MISSISSIPPI STATE UNIVERSITY

State College, Mississippi

Interim Report

Contract No. NAS8-21377

Covering Period Period January 1, 1972 - April 30, 1972

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

$CR-123747$
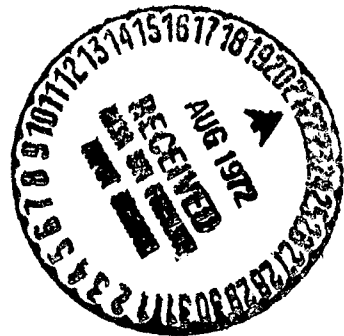
AN INNOVATIVE APPROACH TO COMPENSATOR DESIGN

By

Jerrel R. Mitchell

Submitted on behalf of

Willie L. McDaniel, Jr.

Professor of Electrical Engineering

ABSTRACT

The primary goal of this work is to present for a control system a computer-aided-compensator design technique from a frequency domain point of view. Up to the present there have only been two similar procedures developed, and they are somewhat limited. The thesis for developing the above said technique is to describe the open loop frequency response by n discrete frequency points which result in n functions of the compensator coefficients; several of these functions are chosen so that the system specifications are properly portrayed; then mathematical programming is used to improve all of these functions which have values below minimum standards.

In order to do this several definitions in regard to measuring the performance of a system in the frequency domain are given, e.g., relative stability, relative attenuation, proper phasing, etc. Next, theorems which govern the number of compensator coefficients necessary to make improvements in a certain number of functions are proved. After this a mathematical programming tool for aiding in the solution of the problem is developed. This tool is called the constraint improvement algorithm. Then for applying the constraint improvement algorithm generalized gradients for the constraints are derived.

Finally, the necessary theory is incorporated in a computer program called CIP (Compensator Improvement Program). The practical usefulness of CIP is demonstrated by two large system examples.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND SYMBOLS

Abbreviations

LP   Linear programming

NLP   Nonlinear programming

GRG   Generalized reduced gradient

SUMT   Sequentially unconstrained minimization technique

TIFR   Total improved frequency response

SIFR   Sum improved frequency response

CIP   Compensator improvement program

P or PM   Phase margin

G or GM   Gain margin

S   Stability margin

A   Attenuation margin

Symbols*

$G_k(s)$ Total compensation in the kth channel of the general unforced feedback control system (Figure 1)

α Point where loop of general feedback system is to be broken so that the open loop frequency response is obtainable

$C(s)/R(s)$ Open loop transfer function

$\theta_k(s)/R(s)$ Uncompensated open loop transfer function of kth channel when all channels are opened

---

*These symbols have these meanings unless specified otherwise.

LIST OF ABBREVIATIONS AND SYMBOLS (Continued)

Symbols

W	Coffey's diagonal weighting matrix
$(\ )^T$	Super T denotes transpose
K_i	Stear and Page's i^{th} weighting constant
$GH(j\omega)$	Open loop frequency response
∇g_m	Gradient of $g_m(x^T)$
D	Directional vector
x_K^T	Solution vector at K^{th} iteration
∇G	An n x m matrix whose columns are comprised of the gradients of the active constraints
c	In the improved constraint algorithm, the desired dot product vector
h	In the improved constraint algorithm, a scalar for determining the step size
-K	Point in the $GH(j\omega)$ plane from which a point on the $GH(j\omega)$ frequency response is to be perturbed with respect to this point
$d(\omega)$	Distance squared from some $GH(j\omega)$ frequency response point to some other point in the $GH(j\omega)$ plane
$G_{qp}(s)$	The pth sub-compensator in the qth channel of the generalized feedback control system
$\{\leq, =, \geq\}$	This means a quantity could be less than or equal, equal, or greater than or equal. The one used depends upon the particular problem.

CHAPTER I

INTRODUCTION

Before the invention of the digital computer, elaborate and complicated numerical techniques for solving problems in science, mathematics, and engineering were only given secondary consideration. As the refinement of the digital computer progressed, its comprehensive usefulness became more apparent. Today, the employment of the digital computer is found in almost every discipline of science and engineering.

Mathematical Programming

One area in which the digital computer has been of tremendous aid is in the solution of mathematical programming problems. The general mathematical programming problem may be stated as:

determine the n component vector $x^T = (x_1, x_2, \ldots, x_n)$ so that

the maximum (or minimum) of $\qquad f(x^T)$

subject to $\qquad g_i(x^T)\{\leq, =, \geq\} c_i$

$$i = 1, 2, \ldots, m \qquad (1\text{-}1)$$

is obtained. Each relation in (1-1) is assumed to be algebraic in nature. The relation, $f(x^T)$, is called the cost function, whose extremal with respect to the m constraints of the second relation is desired. If all the functions in (1-1) are linear and if the variables are not required to be integral valued, then the above optimization

problem is said to be a continuous linear programming problem (LP). The solution of the continuous linear programming problem may be accomplished with the aid of the simplex algorithm first introduced by George Dantzig.[1] Today the solution of continuous linear programming problems is treated extensively in many text books.[2,3,4,5]

On the other hand, if any of the algebraic functions in (1-1) are nonlinear, then the problem is called a nonlinear programming problem (NLP). Up to now there has been no one algorithm developed that will solve all nonlinear programming problems. Generally the existence and uniqueness of a solution cannot even be assured without the cost function and the constraints possessing certain convexity and concavity properties. By placing various restrictions on the functions in (1-1), there have been several algorithms developed for obtaining solutions.[6] In general, NLP algorithms are classed as either simplex in nature or as gradient in nature.[*]

Simplex Algorithms - Probably the first NLP algorithms developed were the separable programming algorithms. Problems for which they are applicable are of the following form:

$$\sum_{j=1}^{n} g_{ij}(x_j)\{\leq, =, \geq\}c_i$$

$$x_j \geq 0 \qquad j = 1, \ldots, n$$

$$\text{maximize (or minimize)} \quad z = \sum_{j=1}^{n} f_j(x_j) \ . \qquad (1-2)$$

---

[*]Here dynamic programming is not considered as a NLP algorithm but is considered as another branch of mathematical programming.

In order to apply separable programming both the constraints and the cost functions must be separable into functions of single variables. The mono-variable functions are then approximated over some finite interval by sequences of straight lines. Then a simplex algorithm is used to solve the approximate problem. The separable programming algorithms differ in the way the approximations are made and in the type of simplex algorithm necessary to solve the problem.[6]

Another simplex NLP algorithm is the quadratic programming of Wolfe.[11] It was especially developed to solve problems of the form:

$$Ax = b$$

$$x \geq 0$$

$$\text{maximize (minimize)} \quad z = cx + x^T Dx \qquad (1\text{-}3)$$

where A is an $m \times n$ matrix, c is an $n \times 1$ matrix, and D is an $n \times n$ negative semidefinite matrix. In this case the constraints are linear and the cost function is quadratic and concave. The development of the algorithm for solving (1-3) depends heavily upon the Kuhn-Tucker conditions.[2,7,11]

Still another simplex type algorithm is the Hocking-Hartley convex programming technique.[12] It is used to solve general NLP programming problems with certain convexity and concavity conditions. This method is derived by approximating the cost function and the constraints by an infinite number of supporting hyperplanes. Of course this produces a LP with an infinite number of rows. Then by using the duality principle of LP the problem is transformed into an infinite column problem which is amenable to solution by the simplex method. The convergence

properties of this algorithm are very reminiscent of the Newton-Raphson method for finding the roots of a polynomial, i. e., whenever the algorithm converges, it usually converges very rapidly.[12]

Of course, there are many other simplex type NLP algorithms; in fact, there are several versions of those given above. However, for brevity only the more publicized algorithms and the basic thoughts behind them have been mentioned here.

Gradient Algorithms -- In contrast to the simplex type algorithms there exist the gradient algorithms. The premier algorithms of this type are the gradient projection method[13,14], the generalized reduced gradient method (GRG)[18], and the sequentially unconstrained minimization technique (SUMT)[15,16,17]

The basic idea of the gradient projection method is to start with a feasible solution and move in the direction of the gradient of the cost function (for maximization problems) until the solution is found or until the violation of a constraint is attempted.[*] If the violation of a constraint is attempted, a direction is determined so that an increase in the cost function results and no violation of the constraints occurs. If no direction can be determined then the solution has been found.

In the case of linear constraints this simply requires projecting the gradient into the space defined by the intersection of all constraints which are equalities at the point under consideration. This is done by determining

$$r = Pd \qquad (1-4)$$

---

[*]A feasible solution is any point where no constraint is violated.

where r is the directional vector which points in the direction to move, d is the gradient of the cost function, and P is a projection matrix. The projection matrix is determined as

$$P = I - Q(Q^T Q)^{-1} Q^T \qquad (1-5)$$

where Q is a matrix whose columns are the gradients of the constraints which are strict equalities at the point of question. Of course if Q becomes square then P = 0. This does not indicate a solution but simply indicates that the feasible solution is located at a corner of the solution space. (For determining the projected gradient for this case, see Hadley [6], p. 167).

Another so-called gradient NLP method is the GRG method mentioned previously. This technique is a natural extension of the reduced gradient method of Wolfe to include nonlinear constraints. The reduced gradient method was developed to determine relative extremals of

$$\text{maximize} \quad f(x)$$

$$\text{subject to } Ax \leq b \qquad (1-6)$$

$$x_i \geq 0 \qquad i = 1,2,\ldots,n \quad .$$

It is assumed that any n-row submatrix of A has rank n. Next, A is partitioned into an n x n submatrix C and a submatrix D, and b is similarly partitioned into c and d. Then slack variables y and z are added so that the constraints in (1-6) become

$$Cx + y = c \qquad (1-7)$$

$$Dx + z = d \quad . \qquad (1-8)$$

All the constraints which are equalities are included in the C matrix. The variables of z are considered as dependent and those of y as independent. From (1-6), (1-7), and (1-8) it is easily seen that

$$\Delta x = -C^{-1}\Delta y \qquad (1-9)$$

$$\Delta z = DC^{-1}\Delta y \qquad (1-10)$$

$$\nabla_y f(x) = -\nabla f(x)C^{-1} \qquad (1-11)$$

where Δx and Δz represent the changes in the x's and y's. $\nabla_y f(x)$ is called the reduced gradient and $\nabla f(x)$ is the gradient of the cost function. From (1-9), (1-10), and (1-11) a set of rules has been devised for determining the correct changes in the x's and y's so that an increase in $f(x)$ is registered (For additional information see [30]).

Somewhat different from the gradient projection and GRG methods is the SUMT. The problems amenable to this technique are those which can be cast into the following form:

minimize $\quad f(x^T)$

subject to $\quad g_i(x^T) \geq 0$, $\quad i = 1, 2, \ldots, q$

$\qquad\qquad h_j(x^T) = 0$, $\quad j = 1, 2, \ldots, p$. $\qquad (1-12)$

In applying SUMT the above constrained minimization problem is transformed and solved as a sequence of unconstrained minimization problems which in the limit converges to a solution. This is done by forming from the above cost function and constraints a penalty function of the following form:

$$P(x^T,R) = f(x^T) + R \sum_{i=1}^{q} \frac{1}{g_i(x^T)} + \frac{1}{R} \sum_{j=1}^{p} h_j^2(x^T) \tag{1-13}$$

where R is a weighting constant greater than 0. For some initial value of R the unconstrained penalty function, (1-13), is minimized by some unconstrained minimization technique. Then R is decreased by dividing it by some number greater than 1 and the process is repeated. As R → 0 the unconstrained solution approaches a constrained solution. The physical effects of the two latter terms in (1-13) is to penalize a trial solution for getting too close to the boundary of the feasible region.

There has been no attempt here to be all inclusive with respect to gradient algorithms. There are several other gradient algorithms that have been developed. However, the ones mentioned above are considered by many as the most prominent and useful methods today.

Mathematical Programming in the Design of Control Systems

Over the past ten years there has been a great thrust to use mathematical programming in the design of control systems. The major effort has been in the solution of optimal control problems, and the results in this area have been very fruitful—not only in the application of mathematical programming but also in theoretical developments. In fact, it has been shown that the Kuhn-Tucker necessary conditions of mathematical programming and the maximum principle of optimal control can be derived from the same set of general optimization theorems[19,20,21,22]. As can be seen from the lengthy reference

list by Tabak[23], much of the work has been directed toward the applications of linear and quadratic programming. Recently, uses of the SUMT and the GRG algorithms in the solution of optimal control problems have been made.[24,25,26]

On the other hand, the use of mathematical programming in the classical design of control systems has been meager--particularly in the design of compensators from a frequency domain point of view. This is very unfortunate because most practical system designs even today are still by classical frequency domain approaches. Furthermore, these approaches are more artful than analytical. The few techniques which have been developed can be classified as modern control oriented or strictly classical control oriented. This classification results from the choices of the performance indices. Those methods in which system specifications are submerged in a cost functional are labeled as modern control approaches, while those methods which represent the system performance by classical standards such as gain margins, phase margins, bandwidth, etc., are termed as classical approaches.

One of the first successful computerized compensator algorithms was developed by Coffey.[27] In his paper consideration is given to a system similar to that shown in Figure 1. In this figure j parameters of the system are sensed; each parameter is operated on by some compensation device; the results of these are summed and fed back. Figure 1 is considered typical of large aircraft or space vehicles. Each compensator is assumed in the following form:
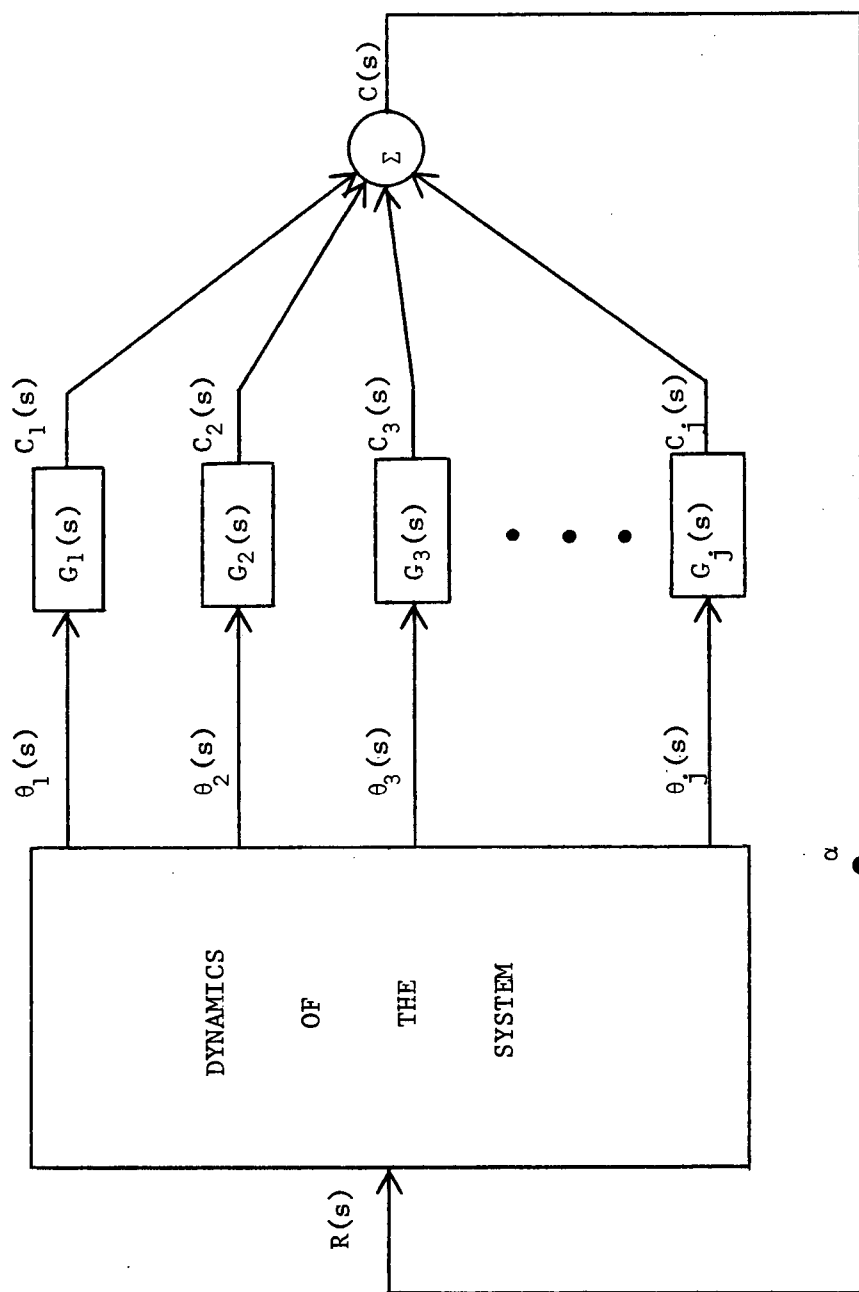
Figure 1. A Typical Unforced Multi-channel Control System

$$G_e(s) = [\sum_{i=1}^{M_e} a_{ei}s^{i-1}]/[1 + \sum_{i=2}^{N_e} b_{ei}s^{i-1}] \qquad (1\text{-}14)$$

where s is a complex variable and $M_e - 1$ and $N_e - 1$ are the numerator and the denominator orders, respectively, of the eth compensator. The goal is to select the compensator coefficients so that the compensated open loop frequency response is a weighted least squares fit to a desired open loop frequency response (Of course, the open loop frequency response is obtained by calculating $C(j\omega)/R(j\omega)$ when the feedback loop is broken at α).

The weighted least-squares fit is obtained by minimizing the following cost function:

$$J = |(y^* - \hat{y}^*)^T WW^T (y - \hat{y})| \qquad (1\text{-}15)$$

where $\hat{y}$ is a vector of the desired frequency response points, y is a vector of frequency response points, W is a diagonal weighting matrix, the asterisk (*) denotes conjugate, and the super T denotes transpose. For minimizing the cost function, J, with respect to the compensator coefficients a gradient search algorithm is chosen; and, then, the necessary gradient vector is calculated.

Next, geometrical properties of the cost function are considered. It is demonstrated that even for relatively simple systems the cost function is geometrically complicated. From this it is seen that the cost functions can have relative extremals and unbounded solutions. Furthermore, the design of unstable compensators is possible. Even with the possibility of these difficulties, it is demonstrated that

this procedure can be utilized to design practical compensators. This is done by applying the technique to a sixth order ballistic missile example. For this system two compensators are designed--a pure gain and a fourth order over a sixth order. The pure gain compensator approximated the desired frequency response for low frequencies but was completely unsatisfactory for higher frequencies. In fact, for this compensator the closed loop system is unstable. On the other hand the higher order compensator exhibited very good properties when compared to the desired frequency response.

Coffey indicates that in some instances a judicious choice of the elements of the weighting matrix, W, is required before an acceptable design can be achieved. Thus, a computer program of this algorithm might require several runs--while juggling these elements between runs--before the proper values are conceived. Even with this disadvantage the algorithm is definitely superior to classical means.

Another technique for computerized design of compensators for control systems has been presented by Page and Stear.[28,29] The thesis of this procedure is to vary the compensator coefficients until certain chosen frequency response specifications are satisfied. The procedure for attempting to do this is

$$\text{minimize} \quad F = \sum_{i=1}^{N} K_i (1 - S_i^a / S_i^d)^2 \tag{1-16}$$

where N is the number of specifications considered, S_i^a is the specification as a function of the compensator coefficients, S_i^d is the desired specification, and K_i is a weighting constant. The constant K_i is chosen as positive, in general one, for $S_i^a \leq S_i^d$ and as zero

for $S_i^a > S_i^d$. This results in a satisfied specification being neglected. The goal is to drive F to zero. The reason for the choice of the above criterion function (1-16) is to try to place the most emphasis on the specifications which have the greatest violations.

