1, \beta_\infty = +1) = \frac{\frac{1}{2}(1 - p_{22} - p_{24})}{1 - p_{22} + p_{12} - p_{24} + p_{14}} \tag{36}$$

$$w_2 = P(a_\infty = +1, \beta_\infty = -1) = \tfrac{1}{2} - w_1 \tag{37}$$

$$w_3 = P(a_\infty = -1, \beta_\infty = +1) = w_1 \tag{38}$$

$$w_4 = P(a_\infty = -1, \beta_\infty = -1) = \tfrac{1}{2} - w_1 \tag{39}$$

where the transition probabilities p_{12}, p_{14}, p_{22}, and p_{24} are given in Table I as functions of c, R, and σ.

The expected value of the multiplier output at time infinity can now be written in terms of the steady-state probabilities w_i and the transition probabilities p_{ij}.

$$E[v_\infty] = w_1[p_{11} - p_{12} - p_{13} + p_{14}] + w_2[p_{22} + p_{23} - p_{21} - p_{24}]$$
$$+ w_3[p_{32} + p_{33} - p_{31} - p_{34}] + w_4[p_{41} + p_{44} - p_{42} - p_{43}] - c. \tag{40}$$

Again some algebraic manipulation yields the result

$$E[v_\infty] = \frac{R[1 - p_{14} - p_{24} - p_{22} - p_{12}] + 2[p_{14} - p_{12}] + 4[p_{22}p_{12} - p_{24}p_{14}]}{1 - p_{22} + p_{12} - p_{24} + p_{14}} - c. \tag{41}$$

The value of the tap gain at time infinity can be found by trial and error. A value of c is assumed, the transition probabilities are computed and $E[v_\infty]$ is found. The value of c for which $E[v_\infty] = 0$ is c_∞. Notice that under suitable assumptions $E[v_\infty]$ gives the rate of change of the coefficient c in the dynamic action of the system.

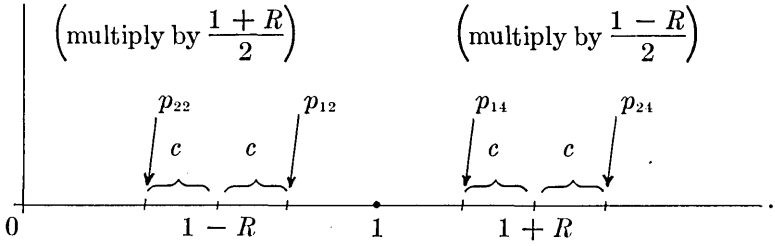
The probability of error after the system has settled is simply the probability that $\bar{\alpha}_\infty$ is in a state where $\beta_\infty = -1$, which is simply $(w_2 + w_4)$.

$$P_e = \frac{p_{12} + p_{14}}{1 - p_{22} + p_{12} - p_{24} + p_{14}}. \tag{42}$$

The transition probabilities here must be computed using c_∞.

Expressions (41) and (42) have been written in terms of only those transition probabilities which involve errors. Thus, as $\sigma \to 0$, each of the transition probabilities in (41) and (42) approaches zero,

$c_\infty \to R$, and $P_e \to 0$. Each of these probabilities can be visualized as the probability that the noise (zero mean, variance σ^2) is greater than the one of these four thresholds:



Thus p_{24} is the smallest transition probability, while p_{22} is the largest.

If the transition probabilities are small, it can be seen from equation (42) that P_e is principally determined by $(p_{12} + p_{14})$, which is minimized by $c = R$. Also we notice from equation (42) that the tap gain c approaches R very closely for small transition probabilities. In general, however, $c = R$ will not be the best setting to minimize the error probability in equation (42), nor is it the setting to which the loop settles. Unfortunately it appears that these are not compensating offsets. For example, in Figure 6 we have plotted P_e and $E[v_\infty]$ against c, for a case in which $R = 0.4$ and $\sigma = 0.4$. Although neither

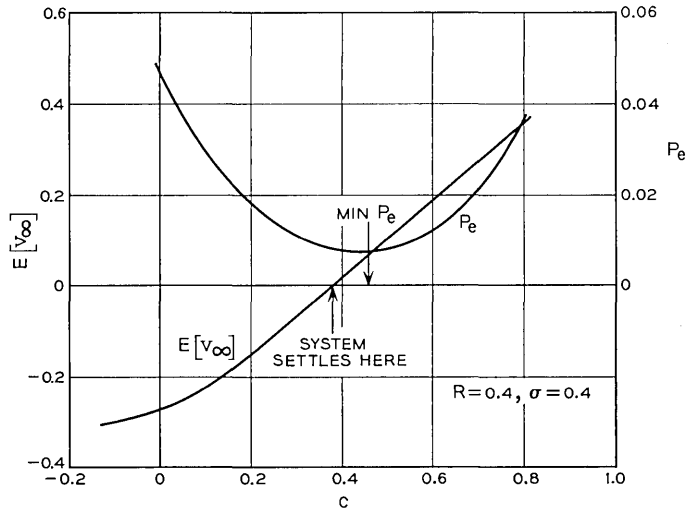


Fig. 6 — Probability of error and $E[v\infty]$ vs receiver tap gain c.

effect is very significant, it can be seen that the system settles ($E[v_\infty] = 0$) for a value of c somewhat smaller than R, while the minimum error probability is obtained at a value of c somewhat larger than R.

In all but the most severe noise conditions the approximation of $c_\infty = R$ would be satisfactory and we would have

$$P_e|_{c=R} = \frac{Q\left(\frac{1}{\sigma}\right)}{1 - \left(\frac{1+R}{2}\right)Q\left(\frac{1-2R}{\sigma}\right) - \left(\frac{1-R}{2}\right)Q\left(\frac{1+2R}{\sigma}\right) + Q\left(\frac{1}{\sigma}\right)}. \tag{43}$$

But $Q(1/\sigma)$ is the probability of error in the original system (no redundancy removal). If this probability, called P_{e0}, is small, then $Q(1 + 2R/\sigma)$ is much smaller and we have the very good approximation

$$P_e|_{P_{e0}\text{ small}} \cong \frac{P_{e0}}{\left[1 - \left(\frac{1+R}{2}\right)Q\left(\frac{1-2R}{\sigma}\right)\right]}. \tag{44}$$

The factor in the denominator gives the amplification of the original error rate due to error propagation. Finally if $R > 1/2$, then $Q(1 - 2R/\sigma)$ approaches unity and we get the severe dependence upon R

$$P_e|_{\substack{P_{e0}\text{ small} \\ R>\frac{1}{2}}} \cong \frac{2P_{e0}}{1-R}. \tag{45}$$

The most significant aspect of the error propagation behavior of the circuit is that the redundancy removal and restoration system has impressed the statistics of the input data (Markov here) upon the error statistics of the output. It is clear that this philosophy would hold in general. In the case of highly correlated input we would end with highly correlated errors. The problems of error control could be made quite severe in this manner.

VI. EXPERIMENTAL RESULTS

A three-tap, adaptive transmitter and a similar receiver were designed and constructed by V. G. Koll. The system was designed for binary data transmission so that the multipliers in Figure 3 became polarity switches, while the delay line took the form of a shift register. The filters $W(s)$ consisted of simple RC low pass sections followed

by integrators, that is,

$$W(s) = \frac{\alpha}{s(s + \alpha)}. \tag{46}$$

With this choice of smoothing, the steady-state error for a periodic input (period 3 or less here) was zero. It was in fact observed that during the transmission of periodic data the transmitter could be disconnected with no effect on the received data pattern.

The input data for the system was obtained by passing white Gaussian noise through a variable cutoff, low pass filter. If we assume an ideal low pass filter, with cutoff frequency W Hz, then the autocorrelation function of the filter output is

$$R_1(\tau) = 2N_0W\left[\frac{\sin 2\pi W\tau}{2\pi W\tau}\right]. \tag{47}$$

This voltage is then sampled at rate $(1/T)$ and subjected to infinite clipping so as to produce the correlated input bits. Van Vleck and Middleton[19] show that the resulting autocorrelation is

$$R(n) = \frac{2}{\pi}\sin^{-1}\left[\frac{\sin 2\pi nWT}{2\pi nWT}\right]. \tag{48}$$

For a filter cutoff of $1/2T$ Hz the data is uncorrelated. By decreasing the filter cutoff frequency the redundancy in the data can be increased.

The action of the adaptive redundancy remover is shown in Figure 7 for two different values of filter cutoff. Notice that as the redundancy is increased the transmitted waveform has longer periods of near zero voltage where predictability is good and occasional peaks where the predictor is "surprised." Except for a few minor discontinuities the reconstructed signal before slicing at the receiver is the same as the original input waveform at the transmitter. The relative power saving as a function of filter cutoff is shown in Figure 8.

In order to predict system performance in Gaussian noise we make the crude approximation that the input process is Markov with $R(1)$ as given in equation (48). According to this approximation the transmitted power should be $1 - R(1)^2$. This value is also shown in Figure 8 in comparison with the actual measured power output. Since the exact correlation function is known, the theoretical signal power output could be computed precisely through equation (4). However, we have no corresponding means of computing the degree of error propagation

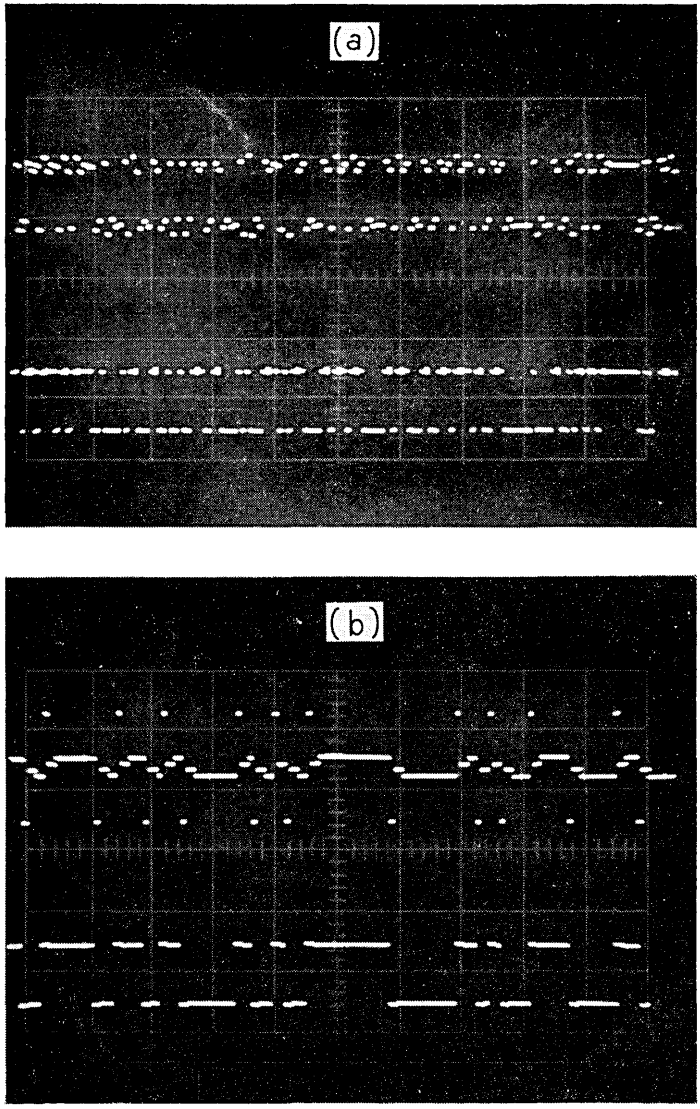


Fig. 7 — Transmitted and reconstructed signals. (a) Filter cutoff $\omega T = 0.4$ [little redundancy, $R(1) = 0.15$]. (b) Filter cutoff $\omega T = 0.1$ [moderate redundancy, $R(1) = 0.77$].

for the non-Markov source. The approximate curve of signal power in Figure 8 is shown only as a way of evaluating the Markov approximation for later use in predicting error propagation values.

Bandlimited white Gaussian noise was added to the transmitted signal, and error rates were experimentally determined by V. G. Koll at a number of filter cutoff (redundancy) positions. The results of these tests are shown in Figure 9 in curves of probability of error versus signal-to-noise ratio. Beside these measured curves have been plotted theoretically computed curves which are based on the Markov approximation and on the use of equation (43) for P_e.

