

1966

Sampled-data adaptive control utilizing variable sampling rates

Richard Marinus Willett
Iowa State University

Follow this and additional works at: <https://lib.dr.iastate.edu/rtd>

 Part of the [Electrical and Electronics Commons](#)

Recommended Citation

Willett, Richard Marinus, "Sampled-data adaptive control utilizing variable sampling rates " (1966). *Retrospective Theses and Dissertations*. 3142.
<https://lib.dr.iastate.edu/rtd/3142>

This Dissertation is brought to you for free and open access by the Iowa State University Capstones, Theses and Dissertations at Iowa State University Digital Repository. It has been accepted for inclusion in Retrospective Theses and Dissertations by an authorized administrator of Iowa State University Digital Repository. For more information, please contact digirep@iastate.edu.

**This dissertation has been
microfilmed exactly as received 67-5636**

**WILLET, Richard Marinus, 1929-
SAMPLED-DATA ADAPTIVE CONTROL UTILIZING
VARIABLE SAMPLING RATES.**

**Iowa State University of Science and Technology, Ph.D., 1966
Engineering, electrical**

University Microfilms, Inc., Ann Arbor, Michigan

SAMPLED-DATA ADAPTIVE CONTROL UTILIZING
VARIABLE SAMPLING RATES

by

Richard Marinus Willett

A Dissertation Submitted to the
Graduate Faculty in Partial Fulfillment of
The Requirements for the Degree of
DOCTOR OF PHILOSOPHY

Major Subject: Electrical Engineering

Approved:

Signature was redacted for privacy.

In Charge of Major Work

Signature was redacted for privacy.

Head of Major Department

Signature was redacted for privacy.

Dean of Graduate College

Iowa State University
Of Science and Technology
Ames, Iowa

1966

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
A. Problem Description	1
B. Adaptive Control Systems	2
C. Description of the Sampled-Data Adaptive System	11
D. Process Identification	14
II. EXPERIMENTAL INVESTIGATION OF SAMPLED-DATA PROCESS IDENTIFICATION	26
III. VERIFICATION OF ADAPTATION BY VARIATION OF SAMPLING RATE	48
IV. CONCLUSIONS	61
V. LITERATURE CITED	62
VI. ACKNOWLEDGMENT	64
VII. APPENDIX A - COMPUTER PROGRAM FOR THE SIMULATION OF THE CLOSED LOOP SYSTEM WITH COMPENSATION	65
VIII. APPENDIX B - COMPUTER PROGRAM FOR LEAST SQUARES COEFFICIENT IDENTIFICATION	69
IX. APPENDIX C - PROGRAM TO CONVERT LAPLACE TRANSFORMS TO Z-TRANSFORMS	72

I. INTRODUCTION

A. Problem Description

Modern aircraft and missiles operate over an extremely wide range of conditions and their dynamics or response characteristics vary widely. For this reason, a control system which gives a stable well behaved response for low speeds and low altitudes may not be satisfactory at high speeds and/or high altitudes. In fact, the system could be completely unstable.

This type of problem has been solved in the past by control systems in which the gains of the various feedback loops are changed in response to signals generated by a computer. In effect, the computer determines altitude and velocity and then adjusts the control system gains to some predetermined value which should give optimum characteristics. This is essentially an "open loop" type of operation. That is, if the predetermined gains are not correct and the system should become unstable, there is no feedback mechanism to allow further correction.

An additional complication to the problem appears in large space vehicles or boosters, and in advanced aircraft such as the XB-70. In these vehicles, the structure cannot be considered a rigid body. Bending of the body about the transverse axes can be sensed by rate gyros. This signal from the gyros is fed into the control system along with the rigid body motion of the vehicle and can result in an instability at the natural frequency of the structure. Furthermore, the frequency and amplitude of the body bending varies with flight conditions and flight time. That is, the amount of mass (fuel) will change during the flight.

and hence the natural frequency and damping of any resonant mode must be expected to change.

In addition to these problems there are many cases, particularly in experimental vehicles, in which the exact dynamic characteristics are not known and cannot be predicted with any reliability. The X-15 and the XB-70 are two examples of this. In these cases, attempts to change the gains in the flight control system while in operation are not satisfactory.

B. Adaptive Control Systems

Many of the problems described above are solved, at least to a large extent, by the use of adaptive control techniques. Adaptive control differs from ordinary gain changing in that adjustments are made on the basis of the performance of the system itself as it is operating, and not on the basis of measurements of some external phenomena such as altitude or velocity.

One of the simplest and most successful techniques used to date was developed by Minneapolis-Honeywell (1). This technique is further described in a M.S. Thesis by R. C. Hendrick (2). It has been used with good results on the X-15. In the Honeywell system, a controller is designed such that there are a total of three more poles than zeros in the overall open loop transfer function. The compensation is designed so that for some very high gain, all but two of the poles are well into the left half plane, and two complex poles are approaching the imaginary axis at a frequency well above that of the aircraft response. See Figure 1. The response of the system is extremely rapid so that input signals must

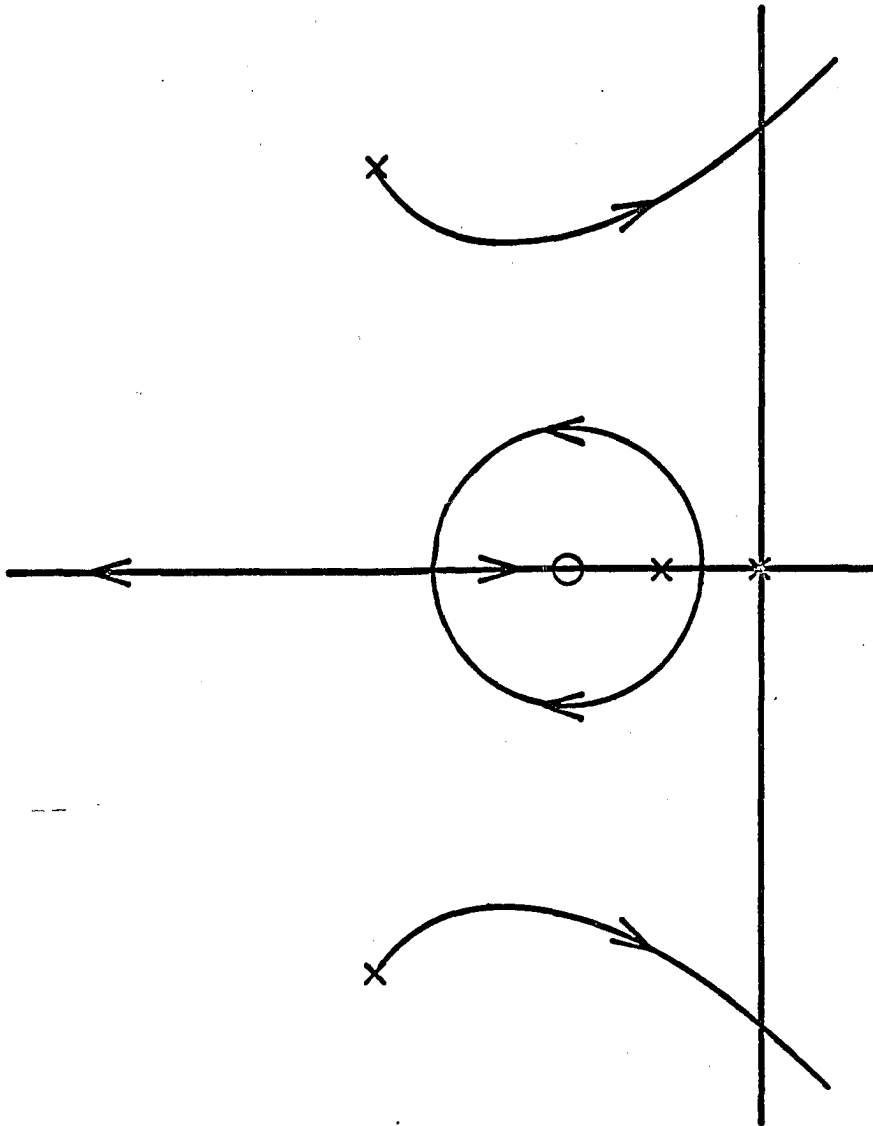


Figure 1. Root locus of high gain adaptive control system

be filtered to prevent excessive g-force buildup, and prevent damage to the structure. To ensure that the gain is held at a sufficiently high level without allowing the two complex poles to cross into the right half plane, a gain changing device must be used. A bandpass filter tuned to the natural frequency of the complex poles detects the buildup of oscillations. When the amplitude becomes too large, the gain is reduced. If no oscillation at all appears, the gain is slowly increased. The result is a continuous, small oscillation in the system gain, and a small limit cycle in the system output at the frequency of the complex poles. A block diagram of the Honeywell High Gain Adaptive Control system (2) is shown in Figure 2.

Another system, which uses a variation of the Honeywell technique, has been developed by General Electric (3). In this system the two complex poles are not allowed to reach the imaginary axis. The sensing for gain setting is accomplished by noting the presence of signals in the output at the natural frequency of the complex poles. As the gain is varied, the frequency as well as the damping of this pair of poles will change. The gain is adjusted to keep these poles at the same frequency and therefore nearly the same position in the s-plane. The closed loop system then has effectively the same characteristics as the Honeywell system except there is no low amplitude oscillation or limit cycle in the output of this system.

In practice, this system is somewhat more sensitive to noise and wind gusts than the Honeywell system and has not been too satisfactory in operation.

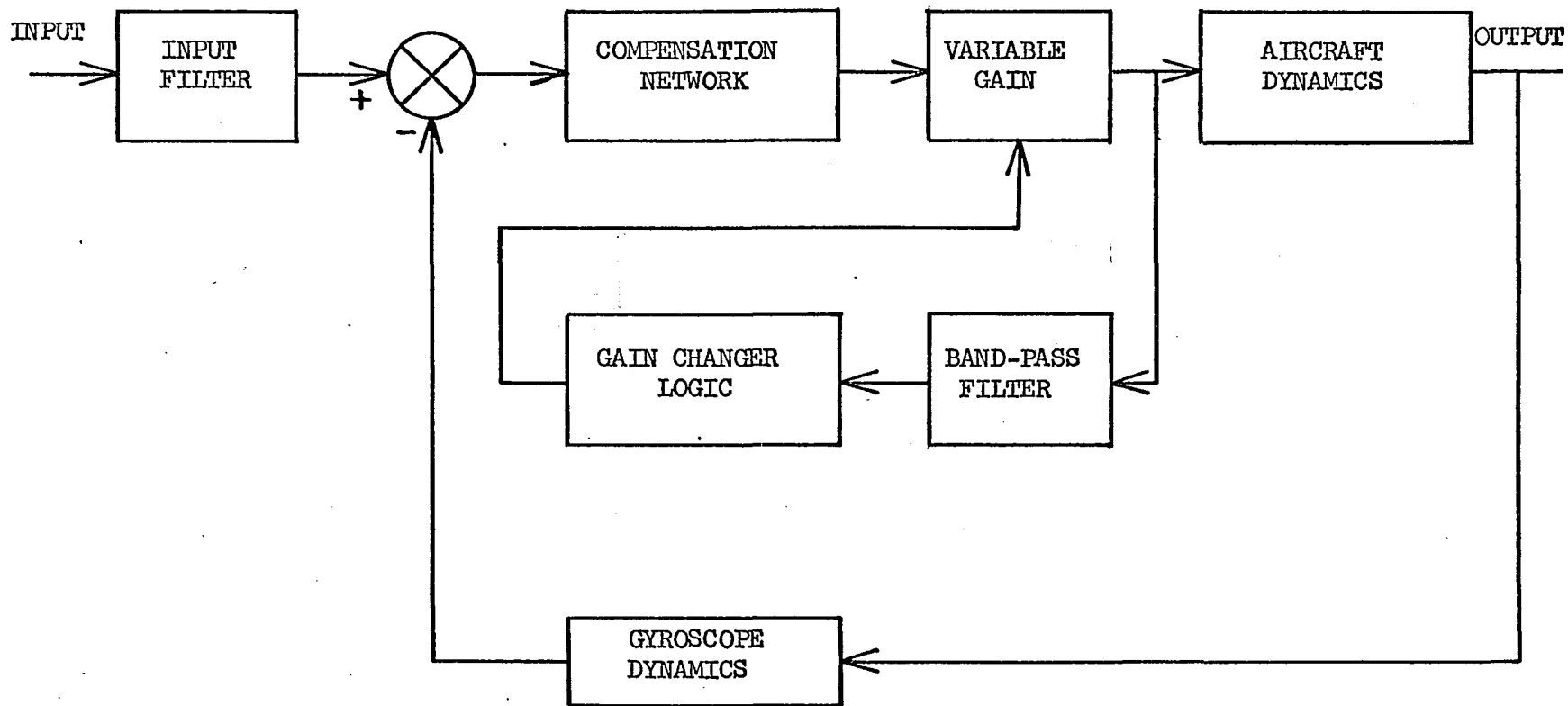


Figure 2. Block diagram of a high gain adaptive flight control system

The two systems described here are not satisfactory for the control of missiles and aircraft when body bending cannot be ignored. Figure 3 is the pole-zero plot in the s-plane of the transfer function describing the pitch response of a flexible missile to control surface deflections. Only the first two bending modes are indicated since higher order modes are usually very small in amplitude. Since the body bending poles are so close to the imaginary axis, any high gain system will have stability problems unless some method is found to control the motion of these poles.

From the mode shapes shown in Figure 4 it is apparent that the amount of body bending sensed by the gyro depends upon the position of the gyro with respect to the node. If the gyro is at position A, there will be relatively large body bending signals present. If the gyro is at position C, there will also be large signals, but 180° out of phase with the signals picked up at A. At B there will be relatively little signal sensed for the second mode and none for the first mode. Unfortunately, this node shifts during flight, and even if it were stationary its position cannot be predicted with any accuracy. For this reason, attempts to eliminate the first bending mode by placing the gyro at the correct location are not practicable.

Several methods have been proposed to overcome this problem. Minneapolis-Honeywell has developed a so-called gyro-blender technique which effectively reduces the bending mode signal (4). In this system (see Figure 5) two gyroscopes are used. One is placed well ahead of the node and one well behind. The output of the two gyros are added and their gains adjusted. Since the two bending signals are 180° out of phase they are effectively canceled. As the node position changes,

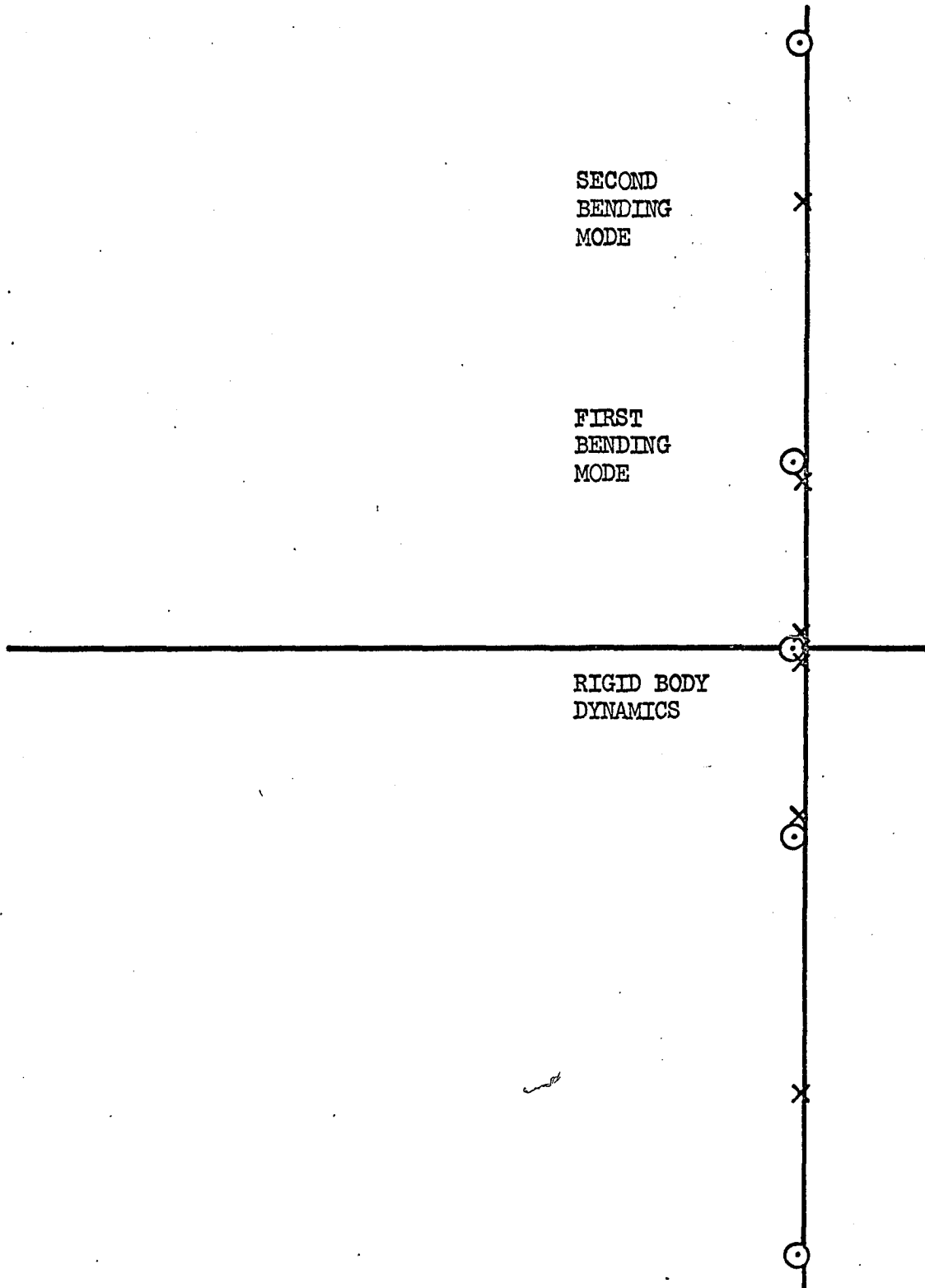


Figure 3. S-plane pole zero plot representing the dynamics of a flexible missile

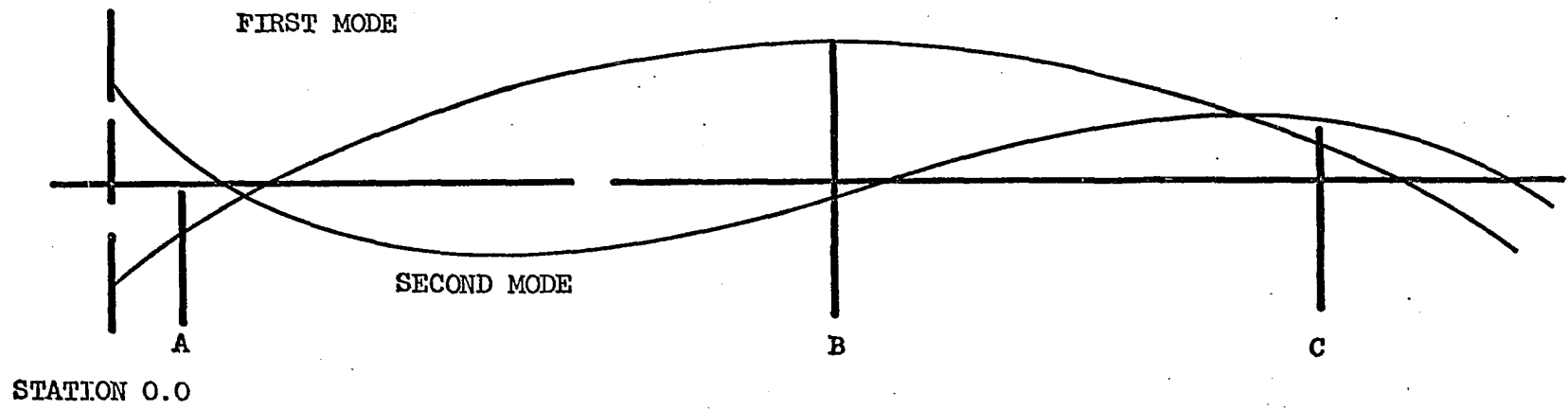


Figure 4. Mode shapes for a typical flexible missile

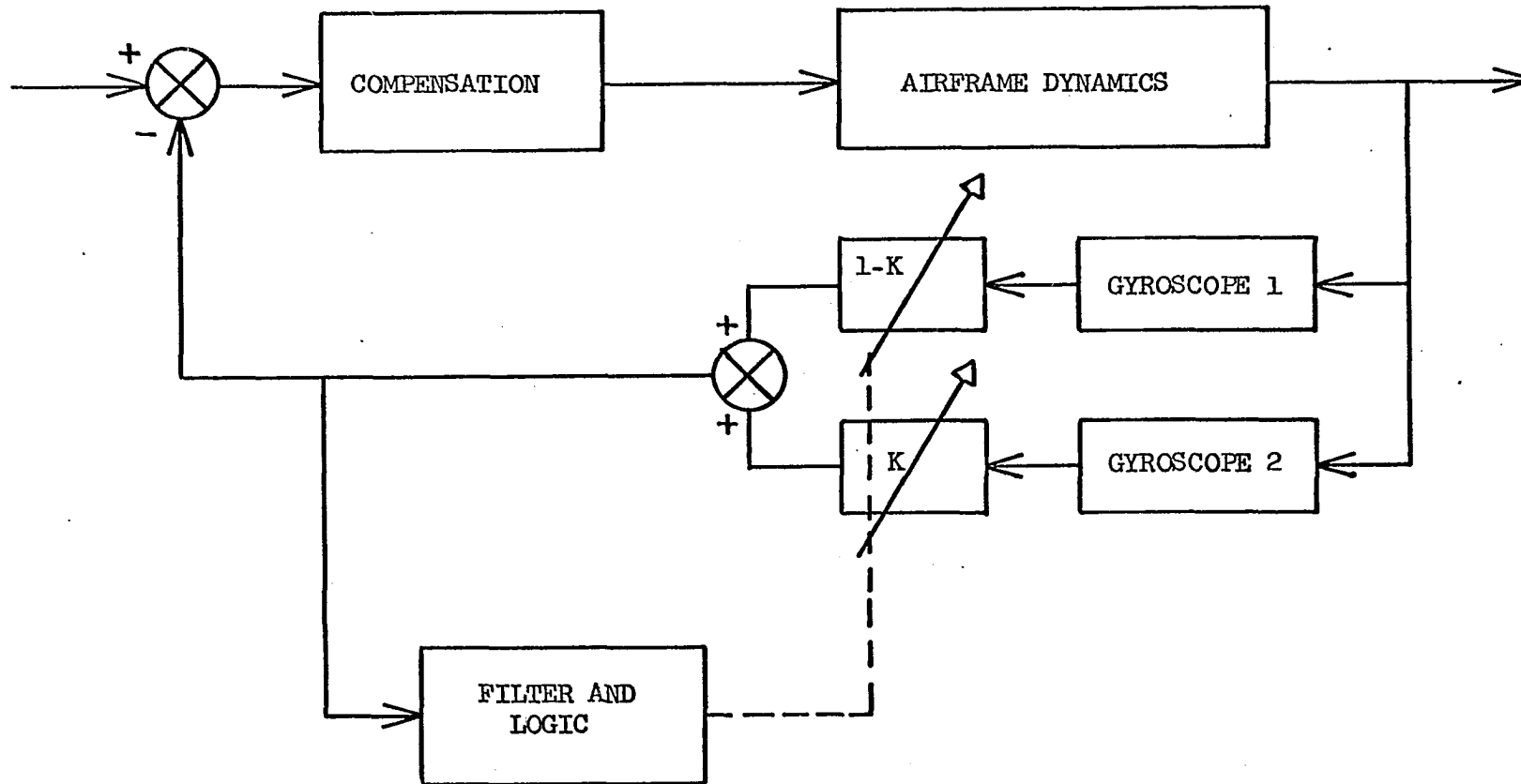


Figure 5. Gyro-blender adaptive control system block diagram

a filter and detector will detect the presence of the bending mode frequency in the blended signal, and adjust the gains to drive this unwanted signal to zero. This method can be used to eliminate one mode, but since the node for the second bending mode will not necessarily be near that of the first mode, it will not eliminate both bending signals. This is usually not too much of a problem, since the second mode can generally be kept stable more easily than the first mode.

A similar effect can be obtained using a band rejection filter (5). In this case, however, the parameters of the filter must be varied so that the rejection band will track the bending mode frequency.

Another method which has been discussed in the literature involves the use of a "synthetic" control signal (6). The actual feedback signal from the sensors is used only to determine the characteristics of the "synthetic" signal. The rigid body dynamics are usually approximated as a second order system and the frequency and damping are adjusted to give a least square fit with the actual feedback signal. This requires a digital computer with a fairly high speed since these computations must be done in real time. The resulting damped sinusoid is then used to actuate the control devices and maintain control of the vehicle.

All of the techniques discussed except the last one make use of some sort of identification of the system characteristics. In the Honeywell and General Electric high gain systems the critical gain is determined so that the gain can be set accordingly. In the gyro blender, the system effectively finds the node position, and in the band rejection filter, the mode frequency must be found.