In order to illustrate the given procedure Stear and Page present an example of the design of an autopilot for an aircraft. In accomplishing this design four unconstrained optimization procedures are used. Three are local search procedures, and one is a global search technique. As in the case of Coffey's cost function it is discovered that even for simple compensators the specification function (1-16) has relative extremals. From this it is deduced that the global search procedure is more applicable than the local search techniques if the starting compensator is strictly arbitrary. However, if a priori knowledge is used in picking the initial compensator this deduction is not necessarily true.

Pitfalls of Previous Works on Computerized Compensator Design Procedures

The two previously mentioned works on computerized compensator design procedures suffer from several drawbacks. First the procedure presented by Coffey is basically a frequency response shaping technique. In the design of compensators for most control systems, this is too rigorous; i. e., this requires the compensator to satisfy more constraints than are necessary. Thus, the probability of all system specifications being satisfied is less. Another interesting fact is that in many instances the frequency responses of control systems

are not required to match a desired frequency response--frequency to
frequency--but are desired to have some general shape which can be
translated with respect to frequency. Even more conceivable is the
desirability to have several bands of the frequency response to be
various distances from the $-1 + j0$ point of the $GH(j\omega)$-plane and to
have other bands of the frequency response constrained to be greater
than or less than limitations with respect to the origin of the
$GH(j\omega)$-plane. Constraints such as these are not as strenuous as those
requiring the frequency response to fit closely to some desired fre-
quency response.

A pitfall which is common to both the Coffey method and the
Stear and Page method is the necessity of choosing some constants--in
particular, the elements of the diagonal matrix, W, and the K_i's. It
is obvious that in many situations a judicious choice of these must
be made before any useful results will emerge. It was suggested by
these authors that computer programs containing the algorithms may
require several runs with various values of these constants before an
acceptable design is achieved. However, this involves trial and
error which was one of the justifications for going to a computerized
procedure.

Another drawback of the two algorithms presented is that some
specifications may become worse while others become better. This
immediately poses some serious questions, such as, what is a reason-
able trade-off and where does it exist? If minimum standards of
system performance have been set, it is very probable that nothing
short of these are acceptable. In this case there is no trade-off.

On the other hand, it may be viewed that in practical designs it is not unusual to accept performances a little less than that desired. In instances such as this, performance tolerances must be set.

Another shortcoming of the two methods is their failure to include inherent devices for maintaining compensator stability. If the designed compensator is unstable, then the stability criterion of the system changes completely. The result might be system instability which removes the compensator from the realm of a practical design. What is needed is an algorithm which tends to improve system specifications at every iteration. Of course this might require the allowance of only incremental changes in the compensator coefficients.

Another pitfall of the two previously mentioned works is the lack of any theoretical inclusions on compensator limitations. That is, none of the authors presented any theoretical developments showing what could be expected from their algorithms for a certain compensator order in a particular system. Thus, initially there is no way to know what minimum amount of compensation is necessary. In addition, these works presented no theory which indicates that the algorithms will produce a final compensator that is any better than the initial compensator.

In essence, the techniques of Coffey and Stear and Page are "firsts" in the use of the computer for compensator designs, but they are somewhat limited. They do not present universal solutions in regard to computerized-compensation. It is the purpose of this dissertation to present the theory and a method of computer-aided compensator design that does not have the drawbacks of the previously presented techniques and is thus more universal.

CHAPTER II

FREQUENCY RESPONSE CONSIDERATIONS IN THE

DESIGN OF A CONTROL SYSTEM

Before the design of a system can be accomplished, the limitations
or constraints and the desired performance of the system must be
established. The measurement of the performance of the system is
determined by comparing that obtained to that desired. Because of the
limitations, in many instances, the desired performance cannot be
achieved. In designing compensators for practical control systems
there are, in general, two types of performance indices--time domain
indices and frequency domain indices. Although it is quite obvious
that these are related, no analytical means, up to now, have been
devised for defining this relation except for the simplest control
systems--less than third order. In practical designs the main limi-
tations are system stability, nonlinearity, time variance, and
sensitivity. Today many systems are designed by using linearized
frozen time models and applying frequency domain concepts.

Concept of Relative Stability

In most practical systems stability is a major constraint. In
fact, in most system designs a specified degree of stability is
required. A specific degree of relative stability is required because
of inaccuracies in the model of the system or in order to deter insta-
bility if future parameter variations in the system plant result.

Sometimes a certain amount of relative stability is desired to keep the system from resonating unnecessarily.

In the past the degree of relative stability has been denoted by the classical gain (GM) and phase margins (PM). However, in some instances these can be very misleading. For example, consider the hypothetical s-plane frequency response shown in Figure 2 which possesses acceptable classical stability margins (GM $\geq$ 2.0; PM $\geq$ 30°) but which comes within some small distance of the -1 + j0 point. Such a condition could represent a system which was very close to instability. A better measurement of relative stability is defined as follows:

> A stability margin is defined as the magnitude of the 1 + GH(jω) frequency response at one of its minima relative to the origin of the 1 + GH(jω) plane.

It is deemed by this author that by measuring stability in this fashion, a measure of the true relative stability of a system is achieved. Next, a system is said to be relative stable if the frequency response does not cross a designated closed contour located around the -1 + j0 point. This closed contour around the -1 + j0 point is called the *margin of stability limit*.[7] The shape and the size of this contour depend upon system specifications. Furthermore, there is nothing wrong with making the size and shape of the contour frequency dependent. (In doing this the designer would be indicating that the frequency response is to be shaped to some extent.)

Relative Attenuation Concept

Although relative stability plans a major role in compensator determination, there are several other factors which are considered.

RD

GH(jω)-PLANE
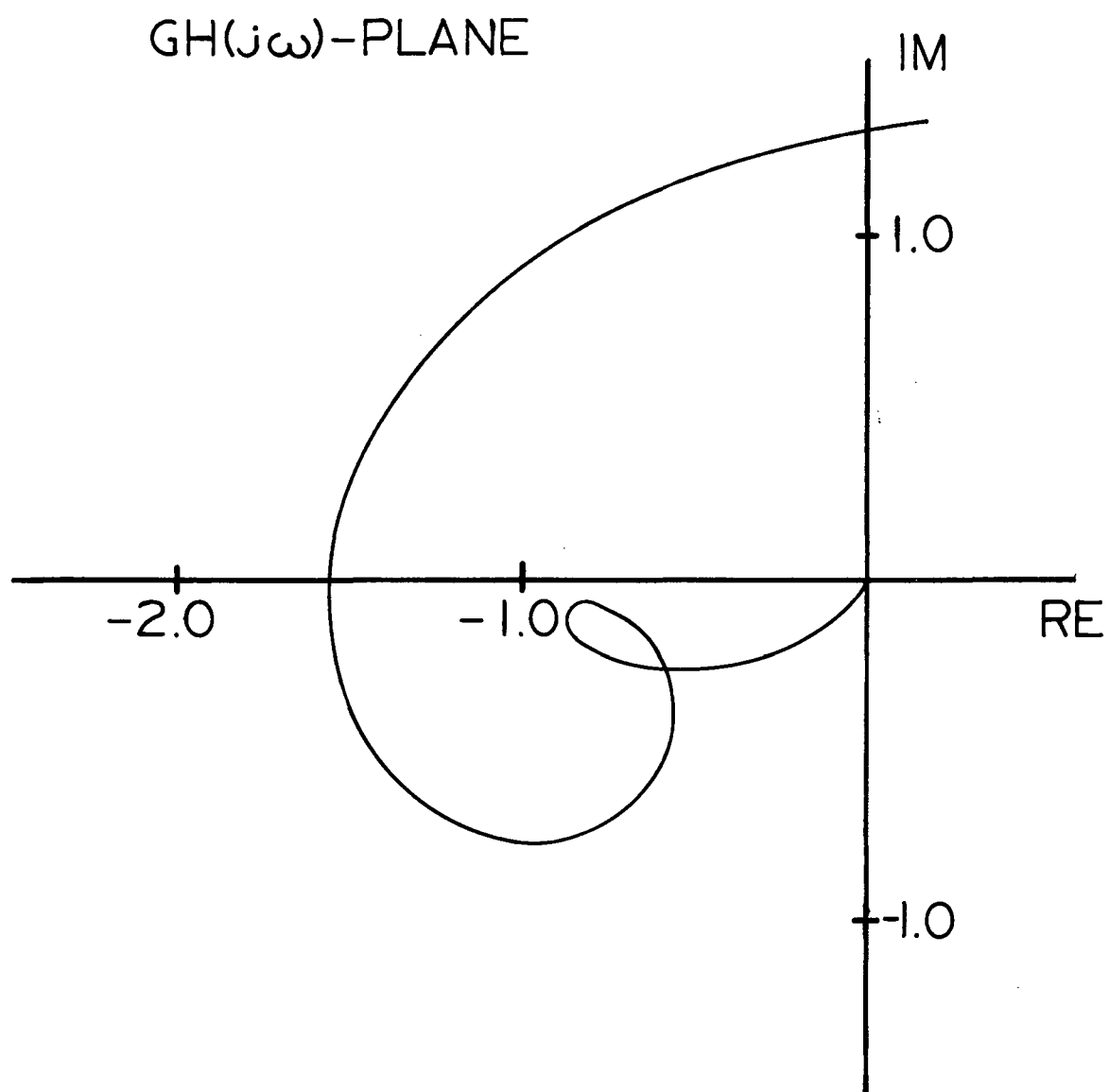
Figure 2. A Hypothetical GH(jω) Frequency Response

One of these is the attenuation of certain frequency bands. The reason for frequency band attenuation is to discourage the control system from resonating at some natural frequency of the system. Of course if the system is linear and time-invariant this is not necessary. Unfortunately, many practical systems do not fit into the linear, time-invariant category.

Frequency band attenuation may be treated by requiring that all frequency points that are to be attenuated fall within a chosen contour around the origin in the GH(jω) plane. This contour is called the *margin of attenuation limit*. It then follows that:

> An attenuation margin is the magnitude of the GH(jω) frequency response at one of its maxima with respect to the origin of the GH(jω) plane.[7]

Other Frequency Response Concepts

Relative stability and attenuation are considered as the most important frequency response design criteria. However, they do not yield acceptable designs in all instances. Sometimes it is necessary to employ proper phasing of certain frequencies. This is usually employed when it becomes difficult to determine a compensator to attenuate certain natural frequencies of the system and in addition to satisfy other system requirements. The general idea is to determine a compensator so that these frequencies are phased toward the right half of the GH(jω) plane. This results in these frequencies being attenuated in the closed loop system.

In some cases it is even necessary to place special emphasis on certain points of the frequency response. In most instances these

points are closely related to dynamical responses of the control system. Examples of dynamical responses considered for a space craft are wind response and "engine-out" response. In order that these responses possess acceptable characteristics it is usually necessary to require certain frequency response points to be placed in certain regions of the $GH(j\omega)$ plane.

Still another frequency response design concept is bandwidth. However, this can be handled by either the stability margin or the attenuation margins. For example, the maximum open loop bandwidth can be achieved by requiring a certain frequency and all frequencies above it to have a certain margin of attenuation limit. Similarly, closed loop bandwidth could be controlled by a combination of these.

Problem Formulation

Assuming that the desired frequency response characteristics have been determined so that if they are achieved the performance of the system will be acceptable, it must be decided how to determine a compensator for achieving these. The classical means of doing this is by trial and error; however, a more efficient method would be an iterative method that makes improvements upon the system's frequency response from iteration to iteration or indicates that no further improvement could be made. In fact, if a total of n critical frequency points have been chosen, then the problem may be formulated as the following nonlinear programming problem:

Determine a vector x^T such that

$$g_i(x^T, \omega_i) \leq b_i$$

$$x_i \{\leq, =, \geq\} d_i$$

$$i = 1, \ldots, n \qquad\qquad (2\text{-}1)$$

In (2-1) x^T is a vector of the compensator coefficients; g_i is a function of the ith frequency, ω_i, and the compensator coefficients. The functions, g_i, $i = 1, \ldots, n$, are chosen so as to represent the frequency response limitations and constraints which have been imposed. For example, g_i could be representative of a stability margin or an attenuation margin. The second relation in (2-1) takes into account any constraints that might be placed on the compensator coefficients. It may be necessary to constrain some of the coefficients if it is desired to keep the d. c. gain, $G(j0)$, of the system constant or above or below a certain level. Also, it may be necessary to constrain certain compensator coefficients to insure the stability of the compensator or to take into account realizability conditions.

The above formulated nonlinear programming problem differs from the classical nonlinear programming problem in the respect that it is strictly a constraint problem.[6] There is no cost function to maximize or minimize. However, this does not simplify matters. In fact, the above problem can be thought of as a normal nonlinear programming problem in which it is desired to find a solution which obtains a certain objective function value. In this case the objective function just becomes a constraint. If the objective function is added to the constraint list, then the result is a strict constraint problem as given above. The desired solution to this problem is a feasible solution which may or may not exist.

CHAPTER III

COMPENSATOR LIMITATIONS

At any iteration in solving the problem mentioned in Chapter II, there will result conditions of the form of (2-1) to be improved. (The number n can change from one iteration to another since the frequency response changes with respect to the compensator.) The general idea is to change the compensator coefficients so that each constraint comes closer to being satisfied. The question then is, how many compensator coefficients are required to insure that some improvement on each constraint at a certain iteration can be made? This question is answered by the following definitions and theorems.

Definition 1

An optimal direction in the GH(jω) plane is any chosen direction in which it is desired to perturb a point on the frequency response.

Optimal directions are illustrated in Figure 3 at points A, B, and C. The number of compensator coefficients sufficient to perturb n polar frequency response points in n optimal directions is given by the following theorem:

Theorem 1

A sufficient condition to perturb n points on a polar frequency response curve in n optimal directions with a realizable compensator is that there be at least 2n independent compensator coefficients which are available to be varied.
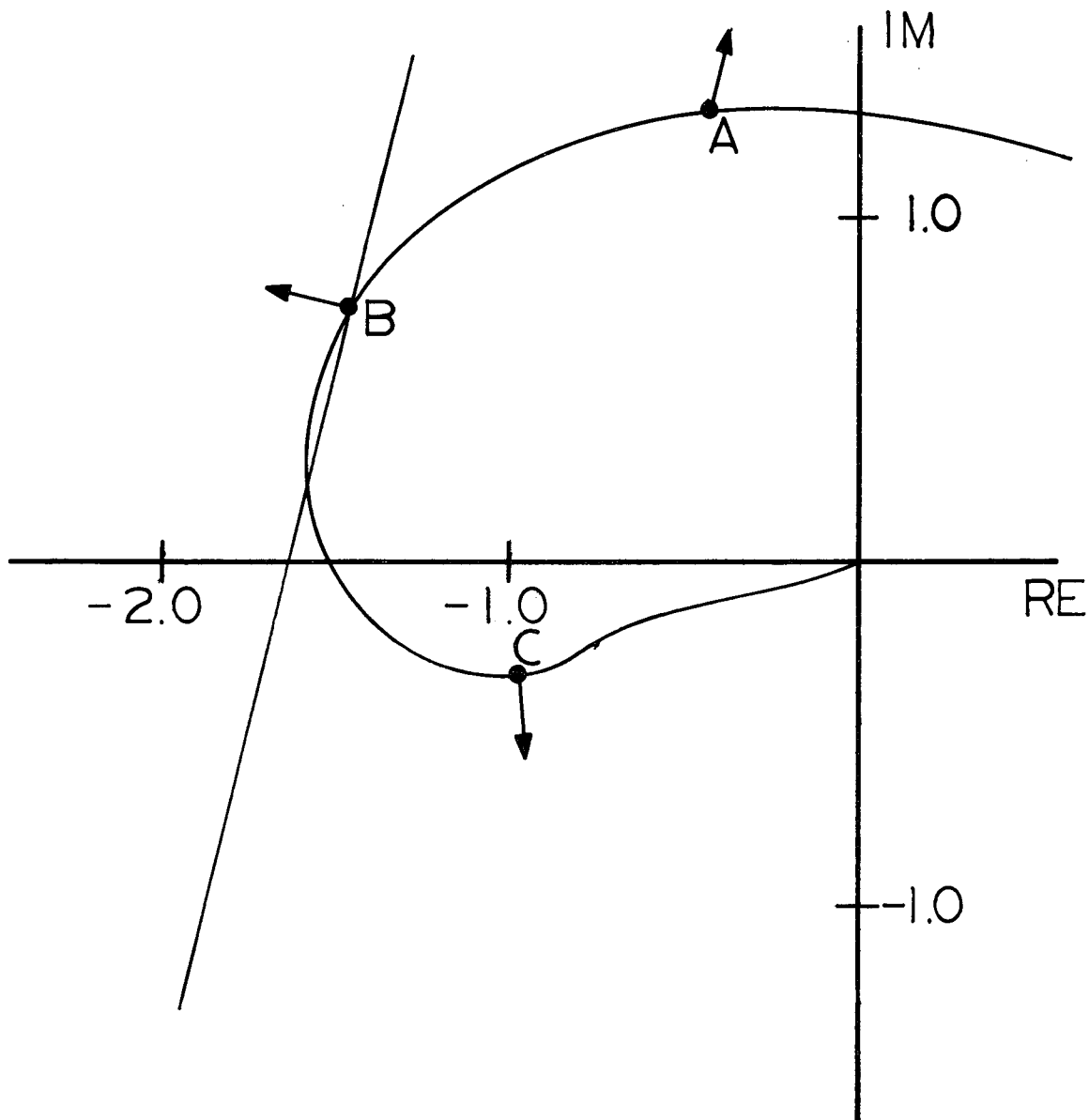
Figure 3. A GH(jω) Frequency Response for Illustrating
Optimal and Sub-optimal Directions

Proof: Let the open loop frequency response be denoted by $G_o(j\omega, x^T)$ where x^T is an m dimensional vector of the functionally independent compensator coefficients. Also, let the optimal direction at a frequency ω_k be denoted by d_k^*. Suppose there are n points on the frequency response which are to be moved in the n chosen directions, respectively. The change of the open loop transfer function at the kth frequency with respect to the ith compensator coefficient is of the form

$$\frac{\partial G_o(j\omega_k, x^T)}{\partial x_i} = c_{ki} + je_{ki} \qquad (3-1)$$

where c_{ki} and e_{ki} are real constants. There are, for a particular frequency, m such partials as (3-1) and, if they were included as the components of a single vector, the result would be the complex gradient. It is well known that this points in the direction of the most rapid change. However, this is not the desired direction of movement. Essentially what is needed is a directional vector [w] in complex m-space whose dot product with the m dimensional gradient vector $[c_k + je_k]$ will yield the desired directional derivative d_k^*, or in equation form (See [32])

$$d_k^* = [c_k + je_k]^T[w] \quad . \qquad (3-2)$$

It should be obvious that the components of [w] are proportional to the amount that each compensator coefficient must be varied in order that movement in the d_k^* direction can be accomplished. Thus if the compensator is to be realizable, [w] must be a real vector.

