N69-16811
NASA CR-97365

BCL

REPORT NO. 5.8

TITLE: CRITICAL DEGENERATENESS IN LINEAR SYSTEMS

AUTHOR: Mark R. Gardner

DATE: June 15, 1968

SPONSOR: AFOSR 7-67, AF33(615)-3890, NASA NGR

BIOLOGICAL COMPUTER LABORATORY

DEPARTMENT OF ELECTRICAL ENGINEERING, UNIVERSITY OF ILLINOIS, URBANA, ILLINOIS

CRITICAL DEGENERATENESS IN LARGE LINEAR SYSTEMS

by

Mark R. Gardner
University of Illinois

BCL Report 5.8

June 15, 1968

This work was jointly sponsored by

U.S. Air Force Office of Scientific Research
AF-AFOSR 7-67

U.S. Air Force Systems Engineering Group
AF 33(615)-3890

National Aeronautics and Space Administration
Grant NGR 14-005-111

BIOLOGICAL COMPUTER LABORATORY
DEPARTMENT OF ELECTRICAL ENGINEERING
UNIVERSITY OF ILLINOIS
URBANA, ILLINOIS

ABSTRACT

A class of linear systems, composed of intrinsically stable elements, was studied. These linear systems were represented by the coefficient matrices of their differential equations. A sample space of these matrices was defined by specifying the nature of the distributions from which the matrix entries were selected.

Matrices of given size were generated by random sampling from the defined sample space. Appropriate weighting of the distributions gave control of the degenerateness, a measure of the number of zero entries. The Hurwitz criterion was used to test whether each matrix represented a stable system. The primary goal was to find the probability of stability as a function of degenerateness.

It was found, even for the relatively small matrices within the range of this study, that the degenerateness is critical. For values of degenerateness less than a particular amount (about 85%), the system is almost certainly unstable, whereas for values of degenerateness greater than this amount, the system is almost certainly stable.

ACKNOWLEDGMENT

I wish to thank Professor W. Ross Ashby for his patient guidance in this effort. His ideas and insight have been invaluable.

I am also grateful to Professor Heinz Von Foerster of the Biological Computer Laboratory, who took an interest in the project and arranged support for it.

Many thanks to the secretarial staffs of the Biological Computer Laboratory, and the Electrical Engineering Publications Office, for their skillful preparation of the manuscript.

TABLE OF CONTENTS

LIST OF FIGURES

I. THE LINEAR SYSTEM

A linear system of n variables is commonly described by a set of n first order differential equations:

$$\begin{aligned}
\dot{x}_1 &= a_{11}x_1 + a_{12}x_2 + \cdots\cdots + a_{1n}x_n \\
\dot{x}_2 &= a_{21}x_1 + a_{22}x_2 + \cdots\cdots + a_{2n}x_n \\
&\vdots \\
\dot{x}_n &= a_{n1}x_1 + a_{n2}x_2 + \cdots\cdots + a_{nn}x_n
\end{aligned}$$

For purposes of manipulation, these equations, and hence the original system, are often represented by the coefficient matrix

$$\begin{bmatrix}
a_{11} & a_{12} & a_{13} & \cdots\cdots & a_{1n} \\
a_{21} & a_{22} & a_{23} & \cdots\cdots & a_{2n} \\
\vdots & & & & \vdots \\
a_{n1} & a_{n2} & a_{n3} & \cdots\cdots & a_{nn}
\end{bmatrix}$$

This matrix completely characterizes the system, and so this paper will make little effort to differentiate between the two words, system and matrix. This will help avoid cumbersome repetition of phrases like 'the system which the matrix represents...' No confusion is expected from this usage.

In any one of the n equations

$$\dot{x}_i = a_{i1}x_1 + a_{i2}x_2 + \cdots\cdots a_{ii}x_i + \cdots\cdots + a_{in}x_n$$

the coefficient a_{ii} indicates the intrinsic nature of the system element x_i. If all the other coefficients are set equal to zero, then the equation reduces to

$$\dot{x}_i = a_{ii}x_i \ .$$

The solution of this simple equation is

$$x_i = e^{a_{ii}t} + c \qquad c \text{ is a constant}$$

so that if a_{ii} is negative, x_i will converge to c, and we will say that x_i is intrinsically stable. Constants like c, which could in fact occur in each of the original equations as well as appearing in solutions, will here all be assumed to be zero. We could deal with them, but it has been established[1] that such constants do not affect the stability of the elements or of the system as a whole.

To represent a system composed of only intrinsically stable elements, then, it is necessary and sufficient that the coefficients a_{11}, a_{22}, ..., a_{nn} all be negative. These are, of course, the principle diagonal entries in the matrix.

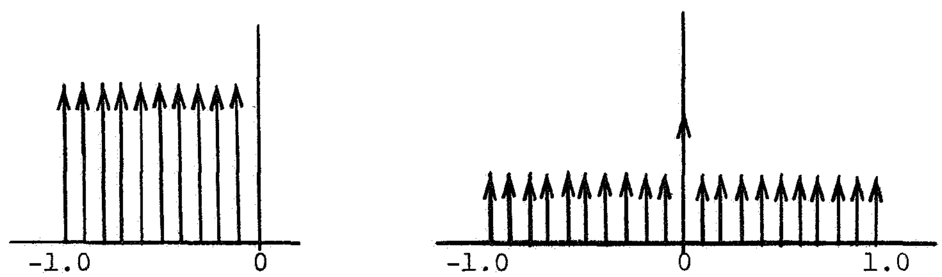
The off-diagonal entries represent interactions, or connections, so to speak, between the system elements. These can be positive, negative, or zero. If an entry a_{ij} is zero, then there is no direct effect on x_i by x_j, and we will say that the system is partially degenerate.

Delimiting a Class of Systems

To be practical, it is necessary to select some finite class of linear systems to investigate. In this study, the selection is done by specifying the distributions from which the matrix entries are derived.

We choose the diagonal entries, a_{ii}, to be negative, in the range -1.0 to -0.1 in increments of 0.1. The selection is made equiprobably from this range. This distribution is shown in Figure 1a.

The off-diagonal elements are selected from the range -1.0 to +1.0, in increments of 0.1. The selection is made equiprobably from this range, except that, in order to achieve various degrees of degenerateness in the matrices, the weight for the entry zero is varied. This distribution is shown in Figure 1b.



a. diagonal

b. off-diagonal

Figure 1. The distribution of matrix entries.

Since the diagonal entries cannot be zero, we define degenerateness as the percent of off-diagonal entries which are zero. The number of off-diagonal entries in a matrix of size n x n is n^2-n, or $n(n-1)$. If m of these are zero, then the degenerateness D is defined as

$$D = \frac{m}{n(n-1)} \cdot 100\%$$

The term degenerateness is used in this paper in preference to the term degeneracy, the latter being commonly used to describe the relation between the rank and the size of a matrix. That is quite another subject, and so the word is not used here.

The researcher recognizes that the class of linear systems chosen for this study might seem very restricted. However, he feels that the results are widely applicable. By normalization techniques, any linear system of intrinsically stable elements can be approximated by a system in the chosen class. For example, consider the system

$$\dot{x}_1 = -24x_1 + 7x_2 - 3x_3$$

$$\dot{x}_2 = 7x_1 - 18x_2 - 10x_3$$

$$\dot{x}_3 = -3x_1 + 10x_2 - 12x_3.$$

The system matrix is

$$\begin{bmatrix} -24 & 8 & -3 \\ 8 & -18 & 10 \\ -3 & 10 & -12 \end{bmatrix}$$

which is equal to

$$24 \begin{bmatrix} -1 & .333 & -.125 \\ .333 & -.75 & .417 \\ -.125 & .417 & -.5 \end{bmatrix}$$

The roots of the characteristic equation are the same, except for the scaling factor, for both matrices. Hence, for purposes of stability, testing the second matrix is as good as testing the first. It is only necessary now to round entries to the nearest tenth to obtain a system from the chosen class:

$$\begin{bmatrix} -1.0 & .3 & -.1 \\ .3 & -.8 & .4 \\ -.1 & .4 & -.5 \end{bmatrix}$$

This approximation technique will then allow the extension of the results of this paper to linear systems in general, in spite of the fact that only a limited class of systems was studied.

II. FEEDBACK

One usually thinks of feedback as the effect of a variable upon itself. The entry a_{ii} characterizes the intrinsic feedback of element x_i. If a_{ii} is negative, this feedback results in x_i being intrinsically stable, as has already been noted. Further feedback, possibly creating instability, is possible by interactions with other elements, i.e., by external feedback. The simplest case is a two-step loop, for example:

$$\left[\begin{array}{ccccc} a_{11} & \cdots & \cdots & \cdots & \cdot \\ \vdots & a_{22} & \longleftarrow & a_{2i} & \vdots \\ \vdots & \downarrow & & \uparrow & \vdots \\ \vdots & a_{i2} & \longrightarrow & a_{ii} & \vdots \\ \vdots & & & & \vdots \\ \cdot & \cdots & \cdots & \cdots & a_{nn} \end{array}\right]$$

It is meaningless to discuss the direction of feedback, or even which element (for example, a_{22} or a_{ii}, above) is receiving the feedback, except by convention. These conventions depend on the physical system, have no bearing on the matrix or mathematics, and so will not be discussed here.

In a system of intrinsically stable elements, only external feedback can cause system instability. In the matrix, external feedback can occur only if there are non-zero entries on both sides of the principal diagonal, and further, they must form some sort of a loop. Consider the following examples.

Here is a third order system with non-zero entries only above the diagonal:

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ 0 & a_{22} & a_{23} \\ 0 & 0 & a_{33} \end{bmatrix} \qquad a_{11},\ a_{22},\ a_{33} < 0$$

The variable x_3 is affected only by itself, and hence converges to zero. As x_3 becomes smaller, its effects on x_1 and x_2 through a_{13} and a_{23} become smaller, approaching zero. Then x_2 converges to zero, being affected only by its intrinsic feedback. And so on, so that the entire system must be stable regardless of the off-diagonal entries.

Here are two third order systems with diagonal entries on both sides of the diagonal. The arrows indicate the "route" of interactions, and hence must alternate in row and column.

$$\begin{bmatrix} a_{11} & \leftarrow a_{12} & 0 \\ 0 & a_{22} \leftarrow & a_{23} \\ a_{31} & 0 & a_{33} \end{bmatrix} \qquad\qquad \begin{bmatrix} a_{11} & a_{12} & 0 \\ 0 & a_{22} & 0 \\ a_{31} & 0 & a_{33} \end{bmatrix}$$

In the first system, a feedback loop is formed, as shown by the arrows. This system, then, could be unstable. In the second system, no feedback loop is present, and the system must be stable.

A plausible suggestion is that the greater the number of loops, the greater is the chance that the system will be unstable. Therefore, increasing the number of zero entries in the matrix, thereby decreasing the likelihood of a loop, would be expected to make the system more likely to be stable. We would expect this relation to be monotonic, up to the point where all the off-diagonal elements are zero, at which point the system is certainly stable.

III. STABILITY OF THE SYSTEM

There have been worked out several agreeable devices to determine whether a linear system is stable. These are based primarily on the criterion that for all time, every system element will converge to some value, or will at least be bounded. This is equivalent to requiring that none of the latent roots of the system have positive real parts.

The latent roots of the system are the roots of the characteristic equation of the matrix. The characteristic equation is found by evaluating the determinant $|A - \lambda I|$, where A is the system matrix, and I is an identity matrix of the same size. The result is set equal to zero.

