19

INTRODUCTION TO THE SIMULATION OF CONTROL SYSTEMS USING THE ANALOG COMPUTER

by

JOHN WAYNE HERMAN

B.S., Kansas State Teachers College, 1973

A MASTER'S REPORT

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Electrical Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1975

Approved by:

Richard R. Gallagher
Major Professor

ACKNOWLEDGMENTS

I wish to express my appreciation to Dr. Richard R. Gallagher of the Department of Electrical Engineering for his many hours of assistance in completing this paper. His scholarship and understanding contributed to my being able to successfully complete this degree. My thanks also go to Drs. W. W. Koepsel, Donald R. Hummels and Richard F. Sincovec for serving on my committee and for the many hours they spent in class and in private guiding me through the program.

I would also like to thank my wife for her encouragement, patience, and typing ability, all of which were of immeasurable value in my success.

TABLE OF CONTENTS

Chapter Page

Chapter | Page

LIST OF FIGURES

Figure Page

Chapter 1

1. INTRODUCTION

The purpose of this set of experiments is to introduce the student to the simulation of physical systems. Being able to approximate real physical systems by models is an important basis of engineering design since the exact mathematical model of the actual device either doesn't exist or is usually too complicated to yield usable results with a reasonable amount of work. For this reason, a major thrust of most design courses is to acquaint the student with some useful models. This also is the purpose of simulation, to create models of physical systems that yield meaningful results with a minimum of work and error. With this in mind, a definition of simulation for the purposes of these experiments is associated with the creation of a model of a physical system for the purpose of analyzing the operation of that system.

The tools that are used to approximate the real physical systems are the standard mathematical tools of the practicing engineer, those associated with algebraic and differential equations. The analog computer can be used to solve these equations while most of the mathematical operations are done in the LaPlace transform domain. For these reasons the student should have a good practical grasp of differential equations and their relationship to LaPlace transform theory. The physical systems that are simulated relate to simple linear and non-linear feedback control systems. These are presented in most undergraduate engineering curricula and are not so complex that the student loses the intuitive feel for what is happening within the system. It must be remembered that these simple systems in no way approach the limitations of the analog computer or the LaPlace transform techniques. Nor are these the only techniques available to simulate physical systems; however, these tools are very useful in developing a first approximation to a physical system.

These experiments are divided into four major areas. The first of these is designed to familiarize the student with the analog computer. The second section is to use the analog computer to simulate simple position control systems and to demonstrate the effect of changing both the model and the parameters of the model on the open and closed loop response of the control system. The second part also includes a demonstration of the non-linear and delay effects encountered in most practical control systems. The third section treats the improvement of the system using the classical techniques of the system specification, gain, and lead and lag compensation along with the more modern cost function techniques using the state variable formulation, the stability theories of Lyapunov, and basic matrix theory. The last section consists of a short description of some special projects the student may use to expand his knowledge of control theory or analog computer simulation techniques.

The first block of three experiments is designed to acquaint the student with the analogue computer. Experiment I deals with programming the solution of differential equations in the classical and transfer function form. In order to produce an accurate and usable solution to these differential equations this section covers amplitude and time scaling of these differential equations. Experiment II covers this same material with regard to non-linear equations. Experiment III deals with equations in state variable form.

The second block of experiments treats the simulation of simple models of position control systems. Experiment IV demonstrates the open and closed loop responses of types zero, one, and two control systems when excited by step, ramp, and sine wave inputs. This experiment also demonstrates the effect of changing the gain on the response of the system. Experiment V deals with comparison of the model with the actual system and with improving the model. Model improvement may consist of adding circuits that approximate non-linearities in the actual system or with slight changes in coefficients in a linear model creating a more realistic response to the input excitation. Experiment VI deals with the effect of time delay on the operation of simple control systems. The system under consideration is a temperature control system with proportional control and cooling.

The next section emphasizes improvement of control system performance. Experiment VII deals with the subject of compensation from both the classical (frequency domain) point of view and the state variable point of view. The first experiment is a non-lab experiment involving the mathematical operations necessary to compensate the control system. Experiment VIII compares the performance of control systems compensated by classical and state variable techniques to demonstrate the relative advantages and disadvantages of each method and the performance of each with the specifications from both.

The last block consists of a number of descriptions of problems the student may attempt to increase his knowledge of analog computer techniques and control systems. Included in each experiment is a short description of the problem and a few references the student may use to increase his understanding of the problem. This is not intended to show the limitations of the analog computer but to demonstrate some techniques that may be used in conjunction with the analog computer to obtain solutions to problems in engineering.

Chapter 2

2. INTRODUCTION TO THE ANALOG COMPUTER

2.1 EXPERIMENT I: LINEAR EQUATIONS

2.1.1 Purpose

The purpose of this experiment is to acquaint the student with the programming of the analog computer. Upon performing this experiment the student should be able to scale a differential equation and program the equation on the analog computer.

2.1.2 Background Material

The operational amplifiers that are used in the analog computer are capable of performing four linear operations; namely, inversion, multiplication of a variable by a constant, addition, and integration. Additionally, the operational amplifiers that form the heart of the analog computer are not ideal so that they have a finite input impedance, gain, bandwidth, and output voltage. Also a non-zero output impedance, phase shift (except at DC), offset, and noise voltage are characteristic of the amplifiers. These limitations on the performance of the operational amplifiers partially justify this experiment since they contribute to the necessity for amplitude and time scaling.

2.1.3 Magnitude Scaling

Magnitude scaling consists of adjusting the coefficients of the differential equations in such a manner as to decrease the effect of the non-ideal characteristics of the operational amplifiers. This consists of increasing the relative amplitude of those variables that have a small maximum magnitude and to decrease the relative amplitude of those variables that have a large maximum magnitude. The maximum magnitudes are quite often known or may be accurately estimated. These may be measured in the case of an operating system, estimated from the specifications of a system to be built, or estimated from the differential equations.

If the maximum magnitudes of the variables are known or have been estimated then the following procedure may be used to scale the equations describing the system for programming on the analog computer.

(a) Solve the differential equations for the highest order derivative.

(b) Noting that each differential may be obtained by integrating the next highest order derivative and that each amplifier inverts its input to produce its output, write out each of these equations.

(c) Multiply and divide each variable by its maximum magnitude. Do not forget to multiply and divide the initial conditions.

(d) The divisor remains with the variable name and is the scale factor relating the output of the amplifier to the real variable represented by the output of that amplifier.

(e) Solve each of the equations for the scaled variable on the left.

(f) Separate the multipliers of each variable into a pot setting whose value is between zero and one and a gain that is a power of the standard gains that are available on the amplifiers. Pot settings in the range between 0.2 and 0.8 are the most desirable because of accuracy considerations. Time scaling should be used where possible to obtain pot settings in this range. Where time scaling cannot be performed because of limitations of the devices used to record the output or fixed relationships between real time and machine time, multiply-divide circuits are available to bring the pot settings into the desirable range.

(g) Program the resulting equations on the analog computer using the pot settings and gains obtained during the scaling operation. (1:95-108)

2.1.4 Example of Magnitude Scaling

The following worked example may aid in understanding the scaling procedure. Given the equation

$$3\ddot{y} + 2\dot{y} + y = 4 \qquad y(0) = y_0 = 1 \qquad \dot{y}(0) = \dot{y}_0 = 0 \tag{2.1.1}$$

(a) Solve the Equation (2.1.1) for $\ddot{y}$

$$\ddot{y} = -\frac{2}{3}\dot{y} - \frac{1}{3}y + \frac{4}{3} \tag{2.1.2}$$

(b) Write the auxillary equations

$$y(t) - y(t_0) = \int_{t_0}^{t} \dot{y}(t)dt \tag{2.1.3}$$

$$\dot{y}(t) - \dot{y}(t_0) = \int_{t_0}^{t} \ddot{y}(t)dt \tag{2.1.4}$$

(c) Scale the variables by their respective maximums.

$$|y|_{max} = 9 \tag{2.1.5}$$

$$|\dot{y}|_{max} = 20 \tag{2.1.6}$$

$$|\ddot{y}|_{max} = 45 \tag{2.1.7}$$

Equations 2.1.2, 2.1.3, and 2.1.4 then become

$$45\left[\frac{\ddot{y}}{45}\right] = -\frac{2}{3}\left(20\right)\left[\frac{\dot{y}}{20}\right] - \frac{1}{3}\left(9\right)\left[\frac{y}{9}\right] + \left[\frac{4}{3}\right] \tag{2.1.8}$$

$$9\left[\frac{y}{9}\right] - 9\left[\frac{y_0}{9}\right] = \int_{t_0}^{t} 20\left[\frac{\dot{y}}{20}\right] dt \tag{2.1.9}$$

$$20\left[\frac{\dot{y}}{20}\right] - 20\left[\frac{\dot{y}_0}{20}\right] = 45\int_{t_0}^{t}\left[\frac{\ddot{y}}{45}\right]dt \tag{2.1.10}$$

(d) The expressions in square brackets are the outputs of the amplifiers with the possible exception of a plus or minus sign.

(e) Solving each of these equations for the dependent variables on the left yields

$$\left[\frac{\ddot{y}}{45}\right] = \frac{\frac{2}{3} \cdot 20}{45} \left[\frac{\dot{y}}{20}\right] - \frac{\frac{1}{3} \cdot 9}{45} \left[\frac{y}{9}\right] + \frac{\frac{4}{3}}{45} \tag{2.1.11}$$

$$\left[\frac{y}{9}\right] = \frac{20}{9} \int_{t_0}^{t} \left[\frac{\dot{y}}{20}\right] dt + \left[\frac{y_0}{9}\right] \tag{2.1.12}$$

$$\left[\frac{\dot{y}}{20}\right] = \frac{45}{20} \int_{t_0}^{t} \left[\frac{\ddot{y}}{45}\right] dt + \left[\frac{\dot{y}_0}{20}\right] \tag{2.1.13}$$

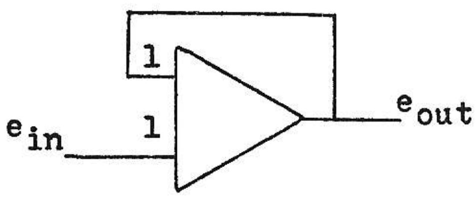
(f) Separating the multiplier of each variable in brackets into a decimal between zero and one and a power of a standard gain yields

$$\left[\frac{\ddot{y}}{45}\right] = -(0.2963) \cdot 1 \left[\frac{\dot{y}}{20}\right] - (0.06667) \cdot 1 \left[\frac{y}{9}\right] + (0.02962) \cdot \left[\frac{1}{1}\right] \tag{2.1.14}$$

$$\left[\frac{y}{9}\right] = (0.2222)\cdot 10 \int_{t_0}^{t} \left[\frac{\dot{y}}{20}\right] dt + \left[\frac{y_0}{9}\right] \tag{2.1.15}$$

$$\left[\frac{\dot{y}}{20}\right] = (0.2250)\cdot 10 \int_{t_0}^{t} \left[\frac{\ddot{y}}{45}\right] + \left[\frac{y_0}{20}\right] \tag{2.1.16}$$

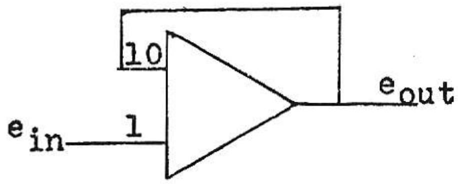
Since the pot settings of the bracketed variable, y, and the forcing function are outside the desirable range, 0.2 to 0.8, either time scaling must be done or a multiply-divide circuit must be used. As stated above, the primary reason this is done is to increase accuracy. If the pot setting is below 0.2, the noise applied to the input of the following amplifier is increased relative to the desired signal and if the pot setting is greater than 0.8 the loading error is increased excessively due to the appearance of a virtual ground at the input of an operational amplifier. If the equation cannot be time scaled, one of the following circuits may be used to obtain a pot setting in the desired range.



$e_{out} = -e_{in} - e_{out}$

or

$e_{out} = -e_{in}/2$



$$e_{out} = -e_{in} - 10e_{out}$$

or

$$e_{out} = -e_{in} / 11$$

Figure 2.1.1 Divide by two and divide by eleven circuits.

Using the divide by eleven circuit for the bracketed variable containing y and the forcing function causes the following changes in Equation 2.1.14.

$$\left[\frac{\ddot{y}}{45}\right] = -(0.2963)\cdot 1\left[\frac{\dot{y}}{20}\right] - (0.7334)\,\frac{1}{11}\left[\frac{y}{9}\right] + (0.3258)\frac{1}{11} \qquad (2.1.17)$$

(g) Using the pot settings and gains found in step six, a scaled computer diagram may be constructed. A note of warning at this point, the signs of the outputs of the amplifiers and the inputs to the following amplifiers must be correct or compensation must be made in the circuit. Beginning the programming with Equation (2.1.15) gives the following scaled computer diagram.

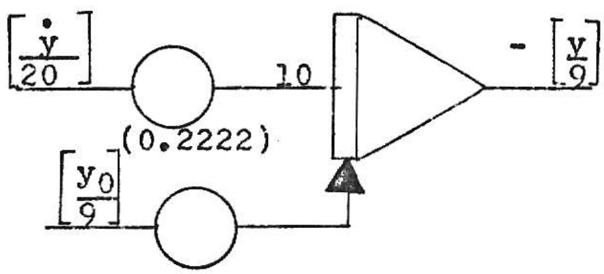


Figure 2.1.2. Scaled Computer Diagram Representing Equation (2.1.15).

The scaled computer diagram representing Equation (2.1.16) is illustrated in Figure 2.1.3.

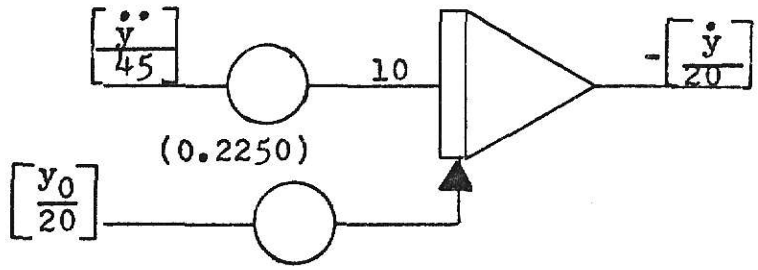


Figure 2.1.3. Scaled Computer Diagram Representing Equation (2.1.16).

The scaled computer diagram for Equation (2.1.17) is shown in Figure 2.1.4.

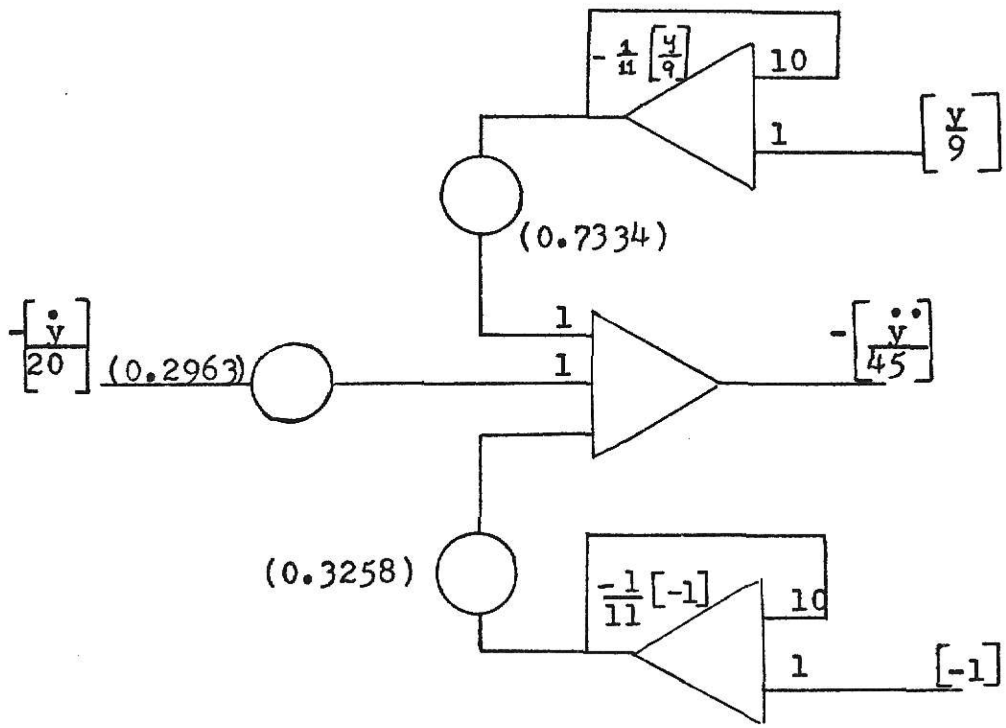


Figure 2.1.4. Scaled Computer Diagram Representing Equation (2.1.17)

Putting the scaled computer diagrams together, matching signs, and reducing the number of amplifiers yields the resulting scaled computer diagram, Figure 2.1.5.