Although all necessary information for performance determination is contained in Figure 9, it is instructive to plot two additional curves of probability of error versus filter cutoff. These curves are shown in Figure 10. In one curve the transmitter and receiver gains are held constant so that the line power decreases according to the curve of Figure 8 while the probability of error increases with increasing redundancy because of the effects of error propagation. In the other curve of Figure 10 the transmitter and receiver gains have been adjusted with increasing redundancy so as to hold line power constant. In this case the probability of error decreases with increasing redundancy.

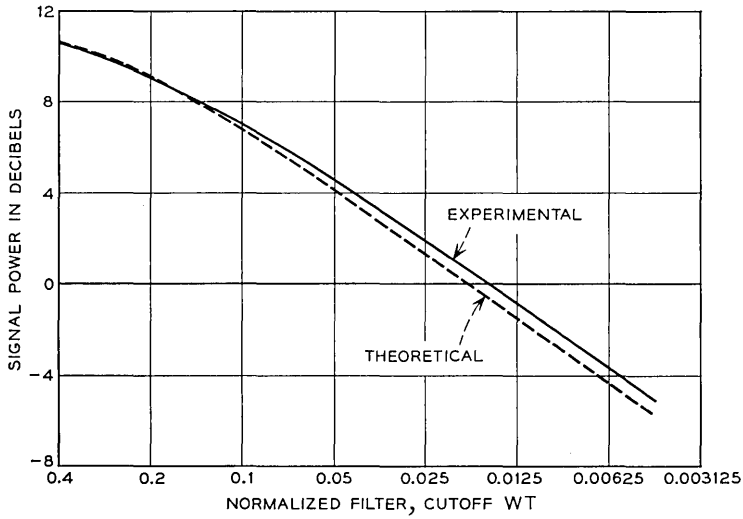


Fig. 8 — Signal power saving by redundancy removal.

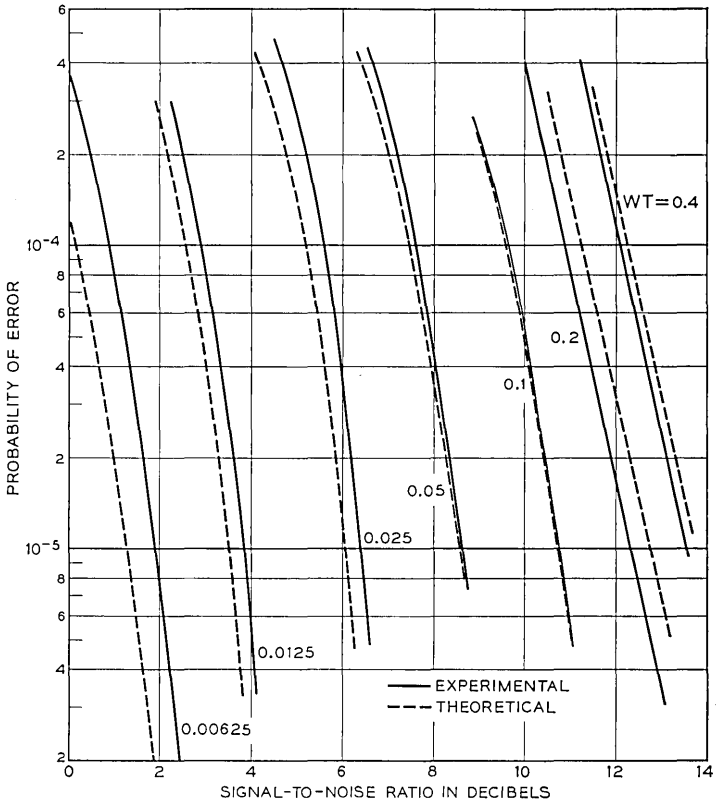


Fig. 9 — Performance of redundancy removal system at various values of normalized filter cutoff ωT.

VII. CONCLUSION

We have advanced two main points. First we suggest the possibility of using an easily-implemented adaptive predictor for data compression systems. Second, we investigated the use of this adaptive predictor in digital transmission.

We have seen that the predictor can be used to increase transmission efficiency for redundant data either by decreasing signal power for a given error rate or by decreasing probability of error for a given signal power. Although the required circuitry for the digital application is quite simple, it is nearly impossible to make an economic evaluation of the system because of the complete lack of knowledge of the prevalence and degree of redundancy in customer input data.

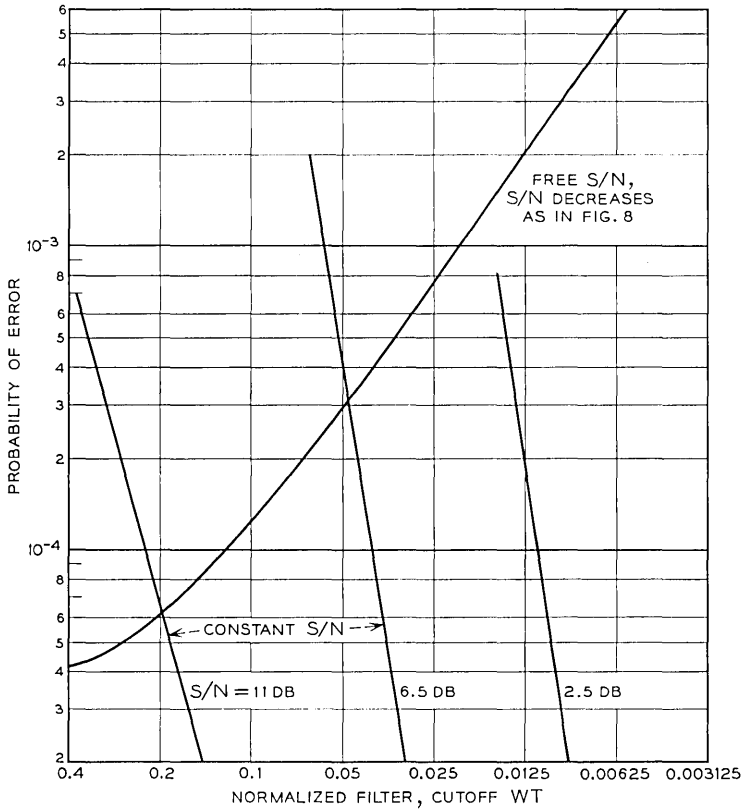


Fig. 10 — Probability of error vs filter cutoff for constant and for free S/N.

VIII. ACKNOWLEDGEMENT

The author is indebted to V. G. Koll who designed, constructed, and tested the three-tap experimental ssytem. Mr. Koll also made the photos and the experimental performance curves in Figures 7 through 10.

REFERENCES

1. Oliver, B. N., "Efficient Coding," B.S.T.J., *31*, No. 4 (July 1952), pp. 724–750.
2. Kretzmer, E. R., "Statistics of Television Signals," B.S.T.J., *31*, No. 4 (July 1952), pp. 751–763.
3. Harrison, C. W., "Experiments with Linear Prediction in Television," B.S.T.J., *31*, No. 4 (July 1952), pp. 764–783.
4. Elias, P., "Predictive Coding," IRE Trans. Inform. Theory, *IT*-1, No. 1 (March 1955), pp. 16–33.

5. Wiener, N., *Extrapolation, Interpolation and Smoothing of Stationary Time Series,* Cambridge, Mass.: MIT Press, 1949.
6. Kortman, C. M., "Redundancy Reduction—A Practical Method of Data Compression," Proc. IEEE, *55,* No. 3 (March 1967), pp. 253–263.
7. Davison, L. D., "Theory of Adaptive Data Compression," in *Recent Advances in Communication Systems,* vol. 2, ed. A. V. Balakrishnan, New York: Academic Press, 1966.
8. O'Neal, J. B., "Predictive Quantizing Systems," B.S.T.J., *45,* No. 5 (May-June 1966), pp. 689–721.
9. Savage, J. E., "Some Simple Self-Synchronizing Digital Data Scramblers," B.S.T.J., *46,* No. 2 (February 1967), pp. 449–487.
10. Doob, J. L., *Stochastic Processes,* Chapter 12, New York: John Wiley and Sons, Inc., 1953.
11. Balakrishnan, A. V., "An Adaptive Non-Linear Data Predictor," 1962 Nat. Telemetering Conf. Proc.
12. Lucky, R. W. and Rudin, H. R., "An Automatic Equalizer for General-Purpose Communication Channels," B.S.T.J., *46,* No. 9 (November 1967), pp. 2179–2208.
13. Becker, F. K. and Rudin, H. R., "Application of Automatic Transversal Filters to the Problem of Echo Suppression," B.S.T.J., *45,* No. 10 (December 1966), pp. 1847–1850.
14. Sondhi, M. M. and Presti, A. J., "A Self-Adaptive Echo Canceller," B.S.T.J., *45,* No. 10 (December 1966), pp. 1851–1853.
15. Davenport, W. B. and Root, W. L., *An Introduction to the Theory of Random Signals and Noise,* New York: McGraw-Hill Book Co., Inc., 1958, p. 105.
16. Zador, P. L., "Error Probabilities in Data System Pulse Regenerator with DC Restoration," B.S.T.J., *45,* No. 6 (July-August 1966), pp. 979–984.
17. Aaron, M. R. and Simon, M. K., "Approximation of the Error Probability in a Regenerative Repeater with Quantized Feedback," B.S.T.J., *45,* No. 10 (December 1966), pp. 1845–1847.
18. Kemeny, J. G. and Snell, J. L., *Finite Markov Chains,* New York: D. Van Nostrand Co., Inc., 1959.
19. Van Vleck, J. H. and Middleton, D., "The Spectrum of Clipped Noise," Proc. IEEE, *54,* No. 1 (January 1966), pp. 2–19 (reprint).

Group Codes for the Gaussian Channel

By DAVID SLEPIAN

(Manuscript received April 27, 1967)

A class of codes for use on the Gaussian channel, called group codes, is defined and investigated. Roughly speaking, all words in a group code are on an equal footing: each has the same error probability and the same disposition of neighbors. A decomposition theorem shows every group code to be equivalent to a direct sum of certain basic group codes generated by real-irreducible representations of a finite group associated with the code. Some theorems on distances between words in group codes are demonstrated. The difficult problem of finding group codes with large nearest neighbor distance is discussed in detail.

I. INTRODUCTION

In a communication model first introduced by Kotel'nikov[1] in 1947, and independently by Shannon[2] in 1948, and since studied by many authors,[3–22] messages for transmission are represented by vectors in a Euclidean space, $\mathcal{S}_n$, of n dimensions called signal space. In this model, known as the Gaussian channel, when $\mathbf{X}$ is transmitted, the received signal is represented by a vector $\mathbf{Z} = \mathbf{X} + \mathbf{Y}$ which consists of the sum of the sent vector and a noise vector $\mathbf{Y}$ whose components are independent Gaussian variates with mean zero and variance σ^2. Some physical circumstances that lead to this model, as well as further details, can be found in Refs. 3, 10, and 13.

An equal-energy block code of size M for use on this Gaussian channel is a collection of M distinct vectors $\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_M$ in signal space all of the same length. We shall always suppose $M \geqq n$ and that the vectors span $\mathcal{S}_n$. The length of the vectors serves to define an important parameter S called the average power of the code through the equation

$$nS = |\mathbf{X}_i|^2. \tag{1}$$

The vectors of the code are called code words or code points. Their termini lie on the sphere of radius $\sqrt{nS}$ centered at the origin of $\mathcal{S}_n$.

Associated with each code point $\mathbf{X}_i$ of an equal-energy block code

is a region $\mathcal{R}_i$ of signal space called a maximum likelihood region and defined by

$$\mathcal{R}_i = \left\{ \mathbf{X} \;\middle|\; | \mathbf{X} - \mathbf{X}_i | \leqq | \mathbf{X} - \mathbf{X}_j |, j \neq i \right\}. \tag{2}$$

That is, $\mathcal{R}_i$ is the set of all points in $\mathcal{S}_n$ at least as close to $\mathbf{X}_i$ as to any other code word. These regions are convex flat-sided cones with apex at the origin. The interiors of $\mathcal{R}_i$ and $\mathcal{R}_j$ are disjoint for $i \neq j$: the union of the $\mathcal{R}_i$ is all of $\mathcal{S}_n$.