The technique which was investigated is similar to these well known techniques in that it is necessary to determine the position of the poles characteristic of the first body bending mode. When this identification is accomplished, then the adaptive characteristic of the system will hold the pole in the proper position with respect to the compensation zero. This is accomplished by Smythe et al. (5) by moving the zero position. The technique which is the subject of this dissertation does not move the zero, but rather moves the pole associated with the first bending mode. This is accomplished by varying the sampling rate.

C. Description of the Sampled-Data Adaptive System

Advances in the state-of-the-art are continually making digital computers more attractive for flight control systems. This is particularly true for large missiles and boosters since the digital computer is already present for the guidance system and can be designed to include the flight control functions with very little additional hardware.

To understand the operation of the digital adaptive control system which was investigated, consider the complex pair of poles in the Z-plane in Figure 6 as being the first bending mode poles. If the poles are at the positions designated l , l' , and compensation zeros are added as shown, then a root locus plot might be made which would indicate that the poles will never move outside the unit circle. (Note that a locus such as this would be obtained if other poles and zeros for other bending modes, rigid body dynamics etc. were included. These have been omitted in Figure 6 for the sake of clarity.)

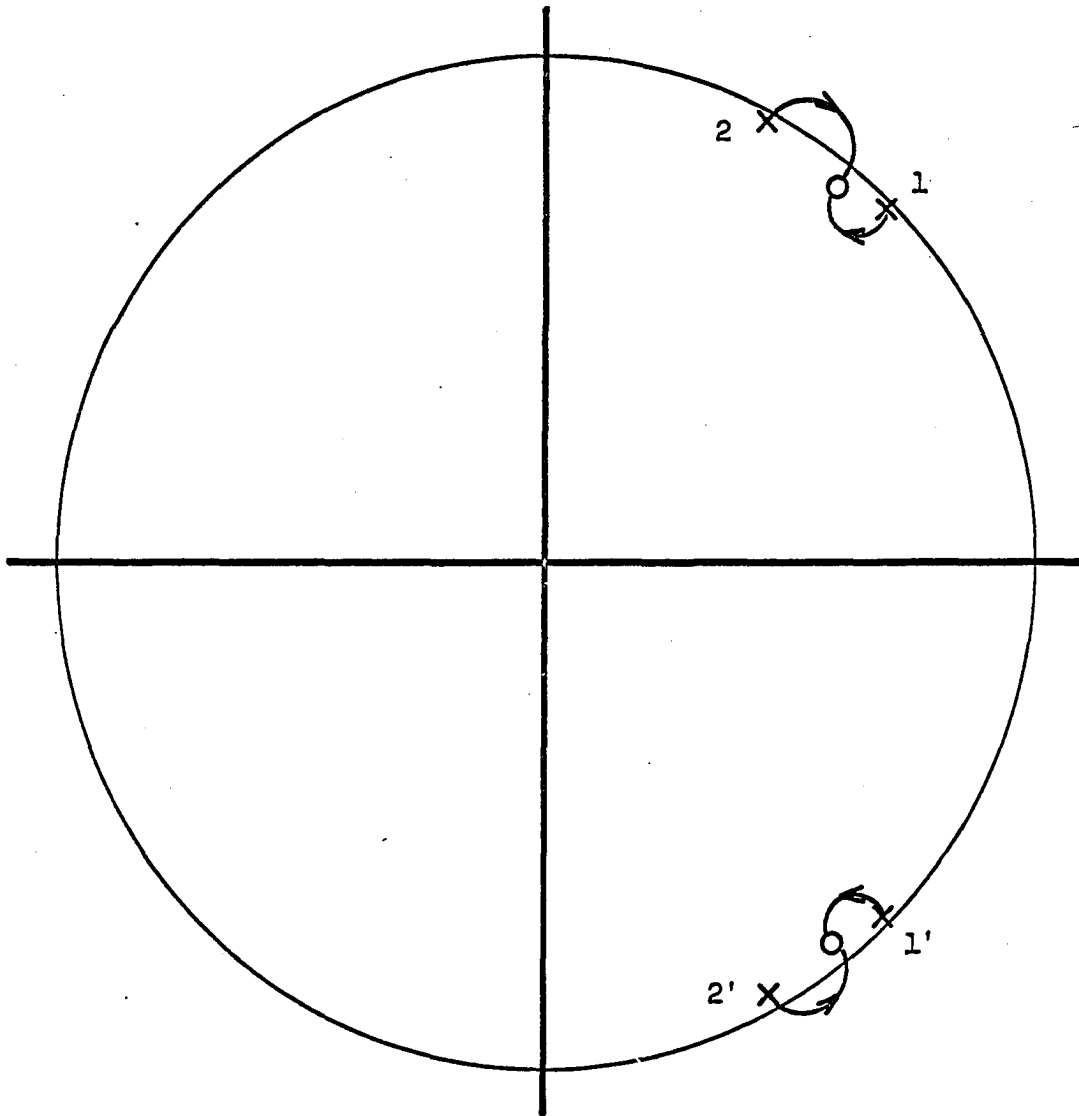


Figure 6. Bending mode poles in the Z-plane

Assume that the bending mode poles move to positions 2, 2' due to changes in fuel load, dynamic pressure, etc., and the compensation zeros remain fixed. Then the root locus will have quite a different shape as shown in Figure 6. Since the locus is outside the unit circle, the system will be unstable for any other than very high or very low values of gain. In the majority of control systems, the gain setting will be determined by other factors such as error specifications or rigid body stability as in the Honeywell adaptive system. (The system investigated might be used in conjunction with such a high gain system.) Since the gain must be set to meet other requirements, the bending mode can't be kept stable by gain adjustment. The method most frequently suggested is to change the compensation zero location. This is the method of Smythe et al. (5) and Prince (4), but is not the method described here.

The pole positions in the Z-plane are given by the zeros of the expression

$$Z^2 - 2Z e^{-\alpha T} \cos \omega_0 T + e^{-2\alpha T} \quad (1)$$

where ω_0 and α are defined in the time function $e^{-\alpha t} \sin \omega_0 t$, and T is the sampling period of the sampled data system. The zeros of this polynomial are

$$Z = e^{-\alpha T} [\cos \omega_0 T \pm j \sin \omega_0 T] = e^{-\alpha T} e^{\pm j \omega_0 T} \quad (2)$$

If α is small, which is the case for a lightly damped characteristic such as will describe the body bending modes, and if ω_0 is relatively large, then changes in T will result in motion of the body bending mode

pole on a path which is nearly a circle with its center at the origin of the Z-plane. In this case then, motion of the body bending pole from positions 1, 1' to 2, 2' in Figure 6 can be compensated for by reducing T by the appropriate amount.

This change in T will also change the positions of the rigid body poles and the other bending mode poles. In the case of the rigid body poles, ω_0 will be small enough so that the change in position will not be troublesome. In the case of the poles corresponding to the second and third bending modes, their characteristics are such that they can usually be kept stable over a fairly wide range of T.

D. Process Identification

In order to properly adjust the value of T, the sampling interval, there must be some means of determining the position of the poles in the Z-plane. The identification of the pole positions in the Z-plane (or in the s-plane) is usually referred to as process identification. In the case of sample-data systems this is not a difficult theoretical problem.

The pulse transfer function of any process under investigation may be written in the form

$$\frac{Y(z)}{X(z)} = \frac{a_0 + a_1 z^{-1} + \dots + a_n z^{-n}}{b_0 + b_1 z^{-1} + \dots + b_n z^{-n}} = G(z) \quad (3)$$

The process identification problem is then that of evaluating the coefficients $a_0, a_1 \dots a_n, b_0, b_1 \dots b_n$. The assumption may be made that a_0 is equal to zero since no physical system will respond instantaneously. Furthermore, b_0 may be set equal to unity without

loss of generality. Then

$$Y(z) = (a_1 z^{-1} + a_2 z^{-2} + \dots + a_n z^{-n}) X(z) - (b_1 z^{-1} + b_2 z^{-2} + \dots + b_n z^{-n}) Y(z) \quad (4)$$

Since z^{-n} denotes a delay of n sample periods, the inverse z -transform gives the output at the m^{th} sample period as

$$Y(mT) = a_1 X([m-1]T) + \dots + a_n X([m-n]T) - b_1 Y([m-1]T) - b_2 Y([m-2]T) - \dots - b_n Y([m-n]T) \quad (5)$$

The expressions for the output at the $(m+1)^{\text{st}}$ sample period, $Y([m+1]T)$, and succeeding sample periods may also be written in a similar fashion. When enough simultaneous equations have been obtained it is possible to solve for all of the unknown coefficients ($a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n$) of the pulse transfer function. This will obviously require $2n$ equations, and is the simplest identification technique. It is assumed here and in the following work that the orders of numerator and denominator are both known.

This approach has the very serious disadvantage of being extremely sensitive to noise and to inaccuracies in the measurement of the output and input signals $X(kT)$ and $Y(kT)$. To eliminate, or at least to minimize, the effects of noise in the system, correlation functions are often employed in process identification (7). These techniques make use of the fact that the ratio of various correlation functions is equal to the pulse transfer function. For example, if $\frac{Y(z)}{X(z)} = G(z)$, then

$$\frac{\phi_{xy}(z)}{\phi_{xx}(z)} = \frac{\phi_{yy}(z)}{\phi_{yx}(z)} = G(z) \quad (6)$$

where ϕ_{xy} is the cross spectral function and ϕ_{xx} the spectral function.

Also,

$$\frac{\phi_{wy}(z)}{\phi_{wx}(z)} = G(z) \quad (7)$$

where w is some function which correlates with both the input and output.

The technique investigated here, however, does not rely on the calculation of correlation functions but, in effect, uses a least square smoothing approach. The equations describing the output are rewritten as

$$\begin{aligned} F = & a_1 X ([m-1]T) + a_2 X ([m-2]T) + \dots + a_n X ([m-n]T) \\ & - Y (mT) - b_1 Y ([m-1]T) - b_2 Y ([m-2]T) - \dots \\ & - b_n Y ([m-n]T) \end{aligned} \quad (8)$$

If the coefficients are known exactly, then $F = 0$. In general, noise and errors will be such that $F \neq 0$. To minimize F , both sides of the equation are squared. A large number of these equations may be written and summed to give the single equation

$$\begin{aligned} H = \sum_{m=1}^i F^2 = & \sum_{m=1}^i [a_1 X ([m-1]T) + a_2 X ([m-2]T) + \dots \\ & + a_n X ([m-n]T) - Y (mT) - b_1 Y ([m-1]T) \\ & - \dots - b_n Y ([m-n]T)]^2 \end{aligned} \quad (9)$$

H must now be minimized by finding the best set of coefficients. This is done by taking partial derivatives of H with respect to the $2n$ unknown coefficients and setting these derivatives equal to zero. This

gives $2n$ simultaneous equations in $2n$ unknowns which can be solved to find the set of coefficients which best fit the input and output data. By summing a fairly large number of equations to get the function H , the effects of random noise are greatly reduced.

After cancellation and simplification, the equations are as shown in Equation (10), page 18. The summation terms in the matrices may be considered as good approximations to correlation functions if a large number of terms are summed, that is, if i is a large number. This is apparent if the summation takes place over a large enough number of terms so as to approximate the usual integration in the correlation function. In that case, the equations may be rewritten in terms of correlation functions as shown in Equation (11), page 22, with the computation being simplified. This then reduces to the technique discussed by Tou (7). If a relatively small number of terms are summed, significant errors may result when Equation (11) is used. For example, the difference between $\sum_{m=1}^i X^2 ([m-1]T)$ and $\sum_{m=1}^i X^2 ([m-2]T)$, may be significant even though both of these terms have been replaced by $\phi_{xx}(0)$ in the correlation function equations. Since much emphasis must be placed on the speed with which the identification can be performed, it is desirable to make i as small as possible. Therefore, in the work done in this investigation the technique has not used correlation functions but the least squares fit for the input and output data.

$$\begin{array}{l}
 \sum_{m=1}^1 X[(m-1)T]Y[mT] \\
 \sum_{m=1}^1 X[(m-2)T]Y[mT] \\
 \sum_{m=1}^1 X[(m-3)T]Y[mT] \\
 \vdots \\
 \sum_{m=1}^1 X[(m-n)T]Y[mT] \\
 \vdots \\
 \sum_{m=1}^1 Y[(m-2)T]Y[mT] \\
 \vdots \\
 \sum_{m=1}^1 Y[(m-n)T]Y[mT]
 \end{array}
 =
 \begin{array}{l}
 \sum_{m=1}^1 X^2[(m-1)T] \\
 \sum_{m=1}^1 X[(m-1)T]X[(m-2)T] \\
 \sum_{m=1}^1 X[(m-1)T]X[(m-3)T] \\
 \vdots \\
 \sum_{m=1}^1 X[(m-1)T]X[(m-n)T] \\
 \vdots \\
 \sum_{m=1}^1 X[(m-1)T]Y[(m-1)T] \\
 \sum_{m=1}^1 X[(m-1)T]Y[(m-2)T] \\
 \vdots \\
 \sum_{m=1}^1 X[(m-1)T]Y[(m-n)T]
 \end{array}$$

Equation 10

$$\begin{array}{rcl}
\sum_{m=1}^{\frac{1}{2}} X[(m-2)T]X[(m-1)T] & & \sum_{m=1}^{\frac{1}{2}} X[(m-3)T]X[(m-1)T] \quad \dots \\
\sum_{m=1}^{\frac{1}{2}} X^2[(m-2)T] & & \sum_{m=1}^{\frac{1}{2}} X[(m-3)T]X[(m-2)T] \quad \dots \\
\sum_{m=1}^{\frac{1}{2}} X[(m-2)T]X[(m-3)T] & & \sum_{m=1}^{\frac{1}{2}} X^2[(m-3)T] \quad \dots \\
\vdots & & \vdots \\
\sum_{m=1}^{\frac{1}{2}} X[(m-2)T]X[(m-n)T] & & \dots \\
\vdots & & \vdots \\
\sum_{m=1}^{\frac{1}{2}} X[(m-2)T]Y[(m-2)T] & & \dots \\
\vdots & & \vdots \\
\sum_{m=1}^{\frac{1}{2}} X[(m-2)T]Y[(m-n)T] & & \dots
\end{array}$$

Equation 10 (Continued)

$$\sum_{m=1}^{\frac{1}{2}} X[(m-n)T]X[(m-1)T]$$

$$\sum_{m=1}^{\frac{1}{2}} Y[(m-1)T]X[(m-1)T]$$

$$\sum_{m=1}^{\frac{1}{2}} X[(m-n)T]X[(m-2)T]$$

$$\sum_{m=1}^{\frac{1}{2}} Y[(m-1)T]X[(m-2)T]$$

•
•
•

•
•
•

$$\sum_{m=1}^{\frac{1}{2}} X^2[(m-n)T]$$

$$\sum_{m=1}^{\frac{1}{2}} Y[(m-1)T]X[(m-n)T]$$

$$\sum_{m=1}^{\frac{1}{2}} X[(m-n)T]Y[(m-1)T]$$

$$\sum_{m=1}^{\frac{1}{2}} Y^2[(m-1)T]$$

$$\sum_{m=1}^{\frac{1}{2}} X[(m-n)T]Y[(m-2)T]$$

$$\sum_{m=1}^{\frac{1}{2}} Y[(m-1)T]Y[(m-2)T]$$

$$\sum_{m=1}^{\frac{1}{2}} X[(m-n)T]Y[(m-n)T]$$

$$\sum_{m=1}^{\frac{1}{2}} Y[(m-1)T]Y[(m-n)T]$$

Equation 10 (Continued)

$$\begin{array}{l}
 \left[\begin{array}{l}
 \sum_{m=1}^1 Y[(m-2)F]X[(m-1)F] \quad \dots \quad \sum_{m=1}^1 Y[(m-n)F]X[(m-1)F] \\
 \sum_{m=1}^1 Y[(m-2)F]X[(m-2)F] \quad \dots \quad \sum_{m=1}^1 Y[(m-n)F]X[(m-2)F] \\
 \sum_{m=1}^1 Y[(m-2)F]X[(m-n)F] \quad \dots \quad \sum_{m=1}^1 Y[(m-n)F]X[(m-n)F] \\
 \sum_{m=1}^1 Y^2[(m-2)F] \quad \dots \quad \sum_{m=1}^1 Y[(m-n)F]Y[(m-1)F] \\
 \sum_{m=1}^1 Y^2[(m-2)F] \quad \dots \quad \cdot \\
 \sum_{m=1}^1 Y[(m-2)F]Y[(m-n)F] \quad \dots \quad \sum_{m=1}^1 Y^2[(m-n)F] \\
 \sum_{m=1}^1 Y^2[(m-2)F]
 \end{array} \right]
 \end{array}
 \times
 \begin{array}{l}
 a_1 \\
 a_2 \\
 \cdot \\
 \cdot \\
 \cdot \\
 a_n \\
 -b_1 \\
 -b_2 \\
 \cdot \\
 \cdot \\
 -b_n
 \end{array}$$

Equation 10 (Continued)

$$\begin{array}{c}
 \phi_{xy}[T] \\
 \phi_{xy}[2T] \\
 \phi_{xy}[3T] \\
 \vdots \\
 \phi_{xy}[nT] \\
 \phi_{yy}[T] \\
 \phi_{yy}[2T] \\
 \vdots \\
 \phi_{yy}[nT]
 \end{array}
 =
 \begin{array}{c}
 \phi_{xx}[0] \\
 \phi_{xx}[T] \\
 \phi_{xx}[2T] \\
 \vdots \\
 \phi_{xx}[(n-1)T] \\
 \phi_{xy}[0] \\
 \phi_{xy}[-T] \\
 \vdots \\
 \phi_{xy}[-(n-1)T]
 \end{array}
 \begin{array}{c}
 \phi_{xx}[T] \\
 \phi_{xx}[0] \\
 \phi_{xx}[T] \\
 \vdots \\
 \phi_{xx}[(n-2)T] \\
 \phi_{xy}[T] \\
 \phi_{xy}[0] \\
 \vdots \\
 \phi_{xy}[-(n-2)T]
 \end{array}$$

Equation 11

$$\begin{array}{ccc}
 \phi_{xx}[2T] & \dots & \phi_{xx}[(m-1)T] & \phi_{xy}[0] \\
 \\
 \phi_{xy}[T] & & \phi_{xx}[(n-2)T] & \phi_{xy}[+T] \\
 \\
 & & \vdots & \vdots \\
 \phi_{xx}[0] & & \vdots & \vdots \\
 \\
 \vdots & & & \\
 \\
 \phi_{xx}[(n-3)T] & & \phi_{xx}[0] & \phi_{xy}[(n-1)T] \\
 \\
 \vdots & & \phi_{xy}[(n-1)T] & \phi_{yy}[0] \\
 \\
 & & \vdots & \vdots \\
 \\
 & & \phi_{xy}[0] & \phi_{yy}[(n-1)T]
 \end{array}$$

Equation 11 (Continued)

$\phi_{xy}[-T]$	$\phi_{xy}[-(n-1)T]$	x	a_1
$\phi_{xy}[0]$...		$\phi_{xy}[-(n-2)T]$		a_2
.			.		a_3
.			.		.
.			.		.
$\phi_{xy}[+(n-2)T]$			$\phi_{xy}[0]$		a_n
$\phi_{yy}[T]$			$\phi_{yy}[(n-1)T]$		$-b_1$
.			.		$-b_2$
.			.		.
$\phi_{yy}[(n-2)T]$			$\phi_{yy}[0]$		$-b_n$

Equation 11 (Continued)

II. EXPERIMENTAL INVESTIGATION OF SAMPLED- DATA PROCESS IDENTIFICATION

An experimental investigation of the least-squares process identification technique described earlier was undertaken as part of this project. The preliminary work was done on a second order system as shown in Figure 7 in order to simplify the computation. The initial investigation was carried out without any feedback, but with noise added to the output. The results of the identification were then compared with the results obtained by simply taking $2n$ sets of data and solving for the $2n$ unknown coefficients. The results are shown in Figures 8a and 8b and Figures 9a and 9b. Ten identifications were made for each technique for each of two noise levels. The figures show that the least square technique gives substantially better accuracy than the solution of the set of $2n$ equations even though the noise level was from five to ten times as great. The least squares results were obtained by summing twenty equations to form the matrices. (See Appendix B.)

The next step in the experimental investigation was to determine how well the technique would work if there was a closed feedback loop and noise was allowed to circulate around the loop. The closed loop system shown in Figure 10 was simulated digitally. (See Appendix A.) The input to the closed loop was supplied by a random number generator. The noise, which was added to the output and fed back around the loop, was also supplied by a random number generator. It was found that this configuration, i.e. with the added noise circulating around the loop,

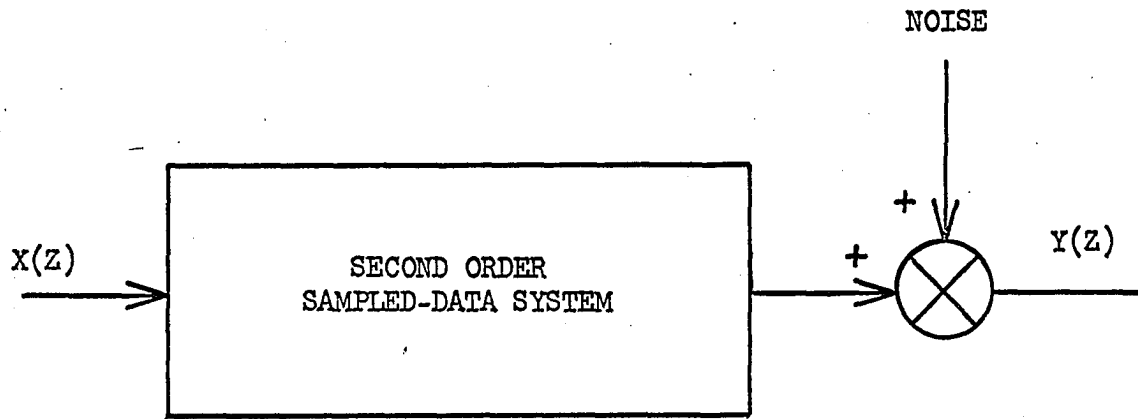


Figure 7. System used in initial process identification work

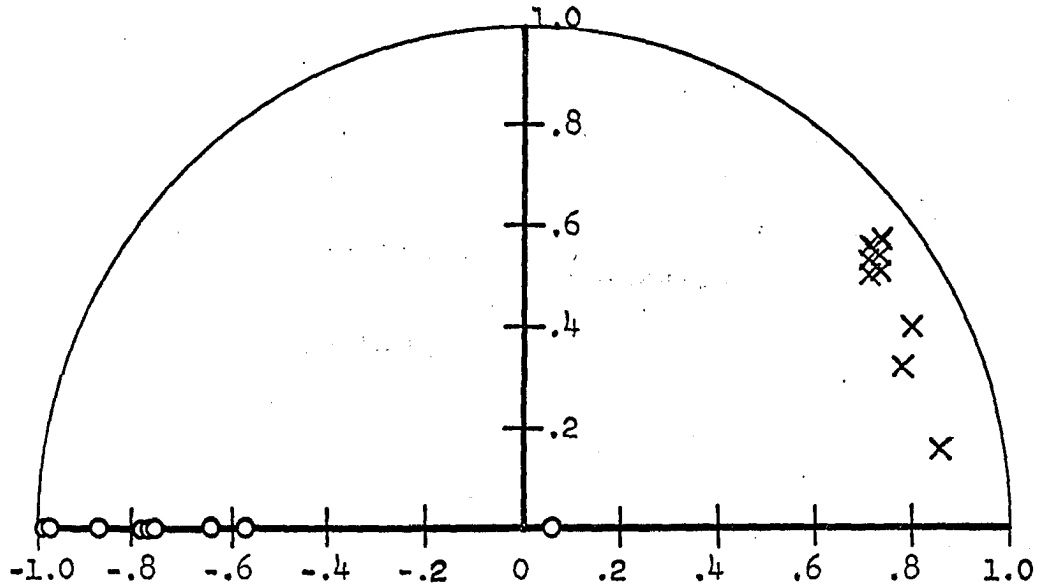


Figure 8a. Process identification results using $2n$ equations
(noise - 2% of input)