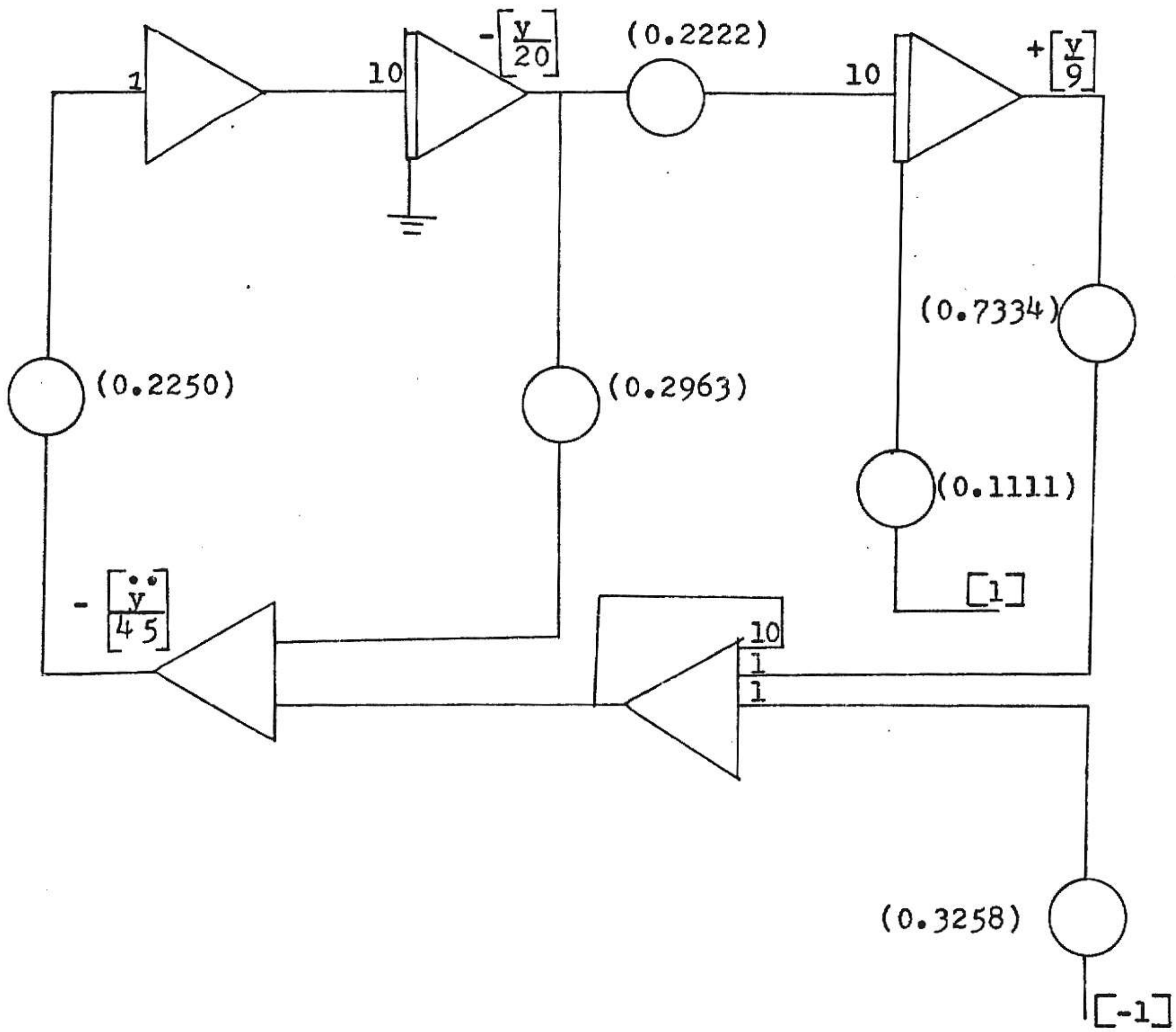


Figure 2.1.5 Complete Scaled Computer Diagram

2.1.5. Time Scaling

Time scaling adjusts the rate of solution of a system of differential equations. At times this adjustment is required in order to use a particular display device, to obtain reasonable solution rates, to reduce the dependence of the solution on high frequency components of the solution that may be distorted by the frequency response of the amplifiers, or to modify pot settings or required gains that lie outside the range of reasonable values. Examination of the operations performed by the analog computer reveals that integration is the only operation that depends directly upon time. Therefore, if we desire to time scale a problem with a time scaling factor, k, given by

$$k = \tau/t \tag{2.1.18}$$

where τ is the machine time and t is the problem time. Only the gains and pot settings associated with the inputs to the integrators need to be modified. Time scaling does not effect either the initial conditions or the maximum magnitudes associated with the problem variables. The modification, then, consists of multiplying the product of the gain and pot setting for each input to the integrators by a factor of 1/k and then developing a new pot setting and a new gain using the method described in Section 2.1.3, step f.

One note of caution, the same time scaling factor must be used for all parts of the problem so that if forcing functions are generated externally or coupled systems of equations are programmed, then these must all be time scaled using the time scaling factor, k.

2.1.6. Example of Time Scaling

Given the scaled computer diagram, Figure 2.1.5, time scale this problem with k = 5. Figure 2.1.5 is repeated here for convenience.

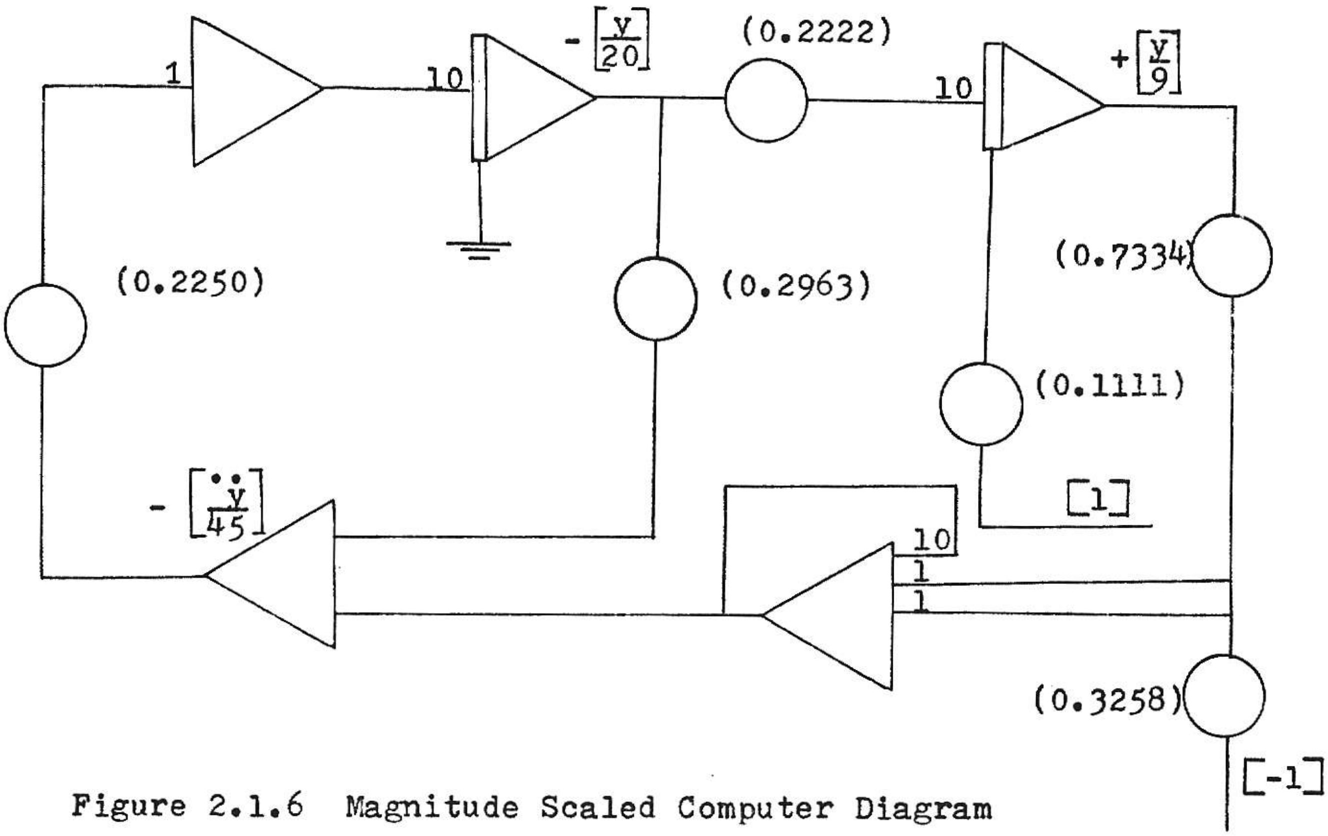


Figure 2.1.6 Magnitude Scaled Computer Diagram

As noted in Section 2.1.5, the time scaling operation consists of multiplying the product of the input gains and pot settings of each input to each integrator by a factor of 1/k, where k is the ratio between machine time and problem (or real) time.

In Figure 2.1.6, there are only two integrators, one producing $[y/9]$ and one producing $[\dot{y}/20]$. The inverter at the input to the integrator producing $[\dot{y}/20]$ is ignored for time scaling purposes so that a pot setting of 0.2250 and a gain of 10 is associated with this input. A pot setting of 0.2222 and a gain of 10 is associated with the input to the integrator producing $[y/9]$. The products of these gains and pot settings are 2.250 and 2.222 respectively yielding 0.4500 and 0.4444 when multiplied by 1/5 . Therefore, a gain of 1 and a pot setting of 0.4500 is used for the integrator producing $\left[\frac{dy}{d\tau}/20\right]$ and a gain of 1 and a pot setting of 0.4444 is used for this integrator producing $[y(\tau)/9]$. Note that these variables are in terms of computer time, τ, instead of problem time, t, and that time scaling did not modify the unreasonable pot settings associated with the input to the summer. The scaled computer diagram is shown on the next page.

THIS BOOK CONTAINS NUMEROUS PAGES WITH DIAGRAMS THAT ARE CROOKED COMPARED TO THE REST OF THE INFORMATION ON THE PAGE.

THIS IS AS RECEIVED FROM CUSTOMER.

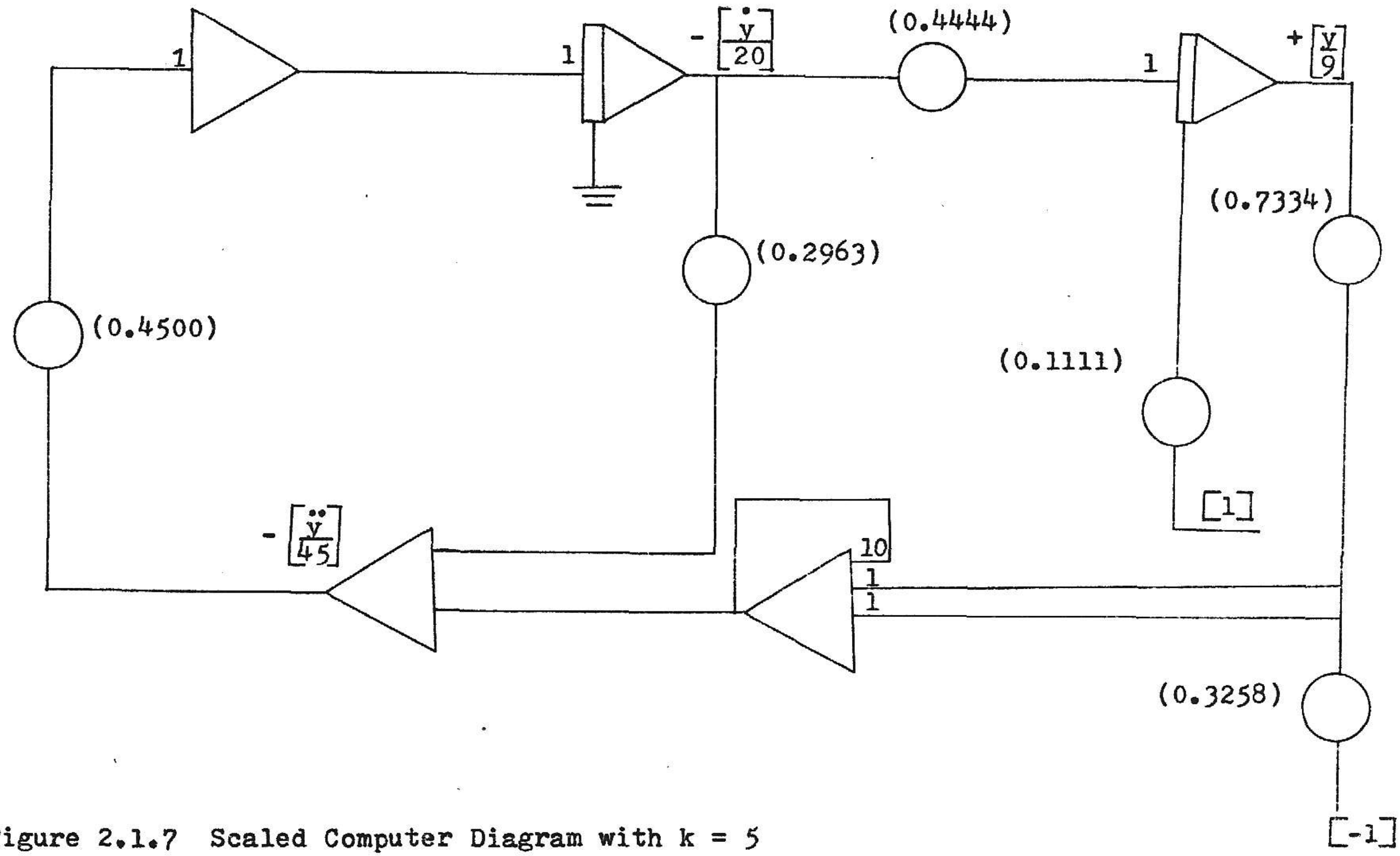


Figure 2.1.7 Scaled Computer Diagram with k = 5

2.1.7 Simulation of Systems Represented by Transfer Functions

A second way of representing the differential equations describing the system is the transfer function. Transfer functions describe the relationship between the input to the system and the output from the system by taking the LaPlace transformation of the system of equations that describe this relationship in the time domain. Generally, the transformed equations are easier to solve than the original time domain equations since they often consist of sums of powers of the transformed variables. For the purposes of this experiment all initial conditions associated with the original time domain equations are assumed to be zero and that the transformed equations may be expressed as

$$D(s)\ Y(s) = N(s)\ X(s) \qquad (2.1.19)$$

where

$$D(s) = s^n + b_1 s^{n-1} + \dots + b_i s^{n-i} + \dots + b_{n-1}s + b_n \qquad (2.1.20)$$

$$N(s) = a_0 s^m + a_1 s^{m-1} + \ldots + a_i s^{m-i} + \ldots +$$

$$a_{m-1} s + a_m \tag{2.1.21}$$

Y(s) is the LaPlace transform of the dependent variable. X(s) is the LaPlace transform of the independent variable and the a_i's and b_j's are real numbers.

Equation 2.1.29 may be expressed in the standard form associated with transfer functions, namely,

$$\frac{Y(s)}{X(s)} = \frac{N(s)}{D(s)} \, . \tag{2.1.22}$$

N(s) and D(s) can both be factored into products of gains and first and second order polynomials with real coefficients. This permits the ratio of N(s) and D(s) to be expressed in the form

$$\frac{N(s)}{D(s)} = \frac{G(s+z_1)(s+z_2)\ \ldots}{(s+p_1)(s+p_2)\ \ldots}$$

$$\frac{(s+z_c)(s^2+e_1 s+f_1)(s^2+e_2 s+f_2)\ldots(s^2+e_d s+f_d)}{(s+p_k)(s^2+r_1 s+t_1)(s^2+r_2 s+t_2)\ldots(s^2+r_L s+t_L)} \tag{2.1.23}$$

(1:193) which may be written as products of terms with the following general forms:

$$\frac{k(s+a)}{(s+b)} \tag{2.1.24}$$

$$\frac{k(s^2+cs+d)}{s^2+es+f} \tag{2.1.25}$$

Factors represented by Equations 2.1.34 and 2.1.35 may be generated on the analog computer and when connected in series produce the desired transfer function. The circuits in Figures 2.1.6 and 2.1.7 may be used to create blocks corresponding to the factors found in the transfer function while the transfer functions in terms of the pot settings are given by the following equations for the output of each amplifier. For first order transfer function blocks, the equations which describe the outputs of the amplifiers in terms of the input to the transfer function block are

$$\frac{y(s)}{x(s)} = \frac{c(s+d - {}^{ab}/_{c})}{s+d-eb} \tag{2.1.26}$$

and

$$\frac{z(s)}{x(s)} = \frac{a - ec}{s+d-eb} \tag{2.1.27}$$

and the equations for the second order transfer functions are

$$\frac{y(s)}{x(s)} = \frac{g\ s^2+(b+e-\frac{ak}{g}-\frac{ck}{g})s+be+\frac{adk}{g}+\frac{chq}{g}-dq-\frac{aeh}{g}-\frac{bck}{g}}{s^2+(b+e-hr-fk)s+be-fhq+dkr-dq-bfk-ehr} \tag{2.1.28}$$

$$\frac{z(s)}{x(s)} = \frac{s(a-gr)+gfq+rck-afh-cq-egr+ae}{s^2+(b+e-hr-fk)s+be-fhq+dkr-dq-bfk-ehr} \tag{2.1.29}$$

$$\frac{w(s)}{x(s)} = \frac{s(gf-c)+ad+chr-bc-afh-rdg}{s^2+(b+e-hr-fk)s+be+fhg+dkr-dq-bfk-ehr} \tag{2.1.30}$$

Note that the gain of each transfer function is a pot setting and is never greater than 1.