The resulting equation, the characteristic equation, is of the form

$$\lambda^n + m_1\lambda^{n-1} + m_2\lambda^{n-2} + \ldots + m_i\lambda^{n-i} + \ldots + m_n = 0.$$

The coefficients $m_1, m_2, \ldots, m_n$ are obtained from the evaluation of $|A - \lambda I|$, or by an alternative procedure. This alternative procedure seems to be mechanically simpler, and is used in this work: any particular coefficient m_i is the sum of all i-rowed principal sub-determinants of A, multiplied by $(-1)^i$. An example from Ashby[1] will illustrate this procedure.

Consider the third order system

$$\begin{bmatrix} -5 & 4 & -6 \\ 7 & -6 & 8 \\ -2 & 4 & -4 \end{bmatrix}$$

For this system,

$$m_1 = [(-5) + (-6) + (-4)] \quad (-1)^1 = 15$$

$$m_2 = \left[\begin{vmatrix} -5 & 4 \\ 7 & -6 \end{vmatrix} + \begin{vmatrix} -5 & -6 \\ -2 & -4 \end{vmatrix} + \begin{vmatrix} -6 & -8 \\ 4 & -4 \end{vmatrix} \right] (-1)^2 = 2$$

$$m_3 = \begin{vmatrix} -5 & 4 & -6 \\ 7 & -6 & 8 \\ -2 & 4 & -4 \end{vmatrix} \quad (-1)^3 = 8$$

and the characteristic equation is

$$\lambda^3 + 15\lambda^2 + 2\lambda + 8 = 0 \ .$$

There are several ways to find whether any of the roots of the characteristic equation have positive real parts. One way, of course, is to solve the equation. This, however, is difficult, especially since no analytical solution exists for n greater than four. Among alternate techniques described in the literature is Hurwitz' criterion.[1,3] This method requires, for the system in question to be stable, that in the sequence of determinants,

$$\begin{vmatrix} m_1 \end{vmatrix} \quad \begin{vmatrix} m_1 & 1 \\ m_3 & m_2 \end{vmatrix} \quad \begin{vmatrix} m_1 & 1 & 0 \\ m_3 & m_2 & m_1 \\ m_5 & m_4 & m_3 \end{vmatrix} \quad \ldots\ldots ,$$

if $q > n$

then $m_q = 0$

all are positive. This is the method of determining stability which was selected for this study.

Applying this criterion to the above example gives

$$\begin{vmatrix} m_1 \end{vmatrix} = 15 ,$$

$$\begin{vmatrix} m_1 & 1 \\ m_3 & m_2 \end{vmatrix} = \begin{vmatrix} 15 & 1 \\ 8 & 2 \end{vmatrix} = 22 ,$$

$$\begin{vmatrix} m_1 & 1 & 0 \\ m_3 & m_2 & m_1 \\ 0 & 0 & m_3 \end{vmatrix} = \begin{vmatrix} 15 & 1 & 0 \\ 8 & 2 & 15 \\ 0 & 0 & 8 \end{vmatrix} = 176 .$$

All three determinants are positive, and hence the system must be stable.

IV. THE RESULTS OF RANDOM SAMPLING

In order to determine the effect of degenerateness on the probability of stability for systems of even moderately high order, it is necessary to abandon any hopes for a complete solution, and turn to random sampling as the only possible avenue of exploration. That complete solution is nearly impossible is made evident by the observation that there are over 10^{100} tenth-order systems choosable from even the limited distributions of this study, and it takes the IBM 7094 computer about 1.25 minutes to check each one--the task would take 2.6×10^{94} years.[2]

The number of manipulations, and hence the time for solution, increase roughly with n!, thereby greatly limiting the number of matrices of higher order that can be checked. We turn, then to random sampling. Although a few of 10^{100} would not appear to give a very good sample, results justify the approach.

Programming for this approach was difficult only in that the program needed nearly complete generality. The method is straightforward, however. Details of the program can be found in the appendix.

Data were taken over a wide range of degenerateness for the lower order cases. Since time threatened to become excessive, data were taken over only a limited range of degenerateness for the higher order cases. In fact, for the 10 x 10 system the program was modified to check only systems within a specified range of degenerateness.

The remarkable result is the way degenerateness influences the probability of stability. For the smaller systems, to about n=6, the probability of stability is rather smoothly monotonic increasing with degenerateness. But for the systems of orders n=7 to n=10, degenerateness becomes critical, so that the transition from low probability to high probability occurs within a narrow range of degenerateness.

These facts are made evident in Figures 2a through 2h, where the computer results have been plotted, and a line sketched to indicate the trend of each graph. Although one might question the exact position of the graph lines through the data, the trend to step function form is indisputable. Figure 2i collects the suggested graphs so that the trend becomes evident. This trend is pronounced and regular, so that as order increases, so does the graph more closely approximate a step function.

For n=10, this step function form is pronounced, so that for values of degenerateness less than about 85%, the probability is virtually zero, whereas for greater values, the probability is very high, nearly 100%.

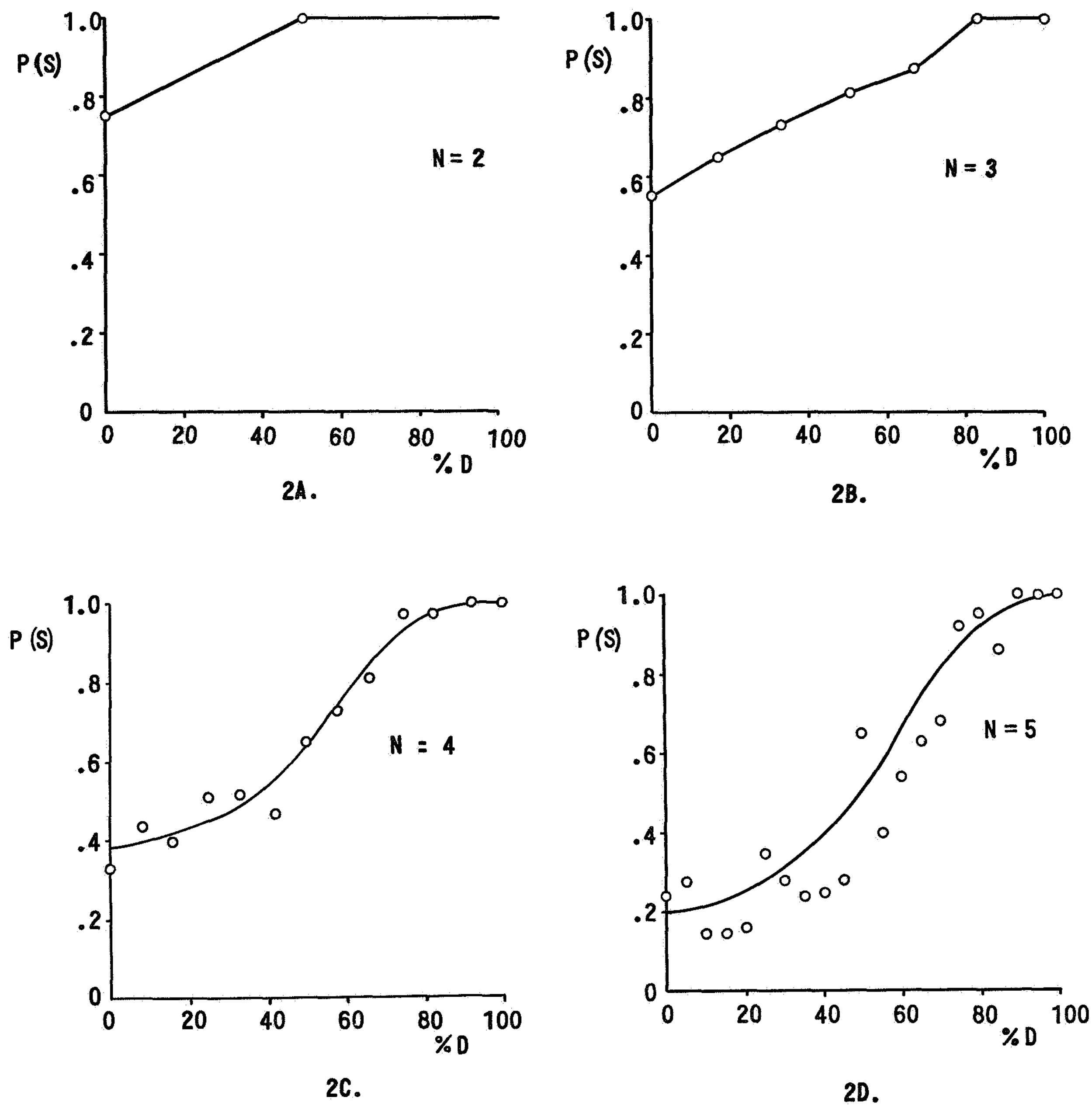


FIGURE 2.

PROBABILITY OF STABILITY VERSUS DEGENERATENESS FOR RANDOM LINEAR SYSTEMS, ORDERS TWO THROUGH TEN.