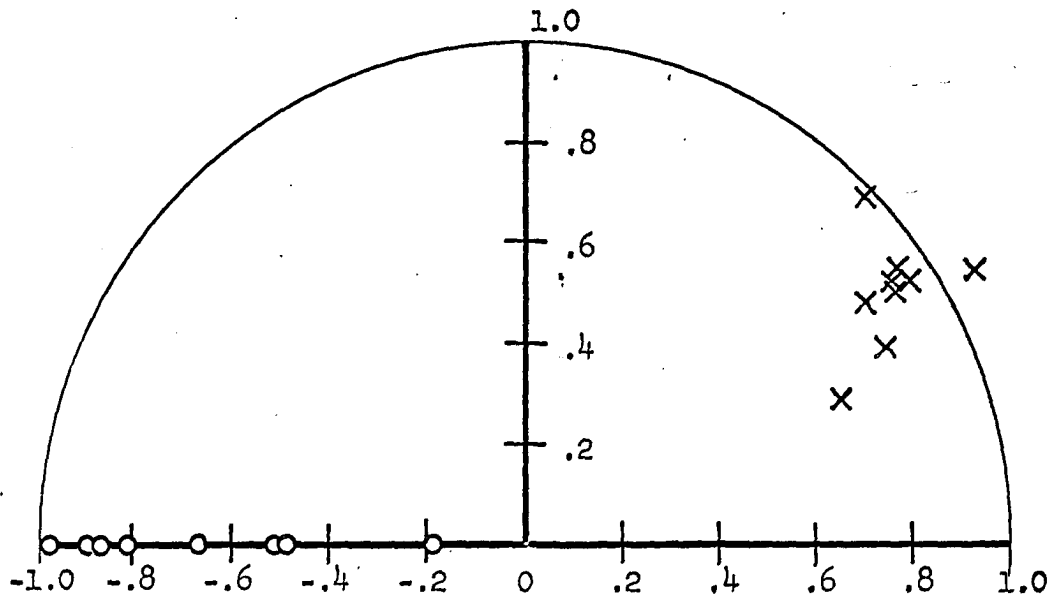


Figure 8b. Process identification results using $2n$ equations
(noise - 5% of input Note: one pair of poles
off scale at $1.81 \pm j1.2$)

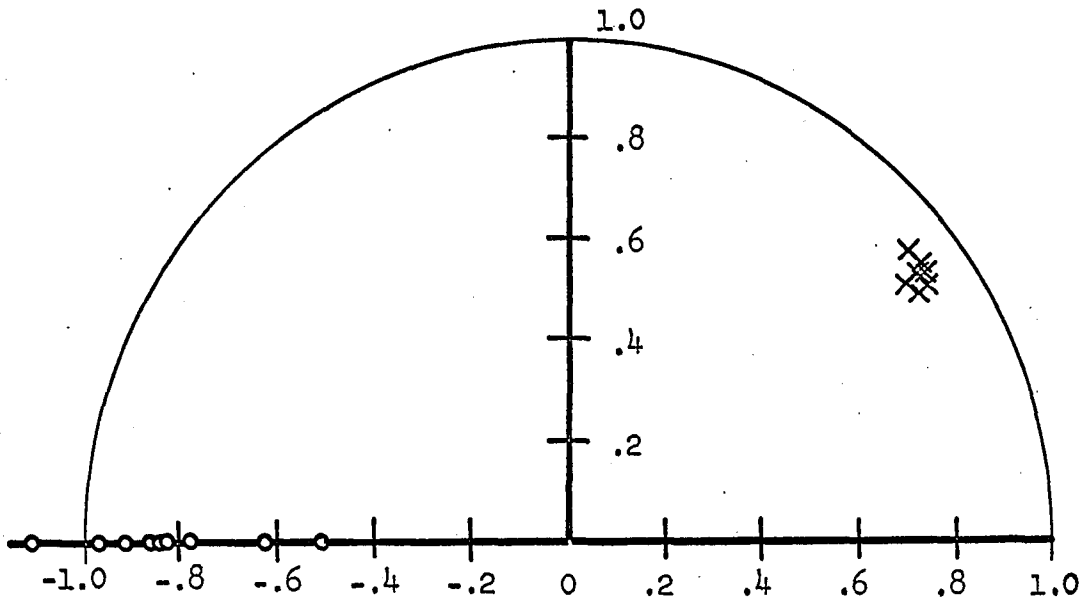


Figure 9a. Process identification results using least squares technique (noise - 5% of input)

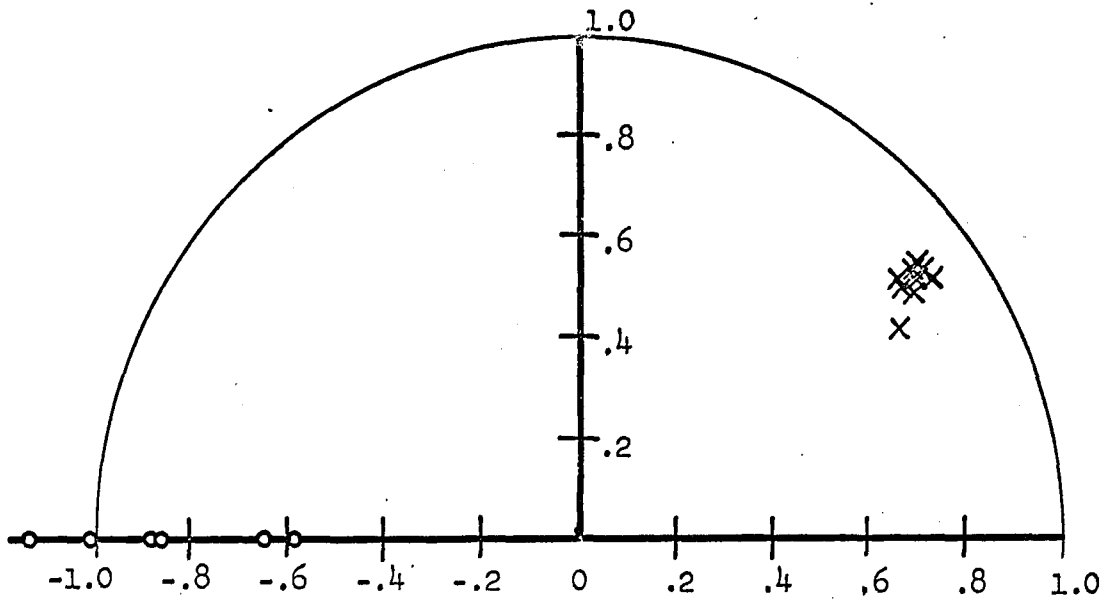


Figure 9b. Process identification results using least squares technique (noise - 10% of input)

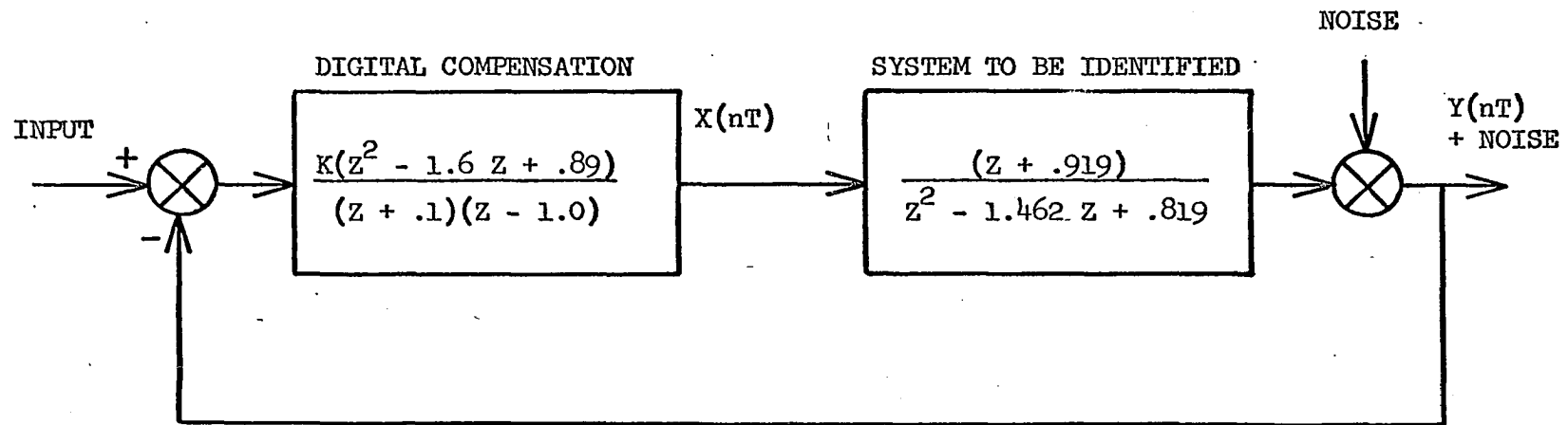


Figure 10. Closed loop system used in process identification work

gave the greatest errors in the identification.

Data were obtained for twelve different pole locations in the Z-plane. Figures 11 through 34 show the results of the identification runs. In each case two different noise levels were used, and twenty equations were summed in the matrices for each run. Ten runs were made for each pole position and noise level.

From these figures it appears that fair accuracy can be obtained for reasonably modest noise levels if the system is lightly damped. This is indicated by the amount of scattering of the solutions in the Z-plane. The scattering is considerably reduced for pole locations near the unit circle. This is true except for the systems described in Figures 29 through 34 in which the pole is approaching the negative real axis in the Z-plane. The large amount of scattering in these cases is not too surprising considering that the poles on the negative real axis correspond to a system with a natural frequency of one-half the sampling frequency.

The conclusion drawn from these data is that the identification process will give reasonable results for modest amounts of noise if the poles are near the unit circle, but not when the frequency is close to the half sampling frequency. Since the body bending response of a missile is very lightly damped, and since the sampling frequency can be selected at will, the technique appears to have some merit for the adaptive control of flexible airframes.

The increased accuracy for poles near the unit circle can be explained in terms of the signal-to-noise ratio in the output. For a

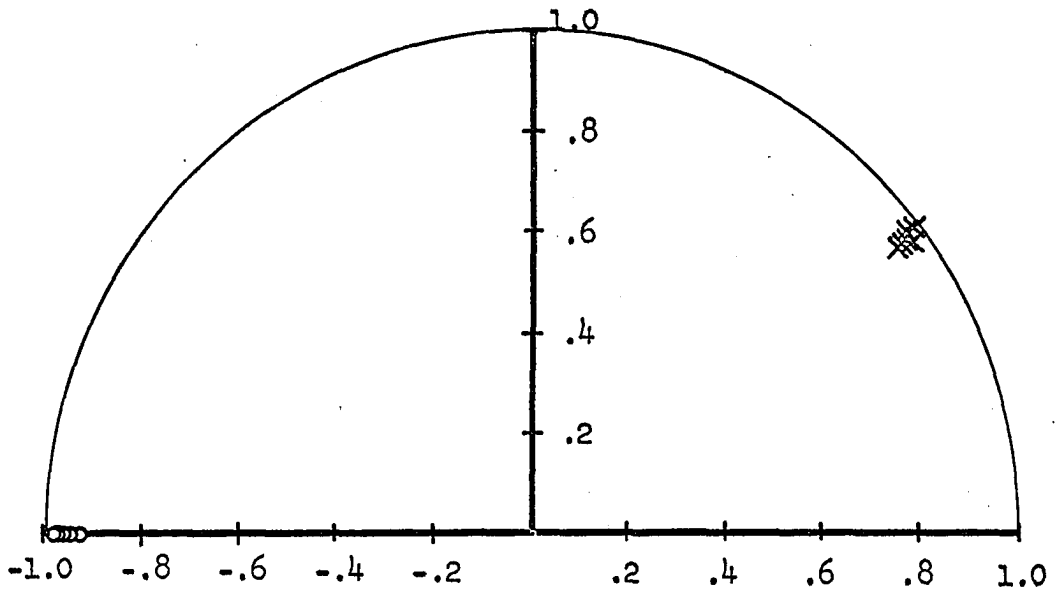


Figure 11. Process identification results
(noise level - 5% of input)

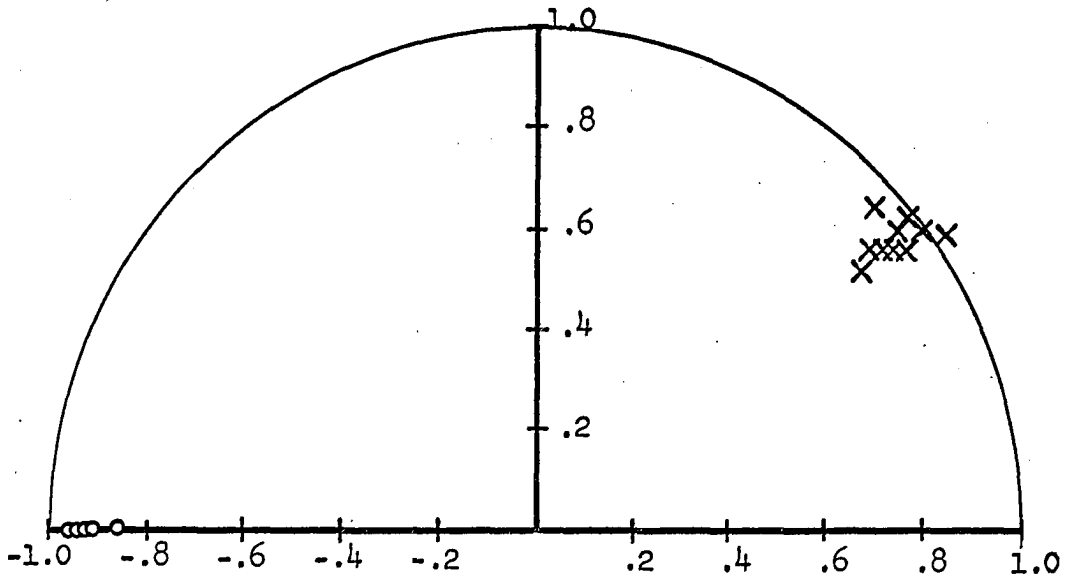


Figure 12. Process identification results
(noise level - 10% of input)

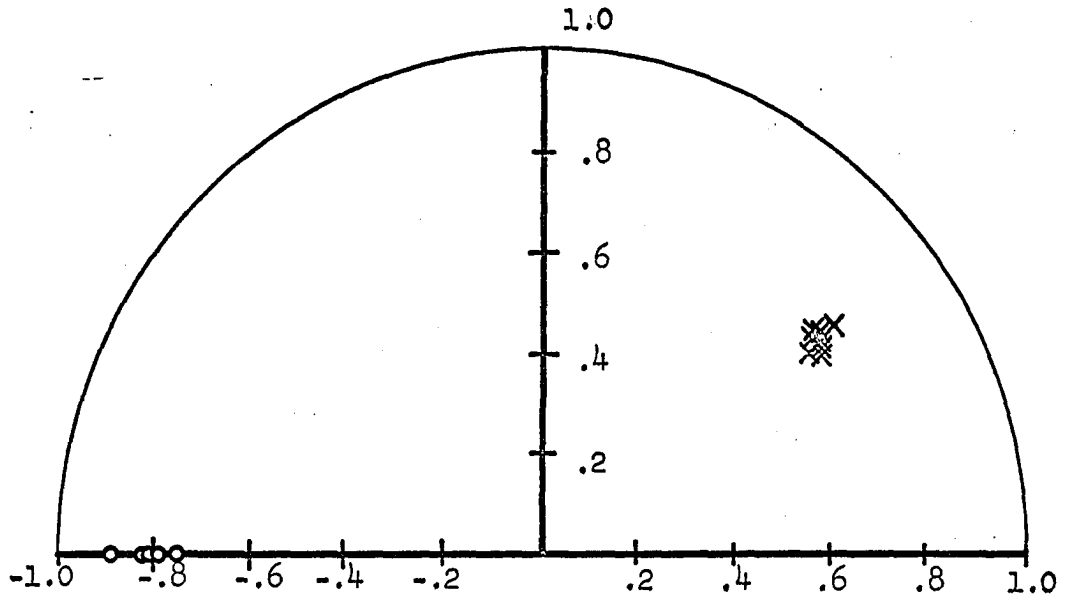


Figure 13. Process identification results
(noise level - 5% of input)

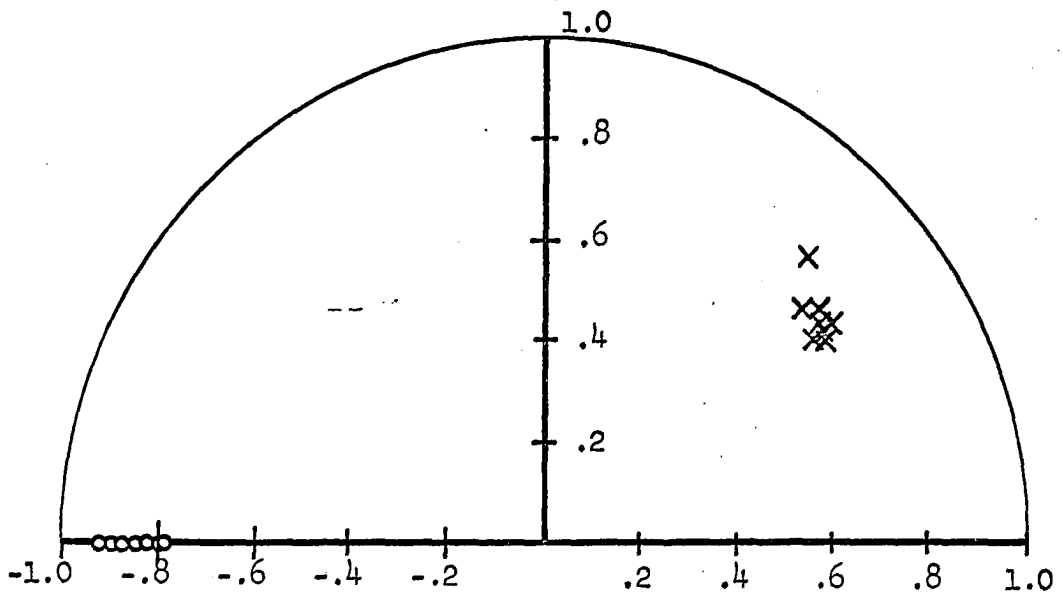


Figure 14. Process identification results
(noise level - 10% of input)

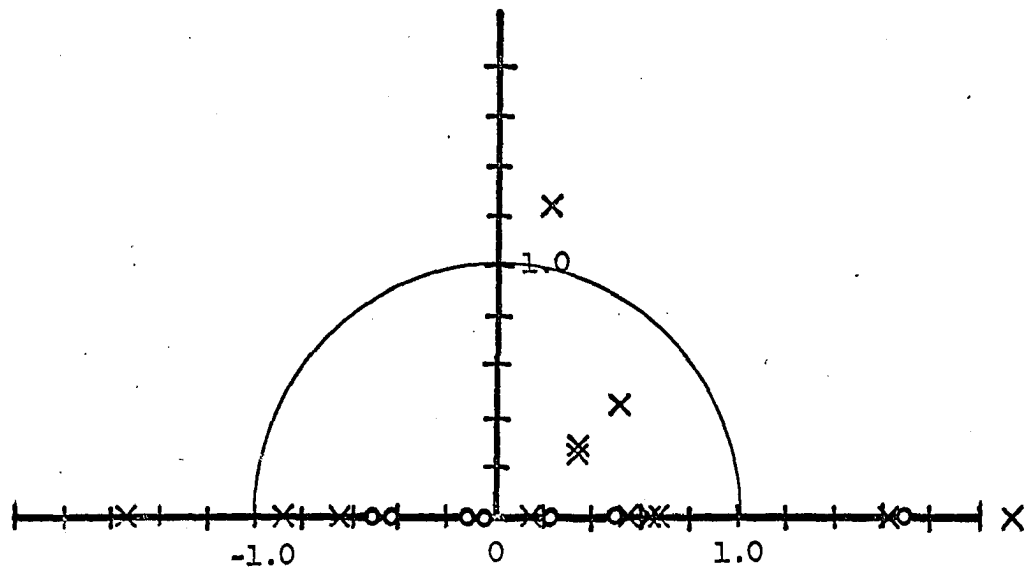


Figure 15. Process identification results
 poles at $.25 + j .20$ (noise
 level - 5% of input Note: poles
 off scale at 3.3, 3.85 and 20.94)

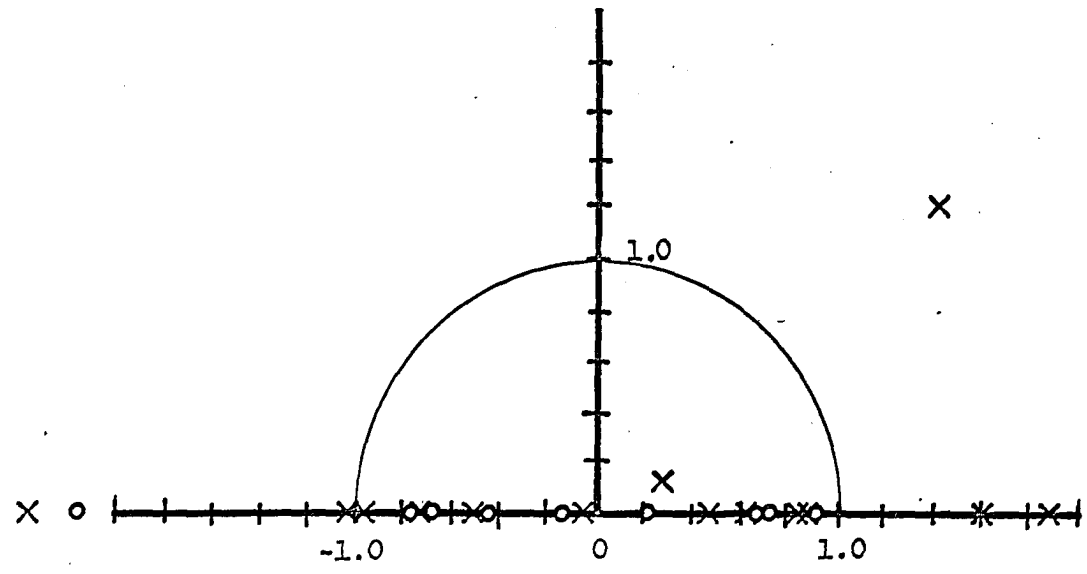


Figure 16. Process identification results
 poles at $0.25 + j .20$ (noise
 level - 10% of input)

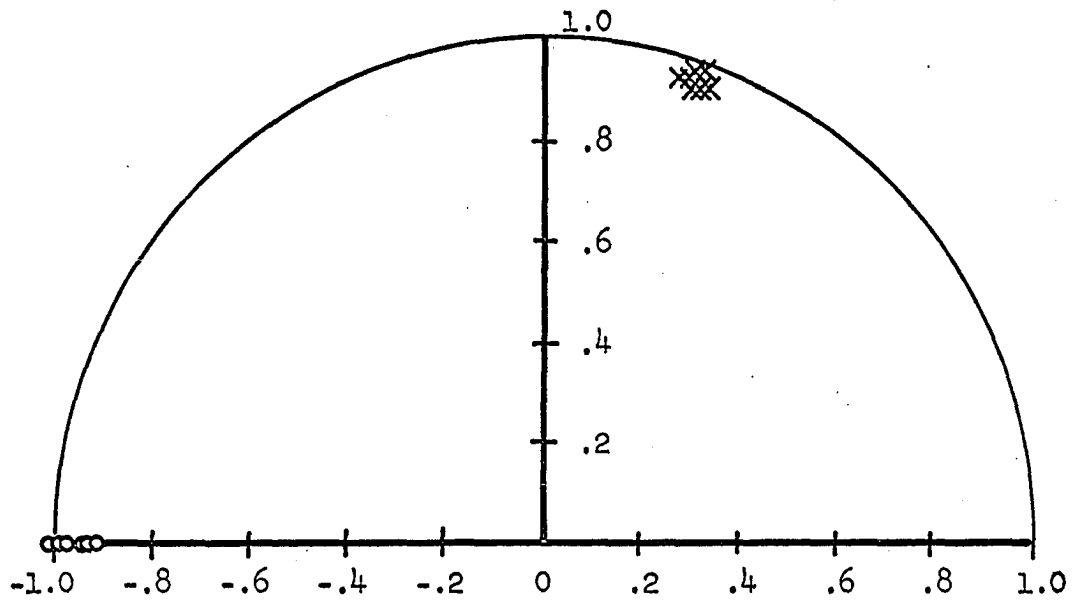


Figure 17. Process identification results
poles at 0.3 ± 0.925 (noise
level - 5% of input)

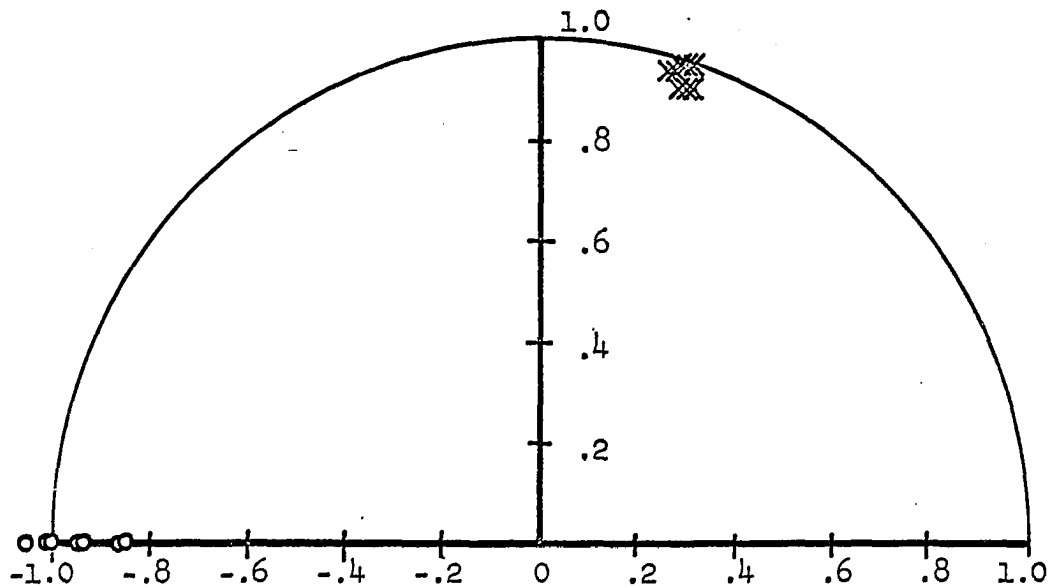


Figure 18. Process identification results
poles at $0.3 \pm j 0.925$ (noise
level - 10% of input)