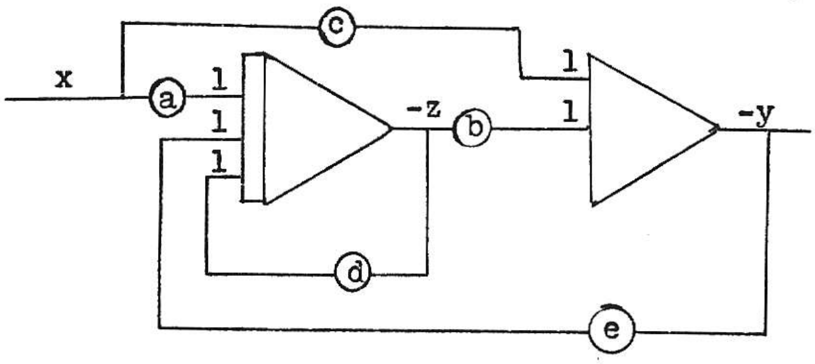


Figure 2.1.8 General First Order Transfer Function Circuit (1:210-211)

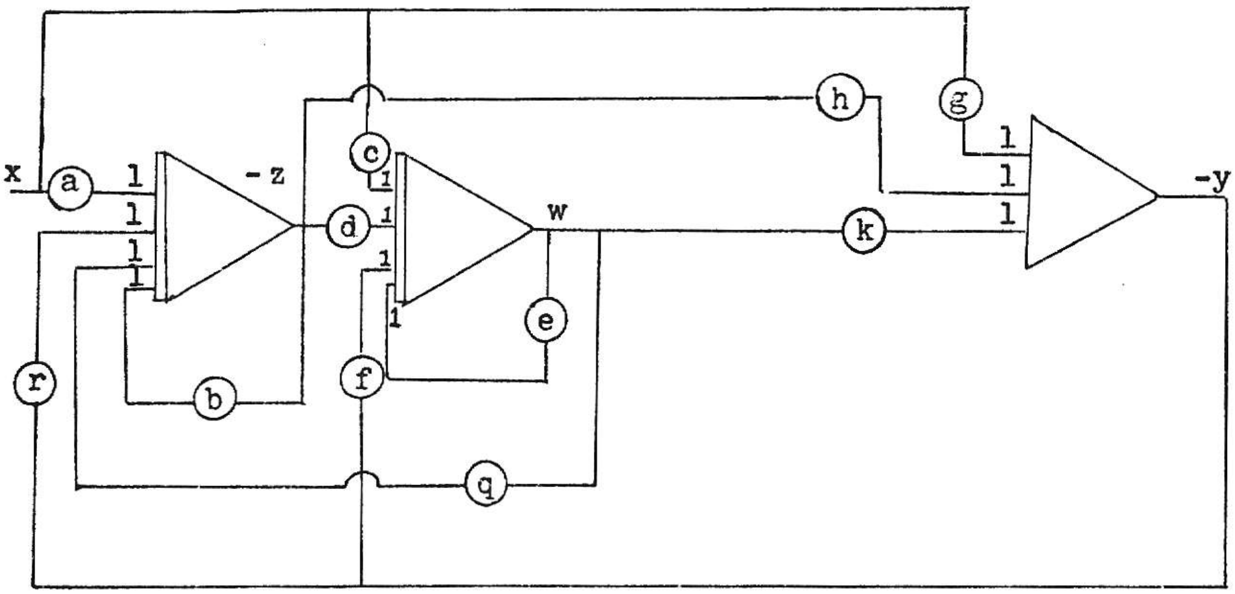


Figure 2.1.9 General Second Order Transfer Function Circuit (1:210-211)

Transfer functions represent differential equations and must be time and amplitude scaled. Differential equations expressed as factored transfer functions have one advantage in that each block may be scaled separately from a knowledge of the input to the block and the output from the block. This scaling may be done in exactly the same manner as that for differential equations since s and s^2 correspond to the first and second derivatives respectively. For example

$$(s^2 + 2s + 1)Y(s) = (3s + s)X(s) \tag{2.1.31}$$

corresponds to

$$\frac{d^2y}{dt^2} + 2\frac{dy}{dt} + y = 3\frac{dx}{dt} + 2x \tag{2.1.32}$$

2.1.8 Checking the Program

The checking of an analog computer program may be divided into four parts; namely, checking the equations, checking the computer diagram and wiring, static checking of the voltages at the outputs of the amplifiers used in the program, and a dynamic check of the amplifiers to ensure that the proper values were used when magnitude scaling. These checks are all relatively easy and for most problems save time by revealing many errors.

The check of the scaled computer equations is similar to the addition of a long column of figures in both directions. It provides a different perspective on the problem which helps reduce repeated mistakes. The checking of the equations entails working backwards through the time and amplitude scaling procedure to produce the original equations.

The diagram and wiring checks are performed after the equation check. The diagram check, like the equation check involves working backwards from the scaled computer diagram to the scaled equations and generally indicates any patching errors. The wiring check is a comparison of the patching of each input and each output from each device on the board with the scaled computer diagram. This check reveals most wiring errors.

In contrast to the above tests, which can be performed without the actual computer, the static and dynamic tests require that the patching panel actually be mounted on the computer and that the pots be set to their correct values. The actual procedure varies with the computer type; therefore, it is not covered here. The operating manual for each computer normally describes the procedure to use for both of these tests. Static checks generally find patching errors missed in the wiring check, incorrectly set pots, and faulty components. The dynamic check, which is done after completion of the static check, reviews the original values used in scaling the equations. If these values are too large or too small, then the variable in question must be rescaled.

2.1.9 Assignment

(a) Program the following equation on the analog computer and display the results on the X-Y plotter.

$$\ddot{x} + 2\dot{x} + 100x = \sin(t) \quad x(0)=1,\ \dot{x}(0)=0$$

(b) Given the following control system, plot the system response to a unit step input on an X-Y plotter.

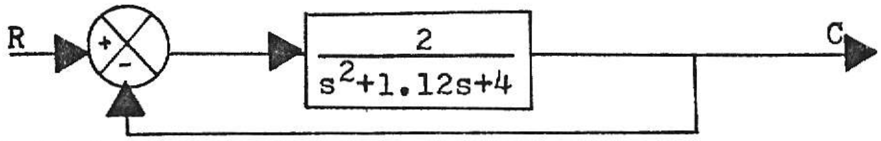


Figure 2.1.10 Second Order Linear System

2.2. EXPERIMENT II: NON-LINEAR EQUATIONS

2.2.1 Purpose

The purpose of this experiment is to introduce the student to the programming of non-linear differential equations.

2.2.2 Background Material

A non-linear differential equation is one in which one or more of the dependent variables or their derivatives appear in a non-linear term in the equation. Examples of non-linear terms are x^2, $\sin(x)$, $1/x$, and $\exp(x)$ where x is the dependent variable. Non-linear equations generally have no analytical solutions which emphasizes the importance of simulation of systems described by these equations on analog or digital computers. Other reasons for their importance are the frequency of their appearance in nature and the properties of their response to input excitations. Some of the interesting responses generated by devices described by non-linear differential equations are jump resonance, limit cycles, the generation of sub harmonics, and output limiting. Non-linear devices may be used to improve the performance of control systems by increasing the speed of response while decreasing overshoot.

2.2.3 Scaling of Non-linear Equations

The major problems in the programming of a non-linear equation is that of estimating the maximum amplitude of the dependent variable, if this is known a priori, its derivatives and the bandwidth required for sufficient accuracy of the output. These values are required before the equations may be scaled for programming on the analog computer. In cases where the various maximum amplitudes and the bandwidth are known, scaling proceeds in the same manner as the scaling of linear equations. The non-linear characteristic or property of the system must also be scaled or the problem has been changed.

If amplitudes are not known, there are several methods of estimating the maximum amplitudes of the variables. Perhaps the simplest of these may be called cut and try. The steps involved in this method are as follows.

(a) A guess is made of the maximum magnitude of each variable and the correct time scaling factor.

(b) The equations are scaled using this guess and programmed on the analog computer.

(c) The program is run on the analog computer. If no amplifiers saturate, step (d) is pursued. If one or more amplifiers saturate then the maximum amplitude assigned to the variable represented by the output of the saturating amplifier is multiplied by a constant greater than one and steps (a) and (b) are repeated.

(d) The program is executed on the analog computer and each variable is plotted. The maximum value of each variable is determined and the problem is once again scaled using the maximum values from the graphs.

(e) If the solution speed is too fast or slow or the pot settings and gains are too large or small then the problem is time scaled to the correct value.

(f) A solution is then obtained.

(g) The speed of solution is reduced by a factor of two or more and the problem is run again.

(h) The solutions obtained in steps (f) and (g) are compared and if there is no difference in the solutions the problem solution is considered complete. If a difference is found then steps (g) and (h) are repeated.

This method is time consuming and tedious but it is the only method that can be used when programming non-linear systems about which little is known. Time spent iterating to find the maximum magnitudes of the variables and the correct time scaling factor can be reduced by any knowledge of the system response.

2.2.4 Non-linear Elements in the Analog Computer

The non-linear elements contained in the analog computer are the diode and the servo-motor driven function generator. The diode is a common electronic device that, ideally, allows current in only one direction while blocking current in the opposite direction. The servo-driven function generator is a simple rotational control system utilizing a servo-motor driving two or more pots coupled by a common shaft. One of the pots is linear and is used to create a voltage which is compared with the input voltage. Thus, the error voltage generated is used to drive the servo-motor. The second pot is tapered in such a way so as to generate the desired output as a function of shaft position. Servo-motor driven function generators have a very severe frequency response restriction, the bandwidth generally being limited to a few Hertz.

Since the only two non-linear elements in the analog computer are the diode and the servo-motor driven function generator, all other non-linear functions are generated using these two elements. Servo-driven function generators accomplish this by changing the pots generating the functions. The diode is used in combination with pots and amplifiers, to produce the desired functions either directly or by using the diode function generator and straight line approximations.

One of the most common functions commonly created using the above non-linear elements is the multiplier. If the multiplier utilizes the servo-driven function generator it is called a servo-multiplier while the multiplier using the diode function generator is called the quarter-square multiplier. The servo-multiplier is generally more accurate while the quarter-square multiplier has a much higher frequency response, the bandwidth generally being a few hundred Hertz or more. Most modern general purpose analog computers use the quarter-square multiplier because of its lower cost and higher frequency response.

Two general rules about the use of multipliers should be mentioned. The first is that no multiplier should be directly driven from a pot because multipliers present a low varying resistance to the source. The second rule is that the output of a multiplier should never be loaded by a pot since the output is correctly scaled if the inputs are correctly scaled. If gain is required following a multiplier then any pots required should be placed in the output of the amplifier. In addition, the inputs to multipliers should be scaled as near one as possible since the multiplier is very inaccurate for small inputs. (1:104-105)

2.2.5 Assignment

(a) Program the following differential equation and plot $\dot{x}$ versus x on the X-Y plotter for a = 0.1, 1, and 2.

$$\ddot{x} + a\left(1 - \frac{x^2}{3}\right)\dot{x} + x = 0 \quad x(0) = 0.1 \quad \dot{x}(0) = 0 \qquad (2.2.1)$$

2.3. EXPERIMENT III: STATE VARIABLE SYSTEMS

2.3.1 Purpose

The purpose of this experiment is to introduce the student to the programming of differential equations appearing in the state variable and transfer function form.

2.3.2 Background Material

The general form of a state variable differential equation is

$$\dot{\bar{x}} = \bar{f}(\bar{x}, \bar{u}, t) \tag{2.3.1}$$

$$\bar{y} = G(\bar{x}, \bar{u}, t) \tag{2.3.2}$$

(2:161), where $\bar{x}$ is the vector of state variables, $\bar{u}$ is the input vector, $\bar{y}$ is the output vector and $\bar{f}$ and $\bar{g}$ are vector functions. In the case of linear systems described by state variable equations, which is all that will be considered in this series of experiments, they may be expressed in state variable form as

$$\dot{\bar{x}} = \bar{\bar{A}}(t)\ \bar{x}(t) + \bar{\bar{B}}(t)\ \bar{u}(t) \tag{2.3.3}$$

$$\bar{y} = \bar{\bar{C}}(t)\ \bar{x}(t) + \bar{\bar{D}}(t)\ \bar{u}(t) \tag{2.3.4}$$

where $\bar{x}$ is the n dimensional state vector, $\bar{u}$ is the m dimentional input vector of inputs, and $\bar{y}$ is the p dimensional output vector. $\bar{\bar{A}}$, $\bar{\bar{B}}$, $\bar{\bar{C}}$, and $\bar{\bar{D}}$ are nxn, nxm, pxn, and pxm matrices respectively.

Differential equations expressed in standard form may be transformed to state variable form using the following algorithm.

Given the standard differential equation form

$$\frac{d^n y}{dt^n} + a_1 \frac{d^{n-1} y}{dt^{n-1}} + \dots + a_n y = b_0 \frac{d^n u}{dt} + b_1 \frac{d^{n-1} u}{du^{n-1}} + \dots u \quad (2.3.5)$$

where a_i's and b_i's may be functions of the independent variable, t, the following procedure is established

(a) Define $\beta_0 = b_0$

$$\beta_1 = b_1 - a_1\beta_0$$

$$\beta_2 = b_2 - a_1\beta_1 - a_2\beta_0$$

$$\beta_3 = b_3 - a_1\beta_2 - a_2\beta_1 - a_3\beta_0$$

$$\vdots$$

$$\beta_n = b_n - a_1\beta_{n-1} - a_2\beta_{n-2} - \dots - a_n\beta_0 \quad (2.3.6)$$

(b) Set $x_1 = y - \beta_0 u$

(c) Then

$$\dot{x}_1 = x_2 + \beta_1 u$$

$$\dot{x}_2 = x_3 + \beta_2 u$$

$$\dot{x}_3 = x_4 + \beta_3 u$$

$$\vdots$$

$$\dot{x}_{n-1} = x_n + \beta_n u$$

and

$$\dot{x}_n = -a_n x_1 - a_{n-1} x_2 - \dots - a_1 x_n + \beta_n u \qquad (2.3.7)$$

(d) The state variable equations become

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \cdot \\ \cdot \\ \cdot \\ \dot{x}_{n-1} \\ \dot{x}_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & 1 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & 0 & 1 & \cdot & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & 1 \\ -a_n & -a_{n-1} & -a_{n-2} & \cdot & \cdot & \cdot & \cdot & a_1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \cdot \\ \cdot \\ \cdot \\ x_{n-1} \\ x_n \end{bmatrix} + \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \cdot \\ \cdot \\ \cdot \\ \beta_{n-1} \\ \beta_n \end{bmatrix} u$$

(2.3.8)

$$y = \begin{bmatrix} 1 & 0 & 0 & \cdot & \cdot & \cdot & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} + \beta_0 u \tag{2.3.9}$$

or using the standard form, Equations 2.3.3 and 2.3.4, (4:676-77)

$$\bar{x} = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix} \qquad \bar{\bar{A}} = \begin{bmatrix} 0 & 1 & 0 & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & 1 & \cdot & \cdot & \cdot & 0 \\ \cdot & & & & & & \cdot \\ \cdot & & & & & & \cdot \\ \cdot & & & & & & \cdot \\ 0 & \cdot & \cdot & \cdot & 0 & 0 & 1 \\ -a_n & -a_{n-1} & a_{n-2} & \cdot & \cdot & \cdot & a_1 \end{bmatrix}$$

$$\bar{u} = \begin{bmatrix} u \end{bmatrix} \qquad \bar{\bar{B}} = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \cdot \\ \cdot \\ \cdot \\ \beta_n \end{bmatrix} \qquad \bar{\bar{C}} = \begin{bmatrix} 1 & 0 & 0 & \cdot & \cdot & \cdot & 0 \end{bmatrix} \qquad \bar{\bar{D}} = \begin{bmatrix} \beta_0 \end{bmatrix} \tag{2.3.10}$$

2.3.3 Example of Conversion to State Variable Form.

Given the equation

$$\dddot{y} + 3\ddot{y} + t\dot{y} + y = \sin t + 4\cos t \qquad (2.3.11)$$

convert this equation to state variable form.