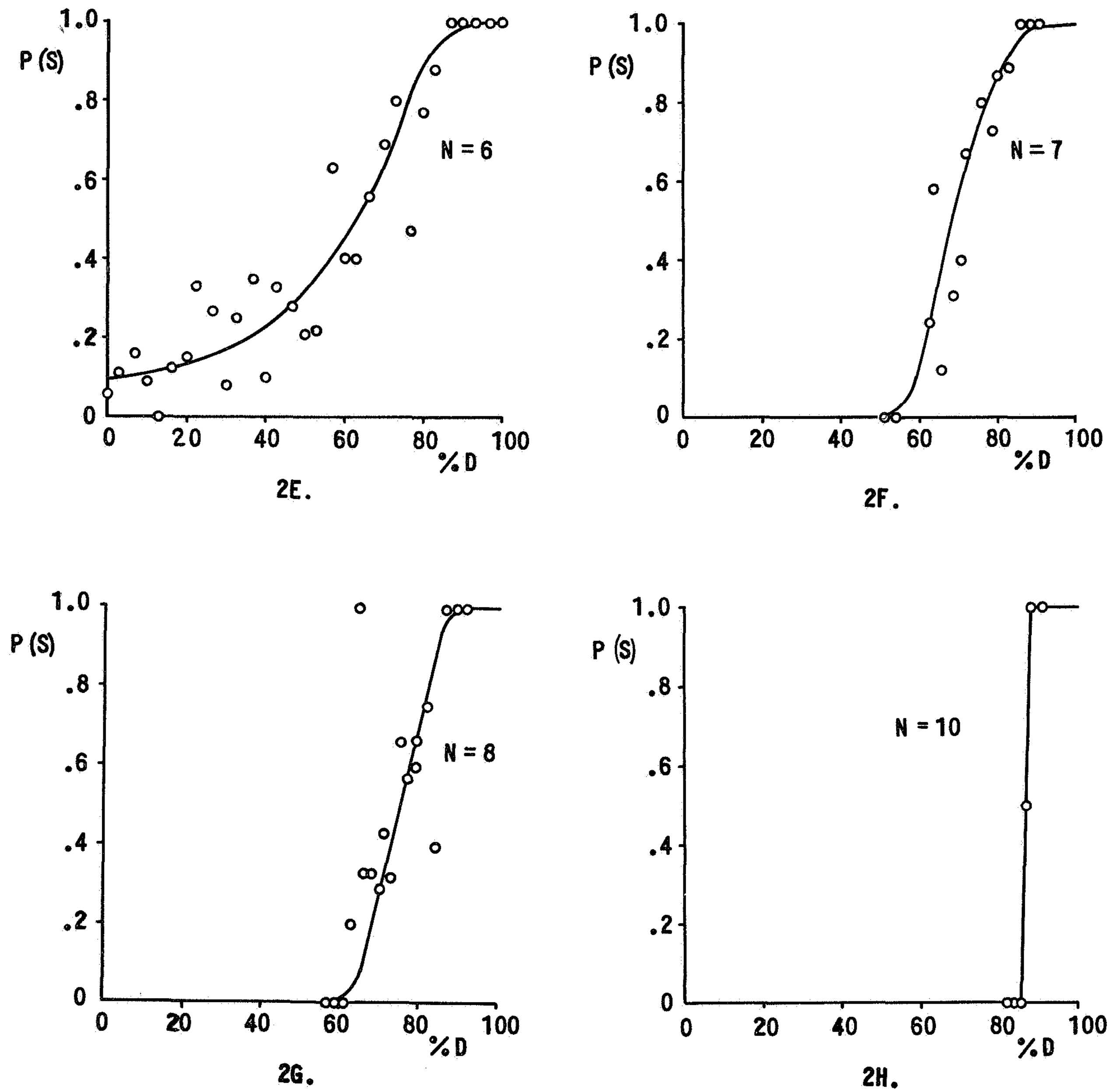
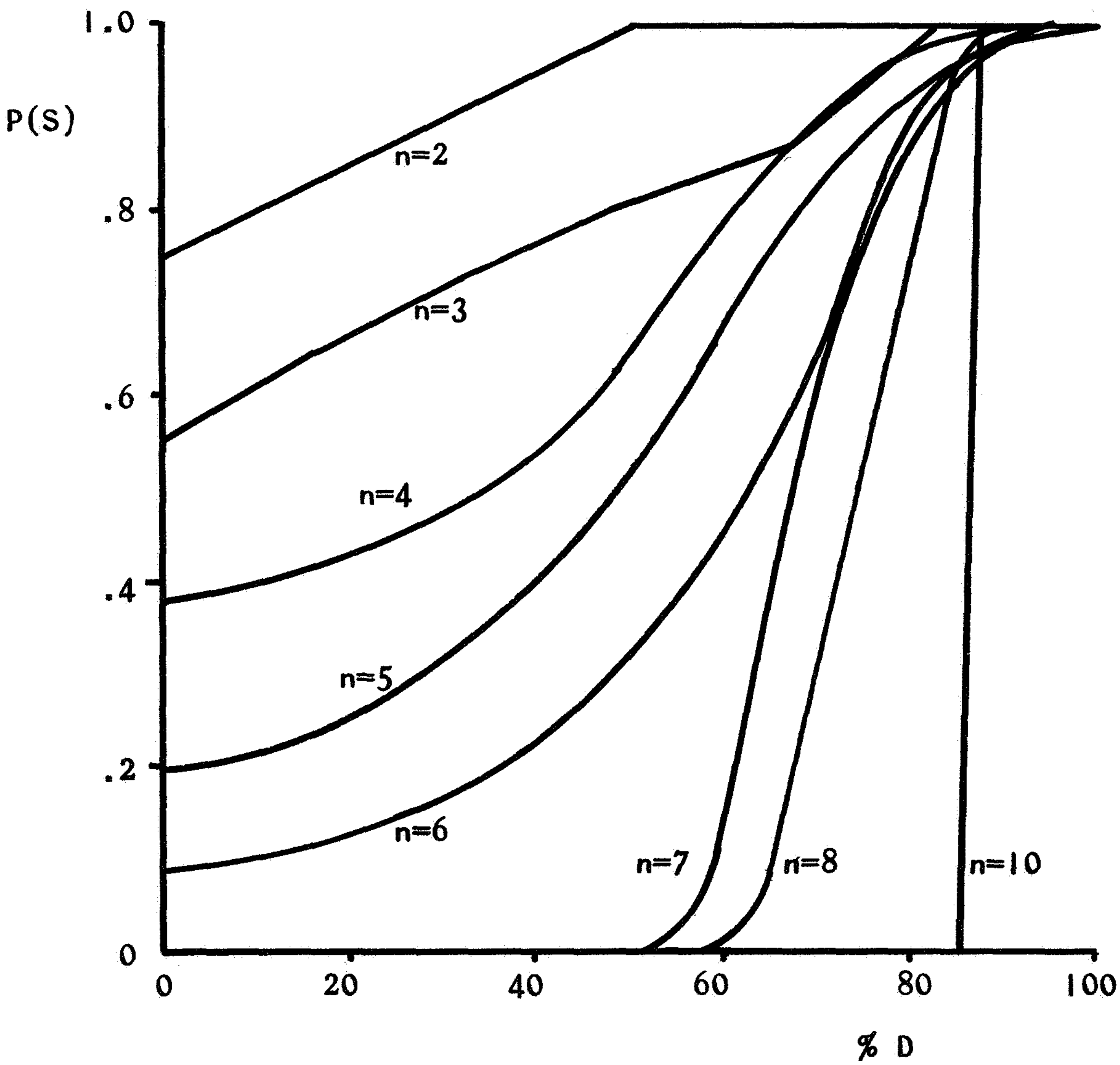


FIGURE 2.

PROBABILITY OF STABILITY VERSUS DEGENERATENESS
FOR RANDOM LINEAR SYSTEMS, ORDERS TWO THROUGH TEN.



2I.

Figure 2. Probability of Stability versus Degenerateness for random linear systems, orders two through ten.

V. POSSIBLE EXTENSION OF RESULTS

Another presentation of the data will show the regularity of the trend, and perhaps give a clue for extension to higher order cases. Consider the value of degenerateness at which a system achieves a probability of stability of 75%. The relation between this value and order is revealing. Table 1 and Figure 3 present the relation.

n	D %
1	0
2	12
3	37
4	58
5	65
6	74
7	77
8	80
10	83

Table 1.

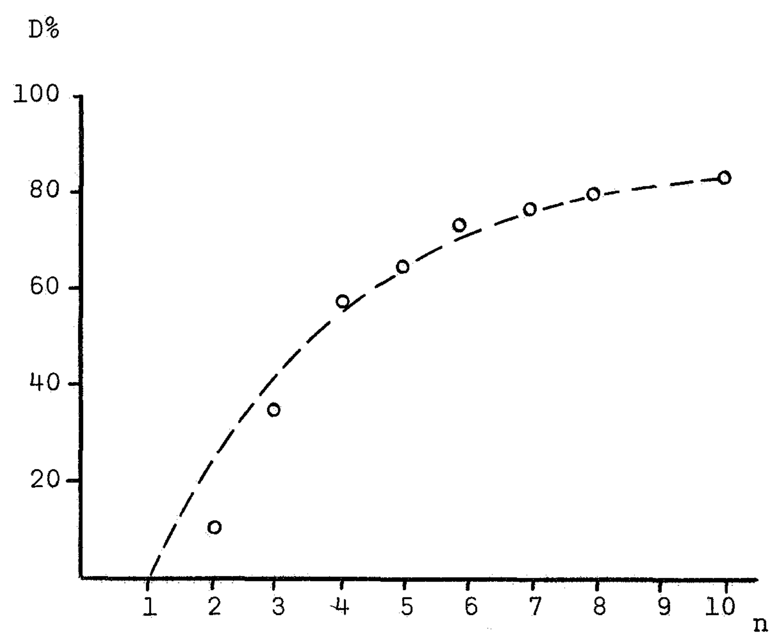


Figure 3. Degenerateness for 75% probability of stability versus system order.

The selection of 75% is arbitrary, and any other value would have served as well. What is important is the regularity, and the trend to step function form. For purposes of prediction, Figure 3 could very well be approximated by an exponential curve. Roughly,

$$D_{75\%} = 100\ (1-e^{-.254(n-1)}) \quad n = 1, 2, 3, \ldots$$

However, though we might expect this kind of relation, it gives a clearly incorrect result. The following table gives, for some larger orders, the number of non-zero entries allowable in a matrix for it to have the degenerateness required by the above approximate equation.

n	10	11	12	13	20
# non-zeros	12	11	10	10	5

Table 2

This does not agree with intuitive ideas about loops and feedback within the matrix. It would be more plausible that the curve of Figure 3 actually become nearly level for larger systems, at some constant value. If the asymptotic value were 90%, then Table 2 would look a little different, as in the following Table.

n	10	11	12	13	20
# non-zeros	12	12	13	16	38

Table 3

This is much more agreeable with the idea that, with larger matrices, there are more ways to slip in entries without forming loops.

VI. ANOTHER APPROACH

Although the main body of the work must be done by random sampling, some insight can be gained by some further investigation. A study of the characteristics of the Hurwitz criterion, and complete solution for systems of order two and three, will provide an understanding that the author feels essential to further progress in the right direction.

Starting with the distributions of the matrix entries, it is possible to arrive at distributions of their sums and products. This allows an overview of all that is significant, and does not depend on random sampling.

The distributions of the matrix entries were given in Figure 1, and are repeated here, approximated by straight line segments. This practice will be continued, though it must be remembered that all the distributions are discrete.

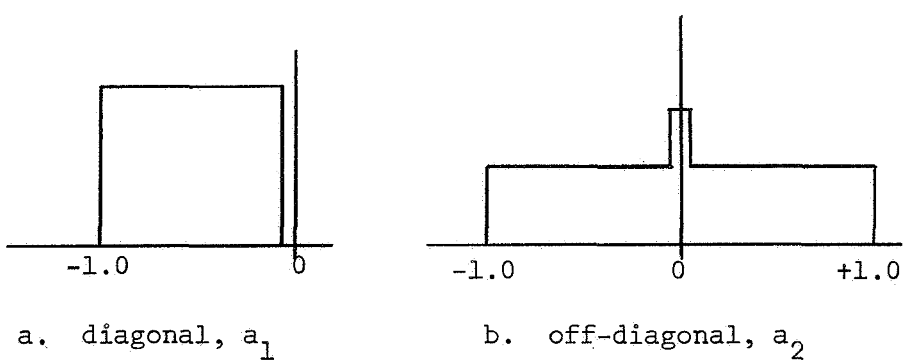


Figure 4. Entry Distributions, approximated by continuous line segments.

Now let the symbol a_1 represent any diagonal entry, since they are indistinguishable, being represented by a single distribution. Similarly, let a_2 represent any off-diagonal entry. Conventional arithmetic notation will be retained, but with a slightly different interpretation. For example,

$$k = a_1 + a_1$$

will mean 'k is the distribution of the sum of two numbers, the first chosen from distribution a_1 and the second chosen from distribution a_1.' Figure 5 gives a_1 and $k = a_1 + a_1$. A more lengthy discussion of forming sums and products of distributions is given in the appendix.

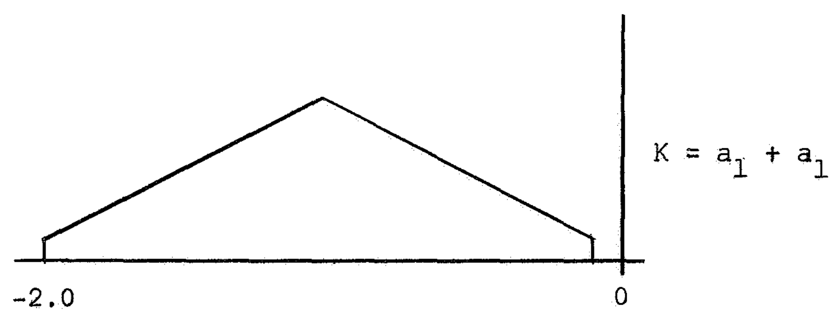


Figure 5. Distribution of a Sum.

To see the application of this sort of idea, let us investigate a simple second order system:

$$\begin{bmatrix} a_1 & a_2 \\ a_2 & a_1 \end{bmatrix}$$

Since the entries are distributions, rather than specific numbers, the notation represents all possible systems which can be chosen from the specified distributions. Applying the Hurwitz criterion, a sequence of two tests in this case, we get

$$m_1 = -(a_1 + a_1), \text{ and } m_2 = a_1 . a_1 - a_2 . a_2$$

These distributions are shown in Figure 6.