Letting

$$d_k^* = a_k^* + jb_k^* \ , \tag{3-3}$$

then (3-2) can be written by the following two real equations:

$$a_k^* = [c_k]^T [w] \tag{3-4a}$$

and

$$b_k^* = [e_k]^T [w] \ . \tag{3-4b}$$

Hence, for n points on the frequency response to be moved in n optimal directions there result 2n equations or

$$a_1^* = [c_1]^T [w]$$

$$b_2^* = [c_2]^T [w]$$

$$\vdots$$

$$a_n^* = [c_n]^T [w]$$

$$b_1^* = [e_1]^T [w]$$

$$b_2^* = [e_2]^T [w]$$

$$\vdots$$

$$b_n^* = [e_n]^T [w] \ . \tag{3-5}$$

In matrix notation (3-5) becomes

$$\begin{bmatrix} a^* \\ \cdots \\ b^* \end{bmatrix} = \begin{bmatrix} c^T \\ \cdots \\ e^T \end{bmatrix} [w] \tag{3-6}$$

where the dimensions of $\begin{bmatrix} a^* \\ \cdots \\ b^* \end{bmatrix}$, $\begin{bmatrix} c^T \\ \cdots \\ e^T \end{bmatrix}$, and [w] are

respectively 2n x 1, 2n x m, and m x 1. If 2n > m there will result

more equations than unknowns and possibly an incompatibility.[31] Hence there may not exist a vector [w] such that all equations can be satisfied. This says there are not enough compensator coefficients available. On the other hand if $2n \leq m$, there either results less equations than unknowns or the same equations as unknowns. For the first case there will exist an infinite number of vector [w]'s and an infinite number of solutions to the equations. This indicates an excessive number of compensator coefficients. In the second case there will be a unique [w] and, thereby, a unique solution for the equations. This means that the exact number of compensator coefficients necessary is being employed.[8]

The preceding proof has shown the sufficiency condition for moving the frequency response in n optimal directions. Suppose, however, that it is desirable to use a compensator with a fewer number of coefficients than those needed to move in the optimal directions. Consider the following definition:

<u>Definition 2</u>

A sub-optimal direction is any direction within $\pi/2$ radians of an optimal direction.

An optimal direction is just a two-space vector; then, a sub-optimal direction is any two-space vector which has a positive dot product with an optimal direction. Thus, a sub-optimal direction is any vector which falls within a certain open half space, e.g., a sub-optimal direction to B in Figure 3 is any vector which points to the left of the line passing through B.

If the optimal and sub-optimal directions for ω_k are respectively represented in 2-space by the following vectors :

$$\vec{d}_k{}^* = (a_k{}^*, b_k{}^*) \quad , \tag{3-7}$$

and

$$\vec{d}_k = (a_k, b_k) \quad , \tag{3-8}$$

then the sub-optimal direction would be any direction such that the dot product

$$\vec{d}_k \cdot \vec{d}_k{}^* > 0 \tag{3-9}$$

or

$$a_k a_k{}^* + b_k b_k{}^* > 0 \quad . \tag{3-10}$$

Then the question is, how many compensator coefficients are necessary in order to assure that movement in some sub-optimal direction can be achieved? The answer to this is stated and proved in the supervening theorem.

Theorem 2

In order to be assured of perturbing n points of an open loop frequency response in n sub-optimal directions, by varying the compensator coefficients, it is necessary that there be n independent compensator coefficients available for variance.

Proof: The components of the kth sub-optimal vector direction in terms of the real and imaginary parts of the partials at the kth frequency are given by

$$a_k = \sum_{i=1}^{m} c_{ki} w_i \tag{3-11}$$

$$b_k = \sum_{i=1}^{m} e_{ki} w_i \qquad (3\text{-}12)$$

where c_{ki} and e_{ki}, respectively, are the real and imaginary parts (evaluated at ω_k) of the partial of the open loop transfer function with respect to the ith compensator coefficient, and w_i is the ith unknown constant which is to be determined so that (a_k, b_k) points in a sub-optimal direction. Substituting (3-11) and (3-12) into (3-10) results in

$$\sum_{i=1}^{m} c_{ki} w_i a_k^* + \sum_{i=1}^{m} e_{ki} w_i b_k^* > 0 \qquad (3\text{-}13)$$

or

$$\sum_{i=1}^{m} (c_{ki} a_k^* + e_{ki} b_k^*) w_i > 0 \quad . \qquad (3\text{-}14)$$

Remembering that there are n frequency points, n inequalities like (3-14) will result. Hence the following matrix inequality can be obtained:

$$[c^T a^* + e^T b^*] \, [w] > 0 \quad . \qquad (3\text{-}15)$$

The dimension of $[c^T a^* + e^T b^*]$ is n x m. In order to be assured that all n inequalities can be satisfied, it is necessary that there be at least the same number of unknowns as inequalities. Hence, this says there must be at least n independent compensator coefficients in order to be assured that n frequency points can be perturbed in the sub-optimal directions.[8]

The above two theorems place limitations on the overall compensator order. Thus for any algorithm to be assured of being able to

make the changes given in the theorems, the theorem must be sat-isfied.

CONSTRAINT IMPROVEMENT ALGORITHM

It is very desirable to have an algorithm which starts with some initial compensator and, then, in an iterative fashion produces an improved frequency response. This statement immediately suggests the question--what is an improved frequency response? This is answered by the following two definitions.

Definition 3

A total improved frequency response (TIFR) in an iterative scheme is one whose unsatisfied constraint values at a certain iteration are better than they were at the last iteration.

Definition 4

A sum improved frequency response (SIFR) in an iterative scheme is one whose sum of the differences in the unsatisfied constraint values and their desired values is a positive value from one iteration to the next.*

It is obvious that an algorithm which is capable of producing a TIFR is also capable of producing a SIFR; however, this statement is not reversible. A TIFR algorithm requires every constraint which is unsatisfied to be improved or bettered at every iteration, while a SIFR algorithm only necessitates a sum improvement, i. e., the sum

---

*It is assumed, here, that all constraints in (2-1) have been represented in the $\leq$ form by multiplying $\geq$ constraints by -1 and changing = constraints to two inequality constraints (See Hadley [6]). No generality is lost by doing this.

increase must be better than the sum decrease. The goal is then to derive an algorithm which is compatible to both TIFR and SIFR.

Thus, an algorithm is needed for solving a nonlinear programming problem of the following form:

Determine the vector x^T such that

$$g_i(x^T) \geq b_i \qquad i = 1, \ldots, m \, . \qquad (4-1)$$

Again this is strictly a constraint problem. If this problem has a solution, then it is a point in a solution space (Theoretically the solution space could be a single point). The functions in (4-1) are not assumed either concave or convex. What is desired is an iterative algorithm which, when started at some initial guess at the solution, will at each iteration produce an improved solution from the solution at the last iteration or will indicate that no further improvement can be made. An *improved solution* is defined as one which brings the constraints closer to being satisfied.

Constraint Improvement Algorithm Derivation

Suppose that some initial starting point, x_k^T, has been chosen. Of the m constraints, let n be the number not satisfied by this point. The constraints not satisfied are defined as the *active constraints*, and those satisfied are called the *inactive constraints*. Let J contain the index numbers of the active constraints, i. e., $J = \{k_1, k_2, \ldots, k_n\}$. Essentially what is desired is a directional vector, D, by which the vector x can be changed, and it will be possible to get an improved solution. This vector can be calculated as

$$D = a_1 \nabla g_{k_1} + a_2 \nabla g_{k_2} + \ldots + a_n \nabla g_{k_n} \, . \qquad (4-2)$$

In (4-2)

$$k_1, k_2, \ldots, k_n \in J \quad,$$

∇g_{k_i} denotes the gradient of the constraint corresponding to the k_i index evaluated at x_k^T, and $\{a_k\}$ is a set of constants that are to be determined. An improved solution can be assured if the a's are determined so that

$$D \cdot \nabla g_{k_i} > 0 \qquad i = 1, \ldots, n \quad . \qquad (4-3)$$

In other words the maximum rate of increase of g_{k_i} at x_k^T is in the direction of ∇g_{k_i}, but an increase in g_{k_i} can be registered by traveling in the direction of any vector which has a positive component in the direction of the gradient. In fact, suppose that a value for each of the dot products in (4-3) is chosen. Then (4-3) becomes

$$D \cdot \nabla g_{k_1} = c_1$$

$$D \cdot \nabla g_{k_2} = c_2$$

$$\vdots$$

$$D \cdot \nabla g_{k_n} = c_n \qquad (4-4)$$

where the vector $c^T = (c_1, c_2, \ldots, c_n)$ contains the chosen dot product resultants. Substituting (4-2) into (4-4) results in the following set of linear equations,

$$(\nabla g_{k_1} \cdot \nabla g_{k_1})a_1 + (\nabla g_{k_1} \cdot \nabla g_{k_2})a_2 + \ldots + (\nabla g_{k_1} \cdot \nabla g_{k_n})a_n = c_1$$

$$(\nabla g_{k_2} \cdot \nabla g_{k_1})a_1 + (\nabla g_{k_2} \cdot \nabla g_{k_2})a_2 + \ldots + (\nabla g_{k_2} \cdot \nabla g_{k_n})a_n = c_2$$

$$\vdots$$

$$(\nabla g_{k_n} \cdot \nabla g_{k_1})a_1 + (\nabla g_{kn} \cdot \nabla g_{k_2})a_2 + \ldots + (\nabla g_{k_n} \cdot \nabla g_{k_n})a_n = c_n \quad . \quad (4-5)$$

Using matrix notation (4-5) becomes

$$[\nabla G^T \quad \nabla G]a = c \quad, \tag{4-6}$$

in which $a = [a_1 \ a_2 \ \ldots \ a_n]^T$ and ∇G is a matrix whose columns are composed of the gradients of the active constraints (The matrix $[\nabla G^T \quad \nabla G]$ is the Gramian matrix of the gradient vectors under consideration--see Hildebrand [31].).

If the gradient vectors are linearly independent then

$$a = [\nabla G^T \cdot \nabla G]^{-1} c \quad. \tag{4-7}$$

Hence, this will yield a's for a desired dot product between the directional vector D and each gradient of the active constraints.[9] By moving in the direction of D then it is possible to improve the present solution.*

Algorithm Summation

Using the derivation and the preceding terminology, the constraint improvement algorithm may be summarized as follows:

$$x_{k+1}^T = x_k^T + h[\nabla G]a$$

in which x_{k+1}^T and x_k^T are the solution points at the $(k + 1)^{th}$ and k^{th}

---

*In the above derivation the gradients were used. However, vectors in the directions of the gradients will suffice. In fact, it has been found in practice that unit vectors in the directions of the gradients are more suitable when the gradient magnitudes become disproportioned. The main advantage is a greater convergence rate.

iterations respectively, $[\nabla G]$ is a matrix whose columns are composed of the gradients of the active constraints evaluated at x_k^T ,

$$a = [\nabla G^T \quad \nabla G]^{-1} c$$

where c is a column matrix of positive constants, and h is a positive constant.

The choice of h (the step size constant) determines how much or whether any improvement in the constraints is made. In a compensator design program h also determines whether the program is a TIFR or SIFR algorithm. As a general rule small positive values of h produce a TIFR and larger values of h produce a SIFR. Of course there is a maximum limit on h for producing a SIFR, i. e., values of h above the maximum do not produce either a TIFR or a SIFR. On the other hand, negative values of h are out of the question since they tend to decrease the constraints--making them even worse.

In addition to choosing h, a choice of the components of the c vector must be made. As has been pointed out previously, the components of c are the dot products of the directional vector, D, and the gradients of the active constraints. Thus by properly choosing the c's the amount of increase in some of the constraints can be, to some extent, controlled. In other words by judicious choice of the c's some constraints can be weighted more heavily than others. However, the actual amount of change in a constraint is related to h and the constraints' partial derivatives. In practice it has been found that when using unit vectors in the directions of the gradients of the constraints a good choice of the elements of the c vector is 1's. This choice gives the best convergence rate.

On the other hand, there is nothing wrong with making the c's dependent upon the constraint values, e. g., by letting a c decrease as its constraint comes closer to being satisfied. However, as a c approaches zero the algorithm would tend to determine a direction that was parallel to the boundary of the feasible region. Hence, the probability of the constraint corresponding to this c becoming inactive decreases. Nevertheless, it has been discovered that in many instances that by holding the c's at respectable positive levels many of the constraints are driven to inactivity and they do not return to activity again. In this case the order of the matrix whose inverse is required can be reduced, whereas, if all constraints always linger in activity the order can increase if other constraints become active on higher iterations.

Algorithm Limitations and Termination

Next, attention is focused on algorithm termination. There are three conditions in which the algorithm will terminate. These are

1. All constraints are inactive.

2. One of the gradients of one of the constraints becomes zero.

3. The gradients of the active constraints become linearly dependent.

The first of these simply indicates that a solution has been obtained. The second and third represent relative extremal solutions. In fact, the second one shows that the solution point is a relative extremal of one of the constraints. On the contrary, the third termination condition indicates that at least one of the constraint gradients is a

linear combination of the others' gradients or there are more active constraints than there are variables (This could represent an incompatibility condition.). Whenever 2 or 3 occurs either the solution obtained will have to be accepted or a new starting point will have to be chosen and the algorithm reinitiated.

CHAPTER V

GENERALIZED PARTIAL CALCULATIONS

In essence, the goal of the designer is to pull and push various points on the frequency response until system specifications have been met or until no further improvements can be accomplished by the present compensator. In general, this can be accomplished by pushing and pulling the various points with respect to other points in the complex $GH(j\omega)$ plane. For example, relative stability can be obtained by pushing the points of the stability margins away from the $-1 + j0$ point. On the other hand, the attenuation margins can be improved by pulling these points toward the origin. Similarly, proper phasing could be achieved by attempting to pull or push these points with respect to real axis points. Of course, in some specialized cases it may even be advantageous to pull or push a point with respect to more than one point. Regardless of whether a point is to be pushed or pulled it is necessary to know how these points change with respect to other points in the $GH(j\omega)$ plane. This is especially true if the algorithm in Chapter IV is to be used in perturbing these points.

A point can be pushed or pulled with respect to another point, $- K$, in the complex $GH(j\omega)$ plane by varying the distance squared, $d(\omega)$, between the point and $- K$. In order to determine how this distance changes with respect to the compensator coefficients, consideration is given to the general feedback control system shown in

Figure 1. The open loop frequency response of this system is determined by breaking the feedback loop at α and then calculating

$$GH(j\omega) = C(j\omega)/R(j\omega) \quad . \tag{5-1}$$

Furthermore, to generalize even further in Figure 1 each channel's compensator is assumed to be made up of a product of sub-compensators, i.e., the kth channel's compensator is given as

$$G_k(s) = \prod_{i=1}^{n_k} G_{ki}(s) \quad , \tag{5-2}$$

where n_k is the number of sub-compensators in the kth channel.[*] The uncompensated open loop state frequency response of the kth channel with all channels opened is defined as

$$\frac{\theta_k}{R}(j\omega) = a_k(\omega) + jb_k(\omega) \tag{5-3}$$

where a_k is the real part and b_k is the imaginary part.

From the above equations and statements it then follows that

$$d(\omega) = \left| K + \sum_{k=1}^{j} \{[a_k(\omega) + jb_k(\omega)][\prod_{i=1}^{n_k} G_{ki}(j\omega)]\} \right|^2 \quad . \tag{5-4}$$

By assuming each sub-compensator to be a general rational function of the following form

$$G_{qp}(s) = \frac{x_n s^n + x_{n-1} s^{n-1} + \ldots + x_0}{y_m s^m + y_{m-1} s^{m-1} + \ldots + y_0} \quad , \tag{5-5}$$

---

[*]This is called the factored form of a compensator.

it becomes necessary to derive only how $d(\omega)$ changes with respect to the coefficients of this general compensator, because the change in $d(\omega)$ with respect to any compensators' coefficients will assume the same general form, only differing by the orders, n and m, and the numerical values of the x's and y's. Since $G_{qp}(j\omega)$ is completely independent of all the other compensators, then it may be isolated from the others in (5-4). This is easily done by letting

$$A + jB = K + \sum_{\substack{k=1 \\ k \neq q}}^{j} \{[a_k(\omega) + jb_k(\omega)][\prod_{i=1}^{n_k} G_{ki}(j\omega)]\} \qquad (5\text{-}6)$$

and

$$c + jd = [a_q(\omega) + jb_q(\omega)] \prod_{\substack{i=1 \\ i \neq p}}^{n_q} G_{qi}(j\omega) \qquad . \qquad (5\text{-}7)$$

Using (5-6) and (5-7), (5-4) is rewritten as

$$d(\omega) = \left| A + jB + (c + jd)G_{qp}(j\omega) \right|^2 \qquad . \qquad (5\text{-}8)$$

Substituting (5-5) into (5-8) and carrying out the necessary manipulations (5-8) evolves as

$$d(\omega) = \frac{(\sum_{i=0}^{n} C_i x_i + A \sum_{j=0}^{k} E_{2j} y_{2j} - B \sum_{j=0}^{p} E_{2j+1} y_{2j+1})^2 + (\sum_{i=0}^{n} D_i x_i + A \sum_{j=0}^{p} E_{2j+1} y_{2j+1} + B \sum_{j=0}^{k} E_{2j} y_{2j})^2}{(\sum_{j=0}^{k} E_{2j} y_{2j})^2 + (\sum_{j=0}^{p} E_{2j+1} y_{2j+1})^2} \qquad (5\text{-}9)$$

where in (5-9) $\quad k = m/2 \quad$ and $\quad p = m/2 - 1$

if $\qquad\qquad$ m is even

or $\qquad\qquad\qquad k = (m-1)/2 \quad$ and $\quad p = (m-1)/2$

if $\qquad\qquad$ m is odd;

the C's, D's, and E's are defined by the following sets:

$$\{C_0, C_1, C_2, C_3, C_4, C_5, \ldots\} = \{c, -d\omega, -c\omega^2, d\omega^3, c\omega^4, -d\omega^5, \ldots\}$$

$$\{D_0, D_1, D_2, D_3, D_4, D_5, \ldots\} = \{d, c\omega, -d\omega^2, -c\omega^3, d\omega^4, c\omega^5, \ldots\}$$

$$\{E_0, E_1, E_2, E_3, E_4, E_5, \ldots\} = \{1, \omega, -\omega^2, -\omega^3, \omega^4, \omega^5, \ldots\}.$$

$$(5\text{-}10a,b,c)$$

Next, letting

$$FN1 \ = \ \sum_{i=0}^{n} C_i x_i + A \sum_{j=0}^{k} E_{2j} y_{2j} - B \sum_{j=0}^{P} E_{2j+1} y_{2j+1} \qquad (5\text{-}11)$$

$$FN2 \ = \ \sum_{i=0}^{n} D_i x_i + A \sum_{j=0}^{p} E_{2j+1} y_{2j+1} + B \sum_{j=0}^{k} E_{2j} y_{2j} \qquad (5\text{-}12)$$

$$FD1 \ = \ \sum_{j=0}^{k} E_{2j} y_{2j} \qquad\qquad\qquad (5\text{-}13)$$

$$FD2 \ = \ \sum_{j=0}^{P} E_{2j+1} y_{2j+1} \qquad\qquad\qquad (5\text{-}14)$$

$$FD \ = \ (FD1)^2 + (FD2)^2 \qquad\qquad (5\text{-}15)$$

$$FN \ = \ (FN1)^2 + (FN2)^2 \qquad\qquad (5\text{-}16)$$

then

$$\frac{\partial d(\omega)}{\partial x_q} = \frac{2[FD(A \cdot FN1 + B \cdot FN2) - FN \cdot FD1]E_q}{(FD)^2} \qquad (5\text{-}17)$$

for q even or

$$\frac{\partial d(\omega)}{\partial x_q} = \frac{2[FD(-B \cdot FN1 + A \cdot FN2) - FN \cdot FD2]E_q}{(FD)^2}$$ (5-18)

for q odd and

$$\frac{\partial d(\omega)}{\partial y_q} = \frac{2[FN1 \cdot C_q + FN2 \cdot D_q]}{FD}$$ (5-19)

for q even or odd.