The capabilities of equal energy block codes for communicating over the Gaussian channel are well known. If the words of a code are presented equally likely and independently for transmission over the channel, the communication rate is

$$R = \frac{\alpha}{n} \log M \tag{3}$$

natural units per second where α (measured in numbers per second) is the rate at which vector components are transmitted. The receiver which minimizes the average error probability[5,13] operates by asserting that code word $\mathbf{X}_i$ was transmitted when the received vector $\mathbf{Z}$ lies in $\mathcal{R}_i$, $i = 1, 2, \cdots, M$. (The received vector lies in the boundary of some $\mathcal{R}_i$ with probability 0.) When $\mathbf{X}_i$ was transmitted the error probability of this best receiver is

$$P_{ei} = \frac{1}{(2\pi\sigma^2)^{n/2}} \int_{\mathcal{R}_i'} \cdots \int \exp\left(-\frac{1}{2\sigma^2} |\mathbf{Y} - \mathbf{X}_i|^2 \right) dy_1 \cdots dy_n \tag{4}$$

where $\mathcal{R}_i'$ is the complement of $\mathcal{R}_i$. The average error probability is

$$P_e = \frac{1}{M} \sum_{i=1}^{M} P_{ei}. \tag{5}$$

Upper and lower bounds are known[4,6,7,11,17] for $P_{e\,\min}(M, n, S)$, the smallest attainable value of P_e for an equal-energy block code with the indicated parameters. In the limit as $n \to \infty$, these bounds lead to the famous capacity formula $C = \alpha/2 \log (1 + S/\sigma^2)$ whose interpretation we suppose known. For fixed finite values of M and n, however, little is known in the general case about codes for which P_e attains its minimal value (optimal codes). The cases $M = n + 1, n + 2, \cdots, 2n$ have been studied in some detail.[8,9,14] For $n = 2$, Weber[14] showed that the regular M-gon is globally optimal: for $M = n + 1$, $n = 2, 3, \cdots$, it has been shown[20] that the regular simplex is optimal. No other optimal codes with $n > 3$ are known.

Recently Wyner[12] has investigated the capabilities of equal-energy block codes when a suboptimal receiver, known as a bounded distance decoder, is used. Here the regions $\mathcal{R}_i$ of the maximum likelihood receiver are replaced by spheres of radius $d/2$ centered on the termini of the code vectors $\mathbf{X}_i$, where d is the minimum distance between any two words of the code. If the received vector is not in one of these spheres, a decoding error is assumed. Wyner established upper and lower bounds on the smallest error probability attainable with an equal-energy block code using bounded distance decoding. In the limit as $n \to \infty$ he obtained coding theorems and a capacity analogous to the usual ones. For finite M and n, the error probability using bounded distance decoding is a monotone decreasing function of the minimum distance d between code words of an equal-energy block code. In the general case little is known about equal-energy block codes with largest nearest neighbor distance.

For equal energy block codes of M vectors spanning $\mathcal{S}_n$ two optimization problems thus present themselves: to find a code for which P_e, as given by (4) and (5), is a minimum; and to find a code with largest nearest neighbor distance between its code words. We have made little progress in solving these problems.

In this paper we investigate instead a class of equal-energy block codes called group codes. It is conjectured that this class includes solutions to the problems just mentioned for many values of M and n. Quite apart from these questions of optimality, however, group codes possess an important symmetry property that makes their study of interest in its own right. Roughly speaking, all code words in a group code are on an equal footing. This notion is made precise in the next section.

Most codes that have been investigated for the Gaussian channel are group codes: it is likely that any code used in practice will be of this type. Group codes for the Gaussian channel are a natural extension of the group codes introduced for the binary channel in Ref. 21, and these latter codes are obtained as a special case of the codes described here.

In what follows, we define equivalence for group codes, then investigate the possible classes of group codes. Here the theory of group representations plays a key role.[25] The appendix gives a summary of results needed from this field. The problem of constructing group codes is considered and an optimization problem of some difficulty is encountered. A number of interesting properties of group codes are disclosed.

Many of the results reported here are contained in the author's Bell Telephone Laboratories report of May 7, 1951, a document that received a limited circulation outside the Laboratories. A number of these results were recently rediscovered independently by J. G. Dunn and appear in his thesis[22] along with extensions in directions different from those reported here. The discovery of an easy decoding algorithm for certain group codes[15] has led to a revival of the author's interest in this subject, and so the present paper, while in part very old, is a report on research now in progress. It examines the general structure of group codes. In a later paper we hope to give a detailed treatment of some group codes associated with the symmetric group.

II. GROUP CODES

In studying the geometric properties of equal-energy block codes, it is convenient to deal only with code vectors of unit length. That is, we set S in equation 1 equal to $1/n$, and deal with normalized codes. To compute error probabilities associated with the use of the code, one must scale up the vectors by a factor $\sqrt{nS}$.

Let $\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_M$ be the (unit) vectors of an equal-energy block code. It is clear from the definition of the regions $\mathcal{R}_i$ and from (4) and (5) that P_e is invariant under a rotation of the code as a whole. That is, if O is an arbitrary $n \times n$ orthogonal matrix and

$$\mathbf{X}_i' = O\mathbf{X}_i, \qquad i = 1, 2, \cdots, M, \tag{6}$$

the error probability P_e' for the code $\mathbf{X}_1', \cdots, \mathbf{X}_M'$ is the same as that for the code $\mathbf{X}_1, \mathbf{X}_2, \cdots \mathbf{X}_M$. The set of interword distances for the two codes is the same, and in particular both codes have the same minimum nearest neighbor distance d. Two codes whose vectors (with possible renumbering) can be related as in equation (6) are called *equivalent.* Equivalent codes have the same communication capabilities.

We now examine in what sense the words of an equal-energy block code in S_n might be "alike". Given the M unit vectors $\mathbf{X}_i$ that define the code, the real orthogonal n by n matrix O is said to leave the code invariant if the $\mathbf{Y}_i$ are a permutation of the $\mathbf{X}_i$ where $\mathbf{Y}_i = O\mathbf{X}_i$, $i = 1, 2, \cdots, M$. The collection $\theta = \{O_1, O_2, \cdots, O_g\}$ of all real orthogonal n by n matrices that leave the code invariant clearly forms a finite* group under ordinary matrix multiplication. Now transformation by

* By hypothesis, the $\mathbf{X}_i$ span S_n. An $n \times n$ orthogonal matrix is completely determined by its effect on a set of n vectors that span its carrier space. Since the words of the code are permuted along themselves by each element of θ, $g \leqq M!$.

an orthogonal matrix preserves distances between points, so that a possible definition of "alikeness" for the points of the code is to require that in the group θ there be elements O_1 , O_2 , $\cdots$, O_M that transform any particular word, say $\mathbf{X}_i$, into each of the M vectors of the code. A collection of M unit vectors spanning S_n that satisfies this condition will be called a *group code* and denoted by the symbol $\{M, n\}$. In a group code, if O_i sends $\mathbf{X}_1$ into $\mathbf{X}_i$ and O_j sends $\mathbf{X}_1$ into $\mathbf{X}_j$, then $O_jO_i^{-1}$ sends $\mathbf{X}_i$ into $\mathbf{X}_j$. We have then

Proposition 1: For a group code, the set of distances from $\mathbf{X}_i$ *to all other points of the code is the same as the set of distances from* $\mathbf{X}_j$ *to all other points of the code,* $i, j = 1, 2, \cdots, M$.

Each point has the same number of nearest neighbors, the same number of next nearest neighbors, and so on.

The maximum likelihood regions $\mathfrak{R}_i$ for a code are defined by equation (2) in terms of distances from code points. Since orthogonal matrices leave distances invariant, it follows that for a group code a matrix $O \in \theta$ that sends $\mathbf{X}_i$ into $\mathbf{X}_j$ also sends $\mathfrak{R}_i$ into $\mathfrak{R}_j$. From this fact and the form of (4) we have

Proposition 2: For a group code $\{M, n\}$ *the maximum likelihood regions* $\mathfrak{R}_1$, $\mathfrak{R}_2$, $\cdots$, $\mathfrak{R}_M$ *are all congruent and all words have the same error probability, that is,* $P_{e1} = P_{e2} = \cdots = P_{eM} = P_e$.

III. GENERATION AND CLASSIFICATION OF GROUP CODES

To each matrix O of the group θ of orthogonal matrices that leaves a group code $\{M, n\}$ invariant, there corresponds a permutation on M letters, namely the permutation effected by O on the M vectors of the code. That these permutations form a transitive permutation group follows from the definition of a group code. No two different elements of θ can effect the same permutation of the words of $\{M, n\}$ since the effect of an $n \times n$ matrix on a set of vectors spanning S_n completely determines the matrix. We have then

Proposition 3: The group θ *of all orthogonal* $n \times n$ *matrices leaving a group code* $\{M, n\}$ *invariant forms a faithful representation of (is simply isomorphic to) a transitive permutation group on* M *letters.*

Group codes $\{M, n\}$ do not exist for every M and n. For example, it is not hard to prove that it is impossible to arrange 5 points on the sphere in 3 dimensions to form a group code. Necessary and sufficient conditions on M and n for the existence of an $\{M, n\}$ are not known.

Group codes do exist in great abundance, however, and we shall give examples later. Indeed, from any set of $n \times n$ orthogonal matrices $O_1, O_2, \cdots, O_M$ that form a finite group $\mathcal{G}$ under matrix multiplication we can form a group code by choosing a unit n-vector $\mathbf{X}$ and forming the set of vectors

$$\mathbf{X}_i = O_i\mathbf{X}, \qquad i = 1, 2, \cdots, M. \tag{7}$$

Elements of $\mathcal{G}$ leave this configuration of vectors invariant by the group property. Since $\mathcal{G}$ must contain the $n \times n$ unit matrix, $\mathbf{X}$ is among the collection of vectors and it is sent into each of the other vectors. A group code therefore results. This code may not have M distinct vectors, however, and it may not span $\mathcal{S}_n$. The code depends on the initial vector $\mathbf{X}$.

If the code has fewer than M vectors, then for some $i \neq j$, $\mathbf{X}_i = \mathbf{X}_j$ or $O_i\mathbf{X} = O_j\mathbf{X}$, or $O_j^{-1}O_i\mathbf{X} = O_k\mathbf{X} = \mathbf{X}$ for some $O_k \in \mathcal{G}$. That is, $\mathbf{X}$ must be an eigenvector with eigenvalue unity for at least one $O \in \mathcal{G}$ different from the unit matrix. The set of all such $O \in \mathcal{G}$ forms the subgroup $\mathcal{H}$ of order h of $\mathcal{G}$ that sends $\mathbf{X}$ into itself. It is easy to show that by (7) $\mathcal{G}$ generates $\nu = M/h$ distinct vectors. Since the matrices of $\mathcal{G}$ have only a finite number of eigenvectors, however, it is always possible to choose an $\mathbf{X}$ so that the M vectors (7) are distinct.

It may not be possible, however, to choose $\mathbf{X}$ so that the vectors span $\mathcal{S}_n$. To discuss this matter further we must recall the notion of real-reducibility. A finite group of (real) orthogonal matrices $\mathcal{G} = O_1, O_2, \cdots, O_M$ is said to be real-reducible if there exists an $n \times n$ real orthogonal matrix O such that for $i = 1, 2, \cdots, M$

$$OO_iO^{-1} = \left(\begin{array}{c|c} A_i & D \\ \hline C & B_i \end{array}\right) \tag{8}$$

where A_i is an l by l matrix, B_i is an $n - l$ by $n - l$ matrix, $0 < l < n$ and C and D are matrices all of whose elements are zero. It is assumed that l does not depend on i. A group of real orthogonal matrices that is not real-reducible is said to be real-irreducible. In words, a real-reducible collection of matrices can be simultaneously transformed to block diagonal form by a real orthogonal matrix: a real-irreducible collection cannot be so reduced.* The reduced matrix in block form in equation (8) is said to be the direct sum of the two square matrices A_i and B_i.

* In the theory of group representations (see the appendix) reducibility is usually defined over the field of complex numbers. The definition is as above with O replaced by a *unitary* matrix. We shall speak simply of "reducibility" in this case as opposed to "real-reducibility".

It is easy to show that if the matrices O_i of equation (7) are real-irreducible, then the code they generate spans S_n for all choices of $\mathbf{X}$: if they are real-reducible, for some choices of $\mathbf{X}$ the code will not span S_n.

These comments lead to

Proposition 4: Every real-irreducible group $\mathcal{G} = O_1, O_2, \cdots, O_M$ *of real orthogonal* $n \times n$ *matrices serves by means of equation* (7) *to generate a group code* $\{M', n\}$ *for each unit vector* $\mathbf{X}$ *in* S_n. *Here* $M' \leqq M$. *If* $M' < M$, *it is a divisor of* M.