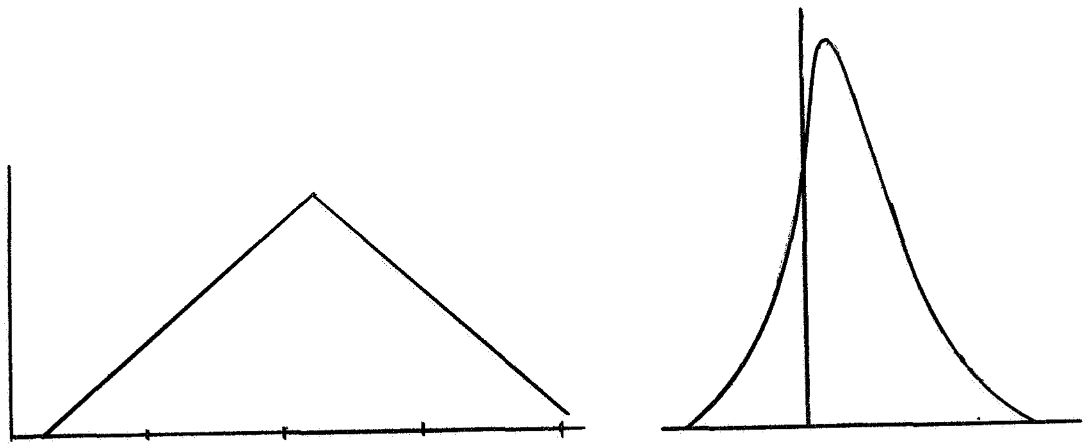


Figure 6. Distributions m_1 and m_2 for a second order system.

Of course, a_2 has a variable weight for zero, and hence any distribution dependent on a_2 will depend then on the weight of zero in a_2. The distribution shown in Figure 6 is for a_2 weighted uniformly, including zero.

It is convenient here to introduce a bit of notation. The Hurwitz criterion contains n tests for an n x n system. These will be labeled $T_1^n, T_2^n, \ldots\ldots, T_n^n$.

For the second order system in the present work,

$$T_1^2 = m_1 \quad .$$

The criterion states that if T_1^2 is positive then the system has passed this test. From Figure 6, we see that the entire distribution m_1 is positive, and hence every second order system chosen from the given distributions must pass the first test of the Hurwitz criterion. This, of course, could have been noted by inspection, but the author believes there is value in pursuing this approach, for the understanding of the method it will provide.

The second test of the criterion is

$$T_2^2 = \begin{vmatrix} m_1 & 1 \\ 0 & m_2 \end{vmatrix} = m_1 \cdot m_2$$

This distribution is given in Figure 7.

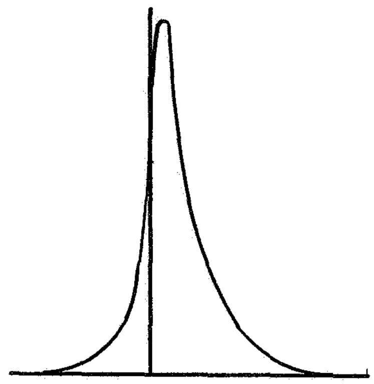


Figure 7. T_2^2 for a second order system.

Again the criterion states that T_2^2 be positive for the system to pass this test. In this case, a portion of the distribution is negative, representing those second order systems that are rejected at this test. The ratio of the positive area to the total area is the probability that a second order system will be stable. In this case, the ratio is .75. We have then determined that the probability of stability of a second order system weighted uniformly for zeros is 75%.

This procedure was repeated with a weight for zero of 25% in a_2, giving a new T_2^2. This is given in Figure 8.

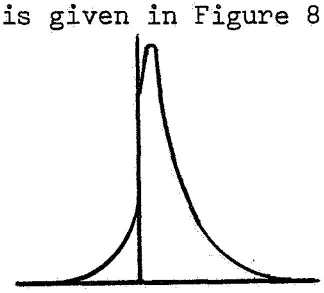


Figure 8. T_2^2 for a second order system, weighted for 25% zero entries.

There is a definite shift to the positive side of the distribution, indicating a higher probability of stability. For this distribution, the probability of stability is about 85%. We now draw an approximate graph of probability of stability versus percent zeros, as in Figure 9. Indeed, this is crude, with only two points--but it is sufficient to indicate the trend and the method. Compare this with Figure 2a.

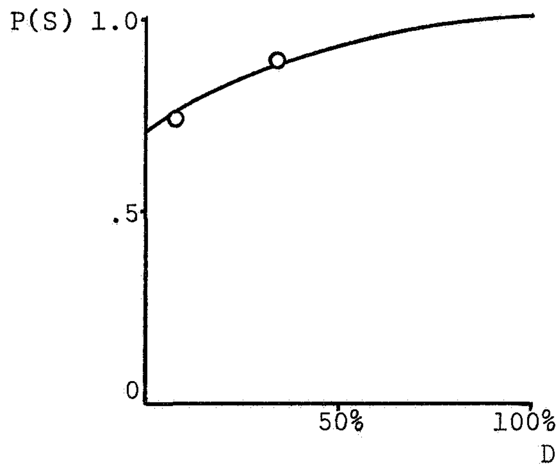


Figure 9. Probability of stability versus percent zeros for a second order system.

For a third order system, we have

$$\begin{bmatrix} a_1 & a_2 & a_2 \\ a_2 & a_1 & a_2 \\ a_2 & a_2 & a_1 \end{bmatrix} \qquad \begin{aligned} m_1 &= -(a_1 + a_1 + a_1) \\ m_2 &= 3.(a_1 \cdot a_1 - a_2 \cdot a_2) \\ m_3 &= -(a_1 \cdot a_1 \cdot a_1 + 2 \cdot a_2 \cdot a_2 \cdot a_2 - 3 \cdot a_1 \cdot a_2 \cdot a_2) \end{aligned}$$

These distributions, for various weights of zeros, are given in Figure 10. The sequence of tests is

$$\begin{vmatrix} m_1 & 1 & 0 \\ m_3 & m_2 & m_1 \\ 0 & 0 & m_3 \end{vmatrix} \qquad \begin{aligned} T_1^3 &= m_1 \\ T_2^3 &= m_1 m_2 - m_3 \\ T_3^3 &= m_1 m_2 m_3 - m_3 m_3 \end{aligned}$$

All matrices, of course, pass T_1^3. The distributions of T_2^3 for the various weights for zero are given in Figure 11.

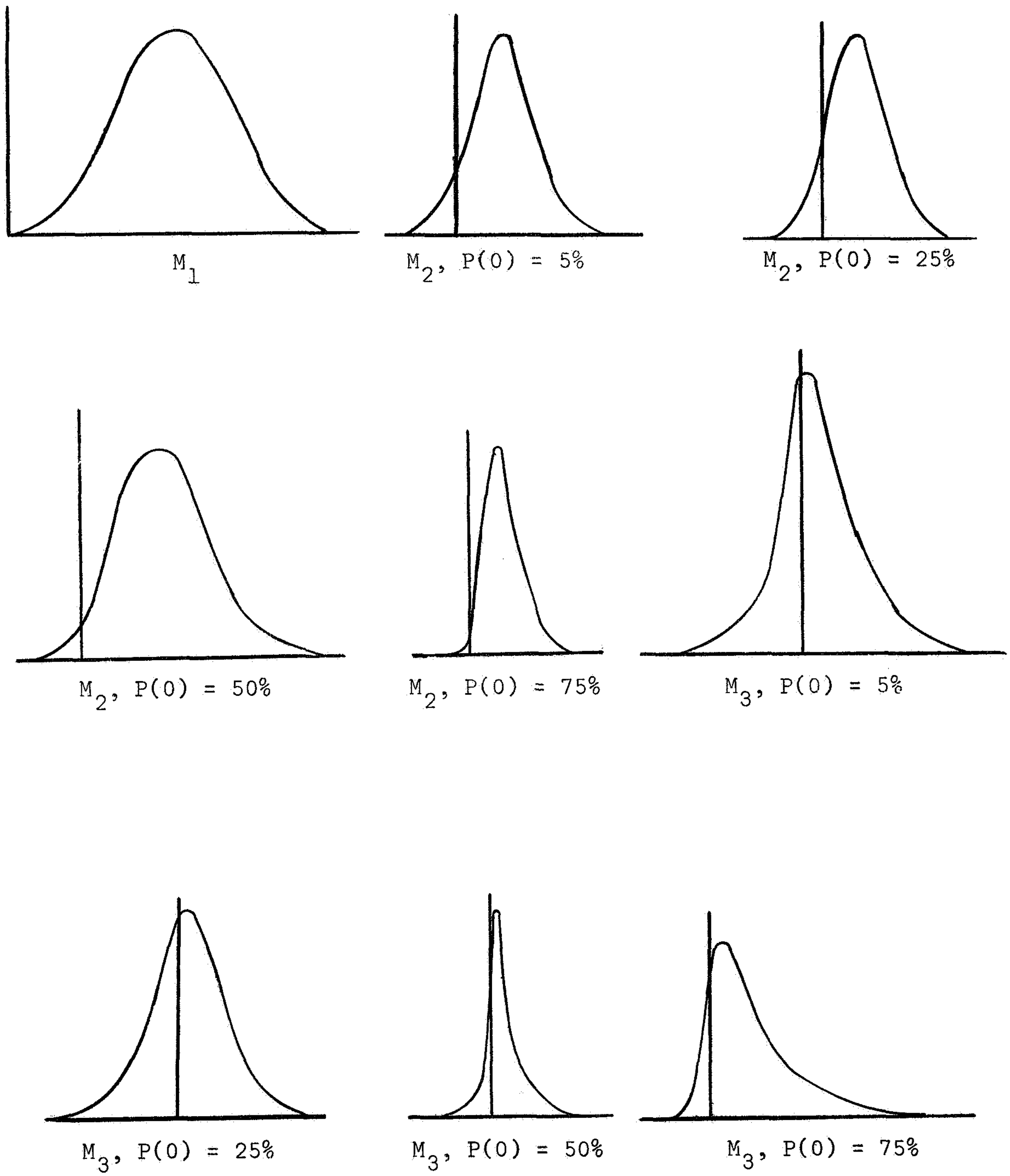


Figure 10. m_1, m_2, and m_3 for various weights for zero.