By programming equations (5-4), (5-6), (5-10a,b,c), and (5-11) - (5-19) on the digital computer the partials of $d(\omega)$ with respect to the coefficients $G_{qp}(s)$ can be obtained.[9,10]

The above derivation provides the key for determining how any sub-compensator affects $d(\omega)$ in a first order sense. With a complete comprehension of this derivation it becomes clearly apparent how to proceed either from channel to channel or from sub-compensator to sub-compensator in order to determine the necessary partial derivatives for a particular frequency point. Of course, this process must be completely repeated for each individual frequency point. Once the gradient vectors of each chosen frequency point are determined, then the calculation of the directional vector is accomplished as described in Chapter IV.

CHAPTER VI

COMPENSATOR IMPROVEMENT PROGRAM

The preceding ideas were programmed in a digital computer program called CIP (Compensator Improvement Program). A complete fortran version of this program is contained in the Appendix. The general iterating procedure employed by CIP is as follows:

1. Using the compensator at hand, the program calculates the critical points, i. e., stability margins, attenuation margins and other points of interest.*

2. If this is the first iteration a preselected step size is chosen. Otherwise, a step size is selected according to one of two criteria.

3. Next the active constraints are separated from the inactive constraints.

4. After this, unit vectors in the direction of the gradients with respect to the variable compensator coefficients are obtained (The numerator partials are listed first).

5. Then using a chosen dot product vector the directional vector is determined (For the normalized gradient vectors calculated in 4, a suitable dot product vector has been found to be a vector whose components are 1's).

---

*The other points of interest are frequency response points on which special attention is to be placed, for example, points to be properly phased, certain gain or phase margins, etc.

6. Finally, the directional vector is normalized with respect

to its magnitude; the compensator coefficients are changed

according to the normalized directional vector and the step

size; then, the complete process is repeated.

In order to initiate the program, an input of discrete open loop

frequency responses in the form of frequency and real and imaginary

parts are required. Allowances are made for five channels of such

information with a maximum of 999 points for each channel. This means

that in Step 1 the actual critical points of the frequency response

are not located--only approximate values are found. However, exper-

ience has shown that the approximate values suffice.

In order to determine better approximations to the critical

points the input would require open loop transfer functions (Equation

5-3) for each channel. The more accurate approximations of the

critical points could be found by finding the real roots of equations

of degree 2n, where n is the total number of the open loop system (See

5-1).* For systems above tenth order this is completely impractical

due to the amount of computation time necessary to perform this task.

Furthermore, in many practical situations an experimental discrete

frequency response is the best information available for describing

the system. In other words an experimental frequency response is

obtained, and using this data a transfer function of the system is

approximated.

Also, some initial compensator for each allowable channel is

required. The amount of initial compensation must be enough to

---

*In this discussion it is assummed that due to round-off error
a computer is not capable of getting exact solutions of non-integer
problems.

stabilize the system.* If the system is open loop stable then each initial compensator can be chosen as an equivalent 1 compensator, i.e., the numerator and denominator factors are chosen to be the same. The compensators may be either in a factored or unfactored form (It is apparent that the unfactored form is just a special case of the factored form).

In Step 2 the proper step size is chosen. In the CIP one of two procedures for selecting the step size is employed. These are

a. Require the betterment of all active constraints from the last iteration.

b. Require the sum of the differences of all active constraint values and their desired values to increase from the last iteration (For this sum all active constraints of the $\leq$ form have been changed to the $\geq$ form by multiplying by -1).

Procedure a indicates the program is to be used in the TIFR phase, while procedure b designates the program as SIFR. The choice of the criteria used is left to the designer. If the one chosen is satisfied, the present step size is doubled; provided that the doubling process does not exceed some preselected maximum step size value.** Otherwise, the maximum step size value is utilized. Regardless of which of these occurs the program continues to the next iteration. On the other hand, if the continuance criterion is not satisfied then the step size is halved and the present iteration is repeated if the step size is

---

*If the system is not stable then relative stability has no meaning--although relative instability might.

**The main reason for limiting the step size is to keep the compensator from becoming unstable on a single iteration.

greater than some chosen minimum step size. When the step size

becomes less than the minimum value the program is terminated.[*]

Steps 3, 4, and 5 are simply operations necessary for employ-

ment of the constraint improvement algorithm of Chapter V, whereas,

in Step 6, the compensator coefficients are actually changed. In

Step 5 the reason for reducing the directional vector to a unit

vector is so that the step size actually designates the overall change

in the compensator coefficients. Otherwise this would not be the

case.

The output of the CIP can be controlled to occur at every

iteration or at set increments, i. e., a set number of iterations

can be skipped between outputs. At any iteration at which an output

occurs the following information is printed by the CIP:

1. Iteration number

2. Constraint values

3. Frequencies where constraints occur

4. Desired constraint values

5. Type of constraints

6. Directional vector at the last iteration

7. Compensators at the present iteration

In 5 the type of constraints denotes whether it is a phase margin, a

gain margin, a stability margin, or an attenuation margin, and the

symbols used to denote these are respectively P, G, S, and A.

In the program stability margins are the main vehicles for

determining the relative stability of the system. The concepts of

_____

[*]The program, also, has a maximum iteration termination condition. Since this has no effect on convergence, it was not included.

classical phase and gain margins have been included in the program because in some special cases these can be used to control proper phasing and various dynamical responses of the system. Furthermore, it should be pointed out that the measurement of these concepts is carried out exactly as stability margins, i. e., distances from the $-1 + j0$ point. Of course there is a one-to-one correspondence between this measuring method and the normal methods of measuring phase and gain margins.

CHAPTER VII

LARGE SYSTEM EXAMPLES

In order to illustrate the practical usefulness of CIP, the
improvements of the compensators for large systems are presented.
This is done by way of two examples; the first example is a single
channel system, while the second example is a dual channel system.
The two systems are not the same, although they are very similar.

Single Channel Example

In this example the system under consideration is similar to
that shown in Figure 1, but only one channel is fed back. The
system's dynamics, $\theta_1(s)/R(s)$, are described by the gain vs frequency
and the phase vs frequency plots shown in Figures 4 and 5. This sys-
tem is a model of the Saturn V/S1-C Dry Work Shop at a flight time of
80 seconds. By an inspection of these frequency response plots it
is revealed that this system has several poles near the $j\omega$-axis.
This deduction is based on the spike shaped gain response and the
almost discontinuous changes in the phase response. These poles near
the $j\omega$-axis are due to various sloshing and bending modes of the
vehicle.

This vehicle is inherently open loop unstable. Thus, it is
necessary to use a control scheme, such as depicted by Figure 1, to
stabilize it. Also, unity feedback with a pure gain compensator is
not sufficient to stabilize the system. A compensator with unity
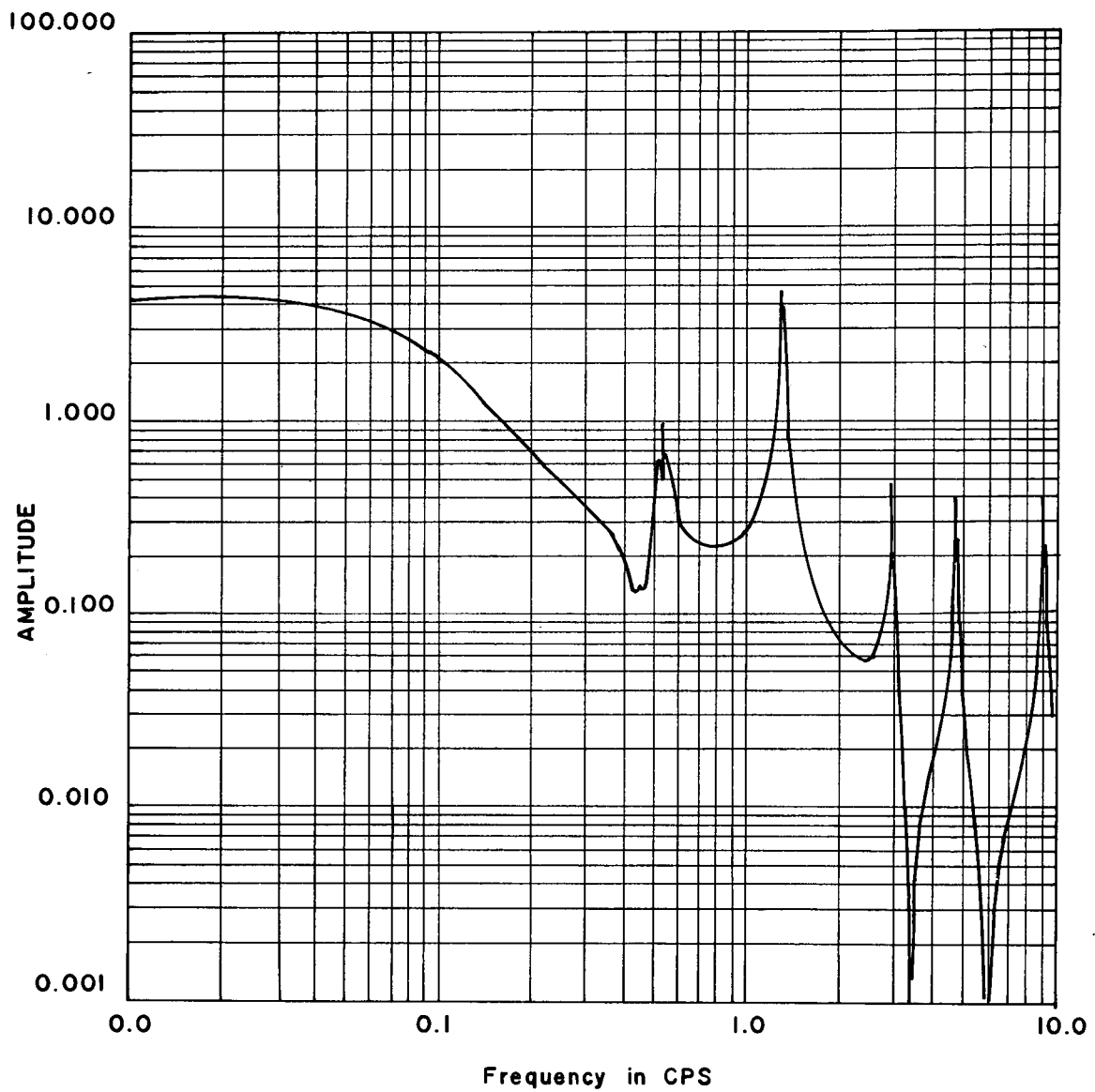
Figure 4. Gain vs Frequency for Uncompensated System
of Single Channel Example
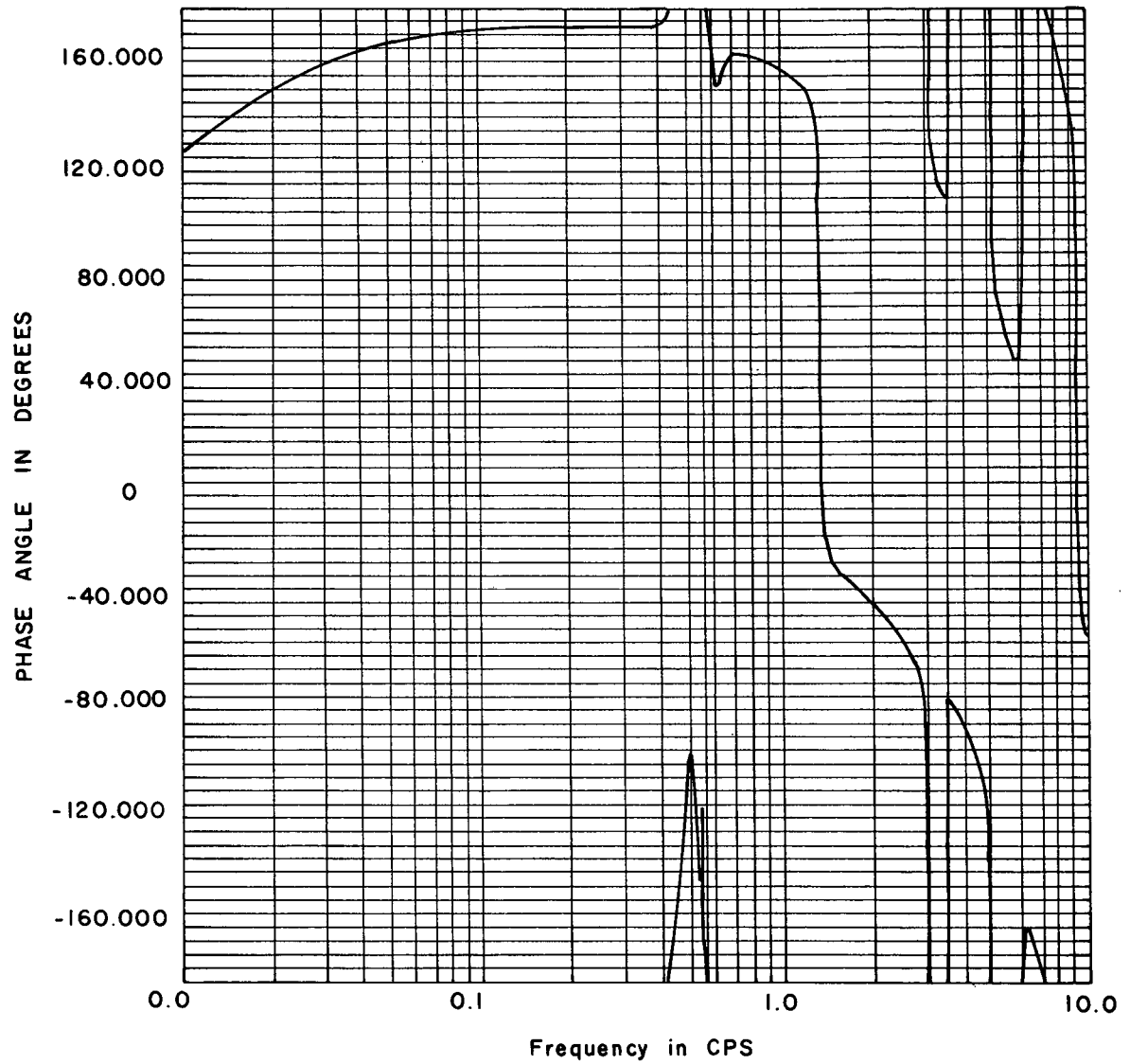
Figure 5. Phase vs Frequency for Uncompensated System of Single Channel Example

feedback which is capable of stabilizing the system is

$$G_c(s) = 0.9 \; \frac{1.0 + 11.79440s + 28.59200s^2}{1.0 + 21.56500s + 6.05650s^2} \; \frac{100.0 + 6.05720s + 7.56640s^2}{100.0 + 10.06500s + 6.32880s^2}$$

$$\frac{1000.0 + 19.08700s + 3.73500s^2}{1000.0 + 330.35200s + 19.02000s^2} \; . \qquad (E1-1)$$

The $GH(j\omega)$ compensated frequency response is shown in Figure 6. Including the compensator, this frequency response represents a 29th over a 35th order system.

In the design of the preceding compensator several physical limitations and constraints were considered—other than just stability of the system (In fact, stabilization of the system can be easily accomplished by a simple lead network with a reduced d. c. gain). Some of these are

1. From past history it is known that compensators with very small d. c. gains produce poor wind responses. An acceptable value of d. c. gain is 0.9.

2. On the $GH(j\omega)$ frequency response the first negative real axis crossing with respect to increasing frequency is called the aerodynamical gain margin. Experimentation has shown that the major effect of an "engine-out" is a reduction of this margin. A safe crossing point is considered as -2 or less (or a frequency response magnitude greater than 2).

3. For a small band of frequencies around 1.199 Hz the frequency response is dominated by the first bending mode. It is desirable to attenuate this band of frequencies. However, to even approach other system requirements and perform this
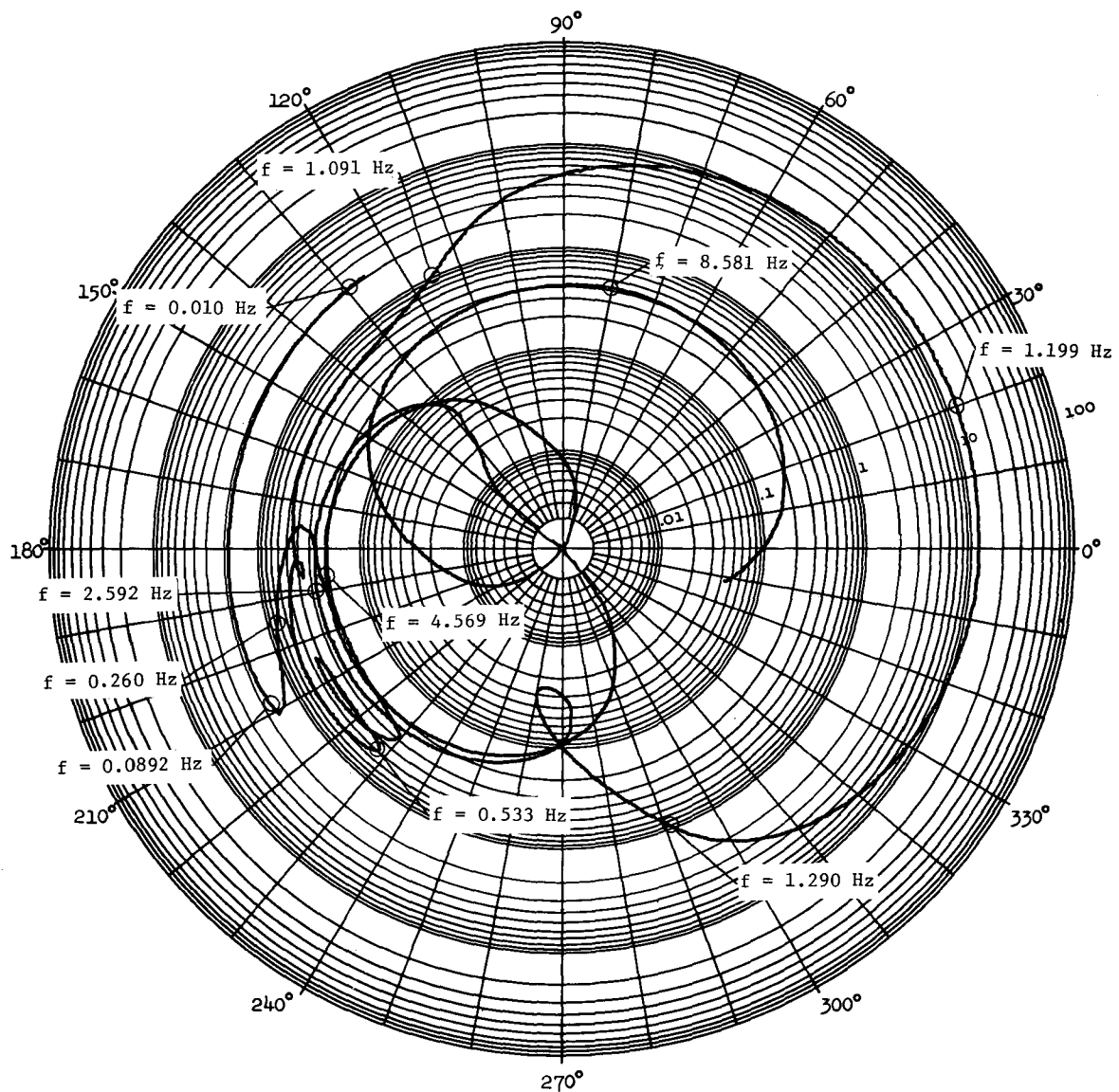
Figure 6. Initial Compensated GH(jω) Frequency Response
for the Single Channel Example

attenuation has been practically impossible. It has been found that the same effect results if this band of frequencies is phased in the right half of the GH(jω) plane. Due to the fact that the frequency of this mode is not known exactly, it is necessary to require larger phase margins for this mode than normally required. Acceptable margins are a lead phase margin of about 55° and a lag phase margin of about 90° (The reason for the difference is that in most physical systems phase lag is more probable to occur than phase lead).