Propositions 3 and 4, together with the theory of group representations† suggest a means of classifying and generating all group codes. From Proposition 3 we can associate with a given group code $\{M, n\}$ a unique abstract group and a faithful representation θ of this group by orthogonal matrices. The code can be thought of as generated from one of its vectors, $\mathbf{X}$, say, by the operation of the matrices of this representation in the manner of equation (7). Now the representation θ will in general be real-reducible. There will exist then a real orthogonal matrix O that will exhibit θ in block form (8) as the direct sum of a number of real-irreducible representations. Denote this new reduced representation by θ'. It is easily seen that the matrices of θ' operating on the vector $\mathbf{Y} = O\mathbf{X}$ generate a group code $\mathbf{Y}_1, \mathbf{Y}_2, \cdots, \mathbf{Y}_M$ equivalent to the originally given $\{M, n\}$. We can further regard $\mathbf{Y}$ as the sum of its projections $\mathbf{Y}^1, \mathbf{Y}^2, \cdots$ on the various invariant subspaces of θ' indicated by its block structure.

By the procedure just outlined, for each equivalence class of group codes we arrive at a particular set of real-irreducible representations, say $\theta_1, \theta_2, \cdots, \theta_j$ of an abstract group, each with a corresponding associated vector $\mathbf{Y}^1, \mathbf{Y}^2, \cdots, \mathbf{Y}^j$. We regard $\mathbf{Y}^i$ as lying in the carrier space of θ_i, so that if θ_i is of dimension l_i, then $\mathbf{Y}^i$ is a vector of l_i components, $i = 1, 2, \cdots, j$. Let the length of $\mathbf{Y}^i$ be λ_i. We have $\sum \lambda_i^2 = 1$. The θ_i are determined by $\{M, n\}$ only up to equivalence in the sense of representation theory, owing to the possibility of reduction of θ by different matrices O. The vectors $\mathbf{Y}^i$ inherit some additional freedom owing to the M possible choices of $\mathbf{X}$ in the preceding paragraph.

We can think of the $\{M, n\}$ as decomposed by the above process into an equivalent direct sum of j group codes, the ith code being generated by the matrices of θ_i operating on the initial unit vector $\mathbf{Z}^i = \mathbf{Y}^i/\lambda_i$, $i = 1, 2, \cdots, j$. The constituent group codes are weighted by the numbers $\lambda_1, \lambda_2, \cdots, \lambda_j$ in forming the direct sum code $\{M, n\}$. Notice

† Knowledge of the material in the appendix is necessary for understanding much of the remainder of this paper.

that some of the constituent codes may have fewer than M distinct words.

Conversely, within equivalence we can construct any group code as the weighted direct sum of codes generated by real-irreducible groups of matrices after the manner of Proposition 4. In synthesizing codes in this manner, we may, of course, arrive at equivalent codes by several different constructions. The group $\mathcal{G}$ of Proposition 4 may be only a subgroup of the group of all orthogonal transformations that leave the code generated by $\mathcal{G}$ invariant. Different initial vectors operated on by the same group of matrices may give rise to equivalent codes.

Every group possesses the trivial real-irreducible one-dimensional *identity representation* in which each group element is represented by the one-dimensional unit matrix. The inclusion of this identity representation in the constituent codes making up a direct sum code represents a waste of one dimension, since the code is then equivalent to one in which each code vector has the same first component. This first component then carries no information. By omitting the first component of each vector (and rescaling the length of the resultant vectors), a new code of dimension $n - 1$ is obtained with error probability no greater than the original $\{M, n\}$. In general in what follows we will not be concerned with codes that contain this identity representation.

We turn our attention now to the basic problem of constructing good group codes as the weighted sum of properly chosen group codes generated by real-irreducible groups of orthogonal matrices.

IV. THE INITIAL VECTOR PROBLEM AND THE FUNDAMENTAL REGION

As in Proposition 4, let a code be constructed from a given group $\mathcal{G} = O_1, O_2, \cdots, O_M$ of orthogonal $n \times n$ matrices by means of equation (7). We think of these matrices as a faithful representation of an abstract group isomorphic to the matrix group. The code obtained in this manner depends upon the initial $\mathbf{X}$ on which the matrices operate. The regions $\mathcal{R}_i$ of equation (2) and hence also $P_e = P_{e,i}$ by (4) also depend on this choice. We suppose now that $\mathbf{X}$ is not an eigenvector of any of the O_i so that the code has M distinct words. It would be desirable to be able to choose an $\mathbf{X}$ of this sort to either minimize P_e or to maximize d, the nearest neighbor distance. We have not seen how to solve either of these problems in general. A few words about them are in order.

Consider first the problem of choosing $\mathbf{X}$ to maximize d. The squared

distance between $\mathbf{X}$ and $\mathbf{X}_i$ is

$$d^2(\mathbf{X}, \mathbf{X}_i) = |\mathbf{X} - \mathbf{X}_i|^2 = 2 - 2\mathbf{X}\cdot O_i\mathbf{X}$$

a monotone decreasing function of the quadratic form $\mathbf{X}\cdot O_i\mathbf{X}$ in the components of $\mathbf{X}$. This form is the cosine of the angle between $\mathbf{X}$ and $\mathbf{X}_i$. Solution of the maximum nearest neighbor distance is equivalent to finding

$$\alpha = \min_{\mathbf{X}} \max_{i} \mathbf{X}\cdot O_i\mathbf{X} \tag{9}$$

where the maximization over the matrices of $\mathcal{G}$ must omit the identity matrix. The quantity α is an invariant of the representation (is the same for every equivalent representation) and should ultimately be expressible in terms of properties of the group. The vector $\mathbf{X}$ which minimizes (9) is not unique: any word in the code generated by $\mathbf{X}$ would serve as well.

Given $\mathcal{G}$, we define two points $\mathbf{X}$ and $\mathbf{Y}$ on the unit sphere to be equivalent if one can be obtained from the other by an operation of $\mathcal{G}$. The surface of the sphere is thus divided into equivalence sets. A connected region on the sphere such that no two points in its interior are equivalent and such that every point on the sphere is equivalent to some point in the region will be called a fundamental region of $\mathcal{G}$. The maximum likelihood regions, $\mathcal{R}_i$, associated with any $\{M, n\}$ generated by $\mathcal{G}$ intersect the unit sphere in fundamental regions. These intersections are very special fundamental regions: they are convex and bounded by hyperplanes.

In attempting to minimize P_e or maximize d it clearly suffices to consider initial vectors $\mathbf{X}$ restricted to some fundamental region. It is natural then to ask what fundamental regions are possible for a given $\mathcal{G}$.

The situation is complicated. For some groups, the fundamental region is completely determined (up to equivalence under the group operations, of course): for other groups only certain features of its boundaries are determined, or no points at all may be determined.

For example, in the plane consider the group $\mathcal{G}_1$ generated by the three matrices corresponding to reflections in three lines through the origin that make angles of 60° with each other. This group is of order 6 and is a subgroup of the symmetry group of a regular hexagon having the given lines as diagonals. The fundamental region of this group is completely determined. It is a 60° arc of the unit circle with end points on two of the given lines. Any group code $\{6, 2\}$ generated by $\mathcal{G}_1$ has

this fundamental region for the intersection of one of its maximum likelihood regions $\mathfrak{R}_i$ with the circle. Choice of **X** serves only to position the initial vector within the maximum likelihood region. (When **X** is chosen to lie on one of the reflection lines, a {3, 2} results and the maximum likelihood region changes discontinuously to the union of two adjacent regions of the sort just discussed.)

On the other hand, consider the group $\mathcal{G}_2$ of rotational symmetries of the regular hexagon. $\mathcal{G}_2$, of order 6, consists of a 2×2 matrix representing a rotation of 60° in the plane along with the distinct powers of this matrix. *Any* 60° arc of the unit circle is a fundamental region for this group. Codes {6, 2} generated by $\mathcal{G}_2$ are equivalent for all choices of the initial vector **X**.

An example illustrating a partly determined fundamental region is obtained by considering the pure rotational symmetries of a cube in three dimensions. We imagine the cube centered at the origin and inscribed in a unit sphere. We speak in terms of the operations on the cube rather than in terms of the 3×3 matrices which describe these operations. $\mathcal{G}_3$, a group of order 24, consists of rotations of the cube by 120° around the body diagonals, of rotations by 90° about axes through the origin and centers of faces and of rotations of 180° about axes through the midpoints of edges and the origin. One axis of each kind is shown on Fig. 1. In discussing the fundamental region of $\mathcal{G}_3$ and codes generated by $\mathcal{G}_3$, it is convenient to speak of points on the cube, rather than on the circumscribed unit sphere. It is to be understood

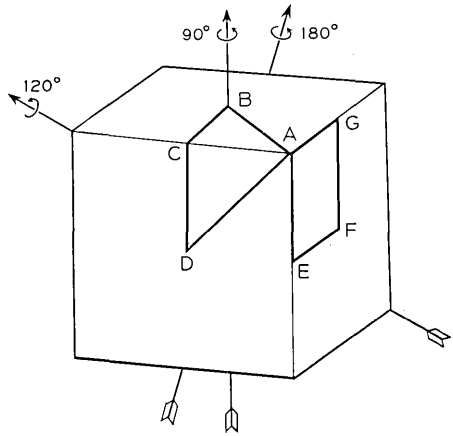


Fig. 1 — Example of partly-determined fundamental region.

then that when a point on the cube is mentioned it is really the corresponding point on the sphere obtained by projecting along a radius that is under discussion.

The vertices of the cube, the centers of faces and the midpoints of edges must all lie in boundaries of fundamental regions, for these points are on axes of rotation of $\mathcal{G}_3$. For example, a point distance ϵ from a vertex of the cube has two nearby equivalent points forming an equilateral triangle with the vertex at the center of the triangle. These three points cannot all lie in the interior of one fundamental region. The cube vertex therefore cannot be an interior point of a fundamental region. In fact at least 3 fundamental regions must meet at each vertex, at least 4 at each face center and at least two at each edge midpoint. Cube vertices and face centers must therefore be vertices of fundamental regions. Now all vertices of the cube are equivalent under $\mathcal{G}_3$ as are all face centers and all edge midpoints; no two of these three types of points are equivalent. A fundamental region of $\mathcal{G}_3$ must therefore contain at least one cube vertex and one face center among its vertices and at least one cube edge midpoint along its boundary.

Two distinct types of fundamental regions for $\mathcal{G}_3$ bounded by hyperplanes (great circles on the sphere) are shown in Fig. 1. Region *AEFG* is bounded by four hyperplanes. Edge midpoints are vertices of this type or region. Four fundamental regions surround each face center and each edge midpoint: three surround each cube vertex. Region *ABCD* is bounded by only three hyperplanes. Edge midpoints are no longer vertices of the fundamental region. Eight regions meet at each face center. The fundamental region *ABCD* corresponds to the maximum likelihood region of a group code having an initial vector (and hence all vectors) pass through a cube edge: region *AEFG* results when the initial vector passes through a face diagonal. All other positions of the initial vector give maximum likelihood regions that are fundamental regions bounded by four hyperplanes but not congruent to *AEFG*.

$\mathcal{G}_3$ is the irreducible representation of the symmetric group on four letters derived from the Young tableau[25] associated with the partition (2, 1, 1). The irreducible representation belonging to the partition (3, 1) is also three dimensional. It is equivalent to the group of symmetries of the regular tetrahedron and can be generated by reflections in planes through the centroid of the tetrahedron and its edges. The fundamental region here is completely determined. It is bounded by three of these generating reflection planes. Maximization of nearest neighbor distance for a {24, 3} generated by this group can be easily accomplished by

choosing the initial vector equidistant from the three bounding planes of the fundamental region.

More generally, Coxeter[23] has shown that if a real-irreducible finite group of $n \times n$ orthogonal matrices is generated by reflections, the fundamental region is completely determined, and in fact the region is bounded by n hyperplanes. Indeed, Coxeter has enumerated all possible groups of this sort. In dimensions n greater than 8, there are only three such groups, called by him A_n, B_n, and C_n of order $(n+1)!$, $2^{n-1}n!$, and $2^n n!$, respectively. These groups generate permutation modulation codes[15]—A_n generates Variant I codes, B_n generates Variant II codes with $\mu_1 = 0$, and C_n generates Variant II codes with $\mu_1 \neq 0$. The various permutation modulation codes are obtained by choosing the initial vector to lie in boundaries of various dimensionality of the fundamental regions of these groups.

Returning to the general case (when $\mathcal{G}$ is not generated by reflections), the real eigenvectors of the O_i with eigenvalue unity serve to determine landmarks of the fundamental region. Such an eigenvector must lie in the boundary of the region. If O_i has l such eigenvectors, their span is an l-dimensional boundary of the fundamental region. The situation has been studied by Robinson[24] in some detail, but no simple method of classifying the possible regions is available.