(a) Noting that $\frac{d}{dt}(\sin t) = \cos t$ (2.3.12)

$$\begin{array}{ll} b_0 = 0 & \\ b_1 = 0 & a_1 = 3 \\ b_2 = 4 & a_2 = t \\ b_3 = 1 & a_3 = 1 \end{array} \qquad (2.3.13)$$

which yields

$$\begin{aligned} \beta_0 &= 0 \\ \beta_1 &= 0 - 3(0) = 0 \\ \beta_2 &= 4 - 3(0) - t(0) = 4 \\ \beta_3 &= 1 - 3(4) - t(0) - 1(0) = -11 \end{aligned} \qquad (2.3.14)$$

(b) Set $x_1 = y + \beta_0 u = y$ (2.3.15)

(c) Then

$$\dot{x}_1 = x_2 + \beta_1 u = x_2$$

$$\dot{x}_2 = x_3 + \beta_2 u = x_3 + 4u$$

$$\dot{x}_3 = -a_3 x_1 - a_2 x_2 - a_1 x_3 + \beta_3 u = -x_1 - t x_2 - 3x_3 - 11u \qquad (2.3.16)$$

(d) The equations in state variable form are then

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -t & -3 \end{bmatrix} + \begin{bmatrix} 0 \\ 4 \\ -11 \end{bmatrix} \begin{bmatrix} u \end{bmatrix} \qquad (2.3.17)$$

$$y = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} + 0 \qquad (2.3.18)$$

2.3.4 Programming of Equations in State Variable Form

The programming of equations in state variable form is relatively simple. If the maximum values of the variables and the time scaling factor are known then scaling may be done on the individual equations as was done in Experiment I. Static checks may also be performed.

Scaling of state variable equations is also straight forward. Magnitude scaling is accomplished equation by equation in the same manner as in Section 2.1.3. Time scaling is a simple matrix operation combined with the modification of the independent variable in the forcing function. The equation is scaled like a first order equation noting that a constant multiplied by a matrix is equal to the constant multiplied by every element in the matrix. The following example demonstrates amplitude and time scaling.

2.3.5 Example of State Variable Programming

Given the state variable system

$$\dot{x}_1 = x_2 - 3$$

$$\dot{x}_2 = x_3 + 6$$

$$\dot{x}_3 = -x_1 - x_2 - x_3 - 10 \qquad (2.3.19)$$

with the initial conditions

$$x_1(0) = 0$$
$$x_2(0) = 0$$
$$x_3(0) = 0 \qquad (2.3.20)$$

and maximum values

$$|x_1|\max = 1 \qquad |\dot{x}_1|\max = 0.05$$
$$|x_2|\max = 5 \qquad |\dot{x}_2|\max = 0.25$$
$$|x_3|\max = 11 \qquad |\dot{x}_3|\max = 1 \qquad (2.3.21)$$

The amplitude scaled equations are

$$0.05\left[\frac{\dot{x}_1}{0.05}\right] = 5\left[\frac{x_2}{5}\right] - 3\,[1]$$

$$0.25\left[\frac{\dot{x}_2}{0.25}\right] = 11\left[\frac{x_3}{11}\right] + 6\,[1]$$

$$1\left[\frac{\dot{x}_3}{1}\right] = -1\left[\frac{x_1}{1}\right] - 5\left[\frac{x_2}{5}\right] - 11\left[\frac{x_3}{11}\right] - 10\,[1] \qquad (2.3.22)$$

If the desired output equation is

$$y = x_1 - 2x_2 + x_3 \qquad |y|\ \max = 25 \qquad (2.3.23)$$

then the amplitude scaled output is

$$25\left[\frac{y}{25}\right] = 1\left[\frac{x_1}{1}\right] - 2\cdot 5\left[\frac{x_2}{5}\right] + 11\left[\frac{x_3}{11}\right] \qquad (2.3.24)$$

The equations to be programmed are

$$\left[\frac{\dot{x}_1}{0.05}\right] = (100)\left[\frac{x_2}{5}\right] + (0.6000)\ 100\ [-1] \qquad (2.3.25)$$

$$\left[\frac{\dot{x}_2}{0.25}\right] = (0.4400)\ 100\left[\frac{x_3}{11}\right] + (0.2400)\ 100\ [1] \qquad (2.3.26)$$

$$\left[\frac{\dot{x}_3}{8}\right] = -1\left[\frac{x_1}{1}\right] - (0.5000)\ 10\left[\frac{x_2}{5}\right] -$$

$$(0.1100)\ 100\left[\frac{x_3}{11}\right] + 10\ [-1] \qquad (2.3.27)$$

$$\left[\frac{y}{25}\right] = (0.0400)\left[\frac{x_1}{1}\right] - (0.4000)\left[\frac{x_2}{5}\right] + (0.4400)\left[\frac{x_3}{11}\right] \qquad (2.3.28)$$

Noting that gains and pot settings fall outside the desirable range, the gains and pot settings equations are time scaled with a constant, k, of 10. Noting that the first order equations representing the state variable system are

$$\dot{x} = \bar{A}\bar{x} + \bar{\bar{B}}\bar{u} \qquad (2.3.29)$$

where

$$\dot{x} = dx/dt = (dx/d\tau)(d\tau/dt) \qquad (2.3.30)$$

and that

$$\frac{d\tau}{dt} = k. \qquad (2.3.31)$$

Equations 2.3.29, 2.3.30 and 2.3.31 gives

$$k\dot{x} = \bar{A}\bar{x} + \bar{\bar{B}}\bar{u} \qquad (2.3.32)$$

or

$$\dot{x} = \frac{1}{k}\ \bar{A}\bar{x} + \frac{1}{k}\bar{\bar{B}}\bar{u}\ . \qquad (2.3.33)$$

Using equation 3.3.32 with the scaled computer equations gives the equations to be programmed:

$$\left[\frac{\dot{x}_1}{0.05}\right] = 10\left[\frac{x_2}{5}\right] + (0.6000)\,10\left[-1\right]$$

$$\left[\frac{\dot{x}_2}{0.25}\right] = (0.4400)\,10\left[\frac{x_3}{11}\right] + (0.2400)\,10\left[1\right]$$

$$\left[\frac{\dot{x}_3}{1}\right] = -\,0.1000\left[\frac{x_1}{1}\right] - (0.5000)\left[\frac{x_2}{5}\right] - (0.1100)10\left[\frac{x_3}{11}\right] + 1\left[-1\right]$$

(2.3.34)

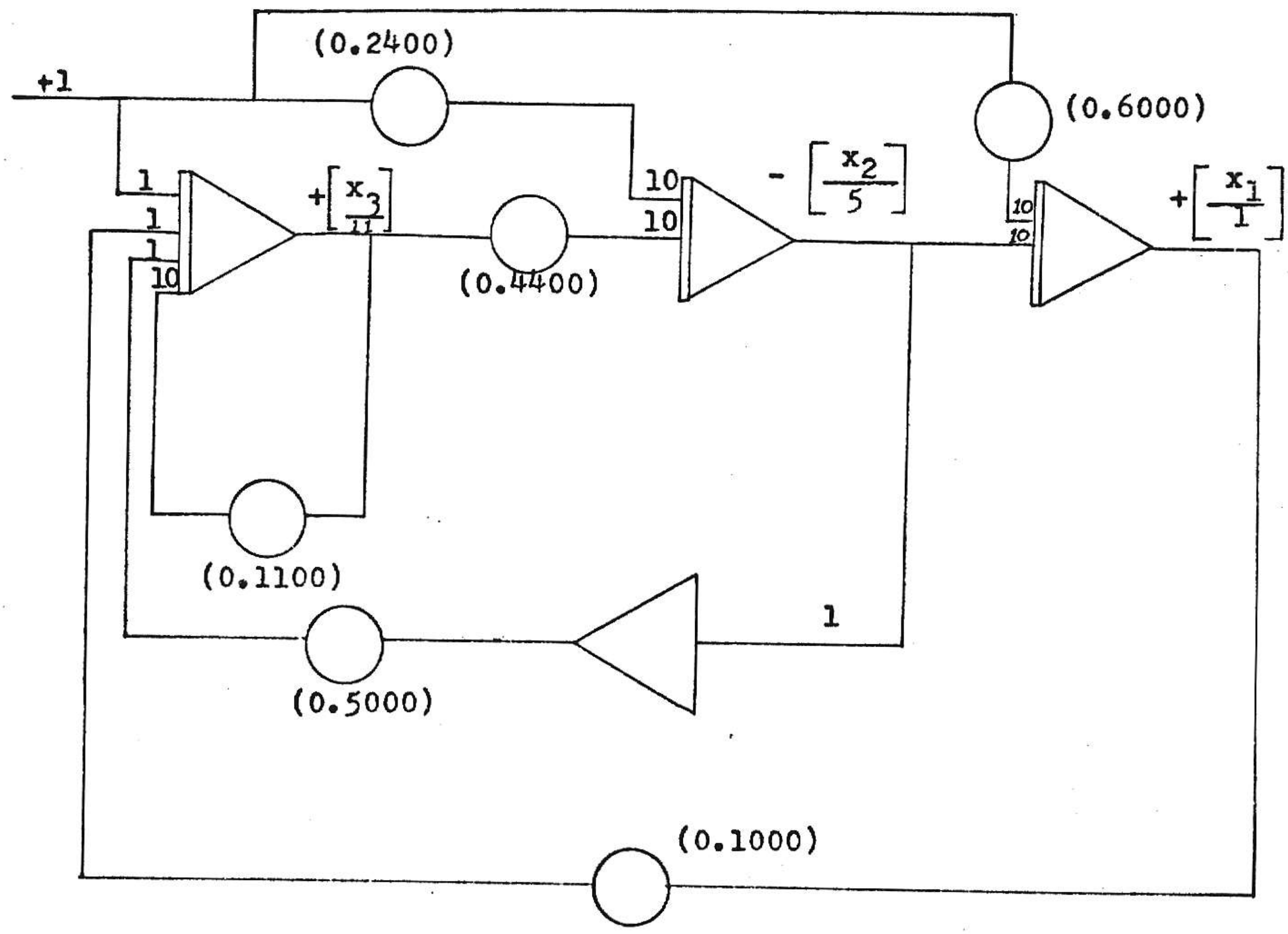


Figure 2.3.1 Scaled Computer Diagram for Example of State Equations

2.3.6 Assignment

Given the equation, contribution of Bessel,

$$t^2\ddot{x} + t\dot{x} + t^2x = 0 \quad x(0) = 1 \quad \dot{x}(0) = 1 \qquad (2.3.35)$$

reduce this equation to state variable form and program this on the analog computer for $0 \leq t \leq 5$.

Chapter 3

3. SIMPLE CONTROL SYSTEMS

3.1 EXPERIMENT IV: TYPE N SYSTEMS

3.1.1 Purpose

The purpose of this experiment is to acquaint the student to the open and closed loop response of type zero, one and two feedback systems to different inputs with various loop gains.

3.1.2 Background

Type N systems may be defined as those with open loop transfer functions of the general form

$$G(s)\,H(s) = \frac{A(s)}{s^N B(s)} \qquad (3.1.1)$$

where N is the order of the pole at the origin or the difference between the order of the zero and the order of the pole at the origin, A(s) and B(s) are polynomials in s. Poles at the origin tend to dominate stable system response, i.e. slowing system response to input excitations and reducing system bandwidth. Type N systems, however, are the only systems with transfer functions in the form of rational fractions that will produce zero steady state error for inputs with the transform $(1/s)^n$ with $N \geq n$

and finite steady-state error for $N \geq n-1$.

Simulation of type N systems is relatively straight forward. Given a control system with the general form as shown in Figure 3.1.1

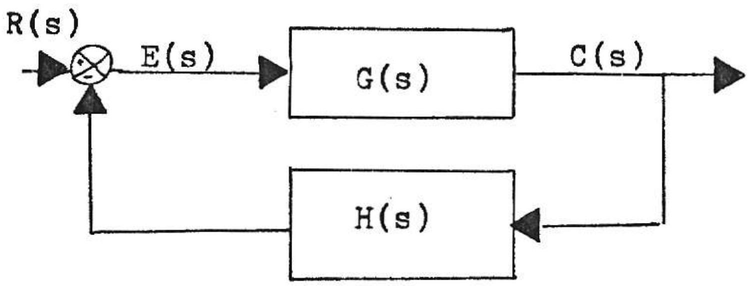


Figure 3.1.1 Single Loop Feedback Control System

the blocks are separated and programmed separately. Factors of the form $(1/s)^c$ are programmed as c integrators connected in series with the initial condition terminal grounded. The remainder of each block is programmed as in section 2.1.8. Gains greater than ten may be required in addition to time scaling. These large gains may be required in Type N systems to generate the required steady state error.

3.1.3 <u>Assignment</u>

(a) For the following type zero system, plot the response versus time for a unit step input and a unit ramp input with K = 1, 2, 3.

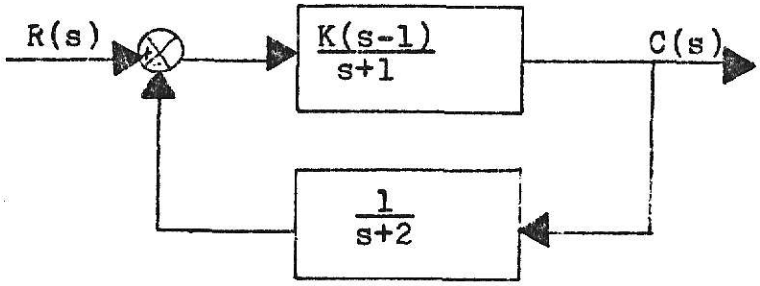


Figure 3.1.2 Type Zero System

(b) Open the loop between the feedback element and the summer and repeat the above experiment plotting the output of the feedback element for K=1, 2, 3 and unit step and unit ramp inputs. Ramp time need not exceed 10 seconds.

(c) For the following type one system plot the open and closed loop responses to unit step and ramp input excitations with K=1, 3, 5.

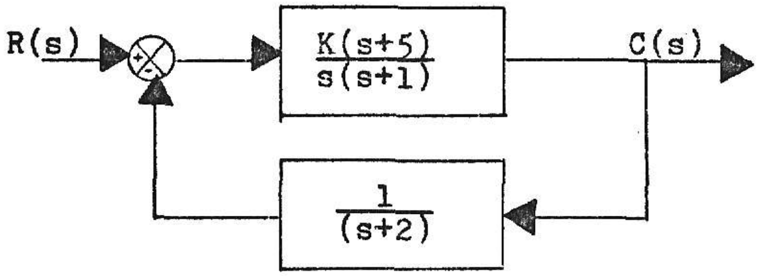


Figure 3.1.3 Type One System.

(d) For the following type two system, plot the open and closed loop response to the unit step and unit ramp inputs with K=1, 17, 18.

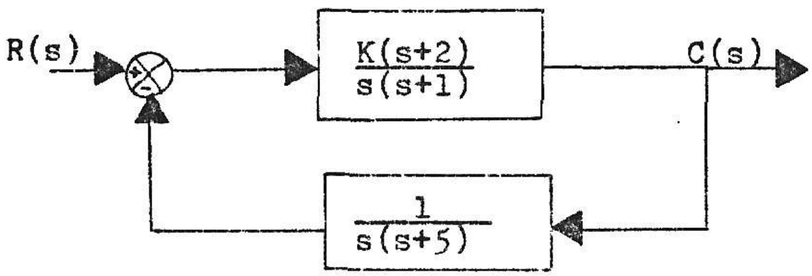


Figure 3.1.4 Type Two System

3.2 EXPERIMENT V: REAL TIME SIMULATION AND MODEL IMPROVEMENT

3.2.1 Purpose

The purpose of this experiment is to introduce the student to real time simulation of physical systems, including the comparison of the real system to the analog computer model using different error measures and model improvement.

3.2.2 Background

At some point in the design of a system it may prove desirable to compare the performance of the model with the actual system. The comparison is generally performed near the middle of the design sequence to check the first approximation to the actual system and to aid in the formation of a more accurate, possibly non-linear, model. This comparison necessitates a time scaling factor of one to allow the comparison to be made (the actual system cannot be time scaled), the selection of a suitable input to both the model and the system to produce meaningful outputs, and the selection of a measure or norm of the difference between the model response and the system response. The choice of input excitation and error measure to obtain the best results is a complicated procedure, thus the mechanics of the selection process is not presented here. At this point an attempt may be made to improve the model with respect to the actual system by using parameter search techniques. Once again, only the simplest search techniques are included here. Quite often a non-linear model is required to provide sufficient model accuracy. However, parameter search techniques, while applicable, may be much more difficult than the measurement of the non-linearity and the generation of the non-linearity via the diode function generator or other special circuits.

3.2.3 Real Time Simulation

Simulation of system response in real time means that the equations describing system performance are not time scaled prior to their programming on the analog computer. Not being able to time scale the equations quite often leads to unreasonable pot settings and amplifier gains. The divider circuits described in Experiment I may be used to aid in correcting unreasonable pot settings; however, nothing may be done in the case of unreasonable gain values. A criterion for the choice of display device must be the system response time rather than choosing the time scaling factor to match the desired display device. If the speed of the actual system is such that the analog computer bandwidth is exceeded then something different must be done. Magnitude scaling is done in the same manner as described in Experiment I.

3.2.4 Choosing the Input Excitation and Error Measure

Ideally for a thorough evaluation of the system's response the input excitation should include all possible inputs to the system. Practically, this cannot be done because of the finite nature of the error integrator so that some representative subset of all of the possible inputs must be chosen. In addition, the excitation should be chosen so that excessively accurate timing of the input excitation is not required in the case of multiple runs. This severely limits the choice of input excitations. Types of excitations normally chosen are the impulse, step, ramp, sinusoid and the random or pseudo-random wave form. The random waveform is not normally chosen for multiple runs because of repeatability, a random waveform is not normally repeatable so that the accumulated error from one run to the next must be analyzed statistically. However, the random waveform is the most general form of input excitation and provides the most accurate results.