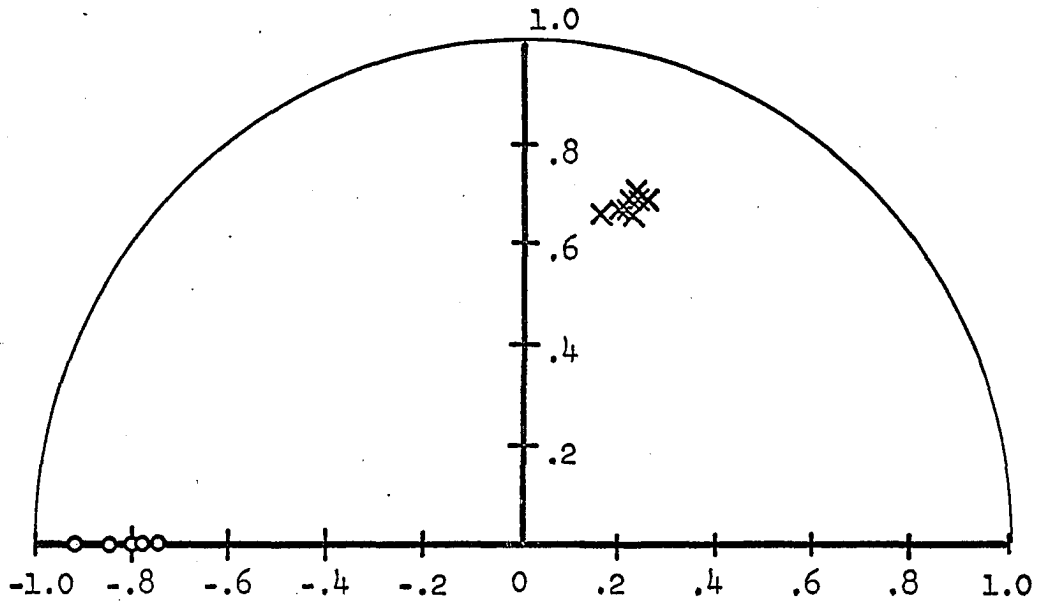


Figure 19. Process identification results
poles at $0.25 \pm j .7$ (noise
level - 5% of input)

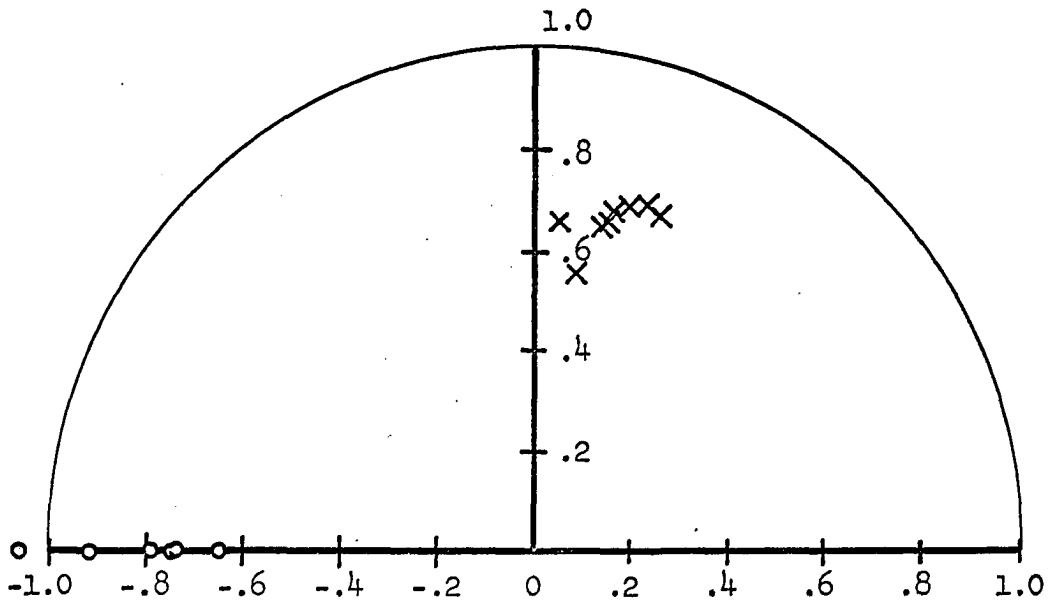


Figure 20. Process identification results
poles at $0.25 \pm j 0.7$ (noise
level - 10% of input)

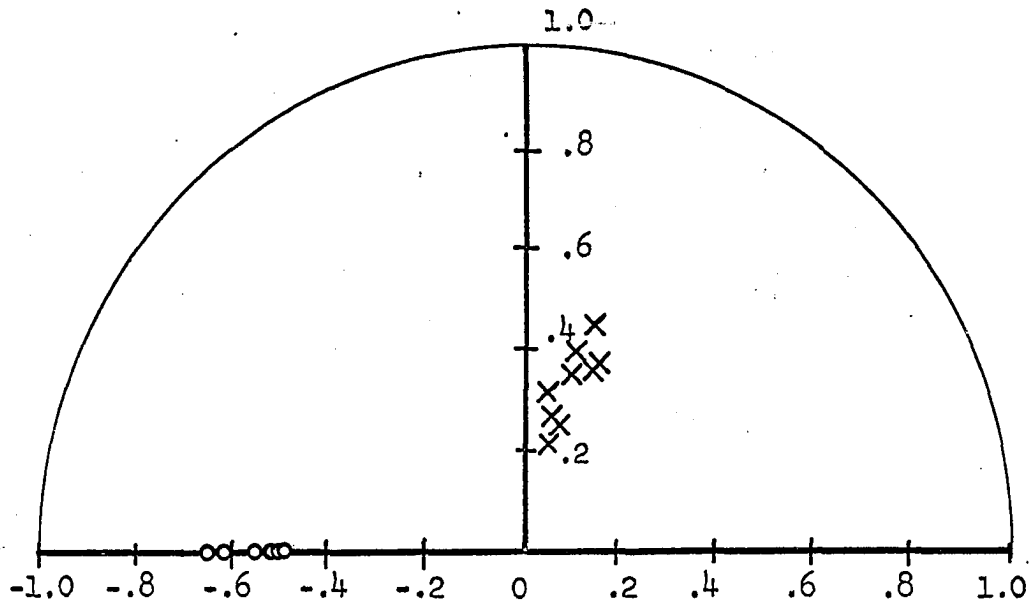


Figure 21. Process identification results
poles at $0.1 + j 0.3$ (noise
level - 5% of input)

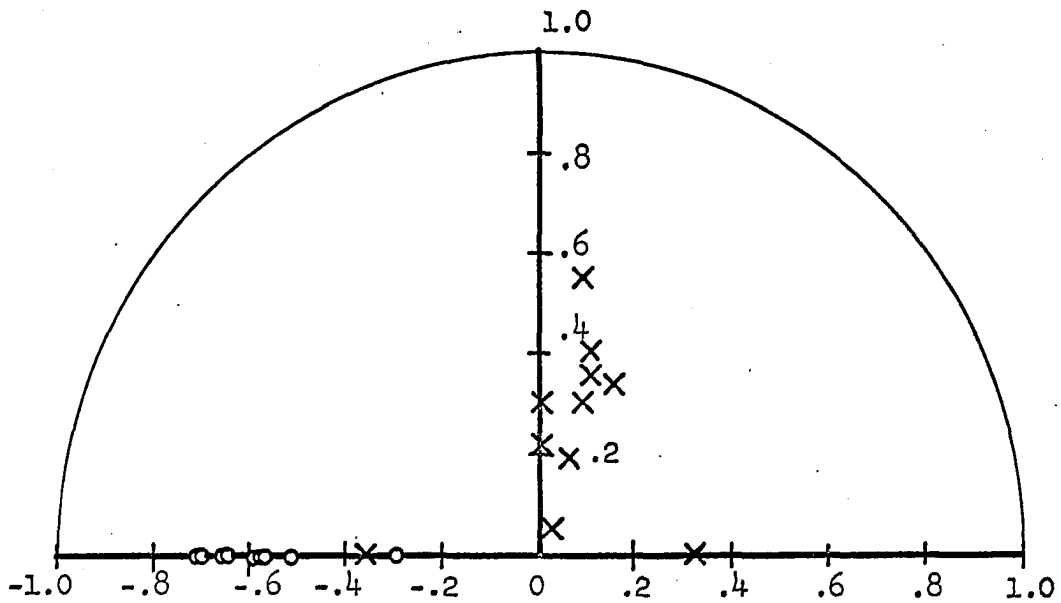


Figure 22. Process identification results
poles at $0.1 + j 0.3$ (noise
level - 10% of input)

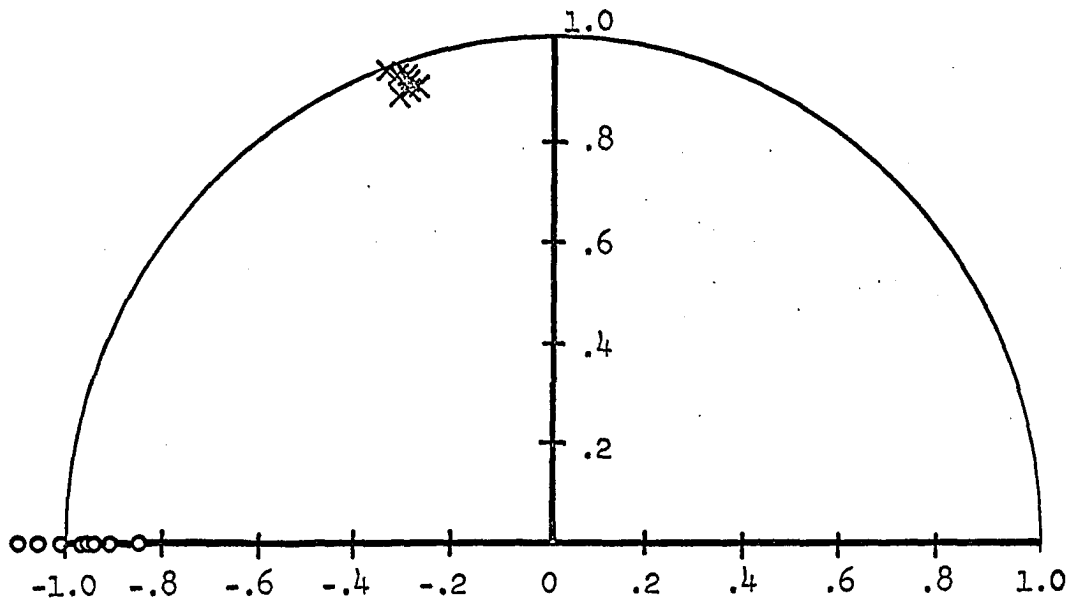


Figure 23. Process identification results
poles at $-0.3 \pm j 0.975$ (noise
level - 5% of input)

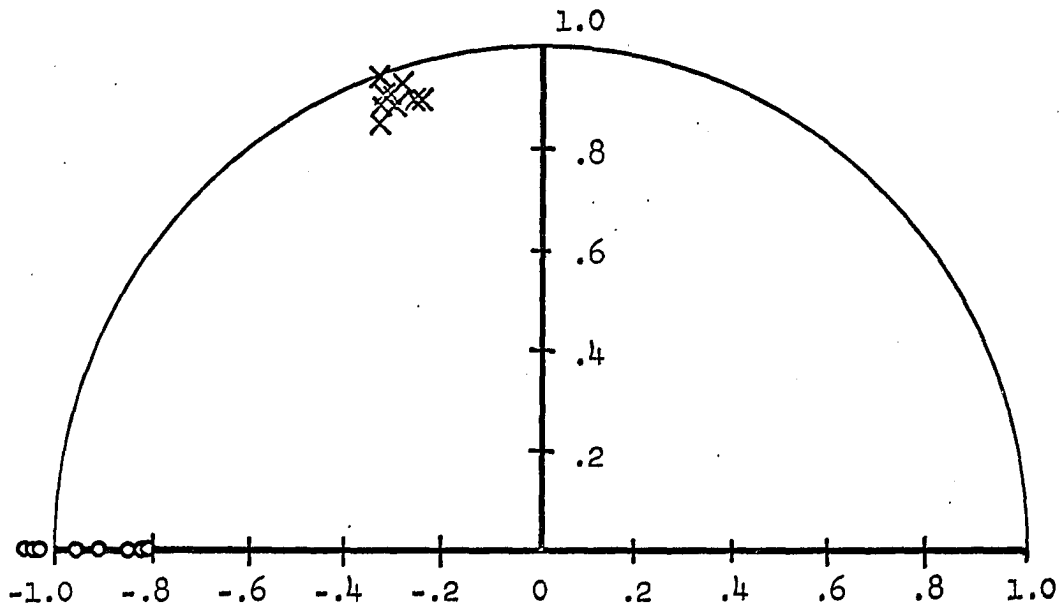


Figure 24. Process identification results
poles at $-0.3 \pm j 0.975$
(noise level - 10% of input)

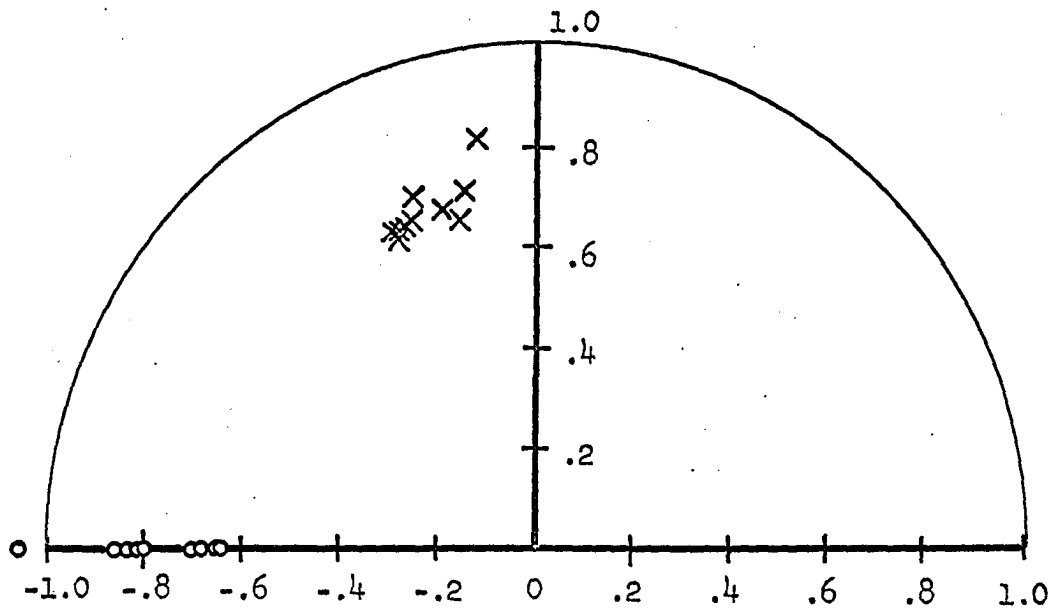


Figure 25. Process identification results
poles at $-0.25 + j 0.7$ (noise
level - 5% of input)

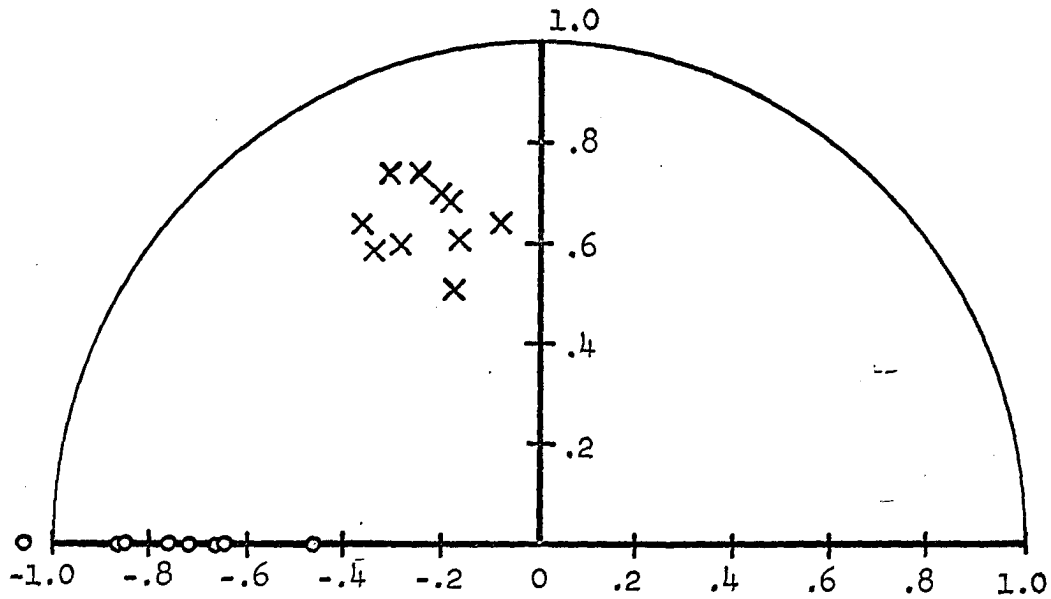


Figure 26. Process identification results
poles at $-0.25 + j 0.7$ (noise
level - 10% of input)

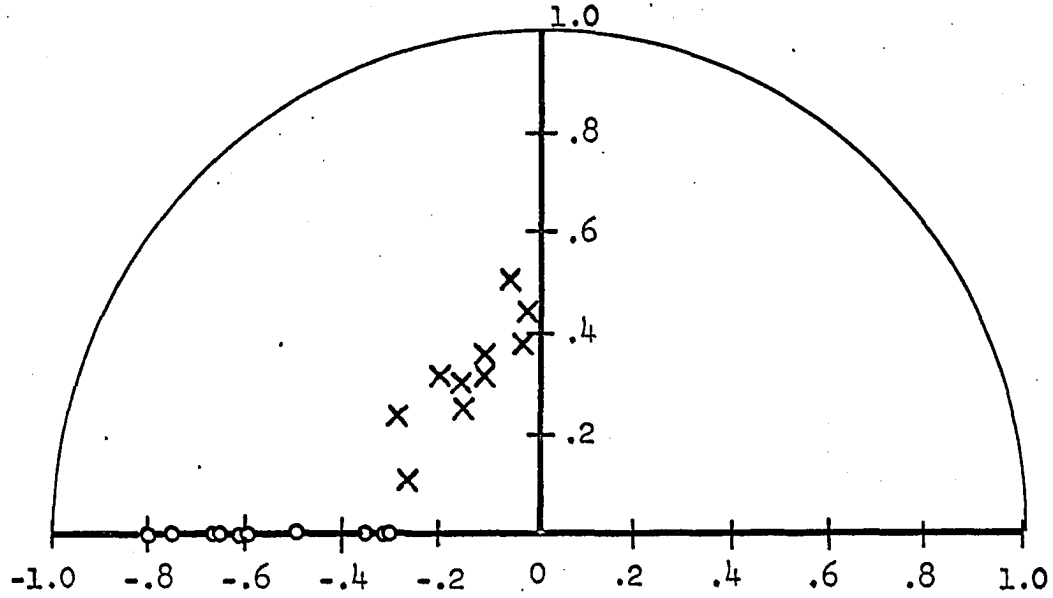


Figure 27. Process identification results
poles at $-0.15 + j 0.25$
(noise level - 5% of input)

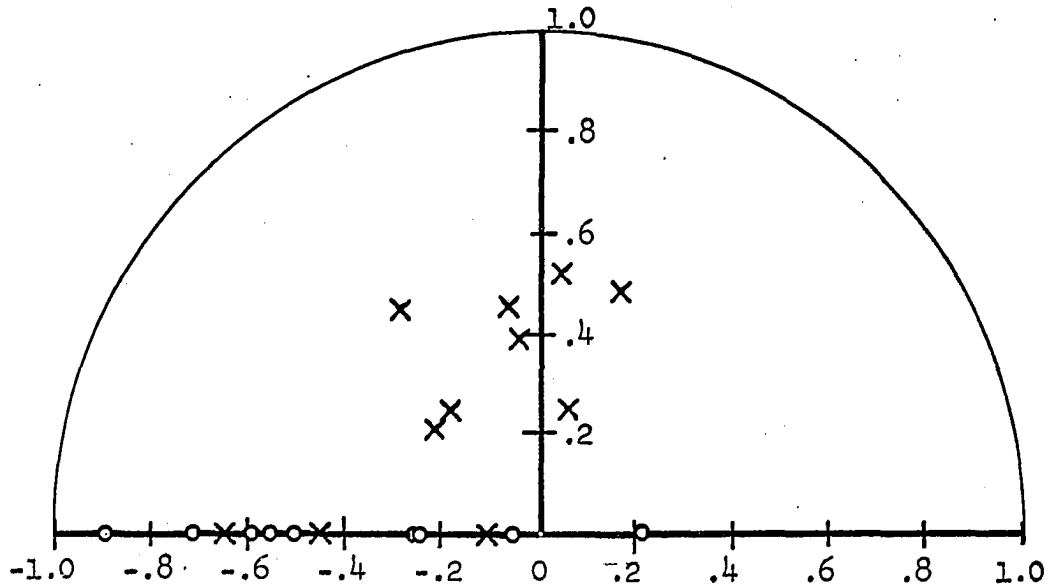


Figure 28. Process identification results
poles at $-0.15 + j 0.25$ (noise
level - 10% of input)

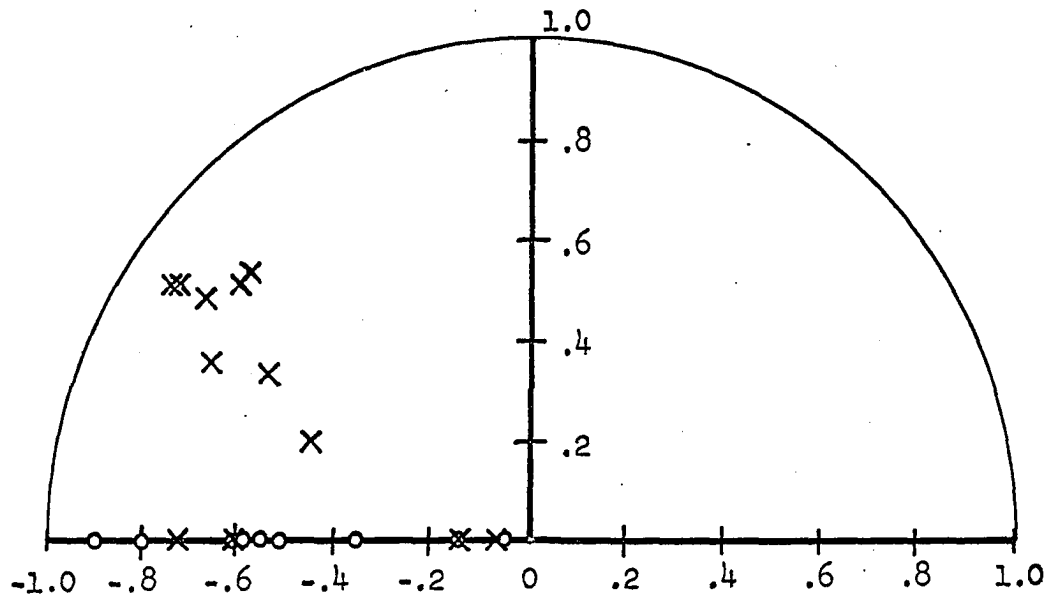


Figure 29. Process identification results
poles at $-0.8 + j .55$ (noise
level - 5% of input)

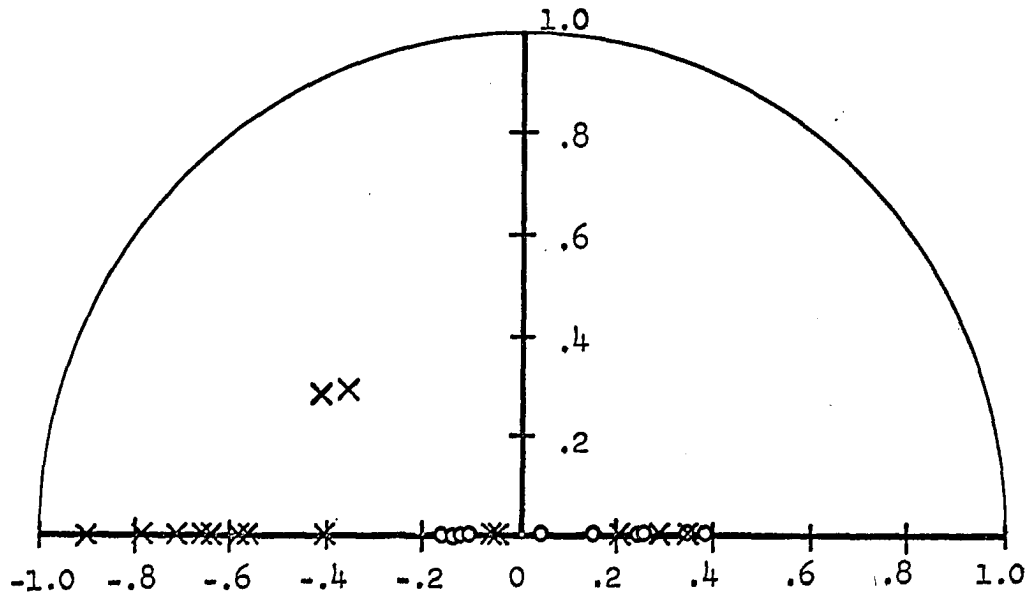


Figure 30. Process identification results
poles at $-0.8 + j .55$ (noise
level - 10% of input)