V. THE DIRECT SUM

Since any group code is equivalent to the weighted direct sum of codes generated by real-irreducible representations of a group, it is natural to investigate the relationship between interword distances in the sum code and the corresponding distances in the summand codes.

Let $\mathcal{G} = A_1, A_2, \cdots, A_g$ be a finite group of order g with A_1 the identity. Let $D^1(A)$ and $D^2(A)$ be two real-irreducible representations of $\mathcal{G}$ by real orthogonal matrices of dimensions l_1 and l_2 respectively. Let $\mathbf{X}_i = D^1(A_i)\mathbf{X}$, and $\mathbf{Y}_i = D^2(A_i)\mathbf{Y}$, $i = 1, 2, \cdots, g$ be group codes generated by D^1 and D^2. (Neither code need have g distinct vectors.) The direct sum code with weights λ_1 and λ_2 has vectors

$$\mathbf{Z}_i = \lambda_1 \mathbf{X}_i \oplus \lambda_2 \mathbf{Y}_i \qquad i = 1, 2, \cdots, g \tag{10}$$

$$\lambda_1^2 + \lambda_2^2 = 1, \qquad 0 < \lambda_1, \lambda_2 < 1$$

of $l = l_1 + l_2$ components. We seek to choose the weights so that the nearest neighbor distance, d, for the sum code $\mathbf{Z}$ is a maximum.

Let $\alpha_i = d^2(\mathbf{X}_i, \mathbf{X}_1)$ and $\beta_i = d^2(\mathbf{Y}_i, \mathbf{Y}_1)$ be the squared distance from the code word generated by A_i to the initial vector in the two codes,

$i = 1, 2, \cdots, g$, respectively. For the sum code we have

$$d^2(\mathbf{Z}_i, \mathbf{Z}_1) = \lambda_1^2\alpha_i + \lambda_2^2\beta_i$$

since the subspaces containing the **X** code and the **Y** code are orthogonal. The desired maximum nearest neighbor distance is thus

$$d^2 = \max_{0 \leq \lambda \leq 1} \min_{i \neq 1} [(1 - \lambda)\alpha_i + \lambda\beta_i] \tag{11}$$

where we have set $\lambda = \lambda_2^2$. The situation is illustrated in Fig. 2. Here we have taken $\alpha_2 \leqq \alpha_3 \leqq \cdots \leqq \alpha_g$ which we can do without loss of generality since this is merely a matter of giving names to the group elements. The bracketed expression on the right of equation (11) is plotted as the line segment with ordinate α_i at $\lambda = 0$ and ordinate β_i at $\lambda = 1$. We seek the highest point on the bottom boundary of this collection of lines, point P in Fig. 2.

Now any of the vectors $\mathbf{Y}_i$, $i = 1, 2, \cdots, g$, not just $\mathbf{Y}_1$, would serve to generate the **Y** code. We can seek a further maximization of the nearest neighbor distance (11) for the **Z** code by choice of the vector from the **Y** code to be called $\mathbf{Y}_1$. Stated otherwise, for the initial vector of the **Z** code we choose a particular vector $\mathbf{X}_1$ from the **X** code and to this we can add (directly) any of the vectors of the **Y** code. Now replacing $\mathbf{Y}_1$ by $\mathbf{Y}_l$ merely amounts to permuting the subscripts on the β_i of Fig. 2. The subscript i is replaced by k where $A_lA_i = A_k$. To combine the two codes to get the largest nearest neighbor distance, we must further

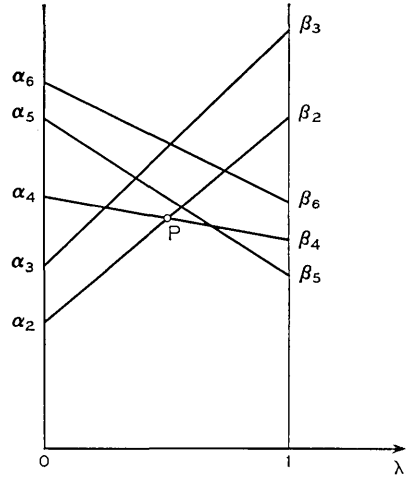


Fig. 2 — Maximum nearest neighbor distance.

maximize (11) over permutations of the β's corresponding to left translations of the group.

The maximization just considered was with respect to the manner of combining the two summand codes. There remains the matter of choosing $\mathbf{X}_1$ and $\mathbf{Y}_1$ to further increase (11). At first it might be thought that these vectors should be chosen to maximize the nearest neighbor distance in each of the summand codes. That this is not necessarily so can be seen from Fig. 2. Choosing $\mathbf{X}_1$ to increase the nearest neighbor distance in the code generated by D^1 would cause α_2 to increase. The line connecting α_2 and β_2 on the figure would move up. However, this change of $\mathbf{X}_1$ might cause α_4 to decrease by a larger amount so that point P on the figure moves downward. The situation is complicated.

The relationship between the maximum likelihood region for the sum code and the corresponding regions for the constituent codes is even more complicated in general. Let $\mathfrak{R}$ be the region belonging to $\mathbf{Z}_1$ of equation (10) and let $\mathfrak{R}^1$ and $\mathfrak{R}^2$ be corresponding regions for $\mathbf{X}_1$ and $\mathbf{Y}_1$ in the summand codes. We write $\mathbf{Z} = \lambda_1\mathbf{X} + \lambda_2\mathbf{Y}$ for a general point in the space of the direct sum representation where $\mathbf{X}$ and $\mathbf{Y}$ lie in the respective invariant subspaces of the summand codes. A point will lie in $\mathfrak{R}$ then if $|\mathbf{Z} - \mathbf{Z}_1| \leqq |\mathbf{Z} - \mathbf{Z}_i|$ for $i = 2, 3, \cdots, g$, or what is the same, if

$$\lambda_1^2 d^2(\mathbf{X}, \mathbf{X}_1) + \lambda_2^2 d^2(\mathbf{Y}, \mathbf{Y}_1) \leqq \lambda_1^2 d^2(\mathbf{X}, \mathbf{X}_i) + \lambda_2^2 d^2(\mathbf{Y}, \mathbf{Y}_i)$$

for $i = 2, 3, \cdots, g$. Thus if $\mathbf{X} \in \mathfrak{R}^1$ and $\mathbf{Y} \in \mathfrak{R}^2$ then $\mathbf{Z} \in \mathfrak{R}$, but the converse is not necessarily so in general.

A special case in which the converse holds is the following. It may happen that both the $\mathbf{X}$ code and the $\mathbf{Y}$ code have fewer than g distinct vectors. In the direct sum code (10) it may happen that each distinct vector of the $\mathbf{Y}$ code is paired at least once with each distinct vector of the $\mathbf{X}$ code. ($\mathcal{G}$ must be homeomorphic to the direct product of two groups.) In this case $\mathfrak{R}$ is the cartesian product of the two regions $\mathfrak{R}^1$ and $\mathfrak{R}^2$. The probability of no error for the sum code is given by $Q_e = Q_e^1(\lambda_1)Q_e^2(\lambda_2)$ where the factors are the probabilities of no error for the separate scaled summand codes. The information rate (3) for the sum code in this case is the weighted sum of the rates for the constituent codes

$$R = \frac{l_1}{l} R_1 + \frac{l_2}{l} R_2 .$$

We are better off using the code with the larger rate uncombined.

VI. THE CONFIGURATION MATRIX

Let $\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_M$ be a collection of unit vectors spanning $\mathcal{S}_n$. The configuration matrix of this code is the M by M matrix ρ whose elements $\rho_{ij} = \mathbf{X}_i \cdot \mathbf{X}_j$ are the cosines of the angles between the words. Equivalent codes have identical configuration matrices except for a possible relabeling of rows and columns. This configuration matrix is real, symmetric, non-negative definite and of rank n. The diagonal elements are unity and the off-diagonal elements are of magnitude no greater than unity.

Conversely, we have the following

Lemma: Every real, symmetric, M by M non-negative definite matrix of rank n with diagonal elements unity and off-diagonal elements of magnitude less than unity is the configuration matrix of a code of M unit vectors that span $\mathcal{S}_n$.

The proof of this lemma follows readily from the fact that a real symmetric M by M matrix ρ can be diagonalized by an orthogonal matrix O, that is, $O\rho O^{-1} = \Lambda$ where, since ρ is non-negative definite and of rank n, Λ has n positive diagonal elements and all other elements are zero. Without loss of generality we can take the first n diagonal elements of Λ, say $\lambda_{ii} = \lambda_i$, $i = 1, 2, \cdots, n$ to be the positive ones. From $\rho = O^{-1}\Lambda O$, it follows that

$$\rho_{ij} = \sum_{\mu} O_{\mu i} \sqrt{\lambda_\mu}\, O_{\mu j} \sqrt{\lambda_\mu} = \mathbf{X}_i \cdot \mathbf{X}_j$$

where $\mathbf{X}_i$ is a vector of n components, the μth component being $\sqrt{\lambda_\mu} O_{\mu i}$, $i = 1, 2, \cdots, M$. We have now exhibited M unit n-vectors whose configuration matrix is the given matrix ρ. We need now only show that they span $\mathcal{S}_n$. But we have written $\rho = \tilde{X}X$ where X is the matrix of M columns and n rows whose ith column is $\mathbf{X}_i$. The tilde denotes transpose. Since the rank of a product of matrices is not greater than the smaller of the ranks of the factors, it follows that X must be of rank n, for if it were of rank less than n, so also would be ρ contrary to hypothesis. The $\mathbf{X}_i$ therefore span $\mathcal{S}_n$.

For group codes, the rows of the configuration matrix are all permutations of the first row of the matrix as can be seen from Proposition 1. Indeed the structure of this matrix is closely related to the multiplication table of the group generating the code. Let the code vector $\mathbf{X}_i = D(A_i)\mathbf{X}$, $i = 1, 2, \cdots, M$ be generated by an orthogonal representation D of a group $\mathcal{G}$ with elements $A_1, A_2, \cdots, A_M$. Here A_1 is the identity and the code need not have M distinct vectors. Denote by $\theta(A_i)$ the

angle between $\mathbf{X}_j$ and $\mathbf{X}_1$. Then $\theta(A_i^{-1}) = \theta(A_j)$, $j = 1, 2, \cdots, M$ and the configuration matrix of the code is found to be given by

$$\rho_{ij} = \cos \theta(A_i^{-1}A_j)$$

$i, j = 1, 2, \cdots, M$. If $\rho_{1j} < 1$ for $j > 1$, then the code has M distinct vectors: if $1 = \rho_{1j_1} = \rho_{1j_2} = \cdots = \rho_{1j_h}$ with $1 = j_1 < j_2 \cdots < j_h$ and these are the only elements of value unity in the first row, then the code has M/h distinct vectors.

Conversely, we have

Theorem 1: *Let $x(A_j)$ be a real-valued function defined on the elements $A_1, A_2, \cdots, A_M$ of a group $\mathcal{G}$ of order M. Let $x(A_1) = 1$, where A_1 is the identity of the group, and let $x(A_j) = x(A_j^{-1})$, $j = 1, 2, \cdots, M$. If the M by M matrix ρ with elements $\rho_{ij} = x(A_i^{-1}A_j)$ is non-negative definite and of rank n, then there exists a group code $\{M' n\}$ generated by an n-dimensional orthogonal representation of $\mathcal{G}$ that has configuration matrix ρ. Here $M' = M/h$ where h is the number of different values of j for which $x(A_j) = 1$.*

Proof: The proof follows easily from the lemma. We can find M unit vectors $\mathbf{X}_i$ (not necessarily distinct) that span $\mathcal{S}_n$ such that $\rho_{ij} = \mathbf{X}_i \cdot \mathbf{X}_j$. Without loss of generality we can suppose that $\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_n$ are linearly independent. Now an n by n real matrix is determined by specifying its effect on n vectors that span its carrier space. For each $\mu = 1, 2, \cdots, M$ we determine the n by n matrix $D(A_\mu)$ by specifying its effect on $\mathbf{X}_1, \cdots, \mathbf{X}_n$, namely that $D(A_\mu)\mathbf{X}_i = \mathbf{X}_{l(i,\mu)}$, $i = 1, 2, \cdots, n$ where $A_\mu A_i = A_{l(i,\mu)}$. Now $\mathbf{X}_i \cdot \mathbf{X}_j = \rho_{ij} = x(A_i^{-1}A_j) = x(A_i^{-1}A_\mu^{-1}A_\mu A_j) = x(A_{l(i,\mu)}^{-1}A_{l(j,\mu)}) = \mathbf{X}_{l(i,\mu)} \cdot \mathbf{X}_{l(j,\mu)}$, so that $D(A_\mu)$ preserves the angles between n vectors spanning its carrier space. It is easy then to show that $D(A_\mu)$ preserves the angle between any two vectors and hence is an orthogonal matrix. For $j > n$,

$$\mathbf{X}_j = \sum_{h=1}^{n} \alpha_{jh}\mathbf{X}_h$$

for some set of α's. Using this representation and the orthogonality of $D(A_\mu)$, it is now easy to show that $D(A_\mu)\mathbf{X}_i = \mathbf{X}_{l(i,\mu)}$ for $i = 1, 2, \cdots, M$. The fact that D is a representation then follows readily.