The second best input excitation in terms of accuracy is the pseudo-random wave form and if the wave form generator is properly designed, timing is handled in the generator. The wave form commonly chosen is the step due to the ease of its generation and timing in addition to the information generated by its application to the system and model.

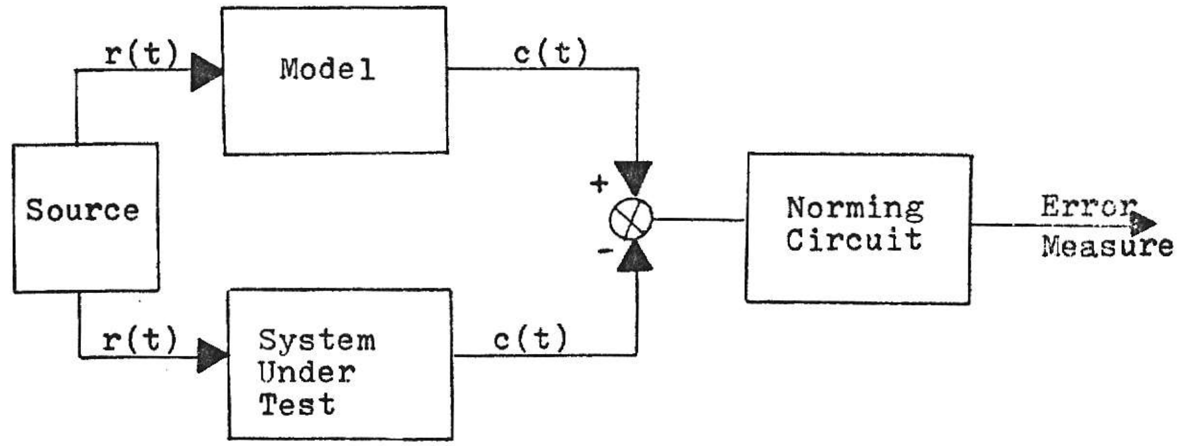


Figure 3.2.1 General Error Measuring Circuit

The next part to be selected is error measure or norm. The norm of a continuous system has the general form

$$\left|\left|u\right|\right|_N = \left[\int_0^T |u|^N \, dt \right]^{1/N} \tag{3.2.1}$$

Normal values for N are 1, 2 and ∞, however, any positive real value may be chosen. The larger the value of N chosen the greater the weight given to large errors until for $N = \infty$, the measure operation yields the largest value. The normal values of N are chosen partly because of the ease with which they may be implemented on the analog computer in addition to the well developed theory behind them.

The circuit for comparing the model with the device is shown in Figure 3.2.1. The error measuring circuits

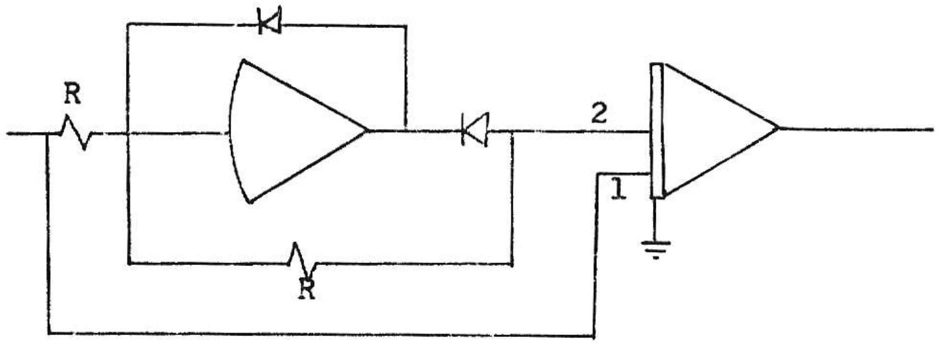


Figure 3.2.2 Circuit for Generating Norm with N=1

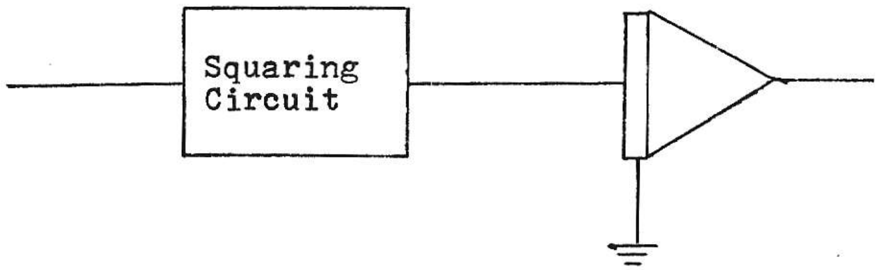


Figure 3.2.3 Circuit for Generating Norm with N=2

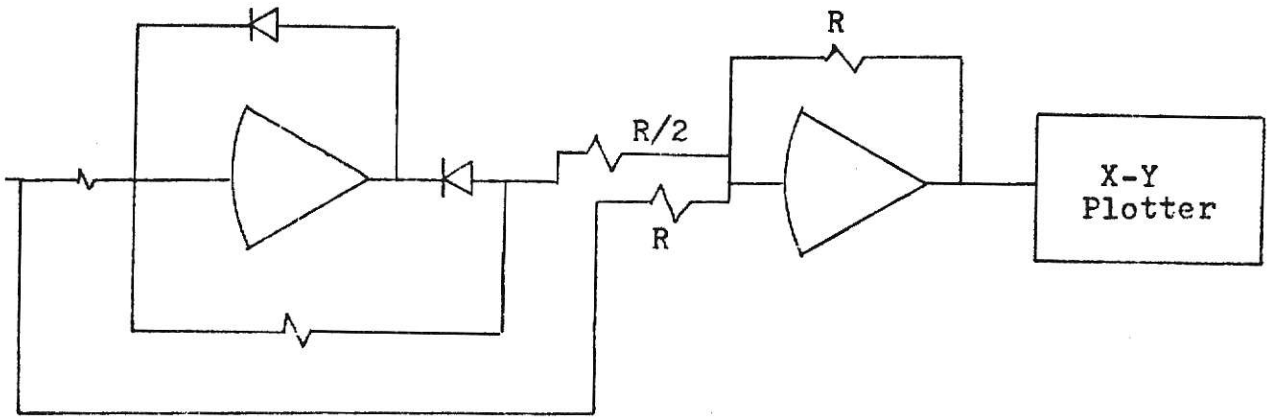


Figure 3.2.4 Circuit for Generating Norm with $N = \infty$

for N equal to one, two, or infinity are shown in Figures 3.2.2, 3.2.3, and 3.2.4. Of the normal values of N the most common choice is N equal to two yielding what is commonly thought of as the distance between two points. This norm is perhaps the best for general use.

3.2.5 Model Improvement

Since most physical systems are non-linear while most models are linear, dramatic improvements in model accuracy may be obtained by using non-linear circuits to approximate the non-linearities in the system. Some of the more important of these, especially those involved with control systems, are covered in Section 3.2.6. Non-linearities are normally measured and analyzed graphically or approximated from equations describing the input-output relationship.

Linear models may also be improved by slight changes in the model parameters causing the input-output relationship of the model to more closely approach that of the system. The technique is simple to explain though very long and tedious in its execution. An outline of the procedure follows.

(a) First a table is created showing different values of each parameter to be tested. These values generally do not range more than a few percent on each side of the original value. The best choice of values is probably elements from the sequence 1, 2, 3, 5, 8, 13, ... , multiplied by some small constant with these values added to and subtracted from the nominal value. The sequence of integers is the Fibonacci sequence.

(b) The norm of the difference between the response of the system and the response of the model is recorded for all combinations of values.

(c) The combination yielding the smallest error is chosen as the new model parameters. The model may be further improved by repeating steps (a) through (c) with a smaller constant chosen in step (a).

3.2.6 Common Types of Non-linearities Found in Control Systems

The most common non-linearities found in systems are dead band, limiting, backlash, and friction. These non-linearities normally degrade system performance. Dead band, friction, and backlash often increase steady-state error by reducing the ability of the system to respond to small error signals. Limiting reduces the ability of the system to respond to large error signals or control signals by constraining the speed of response or the length of travel of the system.

Dead band is a region of the input-output response of a component of a system where the output of the component does not change for changes in the input. A component with a dead band will have an input-output response like that pictured in Figure 3.2.5.

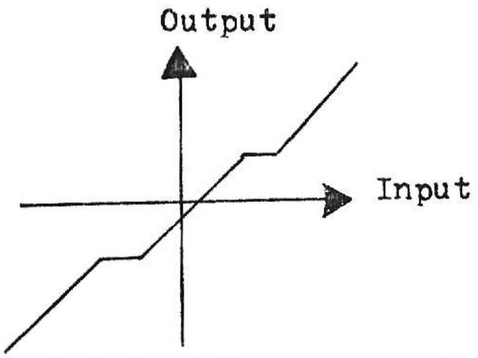


Figure 3.2.5 Input-Output Response of a Component with Dead Band

Dead band most commonly occurs near the zero crossing point of the input-output response but, as is evident from Figure 3.2.5, need not be confined to this region. There may be more than one dead band in the response of a component. These dead bands need not be symmetrical with respect to the axis.

Limiting resembles a dead band that occurs for all values for all inputs that have large magnitudes. The input-output relationship of a device with a limit is shown in Figure 3.2.6.

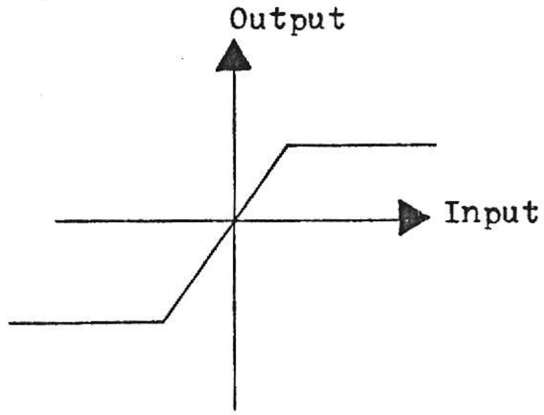


Figure 3.2.6 Response of a Component with Limiting

As with dead band, limiting need not be symmetrical with respect to the axis.

Backlash in systems is generally caused by the energy storage in the transmission system joining the source and load. Examples of this are the energy stored in the shafts in a gear train or the energy stored in a transmission line. When this energy is released, the output of the system is pushed past the desired end point introducing a small error. A component with backlash may introduce a limit cycle in a system with high loop gain. The input-output response curve for a component with backlash may resemble that shown in Figure 3.2.7.

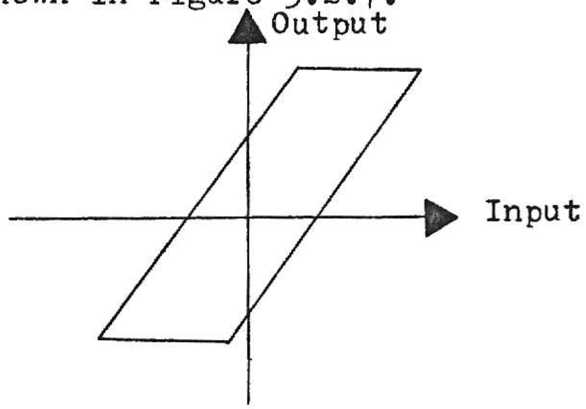


Figure 3.2.7 Response of a Component with Hysteresis

Friction is a force inherent in the system that opposes motion. It is generally velocity dependent but is often approximated by a force that is not velocity dependent but always opposes motion. Non-velocity dependent friction is called Coulomb friction. The magnitude of the Coulomb frictional force is determined using the coefficient of dynamic friction. The relationship between the Coulomb frictional force and the velocity is shown in Figure 3.2.8.

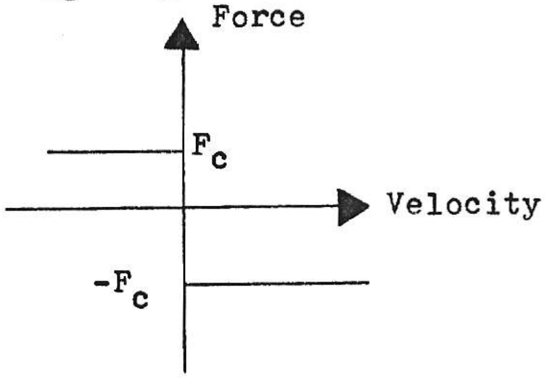


Figure 3.2.8 Response of a Component with Coulomb Friction

Of these non-linearities, the two that are easiest to measure and to simulate are dead band and limiting. Dead space and limiting are also the most commonly programmed of all non-linearities while friction and backlash are generally ignored. Circuits for generating these and other non-linearities may be found on pages 372 to 379 of the Handbook of Analog Computation (1:372-379). Figures 3.2.9 and 3.2.10 showing computer circuits for simulating limiting and dead space are shown below. e_R is the computer reference voltage corresponding to one machine unit.

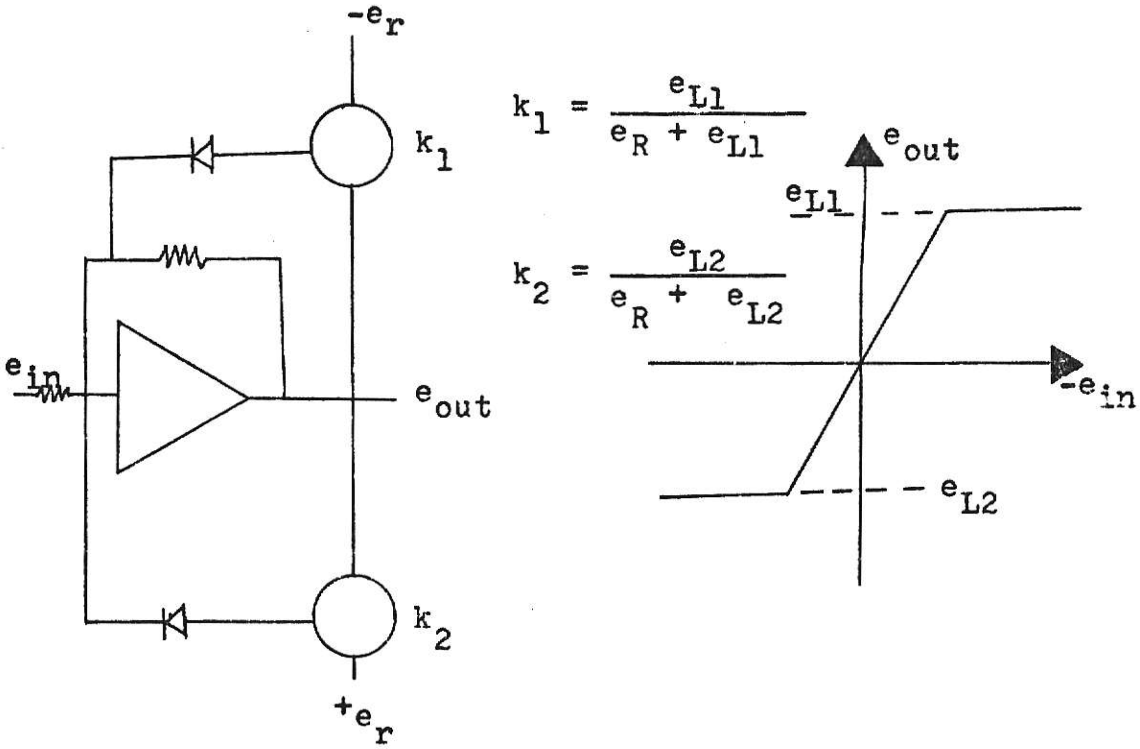


Figure 3.2.9 Limiting Circuit and the Relationship between e_{in} and e_{out}

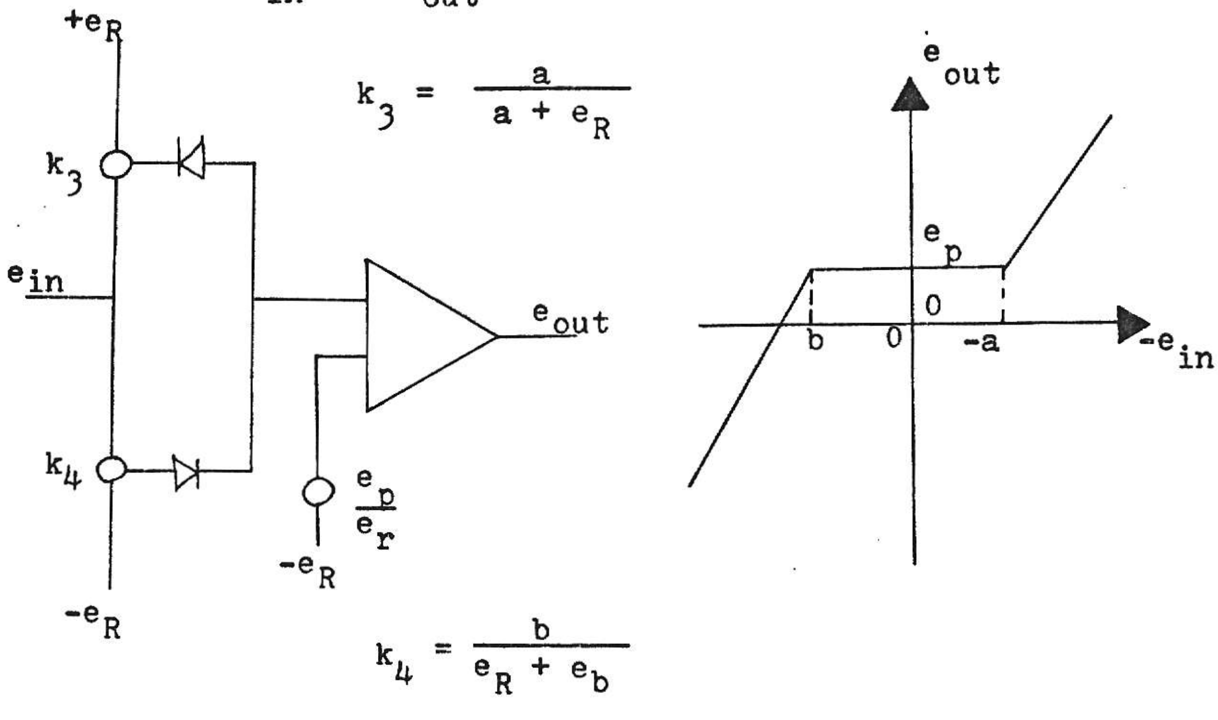


Figure 3.2.10 Dead Band Circuit and the Relationship between e_{in} and e_{out}

3.2.7 Assignment

Input-output relationships for each element in a system is provided the student by the lab instructor. The student then constructs a linear and a non-linear model of the system, scale and program these models, and compare these to the original system.