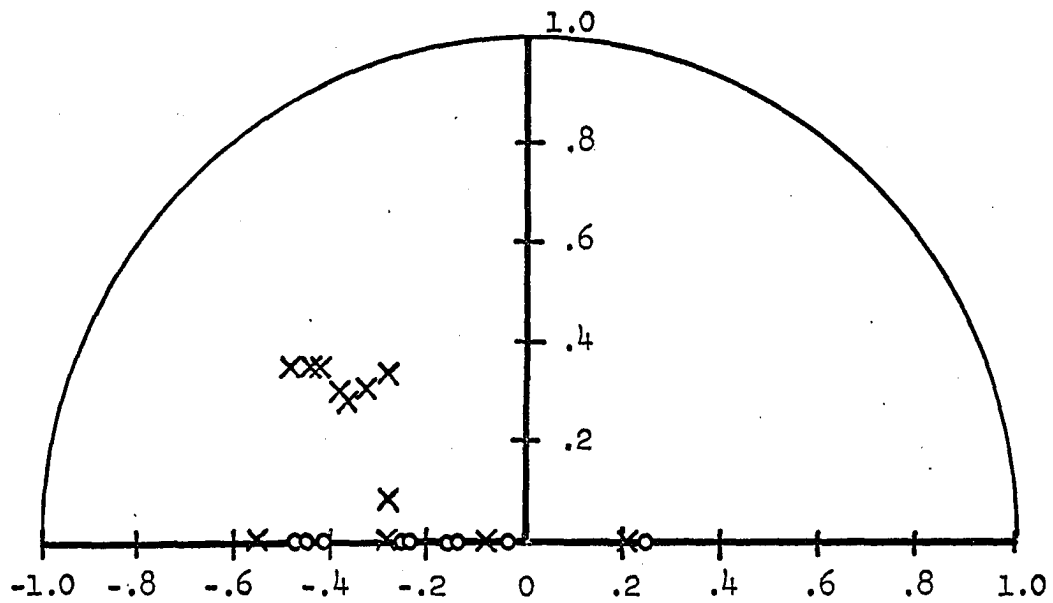


Figure 31. Process identification results
poles at $-0.5 + j 0.3$ (noise
level - 5% of input)

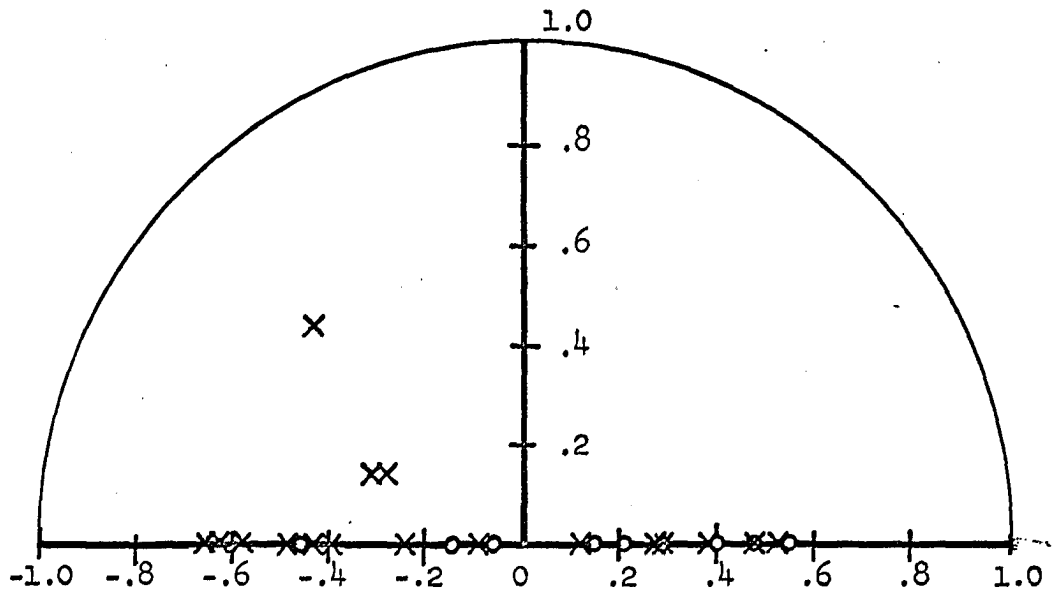


Figure 32. Process identification results
poles at $-0.5 + j 0.3$ (noise
level - 10% of input)

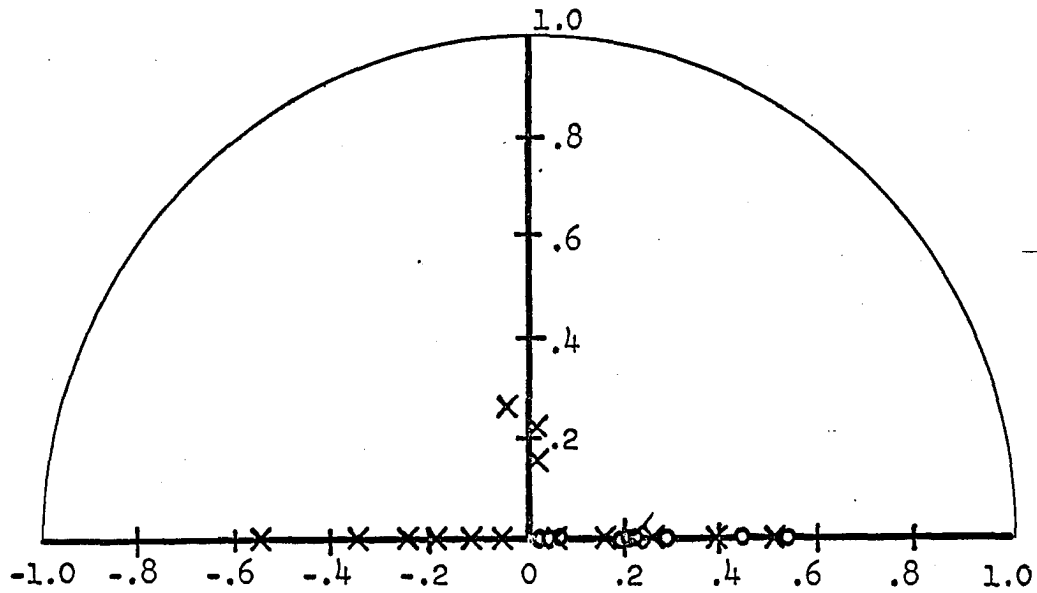


Figure 33. Process identification results
poles at $-0.25 \pm j 0.15$ (noise
level - 5% of input)

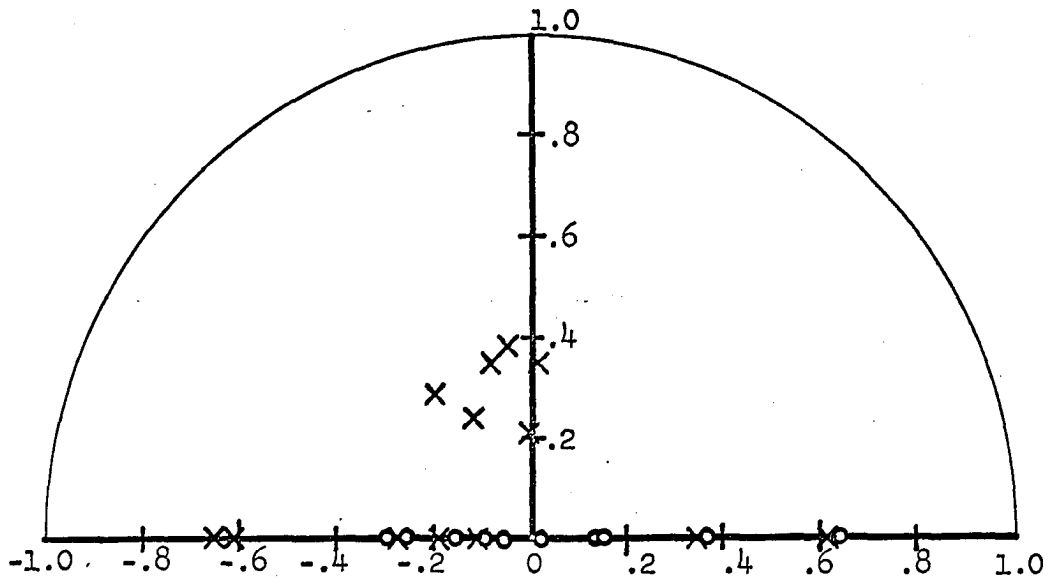


Figure 34. Process identification results
poles at $-0.25 \pm j 0.15$ (noise
level - 10% of input)

lightly damped system the output signal at the natural frequency will be greater than that for a heavily damped system. Since the noise that is introduced is evenly distributed over the frequency range, the signal-to-noise ratio at the natural frequency will be greater for the lightly damped system.

To give a more realistic assessment of the value of this identification method, the Z-transforms of the transfer functions of a typical missile were obtained (8). These transfer functions are not applicable to a specific missile but might be considered as typical of a medium range research missile. These transfer functions were used in an investigation carried out by Minneapolis-Honeywell Aeronautical Systems Division, and include the effect of the Honeywell "gyro-blender". To eliminate the effects of the gyro-blender it was necessary to change the positions of the zeros associated with the bending modes. The positions chosen for the zeros would be more typical for a fixed gyroscope location. In order to emphasize the problem of changing bending frequencies, the poles associated with the bending modes were also altered. This last change is such as to make the control problem more difficult, but the adaptive system described here still gives good results. Finally, the third bending mode was dropped in order to simplify the computation and conserve digital computer time in the investigation.

This transfer function, along with the compensation actually used in the investigation of the adaptive technique (see Section III), was simulated as described in Appendix A. The results of the identification

runs are shown in Figures 35 through 38. The accuracy obtained for these cases is considerably reduced for the first bending mode but the pole location for the second bending mode was located very accurately in all cases. This again can be explained in terms of the signal-to-noise ratio in the output. The first bending mode is accompanied by a zero in the airframe transfer function which is a result of the location of the gyroscope near the bending node. As a result, the residue of this pole is smaller than that of the second bending mode and the output signal at that natural frequency is reduced. This results in a smaller signal-to-noise ratio in the output at the first bending mode frequency.

Figures 35 through 38 also show the increased accuracy that can be obtained by summing a larger number of terms in the matrices.

From these data it appears that reasonable accuracy can be obtained if a large number of equations is summed and if the gyroscope location is selected with more care.

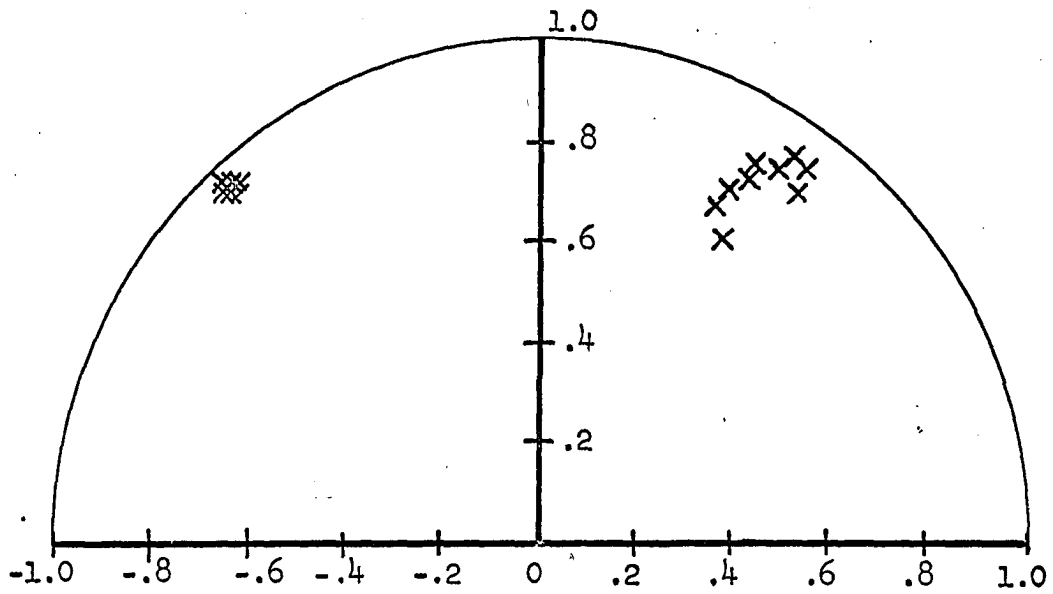


Figure 35. Process identification results 60 equations summed for each matrix (noise level - 2.0% of input)

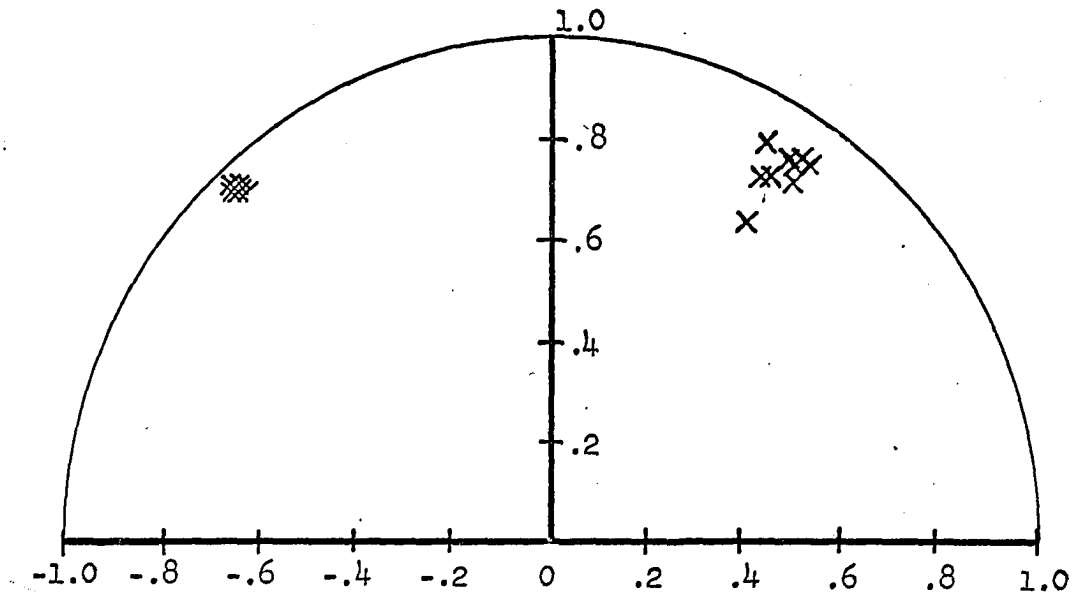


Figure 36. Process identification results 100 equations summed for each matrix (noise level - 2.0% of input)

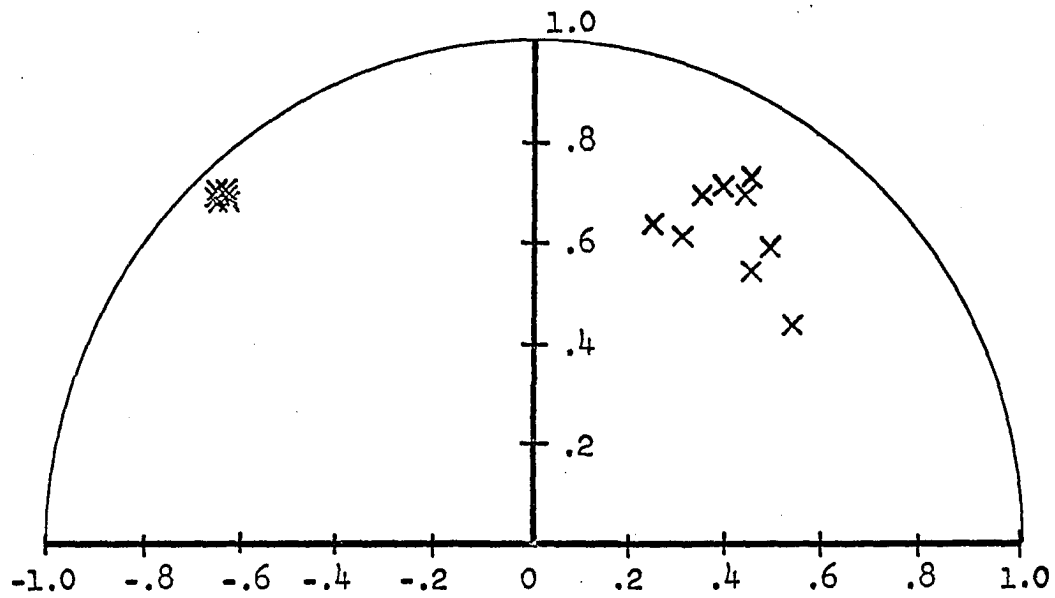


Figure 37. Process identification results 60 equations summed for each matrix (noise level 5% of input)

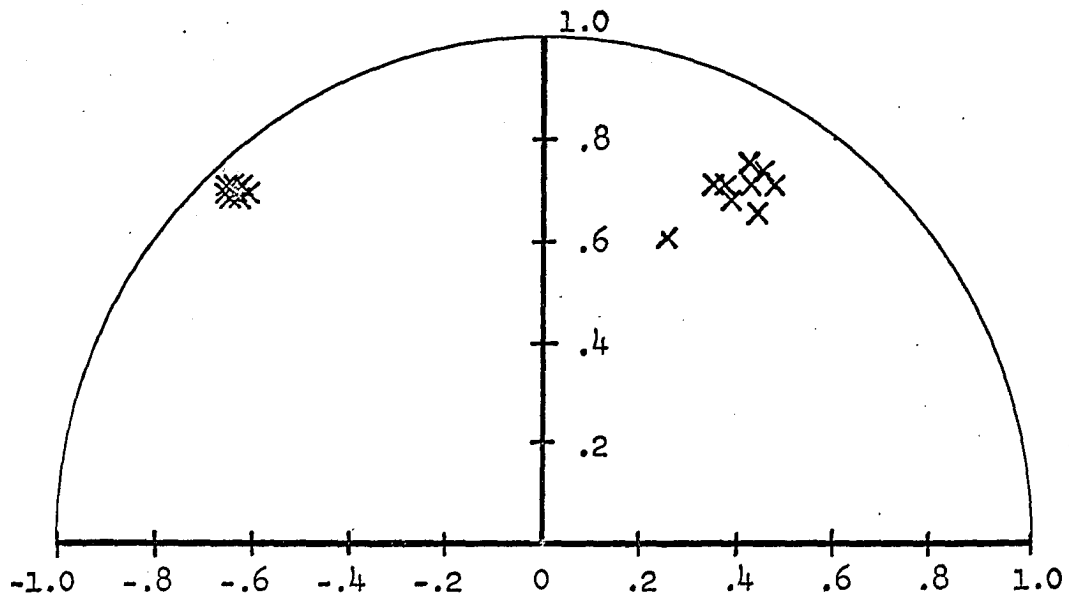


Figure 38. Process identification results 100 equations summed for each matrix (noise level - 5% of input)

III. VERIFICATION OF ADAPTATION BY VARIATION OF
SAMPLING RATE

To verify the adaptive characteristics of this control system, transfer functions for three flight conditions of a typical flexible missile were used (8). The transfer functions in the s-plane are tabulated below.

Table 1. S-plane transfer functions

Flight Condition	Transfer Function
A (Launch)	$\frac{(s - 1 \pm j 13)(s + 4 \pm j 20)}{s(s + .2 \pm j 20)(s + .5 \pm j 45)}$
B (Maximum dynamic pressure)	$\frac{(s + .15)(s + .5 \pm j 30)(s + .02 \pm j 35)}{(s + .2 \pm j 2.0)(s + .3 \pm j 22)(s + .7 \pm j 54)}$
C (Burnout - maximum altitude and velocity)	$\frac{(s + .02)(s + .3 \pm j 20)(s + .8 \pm j 40)}{(s + .04 \pm j 1.4)(s + .25 \pm j 25)(s + .6 \pm j 60)}$

As stated in Part II, these transfer functions have been modified slightly to represent a more optimum gyroscope placement and to emphasize the change in bending mode frequency with flight conditions. In order to determine the effect of variation in sampling rate, the Z-transforms of these functions, preceded by a zero order hold function, were obtained for several sampling intervals.

The choice of sampling rates was dictated by two factors. First, it was desired to place the poles associated with the first bending mode in the most advantageous region of the Z-plane from the standpoint of identification accuracy. Second, it was necessary to have the poles move past the compensation zero as the sampling rate changed. In actual practice, pole locations in the Z-plane were chosen and from the natural frequency in the s-plane the appropriate sampling interval was calculated. These Z-transforms are tabulated below.

Table 2. Z-plane transfer functions

Flight Condition	Sampling Interval	Z-transfer Function
A	T = .0308	$\frac{(Z - .699 \pm j.507)(Z - .952 \pm j.429)}{(Z - 1)(Z - .181 \pm j.968)(Z - .811 \pm j.574)}$
A	T = .0363	$\frac{(Z - .608 \pm j.565)(Z - .925 \pm j.516)}{(Z - 1)(Z + .062 \pm j.98)(Z - .742 \pm j.659)}$
A	T = .041	$\frac{(Z - .521 \pm j.603)(Z - .895 \pm j.596)}{(Z - 1)(Z + .265 \pm j.943)(Z - .677 \pm j.725)}$
A	T = .0473	$\frac{(Z - .388 \pm j.634)(Z - .844 \pm j.712)}{(Z - 1)(Z + .517 \pm j.829)(Z - .579 \pm j.803)}$
A	T = .0605	$\frac{(Z - .0166 \pm j.579)(Z - .657 \pm j.992)}{(Z - 1)(Z + .886 \pm j.395)(Z - .349 \pm j.924)}$
B	T = .028	$\frac{(Z - .996)(Z - .692 \pm j.708)}{(Z - .993 \pm j.056)(Z - .058 \pm j.979)}$
		$\times \frac{(Z - .470 \pm j.876)}{(Z - .809 \pm j.573)}$

Table 2 (Continued)

Flight Condition	Sampling Interval	Z-transfer Function
B	T = .0357	$\frac{(Z - .995)(Z - .538 \pm j.828)}{(Z - .990 \pm j.071)(Z + .341 \pm j.914)}$ $\times \frac{(Z - .128 \pm j.981)}{(Z - .700 \pm j.700)}$
B	T = .043	$\frac{(Z - .993)(Z - .381 \pm j.910)}{(Z - .990 \pm j.0852)(Z + .662 \pm j.709)}$ $\times \frac{(Z + .252 \pm j.952)}{(Z - .577 \pm j.801)}$
B	T = .055	$\frac{(Z - .992)(Z + .852 \pm j.470)}{(Z - .983 \pm j.109)(Z + .948 \pm j.164)}$ $\times \frac{(Z - .115 \pm j.976)}{(Z - .347 \pm j.920)}$
C	T = .022	$\frac{(Z - .998)(Z - .802 \pm j.584)}{(Z - .998 \pm j.0286)(Z - .848 \pm j.520)}$ $\times \frac{(Z - .534 \pm j.836)}{(Z - .138 \pm j.977)}$
C	T = .029	$\frac{(Z - .998)(Z - .673 \pm j.726)}{(Z - .997 \pm j.038)(Z - .743 \pm j.658)}$ $\times \frac{(Z - .188 \pm j.969)}{(Z + .304 \pm j.935)}$
C	T = .034	$\frac{(Z - .997)(Z - .568 \pm j.809)}{(Z - .997 \pm j.044)(Z - .654 \pm j.745)}$ $\times \frac{(Z + .109 \pm j.977)}{(Z + .585 \pm j.786)}$
C	T = .044	$\frac{(Z - .996)(Z + .739 \pm j.632)}{(Z - .995 \pm j.057)(Z - .449 \pm j.881)}$ $\times \frac{(Z - .337 \pm j.926)}{(Z + .936 \pm j.271)}$

The first pair of complex poles listed for each sampling rate for Flight Conditions B and C and the real zero in these same cases represent the rigid body dynamics. For Flight Condition A, the pole at $Z = 1$ represents the rigid body response. The two pairs of complex zeros and the two pairs of complex poles represent the first two bending modes. Notice that the pole positions corresponding to the rigid body dynamics are changed very little by variation of the sampling interval.