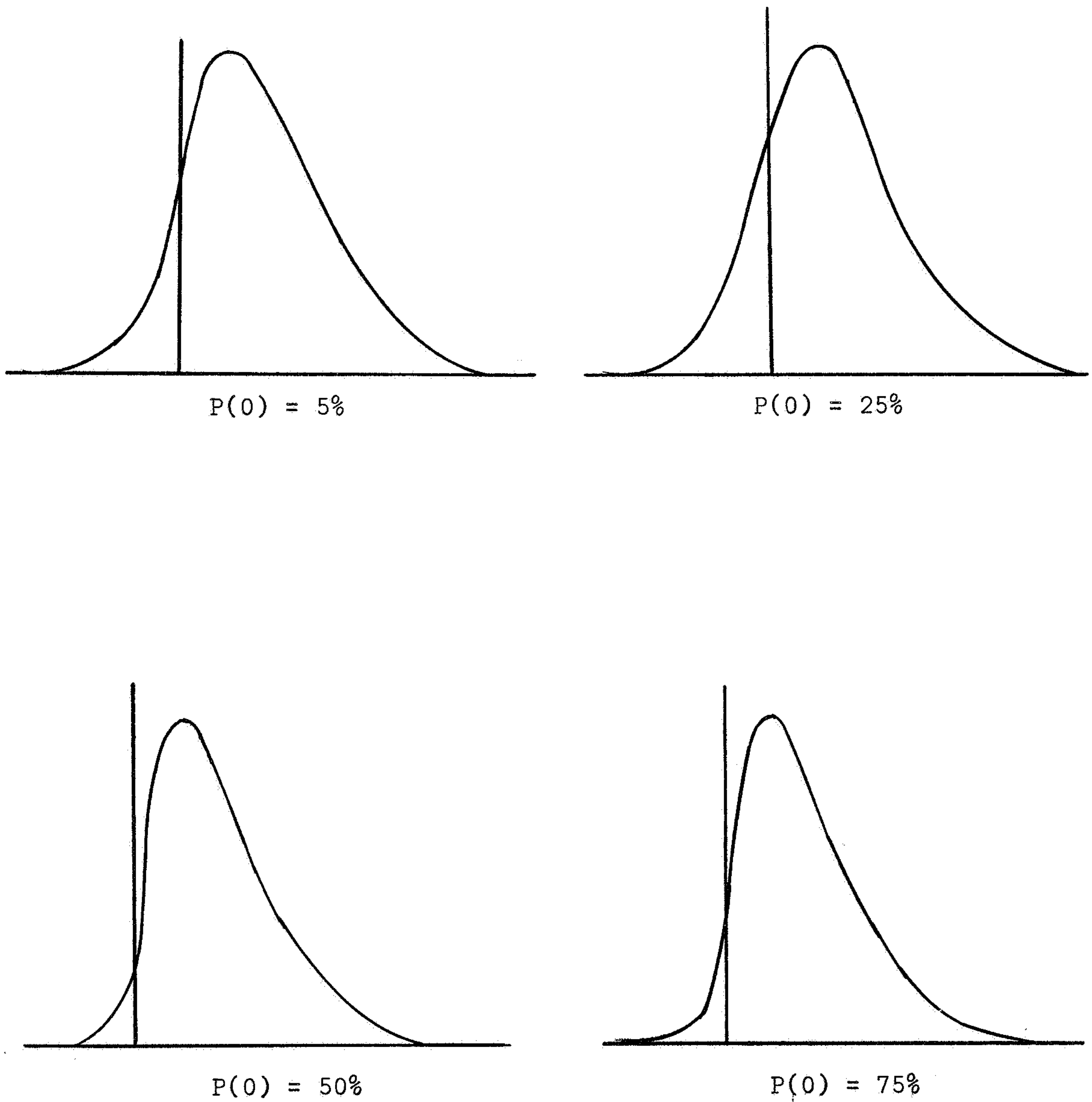


Figure 11. T_2^3 for various weights for zero.

In each case, there is a percentage that is rejected at this step. This percentage decreases as the probability of zero increases. The following table gives the results of mechanical integration using a planimeter, of these distributions.

Weight for zeros	Percent rejected at test T_2^3
5%	20
25	13
50	7
75	6

Table 4

The results in Table 4 compare reasonably well with the results obtained from the random sampling approach. The graphs are presented here for comparison.

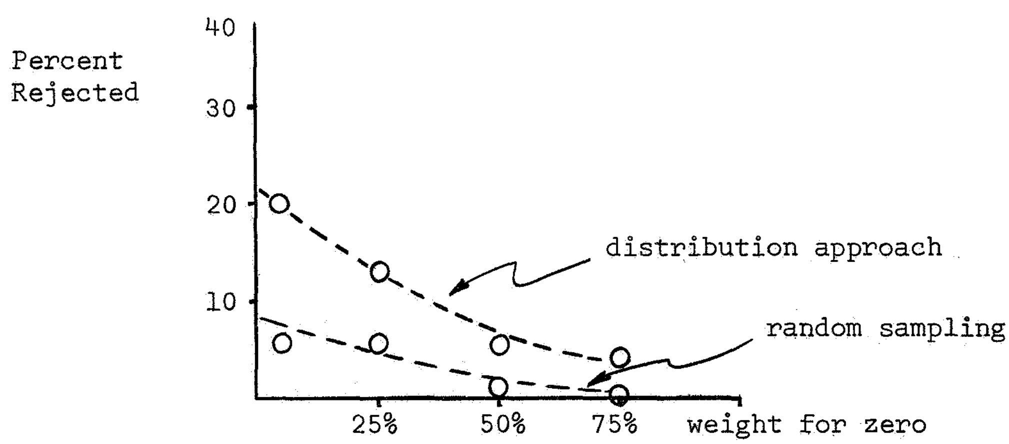


Figure 12. Percent rejected at test T_2^3 versus the weight for zeros.

Once a system has been rejected, further tests are unnecessary. Therefore, the expression for T_3^3 is modified in order that we can eliminate from it the systems that were rejected at test T_2^3.

$$T_3^3 = m_1m_2m_3 - m_3m_3$$

$$\text{but} \qquad T_2^3 = m_1m_2 - m_3$$

$$\text{so} \qquad T_3^3 = T_2^3m_3 \quad .$$

Now we can 'erase' the negative portion of T_2^3 before forming T_3^3, arriving at a distribution which does not include systems previously rejected. Our notation for an 'erased' distribution will be a superscript E. Figure 13 shows the process clearly.

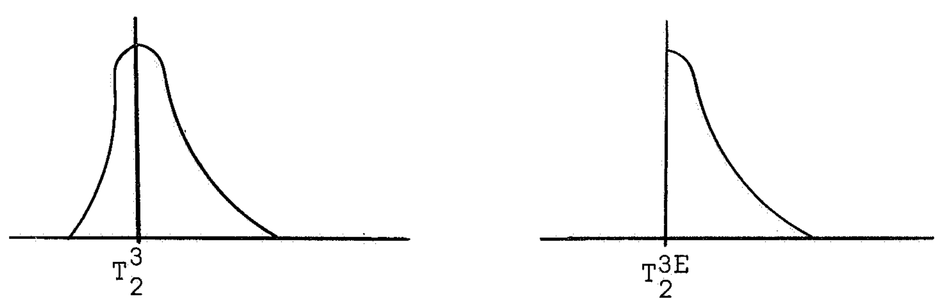


Figure 13. T_2^3, and T_2^{3E} .

Therefore, a more meaningful expression for the third test is

$$T_3^3 = T_2^{3E}m_3$$

and we see that whether our third order system passes its final check depends only on the value of m_3, which is the system determinant. This is because T_2^{3E} contains no negative portion, and hence a portion of T_3^3 can be negative only if a portion of m_3 is negative. The distributions T_3^3 derived from the modified expression are given in Figure 14.

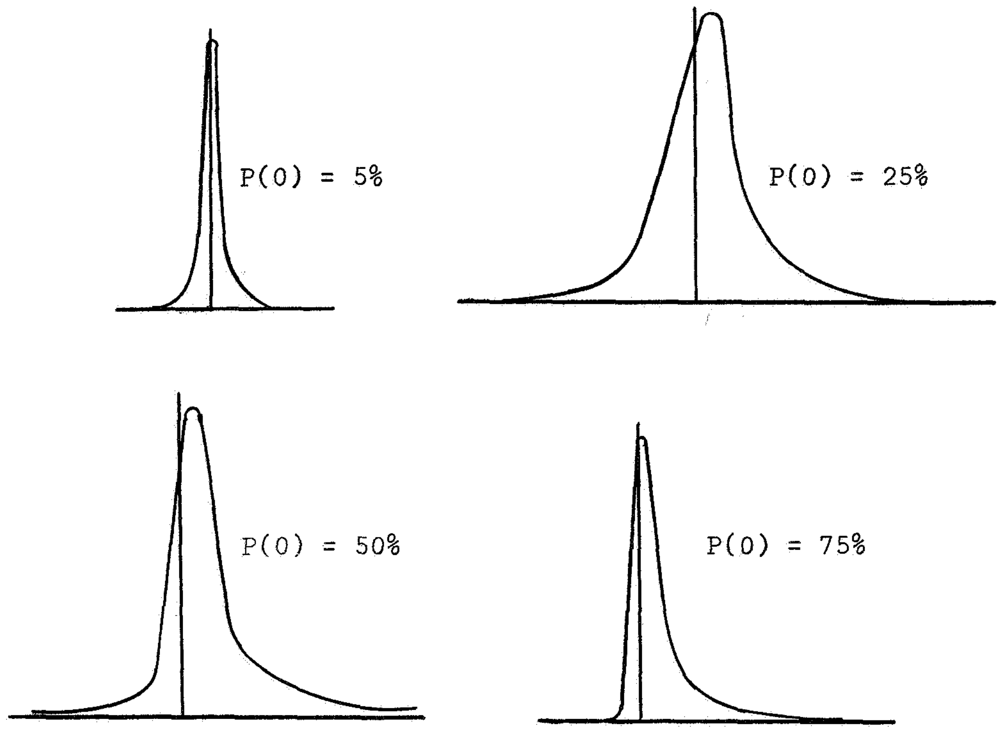


Figure 14. T_3^3 for various weights for zero entries.

The negative portions represent the rejected systems, and again, the ratio of areas gives the percent rejected at check T_3^3. The results of mechanical integration are presented in the following table.

Weight for Zeros	% Rejected at Check T_3^3
5%	39
25	34
50	22
75	13

Table 5

For T_3^3, as for T_2^3 previously, the tabulated results compare well with those obtained by random sampling. These are presented in Figure 15. It is suggested that the differences between the results are due to excessive approximations in forming the distributions in the present approach.

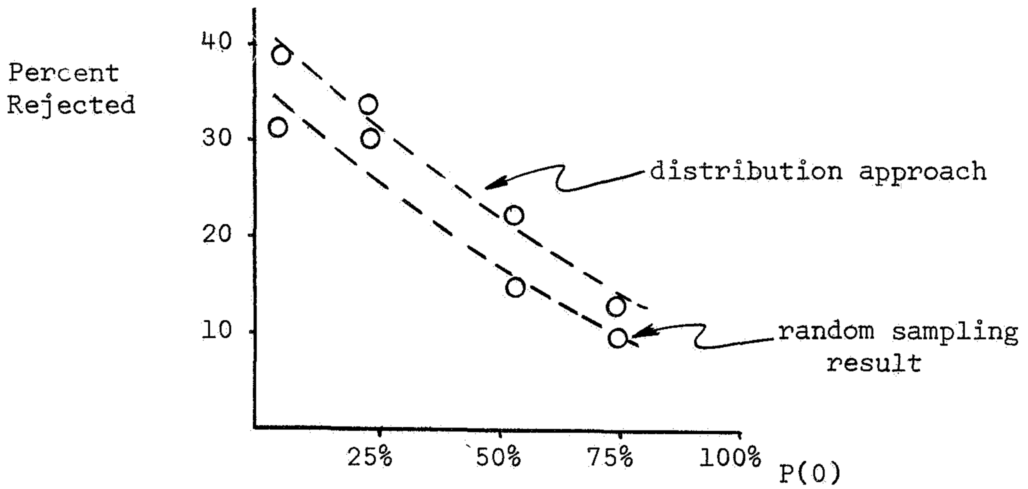


Figure 15. Percent rejected at step T_3^3 versus the weight for zeros.

Combining the results of Table 4 and Table 5, we can arrive at a relation between overall probability of stability and the weighting for zero entries. This is given in Table 6 and Figure 16. Compare this with Figure 2b.

Weight for Zeros	% Stable
5%	49
25	58
50	73
75	82

Table 6.