4. For frequencies greater than 2.1 Hz the GH(jω) frequency response is dominated by the higher order bending modes. The control system can be deterred from resonating at any of these higher modes by attenuating to a certain degree all frequencies above 2.1 Hz. These frequencies are considered satisfactorily attenuated if the magnitude of the GH(jω) frequency response is less than 0.25 for f > 2.1 Hz.

5. Besides the above frequency response requirements, it is desirable for all stability margins to be 0.5 or greater (Notice that in terms of classical stability margins this is approximately equivalent to having phase margins of 30° and gain margins of 2 or better).

By an observation of Figure 6 it becomes evident that all of the above specifications are not met. This becomes even more obvious after an inspection of Table 1. In this table the first margin is the aerodynamical gain margin and the next two margins are the lead and lag

RELATIVE STABILITY INFORMATION

MARGIN RADIUS NO. 1 = .88436	FREQUENCY = .03825 HZ	DESIRED MARGIN = 1.00000
MARGIN RADIUS NO. 2 = 1.13788	FREQUENCY = 1.09081 HZ	DESIRED MARGIN = 0.90000
MARGIN RADIUS NO. 3 = 1.60341	FREQUENCY = 1.29000 HZ	DESIRED MARGIN = 1.30000
MARGIN RADIUS NO. 4 = .88436	FREQUENCY = .03825 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 5 = .94555	FREQUENCY = .08420 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 6 = .78641	FREQUENCY = .11000 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 7 = .31834	FREQUENCY = .26000 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 8 = .63863	FREQUENCY = .52387 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 9 = .47723	FREQUENCY = .53646 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 10 = .51104	FREQUENCY = .69241 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 11 = .51744	FREQUENCY = .75059 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 12 = 2.15979	FREQUENCY = 1.14114 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 13 = .99402	FREQUENCY = 1.88572 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 14 = .72616	FREQUENCY = 2.59447 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 15 = .99999	FREQUENCY = 3.61625 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 16 = .78255	FREQUENCY = 4.57294 HZ	DESIRED MARGIN = .50500
MARGIN RADIUS NO. 17 = .84650	FREQUENCY = 8.51837 HZ	DESIRED MARGIN = .50500

ATTENUATED FREQUENCY INFORMATION

MARGIN RADIUS NO. 18 = .28531	FREQUENCY = 2.59236 HZ	DESIRED MARGIN = 0.25000
MARGIN RADIUS NO. 19 = .00001	FREQUENCY = 3.61625 HZ	DESIRED MARGIN = 0.25000
MARGIN RADIUS NO. 20 = .00001	FREQUENCY = 3.61766 HZ	DESIRED MARGIN = 0.25000
MARGIN RADIUS NO. 21 = .22753	FREQUENCY = 4.56938 HZ	DESIRED MARGIN = 0.25000
MARGIN RADIUS NO. 22 = .00020	FREQUENCY = 6.34472 HZ	DESIRED MARGIN = 0.25000
MARGIN RADIUS NO. 23 = .39610	FREQUENCY = 8.58137 HZ	DESIRED MARGIN = 0.25000

Table 1.   Initial Tableau of Single Channel Example

phase margins of the 1st bending mode, respectively.[*] The remaining

margins listed under attenuated frequency information are stability

margins as defined in this paper, and of course the attenuated infor-

mation is representative of the attenuation margins above f = 2.1 Hz.

In the CIP program the following specifications were made:

1.   Determine the aerodynamical gain margin and improve it if

     it is less than 2. In order to improve any point it is

     necessary to specify what point or points in the complex

     GH(jω) plane this point is to be pulled or pushed with

     respect to. For this example it is chosen to push this

     point with respect to the -1 + j0 point.

2.   Determine the lead and lag phase margins of the first bend-

     ing mode and improve either or both if they fall below 0.9

     and 1.3, respectively. To improve these it is chosen to

     push them from the -1 + j0 point.

3.   Detect all stability margins and increase those less than

     0.505. Again the -1 + j0 point is chosen as a pushing point.

4.   Detect all attenuation margins for f > 2.1 Hz. and decrease

     all of those greater than 0.25. For these margins the origin of

     the GH(jω) plane is chosen as a pulling point.

---

[*] The measurements of these stability margins are made in the
same manner as stability margins defined in Chapter II; i.e., the
distance from the -1 + j0 point. Measuring gain margins in this way
is quite natural. However, measuring phase margins in this way is not
as straight forward, even though there is a one to one correspondence.
The equations relating the two are: $d = 2 \sin \theta/2$ and $\theta = 2 \arcsin d/2$,
where d is the distance from the -1 + j0 point and θ is the phase
margin. Of course d is limited to the closed interval [0,2].

The continuance criterion chosen was b of Chapter VI. With these insertions and the necessary frequency response information in CIP, the following compensator was obtained after 2000 iterations (or approximately 30 minutes on a UNIVAC 1106):

$$G_c(s) = 0.9 \frac{1.0 + 74.40524s + 107.13383s^2}{1.0 + 124.68711s + 16.85849s^2} \frac{100.0 + 7.29719s + 8.68710s^2}{100.0 + 11.98668s + 9.15484s^2}$$

$$\frac{1000.0 + 12.10541s + 3.11162s^2}{1000.0 + 219.54201s + 20.42297s^2} \qquad . \qquad (E1-2)$$

A tableau of the pertinent information at iteration 2000 is shown in Table 2. From this tableau it is seen that most margins are, for practical purposes, satisfied. The reason that several of the margins have values that are only approximately equal to the desired values is that, in most instances, after a margin becomes inactive it has a tendency to oscillate between activity and inactivity on higher iterations. However, by establishing an upper limit on the step size from iteration to iteration these constraints are coerced to remain in a vicinity of their desired values (For this example the maximum step size was chosen as 0.1 for the first 1000 iterations; then, to speed up convergence it was changed to 0.2 for the next 1000 iterations).

The three smallest stability margins do not belong in the same category as those mentioned above because at no time were they inactive. Since program termination was maximum iterations, additional improvements in these constraints is quite conceivable. Nevertheless, the convergence curve shown in Figure 7 indicates many more iterations will be required before any appreciable change in the smallest of these

RELATIVE STABILITY INFORMATION

MARGIN RADIUS NO. 1 = .99924   FREQUENCY = .04825 HZ   DESIRED MARGIN = 1.00000
MARGIN RADIUS NO. 2 = 1.03082   FREQUENCY = 1.04660 HZ   DESIRED MARGIN = .90000
MARGIN RADIUS NO. 3 = 1.45419   FREQUENCY = 1.31500 HZ   DESIRED MARGIN = 1.30000
MARGIN RADIUS NO. 4 = .52214   FREQUENCY = .11000 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 5 = .47025   FREQUENCY = .17000 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 6 = .46513   FREQUENCY = .20000 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 7 = .47731   FREQUENCY = .26000 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 8 = .51368   FREQUENCY = .31400 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 9 = .66040   FREQUENCY = .53119 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 10 = .50547   FREQUENCY = .53646 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 11 = .50256   FREQUENCY = .54039 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 12 = .50323   FREQUENCY = .55165 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 13 = .50457   FREQUENCY = .63616 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 14 = .53471   FREQUENCY = .68002 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 15 = .50412   FREQUENCY = .79459 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 16 = 2.59072   FREQUENCY = 1.14114 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 17 = .97314   FREQUENCY = 1.88572 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 18 = .76936   FREQUENCY = 2.58814 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 19 = .99906   FREQUENCY = 3.23614 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 20 = .85506   FREQUENCY = 4.57294 HZ   DESIRED MARGIN = .50500
MARGIN RADIUS NO. 21 = .91530   FREQUENCY = 8.51837 HZ   DESIRED MARGIN = .50500

ATTENUATED FREQUENCY INFORMATION

MARGIN RADIUS NO. 22 = .25246   FREQUENCY = 2.59236 HZ   DESIRED MARGIN = .25000
MARGIN RADIUS NO. 23 = .00000   FREQUENCY = 3.61625 HZ   DESIRED MARGIN = .25000
MARGIN RADIUS NO. 24 = .00001   FREQUENCY = 3.61766 HZ   DESIRED MARGIN = .25000
MARGIN RADIUS NO. 25 = .14682   FREQUENCY = 4.56938 HZ   DESIRED MARGIN = .25000
MARGIN RADIUS NO. 26 = .00013   FREQUENCY = 6.34472 HZ   DESIRED MARGIN = .25000
MARGIN RADIUS NO. 27 = .25008   FREQUENCY = 8.58137 HZ   DESIRED MARGIN = .25000

Table 2.   Tableau of Single Channel Example at Iteration 2000
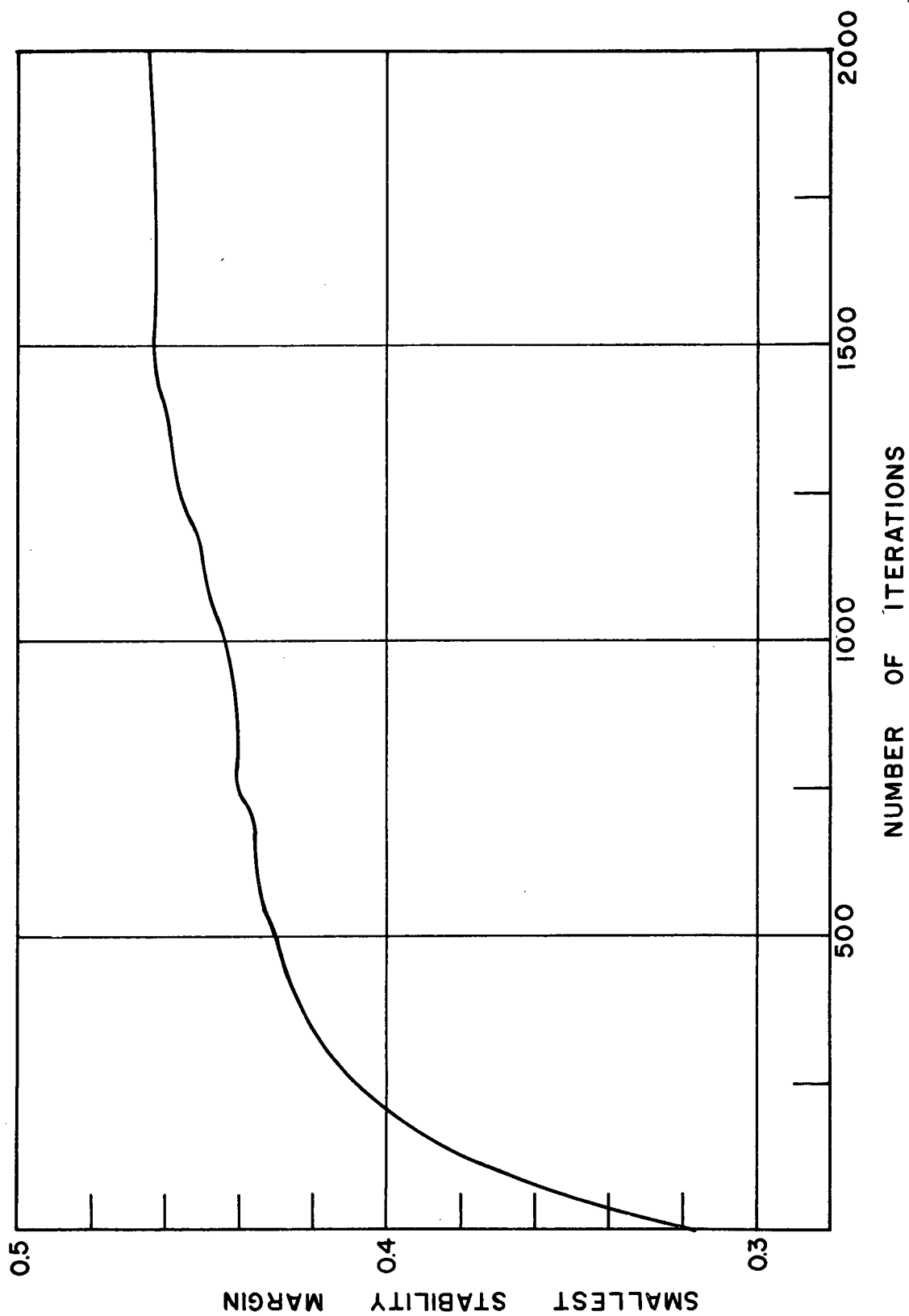
Figure 7. Convergence Curve for Single Channel Example

margins is recorded. With an occurrence such as this the designer is left with three alternatives:

1. Accept the present design.

2. Pay the toll of additional computer time and attempt additional iterations.

3. Change some of the desired constraints and continue the program.

From experience it has been found that small changes in the desired margins can result in marked effects.* As for the case under discussion the GH($j\omega$) frequency response in Figure 8 reveals that for practical purposes the compensator for iteration 2000 is satisfactory.[10]

Dual Channel Example

Again reference is made to Figure 1, except in this case it is assumed that j = 2, i. e., two channels are fed back. The uncompensated open loop system is described by the gain and phase frequency responses shown in Figures 9, 10, 11, and 12. Figures 9 and 10 represent the gain and phase plots of $\theta_1(s)/R(s)$, while Figures 11 and 12 are the gain and phase plots of $\theta_2(s)/R(s)$. This system is typical of the Saturn V/S1-C Sky Lab at a flight time of 105 seconds.

---

*It should be noted that at the end of iteration 2000 the CIP was slightly modified so that a better calculation of the first negative real axis crossing frequency was obtained. After this, additional iterations were attempted and in less than 50 iterations the smallest stability margin was increased from 0.46513 to 0.48177. In another instance the compensator whose smallest stability margin was 0.48177 was used as the starting compensator in another run in which the relative stability requirements were lowered to 0.49 while the other system requirements were the same as previously stated. In less than 50 iterations all system requirements were completely satisfied.