A compensation function was found which could be used for all three flight conditions. The expression for this compensation function is

$$D(z) = \frac{(Z - .9)(Z - .7 \pm j.65)}{(Z + .2)(Z + .1)^2} \quad (12)$$

Root locus plots were made for each flight condition and sampling rate. These are shown in Figures 39 through 51.

When interpreting the root locus plots, it is well to keep in mind that the dynamics of the airframe are never known too accurately. The body bending poles are known to be within certain frequency limits and they are known to be very lightly damped. Just how lightly damped is not really known and, therefore, what is thought to be only a decrease in damping may actually result in instability. For this reason, the body bending poles on the root locus must move inward, away from the periphery of the unit circle, as the system gain is increased from zero.

From Figures 39 through 43, it can be seen that both the first and second bending mode poles move outside the unit circle for sampling intervals of $T = .0308$ and $T = .0363$ seconds. For $T = .041$ seconds, the initial motion of both poles is nearly along the unit circle, and for

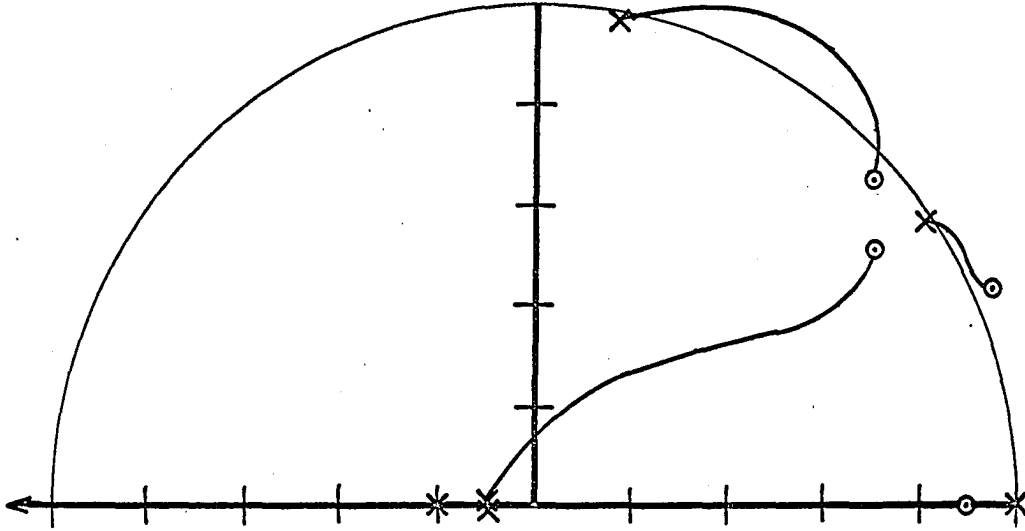


Figure 39. Root locus plot Condition A $T = .0308$ seconds

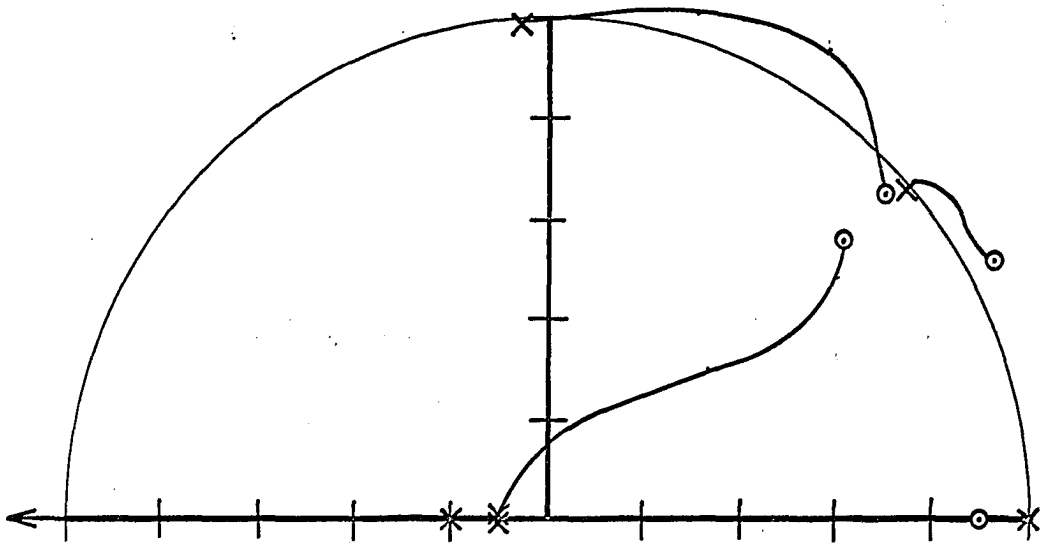


Figure 40. Root locus plot Condition A $T = .0363$ seconds

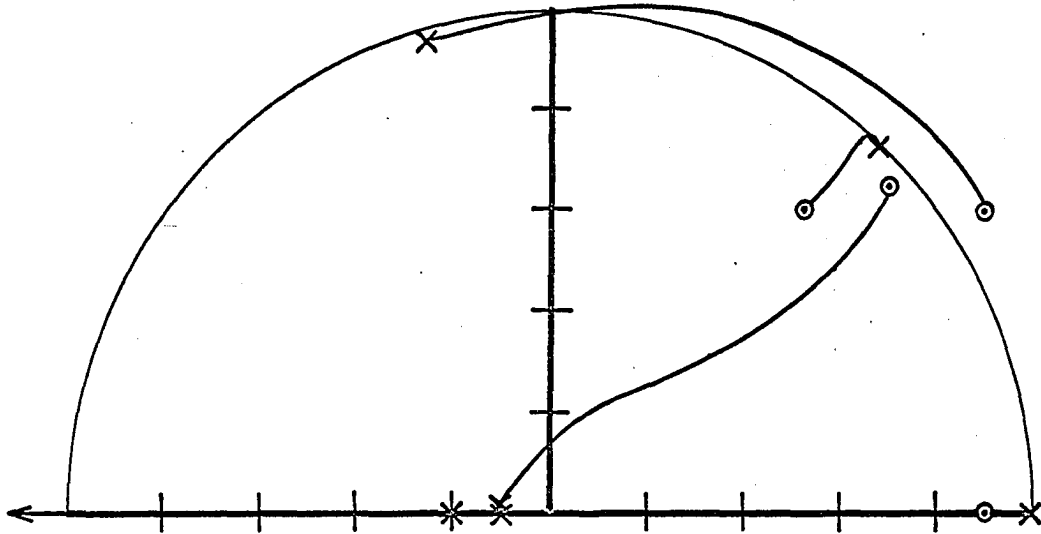


Figure 41. Root locus plot Condition A $T = .041$ seconds

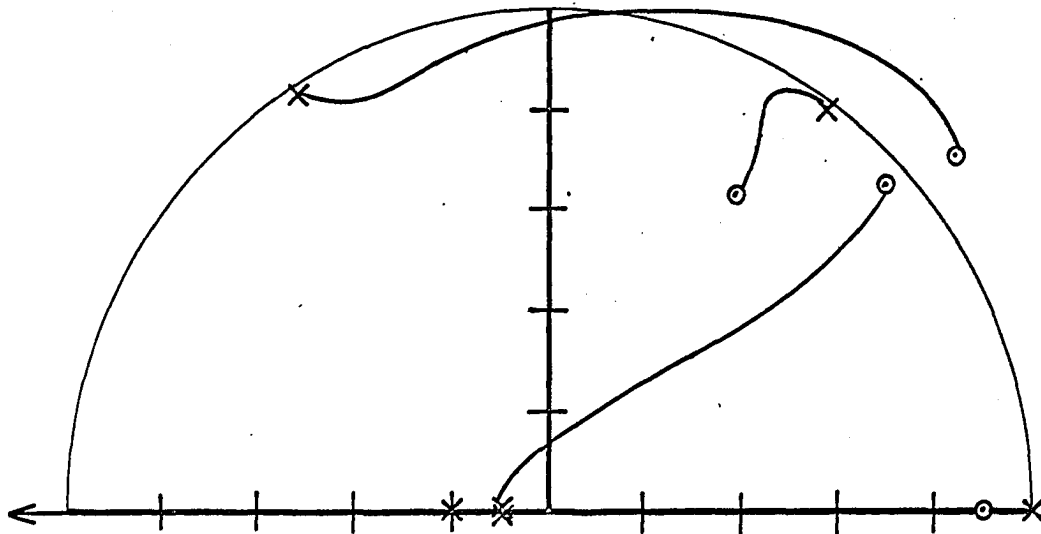


Figure 42. Root locus plot Condition A $T = .0473$ seconds

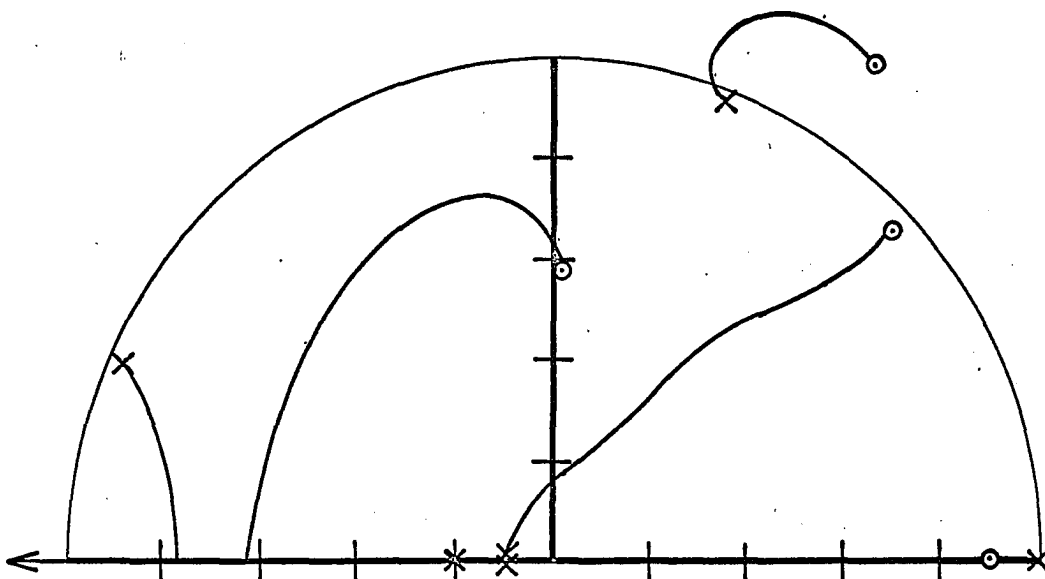


Figure 43. Root locus plot Condition A $T = .0605$ seconds

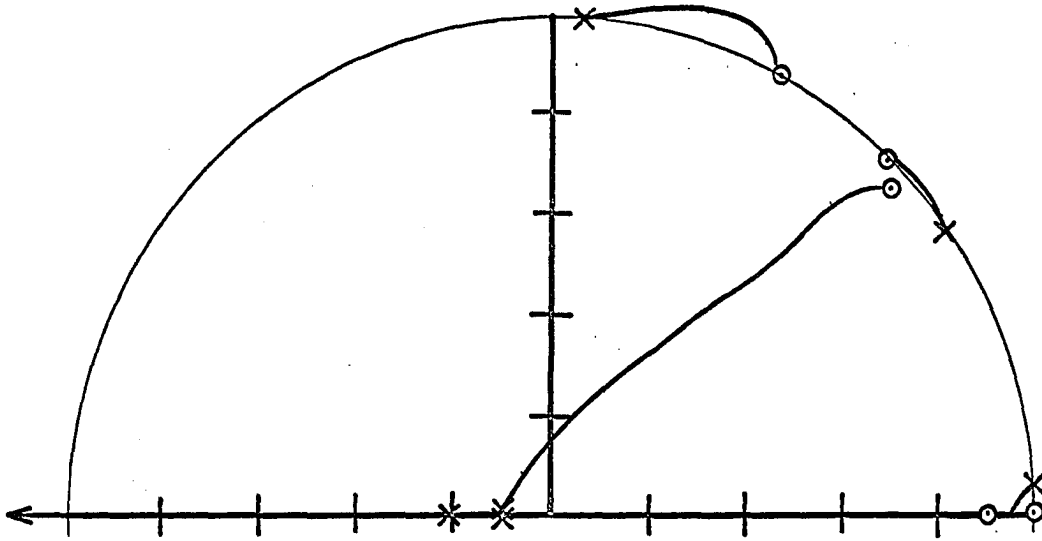


Figure 44. Root locus plot Condition B $T = .028$ seconds

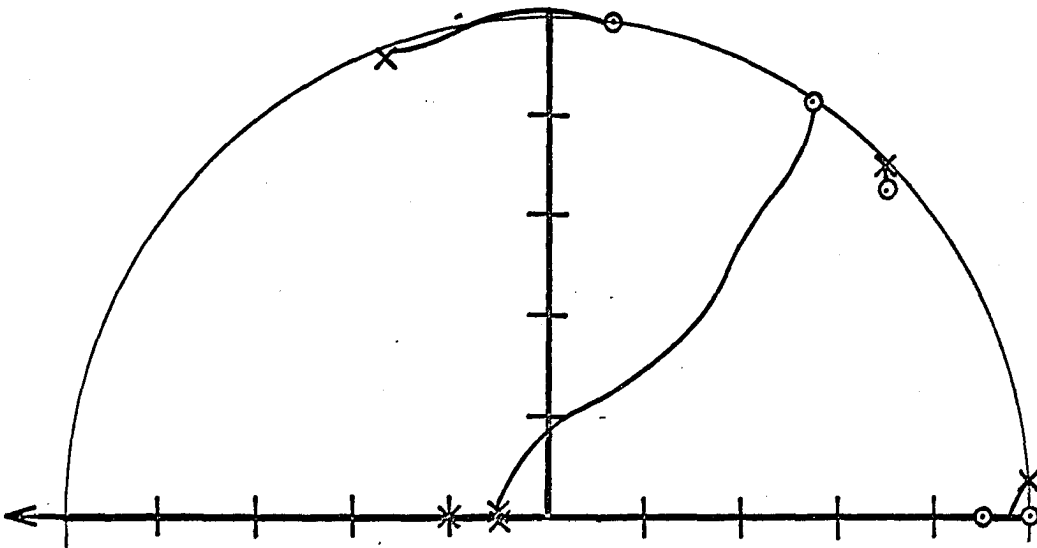


Figure 45. Root locus plot Condition B $T = .0357$ seconds

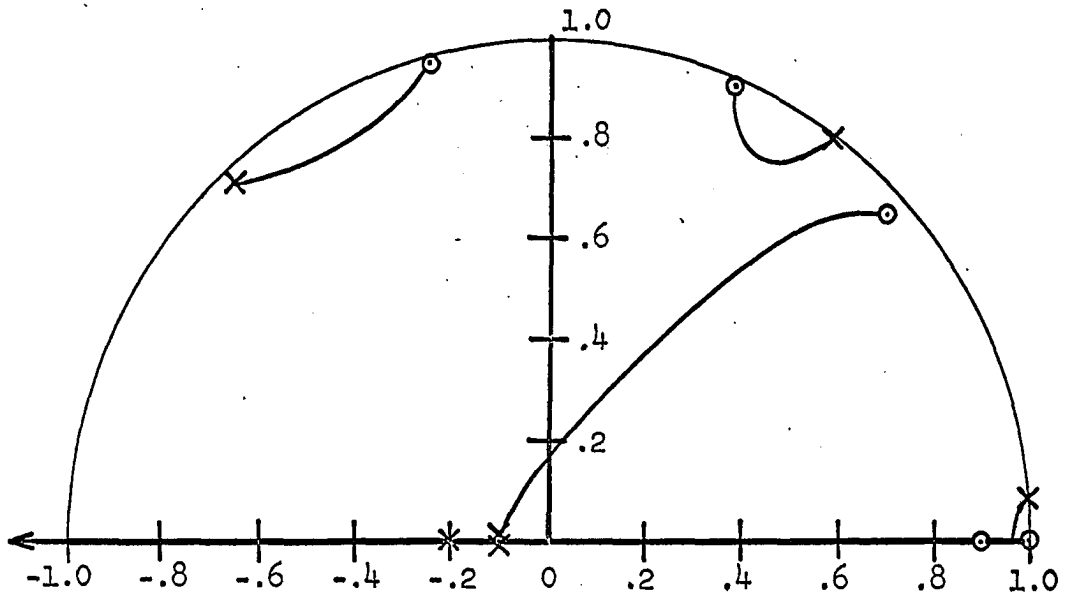


Figure 46. Root locus plot Condition B $T = .043$ seconds

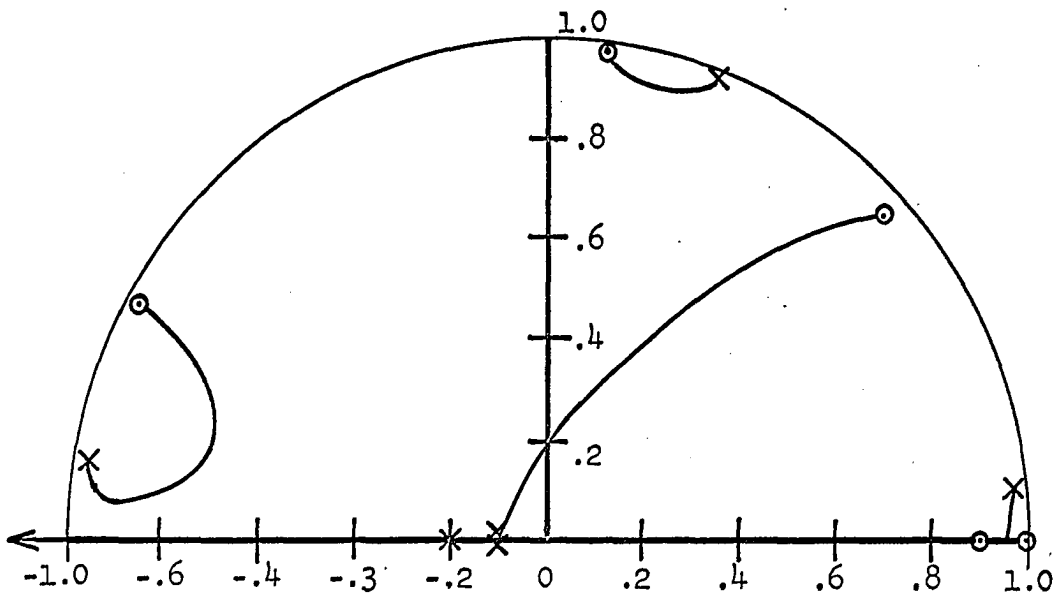


Figure 47. Root locus plot Condition B $T = .055$ seconds.

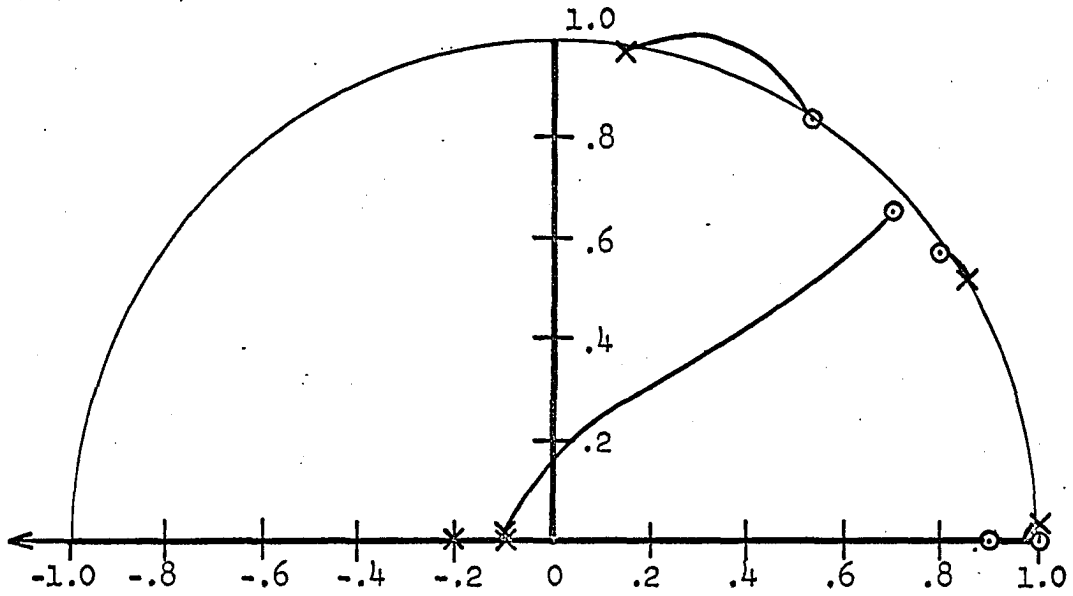


Figure 48. Root locus plot Condition C $T = .022$ seconds

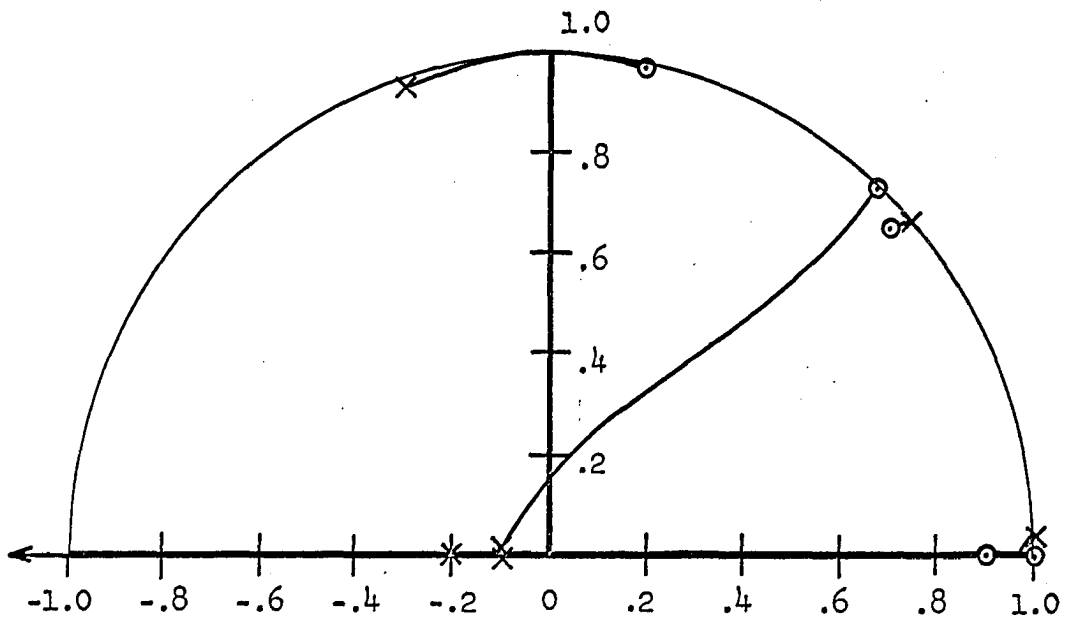


Figure 49. Root locus plot Condition C $T = .029$ seconds

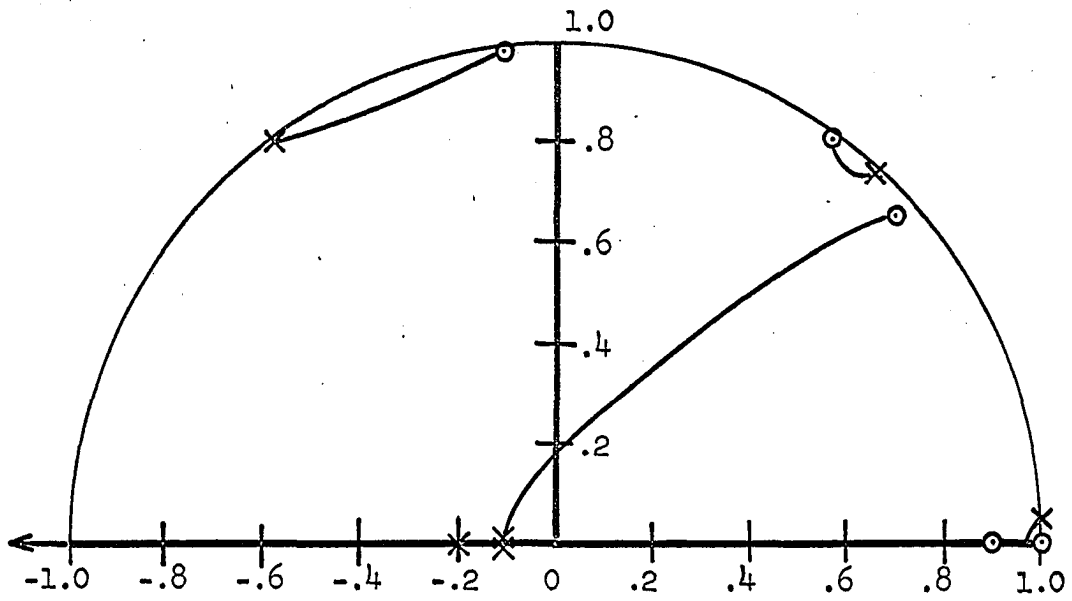


Figure 50. Root locus plot Condition C $T = .034$ seconds

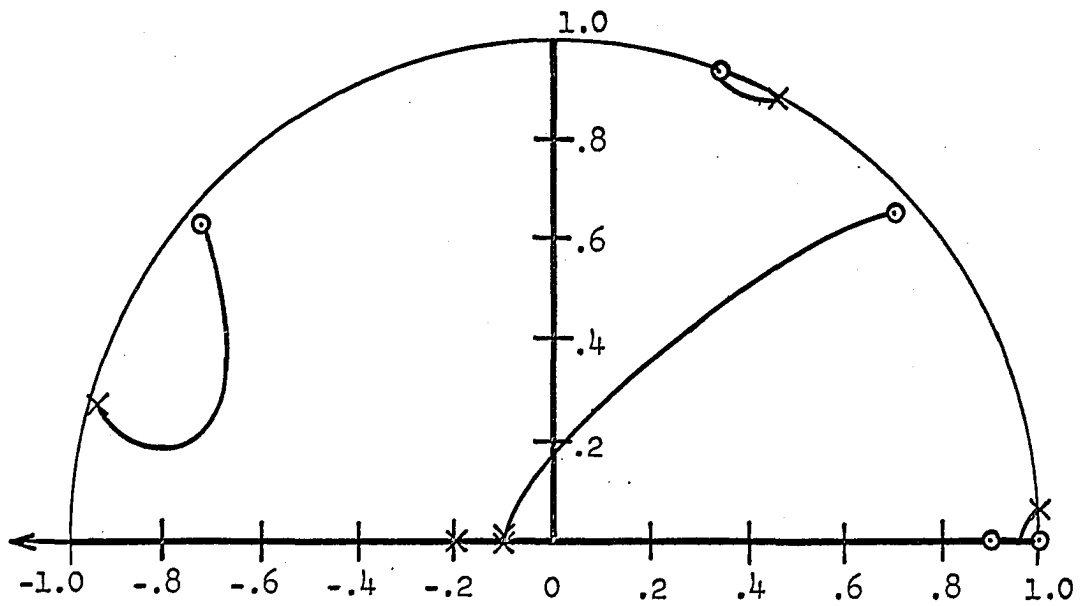


Figure 51. Root locus plot Condition C $T = .044$ seconds

$T = .0473$, both poles move inward away from the unit circle. The root locus for $T = .0473$ represents an acceptable system. If T is increased still further, then the first bending mode pole again moves outward as shown for $T = .0605$ seconds.