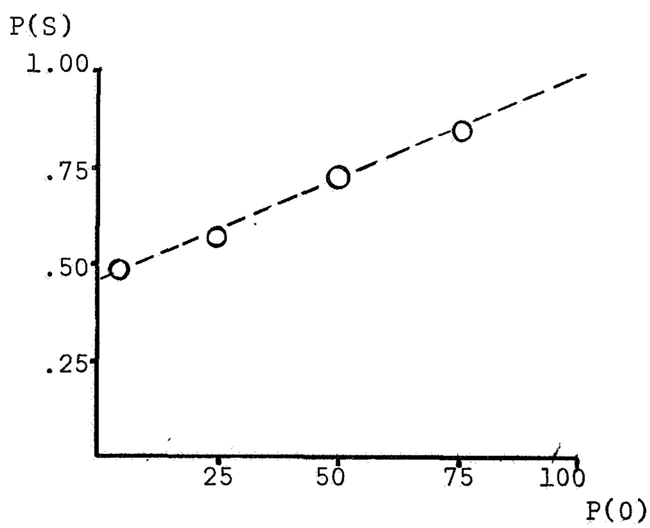


Figure 16. Percent stable versus weight for zeros.

We have thus exhausted, except for detail, the second order and third order cases. Computer time in forming the sum and product of distributions is not excessive, but the number of operations necessary to check each system increases roughly as the factorial of the system size, and hence this approach becomes unduly time consuming for larger systems.

The distribution approach amounts to simultaneous checking of every possible system in the chosen class. Random sampling techniques then have an advantage since they require checking only a very few of the total number of systems.

A last note is in order before the distribution approach is finished. For an arbitrary size system, a Hurwitz criterion matrix of the following form is generated:

$$\begin{bmatrix} m_1 & 1 & 0 & 0 & 0 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ m_3 & m_2 & m_1 & 1 & 0 & 0 & \cdot & \cdot & \cdot & \cdot & \cdot & 0 \\ \cdot & & \cdot & & & & & & & & & \cdot \\ \cdot & & & \cdot & & & & & & & & \\ \cdot & & & & & & & & & & & \cdot \\ \cdot & & & & & \cdot & & & & & & \cdot \\ \cdot & & & & & & \cdot & & & & & \cdot \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & m_n & m_{n-1} & & m_{n-2} \\ 0 & 0 & 0 & 0 & \cdot & \cdot & \cdot & \cdot & 0 & 0 & & m_n \end{bmatrix}$$

The next to last test T^n_{n-1} is the determinant of the submatrix up to but not including the final row or column. The last check is then

$$T^n_n = T^{nE}_{n-1}\, m_n$$

and is clearly dependent only on m_n, which is the system determinant multiplied by $(-1)^n$. No doubt a great deal of time could be saved by checking m_n first, which is much simpler than going through the entire Hurwitz check. If m_n is negative, the system is unstable.

VII. DIRECTION FOR FUTURE WORK

This study has revealed that for a limited class of linear systems, degenerateness is critical. For larger systems, it is very critical, so that the graph of probability of stability versus degenerateness becomes nearly a step function.

This result is important and applicable. But there is a need for continued investigation of this type with some different, specific system types. Probably of most importance would be a study of large systems composed of a number of smaller, individually stable subsystems.

In fact, it would be interesting to see if most of the larger systems studied in this paper might not be decomposable into smaller stable subsystems. No such activity was attempted here.

A second area of study might be of systems composed of intrinsically unstable elements, or with elements selected from essentially different distributions.

The author hopes the present work will be of assistance in these future projects.

APPENDIX A

PROGRAMMING FOR THE RANDOM SAMPLING APPROACH

The flow charts for the random sampling program are presented subsequent to this brief introduction. As they are nearly self-explanatory, little else will be said.

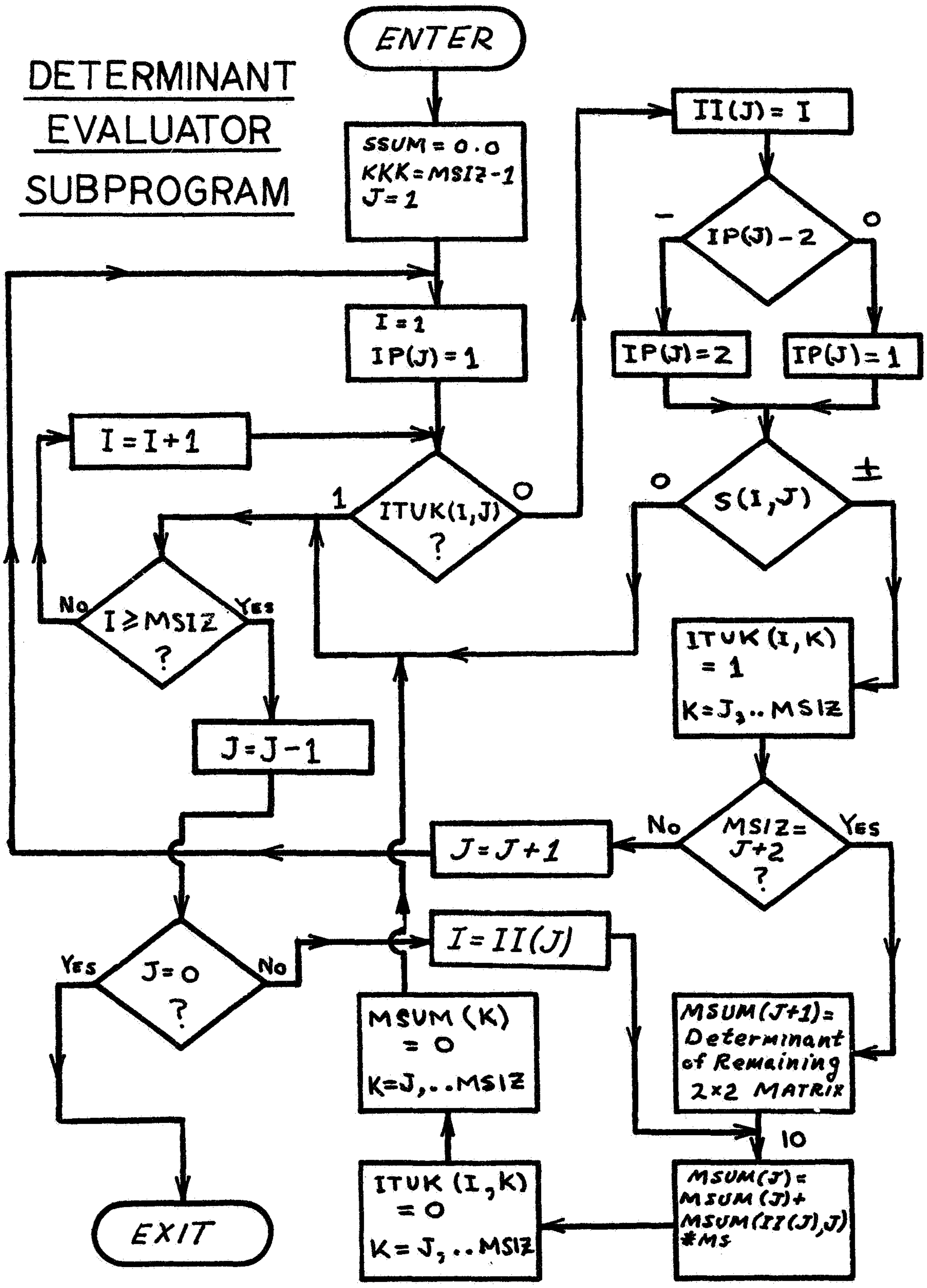
Someone casually inspecting the program might be upset to note that brute force, instead of some clever method, is used to evaluate determinants. The reason for this is multifold: first, many 'clever' methods break down or lose accuracy when dealing with singular matrices, or with matrices whose entries differ widely in magnitude; further, all available methods destroyed the matrix being evaluated--an unforgivable trait in this instance; and lastly, perhaps as a consolation, 'brute force' allows one to take advantage of the large number of zero entries that are anticipated.

It is sufficient here to note that the program is given the two entry distributions, the size of the matrices to be generated, and the number to be checked. It then generates the matrices, counts the number of zero entries in each, notes at which step of the criterion it is rejected, or if stable, stores the matrix on tape for later printout.

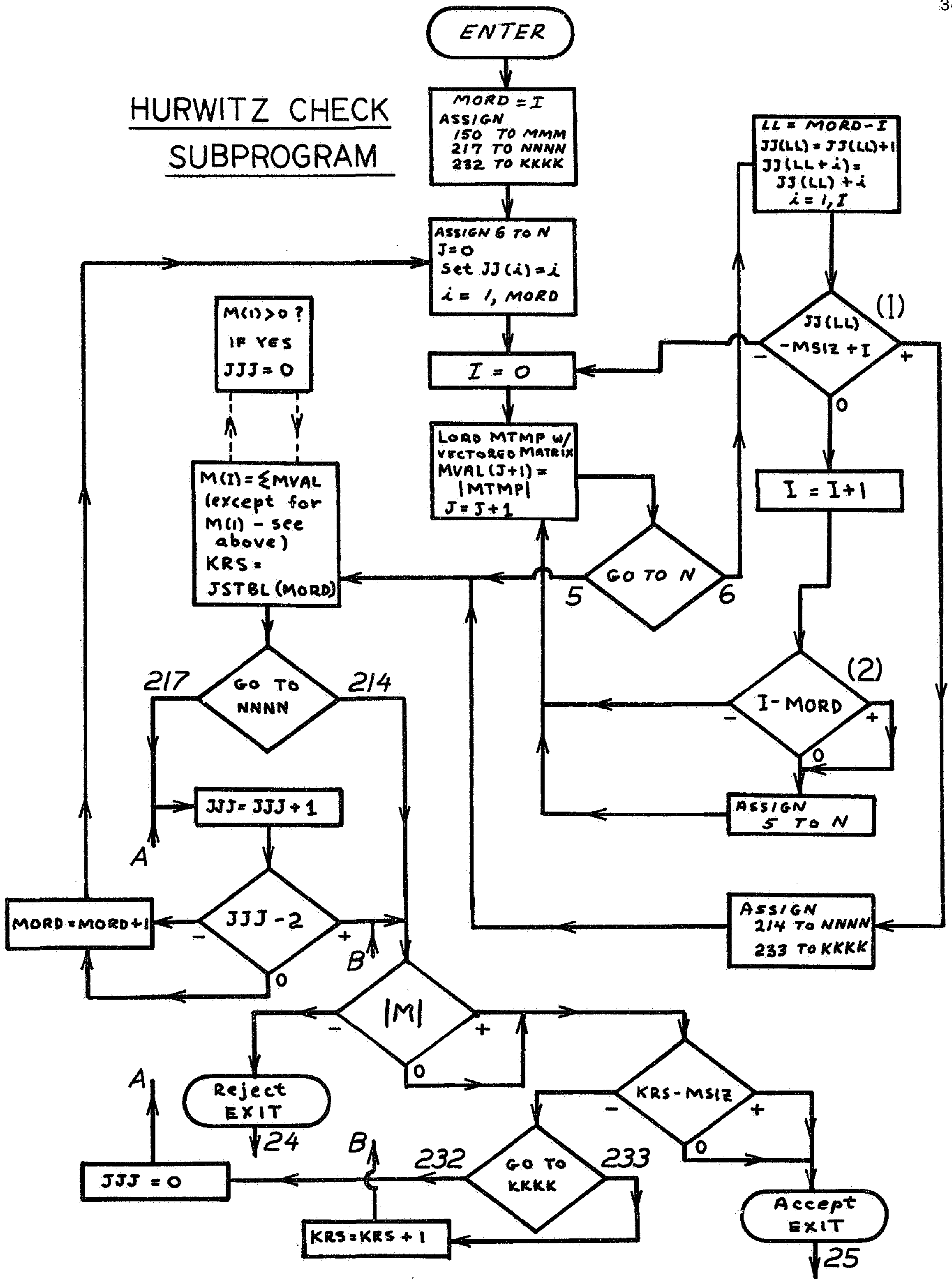
The output consists of, first, a list of the number of systems rejected at each step of the Hurwitz criterion; second, a list of number of matrices with each number of zeros which were found to be stable and unstable; and finally, a printout of all the stable matrices.