Figure 8. GH(jω) Compensated Frequency Response at
Iteration 2000 for the Single Channel Example
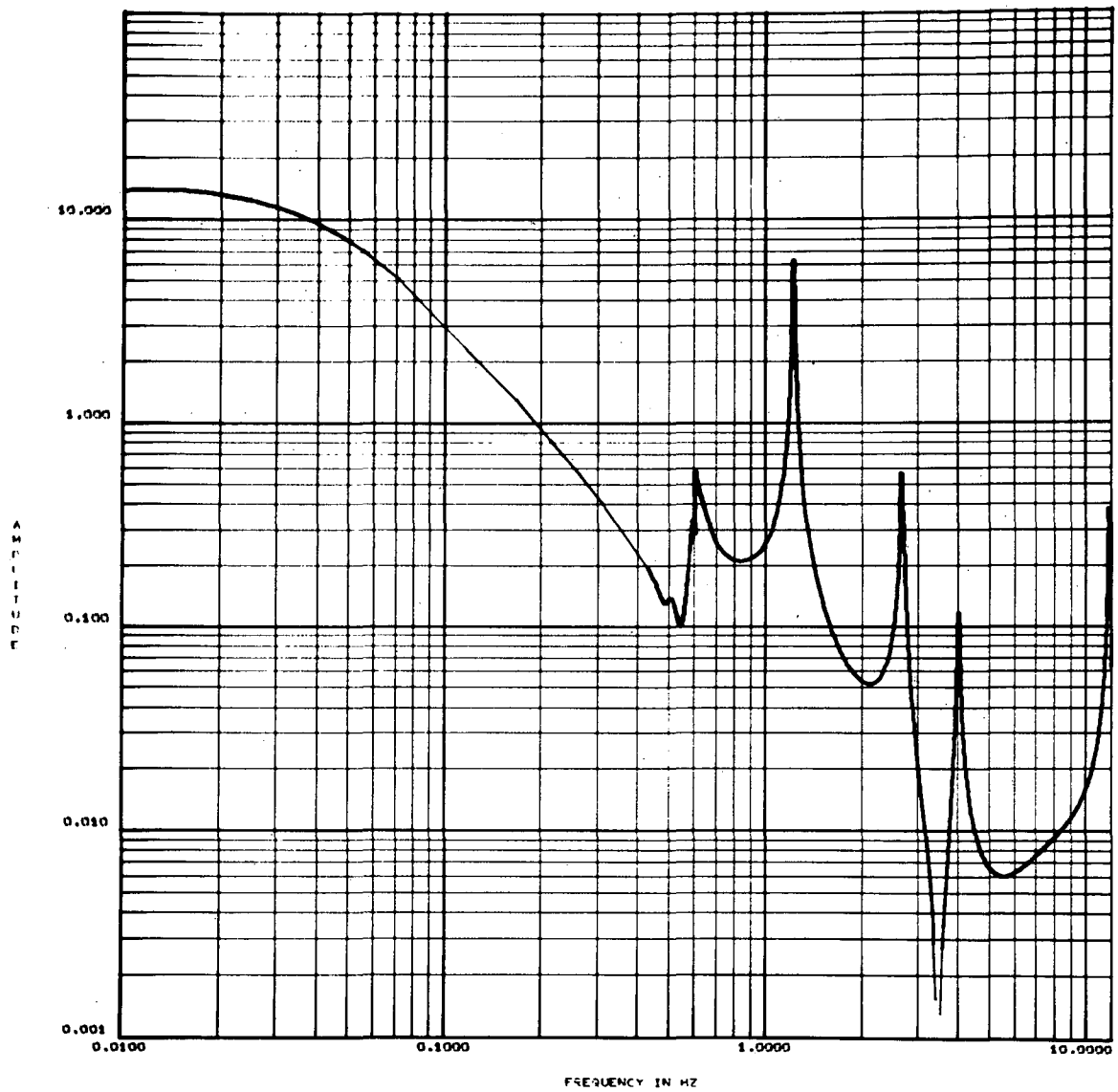
Figure 9. Gain vs Frequency for Channel 1 of
Uncompensated System of Dual Channel Example
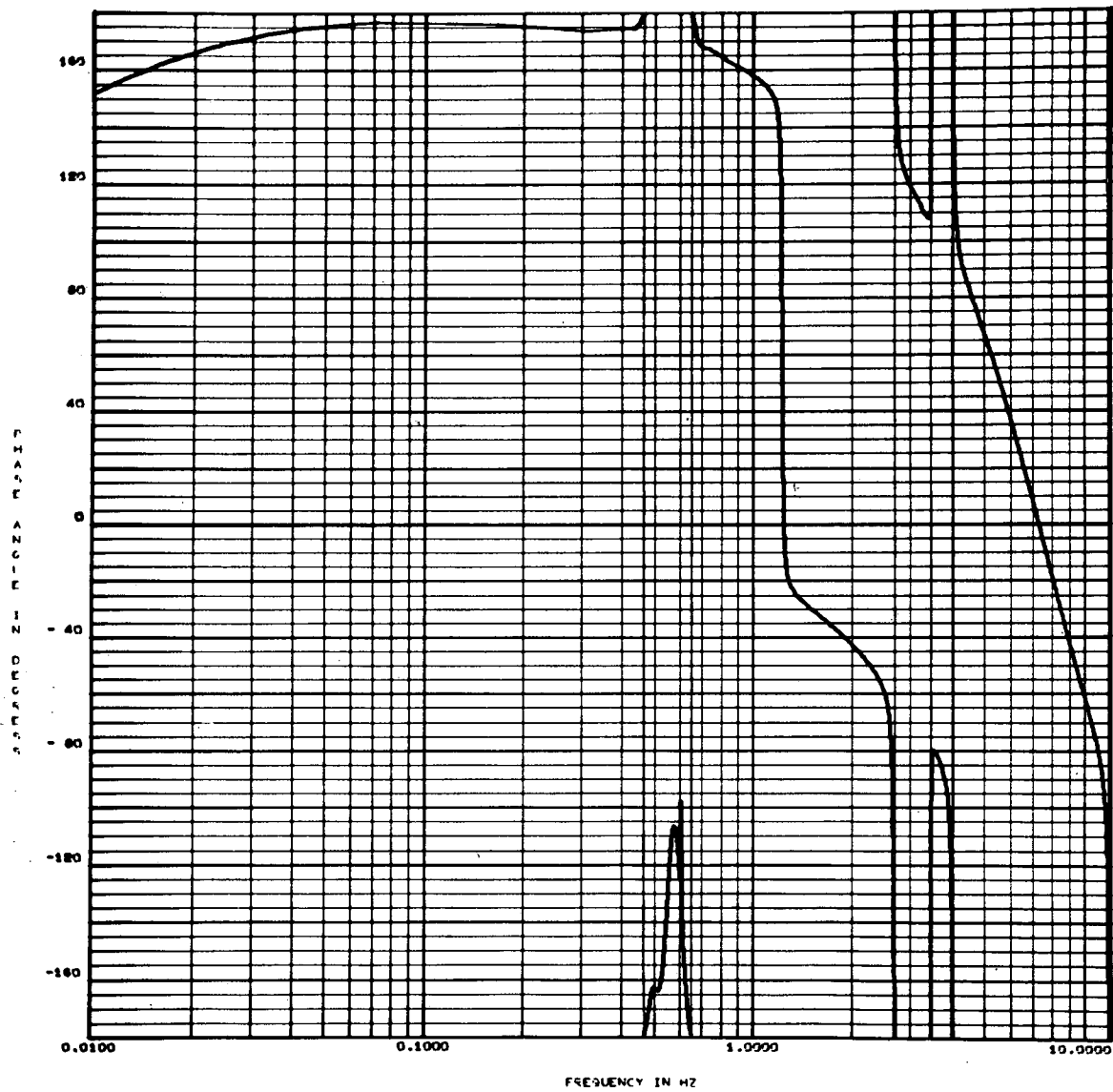
Figure 10. Phase vs Frequency for Channel 1 of Uncompensated System of Dual Channel Example
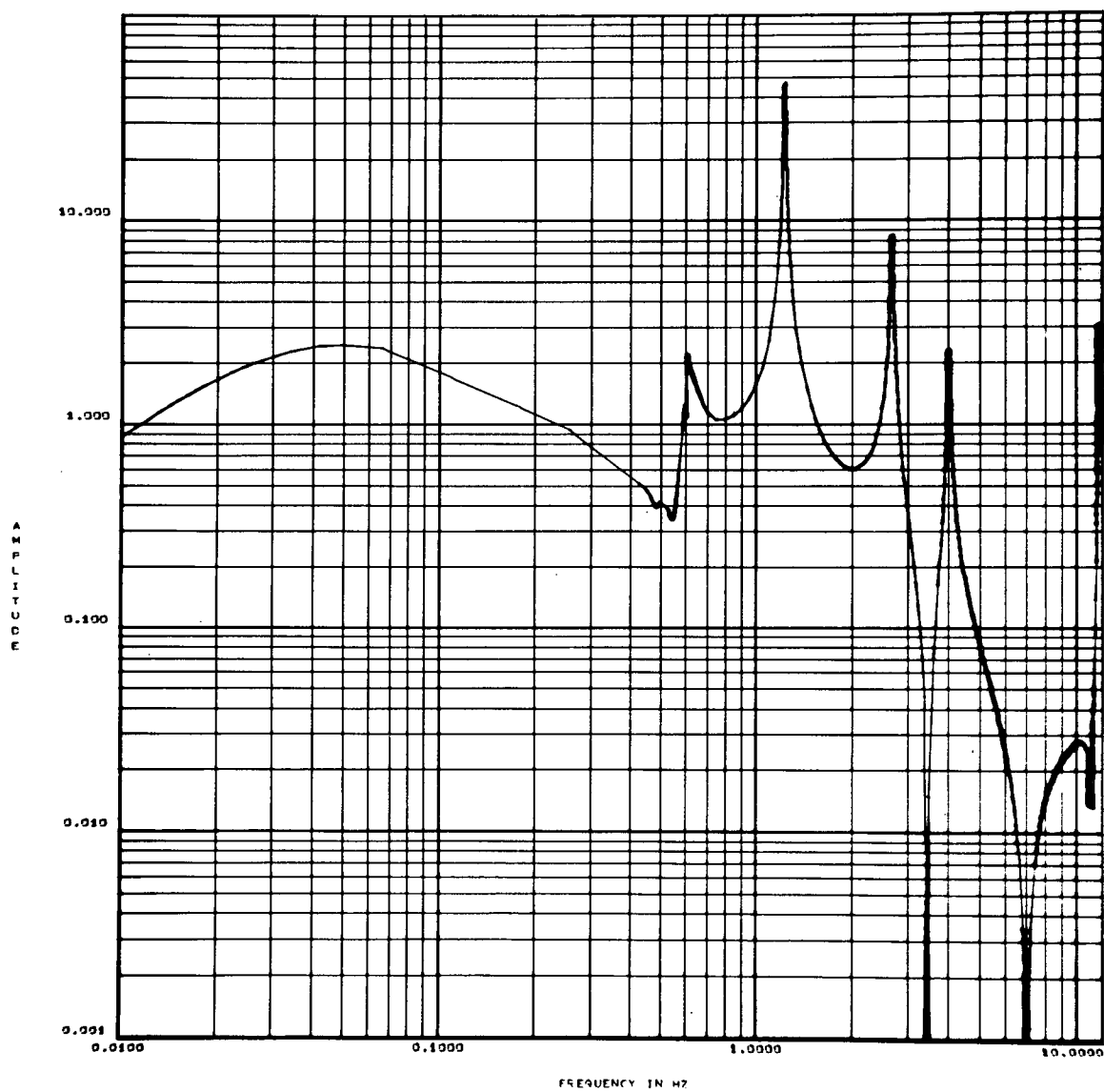
Figure 11.   Gain vs Frequency for Channel 2 of Uncom-
pensated System of Dual Channel Example
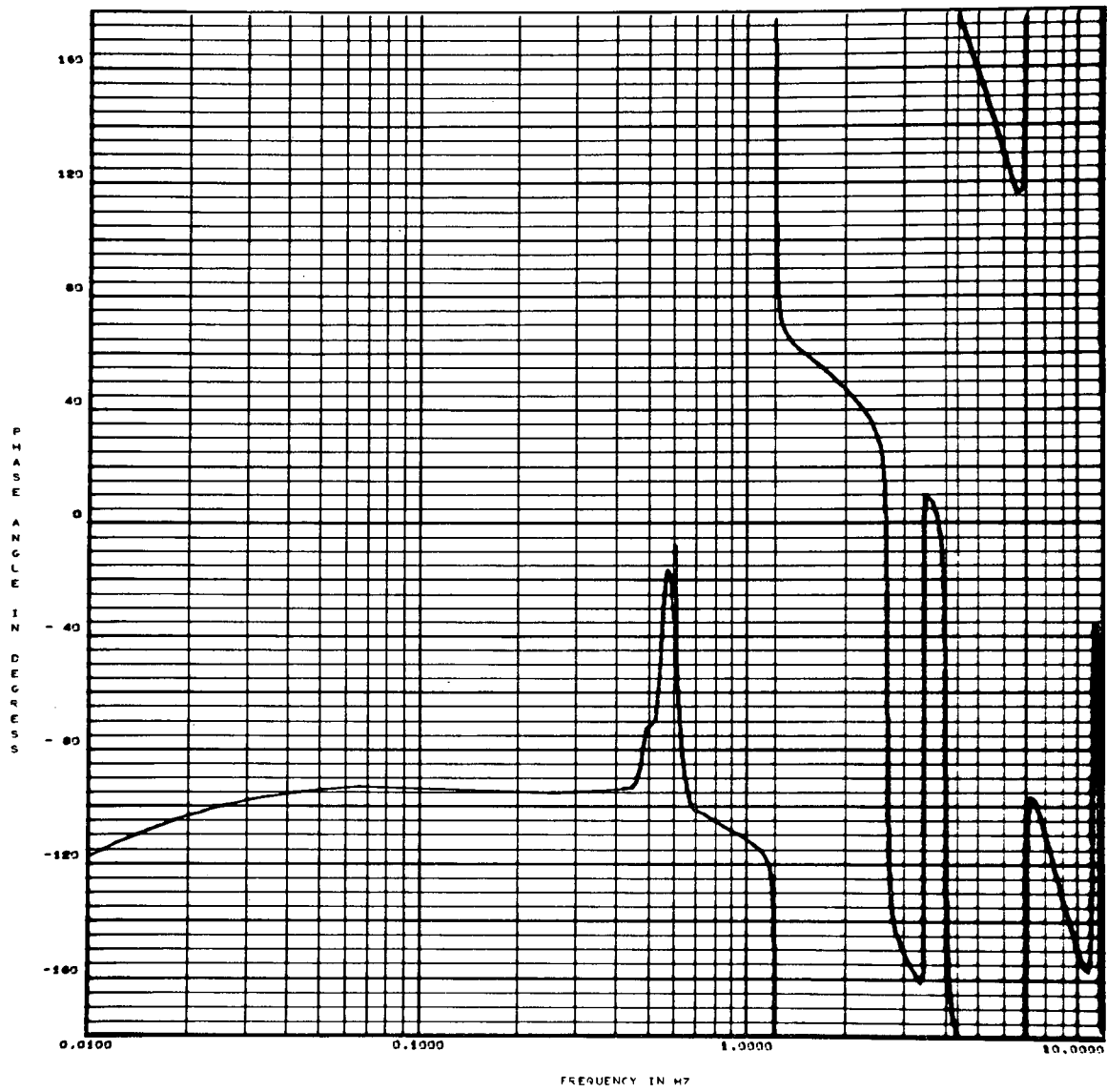
Figure 12. Phase vs Frequency for Channel 2 of Uncom-
pensated System of Dual Channel Example

Compensators which have been designed for this system are

$$G_1(s) = 1.26 \frac{1000.0 + 6.54732s + 4.5732s^2}{100.0 + 1.43424s} \frac{100.0 + 6.04029s}{100.0 + 6.17455s}$$

$$\frac{10.0 + 3.69000s}{10.0 + 2.32980s} \frac{0.1 + 1.04000s}{0.1 + 2.33536s} \frac{1.0}{10.0 + 1.05603s}$$

$$\frac{100.0}{100.0 + 4.13275s} \qquad (E2\text{-}1)$$

$$G_2(s) = 0.58 \frac{1000.0 + 2.91040s + 4.50787s^2}{100.0 + 3.52502s} \frac{100.0 + 4.71096s}{100.0 + 4.61899s}$$

$$\frac{10.0}{100.0 + 5.49396s} \frac{10.0}{10.0 + 1.21426s}$$

$$\frac{10.0}{10.0 + 2.85080s} . \qquad (E2\text{-}2)$$

With these compensators inserted in the system the compensated open loop $GH(j\omega)$ frequency response, $C(j\omega)/R(j\omega)$, with the loop broken at α is that shown in Figure 13.

It is desired to make several improvements in this frequency response. These conditional improvements are

1. Keep the aerodynamical gain margin at 4.37 or greater.

2. Increase all stability margins of 0.49 or less.

3. Maintain the lead and lag phase margins of the first bending mode at 55° and 90° or better.

4. Decrease all attenuation margins occurring at frequencies above 2.0 hz when 0.2 or greater.

RC
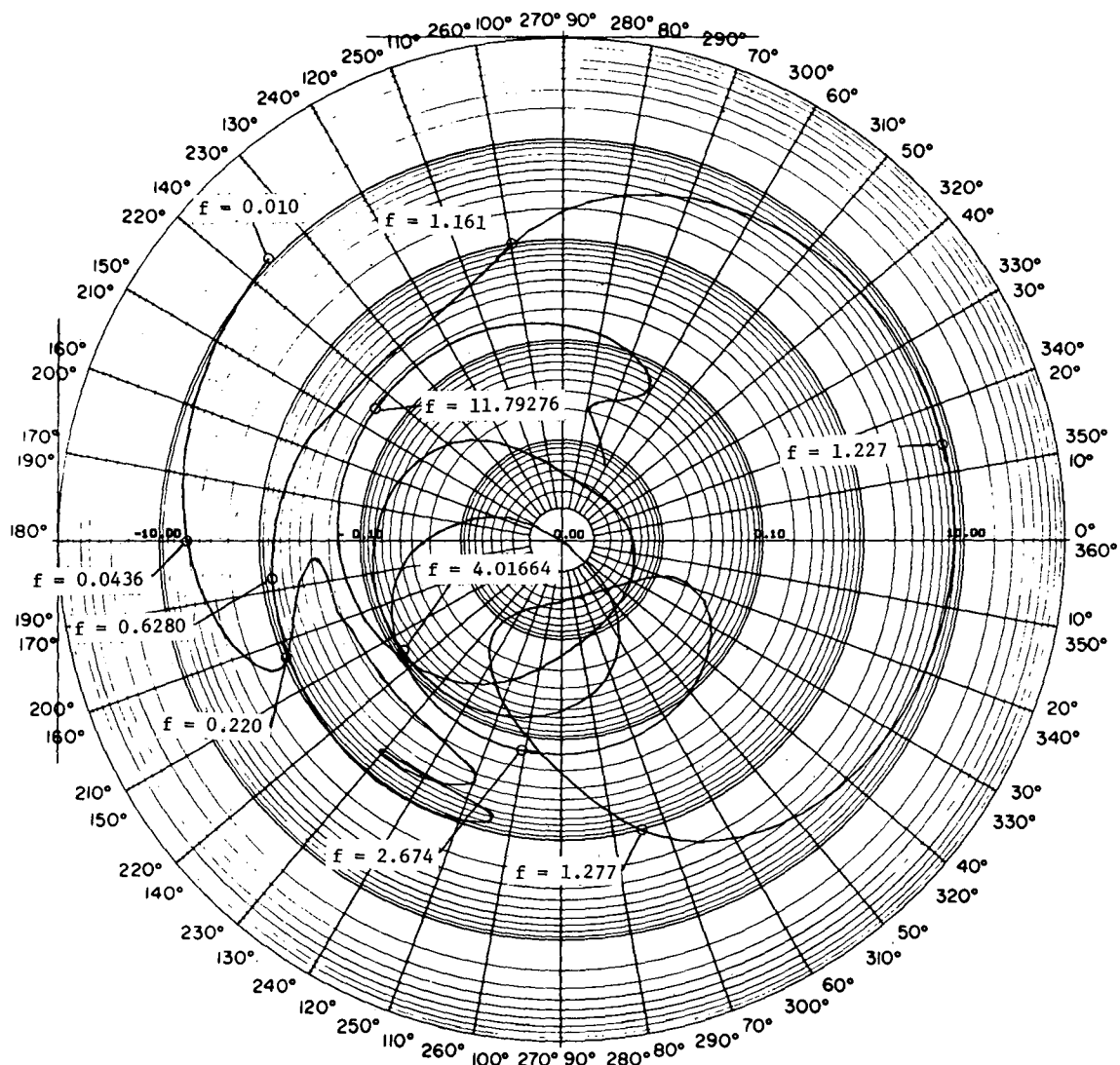
Figure 13. Initial GH(jω) Compensated Frequency Response
for the Dual Channel Example

In order to make these improvements the following specifications are made in CIP:

1. Whenever the aerodynamical gain margin is 4.8 or less it is pulled with respect to the -7 -j3 point and pushed with respect to the -1 + j0 point.

2. All stability margin points less than 0.49 are pushed with respect to the -1 + j0.

3. The lead and lag phase margins are pulled with respect to the 1 + j0 point when less than 0.9 and 1.3 respectively. Also, the attenuation margins occurring at frequencies between these two are decreased by pulling with respect to the origin of the GH(jω) plane if they are greater than 9.0.

4. The attenuation margins above 2.0 hz are decreased by pulling them with respect to the origin.

With these specifications, 357 frequency response points for each channel, and the initial compensators, (E2-1) and (E2-2), in the CIP, the following compensators were obtained after 200 iterations or about 10 minutes on a Univac 1106:

$$G_2(s) = 1.26 \frac{1000.0 + 7.07293s + 7.02583s^2}{100.0 + 1.21230s} \frac{100.0}{100.0 + 10.48567s}$$

$$\frac{10.0 + 3.43938s}{10.0 + 1.14372s} \frac{0.1 + 1.21370s}{0.1 + 2.51497s}$$

$$\frac{1.0}{10.0 + 1.14372s} \frac{100.0}{100.0 + 9.34985s} \qquad (E2-3)$$

$$G_2(s) = 0.58 \frac{1000.0 + 6.74527s + 4.53868s^2}{100.0 + 0.0s} \frac{100.0 + 4.57638s}{100.0 + 1.26840s}$$

$$\frac{10.0}{100.0 + 6.96980s} \frac{10.0}{10.0 + 1.44585s}$$

$$\frac{10.0}{10.0 + 1.44585s} . \tag{E2-4}$$

An evaluation of the amount of improvement can be made by comparing the initial tableau, Table 3, of important information to the final tableau, Table 4. As in the last example the first margin is the aerodynamical gain margin, and the next two margins are the lead and lag phase margins of the first bending mode respectively. The remaining margins under relative stability information are listed as stability margins. The margins under the attenuated frequency information are the attenuation margins above 1.2 hz. The desired margins' values are listed in the right hand column.

Taking into account the desired improvements it is seen that significant improvement has been made. Furthermore, this is reinforced by comparing the initial compensated frequency response, Figure 13, to the compensated frequency response at iteration 200, Figure 14. The termination reason was maximum iterations; thus, as in the first example the designer is left with the same three alternatives. From the convergence curve shown in Figure 15, it appears that several additional iterations may have to be attempted before any significant improvement in the smallest stability margin is observed. The importance of this example is the significant improvement over the initial frequency response.