3.3 EXPERIMENT VI: SYSTEMS WITH TRANSPORT DELAY

3.3.1 Purpose

The purpose of this experiment is to demonstrate the effect of transport delay on system performance.

3.3.2 Background

Transportation delay is a common phenomenon in control systems and may be defined by the equation

$$y(t) = x(t - T) \qquad (3.3.1)$$

where T is the time delay. Time delay appears in linear systems in expressions containing factors of the form e^{-Ts}. In root-locus plots, the expression e^{-Ts} represents a pole at minus infinity and a zero at plus infinity ($T > 0$) with the angle associated with this pole being given by

$$\angle e^{-Ts} = -\omega T \text{ radians} \qquad (3.3.2)$$

(4:347-349) where $s = \sigma + j\omega$ (3.3.3)

Equation 3.3.2 indicates that the phase angle associated with a transportation delay is a function of the frequency. For Nyquist plots, this expression indicates that there will be an infinite number of encirclements of the origin as ω approaches infinity. For Bode plots, the angle approaches minus infinity as ω approaches infinity.

The major problem with transportation lag is in the area of stability, particularly systems with high loop gains. Consider the simple control system with

$$G(s)H(s) = \frac{K}{S(S+1)} . \tag{3.3.4}$$

As is well known, this system is stable for all positive values of K. The addition of a time delay in the open loop with T being one second gives

$$G(s)H(s) = \frac{Ke^{-s}}{S(S+1)} . \tag{3.3.5}$$

From the equations used to generate the Nyquist plot, the values of ω for points at which the curve crosses the negative real axis are given by

$$-\pi/2 - \tan^{-1}\omega - \omega = (2n+1)(\pi) \tag{3.3.6}$$

$$n=0, \pm1, \pm2, \ldots$$

The value of ω for negative real axis crossings for n=-1 is given by

$$\omega + \tan^{-1}\omega + 3\pi/2 = 0 \tag{3.3.7}$$

Equation (3.3.7) is immediately recognizable as an equation which has more than one solution. One value of ω for which a solution exists is 0.86. The magnitude, as a function of loop gain, K, of the transfer function for $\omega = 0.86$ is given by

$$|G(j\,0.86)\,H(j\,0.86)| = \sqrt{\frac{K}{(0.86^2)(1+0.86^2)}} \tag{3.3.8}$$

or

$$|G(j\,0.86)H(j\,0.86)| = \sqrt{K}\,/\,1.13 \tag{3.3.9}$$

Equation (3.3.9) indicates that encirclement of the point, (-1,0) occurs if the open loop gain exceeds 1.28. A transport delay of one second is much larger than that encountered in most control systems, the typical delay being at most, a few milliseconds. Small transport delays can often be approximated in the complex frequency domain by using a truncated MacLaurin or Taylor series expansion.

3.3.3 Temperature Control Systems

Temperature control systems are those that attempt to maintain the temperature of a specified volume within a certain range about a nominal value. The major difficulties encountered in the analysis of temperature systems are the gain or loss of heat to the surroundings through conduction, convection, and radiation and the nature of heat transfer within the volume being heated or cooled. Heat gain or loss to the surroundings generally means that some control action must occur even though the input excitation to the system has not changed. Multiple transport delays are introduced by heat transportation occurring through combinations of conduction, convection, and radiation and by different path lengths between the points of application of heating or cooling and the temperature sensor. These multiple delays are generally modeled for analysis purposes by assuming that all heat is transported using the mechanism that transports the most heat to the sensor and the path over which most of the heat is transported.

3.3.4 Simulation of Time Delays

Transport delays on the analog computer use approximation to e^{-Ts}. These approximations are generally in the form of rational fractions and are derived from the MacLaurin series expansion in such a manner that reduces the number of amplifiers and pots. These circuits may still require amplitude scaling and all will require time scaling. The circuits recommended in the EAI Handbook of Analog Computation (1:227-228) are perhaps the best available since they do not require amplitude scaling for inputs that are properly scaled. Time scaling is still required and gains and pot settings must be adjusted to give reasonable values if possible.

3.3.5 Assignment

Determine the gain required for an external power amplifier-driving the heating element of an oven holding a crystal oscillator. The desired internal temperature of the oven is $65^\circ C \pm 1^\circ C$. The oven and the temperature sensing elements within the oven have the following parameters.

Oven: Dimensions 10cm. x 10cm. x 5 cm

Thickness of walls: 0.5cm

Thermal conductivity of walls: 0.01 watts/meter sec $^\circ C$

Convection coefficient:

external walls - 100 watts/meter^2 sec $^\circ C$

internal walls - 100 watts/meter^2 sec $^\circ C$

Heating capacity of oven: 1 watt/ $^\circ C$

Internal time delays: 4 seconds for convection

25 seconds for conduction

Internal heat transfer: 80% by convection

20% by conduction

Heating element: 2.5 ohm, 10 watt resistor

Heat dissipated by the contents of the oven: 0.1 watts

The internal temperature sensing network produces a 10mV/ $^\circ C$ error signal for each degree of difference between the temperature of the sensor and the desired temperature, $65^\circ C$. Outputs of both polarities are available from the temperature sensor.

Assume no loss of heat from the outside surface of the oven by radiation or conduction and that the heat capacity of the contents of the oven may be neglected.

When the crystal oven has reached steady state, apply a simulated change of ten degrees in the ambient temperature and plot the resulting internal temperature of the oven.

Chapter 4

4. OPTIMIZATION

4.1 EXPERIMENT VII: COMPENSATION

4.1.1. Purpose

The purpose of this experiment is to introduce the student to the compensation of control systems using classical and state variable techniques.

4.1.2. Background Material

The purpose of compensation of control systems is to improve the system response. This improvement may be required to stabilize the system or to meet design specifications. There are two major techniques of compensating a control system, the classical technique using the LaPlace or Fourier transforms of the system time response and the state variable technique using the time response of the system. Both, if performed properly, will fulfill the objective.

4.1.3. Classical Techniques of Compensation

Classical compensation consists of adjusting the position of the closed loop poles so that specifications are met or the system is stabilized. The position of the closed loop poles may be altered by changing the loop gain or by the addition of poles and zeroes to the open loop transfer function.

The specifications that a control system must meet are generally concerned with three things, namely speed of response, relative stability, and system accuracy. Specifications may be written in terms of the time domain response or the frequency domain response of the system. Specifications are sometimes given in terms of both the time domain response and the frequency domain response. Special care must be taken in this case because the required performance may be impossible to achieve.

For second order unity feedback systems, there are expressions relating the frequency domain response characteristics to the time domain response characteristics. This fact along with the mathematical simplicity are the reasons for the importance of the second order unity feedback system. Since the design of the system using classical techniques occurs in the complex frequency domain, the relationship of time domain and frequency domain characteristics derived using the equations describing a unity feedback second order closed loop system are used to translate time domain specifications to the frequency domain.

Common time domain specifications include the following:

(a) Overshoot - the maximum difference between the transient and steady-state response for a unit step input measured after the response passes through one for the first time.

(b) Rise Time - the time required for the response to a unit step input to rise from 10% of its final value to 90% of its final value.

(c) Settling Time - the time required for the response to a unit step function to reach and remain within a specified range around its final value.

(d) Time Constant - the time required for the envelope of the transient response to a unit step input to reach 63% of its final value.

(e) Delay Time - the time required for the response to a unit step input to reach 50% of its final value.

(f) Peak Time - the time required for the response to reach the first peak of the overshoot.

The quantity which tie the two sets of specifications together is the damping ratio. The equations relating the time domain response of the system to this quantity may be derived from the equation describing a second order system. Given the following closed loop transfer function

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2+2\omega_n s+\omega_n^2} \qquad (4.1.1)$$

(4:435) the response for a unit step input is given by

$$c(t) = 1 - e^{-\xi\omega_2 t}\left(\cos\omega_d t + \frac{\xi}{\sqrt{1-\xi^2}}\sin\omega_d t\right) \quad (4.1.2)$$

(4:435) where $\omega_d = \omega_n\sqrt{1-\xi^2}$ = damped natural frequency. (4.1.3)

The maximum value of the overshoot, M_p, may be found from the expression

$$M_p = e^{-\frac{\xi\pi}{\sqrt{1-\xi^2}}} \quad (4.1.4)$$

(4:435). The system time constant, τ, is given by

$$\tau = \frac{1}{\xi\omega_n} \quad (4.1.5)$$

(3:183). If the system is required to settle to within p percent of the final value then the maximum settling time, T_s, is given by

$$T_s = -\tau\log_e\left(\frac{p}{100\%}\right) \quad (4.1.6)$$

The rise time, T_r, is given by

$$T_r = -\frac{1}{\omega_d}\tan^{-1}(\tau\,\omega_d) \quad (4.1.7)$$

(4:235). The peak time, T_p, is given by

$$T_p = \frac{\pi}{\omega_d} \quad (4:235) \tag{4.1.8}$$

Equations 4.1.3 through 4.1.8 may be solved to determine an expression for ξ in terms of ω_n that will be required to meet system specifications. The required system phase margin, ϕ_m, may be found using the expression

$$\phi_m = \tan^{-1} \frac{2\xi}{\sqrt{\sqrt{1 + 4\xi^4} - 2\xi^2}} \tag{4.1.9}$$

Common frequency domain specifications include:

Gain Margin - the reciprocal of the open loop gain at the frequency where the phase difference between the input excitation and output response of the system differs by 180^o. (Given in magnitude or db).

Phase Margin - 180^o plus the phase difference between the input excitation and output response at the frequency where the open loop gain is equal to one.

Bandwidth - the band of frequencies over which the system remains usable (within 3db of midband gain).

Roll-off - the rate at which the gain of the system decreases with increasing frequency usually expressed in decibels per decade of frequency change at one or more frequencies of interest.

Resonance Peak - the maximum value of the magnitude of the closed loop response.

Resonant Frequency - the frequency at which the resonance peak occurs. (3:180-182).

One performance specification that may be considered to be either a time domain specification or a frequency domain specification is steady state error. This specification along with the waveform that is used to determine its value gives the minimum open loop gain of the system. The steady state error coefficient may be determined using the expression

$$e_{ss} = \lim_{s \to 0} \left[\frac{sR(s)}{1 + G(s)H(s)} \right] \qquad (4.1.10)$$

(4:284) where R(s) is the LaPlace transform of the input excitation and 1+G(s)H(s) is the denominator of the closed loop transfer function. The type of system required may be determined from the waveform for which the steady state error is defined. Common waveforms are the unit step, unit ramp, and unit parabola. The minimum system types required to yield a finite steady state error are respectively, Type 0, Type 1, and Type 2 (see Experiment IV).

4.1.4. <u>Example of Classical Compensation</u>

Given the unity feedback control system in Figure 4.1.1

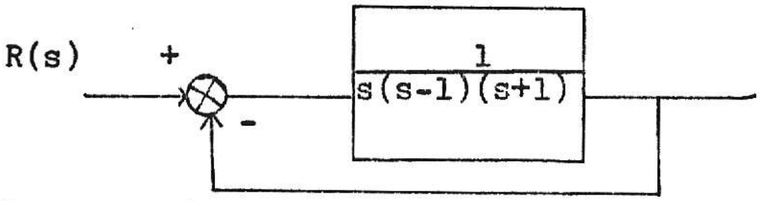


Figure 4.1.1 Unity Feedback Control System

compensate the system so that the maximum overshoot is 0.4 and the system is stable.

The initial root locus plot is shown in Figure 4.1.2.

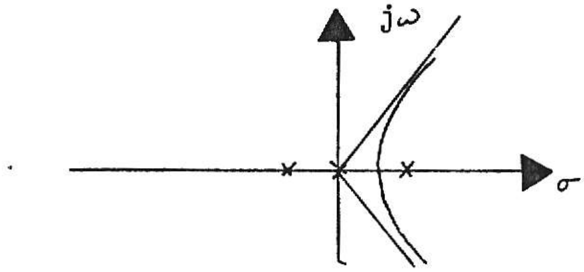


Figure 4.1.2 Root Locus Plot of Control System in Figure 4.1.1

The damping ratio is given by equation 4.1.11

$$\zeta = \frac{\log_e M_p}{\sqrt{\pi^2 + (\log_e M_p)^2}} = 0.280 \qquad (4.1.11)$$

The required phase margin is then

$$\phi_m = \tan^{-1} \frac{2\zeta}{\sqrt{\sqrt{1 + 4\zeta^4} - 2\zeta^2}} = 39.5^\circ \qquad (4.1.12)$$

Knowing the value of the damping ratio, ζ, also allows for the calculation of lines of constant damping which may be drawn on the root locus diagram. The angle, Ψ , that these lines of constant damping make with the imaginary axis is given by

$$\Psi = \sin^{-1} \zeta \tag{4.1.13}$$

Lines of constant damping lie strictly in the left half plane. Lines of constant damping have the property that if the poles of a second order system lie anywhere on these lines that the damping ratio, ζ , is equal to the sine of the constant damping angle, Ψ . For $\zeta = 0.28$ the constant damping angle is given by

$$\Psi = \sin^{-1} \zeta = 16.5^{\circ} \tag{4.1.14}$$

Cancelling the pole at s = -1 and replacing it with a pole at s = -10 gives the root locus plot of Figure 4.1.3

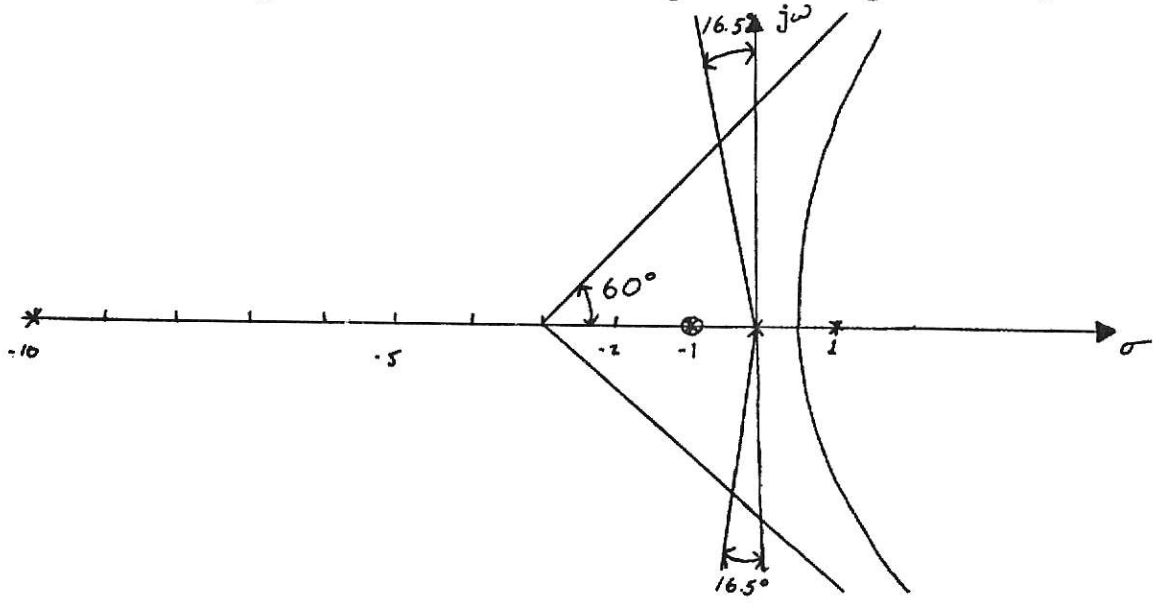


Figure 4.1.3 Root Locus Plot with the Pole at s = -1 Cancelled.