Theorem 1 permits an interesting reformulation of the problem of finding an $\{M, n\}$ of largest nearest neighbor distance generated by a representation of $\mathcal{G}$. Form the modified multiplication table of $\mathcal{G}$,—an M by M array of group elements with $A_i^{-1}A_j$ in the ith row and jth column. From this table we construct a symmetric M by M matrix ρ

by replacing the group identity A_0, say, by unity, by replacing both A_1 and A_1^{-1} by the variable x_1, A_2 and A_2^{-1} by x_2, and so on. If $\mathcal{G}$ has exactly m self-reciprocal elements, there will be $K = m - 1 + (g - m)/2$ variables in ρ. The condition that ρ be non-negative definite and of rank not greater than n obtained by conditioning certain minors of ρ gives rise to polynomial constraints in the variables $x_1, \cdots, x_K$. To find the code of largest nearest neighbor distance, we must minimize $\max_i x_i$ subject to these constraints.

We notice in closing this section that the configuration matrix of an $\{M, n\}$ generated by a group $\mathcal{G}$ of order M commutes with all the matrices of the regular representation of $\mathcal{G}$ (see the appendix). Using Schur's lemma, one can then arrive at a canonical representation for configuration matrices that involves the irreducible representations of $\mathcal{G}$. But we do not pursue this topic further here.

VII. SOME THEOREMS ON DISTANCES

We now adopt the notation of the appendix. Let $\mathcal{G}$ be an abstract group of order g with elements $E, A, B, \cdots$ where E is the identity. Let $D(E), D(A), \cdots$ be a real-irreducible representation of $\mathcal{G}$ by $n \times n$ (real) orthogonal matrices. From an initial unit vector $\mathbf{X} = \mathbf{X}_E$ the representation generates a code by $\mathbf{X}_R = D(R)\mathbf{X}_E$, R runs through $\mathcal{G}$. We denote the squared distance from $\mathbf{X}_R$ to $\mathbf{X}_S$ by $d^2(\mathbf{X}_R, \mathbf{X}_S)$. We have then

$$\begin{aligned} d^2(\mathbf{X}_A, \mathbf{X}) &= 2 - 2 \sum_{i,j=1}^{n} D(A)_{ij} x_i x_j \\ &= d^2(\mathbf{X}_{RA}, \mathbf{X}_R) \end{aligned} \tag{12}$$

for every R and $A \in \mathcal{G}$. Here $x_1, x_2, \cdots, x_n$ are the components of $\mathbf{X}$.

For codes generated from real-irreducible representations in this manner, a number of interesting distance sums are independent of the choice of the initial vector $\mathbf{X}$.

Theorem 2: *Let $D(R)$ be the matrices of a real-irreducible orthogonal representation of a group of order g. Let $\mathbf{X}_R = D(R)\mathbf{X}$. If $D(R)$ is not the trivial one-dimensional representation $D(R) = 1$, then*

$$\sum_{R \in \mathcal{G}} d^2(\mathbf{X}_R, \mathbf{X}) = 2g$$

independent of the unit vector $\mathbf{X}$.

This is really a special case of the more general

Theorem 3: *For any code generated from the initial unit vector* $\mathbf{X}$ *by a real-irreducible orthogonal representation* D *of* $\mathcal{G}$,

$$\sum_{R \in \mathcal{G}} d^2(\mathbf{X}_{R^m}, \mathbf{X}) = 2g(1 - \mu_m)$$

where

$$\mu_m = \frac{1}{gn} \sum_{R \in \mathcal{G}} \chi(R^m)$$

is a constant independent of $\mathbf{X}$. *Here* $\chi(R) = \operatorname{Tr} D(R)$ *is the character of* R *in the representation.*

Proof: Consider the matrix

$$A \equiv \sum_{R \in \mathcal{G}} D(R^m) = \sum_{R \in \mathcal{G}} D(R)D(R) \cdots D(R)$$

where there are m factors in the summand. Since the representation is by orthogonal matrices, $\tilde{D}(R) = D^{-1}(R) = D(R^{-1})$ where the tilde denotes transpose. Thus

$$\tilde{A} = \sum_{R \in \mathcal{G}} D(R^{-1}) \cdots D(R^{-1}) = A$$

since as R runs through $\mathcal{G}$ so does R^{-1}. The matrix A is thus symmetric.

We next show that A commutes with all the matrices $D(R)$. By a theorem quoted in the appendix we can then conclude that $A = \alpha I$ where I is the unit matrix. To see that A commutes with $D(R)$, consider

$$AD(R) = \sum_{S \in \mathcal{G}} D(S)^{m-1} D(S) D(R) = \sum_{S \in \mathcal{G}} D(S)^{m-1} D(SR).$$

Now set $SR = T$ so that $S = TR^{-1}$. Then

$$\begin{aligned} AD(R) &= \sum_{T \in \mathcal{G}} D(TR^{-1})^{m-1} D(T) \\ &= \sum D(TR^{-1})D(TR^{-1}) \cdots D(TR^{-1})D(T) \\ &= \sum_{T \in \mathcal{G}} D(T)D(R^{-1}T)D(R^{-1}T) \cdots D(R^{-1}T) \\ &= \sum_{U \in \mathcal{G}} D(RU)D(U)^{m-1} = D(R) \sum_{U \in \mathcal{G}} D(U)^m = D(R)A \end{aligned}$$

where we have used the substitution $U = R^{-1}T$.

From equation (12) we have

$$\begin{aligned} \sum_{R \in \mathcal{G}} d^2(\mathbf{X}_{R^m}, \mathbf{X}) &= 2g - 2 \sum_{i,j=1}^{n} x_i x_j \sum_{R \in \mathcal{G}} D(R^m)_{ij} \\ &= 2g - 2 \sum_{i,j=1}^{n} x_i x_j A_{ij} = 2(g - \alpha) \end{aligned}$$

by the diagonal property of A just established. To find α consider the trace of A. We have

$$\operatorname{Tr} A = \operatorname{Tr} \alpha I = \alpha n = \sum_{R \in \mathcal{G}} \operatorname{Tr} D(R^m) = \sum_{R \in \mathcal{G}} \chi(R^m).$$

The theorem then follows.

To establish Theorem 2, notice that for the trivial representation $D^1(R) = 1$, we have $\chi^1(R) = 1$. For the character $\chi(R)$ of any other nonequivalent real-irreducible representation we then have

$$\sum_{R \in \mathcal{G}} \chi^1(R)\chi(R) = \sum_{R \in \mathcal{G}} \chi(R) = 0$$

by the orthogonality relations (appendix). Using this fact and setting $m = 1$ in Theorem 3 yields Theorem 2.

Theorem 4: *Let* $\mathcal{C}$ *be a class of* n_c *elements of* $\mathcal{G}$ *with character* $\chi(\mathcal{C})$. *For any code generated from the initial unit vector* $\mathbf{X}$ *by a real-irreducible orthogonal representation* D *of* $\mathcal{G}$,

$$\sum_{R \in \mathcal{C}} d^2(\mathbf{X}_R , \mathbf{X}) = 2n_c\left(1 - \frac{1}{n}\chi(\mathcal{C})\right) \tag{13}$$

independent of the unit vector $\mathbf{X}$.

Proof:

$$\sum_{R \in \mathcal{C}} d^2(\mathbf{X}_R , \mathbf{X}) = 2n_c - 2 \sum x_i x_i \sum_{R \in \mathcal{C}} D(R)_{ii} . \tag{14}$$

Now consider the matrix

$$B = \sum_{R \in \mathcal{C}} D^\alpha(R) = \frac{n_c}{g} \sum_{S \in \mathcal{G}} D^\alpha(SRS^{-1}) = \frac{n_c}{g} \sum_{S \in \mathcal{G}} D^\alpha(S)D^\alpha(R)D^\alpha(S^{-1})$$

where $D^\alpha(R)$ is an irreducible (over the complex field) representation of dimension m of $\mathcal{G}$. Now B commutes with all the matrices of D^α since

$$BD^\alpha(T) = \frac{n_c}{g} \sum_{S \in \mathcal{G}} D^\alpha(S)D^\alpha(R)D^\alpha(S^{-1}T)$$

$$= \frac{n_c}{g} \sum_{U \in \mathcal{G}} D^\alpha(TU^{-1})D^\alpha(R)D^\alpha(U) = D^\alpha(T)B$$

where we have set $S^{-1}T = U$. By Schur's lemma, $B = kI$ where I is the m by m unit matrix. Taking traces we have

$$\operatorname{Tr} B = \operatorname{Tr} \sum_{R \in \mathcal{C}} D^\alpha(R) = n_c\chi^\alpha(\mathcal{C}) = \operatorname{Tr} kI = km$$

so that

$$B = \sum_{R \in \mathcal{C}} D^{\alpha}(R) = \frac{n_c}{m} \chi^{\alpha}(\mathcal{C}) I. \tag{15}$$

If now the real-irreducible orthogonal representation D is also irreducible, by equation (15) the inner sum in (14) is $(n_c/n)\chi(\mathcal{C})\delta_{ij}$ and the theorem (13) follows at once.

Suppose now that D is not irreducible. Then (see appendix) D is equivalent to an orthogonal representation of the form

$$D'(R) = \begin{bmatrix} U^{\alpha}(R) & V^{\alpha}(R) \\ -V^{\alpha}(R) & U^{\alpha}(R) \end{bmatrix} \tag{16}$$

where $D^{\alpha}(R) = U^{\alpha}(R) + iV^{\alpha}(R)$ is an irreducible representation by unitary matrices and U^{α} and V^{α} are real and of dimension m where $n = 2m$. We can suppose the D of equation (14) to be of the form (16). Now let

$$\hat{B} = \sum_{R \in \mathcal{C}} D'(R)$$

and set

$$U = \frac{1}{\sqrt{2}} \begin{bmatrix} I & iI \\ iI & I \end{bmatrix}$$

where as before I is the m by m unit matrix. One then finds by direct computation that

$$U^{-1}\hat{B}U = \sum_{R \in \mathcal{C}} \begin{bmatrix} D^{\alpha}(R) & 0 \\ 0 & D^{\alpha}(R)^* \end{bmatrix}$$

$$= \frac{n_c}{m} \begin{bmatrix} \chi^{\alpha}(\mathcal{C}) I & 0 \\ 0 & \chi^{\alpha}(\mathcal{C})^* I \end{bmatrix} \equiv H$$

where the middle equality follows from equation (15). Now let $\chi^{\alpha}(\mathcal{C}) = \mu + i\nu$ with μ and ν real. Direct computation gives

$$\hat{B} = UHU^{-1} = \frac{n_c}{m} \begin{bmatrix} \mu I & \nu I \\ -\nu I & \mu I \end{bmatrix}. \tag{17}$$

The right of (14) is

$$2n_c - 2 \sum \hat{B}_{ii} x_i x_i$$

and using (17) this becomes

$$2n_c - \frac{2n_c}{m}\mu \sum x_i^2 = 2n_c\left(1 - \frac{\mu}{m}\right).$$

From equation (16), however, $\chi(\mathcal{C}) = 2\mu$, so that (13) then follows and the theorem is proved in all cases.

Since every group code can be thought of as the direct sum of codes generated by real-irreducible representations, and since squared distance in the sum code is the sum of squared distances in the separate codes, Theorems 2, 3, and 4 have ready analogues for all group codes. For example, if a group code does not contain the identity representation, then

$$\sum_{R \in \mathcal{G}} d^2(\mathbf{X}_R , \mathbf{X}) = 2g.$$

Theorems 3 and 4 hold for group codes in general when $\chi(R)$ is replaced by the weighted sum $\sum \lambda_i^2 \chi^i(R)$ of the characters of the constituent real-irreducible codes.