RELATIVE STABILITY INFORMATION

MARGIN RADIUS NO. 1	= 4.36704	FREQUENCY	= .04357 HZ	DESIRED MARGIN	= 4.80000
MARGIN RADIUS NO. 2	= 1.25052	FREQUENCY	= 1.16100 HZ	DESIRED MARGIN	= .90000
MARGIN RADIUS NO. 3	= 1.60738	FREQUENCY	= 1.27658 HZ	DESIRED MARGIN	= 1.30000
MARGIN RADIUS NO. 4	= 4.36704	FREQUENCY	= .04357 HZ	DESIRED MARGIN	= 4.80000
MARGIN RADIUS NO. 5	= .38275	FREQUENCY	= .22000 HZ	DESIRED MARGIN	= .49000
MARGIN RADIUS NO. 6	= .78335	FREQUENCY	= .49542 HZ	DESIRED MARGIN	= .49000
MARGIN RADIUS NO. 7	= .77835	FREQUENCY	= .59614 HZ	DESIRED MARGIN	= .49000
MARGIN RADIUS NO. 8	= .20846	FREQUENCY	= .62803 HZ	DESIRED MARGIN	= .49000
MARGIN RADIUS NO. 9	= .98917	FREQUENCY	= 1.98788 HZ	DESIRED MARGIN	= .49000
MARGIN RADIUS NO. 10	= .92867	FREQUENCY	= 2.68738 HZ	DESIRED MARGIN	= .49000
MARGIN RADIUS NO. 11	= .92365	FREQUENCY	= 2.69525 HZ	DESIRED MARGIN	= .49000
MARGIN RADIUS NO. 12	= .91694	FREQUENCY	= 4.02962 HZ	DESIRED MARGIN	= .49000
MARGIN RADIUS NO. 13	= .81610	FREQUENCY	= 11.76031 HZ	DESIRED MARGIN	= .49000

ATTENUATED FREQUENCY INFORMATION

MARGIN RADIUS NO. 14	= 8.10120	FREQUENCY	= 1.22681 HZ	DESIRED MARGIN	= 9.00000
MARGIN RADIUS NO. 15	= .13636	FREQUENCY	= 2.67400 HZ	DESIRED MARGIN	= .20000
MARGIN RADIUS NO. 16	= .08938	FREQUENCY	= 4.01664 HZ	DESIRED MARGIN	= .20000
MARGIN RADIUS NO. 17	= .00601	FREQUENCY	= 8.39067 HZ	DESIRED MARGIN	= .20000
MARGIN RADIUS NO. 18	= .20468	FREQUENCY	= 11.79276 HZ	DESIRED MARGIN	= .20000
MARGIN RADIUS NO. 19	= .02996	FREQUENCY	= 11.99931 HZ	DESIRED MARGIN	= .20000

Table 3. Initial Table of Values for Dual Channel Example

ITERATION NO. 200

RELATIVE STABILITY INFORMATION

MARGIN RADIUS NO. 1 = 4.67660 FREQUENCY = .04357 HZ DESIRED MARGIN = 4.80000
MARGIN RADIUS NO. 2 = .94284 FREQUENCY = 1.15407 HZ DESIRED MARGIN = .90000
MARGIN RADIUS NO. 3 = 1.80467 FREQUENCY = 1.28287 HZ DESIRED MARGIN = 1.20000
MARGIN RADIUS NO. 4 = 4.67660 FREQUENCY = .04357 HZ DESIRED MARGIN = 4.80000
MARGIN RADIUS NO. 5 = .45316 FREQUENCY = .18500 HZ DESIRED MARGIN = .49000
MARGIN RADIUS NO. 6 = .44542 FREQUENCY = .20000 HZ DESIRED MARGIN = .49000
MARGIN RADIUS NO. 7 = .81087 FREQUENCY = .49542 HZ DESIRED MARGIN = .49000
MARGIN RADIUS NO. 8 = .95164 FREQUENCY = .59614 HZ DESIRED MARGIN = .49000
MARGIN RADIUS NO. 9 = .27879 FREQUENCY = .64219 HZ DESIRED MARGIN = .49000
MARGIN RADIUS NO. 10 = 1.00605 FREQUENCY = 2.03824 HZ DESIRED MARGIN = .49000
MARGIN RADIUS NO. 11 = .90372 FREQUENCY = 2.69603 HZ DESIRED MARGIN = .49000
MARGIN RADIUS NO. 12 = .92889 FREQUENCY = 4.04497 HZ DESIRED MARGIN = .49000
MARGIN RADIUS NO. 13 = .83048 FREQUENCY = 11.73670 HZ DESIRED MARGIN = .49000

ATTENUATED FREQUENCY INFORMATION

MARGIN RADIUS NO. 14 = 9.14419 FREQUENCY = 1.22642 HZ DESIRED MARGIN = 9.00000
MARGIN RADIUS NO. 15 = 9.15694 FREQUENCY = 1.22681 HZ DESIRED MARGIN = 9.00000
MARGIN RADIUS NO. 16 = .20385 FREQUENCY = 2.67400 HZ DESIRED MARGIN = .20000
MARGIN RADIUS NO. 17 = .17308 FREQUENCY = 2.69053 HZ DESIRED MARGIN = .20000
MARGIN RADIUS NO. 18 = .11440 FREQUENCY = 4.02166 HZ DESIRED MARGIN = .20000
MARGIN RADIUS NO. 19 = .20409 FREQUENCY = 11.77801 HZ DESIRED MARGIN = .20000

Table 4. Iteration 200 Tableau of Values for Dual Channel Example
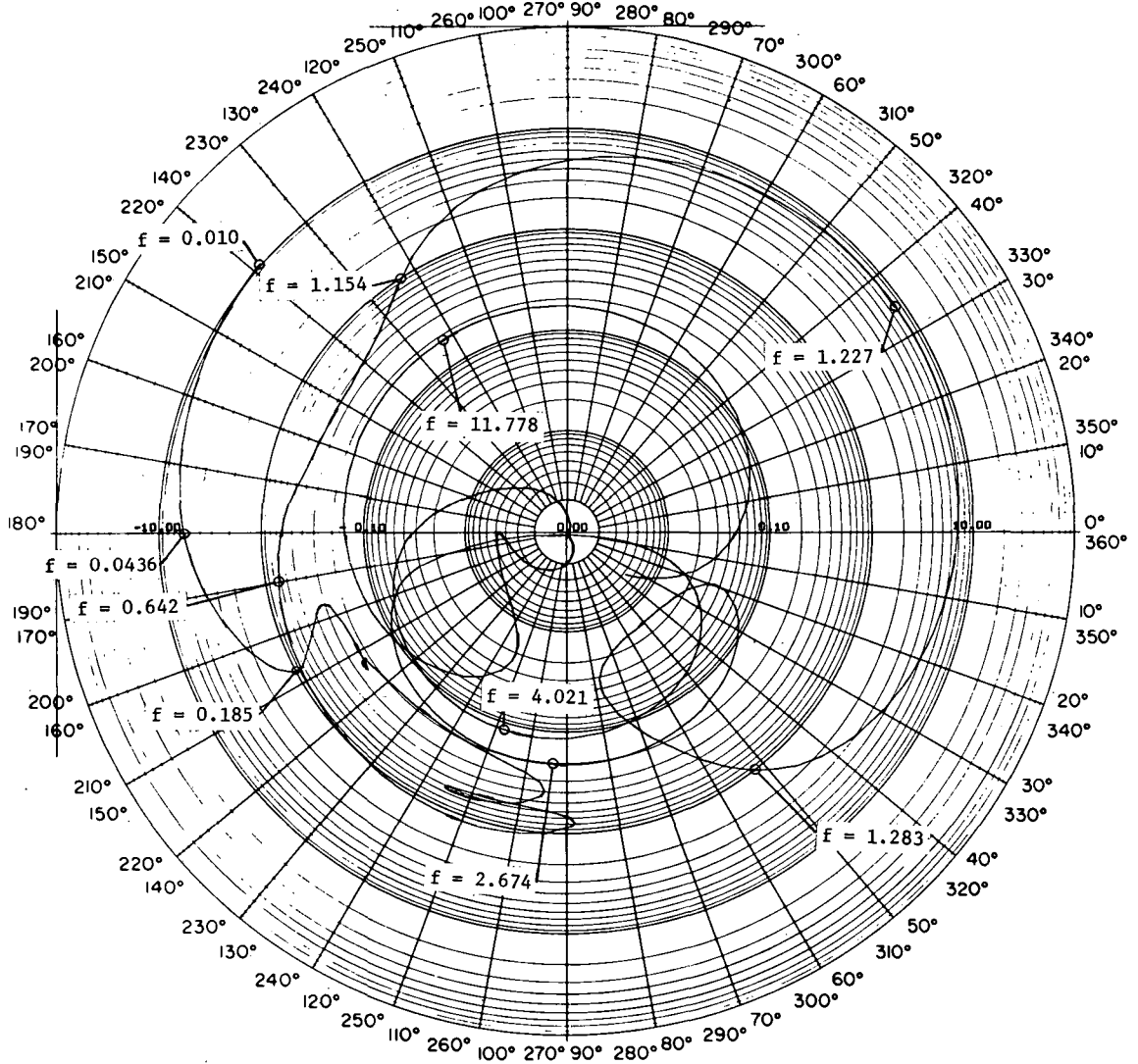
Figure 14. GH(jω) Compensated Frequency Response at
Iteration 200 for Dual Channel Example
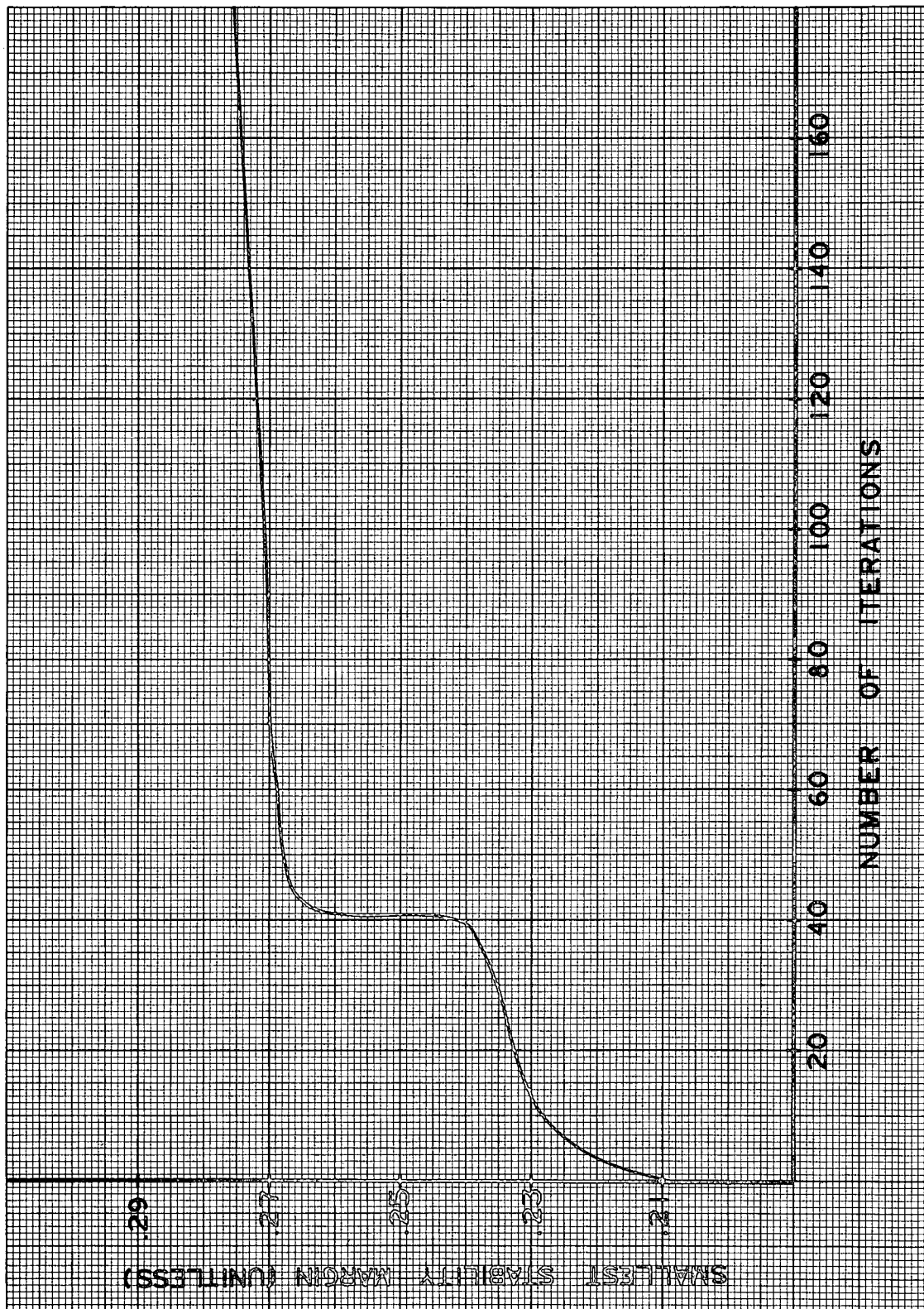
Figure 15. Convergence Curve of Dual Channel Example

Additional Analysis of Results and Comments

The results obtained from the two examples clearly indicate that the CIP can be a valuable design aid. It must be pointed out that as the name, Compensator Improvement Program, imples the program is a design aid, not a design technique. That is, the program does not decide the order, the type, or the number of compensators necessary. All of this requires good engineering judgement before the running of the program is attempted.

As the two examples exemplified the solution cannot be worse than the original compensator if the specifications on the input are made properly. In regard to stability margins and attenuation margins this simply requires pushing and pulling these, respectively, with respect to the $-1 + j0$ and $0 + j0$ points. By doing this, these can always be bettered, except when they proceed from activity to inactivity. However, the amount of slippage in going from inactivity to activity can be minimized by choosing a reasonable maximum step size such as 0.1 or less of the smallest compensator coefficient. As long as a margin stays in a vicinity of the desired value it is acceptable.

The specifications for insuring the improvement in gain and phase margins are not always as simple as those for stability margins and attenuation margins. In fact, in many instances it is necessary to push and pull these with respect to two points in the complex plane. This is especially true if the acute angle between the tangent to the $GH(j\omega)$ frequency response where these occur and either the tangent to the unit circle or real azis is very small. Both of these cases are illustrated in Figure 16 where tangents to some hypothetical
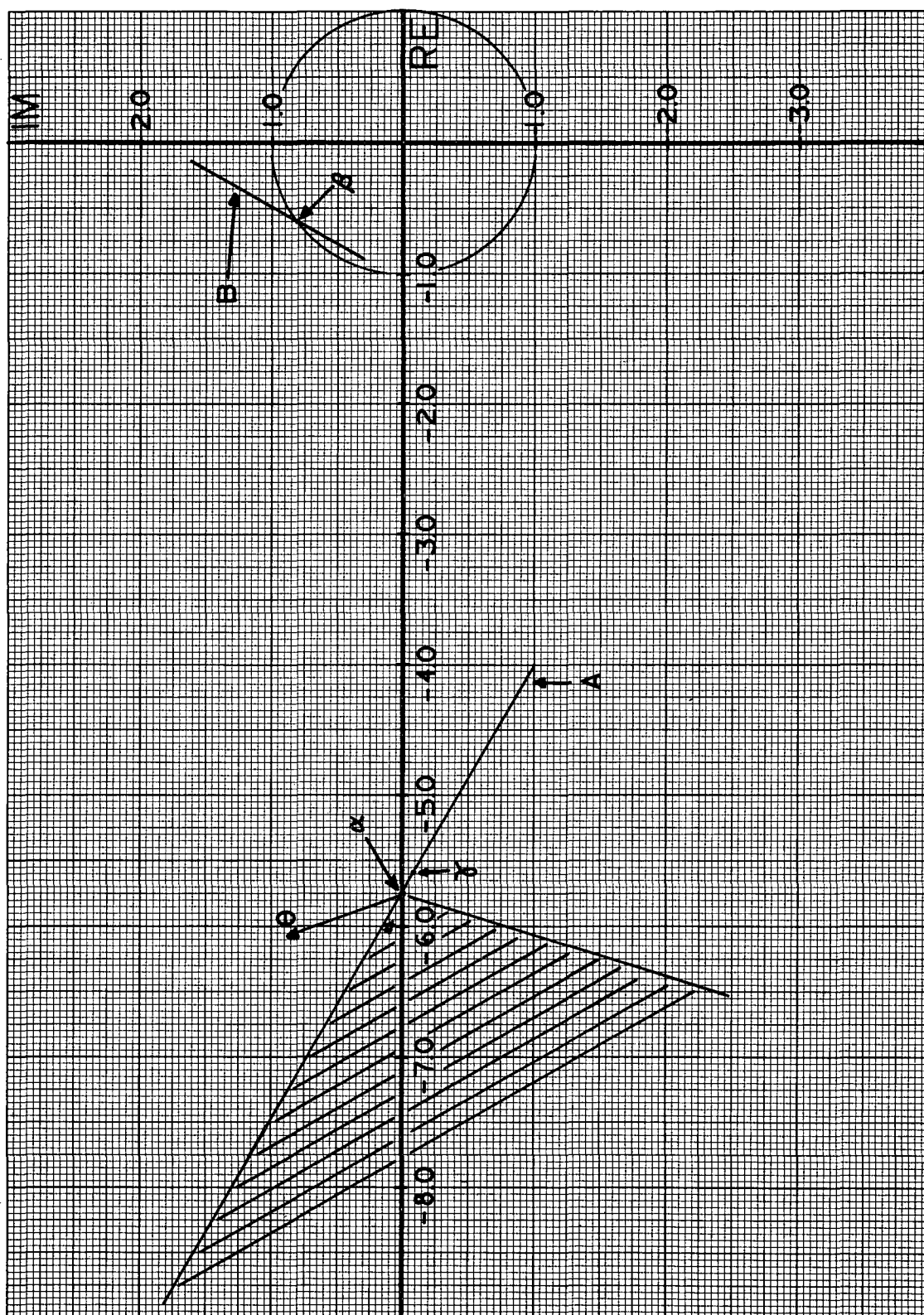
Figure 16. Graph for Showing Certain Programming
Considerations

GH(jω) frequency response are assumed as A and B. The points α and β are the points where the margins occur. If they are perturbed so that the distances between them and the -1 + j0 point are increased, then they are allowed to move in any direction which has a positive dot product with vectors emanating from the -1 + j0 point to these points. Suppose that α was perturbed in the direction θ indicated in Figure 16. It is obvious that by moving α in this direction the vector from -1 + j0 to α is increasing in magnitude. However, after α is perturbed it is no longer the point of interest. Some other point such as λ is then the point under consideration, where λ is in some neighborhood of α. From practical considerations it is known that if α moves in the direction θ then a small neighborhood around α will move in the direction θ. Let λ be in this neighborhood. The result is that λ will be the new point of intersection with the real axis, and, furthermore, its distance from the -1 + j0 point is less than what α's was. Similar results can be demonstrated for β.

These types of problems can be circumvented by perturbing a point with respect to two points in the complex plane. In fact consider the example in the last paragraph. Suppose that α is not only pushed with respect to the -1 + j0 point, but it is also pulled with respect to the -7 - j4 point. The permissible region for the movement of α now becomes the intersection of the permissible region for pushing from the -1 + j0 point and the permissible region for pulling with respect to the -7 - j4 point. The result is the cross-hatched area in Figure 16. Movement of α anywhere in this region cannot result in the gain margin being decreased.

For the single channel example conditions did not exist to warrant pertubations with respect to more than one point. On the other hand the dual channel example required perturbing the aerodynamical gain margin with respect to two points. Runs in which this was not done resulted in a significant reduction in this margin.

In neither example did the lead and lag phase margins of the first bending mode become active. In the single channel example, conditions just never prevailed. As for the dual channel example, conditions would have probably resulted if the magnitude of the first bending mode had not been controlled by the attenuation margin technique. Since the frequencies where these margins occur are very close to the frequency of the first bending mode, then it is quite natural that an increase in the first bending mode magnitude would have resulted in the reduction of at least one of these margins.

The program indicated for the dual channel example that better results could be obtained with one less zero in the numerator of the first channel's compensator and one less pole in the second channel's compensator. It did this by driving these to infinity. It also drove two poles in each channel to equal values. This probably indicates that if these poles were included in second order factors they would split into complex conjugates. However, the first order pole factors were chosen so that complex poles would not be allowed.

One other fact which should be pointed out is that the program was used in the SIFR mode. However, because of the maximum step size choices (0.1 for the first 1000 iterations of the first example and

0.2 for the second 1000 iterations and 0.1 for the second example) the program actually performed in the TIFR mode.*

One phenomenon which should not pass without mention is the apparent unsmoothness of the convergence curves, Figures 7 and 15. In actuality, these curves should be discrete curves. For convenience they were drawn continuously. The sharp, abrupt changes, where the smallest stability margins make much greater gains than on other iterations, occur at iterations where the aerodynamical gain margin became inactive. This allowed the smallest stability margin to make a marked gain for one iteration. While this was occurring the aerodynamical gain margin was returning to activity. Once it became active again the rate of increase of the smallest stability margin decreased. On higher iterations the curve was smooth until the aerodynamical gain margin went inactive again, at which time the process was repeated. The overall effect of the program is a "ratchet" type, i. e., once a margin is increased, it will not decrease.

---

*Of course again this is neglecting instances where constraints went from inactivity to activity.