Addition of a zero at s = -2 and a pole at s = -20 modifies the root locus in the following manner. The addition of the zero at s = -2 will tend to draw the locus of the poles at s = 0 and s = 1 back into the left half plane yielding the root locus plot shown in Figure 4.1.4.

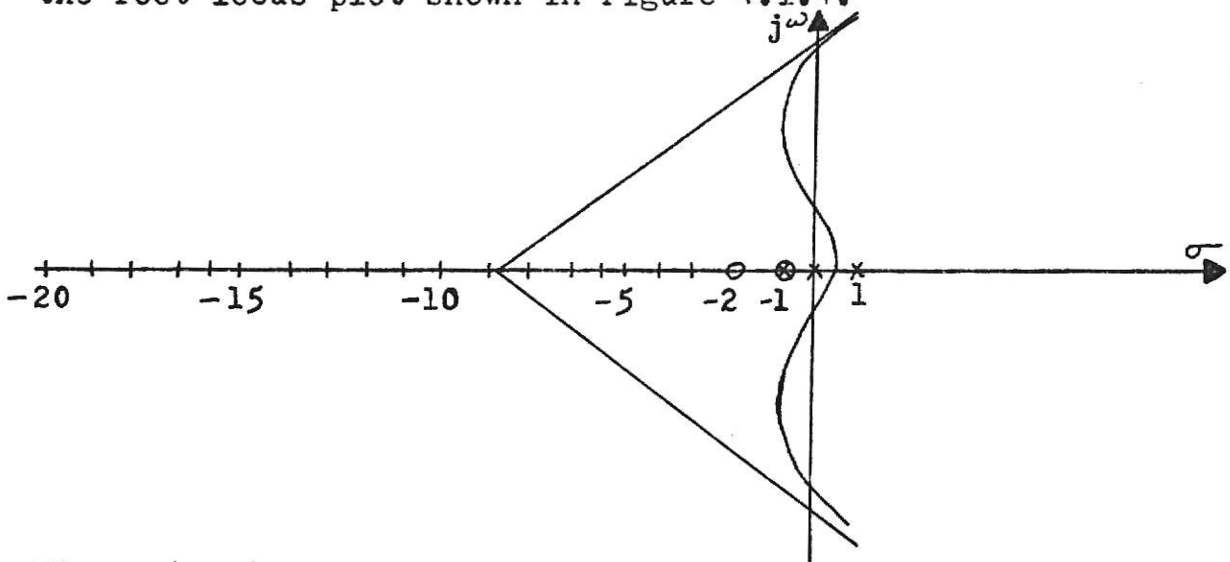


Figure 4.1.4 Root Locus Plot of Compensated System

The control system including both compensators has the block diagram indicated in Figure 4.1.5 and the closed loop equation given by 4.1.15.

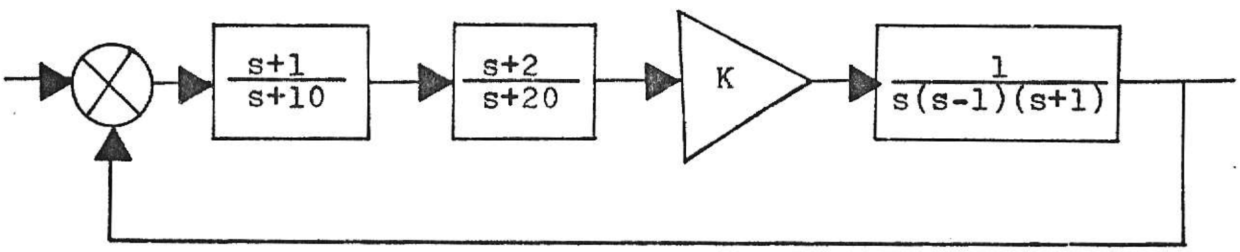


Figure 4.1.5 Block Diagram Including Compensators

$$\frac{C(s)}{R(s)} = \frac{K}{s^4+295^3+k79s^2+(K-200)s+2K} \qquad (4.1.15)$$

The Routh table of the characteristic equation of the closed loop transfer function given in equation 4.1.5 is shown in Figure 4.1.6.

s^4	1	170	2K
s^3	29	K-200	0
s^2	$\frac{5130-K}{29}$	2K	
s^1	$\frac{4748K-K^2-1026000}{5130-K}$	0	
s^0	2K		

Figure 4.1.6 Routh Table for the Characteristic Equation of Feedback System in Figure 4.1.5

Examination of the Routh table reveals that the minimum value of the loop gain, K, required to assure that the system is absolutely stable is 227 while the maximum value of the gain is 4521.

The gain must be chosen so that the dominate closed loop poles fall on the lines of constant damping. Generally, the gain is chosen to obtain the smallest value of gain that will yield the desired final result. In this case, the gain required may be determined from the root locus plot. The general shape of the root locus plot of the root locus is shown in Figure 4.1.7. A more exact plot of the locus for gains between 200 and 5000 is shown in Figure 4.1.7 from which the range of gains that may be used to cause the dominate poles to fall to the left of the line of constant damping may be determined. Since it is generally desirable to choose the minimum gain required to meet system specifications, a gain of 500 is chosen. A Nyquist plot of the system frequency response, shown in Figure 4.1.8, however, reveals, that since the phase margin of this system is only 1.8° which is outside specifications, that the location of the zero and pole is incorrect. A better choice of compensator for the system shown is one with the zero located at $s = -0.5$ and the pole located at $s = -30$. The open loop gain required to meet system specifications is then 1140. The root locus plot of the system with the new compensator is shown in Figure 4.1.9 while the Nyquist plot of the system is shown in Figure 4.1.10. The phase margin of the system with the new compensator is 40.1° which meets system specifications.

4.1.5. <u>Compensation Using State Variable Techniques</u>

Compensation, using state variable techniques, is more flexible than compensation using classical techniques because state variable techniques may be used on systems that are non-linear, time varying or that have more than one input or output. State variable techniques may also be

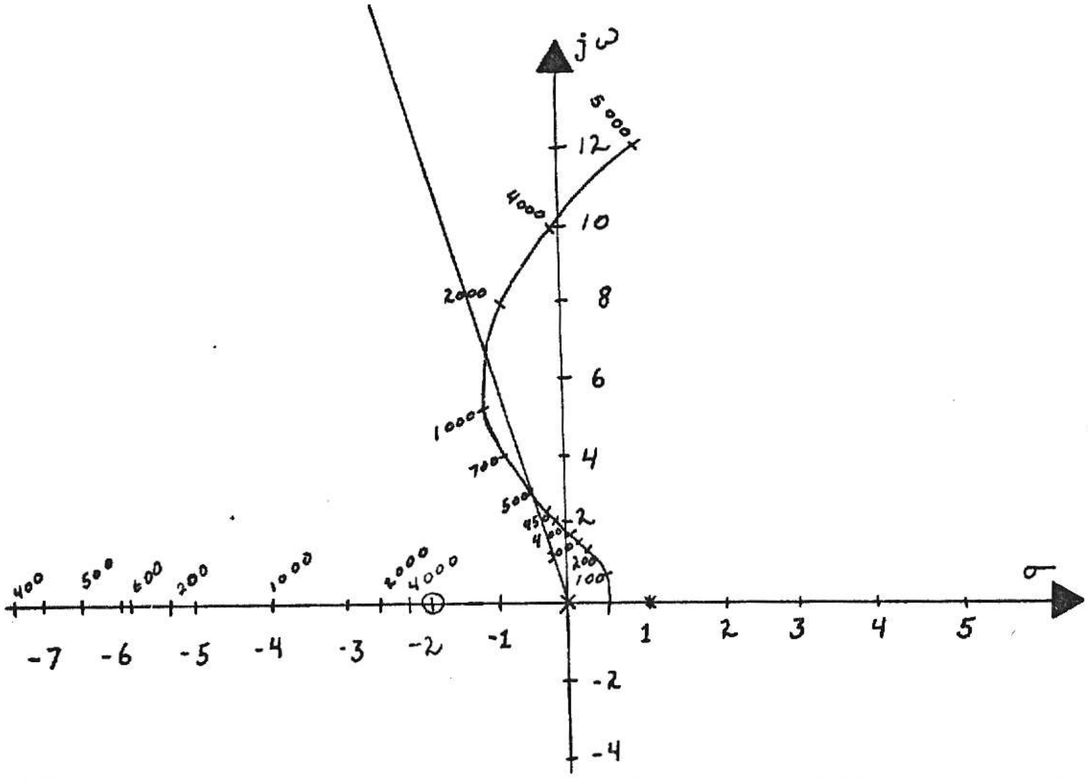


Figure 4.1.7 Root Locus Plot of System with Compensator Zero at S= -2 and Pole at S = -20

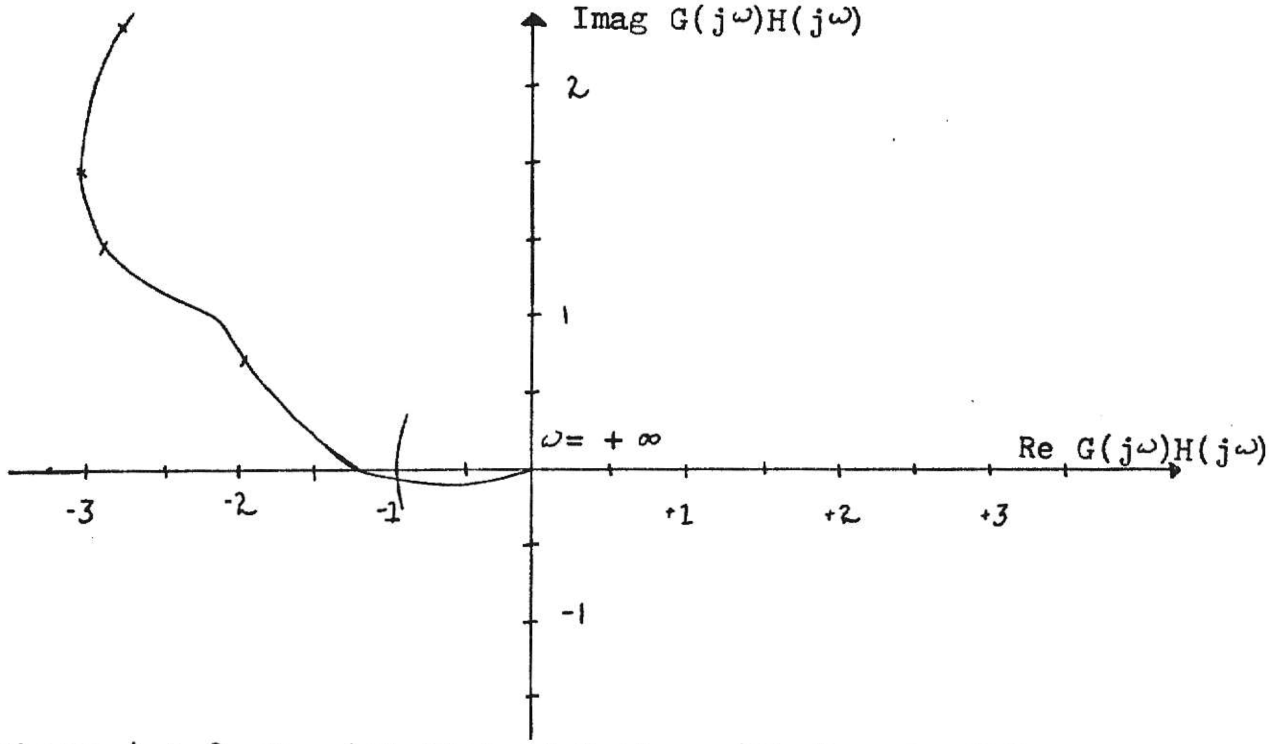


Figure 4.1.8 Nyquist Plot of System with Compensator Zero at S = -2 and Pole at S = -20

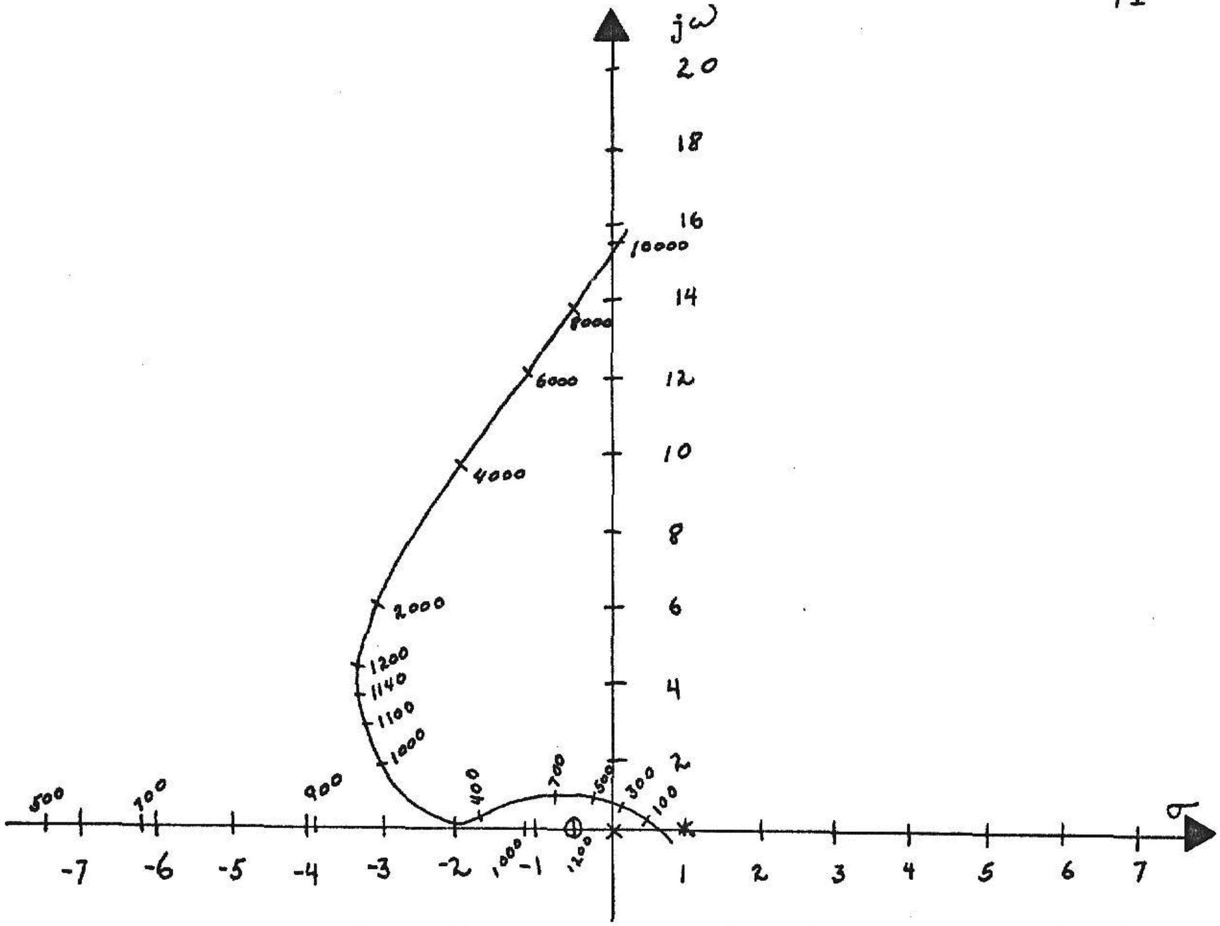


Figure 4.1.9 Root Locus Plot of the System with the New Compensator

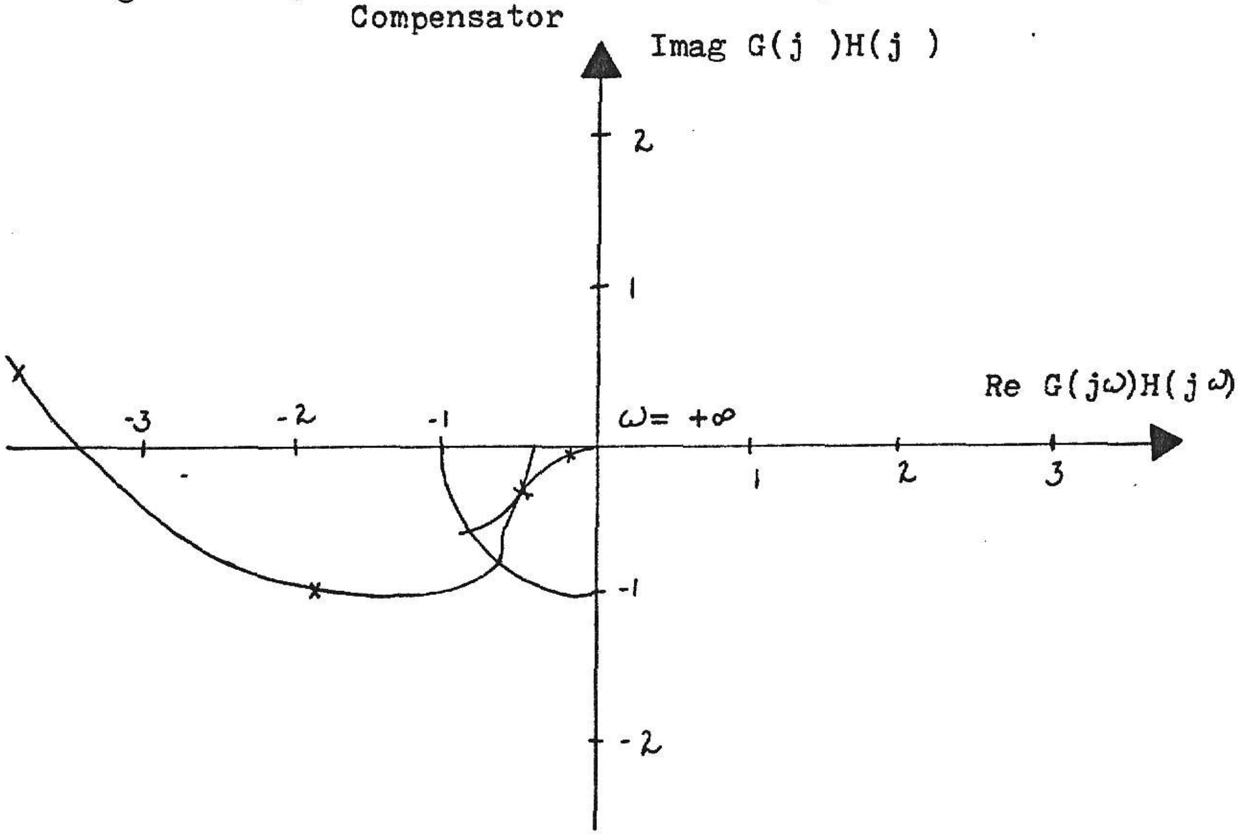


Figure 4.1.10 Nyquist Plot of the System with the New Compensator

used to obtain the compensation that will produce the best possible response subject to a given performance index. In general, determining the compensation that will produce the best system is very difficult, but in certain special classes of problems, the compensator required to yield the best or optimal system may be found rather easily.