A program listing follows the flow charts. Please note that RIT7 should be interpreted as READ; WOT6 should be interpreted as PRINT.

DETERMINANT EVALUATOR SUBPROGRAM



HURWITZ CHECK SUBPROGRAM



The two decision steps numbered in parentheses on the previous page are of a special nature:

(1) If this test is

negative: JJ(MORD) <MS1Z, continue to generate vectors.

zero: JJ(MORD) = MS1Z. Indicates that present (as yet untaken) vector is last with given first components. Set up to change a previous component.

plus: only when vector just generated is 1,2 ... MS1Z. Set up to skip further evaluation of M, and for relooping on check - and go get final matrix.

(2) If this test is

negative - not yet to final vector of size MORD, continue.

zero - present vector is last in series of size MORD. Set up for evaluation, loading. Plus is not used.

Meaning of selected program variable names:

IAR - number of matrices of current size that will fit across a page of output.

III - position index in Hurwitz check portion.

ITP - gives number of matrices of each size that has been stored on tape.

JJJ - special counter in Hurwitz check portion.

K - modified ITP, for output purposes.

KL - possible number of zeros in matrix.

KREJ - table of numbers of matrices rejected at each step of the Hurwitz check.

KRS - carries the test number in the Hurwitz check portion, gives step at which matrix is rejected. KRS is set to zero if matrix is stable.

LEFT - number of matrices to be printed in last row of output.

LIM - number of matrices of each size to check.

MCTR - matrix counter, gives number of matrices of current size which have been checked.

MORD - current size of submatrices being generated.

MS1Z - current size of the matrices under test.

MSVØ - table of numbers of stable matrices versus number of zeros in the matrices.

MVVØ - table of numbers of unstable matrices versus number of zeros in the matrices.

ND - largest size matrix to check.

NMLIST - list of numbers, 0-100, for output purposes.

NN - number of zeros in matrix currently under test.

S - Hurwitz matrix of characteristic equation coefficients.

SAT - the matrix being tested for stability.

SEVAL - the result of subroutine EVAL in general, also used for summing SVAL.

SMAT - overlay matrices, used only to facilitate the orderly dumping of the matrices which have been stored on tape.

STMP - a temporary matrix which holds each submatrix for evaluation.

SVAL - a list which holds the values of submatrices for later summation.

```
C  MAIN PROGRAM MODIFIED TO ACCEPT ONLY MATRICES WITH CLOSE TO MODE ZEROS
C   ---- DATA SEQUENCE FOR THIS PROGRAM
C         1 CARD,3I4, MSTT, ND, LIM,MODE
C         3 CARDS, DOCUMENTATION INFORMATION
C        1 CARD, 10I2, K (RAM2 INITIALIZATION)
C        41 CARDS, DISTRIBUTION TABLE
C  ALSO, 41 CARDS WITH NON-ZERO DISTRIBUTION FOR DIAG TO READ.
      DIMENSION SAT(11,11),STMP(11,11),SMAT(10,10,10),S(11,11),MSVO(101)
     1,MUVO(101),SVAL(260),KREJ(11),ITP(11),NC(4),JJ(11),KTM(3,10),
     2NMLST(101)
      DIMENSION SMNTD(1001),SMNTS(1001)
      DIMENSION ISTBL(11 ),JSTBL(11)
      DIMENSION ITUK(11,11),SSUM(10)
      DIMENSION NAME (36)
      COMMON SAT,STMP,SMAT,S,MSVO,MUVO,SVAL,KREJ,ITP,NC,JJ,MSIZ,KRS,SEVA
     1L,MORD,NMLST
      WOT6, 51
      DO 220 I=1,10
      SSUM(I)=0.0
      DO 220 J=1,10
  220 ITUK(I,J)=0
      RIT7,7,MSTT,ND,LIM,MODE
    7 FORMAT (7I4)
      RIT7,305,((NAME(I)),I=1,36)
  305 FORMAT (2X12A6)
C-    INITIALIZE
      DO 72 I=1,101
   72 NMLST(I)=I-1
      MSIZ=MSTT
      DO 89 I=MSIZ,ND
   89 ITP(I)=0
      DO 237 I=1,10
      ISTBL(I)=(I+2)/2
      IF (2*ISTBL(I)-2-I) 161,162,161
  161  JSTBL(I)=1
      GO TO 237
  162  JSTBL(I)=2
  237  CONTINUE
C  INITIALIZE RANDOM GENERATOR
  991 RIT7,99,K
   99 FORMAT(10I2)
      CALL RAM2A(K)
C  READ IN DISTRIBUTION TABLE, SET DIGEN TABLE
      RIT7,301,((SMNTS(I)),I=1,1001)
  301 FORMAT (25F3.1)
      CALL DIAG (SMNTS,SMNTD)
C-----LOADS PERMANENT ZEROS AND ONES INTO M
      DO 173 I=1,5
      J=2*I
      S(I,J)=1.0
      J=J+1
      DO 173 JJ=J,10
  173 S(I,JJ)=0 .0
```

```
C-----LOADS ZEROS IN CELLS M(I) GT M(MSIZ)
      I=(MSIZ+3)/2
      IF(2*(I-1)-MSIZ) 42,43,42
   43 J=1
      GO TO 44
   42 J=2
   44 DO 45 II=I,10
      DO 46 JJ=1,J
   46 S(II,JJ)=0.0
      J=J+2
   45 CONTINUE
      GO TO 777
  201 MSIZ=MSIZ+1
  777 DO 18 I=1,MSIZ
   18 KREJ(I)=0
      MCTR=0
      KL=MSIZ*MSIZ+1
      DO 19 I=1,101
      MSVO(I)=0
   19 MUVO(I)=0
C    SUBROUTINE TO GENERATE OFF DIAGONAL ELEMENTS
   87 DO 20 I=1,MSIZ
      DO 20 J=1,MSIZ
      IF(I-J) 29,20,29
   29 X=RAM2B(0)
      KK=XFIXF(1000.0*X+1.0)
      SAT(I,J)=SMNTS(KK)
   20 CONTINUE
C   SUBROUTNE TO GENERATE DIAGONAL ELEMENTS
      DO 88 I=1,MSIZ
      X=RAM2B(0)
      KK=XFIXF(1000.0*X+1.0)
   88  SAT(I,I)=SMNTD(KK)
      NN=1
      DO 21 I=1,MSIZ
      DO 21 J=1,MSIZ
      IF (SAT(I,J)) 21,23,21
   23 NN=NN+1
   21 CONTINUE
C ---- CHECK TO SEE IF THE NUMBER OF ZEROS IS ACCEPTABLE
      IF (3-XABSF (NN-MODE-1)) 401,401,400
  401 MUVO(101)=MUVO(101)+1
      GO TO 87
  400 CONTINUE
C-----SUBROUTINE FOR HURWITZ CHECK
      MORD=1
      ASSIGN 150 TO MMM
      ASSIGN 217 TO NNNN
      ASSIGN 232 TO KKKK
  151 ASSIGN 6 TO N
      J=0
```

```
      DO 1 I=1,MORD
    1 JJ(I)=I
   13 I=0
    2 DO 22 K=1,MORD
      DO 22 L=1,MORD
      KK=JJ(K)
      LL=JJ(L)
   22 STMP(K,L)=SAT(KK,LL)
      J=J+1
      CALL EVAL (STMP,SEVAL,MORD,ITUK,SSUM)
      SVAL(J)=SEVAL
      GO TO N,(5,6)
    6 LL=MORD-I
      JJ(LL)=JJ(LL)+1
      DO 32 II=1,I
      KK=LL+II
   32 JJ(KK)=JJ(LL)+II
      IF(JJ(LL)-MSIZ+I) 13,3,55
    3 I=I+1
      IF(I-MORD) 2,4,4
    4 ASSIGN 5 TO N
      GO TO 2
   55 ASSIGN 214 TO NNNN
      ASSIGN 233 TO KKKK
    5 SEVAL =0  .0
      DO 35 K=1,J
   35 SEVAL=SEVAL+SVAL(K)
      S(MORD,MORD)=SEVAL*(-1.0)**MORD
      GO TO MMM,(150,152)
  150 ASSIGN 152 TO MMM
      IF(S(1,1)) 145,153,153
  145 KRS=1
      GO TO 24
  153 JJJ=0
C-----LOAD M INTO MATRIX WHERE NEEDED
C     KRS AND III ARE START POSITIONS OF WRITE IN M. THE STATEMENTS AFTER
C     213 ROUTE CONTROL SO THAT AFTER ALL PERTINANT M ARE CALCULATED, THE
C     M MATRIX IS CHECKED TO COMPLETION WITHOUT FURTHER CALCULATIONS
  152  KRS=ISTBL(MORD)
      III=JSTBL (MORD)
      KK=(MSIZ+MORD)/2
  163  DO 158 II=KRS,KK
      S(II,III)=S(MORD,MORD)
  158 III=III+2
```

```
      GO TO NNNN,(217,214)
C-----SEE IF NEXT TWO M'S HAVE BEEN GENERATED. IF NOT, GET SECOND, IF SO
C     EVALUATE THE AVAILABLE KRS X KRS MATRIX
  217 JJJ=JJJ+1
      IF(JJJ-2) 213,213,214
  213  MORD=MORD+1
      GO TO 151
  215 IF (KRS-MSIZ) 216,25,25
  216 GO TO KKKK,(232,233)
  233 KRS=KRS+1
      GO TO 214
  232 JJJ=0
      GO TO 217
C-----CHECK MATRIX -- IF GOOD, CHECK IS MAT IS TOTALLY CHECKED
  214 CALL EVAL (S,SEVAL,KRS,ITUK,SSUM)          --IF NOT, GO TO 217
      IF (SEVAL) 24,215,215
C-----THIS POINT IS EXIT FROM HURWITZ CHECK SUBROUTINE
   24 MUVO(NN)=MUVO(NN)+1
      KREJ(KRS)=KREJ(KRS)+1
      GO TO 26
   25 MSVO(NN)=MSVO(NN)+1
      ITP(MSIZ)=ITP(MSIZ)+1
      WRITE OUTPUT TAPE 4,12,((SAT(I,J),I=1,MSIZ),J=1,MSIZ)
   12 FORMAT (   F5.1)
   26 MCTR=MCTR+1
      IF (MCTR-LIM) 87,27,27
   27 KK=MSIZ+1
      WOT6,305,((NAME(I)),I=1,36)
      WOT6,50
       WOT6, 28,MSIZ,(NMLST(I),I=2,KK)
   28 FORMAT (12X12HMATRIX SIZE I4//15X5HI =     10I6)
      WOT6, 31,(KREJ(I),I=1,MSIZ)
   31 FORMAT (20H REJECTED AT STEP I 10I6)
      WOT6, 50
   50 FORMAT (//)
      LL=10
      DO 34 I=1,KL,10
      IF(KL-I-10) 61,62,62
   61 LL=KL-I
   62 DO 877 J=1,LL
      KK=J+I  -1
      KTM(1,J)=NMLST(KK)
      KTM(2,J)=MSVO(KK)
  877 KTM(3,J)=MUVO(KK)
```

```
      WOT6,876,(KTM(1,J),J=1,LL)
      WOT6,878,(KTM(2,J),J=1,LL)
      WOT6,879,(KTM( 3,J),J=1,LL)
  876 FORMAT (20H NUMBER OF ZEROS         10I6)
  878 FORMAT (20H NUMBER STABLE         10I6)
  879 FORMAT (20H NUMBER UNSTABLE       10I6//)
   34 CONTINUE
      WOT6, 51
   51 FORMAT (1H1)
      IF (MSIZ-ND+1) 201,201,202
  202 WRITE OUTPUT TAPE 4, 12,((SAT(K,L),L=1,MSIZ),K=1,MSIZ)
      REWIND 4
C     SAVE RANDOM GENERATOR INDEX
      CALL RAM2C(K)
      WOT5,99,K
      WOT6,305,((NAME(I)),I=1,36)
      WOT6,52
   52 FORMAT (//20H   STABLE MATRICES--    //)
      I=MSTT
  209 K=ITP(I)
      IF (K) 203,203,208
  208 IAR=20/I
      NN=K/IAR
      LEFT=K-NN*IAR
      K=NN*IAR-1
      IF(NN) 211,211,212
  212 ASSIGN 204 TO MM
      DO 204 J=1,K,IAR
  210 DO 205 KK=1,IAR
  205 READ INPUT TAPE 4,12,(((SMAT(KK,II,JK)),II=1,I),JK=1,I)
      DO 206 II=1,I
  206 WOT6, 300,(((SMAT(KKK,II,JK)),JK=1,I),KKK=1,IAR)
      WOT6, 50
  300 FORMAT (1X20F5.1)
      GO TO MM,(204,203)
  204 CONTINUE
  211 IAR=LEFT
      ASSIGN 203 TO MM
      GO TO 210
  203 I=I+1
      IF (I-ND) 209,209,999
  999 CALL SYSTEM
      END
```

```
      SUBROUTINE EVAL (S,SEVAL,MSIZ,ITUK,SSUM)
      DIMENSION S(11,11),ITUK(11,11),II(10),S2X2(2,2),SSUM(10)
      DIMENSION IP(11)
      IF (MSIZ-2) 25,26,27
   25 SEVAL=S(1,1)
      RETURN
   26 SEVAL=S(1,1)*S(2,2)-S(1,2)*S(2,1)
      RETURN
   27 SSUM (1)=0 .0
      J=1
      KKK=MSIZ-1
    2 I=1
      IP(J)=1
    3 IF (ITUK(I,J)) 4,4,5
    4 II(J)=I
      IF (IP(J)-2) 30,31,30
   30 IP(J)=2
      GO TO 40
   31 IP(J)=1
   40 IF (S(I,J)) 32,5,32
   32 DO 12 L=J,MSIZ
   12 ITUK(I,L)=1
      IF(J+2-MSIZ) 8,9,8
    5 IF (I-MSIZ) 6,7,7
    6 I=I+1
      GO TO 3
    7 J=J-1
      IF (J) 11,11,13
   11 SEVAL=SSUM(1)
      RETURN
   13 I=II(J)
      GO TO 10
    8 J=J+1
      GO TO 2
    9 KK=KKK
      LL=1
      III=0
   14 III=III+1
      IF(ITUK(III,KK)) 14,15,14
   15 S2X2(LL,1)=S(III,KK)
      S2X2(LL,2)=S(III,MSIZ)
      LL=LL+1
      GO TO (14,14,21),LL
   21 SSUM(KKK)=S2X2(1,1)*S2X2(2,2)-S2X2(1,2)*S2X2(2,1)
   10 KK=J+1
      SSUM(J)=SSUM(J)+S(I,J)*SSUM(KK)*(-1.0)**IP(J)
      ITUK(I,J)=0
      DO 19 L=KK,MSIZ
      ITUK(I,L)=0
   19 SSUM(L)=0.0
      GO TO 5
      END
```