Another theorem of interest concerning codes generated from any group of orthogonal matrices arises from the fact (12) that $d^2(\mathbf{X}_A , \mathbf{X}) = d^2(\mathbf{X}_{RA} , \mathbf{X}_R)$. Let there be a point of the code distant d from the point $\mathbf{X}_E$. Starting from $\mathbf{X}_E$, we imagine moving from word to word of the code restricting our moves so that from any word we can move only to a word distant d away. We shall call the collection of words that can be reached from $\mathbf{X}_E$ in this manner "a d chain starting from $\mathbf{X}_E$". $\mathbf{X}_E$ is to be included in this chain.

Theorem 5: *Let the words of a d chain starting from* $\mathbf{X}_E$ *be* $\mathbf{X}_E$, $\mathbf{X}_{A_1}$, $\mathbf{X}_{A_2}$, $\cdots$ $\mathbf{X}_{A_h}$. *Then the group elements* $E, A_1, A_2, \cdots, A_h$ *form a subgroup* $\mathfrak{H}$ *of* $\mathcal{G}$. *The group elements whose corresponding words are distant d from* $\mathbf{X}_E$ *form a set of generators for* $\mathfrak{H}$. *If* $\mathfrak{H}$ *is a proper subgroup of* $\mathcal{G}$, *then from any word corresponding to a group element not in* $\mathfrak{H}$, *a new d chain may be formed and the group elements corresponding to the points of this new d chain will form a coset of* $\mathfrak{H}$.

Proof: Suppose all the points distant d from $\mathbf{X}_E$ are $\mathbf{X}_{A_1}, \mathbf{X}_{A_2}, \cdots, \mathbf{X}_{A_m}$. Let us construct a table of group elements in the following manner. The first row is $E, A_1, A_2, \cdots, A_m$. The K + 1st row of the table is formed from the preceding K rows as follows. We examine the elements of the table in order, reading from left to right in the first row, then from left to right in the second row, and so on. Let R be the first element arrived at in reading the first K rows that does not appear in the first

column, rows 1, 2, $\cdots$, K. The the K + 1st row is to be

$$R, RA_1, RA_2, \cdots, RA_m .$$

The table thus appears

$$\begin{array}{llll} E, & A_1, & A_2, & A_m \\ A_1, & A_1^2, & A_1A_2, & A_1A_m \\ A_2, & A_2A_1, & A_2, & A_2A_m \\ \vdots & & & \\ A_m, & A_mA_1, & A_mA_2, & A_m^2 \\ B, & BA_1, & BA_2, & BA_m \\ \vdots & & & \\ R, & RA_1, & RA_2, & RA_m \end{array}$$

When j rows have been written and every element in these j rows has appeared once in the first column the process is stopped and the table is considered complete. The table can have at most g rows. Now from $d^2(\mathbf{X}_R, \mathbf{X}_E) = d^2(\mathbf{X}_{SR}, \mathbf{X}_S)$, it follows that the words represented by the elements in the 2nd, 3rd, $\cdots$, m + 1st columns of the Kth row are all distant d from the word represented by the element in the first column of the Kth row. Furthermore, these m words are all the words of the code that are distant d from the word represented by the element in the first column of the Kth row. Thus the elements of the first column of the table give the points of the d chain starting from $\mathbf{X}_E$. That these elements of the first column form a group $\mathfrak{IC}$ and that $A_1, A_2, \cdots, A_m$ are generators of $\mathfrak{IC}$ is clear from the method of constructing the table, for we have formed all possible distinct products of the A's and listed the distinct elements thus obtained in the first column. Let $\mathfrak{IC}$ be a proper subgroup of $\mathcal{G}$ and let S be an element of $\mathcal{G}$ not in $\mathfrak{IC}$. If we multiply every element in the above table by S, we obtain a new table giving all the points that can be reached from point $\mathbf{X}_S$ by steps of distance d. The first column of this table lists the points of the d chain starting from $\mathbf{X}_S$ and the corresponding elements are just the coset $S\mathfrak{IC}$ of $\mathfrak{IC}$.

VIII. BINARY GROUP CODES

The group codes (n, k) for the binary channel introduced in Ref. 21 are group codes in the present sense. Each word of an (n, k) code is an

n-place binary sequence. Replace each zero by 1 and replace each 1 by -1 in each word. Then write each word (a sequence of ± 1's) as a diagonal $n \times n$ matrix. This collection of 2^k $n \times n$ orthogonal matrices forms an Abelian group $\mathfrak{B}_k$ that is isomorphic to the k-fold direct product of the simple two element Abelian group. The matrices generate the code by operating on the n-vector $(1, 1, 1, \cdots, 1)$. The real-irreducible representations of this group are all one dimensional. There are 2^k of them. The representation by $n \times n$ matrices just considered is already exhibited in reduced form as the direct sum of n of these real-irreducible representations.

IX. CONCLUDING REMARKS

The foregoing paragraphs outline some of the theory of group codes for the Gaussian channel. The development of this subject is clearly incomplete: we have raised more questions than we have answered. Perhaps the outstanding problem is that of finding a tractable method of choosing the initial vector to maximize the nearest neighbor distance.

There is a great abundance of groups of arbitrarily large order that can be examined from the point of generating group codes. The symmetric group and the hyperoctahedral group appear most promising for initial investigation since their structure and irreducible representations (which are all real) are comparatively well understood.

APPENDIX

Review of Group Representation Theory[25]

Let $\mathcal{G}$ be a finite group of order g with elements $E, A, B, \cdots$. The letters R and S will be used for the general element of $\mathcal{G}$ and E will denote the identity of $\mathcal{G}$. As R runs through $\mathcal{G}$, the distinct elements of the set RAR^{-1} are said to form a class of $\mathcal{G}$. The elements A and B are said to belong to the same class of $\mathcal{G}$ if there exists an S such that $A = SBS^{-1}$. $\mathcal{G}$ can be divided uniquely into a union of classes, no two classes containing a common element. The number of elements in a class of $\mathcal{G}$ is a divisor of g.

If $\mathcal{H}$ is a subgroup of $\mathcal{G}$ and if $\mathcal{H}$ is of order h, then h is a divisor of g and the number g/h is called the index of $\mathcal{H}$ under $\mathcal{G}$. If, for every R in $\mathcal{H}$, all elements of $\mathcal{G}$ in the same class as R are also contained in $\mathcal{H}$, then $\mathcal{H}$ is said to be a self-conjugate subgroup of $\mathcal{G}$. A subgroup $\mathcal{H}$ of $\mathcal{G}$ is said to be proper if $h < g$.

The matrices in what follows are assumed to have elements in the field of complex numbers.

If to every element R of a finite group $\mathcal{G}$ there corresponds an n by n nonsingular matrix $D(R)$ and if $D(R)D(S) = D(RS)$, the collection of matrices $\Delta = \{D(R), R \text{ runs through } \mathcal{G}\}$ is said to form an n-dimensional representation of $\mathcal{G}$. The matrices of Δ form a group under matrix multiplication. If the correspondence between the matrices of Δ and the elements of $\mathcal{G}$ is one-to-one, Δ is said to be a faithful representation of $\mathcal{G}$. If for some $R \neq S$, $D(R) = D(S)$, Δ is said to be an unfaithful representation of $\mathcal{G}$. The matrix $D(E)$ is always the n by n unit matrix. If a representation is unfaithful, the elements represented by $D(E)$ form a self-conjugate subgroup of $\mathcal{G}$, say of order h, and to each matrix of Δ correspond exactly h elements of $\mathcal{G}$. Δ contains g/h distinct matrices. If $D(E)$, $D(A)$, $\cdots$ is an n dimensional representation of $\mathcal{G}$, so is $MD(E)M^{-1}$, $MD(A)M^{-1}$, $\cdots$ where M is any nonsingular n by n matrix. The two representations Δ and $M\Delta M^{-1}$ are called equivalent. Every representation of a finite group is equivalent to a representation by unitary matrices. Henceforth we shall be concerned only with such unitary representations.

A finite collection of n by n matrices $O_1, O_2, \cdots, O_K$ is said to be reducible if there exists an n by n unitary matrix U such that for $i = 1, 2, \cdots, K$ we have

$$UO_iU^{-1} = \left| \begin{array}{c|c} A_i & D \\ \hline C & B_i \end{array} \right|$$

where A_i is an l by l matrix, B_i is an n–l by n–l matrix, $0 < l < n$, C is an n–l by l matrix all of whose elements are zero, and D is an l by n–l matrix all of whose elements are zero. It is assumed that l is independent of i. A collection of matrices that is not reducible is said to be irreducible.

Every finite group has exactly as many nonequivalent irreducible representations as it has classes. If $l_1, l_2, \cdots, l_c$ are the dimensions of all the nonequivalent irreducible representations of $\mathcal{G}$, of order g, then

$$\sum_1^c l_i^2 = g.$$

If $D^\alpha(R)_{\mu\nu}$ is the element in the μth row and νth column of the matrix representing R in the l_α-dimensional irreducible representation, α, of $\mathcal{G}$, then

$$\sum_{R \in \mathcal{G}} D^\alpha(R)_{\mu\nu} D^\alpha(R)^*_{\mu'\nu'} = \delta_{\mu\mu'}\, \delta_{\nu\nu'} g/l_\alpha \qquad \mu, \mu', \nu, \nu' = 1, 2, \cdots, l_\alpha .$$

Here * means complex conjugate and δ is the usual Kronecker symbol. If the matrices $D^\beta(R)$ form an l_β dimensional irreducible representation

of $\mathcal{G}$ not equivalent to the representation α, then

$$\sum_{R \in \mathcal{G}} D^{\alpha}(R)_{\mu\nu} D^{\beta}(R)^{*}_{\mu'\nu'} = 0,$$

$$\mu, \nu = 1, 2, \cdots, l_{\alpha}, \quad \mu', \nu' = 1, 2, \cdots, l_{\beta}.$$

If $D(R)$ is the n by n matrix representing R in the representation Δ, the trace of $D(R)$, namely

$$\chi(R) = \sum_{\mu=1}^{n} D(R)_{\mu\mu},$$

is called the character of R in the representation Δ. If R and S are in the same class of $\mathcal{G}$, then $\chi(R) = \chi(S)$, for any representation of $\mathcal{G}$. The characters of the irreducible representations α and β of $\mathcal{G}$ satisfy the orthogonality conditions

$$\sum_{R \in \mathcal{G}} \chi^{\alpha}(R) \chi^{\beta}(R)^{*} = g \, \delta_{\alpha\beta}.$$

Here $\delta_{\alpha\beta}$ is unity if α and β are equivalent representations and is zero otherwise.

Let Δ be any representation of $\mathcal{G}$ with character $\chi(R)$. Let the characters of the irreducible representations of $\mathcal{G}$ be $\chi^{j}(R)$, $j = 1, 2, \cdots, c$ where c is the number of nonequivalent irreducible representations of $\mathcal{G}$ (= number of classes of $\mathcal{G}$). Then $\chi(R)$ may be written uniquely in the form

$$\chi(R) = \sum_{j=1}^{c} a_j \chi^{j}(R), \qquad \text{all } R \text{ in } \mathcal{G},$$

where the a_j are nonnegative integers independent of R. In fact,

$$a_j = \frac{1}{g} \sum_{R \in \mathcal{G}} \chi(R) \chi^{j}(R)^{*}.$$

The representation Δ is said to contain the irreducible representation j a_j times and there exists a unitary matrix U independent of R such that

$$UD(R)U^{-1} = \begin{vmatrix} D^{i}(R) & 0 \cdots 0 \\ 0 & D^{j}(R) \cdots 0 \\ \vdots & \vdots \quad \vdots \\ 0 & 0 \quad D^{k}(R) \end{vmatrix} \qquad \text{all } R \text{ in } \mathcal{G}$$

where $D^i(R)$, $D^j(R)$, $D^k(R)$ etc., are matrices of the ith, jth, kth, etc., irreducible representation of $\mathcal{G}$, and 0 stands for the appropriate matrix with all elements zero. The jth irreducible representation will occur exactly a_j times among $D^i(R)$, $D^j(R)$, $D^k(R)$ and so on.

Every group $\mathcal{G}$ possesses a faithful representation, called the regular representation Γ, that consists of g by g permutation matrices. The rows and columns of these matrices can be labelled by the elements of $\mathcal{G}$. The entry in row R and column S of the matrix representing T is unity if $R = TS$ and is zero otherwise. The regular representation is reducible: it contains the irreducible representation D^α exactly l_α times, $\alpha = 1, 2, \cdots, c$.