The special case that will be considered here is optimization of a linear, finite order, time invarient system with a quadratic performance index. The equations describing this system are

$$\left.\begin{aligned} \dot{\bar{x}} &= \bar{\bar{A}}\bar{x} + \bar{\bar{B}}\bar{u} \\ \bar{y} &= \bar{\bar{C}}\bar{x} + \bar{\bar{D}}\bar{u} \end{aligned}\right\} \qquad (4.1.16)$$

where $\bar{x}$ is an nxl matrix, $\bar{y}$ is an mxl matrix, $\bar{u}$ is a pxl matrix, $\bar{\bar{A}}$ is an nxn matrix, $\bar{\bar{B}}$ is an nxp matrix, and $\bar{\bar{C}}$ is an mxn matrix and $\bar{\bar{D}}$ is an mxp matrix. The quadratic performance index, J, has the form

$$J = \int_0^\infty \sum_{i=1}^{n} \sum_{j=1}^{n} q_{ij} x_i x_j \, dt + \int_0^\infty \sum_{i=1}^{p} \sum_{j=1}^{p} r_{ij} u_i u_j \qquad (4.1.17)$$

where q_{ij} and r_{ij} are the costs associated with the state and control respectively. An alternate expression for J is

$$J = \int_0^{\infty} \left[\bar{x}' \, \bar{\bar{Q}}\bar{x} + \bar{u}' \bar{\bar{R}}\bar{u} \right] dt \tag{4.1.18}$$

where $\bar{x}'$ is the transpose of the matrix $\bar{x}$, $\bar{\bar{Q}}$ is an nxn matrix and $\bar{\bar{R}}$ is an rxr matrix. $\bar{\bar{Q}}$ and $\bar{\bar{R}}$ are defined to be

$$\bar{\bar{Q}} = \left[q_{ij} \right] \tag{4.1.19}$$

$$\bar{\bar{R}} = \left[r_{ij} \right] \tag{4.1.20}$$

With the system as described in Equation 4.1.16 and the matrices defined in Equations 4.1.17, 4.1.19, and 4.1.20, it can be shown that the control law with the value of J as defined in Equation 4.1.17 or 4.1.18 has the form

$$\bar{u}(t) = -\bar{\bar{K}}\bar{x}(t) \quad (4{:}778) \tag{4.1.21}$$

where $\bar{\bar{K}}$ is an rxn matrix of constant gains. If $\bar{\bar{Q}}$ is positive definite or positive semi-definite and $\bar{R}$ is positive definite then the system may be optimized using the procedure to be described.

Using Sylvester's criterion, a matrix is positive definite if the determinates of all principle minors are greater than zero. If the matrix is singular and the determinates of the principle minors are non-negative, than the matrix is positive semi-definite. The matrix is negative definite or negative semi-definite if the multiplication of all elements of the matrix by a negative one results in a positive definite or positive semi-definite matrix respectively. The following examples may aid in the understanding of positive definiteness and positive semi-definiteness. Example of a positive definite matrix is

$$\begin{bmatrix} 4 & -1 & -2 \\ -1 & 3 & -1 \\ -2 & -1 & 5 \end{bmatrix}$$

Determinate of first principle minor is

$$|4| = 4 \tag{4.1.22}$$

Determinate of second principle minor is

$$\begin{vmatrix} 4 & -1 \\ -1 & 3 \end{vmatrix} = 11 \tag{4.1.23}$$

Determinate of third principle minor is

$$\begin{vmatrix} 4 & -1 & -2 \\ -1 & 3 & -1 \\ -2 & -1 & 5 \end{vmatrix} = 35 \tag{4.1.24}$$

An example of a positive semi-definite matrix is

$$\begin{bmatrix} 4 & 0 & -2 \\ 0 & 0 & 0 \\ -2 & 0 & 5 \end{bmatrix}$$

Determinate of first principle minor

$$|4| = 4 \tag{4.1.25}$$

Determinate of second principle minor

$$\begin{vmatrix} 4 & 0 \\ 0 & 0 \end{vmatrix} = 0 \tag{4.1.26}$$

Determinate of third principle minor

$$\begin{vmatrix} 4 & 0 & -2 \\ 0 & 0 & 0 \\ -2 & 0 & 5 \end{vmatrix} = 0 \tag{4.1.27}$$

If $\overline{\overline{Q}}$ and $\overline{\overline{R}}$ are positive semi-definite or positive definite and positive definite respectively, then the substitution of the optimal control law yields the state equation,

$$\dot{\overline{x}} = (\overline{\overline{A}} - \overline{\overline{B}}\,\overline{\overline{K}})\,\overline{x} \tag{4.1.28}$$

(4:782). Defining

$$\overline{\overline{S}}{}'\overline{\overline{S}} = \overline{\overline{Q}} \tag{4.1.29}$$

(4:784), and using a result developed by Kalman shows that $(\overline{\overline{A}} - \overline{\overline{B}}\,\overline{\overline{K}})$ is stable if the rank of

$$\left[\overline{\overline{S}}{}' \;\middle|\; \overline{\overline{A}}{}'\overline{\overline{S}}{}' \;\middle|\; (\overline{\overline{A}}{}')^2\overline{\overline{S}}{}' \;\middle|\; \ldots \;\middle|\; (\overline{\overline{A}}{}')^{n-1}\overline{\overline{S}}{}'\right] \tag{4.1.30}$$

(4:784) is equal to n, the order of the state equation.

Solving the reduced Riccati matrix equation (4:783)

$$\bar{\bar{A}}'\bar{\bar{P}} + \bar{\bar{P}}\,\bar{\bar{A}} - \bar{\bar{P}}\,\bar{\bar{B}}\,\bar{\bar{R}}^{-1}\,\bar{\bar{B}}'\bar{\bar{P}} + \bar{\bar{Q}} = 0 \tag{4.1.31}$$

for the nxn matrix, $\bar{\bar{P}}$ allows the calculation of the gain matrix, $\bar{\bar{R}}$, of the optimal system using the equation

$$\bar{\bar{K}} = \bar{\bar{R}}^{-1}\bar{\bar{B}}'\bar{\bar{P}} \tag{4.1.32}$$

(4:783). The value of the cost function J, associated with this system is then

$$J = \bar{x}'(0)\ \bar{\bar{P}}\ \bar{x}(0) \tag{4.1.33}$$

(4:780).

If $(\bar{\bar{A}} - \bar{\bar{B}}\,\bar{\bar{K}})$ is stable as determined by equations 4.1.29 and 4.1.30 then the value of $\bar{\bar{K}}$ as determined in equation 4.1.32 will always yield the best system. If, however, $(\bar{\bar{A}} - \bar{\bar{B}}\,\bar{\bar{K}})$ is unstable, then the gain matrix, $\bar{\bar{K}}$, will yield the optimal system only if the matrix, $\bar{\bar{P}}$, calculated in equation 4.1.31 becomes positive semi-definite (4:784).

4.1.6. Example of State Variable Compensation

Given the control system

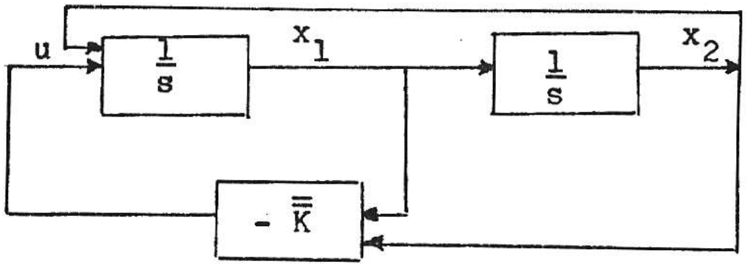


Figure 4.1.11 Block Diagram of State Variable System

and the cost functionals

$$J = \int_{0}^{\infty} \left[x_1^2 + 4x_2^2 + u \right] \; dt \tag{4.1.34}$$

determine the gain matrix, K.

The value of the matrix $\bar{\bar{Q}}$ and $\bar{\bar{R}}$ determined from the cost functional are

$$\bar{\bar{Q}} = \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix} \tag{4.1.35}$$

$$\bar{\bar{R}} = \begin{bmatrix} 1 \end{bmatrix} \tag{4.1.36}$$

The value of $\bar{\bar{S}}$ determined from Equation 4.1.29 is

$$\bar{\bar{S}} = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix} \tag{4.1.37}$$

Since $\bar{\bar{S}}$ is non-singular, $\bar{\bar{S}}'$ has rank 2 so that $(\bar{\bar{A}} - \bar{\bar{B}}\,\bar{\bar{K}})$ is stable. The state equations may be determined from the block diagram of the system and are

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} u \end{bmatrix} \tag{4.1.38}$$

Substituting the values of $\bar{\bar{A}}$ and $\bar{\bar{B}}$ from Equation 4.1.38 $\bar{\bar{R}}$ from Equation 4.1.36 and $\bar{\bar{Q}}$ from Equation 4.1.35 into Equation 4.1.31 gives

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} + \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$- \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} \begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} 1 \end{bmatrix} \begin{bmatrix} 0 \end{bmatrix} \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix} = 0$$

(4.1.39)

Multiplying the matrices in equation 4.1.39 gives

$$\begin{bmatrix} p_{21} & p_{22} \\ p_{11} & p_{12} \end{bmatrix} + \begin{bmatrix} p_{12} & p_{11} \\ p_{22} & p_{21} \end{bmatrix} - \begin{bmatrix} p_{11}^2 & p_{11}p_{12} \\ p_{11}p_{12} & p_{21}p_{12} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix} = 0$$

(4.1.40)

The four equations that must be solved to determine the elements of $\bar{\bar{P}}$ are

$$p_{21} + p_{12} - p_{11}^2 + 1 = 0$$

$$p_{22} + p_{11} - p_{11}p_{12} = 0$$

$$p_{11} + p_{22} - p_{11}p_{21} = 0$$

$$p_{12} + p_{21} - p_{21}p_{12} + 4 = 0 \qquad (4.1.41)$$

These equations when solved give

$$p_{11} = \sqrt{3 + 2\sqrt{5}} \approx 2.78$$

$$p_{12} = p_{21} = 1 + \sqrt{5} \approx 3.36$$

$$p_{22} = (1 + \sqrt{5})^2 - \sqrt{3 + 2\sqrt{5}} \approx 8.52 \qquad (4.1.42)$$

which yields

$$\bar{\bar{P}} = \begin{bmatrix} 2.78 & 3.36 \\ 3.36 & 8.52 \end{bmatrix} \qquad (4.1.43)$$

$\bar{\bar{P}}$ is positive definite since the determinates of both principle minors are positive.

$$|2.78| = 2.78 \qquad (4.1.44)$$

$$\begin{vmatrix} 2.78 & 3.36 \\ 3.36 & 8.52 \end{vmatrix} = 12.4 \qquad (4.1.45)$$

The value of the gain matrix, $\bar{\bar{K}}$, is

$$\bar{\bar{K}} = [1] \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 2.78 & 3.36 \\ 3.36 & 8.52 \end{bmatrix} = \begin{bmatrix} 2.78 & 3.36 \end{bmatrix} \qquad (4.1.46)$$

The cost associated with moving from the point, $x_1(0)=0$, $x_2(0) = 1$ is given by

$$J = \bar{x}'(0)\bar{\bar{P}}\,\bar{x}(0) = \begin{bmatrix} 0 & 1 \end{bmatrix} \begin{bmatrix} 2.78 & 3.36 \\ 3.36 & 8.52 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 8.52 \qquad (4.1.47)$$

4.1.7. Assignment

Obtain a block diagram of a control system from your lab instructor and design a compensator using classical techniques and state variable techniques.

4.2 EXPERIMENT VIII: COMPARISON OF COMPENSATED SYSTEMS

4.2.1. Purpose

The purpose of this experiment is to compare the performance of a control system compensated using classical techniques and state variable techniques.

4.2.2. Assignment

Program both systems compensated in Experiment VII and compare their responses to a unit step input. Change the position of one of the dominate poles in the system compensated using classical techniques by 10% and change one of the feedback gains by 10% for the system compensated using state variable techniques and again compare the responses of both systems to a unit step input.

Chapter 5

5. ADDITIONAL PROJECTS

5.1 PARTIAL DIFFERENTIAL EQUATIONS

5.1.1. Statement of the Problem

Many physical phenomena are not adequately described by ordinary differential equations. When a control system is desired that includes a device described by partial differential equations, the system designer has a choice of an ordinary differential that may not give the required accuracy but that is easy to use or a partial differential equation that has the accuracy but is much more difficult to work with from a mathematical point of view.

Techniques exist which allow a partial differential equation to be reduced to a system of coupled ordinary differential equations. The results obtained are only approximate; however, the order of the approximation may be increased until sufficient accuracy is obtained. The approximations obtained may be programmed on the analog computer.

5.1.2. References

EAI Handbook of Analog Computation, pages 279-287.

5.2 DESCRIBING FUNCTION ANALYSIS

5.2.1. Statement of the Problem

Analysis of a single stationary non-linearity in a control may not be carried out using the classical techniques of control system design. The describing function is a technique that may be used to linearize the amplitude-frequency response of the non-linearity so that an idea of its effect on the control system may be found.

The describing function is basically the magnitude and phase shift of the first term of the Fourier series of the non-linearity. For some kinds of non-linearities, the magnitude and phase angle associated with the describing function are also functions of the input amplitude as well as the frequency so that a large number of parametric plots are generated.

The effect of the describing function on the control system is generally analyzed using the Nyquist plot. On the Nyquist plot, the amplitude and phase shift of the output of the non-linear device are plotted parametrically as functions of the input amplitude and frequency. The linear portion of the control system is plotted separately and all intersections of the non-linear and linear portions of the plot are examined for the presence of limit cycles.

5.2.2. References

Modern Control Engineering, pages 531-561.

REFERENCES

1. Carlson, A., Hannauer, G., Carey, T., & Holsberg, P. J., Handbook of Analog Computation. New Jersey: Electronics Associates, Inc., 1965.

2. Director, S.W., & Rohrer, R.A. Introduction to System Theory. New York: McGraw-Hill, 1972.

3. DiStefano, J.J., III, Stubberud, A.R., & Williams, I.J. Feedback and Control Systems. New York: Prentice-Hall, 1967.

4. Ogato, K. Modern Control Engineering. New York: Prentice-Hall, 1967.

INTRODUCTION TO THE SIMULATION OF CONTROL SYSTEMS USING THE ANALOG COMPUTER

by

JOHN WAYNE HERMAN

B.S., Kansas State Teachers College, 1973

AN ABSTRACT OF A MASTER'S REPORT

submitted in partial fulfillment of the

requirements for the degree

MASTER OF SCIENCE

Department of Electrical Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

1975

ABSTRACT

INTRODUCTION TO THE SIMULATION OF CONTROL SYSTEMS USING THE ANALOG COMPUTER

A series of experiments have been written on the simulation of control systems using the analog computer. These experiments are concerned with an introduction to the use of the analog computer to check the performance of control systems with respect to design specifications in the time and frequency domain. The experiments are designed to follow or be concurrent with an introductory course in control systems. The student is assumed to have a good background in LaPlace Transform Theory, matrix algebra, and be able to use simple graphs associated with the design of control systems.