APPENDIX B

FORMING SUMS AND PRODUCTS OF DISTRIBUTIONS

While analytic techniques exist for finding sums and products of distributions, these are applicable to continuous distributions and quickly become extremely complicated to employ. Frequently, they are not applicable if either distribution includes the point zero.

The author devised a simple means of finding the sum and product of two discrete distributions, based on elementary counting. An example will demonstrate the technique.

Consider two very simple distributions as follows:

a_1

0 with probability 60%

1 with probability 40%

a_2

1 with probability 30%

2 with probability 70%

We desire to find the sum distribution. The method is equivalent to forming the following table:

a_2/a_1	0	0	0	0	0	0	1	1	1	1
1	1	1	1	1	1	1	2	2	2	2
1	1	1	1	1	1	1	2	2	2	2
1	1	1	1	1	1	1	2	2	2	2
2	2	2	2	2	2	2	3	3	3	3
2	2	2	2	2	2	2	3	3	3	3
2	etc.									
2										
2										
2										
2										

The resultant distribution is determined by counting the frequency of entries in the table. The program, of course, does not bother adding, say, 0 to 1 more than once, but the idea is the same.

The program adds every pair once, and notes the product of their weights. In the current example we have

$$0 + 1 = 1 \quad (.60)(.30) = .18$$
$$0 + 2 = 2 \quad (.60)(.70) = .42$$
$$1 + 1 = 2 \quad (.40)(.30) = .12$$
$$1 + 2 = 3 \quad (.40)(.70) = .28$$

The program then adds the probabilities for each value of the sum, so that $a_1 + a_2 =$

1 with probability .18
2 with probability .54
3 with probability .28

The same procedure is followed to obtain product distributions.

This programming effort was subdivided into subroutines, each called on command by an input data card. Thus the programmer has complete external control over the sequence of operations. The programming was not difficult and will not be presented here. The only special feature was that distributions were output as graphs by the computer, thus saving hours of tedious labor.

APPENDIX C

SYNOPSIS OF OUTDATA

Matrix Size 2 x 2

Number zeros	Degenerateness, %	Number Tested	Number Stable	Percent Stable
0	0	192	144	75
1	50			100
2	100			100

Matrix Size 3 x 3

Number zeros	Degenerateness, %	Number Tested	Number Stable	Percent Stable
0	0	152	84	55
1	17	103	69	65
2	33	81	59	73
3	50	47	38	81
4	67	48	42	87
5	83	49	49	100
6	100	20	49	100

Matrix Size 4 x 4

Number zeros	Degenerateness, %	Number Tested	Number Stable	Percent Stable
0	0	49	16	33
1	8	60	26	43
2	16	36	14	39
3	25	28	14	50
4	33	27	14	52
5	42	36	17	47
6	50	26	17	65
7	58	30	22	73
8	66	31	25	81
9	75	33	32	97
10	83	30	29	97
11	92	8	8	100
12	100	6	6	100

Matrix Size 5 x 5

Number zeros	Degenerateness, %	Number Tested	Number Stable	Percent Stable
0	0	36	9	25
1	5	47	17	28
2	10	20	3	15
3	15	20	3	15
4	20	19	3	16
5	25	26	9	35
6	30	21	6	28
7	35	17	4	24
8	40	16	4	25
9	45	18	5	28
10	50	20	13	65
11	55	20	8	40
12	60	13	7	54
13	65	19	12	63
14	70	22	15	68
15	75	25	23	92
16	80	20	19	95
17	85	14	12	86
18	90	6	6	100
19	95	0		
20	100	1	1	100

Matrix Size 6 x 6

Number zeros	Degenerateness, %	Number Tested	Number Stable	Percent Stable
0	0	54	3	6
1	3	74	8	11
2	7	50	8	16
3	10	19	1	5
4	13	12	0	0
5	17	17	2	12
6	20	19	3	15
7	23	18	6	33
8	27	11	3	27
9	30	12	1	8
10	33	12	3	25
11	37	17	6	35
12	40	10	1	10
13	43	12	4	33
14	47	14	4	28
15	50	14	3	21
16	53	18	4	22
17	57	8	5	63
18	60	5	2	40
19	63	5	2	40
20	67	9	5	56
21	70	16	11	69
22	73	15	12	80
23	77	15	7	47
24	80	22	17	77
25	83	8	7	88
26	87	5	0	100
27	90	8	0	100
28	93	2	0	100
29	97	0		100
30	100	0		100

Matrix Size 7 x 7

Number zeros	Degenerateness, %	Number Tested	Number Stable	Percent Stable
22	52	1	0	0
23	55	1	0	0
24	57	0	0	
25	60	2	0	0
26	62	4	1	25
27	64	4	2	50
28	66	8	1	12
29	69	12	4	33
30	71	10	4	40
31	74	13	9	69
32	76	10	8	80
33	79	11	8	73
34	81	8	7	87
35	83	9	8	88
36	86	4	4	100
37	88	1	1	100
38	90	2	2	100

Matrix Size 8 x 8

Number zeros	Degenerateness, %	Number Tested	Number Stable	Percent Stable
32	57	1	0	0
33	59	1	0	0
34	61	1	0	0
35	63	5	1	20
36	64	1	0	100
37	66	12	4	33
38	68	9	3	33
39	70	7	2	29
40	71	7	3	43
41	73	19	6	32
42	75	9	6	66
43	77	7	4	57
44	70	5	3	60
45	80	3	2	66
46	82	4	3	75
47	84	5	2	40
48	86	1	1	100
49	88	0	0	
50	89	2	2	100
51	91	1	1	100

Matrix Size 10 x 10

Number zeros	Degenerateness, %	Number Tested	Number Stable	Percent Stable
74	82	3	0	0
75	83	1	0	0
77	86	1	0	0
78	87	2	1	50
79	88	2	2	100
81	90	1	1	100
82	91	1	1	100

BIBLIOGRAPHY

1. Ashby, W. Ross. Design for a Brain. New York: John Wiley & Sons, 1960.

2. Ashby, W. Ross. Personal Correspondence.

3. D'Azzo, John J., and Houpis, Constantine H. Feedback Control System Analysis and Synthesis. New York: McGraw-Hill Book Company, 1966.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* University of Illinois Biological Computer Laboratory Urbana, Illinois 61801	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

CRITICAL DEGENERATENESS IN LARGE LINEAR SYSTEMS

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*
scientific; ; interim

5. AUTHOR(S) *(Last name, first name, initial)*

Gardner, Mark R.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 15, 1968	52	3

8a. CONTRACT OR GRANT NO. AF-AFOSR 7-67 b. PROJECT AND TASK NO. 9769-04 c. 6144504F 681304 d.	9a. ORIGINATOR'S REPORT NUMBER(S) BCL Report 5.8 9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*

10. AVAILABILITY/LIMITATION NOTICES

Distribution of this document is unlimited.

11. SUPPLEMENTARY NOTES Partial sponsorship under AF33(615)-3890 and NASA NGR 14-005-111.	12. SPONSORING MILITARY ACTIVITY Air Force Ofc. of Scientific Research Directorate of Information Sciences Arlington, Virginia 22209

13. ABSTRACT

A class of linear systems, composed of intrinsically stable elements, was studied. These linear systems were represented by the coefficient matrices of their differential equations. A sample space of these matrices was defined by specifying the nature of the distributions from which the matrix entries were selected.

Matrices of given size were generated by random sampling from the defined sample space. Appropriate weighting of the distributions gave control of the degenerateness, a measure of the number of zero entries. The Hurwitz criterion was used to test whether each matrix represented a stable system. The primary goal was to find the probability of stability as a function of degenerateness.

It was found, even for the relatively small matrices within the range of this study, that the degenerateness is critical. For values of degenerateness less than a particular amount (about 85%), the system is almost certainly unstable, whereas for values of degenerateness greater than this amount, the system is almost certainly stable.

DD FORM 1 JAN 64 1473

UNCLASSIFIED

Security Classification

UNCLASSIFIED

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
stability systems probability of systems decomposition						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization *(corporate author)* issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year, or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8*c*, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers *(either by the originator or by the sponsor)*, also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through ________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through ________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through ________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring *(paying for)* the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS)*, *(S)*, *(C)*, *or (U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.

UNCLASSIFIED

Security Classification