Let $D'(R)$ and $D''(R)$, R runs through $\mathcal{G}$, be irreducible representations of $\mathcal{G}$ of dimension d' and d'' respectively. Let the matrix H satisfy $D'(R)H = HD''(R)$ for all R in $\mathcal{G}$. Then either H is the zero matrix or H is square and nonsingular so that $d' = d''$, and the two representations are equivalent. A matrix that commutes with all the matrices of an irreducible representation of $\mathcal{G}$ is a multiple of the unit matrix. These statements are known as Schur's lemma.

Much of the foregoing remains valid with minor modifications when the number field in question is the real rather than the complex numbers. One easily finds that every real representation of a finite group is equivalent (over the reals) to a representation by orthogonal matrices. The only real symmetric matrix that commutes with all the matrices of a real-irreducible representation is a multiple of the unit matrix. If $D^\alpha(R)$ and $D^\beta(R)$ are nonequivalent real-irreducible representations by real orthogonal matrices, respectively of dimension l_α and l_β, then

$$\sum_{R \in \mathcal{G}} D^\alpha(R)_{\mu\nu} D^\beta(R)_{\mu'\nu'} = 0,$$

$$\mu, \nu = 1, 2, \cdots, l_\alpha \quad \mu', \nu' = 1, 2, \cdots, l_\beta,$$

$$\sum_{R \in \mathcal{G}} [D^\alpha(R)_{\mu\nu} D^\alpha(R)_{\mu'\nu'} + D^\alpha(R)_{\mu\nu'} D^\alpha(R)_{\mu'\nu}] = 2\,\delta_{\mu\mu'}\,\delta_{\nu\nu'} g/l_\alpha,$$

$$\mu, \nu, \mu', \nu' = 1, 2, \cdots l_\alpha.$$

For the characters one has

$$\sum_{R \in \mathcal{G}} \chi^\alpha(R)\chi^\beta(R) = 0$$

if the representations α and β are not equivalent, while

$$\sum_R [\chi^\alpha(R)\chi^\alpha(R) + \chi^\alpha(R^2)] = 2g.$$

Every real-irreducible representation that is reducible (over the complex numbers) is equivalent to the direct sum of an irreducible representation and its complex conjugate. If the irreducible unitary representation $D(R) = U(R) + iV(R)$, with U and V real, is not equivalent to a real orthogonal representation, then

$$\left(\begin{array}{c|c} U(R) & V(R) \\ \hline -V(R) & U(R) \end{array}\right) \tag{18}$$

is a real-irreducible representation by real orthogonal matrices.

For an irreducible representation $D(R)$ with character $\chi(R)$, the sum

$$h = \frac{1}{g} \sum_{R \in \mathcal{G}} \chi(R^2)$$

can have only one of the three different values $0, \pm 1$. If $h = 1$, $D(R)$ is equivalent to a representation by real orthogonal matrices. If $h = -1$, the representation D is equivalent to its complex conjugate, but is not equivalent to a real representation. A real-irreducible representation can be made from each irreducible representation D having $h = -1$ by forming real matrices of the form (18), where U and V are the real and imaginary parts of D. Finally, if $h = 0$, D is not equivalent to its complex conjugate and is not equivalent to a real representation. Nonequivalent irreducible representations for which $h = 0$ occur then in complex conjugate pairs. Each such pair gives rise to a single real-irreducible representation through the recipe (18). Thus, finally, if h has the value 0 for exactly $2p$ of the c nonequivalent irreducible representations of $\mathcal{G}$, then $\mathcal{G}$ has exactly $c - p$ nonequivalent real-irreducible representations.

REFERENCES

1. Kotel'nikov, V. A., Thesis, Molotov Energy Institute, Moscow, 1947, translated as *The Theory of Optimal Noise Immunity,* New York: McGraw-Hill Book Co., 1959.
2. Shannon, C. E., "A Mathematical Theory of Communication," B.S.T.J., *27,* Nos. 3 and 4 (July and October 1948), pp. 379–423; 623–656.
3. Shannon, C. E., "Communication in the Presence of Noise," Proc. IRE, *37,* (January 1949), pp. 10–21.
4. Rice, S. O., "Communication in the Presence of Noise—Probability of Error for Two Encoding Schemes," B.S.T.J., *29,* No. 1 (January 1950), pp. 60–93.
5. Gilbert, E. N., "A Comparison of Signalling Alphabets," B.S.T.J., *31,* No. 3 (May 1952), pp. 504–522.
6. Shannon, C. E., "Probability of Error for Optimal Codes in a Gaussian Channel," B.S.T.J., *38,* No. 3 (May 1959), pp. 611–656.
7. Wolfowitz, J., *Coding Theorems of Information Theory,* Berlin: Springer-Verlag, 1961.
8. Stutt, C. A., "Information Rate in a Continuous Channel for Regular-Simplex Codes," IRE Trans. *IT-6* (December 1960), pp. 516–522.

9. Balakrishnan, A. V., "A Contribution to the Sphere-Packing Problem of Communication Theory," J. Math. Anal. Applications, *3* (December 1961), pp. 485–506. "Signal Selection Theory for Space Communication Channels," Chapter 1 in *Advances in Communication Systems,* ed. A. V. Balakrishnan, New York: Academic Press, 1965.
10. Slepian, D., "Bounds on Communication," B.S.T.J., *42,* No. 3 (May 1963), pp. 681–707.
11. Gallager, R. G., "A Simple Derivation of the Coding Theorem and Some Applications," IEEE Trans., *IT-11* (January 1965), pp. 3–18.
12. Wyner, A. D., "Capabilities of Bounded Discrepancy Decoding," B.S.T.J., *44,* No. 6 (July–August 1965), pp. 1061–1122.
13. Wozencraft, J. M. and Jacobs. I. M., *Principles of Communication Engineering,* New York: John Wiley & Sons, 1965.
14. Weber, C. L., "On Optimal Signal Selection for M-ary Alphabets with Two Degrees of Freedom," IEEE Trans., *IT-11* (April 1965), pp. 299–300, and "New Solution to the Signal Design Problem for Coherent Channels," IEEE Trans., *IT-12* (April 1966), pp. 161–167.
15. Slepian, D., "Permutation Modulation," Proc. IEEE *53* (March 1965), pp. 228–236.
16. Zetterberg, L. H., "A Class of Codes for Polyphase Signals on a Band-limited Gaussian Channel," IEEE Trans., *IT-11* (July 1965), pp. 385–395. "Detection of a Class of Coded and Phase-Modulated Signals," IEEE Trans., *IT-12* (April 1966), pp. 153–161.
17. Peterson, W. W. and Kasami, T., "Reliability Bounds for Polyphase Codes for the Gaussian Channel," Scientific Report No. 3 (July 1965), Dept. of Elec. Eng., U. of Hawaii. Abstract appears in IEEE Trans., *IT-12,* No. 2 (April 1966), p. 277.
18. Reed, I. S. and Scholtz, R. A., "N-Orthogonal Phase-Modulated Codes," IEEE Trans., *IT-12* (July 1966), pp. 388–395.
19. Scholtz, R. A. and Weber, C. L., "Signal Design for Phase-Incoherent Communications," IEEE Trans., *IT-12* (October 1966), pp. 456–463.
20. Landau, H. J. and Slepian, D., "On the Optimality of the Regular Simplex Code," B.S.T.J., *45,* No. 8 (October 1966), pp. 1247–1272.
21. Slepian, D., "A Class of Binary Signaling Alphabets, B.S.T.J., *35,* No. 1 (January 1956), pp. 203–234.
22. Dunn, James G., "Coding for Continuous Sources and Channels," Thesis, Electrical Engineering Department, Columbia University, May 1965.
23. Coxeter, H. S. M., *Regular Polytopes,* New York: The Macmillan Company, 1963.
24. Robinson, G. de B., "On the Fundamental Region of an Orthogonal Representation of a Finite Group," Proc. London Math. Soc., *43,* Sec. 2 (1937), pp. 289–301.
25. Boerner, H., *Representation of Groups,* Amsterdam: North-Holland Publishing Co., 1963.
    Murnaghan, F. D., *The Theory of Group Representations,* Baltimore: Johns Hopkins Press, 1938.
    Weyl, H., *The Classical Groups,* Princeton: Princeton University Press, 1946.
    Wigner, E. P., *Group Theory,* New York: Academic Press, 1959.
    Lomont, J. S., *Applications of Finite Groups,* New York: Academic Press, 1959.

Contributors to This Issue

JACK M. HOLTZMAN, B.E.E., 1958, City College of New York, M.S., 1960, University of California (Los Angeles); Ph.D., 1967, Polytechnic Institute of Brooklyn; Hughes Aircraft Company, 1958–1963; Bell Telephone Laboratories, 1963—. Mr. Holtzman has worked in various aspects of systems and control theory. Member, SIAM, Sigma Xi.

LUDWIK KURZ, B.E.E., 1951, and M.E.E., 1955, City College of New York, Eng. Sc.D., 1961, New York University. Mr. Kurz is a professor of electrical engineering and project director in the Laboratory for Electroscience Research of the School of Engineering and Science at New York University. His major interests are in communication systems optimization and application of the theory of stochastic processes to problems in electrical engineering. Member, Eta Kappa Nu, Sigma Xi.

ROBERT W. LUCKY, B.S.E.E. 1957, M.S.E.E. 1959, and Ph.D. 1961, all from Purdue University; Bell Telephone Laboratories, 1961—. Mr. Lucky has been concerned with various analytical problems in the transmission of digital information over voice telephone facilities. He is Head of the Data Theory Department. Member, IEEE, Tau Beta Pi, Eta Kappa Nu, Sigma Xi.

R. MAGNANI, A.B., 1961, Columbia College; B.S.E.E., 1962, M.S.E.E., 1964, both from Columbia University; Bell Telephone Laboratories, 1962—. Mr. Magnani has worked on specifying system requirements for local, toll, and military electronic switching systems. He is a member of the Local Switching Studies Department and is engaged in statistical analysis and long-range forecasting. Member, Eta Kappa Nu.

BERNARD T. MURPHY, B.Sc., 1953, Ph.D., 1959, both from the University of Leeds, England; Bell Telephone Laboratories, 1963—. Mr. Murphy worked in the field of medical physics at the University of Leeds, on electron beam studies at Mullard Research Laboratories, and since 1959 has been engaged in work on semiconductor devices. At Bell Laboratories, he has worked on both the circuit and structural aspects of semiconductor integrated circuits. He supervises work on

IMPATT diodes, high speed pulse generation, and new integrated circuit structures.

IRWIN W. SANDBERG, B.E.E., 1955, M.E.E., 1956, and D.E.E. 1958, Polytechnic Institute of Brooklyn; Bell Telephone Laboratories, 1958—. Mr. Sandberg has been concerned with analysis of military systems, synthesis and analysis of active and time-varying networks, studies of properties of nonlinear systems, and some problems in communication theory and numerical analysis. He is head of the Systems Theory Research Department. Member, IEEE, Eta Kappa Nu, Sigma Xi, Tau Beta Pi.

H. SHICHMAN, B.S., 1962, University of Michigan; M.S., 1963, University of Illinois; Bell Telephone Laboratories, 1962—. Mr. Shichman is a member of the Digital Device Integration Department where he has worked on characterization and design of logic circuits. He is working on computer analysis and synthesis of nonlinear networks. Member, IEEE, Eta Kappa Nu, Tau Beta Pi, Phi Kappa Phi.

DAVID SLEPIAN, 1941–43, University of Michigan; M.A., 1947. Ph.D., 1949, Harvard University; Bell Telephone Laboratories, 1950—. He has been engaged in mathematical research in communication theory and the theory of noise, as well as in a variety of aspects of applied mathematics. Mr. Slepian has been mathematical consultant on a number of Bell Laboratories projects. During the academic year 1958–59, he was Visiting Mackay Professor of Electrical Engineering at the University of California at Berkeley and during the Spring semester 1967 he was a Visiting Professor of Electrical Engineering at the University of Hawaii. Member, AAAS, American Math. Society, Institute of Math. Statistics, IEEE, SIAM.

NICHOLAS A. STRAKHOV, B.S.M.E., 1959, Massachusetts Institute of Technology; M.E.E., 1961, Ph.D., 1967, New York University; Bell Telephone Laboratories, 1959—. Mr. Strakhov has been designing and developing electronic test sets for transmission media maintenance. He supervises a group responsible for developing new transmission media. Member, Sigma Xi, Pi Tau Sigma, IEEE.