From Figures 44 through 47, it can be seen that the system is unsatisfactory except when $T = .043$ for Flight Condition B. For Flight Condition C, Figures 48 through 51 show that a sampling interval $T = .034$ is needed for an acceptable response.

In each case, the acceptable sampling rate is the one which places the pole somewhat further around the unit circle, in a counterclockwise sense, than the compensation zero. In effect, instead of tracking the pole position with the compensation zero as was done by Smythe et al., the changes in sampling rate forces the pole to stay in the proper position with respect to the zero.

This is the unique feature of this adaptive system. The roots of $D(Z)$ do not change position in the Z -plane when the sampling rate is changed, but the poles characteristic of the airframe dynamics do change and in the case of the body bending poles can be placed in a favorable position.

Figure 52 is a block diagram of the complete system. The pole positions describing the airframe dynamics are determined by the process identification technique described in Sections I and II and the sampling rate continuously adjusted to hold these poles in the desired position relative to the zero.

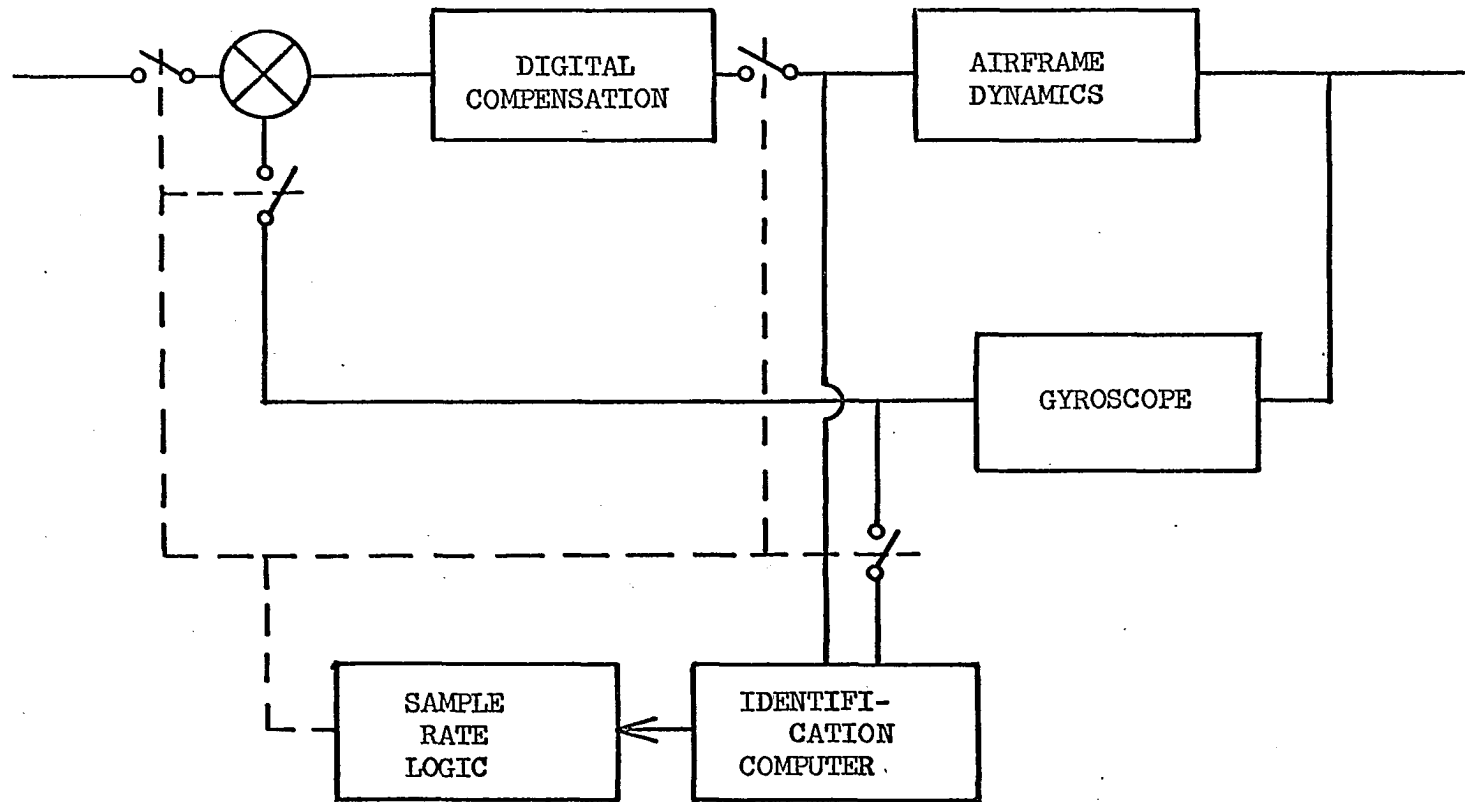


Figure 52. Block diagram of digital adaptive controller with variable sampling rate

IV. CONCLUSIONS

The adaptive control technique investigated can be broken down into two distinct parts. First, the process identification technique determines the location of the Z-plane poles associated with the body bending modes. Second, the actual adaptive feature corrects the pole position relative to the compensation zero.

The actual adaptation process appears to give excellent results provided the pole positions can be observed. That is, changes in the sampling rate will move the pole into the correct position. However, the technique used to determine the pole positions appears to give good accuracy only when very little noise is present in the system. Since there will undoubtedly be fairly high noise levels in any practical control system, this part of the technique will no doubt be marginal in practice.

For this reason, it is suggested that future work be directed at improving the process identification technique and the investigation of other methods such as digital filtering to determine the frequency of the bending mode.

The dynamics of the system have been simplified in this investigation by ignoring the dynamic characteristics of the gyros and hydraulic actuators. These effects should be included in future studies.

Finally, the complete system, including the identification of the body bending poles, should be simulated on a hybrid computation facility. This would allow the system to be optimized and would be very useful in proving or disproving the ultimate feasibility.

V. LITERATURE CITED

1. Schuck, O. Hugo. Honeywell's history and philosophy in the adaptive control field. Unpublished paper presented at Adaptive Flight Control Symposium, Wright Air Development Center, Jan. 1959. Minneapolis, Minnesota, Aeronautical Division, Minneapolis-Honeywell Regulator Co.
2. Hendrick, Russell C. Automatic gain control of a feedback control system by limit-cycle amplitude regulation. Unpublished M.S. thesis. Minneapolis, Minnesota, Library, University of Minnesota. 1961.
3. General Electric Electronics Laboratory. A survey of adaptive control technology. Syracuse, New York, author. ca. 1964.
4. Prince, Luther T. The influence of structural elasticity on the stability of airplanes and multi-stage missiles. Unpublished paper presented at the Advisory Group for Aeronautical Research and Development Meeting on Stability and Control, Brussels, Belgium, April, 1961. Minneapolis, Minnesota, Aeronautical Division, Minneapolis-Honeywell Regulator Co. 1961.
5. Smythe, R. K., DuPlessis, R. M., and Mattingly, L. K. A digital self-adaptive body bending filter for flexible airframe control. Unpublished paper presented at the Institute of Radio Engineers, National Aerospace Electronics Conference, Dayton, Ohio, May 1960. New York, N.Y., Institute of Electrical and Electronic Engineers. 1960.
6. Zaborsky, J. and Luedde, William J. Summary of the digital adaptive filter, digital polynomial filter, and specification set control techniques applied to large launch vehicles. National Aeronautics and Space Administration Contractor Report NASA CR-462. 1966.
7. Tou, Julius T. Digital and sampled-data control systems. New York, N.Y., McGraw-Hill Book Co., Inc., 1959.
8. Willett, R. M. and Lee, J. F. L. Feasibility study of digital control systems for rigid and flexible vehicles. Unpublished Internal Report R-ED 5148. Minneapolis, Minnesota, Aeronautical Division, Minneapolis-Honeywell Regulator Co. 1962.
9. Lendaris, George G. On the identification of linear systems. Unpublished paper presented at the AIEE Winter General Meeting, New York, N.Y., February 1961. New York, N.Y., American Institute of Electrical and Electronic Engineers. ca. 1962.

10. Levin, Morris J. Estimation of a system pulse transfer function in the presence of noise. Unpublished paper presented at the IEEE Joint Automatic Control Conference, Minneapolis, Minnesota, 1963. Lexington, Mass., Lincoln Laboratory, Massachusetts Institute of Technology. ca. 1963.

VI. ACKNOWLEDGMENT

The author wishes to express his appreciation for the suggestions and guidance given by his major professor, Dr. H. W. Hale in the preparation of this dissertation. He is also indebted to Dr. D. R. Boylan and the Engineering Research Institute for their support in this investigation. Finally, the author is indebted to Mr. D. J. Duven without whose expert computer programming ability this research effort would have been greatly prolonged.

VII. APPENDIX A - COMPUTER PROGRAM FOR THE SIMULATION OF THE
CLOSED LOOP SYSTEM WITH COMPENSATION

The program used to simulate the closed loop system is described in this appendix. This program computes X and Y for the system shown in Figure 53.

The input (α) may be a unit step function or a random number (selected from a random number generator which generates normally distributed random numbers for which the mean is zero and the variance is 1.00). D(Z) is a compensation network whose characteristic equation must be of order 20 or less. It has the general form

$$\begin{aligned}
 D(Z) &= \frac{c_0 Z^m + c_1 Z^{m-1} + \dots + c_m Z^0}{d_0 Z^m + d_1 Z^{m-1} + \dots + d_m Z^0} \\
 &= \frac{c_0 + c_1 Z^{-1} + \dots + c_m Z^{-m}}{d_0 + d_1 Z^{-1} + \dots + d_m Z^{-m}}
 \end{aligned} \tag{13}$$

The denominator is of order $m \leq 20$ but is normalized so that $d_0 = 1.0$. G(Z) is the system transfer function whose characteristic equation also must be of order 20 or less. Its general form is

$$G(Z) = \frac{a_0 + a_1 Z^{-1} + \dots + a_n Z^{-n}}{b_0 + b_1 Z^{-1} + \dots + b_n Z^{-n}} \tag{14}$$

where $n \leq 20$, and $b_0 = 1.0$. Any of the a, b, c, or d coefficients in G(Z) and D(Z) may be zero except b_0 and d_0 . The orders m and n of D(Z) and G(Z) may or may not be the same. The noise added to the output of G(Z) is a random number multiplied by a variable input parameter "k".

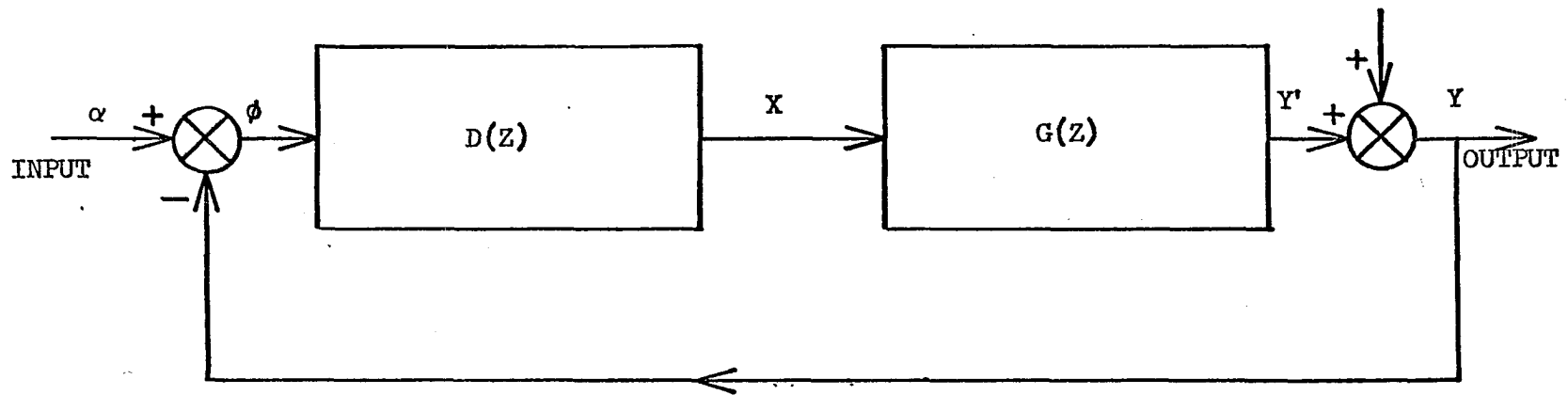


Figure 53. Block diagram of closed loop system used in digital simulation

$D(Z)$ and $G(Z)$ are each treated separately in the simulation. Taking $G(Z)$ as an example

$$G(Z) = \frac{a_0 + a_1 Z^{-1} + \dots + a_n Z^{-n}}{b_0 + b_1 Z^{-1} + \dots + b_n Z^{-n}} = \frac{Y'}{X} \quad (15)$$

or

$$b_0 Y' = (a_0 X + a_1 XZ^{-1} + \dots + a_n XZ^{-n}) - (b_1 Y'Z^{-1} + \dots + b_n Y'Z^{-n}) \quad (16)$$

where XZ^{-1} is X delayed one sample period and can be denoted $X[(m-1)T]$, and $Y'Z^{-n}$ is Y' delayed n sample periods and is denoted $Y'[(m-n)T]$, etc.

Since $b_0 = d_0 = 1.0$

$$Y'[(m)T] = (a_0 X[(m)T] + a_1 X[(m-1)T] + \dots + a_n X[(m-n)T]) - (b_1 Y'[(m-1)T] + \dots + b_n Y'[(m-n)T]) \quad (17)$$

This equation may be used to compute the present $Y' = Y'[(m-n)T]$, given past Y' and past and present X .

We may note from Figure 53 that

$$\phi(mT) = \alpha(mT) - Y[(m)T] \quad (18)$$

In other words, we must know $Y[(m)T]$ in order to find $\phi(mT)$. More involved programming was circumvented by making the constraint that a_0 equal zero. This constraint means that $Y'[(m)T]$ depends only upon past data and no longer depends upon $X[(m)T]$. (This constraint merely means that the numerator of $G(Z)$ must be of at least one order less than the denominator.) The sequence of computation is as follows.

- (1) Compute $Y'[(m)T]$ from past X and Y' data.

- (2) Add noise to $Y'[(m)T]$ to get $Y[(m)T]$.
- (3) Subtract $Y[(m)T]$ from $\alpha_{(0)}$ to get $\phi_{(0)}$.
- (4) Compute $X[(m)T]$ from past X and ϕ data and $\phi(mT)$.

This completes one cycle of computation. The desired values are read out and all present and past values are shifted in storage to simulate delaying them one more time period (20 values are kept to accommodate up to a 20th order characteristic equation). The cycle is repeated for the desired number of iterations.

The input required is listed below.

m - order of characteristic equation of $D(Z)$.

n - order of characteristic equation of $G(Z)$.

n_0 - number of iterations desired.

k - multiplier for noise to be added to the output.

$2m + 1$ coefficients - c and d coefficients for $D(Z)$

(d_0 assumed = 1.0).

$2n + 1$ coefficients - a and b coefficients for $G(Z)$

(b_0 assumed = 1.0).

The number of sets of output values is equal to the input value n_0 .

Each set may consist of one of the two following descriptions.

X, Y - to be used later to identify $G(Z)$ with output noise added.

α, X, Y', Y - used to check out program and data.

VIII. APPENDIX B - COMPUTER PROGRAM FOR LEAST SQUARES

COEFFICIENT IDENTIFICATION

This program forms a matrix of combinations of the X and Y data which were the output from the computer program described in Appendix A. This matrix is used to solve for the a and b coefficients of

$$G(Z) = \frac{a_1 Z^{-1} + \dots + a_n Z^{-n}}{1.0 + b_1 Z^{-1} + \dots + b_n Z^{-n}} \quad (19)$$

The input X is in sampled form. $G(Z)$ is the system transfer function as described in Appendix A. The output Y is in sampled form with noise added.

The method used to solve for the a and b coefficients is the method of least squares. Given an equation

$$0 = aX - Y + b \quad (20)$$

the constants a and b are to be determined such that

$$f(a,b) = \sum_{i=1}^n (aX_i - Y_i + b)^2 \quad (21)$$

is a minimum. To do this it is necessary to first square each equation used. These are then added to form a new equation. Partial derivatives of the resulting equation are taken with respect to each coefficient and set equal to zero. The set of linear equations resulting from taking partial derivatives is solved for the a and b coefficients.

In our specialized case, by rearranging the equations used in Appendix A, we get

$$0 = (a_0 X[(m)T] + a_1 X[(m-1)T] + \dots + a_n X[(m-n)T] - (b_0 Y[(m)T] + b_1 Y[(m-1)T] + \dots + b_n Y[(m-n)T]) \quad (22)$$

But we have a_1 equal to zero and b_0 equal to one. Also, to simplify the notation, let $n = 2$. Thus

$$0 = a_1 X[(m-1)T] + a_2 X[(m-2)T] - b_1 Y[(m-1)T] - b_2 Y[(m-2)T] - Y[(m)T] \quad (23)$$

In easier notation

$$0 = a_1 v + a_2 w - b_1 x - b_2 y - z \quad (24)$$

These equations are formed consecutively from the X and Y data. The number of equations to be added in forming each matrix is a variable input parameter. Squaring this equation gives

$$\begin{aligned} 0 = & (v^2)a_1^2 + (w^2)a_2^2 + (x^2)b_1^2 + (y^2)b_2^2 + (z^2) + \\ & + 2(vw)a_1a_2 + 2(-vx)a_1b_1 + 2(-vy)a_1b_2 + 2(-wx)a_2b_1 \\ & + 2(-wy)a_2b_2 + 2(xy)b_1b_2 + 2(-vz)a_1 + 2(-wz)a_2 \\ & + 2(xz)b_1 + 2(yz)b_2 \end{aligned} \quad (25)$$

Taking partial derivatives and setting the results equal to zero yields

$$\begin{aligned}
\frac{\partial}{\partial a_1} = 0 &= 2v^2 a_1 + 2vwa_2 - 2vxb_1 - 2vyb_2 - 2vz \\
\frac{\partial}{\partial a_2} = 0 &= 2vwa_1 + 2w^2 a_2 - 2wxb_1 - 2wyb_2 - 2wz \\
\frac{\partial}{\partial b_1} = 0 &= -2vxa_1 - 2wxa_2 + 2x^2 b_1 + 2xyb_2 + 2yz \\
\frac{\partial}{\partial b_2} = 0 &= -2vya_1 - 2wya_2 + 2xyb_1 + 2y^2 b_2 + 2yz
\end{aligned}
\tag{26}$$

In matrix form, Equations (26) are written as

$$\begin{bmatrix} v^2 & vw & -wx & -vy \\ vw & w^2 & -wx & -wy \\ -vx & -xw & x^2 & xy \\ -vy & -yw & yx & y^2 \end{bmatrix} \begin{matrix} a_1 \\ a_2 \\ b_1 \\ b_2 \end{matrix} = \begin{bmatrix} wz \\ wz \\ -xz \\ -yz \end{bmatrix}
\tag{27}$$

This matrix equation is the output of this program. A standard "linear equation solution" program is used to solve for the coefficients a_1 , a_2 , b_1 , and b_2 .

The input required is given below.

amt - amount of X and Y data to be read in.

n_o - number of matrices to be computed.

n - number of equations to be added together informing each matrix.

data - X and Y data computed in the "Closed loop system with compensation" program.

The output will be the matrix of combinations of coefficients.

IX. APPENDIX C - PROGRAM TO CONVERT LAPLACE
TRANSFORMS TO Z-TRANSFORMS

In principle, the procedure for finding the poles and zeros of the Z-transform corresponding to a Laplace transform of the form

$$F(s) = \frac{\prod_{i=1}^m (s + a_i)^{q_i}}{\prod_{i=1}^n (s + b_i)^{r_i}} \quad (28)$$

is fairly straightforward. The Laplace transform is first expanded into its partial fraction form, as

$$F(s) = \sum_{i=1}^n \sum_{j=0}^{r_i-1} \frac{K_{i,j}}{(s + b_i)^{i-j}} \quad (29)$$

and then each term is replaced by its Z-transform

$$(-1)^k \frac{K_{i,j}}{k!} \frac{d^k}{db_i^k} \left\{ \frac{Z}{Z - e^{-b_i T}} \right\} \quad (30)$$

where $k = r_i - j - 1$, and T is the sampling time. It can be shown that for $k \geq 1$

$$\frac{d^k}{db_i^k} \left\{ \frac{Z}{Z - e^{-b_i T}} \right\} = \frac{(-1)^k Z^k e^{-b_i T} p_k(Z)}{(Z - e^{-b_i T})^{k+1}} \quad (31)$$

where $p_k(Z)$ is a polynomial of the form

$$\begin{aligned}
p_k(Z) = & Z^{k-1} + P_{k,1} Z^{k-2} e^{-b_i T} + P_{k,2} Z^{k-3} e^{-2b_i T} + \dots \\
& + P_{k,l} Z^{k-l-1} e^{-lb_i T} + \dots + P_{k,k-2} Z e^{-(k-2)b_i T} \\
& + P_{k,k-1} e^{-(k-1)b_i T}
\end{aligned} \tag{32}$$

in which the coefficients, $P_{k,l}$, are given recursively by

$$P_{1,0} = 1, P_{k,l} = (l+1)P_{k-1,l} + (k-l)P_{k-1,l-1} \tag{33}$$

where $1 \leq l \leq k-1$, $P_{k-1,0} = 1$, and $P_{k-1,k-1} = 0$. Thus, the Z-transform, $G(Z)$, of $F(s)$ will be

$$G(Z) = \sum_{i=1}^n \frac{r_i - 1}{\sum_{j=0}^{r_i - 1} K_{i,j} Z^j e^{-b_i T}} \frac{p_k(Z)}{k!(Z - e^{-b_i T})^{k+1}} \tag{34}$$

where $k = r_i - j - 1$ and $p_k(Z)$ are as defined above. These fractions can be added to give a single fraction representing the Z-transform. The numerator of this fraction must be found in polynomial form, and a root-finder must be used to find the zeros, but the denominator can be found in factored form and the poles will be $e^{-b_i T}$ for $1 \leq i \leq n$.

A less general version of this technique was tried. The recursive relations for finding the partial fraction coefficients and the polynomial coefficients, $P_{k,l}$, were not known at the time this technique was tried. The computed locations of the zeros were very inaccurate. Most of the inaccuracy appeared to be due to the round-off error in the formation of the coefficients of the numerator polynomial, but the

inability of the rootfinder being used at that time to accurately find the roots of a polynomial having some of its roots nearly equal probably contributed to the error also. Consequently, several non-standard techniques were employed in the final program to enable the zeros to be found accurate to 5 places using 8-place floating point arithmetic, which was all that was conveniently available.

One of the techniques used to improve accuracy was to avoid the computation of the coefficients of the numerator polynomial. This was done by computing the partial fraction expansion of the Z-transform fraction. The numerator, evaluated at a given value of Z, can then be computed by multiplying the partial fraction expansion times the denominator, both evaluated at Z.

The Z-transform partial fraction expansion coefficients are computed as follows:

The Laplace transform is assumed to be in the form of Equation (1), except the r_i are restricted to the range $1 \leq r_i \leq 3$, and the partial fraction expansion of the Laplace transform is given by Equation (29). The first partial fraction coefficient corresponding to a given pole (and the only coefficient for a first degree pole) is given by

$$K_{i,0} = \frac{\prod_{k=1}^m (-b_i + a_k)^{q_k}}{\prod_{\substack{k=1 \\ k \neq i}}^n (-b_i + b_k)^{r_k}} \quad (35)$$

The second coefficient for a second degree pole and the second and third coefficients for a third degree pole are given by

$$K_{i,1} = K_{i,0} \left[\sum_{k=1}^m \frac{q_k}{-b_i + a_k} - \sum_{\substack{k=1 \\ k \neq i}} \frac{r_k}{-b_i + b_k} \right] \quad (36)$$

$$K_{i,2} = K_{i,0} \left[\sum_{k=1}^m \frac{-q_k}{(-b_i + a_k)^2} - \sum_{\substack{k=1 \\ k \neq i}} \frac{-r_k}{(-b_i + b_k)^2} + \left(\sum_{k=1}^m \frac{q_k}{-b_i + b_k} - \sum_{\substack{k=1 \\ k \neq i}} \frac{r_k}{-b_i + b_k} \right)^2 \right] \quad (37)$$

The Z-transform, $G(Z)$, is obtained by replacing each term in Equation (29) by its Z-transform. Thus, the following sums of terms

$$\frac{K_{i,0} Z^{-b_i T}}{Z - e^{-b_i T}}, \quad \frac{K_{i,0} Z T e^{-b_i T}}{(Z - e^{-b_i T})^2} + \frac{K_{i,1} Z}{Z - e^{-b_i T}},$$

and

$$\frac{\frac{1}{2} K_{i,0} Z T^2 e^{-b_i T} (Z + e^{-b_i T})}{(Z - e^{-b_i T})^3} + \frac{K_{i,1} Z T e^{-b_i T}}{(Z - e^{-b_i T})^2} + \frac{K_{i,2} Z}{Z - e^{-b_i T}}$$

correspond to a first, second, or third degree pole respectively.

Since Z is common to the numerator of all terms of $G(Z)$

$$G(Z) = Z \left[\sum_{i=1}^n \sum_{j=0}^{r_i-1} \frac{H_{i,j}}{(Z - e^{-b_i T})^{r_i-j}} \right] \quad (38)$$

The coefficients, $H_{i,j}$, are as follows

$$\text{for a first degree pole: } H_{i,0} = K_{i,0} \quad (39)$$

$$\text{for a second degree pole: } H_{i,0} = K_{i,0} T e^{-b_i T},$$

$$H_{i,1} = K_{i,1} \quad (40)$$

$$\text{for a third degree pole: } H_{i,0} = K_{i,0} T^2 e^{-2b_i T},$$

$$H_{i,1} = (K_{i,1} + \frac{1}{2} K_{i,0}) T e^{-b_i T},$$

$$H_{i,2} = K_{i,2} \quad (41)$$

Obviously, $Z = 0$ is a zero of every $G(Z)$. The rest of the zeros will be roots of the expression

$$\left(\sum_{i=1}^n \sum_{j=0}^{r_i-1} \frac{H_{i,j}}{(Z - e^{-b_i T})^{r_i-j}} \right) \times \prod_{i=1}^n (Z - e^{-b_i T})^{r_i} \quad (42)$$

The degree of this expression is one less than the degree of the Z -transform (and Laplace transform) denominator. When the degree of the Laplace transform numerator is less than one the degree of the Laplace transform denominator less one, then the coefficient of the highest order term of Equation (42) is zero. However, when round-off errors in the computation of the $H_{i,j}$ are considered, this coefficient may not be

exactly zero. In this case it is advantageous to reduce the degree of Equation (42) by one. This is done by multiplying the partial fraction expansion in Equation (38) by $(Z - e^{-b_1 T})$ and by subsequently dropping the first pole if it is simple, or by reducing r_1 by one if it is not. The new values of $H_{i,j}$ are computed as follows

$$H_{i,j} = H_{i,j} \times (e^{-b_i T} - e^{-b_1 T}) + H_{i,j-1} \quad (43)$$

for $j = 1, \dots, r_{i-1}$ and taken in decreasing order,

$$H_{i,0} = H_{i,0} \times (e^{-b_i T} - e^{-b_2 T}) \quad (44)$$

When a root has been found at $Z = X$, it is divided out of the partial fraction expansion by modifying $H_{i,j}$ as follows

$$H_{i,0} = \frac{H_{i,0}}{e^{-b_i T} - X} \quad (45)$$

and

$$H_{i,j} = \frac{H_{i,j} - H_{i,j-1}}{e^{-b_i T} - X} \quad (46)$$

for $j = 1, \dots, r_i - 1$, and taken in increasing order. Then another pole is multiplied out in the same manner as shown above, and the process is continued until all the roots have been found.

The rootfinder used is a modified version of one that was originally written for the Illiac and was subsequently adapted for the Cyclone. The rootfinder was modified to keep a copy of the original polynomial as well as the reduced polynomial from which the previously found roots had been divided. When a value of Z is found which satisfies

the reduced polynomial, that value is inserted into the original polynomial to determine whether it is also satisfied. In cases in which the original polynomial is not satisfied, other values of Z near to this value are tried in a systematic way until one is found which does satisfy the original polynomial. This technique prevents the accumulation of errors which results when an inaccurately known root is divided out of the polynomial thereby changing the value of all subsequently found roots. Also, a technique was devised for finding a root to a given number of places of accuracy relatively independent of the distance of the root from all other roots. If $f(Z)$ is a polynomial in Z having roots at r_1, r_2, \dots, r_n , then at any given value of Z , $|f(Z)|^2$ is equal to $\prod_{i=1}^n |Z - r_i|^2$. Suppose that the sequence of trial roots, Z , generated by the rootfinder is approaching the root r_1 , and suppose that the criterion for ending iteration is that the distance from Z to r_1 should be less than some small number δ . Then the iteration should cease when $|f(Z)|^2$ is less than or equal to $\delta^2 \cdot \prod_{i=2}^n |Z - r_i|^2$. The variables BDH and SMAR in the rootfinder (see the program) are estimates of $\prod_{i=2}^n |Z - r_i|^2$ for the reduced and original polynomials respectively, and these estimates improve as Z approaches a root. Thus the criterion for ending iteration is $|f(Z)|^2$ less than or equal to $\delta^2 \cdot (\text{BDH})$ or $|f(Z)|^2$ less than or equal to $\delta^2 \cdot (\text{SMAR})$.

The program is shown in Figure 54.

```

array          AAR[60], AAI[60], BBR[60], BBI[60], CCR[60],
               CCI[60]
array          DDR[60], DDI[60], IA[20], IC[20], ID[20]
array          ZPR[20], ZPI[20], RR[60], RI[60]

27  read          T
    if (T) 1,1,2
1   stop

2   continue
e   clock          TIME
    read          T
    if (T-1) 700,704,702
700 do 701          I=1,60
701 RR[I]=RI[I]=0
    go to 704
702 read          IR
    do 703          I=1,IR
703 read          RR[I], RI[I]
704 read          E1,E2
    E1=E1*E1; E3=E2*E2; E4=5*E2
e   code8          3
e   crlf           3
    N=1
    read          ST, R, IIA
    do 3          I=1,IIA
    read          IA[I], AAR[I], AAI[I]
3   N=N+IA[I]
    read          IID
    do 4          I=1,IID
    read          ID[I], DDR[I], DDI[I]
    N=N-ID[I]; T=exp(-DDR[I]*ST); THETA=-DDI[I]*ST
    TR=ZPR[I]=cos(THETA)*T; TI=ZPI[I]=sin(THETA)*T
e   lxn           I,1
e   out           ID,1,010
4   punch         TR, TI, /

c compute Z transform partial fraction expansion
20  M=1
    do 28          L=1,IID
    SR=-DDR[L]; SI=-DDI[L]; FZR=1; FZI=0
e   tsx           Σ210,5
e   tsx           Σ130,6
    T=FZR*FZR+FZI*FZI; FZR=FZR/T; FZI=-FZI/T
e   tsx           Σ200,5
e   tsx           Σ130,6
    if (ID[L]-2) 26,24,25

```

Figure 54. Z-transform computer program

```

24  BBR[M] = (FZR * ZPR[L] - FZI * ZPI[L]) * ST
    BBI[M] = (FZR * ZPI[L] + FZI * ZPR[L]) * ST; M=M+1
    FNR = FNI = 0; INT = 1
e   tsx                               Σ210,5
e   tsx                               Σ170,6
    FNR = -FNR; FNI = -FNI
e   tsx                               Σ200,5
e   tsx                               Σ170,6
    BBR[M] = FNR * FZR - FNI * FZI; BBI[M] = FNR * FZI + FNI * FZR; M=M+1
    go to 28

25  TNR = ZPR[L]; TNI = ZPI[L]; TR = TNR * TNR - TNI * TNI
    TI = 2 * TNR * TNI; T = ST * ST; BBR[M] = (FZR * TR - FZI * TI) * T
    BBI[M] = (FZR * TI + FZI * TR) * T; M=M+1
    FNR = FNI = FWR = FWI = 0; INT = 2
e   tsx                               Σ210,5
e   tsx                               Σ170,6
    FNR = -FNR; FNI = -FNI; FWR = -FWR; FWI = -FWI;
e   tsx                               Σ200,5
e   tsx                               Σ170,6
    TR = ST/2 + FNR; TI = FNI * TNR + TR * TNI; TR = TR * TNR - FNI * TNI
    BBR[M] = (TR * FZR - TI * FZI) * ST; BBI[M] = (TR * FZI + TI * FZR) * ST; M=M+1
    TR = FNR * FNR - FNI * FNI + FWR; TI = 2 * FNR * FNI + FWI
    BBR[M] = (FZR * TR - FZI * TI) / 2; BBI[M] = (FZR * TI + FZI * TR) / 2; M=M+1
    go to 28

26  BBR[M] = FZR; BBI[M] = FZI; M=M+1
28  continue
    IBB = M - 1; LBB = LID = 1

c compute K
13  if (N) 14, 15, 1
14  continue
e   tsx                               Σ250,6
15  M = LBB - 1; SR = 0
    do 16                               L = LID, IID
16  M = M + ID[L]; SR = SR + BBR[M]; IA[L] = IC[L] = ID[L]
    R = R * SR; LAA = LCC = LBB; LIC = LIA = LID; IR = 0; IIA = 1
    punch                               /, R, /
    do 17                               I = LBB, IBB
    AAR[I] = BBR[I] - CCR[I] = BBR[I] / SR; AAI[I] = BBI[I] - CCI[I] = BBI[I] / SR
e   trss1                              *+2
e   tru                                 *+7
e   tab                                 1
    punch                               BBR[I], BBI[I], /
17  continue
    T = 0

```

Figure 54 (Continued)

```

punch          /, T, T, /
go to 50

c load numerator
200 IIC=IIA; K=0
    do 201          I=1,IIA
201 IC[I]=IA[I]; CCR[I]=AAR[I]; CCI[I]=AAI[I]
e   tru          1,5

c load denominator
210 IIC=IID; K=L
    do 211          I=1,IID
211 IC[I]=ID[I]; CCR[I]=DDR[I]; CCI[I]=DDI[I]
e   tru          1,5

c multiply factors
130 do 135          I=1,IIC
    if (I-K) 132,135,132
132 J=IC[I]; TR=CCR[I]+SR; TI=CCI[I]+SI
134 T=FZR*IR-FZI*TI; FZI=FZR*TI+FZI*IR; FZR=T
    J=J-1
    if (J) 1,135,134
135 continue
e   tru          1,6

c subroutine for taking derivative
170 do 176          I=1,IIC
    if (I-K) 174,176,174
174 TR=CCR[I]+SR; TI=CCI[I]+SI; T=IR*TR+TI*TI; TR=TR/T; TI=-TI/T
    ACCR=IC[I]*IR; ACCI=IC[I]*TI; FNR=FNR+ACCR; FNI=FNI+ACCI
    if (INT-2) 176,181,1
181 FWR=FWR-(ACCR*IR-ACCI*TI); FWI=FWI-(ACCR*TI+ACCI*IR)
176 continue
e   tru          1,6

c subroutine for multiplying out pole
250 SR=ZPR[LID]; SI=ZPI[LID]; M=IBB; L=IID
253 TR=ZPR[L]-SR; TI=ZPI[L]-SI; J=ID[L]-1
    ACCR=BBR[M]*IR-BBI[M]*TI; ACCI=BBR[M]*TI+BBI[M]*IR
    do 251          I=1,J
    TWR=BBR[M-1]; TWI=BBI[M-1]
    BBR[M]=ACCR+TWR; BBI[M]=ACCI+TWI; M=M-1
251 ACCR=TWR*IR-TWI*TI; ACCI=TWR*TI+TWI*IR
    BBR[M]=ACCR; BBI[M]=ACCI; M=M-1
    if (L-LID) 1,254,252
252 L=L-1
    go to 253

```

Figure 54 (Continued)

```

254 LBB=LBB+1; ID[LID]=ID[LID]-1
    if (ID[LID]) 1,255,256
255 LID=LID+1
256 continue
e   tru                               1,6

c find roots
50  ERROR=E3
e   tsx                               Σ109,5
    go to 609

609 IR=IR+1; RR[IR]=XR; RI[IR]=XI
    punch XR, XI, SR, SI, /
e   trss2                               *+2
e   tru                                 *+7
e   mode                               3
    punch /, XR, XI, /
e   mode                               2
    if (LBB-IBB+1) 615, 710, 1

710 continue
e   clock
e   sub                               TIME
e   crlf                               1
e   out                               ,,030
e   crlf                               1
e   tru                               Σ27

5   ERROR=E1
e   tsx                               Σ525,5
e   tsx                               Σ109,5
    go to 6
    go to 630

6   continue
e   tsx                               Σ520,5
e   tsx                               Σ540,5
    XMR=XR; XMI=XI; FWR=SR; FWMI=SI; SML=R; INT=4
e   tsx                               Σ120,6
    if ((HR*HR+HI*HI)/XSQ-E3) 625, 625, 7

7   HMR=-HR; HMI=-HI
e   tsx                               Σ540,5
    FNMR=SR; FNMI=SI
    if (R-SML) 8, 8, 9

8   SML=R; INT=3
9   continue
e   tsx                               Σ525,5
e   tsx                               Σ535,7
e   tsx                               Σ145,6
    FWR=SR; FWI=SI

```

Figure 54 (Continued)

```

e   tsx            $\Sigma 111,6$ 
e   tsx            $\Sigma 120,6$ 
R=HR*HR+HI*HI
if (R/XSQ-E3) 620, 620, 10
10  AMR=-((HMR*HR+HMI*HI)/R); AMI=(HMR*HI-HMI*HR)/R
DR=AMR+1; DI=AMI
608 continue
e   tsx            $\Sigma 520,5$ 
e   tsx            $\Sigma 540,5$ 
FZR=SR; FZI=SI; FNR=FNMR; FNI=FNMI; FWR=FWMR; FWI=FWMI
if (R-SML) 11, 11, 12
11  SML=R; INT=1
12  HR=HMR; HI=HMI; XR=XMR; XI=XMI; ERROR=E3
e   tsx            $\Sigma 533,7$ 
e   tsx            $\Sigma 150,5$ 
go to 609
go to 609

620 AMR=sqrt((HMR*HMR+HMI*HMI)/XSQ)/E2
621 AMI=DI=0; DR=AMR+1; T=DR/AMR
XR=XMR-T*HMR; XI=XMI-T*HMI
go to 608
625 HMR=sqrt(XSQ)*E2
626 HMI=0; XR=XMR-HMR; XI=XMI
e   tsx            $\Sigma 540,5$ 
FNR=SR; FNMI=SI
if (R-SML) 627, 627, 628
627 SML=R; INT=2
628 AMR=-0.5
go to 621
630 HMR=sqrt(XSQ)*E2
e   tsx            $\Sigma 520,5$ 
e   tsx            $\Sigma 540,5$ 
XMR=XR; XMI=XI; FWMR=SR; FWMI=SI; SML=R; INT=3
go to 626

c divide by root and multiply by pole
615 M=LBB
do 617 L=LID, IID
TR=ZPR[L]-XR; TI=ZPI[L]-XI; SR=SI=0; J=ID[L]+M
T=TR*TR+TI*TI; TR=TR/T; TI=-TI/T
616 SR=BBR[M]-SR; SI=BBI[M]-SI; T=SR*TR-SI*TI
SI=BBI[M]=SR*TI+SI*TR; SR=BBR[M]=T; M=M+1
if (M-J) 616,617,1
617 continue
e   tsx            $\Sigma 250,6$ 
if (IIA) 619,1,614

```

Figure 54 (Continued)

```

619 IIA=1
613 continue
e   trss1          *+2
e   tru            Σ5
      do 618       M=LBB,IBB
e   tab            1
618 punch          BBR[M], BBI[M], /
      go to 5
614 if (mag(XI)-E4) 613,613,612
612 IIA=-1; XI=-XI
      go to 609

c rootfinder subroutine
109 N=IR+1; XR=RR[N]-1; XI=RI[N]
e   tsx            Σ145,6
      FZR=SR; FZI=SI; XR=RR[N]+1
e   tsx            Σ145,6
      FNR=SR; FNI=SI; XR=RR[N]
e   tsx            Σ145,6
      FWR=SR; FWI=SI; AMR=-0.5; DR=0.5; HR=-1; AMI=DI=HI=0
      R=10+08
      go to 149
148 if (INT-1) 1, 49,110
149 T=AMR*AMR+AMI*AMI; TR=-(DR*AMR+DI*AMI)/T
      AMI=(DR*AMI-DI*AMR)/T; AMR=IR
e   tsx            Σ530,7
      SR=FZR; SI=FZI
e   tsx            Σ535,7
      FWR=SR; FWI=SI; INT=3; R=SML
110 continue
e   tsx            Σ111,6
      if (R-BDH*ERROR) 60, 60, 119
119 continue
e   tsx            Σ120,6
      if ((HR*HR+HI*HI)/XSQ-10-16) 61, 61, 148
148 continue
e   tsx            Σ535,7
e   tsx            Σ145,6
      FWR=SR; FWI=SI
150 R=FWR*FWR+FWI*FWI
      if (R-SML) 149, 149, 48
149 SML=R; INT=3
      go to 110

60 continue
e   tru            1,5
61 continue
e   tru            2,5

```

Figure 54 (Continued)

```

111 CR=FWR*DR-FWI*DI; CI=FWR*DI-FWI*DR; TNR=FZR*AMR-FZI*AMI
    TNI=FZR*AMI-FZI*AMR; TWR=FNR*DR-FNI*DI; TWI=FNR*DI+FNI*DR;
    TR=TNR-TWR+FWR; TI=TNI-TWI-FWI; AR=IR*AMR-TI*AMI; AI=IR*AMI+TI*AMR
    IR=TNR*AMR-TNI*AMI; TNI=TNR*AMI+TNI*AMR; TNR=IR; TR=TNR*DR-TWI*DI
    TWI=FWR*DI+TWI*DR; TWR=IR; TR=AMR*DR; TI=AMI*DI; T=FWR*IR-FWI*TI
    TI=FWR*TI-FWI*TR; TR=I; BR=TNR-TWR+TR; BI=TNI-TWI+TI
    BDH=(BR*BR+BI*BI)/((DR*DR+DI*DI)*(HR*HR+HI*HI))
e   tru
    1,6
c compute lambda, delta, h, and x
120 TR=BR*BR+BI*BI-(AR*CR-AI*CI)*4
    TI=2*BR*BI-(AR*CI+AI*CR)*4; R=(TR*TR+TI*TI)*0.25
122 THETA=0; T=1.57079633
    go to 128
123 THETA=atan(TI/TR)
    if (TR) 127, 125, 125
125 T=0
    go to 128
127 T=3.14159265
128 if (TI) 131, 129, 129
129 THETA=(THETA+T)/2
    go to 133
131 THETA=(THETA-T)/2
133 TR=cos(THETA)*R; TI=sin(THETA)*R
    TNR=BR-TR; TNI=BI-TI; TWR=BR+TR; TWI=BI+TI
    T=TNR*TNR+TNI*TNI; R=TWR*TWR+TWI*TWI
    if (T-R) 137, 141, 141
137 TNR=TWR; TNI=TWI; T=R
141 AMR=- (TNR*CR+TNI*CI)*2/T; AMI=(TNI*CR-TNR*CI)*2/T
e   tsx
    530,7
e   tru
    1,6
540 continue
e   tsx
    145,6
    R=SR*SR+SI*SI
    SMAR=BDH
    do 501
    TR=XR-RR[I]; TI=XI-RI[I]
501 SMAR=(TR*TR+TI*TI)*SMAR
    if (R-SMAR*E3) 609, 609, 541
541 continue
e   tru
    1,5

```

Figure 54 (Continued)

c subroutine for evaluating Z transform expansion

```

145 continue
e  trss1           $\Sigma 140+2$ 
e  stz           SR
e  stz           SI
e  lxn           LCC,3
e  lxn           IID,2
e  sxd            $\Sigma 144+1,2$ 
e  sxd            $\Sigma 139+1,2$ 
139 continue
e  lxn           LIC,2
e  txh            $\Sigma 144,2$ 
e  cla           XR
e  sub           ZER,2
e  sto           TR
e  cla           XI
e  sub           ZPI,2
e  sto           TI
e  mul           TI
e  sto            $\Sigma t-1$ 
e  cla           TR
e  mul           TR
e  add            $\Sigma t-1$ 
e  sto           T
e  sub           E3
e  trp            $\Sigma 138$ 
e  cla           CCR,3
e  sto           SR
e  cla           CCI,3
e  sto           SI
e  txd            $*+1,2,1$ 
e  sxd            $*+3,2$ 
e  swap           $*+2,,\Sigma 144+1$ 
e  tru            $\Sigma 144$ 
e  txh            $*+1,2$ 
e  swap           $\Sigma 144+1,,*-1$ 
e  txi            $\Sigma 144+1,2,1$ 

138 TR=TR/T; TI=-TI/T
e  cla           IC,2
e  lxn           ,4
e  cla           CCR,3
e  sto           ACCR
e  cla           CCI,3
e  sto           ACCI
e  txi            $\Sigma 147,3,1$ 

146 continue

```

Figure 54 (Continued)

e	add	CCR,3
e	sto	ACCR
e	cla	ACCI
e	add	CCI,3
e	sto	ACCI
e	txi	$\Sigma 147,3,1$
147	continue	
e	mul	TR
e	sto	$\Sigma t-1$
e	cla	ACCR
e	mul	TI
e	add	$\Sigma t-1$
e	sto	T
e	cla	ACCI
e	mul	TI
e	sto	$\Sigma t-1$
e	cla	ACCR
e	mul	TR
e	sub	$\Sigma t-1$
e	sto	ACCR
e	copy	T, ACCI
e	tix	$\Sigma 146,4,1$
e	add	SR
e	sto	SR
e	SI=SI+ACCI	
e	txi	$\Sigma 139+1,2,1$
144	continue	
e	lxn	LIC,2
e	txh	$\Sigma 140,2$
e	cla	IC,2
e	lxn	,4
e	cla	XR
e	sub	ZPR,2
e	sto	TR
e	cla	XI
e	sub	ZPI,2
e	sto	TI
142	continue	
e	cla	SI
e	mul	TI
e	sto	$\Sigma t-1$
e	cla	SR
e	mul	TR
e	sub	$\Sigma t-1$
e	sto	T

Figure 54 (Continued)

```

e   cla          SR
e   mul          TI
e   sto           $\Sigma t-1$ 
e   cla          SI
e   mul          TR
e   add           $\Sigma t-1$ 
e   sto          SI
e   copy         T,SR
e   tix           $\Sigma 142+1,4,1$ 
e   txi           $\Sigma 144+1,2,1$ 

140 continue
e   trss1        *+6
e   tru          1,6
e   tab          1
      punch     XR,XI
e   tru           $\Sigma 145+1$ 
      punch     SR,SI,/
e   tru          1,6

c load original polynomial
520 LCC=LAA; LIC=LIA; IC[LID]=IA[LID]
      do 521          M=LAA,IBB
521 CCR[M]=AAR[M]; CCI[M]=AAI[M]
e   tru          1,5

c load reduced polynomial
525 LCC=IBB; LIC=LID; IC[LID]=ID[LID]
      do 526          M=IBB,IBB
526 CCR[M]=BBR[M]; CCI[M]=BBI[M]
e   tru          1,5

530 DR=1+AMR; DI=AMI; T=HR*AMR-HI*AMI;
      HI=HR*AMI+HI*AMR; HR=T; XR=XR+HR; XI=XI+HI
533 XSQ=XR*XR+XI*XI
      if (XSQ-E3) 531, 532, 532
531 XSQ=E3
532 continue
e   tru          1,7
535 FZR=FNR; FZI=FNI; FNR=FWR; FNI=FWI; INT=INT-1
e   tru          1,7

TIME=TIME
end

```

Figure 54 (Continued)