RD

CHAPTER VIII

CONCLUSION, LIMITATIONS, AND SUGGESTED

FUTURE STUDIES

Summary

In this dissertation, the theory for a compensator improvement
algorithm has been presented. The goal from the onset was to accom-
plish this by way of mathematical programming. Thus, in Chapter I
a concise review of the more popular mathematical programming tech-
niques was given. After this review a discussion of the uses of
mathematical programming in the design of control systems was pre-
sented. Also, a discussion of the uses of mathematical programming
in the design of control systems was made. In this discussion it was
pointed out that only a small amount of effort has been devoted to
using mathematical programming as an aid in the design of control
systems by classical means. Furthermore, it was shown that the tech-
niques which have been developed suffer from some serious drawbacks.
Thus, the thesis of this dissertation was to develop a computerized
compensator design procedure which circumvented these pitfalls.

In Chapter II, some important concepts for the measuring of
expected performance of a control system were given. This involved
defining relative stability in a way somewhat different from the
normal textbook definition. Also, concepts of relative attenuation
and proper phasing were defined. Finally, using these the design of
a compensator was formulated as a mathematical programming problem--
which in the end resulted in a strict constraint problem.

In Chapter III compensator limitations for two possible iterative techniques for solving the problem formulated in Chapter II were presented by the proving of two theorems. The first theorem showed that to be assured of being able to perturb n points in the $GH(j\omega)$ plane in n optimal directions there must exist 2n coefficients for variance. On the other hand, Theorem 2 stated that if each point was given 180° of freedom for movement (a sub-optimal direction), then only n coefficients were needed for variance. From this it was deduced that a sub-optimal algorithm would be the most practical.

Then, in Chapter IV the development of a sub-optimal algorithm was made. The result was the evolvement of the constraint improvement algorithm. In this development several definitions were given, e.g., total improved frequency response, sum improved frequency response, improved solution, and active and passive constraints.

In order to employ the constraint improvement algorithm in Chapter IV, it was expedient to have the gradients of the active constraints. These were found in Chapter V for a general j^{th} channel control system. Furthermore, the partials were derived so that pushing or pulling on points of the frequency response could be accomplished with respect to any points desired in the complex $GH(j\omega)$ plane.

Next, the ideas and material in Chapters II, III, IV, and V were included in a computer program called CIP (Compensator Improvement Program). In Chapter VI the general iterating procedure of this program was incorporated. In addition, several special programming techniques employed by CIP were presented in this chapter.

Chapter VII was used to demonstrate the practicality of CIP. This was illustrated by two large system examples. These examples clearly showed the program's capability of handling single or multi-channel control systems. A significant amount of improvement in the frequency response of both systems was seen after an application of CIP. Also, curves to show the convergence properties of CIP were given. In addition, several comments in regard to proper specifications for the program were mentioned.

Limitations and Concluding Remarks

One of the limitations of CIP is that the initial compensator must be chosen to stabilize the system. This is the reason that the program was termed an "improvement program" rather than a design program. A major goal of the program is to improve stability margins, etc., from one iteration to another. Obviously, if the system is initially unstable then stability margins have no meaning.

Another shortcoming of CIP is that a choice of the components of the c vector in Chapter IV must be made. If the strict constraint problem has a solution which is reachable from the initial starting point, the choice of the c vector has little consequence other than to affect the rate of convergence. However, if the problem does not have an obtainable solution, then the choice of this vector will definitely determine the relative extremal where convergence occurs. Nevertheless, it should be pointed out that if the initial guess at the solution is not a relative extremal then the solution at convergence will be better than the initial solution.

A very good property which CIP possesses is an inherent ability not to design an unstable compensator, provided the step size is maintained at a reasonable value. The reason for this is that CIP continuously improves relative stability; thus the stability of the system cannot decrease.

Although CIP requires a choice of the c vector elements, it still has the capabilities of yielding a practical design on every run. As long as the input specifications of the program are properly made, the program cannot yield a compensator worse than the original compensator. CIP is not a design technique, but it is a design aid.

Suggested Future Studies

There are several areas in which the work in this dissertation can be extended. One such study could involve using the constraint improvement algorithm in other design problems in engineering and science. This author does not see any reason that it could not be used to make improvements in any design where the number of variables is greater than the number of constraints to be controlled and where the gradient vectors of the constraints are deterministic.

Also, it is foreseen by this author that the constraint improvement algorithm could be the basis of a new or extended gradient algorithm for nonlinear programming. For example, if any of the elements of the c vector are set to zero then the determined directional vector will lie in the tangent planes of the constraints corresponding to the c's with zero value. Of course this would be similar to the gradient projection technique mentioned in Chapter I. However, it is deemed by this author that by using the constraint improvement approach an

optimal gradient projection algorithm can be developed. Up to the present such an algorithm has not been developed.

In regard to future studies in compensator design, there would be nothing wrong with starting with physical electrical networks, rather than transfer functions. If a program started with a network and varied the elements for making the improvements described previously, the end results would be the actual network needed. The practicality of this network would depend upon the constraints placed on the network elements.

A compensator design procedure could be devised using the constraint improvement algorithm on the Routh-Hurwitz array. By forming the characteristic equation as a function of the compensator coefficients, the first two rows of the Routh array can be formulated as functions of these compensator coefficients. Since it is known how the other rows of the array are formed from the first two rows, the changes in the elements of the first column of the array with respect to the compensator coefficients could be determined by an application of the chain rule for partial derivatives. Then, the constraint improvement algorithm could be used to drive all the negative elements of the first column positive, as long as the number of negative elements did not exceed the number of compensator coefficients. If all the elements are driven positive then a certain amount of relative stability could be achieved by evaluating the characteristic equation at $(s + a)$ where a is a positive real number; the previously mentioned procedure can now be applied to the new characteristic equation. If in this application all elements of the first column could again be driven to positive values, then it would be known that no pole of the closed loop system has a

real part greater than - a. This process could be repeated until a

desired value of a is achieved or until all the elements of the first

column of one of the characteristic equations cannot be driven

positive.

APPENDIX

COMPENSATOR IMPROVEMENT PROGRAM

In the following is a complete Fortran version of the Compensator Improvement Program. The program is completely self-contained, i.e., it does not require any system library, etc. The necessary input to the program is explained in the comment statements at the beginning of the main program. Furthermore, all inputs except the frequency response points are printed out with explanations of the input specifications. The other output is, also, explained by certain comments printed out with the information.

```
C
C MAIN PROGRAM
C
C DEFINITIONS OF I/O VARIABLES
C KCHNL        -NO. OF CHANNELS FED BACK
C NUMC(I)      -NO. OF COMPENSATORS IN I-TH CHANNEL
C NRATOR(I,J) -NUMERATOR ORDER OF J-TH COMPENSATOR IN THE I-TH CHANNEL
C NDENOM(I,J) -DENOMINATOR ORDER OF J-TH COMP. IN I-TH CHANNEL
C XCOMN(I,J)  -NUMERATOR COEFFICIENTS OF J-TH COMP. IN I-TH CHNL.
C YCOMN(I,J)  -DENOM. COEFFICIENTS OF J-TH COMP. IN I-TH CHNL.
C OMEGA(I)     -I-TH FREQ.(ASSUMED TO BE IN HZ.)
C GRA(I,J)     -J-TH REAL PART OF OPEN LOOP FREQ. RESP. OF I-TH CHNL.
C GIA(I,J)     -J-TH IMAG. PART OF OPEN LOOP FREQ. RESP. OF I-TH CHNL.
C
C      KSTART -STARTING ITERATION NO.
C      KQUIT  -STOPPING ITERATION NO.
C      KPOINT -NO. OF POINTS FROM OPEN LOOP FREQ. RESPONSE USED
C      KPRINT    - NO. OF ITERATIONS SKIPPED BETWEEN PRINTING OF INFOR.
C      STPMAX -MAXIMUM CHANGE TO BE MADE IN COMPENSATOR COEFFICIENTS
C              ON ANY ONE ITERATION(PROBABLY NO MORE THAN 30% OF THE
C              SMALLEST COMPENSATOR COEFFICIENT OF THE INITIAL
C              COMPENSATOR)
C      STPMIN    - MINIMUM STEP SIZE DESIGNATOR
C      F10 & F11 - FREQUENCIES BETWEEN WHICH G.M.'S ARE FOUND
C      F12 & F13 - FREQUENCIES BETWEEN WHICH P.M.'S ARE FOUND
C      FMIN      - A.M.'S ARE FOUND FOR FREQS. ABOVE THIS FREQ.
C      VARIABLES FOR GAIN MARGIN RADII DESIGNATIONS
C      IF FREQ. .LE. F1              DESIRED MARGIN = R1
C      IF FREQ. .GT. F1 BUT .LT. F2  DESIRED MARGIN = R2
C      IF FREQ. .GE. F2             DESIRED MARGIN = R3
C
C      VARIABLES FOR PHASE MARGIN RADII DESIGNATIONS
C      IF FREQ. .LE. F3              DESIRED MARGIN = R4
C      IF FREQ. .GT. F3 BUT .LT. F4  DESIRED MARGIN = R5
C      IF FREQ. .GE. F4             DESIRED MARGIN = R6
C
C      VARIABLES FOR STABILITY MARGIN RADII DESIGNATIONS
C      IF FREQ. .LE. F5              DESIRED MARGIN = R7
C      IF FREQ. .GT. F5 BUT .LT. F6  DESIRED MARGIN = R8
C      IF FREQ. .GE. F6             DESIRED MARGIN = R9
C
C      VARIABLES FOR ATTENUATION MARGIN RADII DESIGNATIONS
C      IF FREQ. .LE. F7              DESIRED MARGIN = R10
C      IF FREQ. .GT. F7 BUT .LT. F8  DESIRED MARGIN = R11
C      IF FREQ. .GE. F8             DESIRED MARGIN = R12
C      GAIN(I)-DENOTES INITIAL D. C. GAIN VALUE FOR I-TH CHANNEL
C      KNR(I) -NUMBER OF NUMERATOR COEFS. FOR I-TH CHANNEL
C      KDR(I) -NUMBER OF DENOM. COEFS. IN I-TH CHANNEL
C      KONT(I)-D.C. DESIGNATOR FOR I-TH CHANNEL
C              KONT(I)=1       GAIN ALLOWED TO VARY
```

```
C                 KONT(I)=2        GAIN NOT ALLOWED TO VARY
C       KIFM    -NO. CHANNELS THAT FREQ. RESP. INFIRMATION IS TO BE READ IN
C       PPT(I)  -POINTS THAT THE CRITICAL FREQUENCIES WILL BE
C                 PERTURBED WITH RESPECT TO (COMPLEX POINTS)
C                 I=1  GAIN MARGIN POINT
C                 I=2  PHASE MARGIN POINT
C                 I=3  STABILITY MARGIN POINT
C                 I=4  ATTENUATION MARGIN POINT
C       LSN(I)   - DENOTES WHETHER POINTS ARE TO BE PUSHED OR PULLED
C                 LSN=-1  POINT TO BE PULLED
C                 LSN=+1  POINT TO BE PUSHED
C       INCGMS  -INDICATES WHETHER G.M.'S ARE TO BE ARTIFICALLY
C                 INCLUDED AS S.M.'S
C                 INCGMS=0       NOT INCLUDED
C                 INCGMS=1       INCLUDED
C       INCPMS  -INDICATES WHETHER P.M.'S ARE TO BE ARTIFICALLY
C                 INCLUDEED AS S.M.'S
C                 INCPMS=0       NO INCLUDED
C                 INCPMS=1       INCLUDED
C SOME INTERIOR VARIABLE DEFINITIONS
C
C GCR(I,J)      -REAL PART OF COMPENSATOR FREQ. RESP. AT SOME ITERATION
C GCI(I,J)      -IMAG. PART OF COMPENSATOR FREQ. RESP. AT SOME ITERATION
C GCOMR(I,J)    -REAL PARTS OF I-TH CHNL. OPEN LOOP FREQ. RESP.
C GCOMR(I,J)    -IMAG. PARTS OF I-TH CHNL. OPEN LOOP FREQ. RESP.
C GR(I)         -REAL PARTS OF TOTAL OPEN LOOP FREQ. RESP.
C GI(I)         -IMAG. PARTS OF TOTAL OPEN LOOP FREQ. RESP.
C
C
C
C       ***** THERE ARE 13 READ STATEMENTS *****
        DIMENSION XCOMN(10,50),YCOMN(10,50),PRY(50),PRX(50),STBM(99),
     1    PX(50),PY(50),RQ(99),GR(999),GI(999),OMEGA(999),GRA(5,999),
     2    GIA(5,999),G(20,99),DV(50),WEIGHT(50),BCOMN(10,50),
     3    BCOMD(10,50),GCR(5,999),GCI(5,999),GCOMR(5,999),
     4    GCOMI(5,999), NUMC(20),NRATOR(10,20),NDENOM(10,20),CNUM(10),
     5    CDOM(10),KNR(10),KDR(10),COTN(10,50),COTD(10,50)
        DIMENSION KACT(99),SML(99)
        DOUBLE PRECISION G,DV,WEIGHT
        DIMENSION KONT(20), KPTS(99), GAIN(10)
        DIMENSION TYPE(99)
        DIMENSION PPT(4), LSN(4)
        COMMON TYPE
        INTEGER TYPE
        COMPLEX PPT
        READ(5,5) KCHNL
        READ(5,5) (KONT(I),I=1,KCHNL)
        READ(5,5)(NUMC(I),I=1,KCHNL)
        WRITE(6,1) KCHNL
      1 FORMAT('0',5X,'NUMBER OF CHANNELS FEDBACK=',I5)
        WRITE(6,3) (KONT(I),I=1,KCHNL)
      3 FORMAT('0',5X,'D.C. GAIN CONSTRAINT DESIGNATOR FOR EACH CHANNEL (
```

```
     1KONT=1, ALLOWED TO VARY; KONT=2, HELD CONSTANT )'/6X,8(I2,10X))
       WRITE(6,4) (NUMC(I),I=1,KCHNL)
     4 FORMAT('0',5X,'COMPENSATORS PER CHANNEL',10I5)
       DO 2 I=1,KCHNL
       KNAT=NUMC(I)
       READ(5,5) (NRATOR(I,J),J=1,KNAT)
       WRITE(6,6) I,(NRATOR(I,J),J=1,KNAT)
     6 FORMAT('0',5X,'CHANNEL NO.',I2,2X,'NUMERATOR ORDERS',2X,10I5)
       READ(5,5) (NDENOM(I,J),J=1,KNAT)
       WRITE(6,7) I,(NDENOM(I,J),J=1,KNAT)
     7 FORMAT(' ',5X,'CHANNEL NO.',I2,2X,'DENOMINATOR ORDERS',10I5)
     2 CONTINUE
     5 FORMAT(16I5)
       READ(5,10) KSTART,KQUIT,KIFM,KPOINT,KPRINT, R1,F1,R2,F2,R3,
      1    R4,F3,R5,F4,R6, R7,F5,R8,F6,R9, R10,F7,R11,F8,R12, FMIN,F10,
      2    F11,F12,F13,STPMAX,STPMIN
    10 FORMAT(5I5/5F10.5/5F10.5/5F10.5/5F10.5/8F10.5)
       WRITE(6,11)KSTART,KQUIT,KIFM,KPOINT,KPRINT
    11 FORMAT('0',1X,'START ITER.=',I5,2X,'STOP ITER.=',I5,2X,'NO. CHNL.
      1FREQ. RESP. IN=',I5,2X,'NO. OF FREQ. POINTS=',I5,2X,'PRINT INCREME
      2NT=',I5)
       WRITE(6,25)STPMAX,STPMIN
    25 FORMAT('0',5X,'MAXIMUM DESIGNATED STEP SIZE =',F10.5/6X,'MINIMUM D
      1ESIGNATED STEP SIZE =',F10.5)
       WRITE(6,12)
    12 FORMAT('0'5X,'DESIRED GAIN MARGIN RADII DESIGNATIONS')
       WRITE(6,13) F1,R1, F1,F2,R2, F2,R3
    13 FORMAT('0',5X,'IF FREQUENCY .LE.',F10.5,5X,'DESIRED MARGIN IS',
      1 F10.5/6X,'IF FREQUENCY .GT.',F10.5,2X,'BUT .LT.',F10.5,2X,'DESIRE
      2D MARGIN IS',F10.5/6X,'IF FREQUENCY .GT.',F10.5,2X,'DESIRED MARGI
      3N IS',F10.5)
       WRITE(6,17) F10,F11
    17 FORMAT(' ',5X,'GAIN MARGINS ARE DETERMINED BETWEEN THE FREQUENCIES
      1 OF',F10.5,2X,'AND',F10.5)
       WRITE(6,14)
    14 FORMAT('0',5X,'DESIRED PHASE MARGIN RADII DESIGNATIONS')
       WRITE(6,13) F3,R4, F3,F4,R3, F4,R6
       WRITE(6,18) F12,F13
    18 FORMAT(' ',5X,'PHASE MARGINS ARE DETERMINED BETWEEN THE FREQUENCIE
      1S OF',F10.5,2X,'AND',F10.5)
       WRITE(6,15)
    15 FORMAT('0',5X,'DESIRED STABILITY MARGIN RADII DESIGNATIONS')
       WRITE(6,13) F5,R7, F5,F6,R8, F6,R9
       WRITE(6,16)
    16 FORMAT('0',5X,'DESIRED ATTENUATION MARGIN RADII DESIGNATIONS')
       WRITE(6,13) F7,R10, F7,F8,R11, F8,R12
       WRITE(6,19) FMIN
    19 FORMAT(' ',5X,'ATTENUATION MARGINS ARE FOUND FOR FREQS. ABOVE',
      1    F10.5)
       READ(5,50) (GAIN(I),I=1,KCHNL)
       WRITE(6,20) (I,GAIN(I),I=1,KCHNL)
    20 FORMAT('0',5X,2('CHANNEL NO.',I3,1X,'INITIAL D.C. GAIN IS',F10.5,
```

```
  1    5X))
     READ(5,50) (PPT(I),I=1,4)
     WRITE(6,22) (PPT(I),I=1,4)
 22 FORMAT('0',5X,'PERTUBATION POINTS FOR GAIN, PHASE, STABILITY, AND
    1ATTENUATION MARGINS, RESPECTIVELY:'/6X,4('REAL',F6.2,2X,'IMAG.',
    2   F6.2,2X))
     READ(5,5) (LSN(I),I=1,4)
     WRITE(6,23) (LSN(I),I=1,4)
 23 FORMAT('0',5X,'DENOTING WHETHER EACH OF THE PRECEDING POINTS ARE '
    1    ,'TO BE PUSHING OR PULLING POINTS(PUSHING=+1, PULLING=-1)' /6X,
    2    4(I2,10X))
     READ(5,5) INCGMS,INCPMS
     WRITE(6,24) INCGMS,INCPMS
 24 FORMAT('0',5X,'DENOTING WHETHER GAIN OR PHASE MARGINS ARE ARTIFICA
    1LLY INCLUDED AS STABILITY MARGINS(NOT INCLUDED=0, INCLUDED=1)'/6X,
    2    2(I2,10X))
     KVARY=0
     DO 21 K=1,KCHNL
     LAMP=NUMC(K)
     KNR(K)=0
     KDR(K)=0
     DO 21 I=1,LAMP
     KVARY=KVARY+NRATOR(K,I)
     KVARY=KVARY+NDENOM(K,I)
     KNR(K)= KNR(K) + NRATOR(K,I) + 1
 21 KDR(K)= KDR(K) + NDENOM(K,I) + 1
     DO 29 I=1,KCHNL
 29 IF(KONT(I).EQ.1)KVARY=KVARY+1
     DO 42 K=1,KCHNL
     LNC= KNR(K)
     LDC= KDR(K)
     READ(5,50)(XCOMN(K,I),I=1,LNC)
 42 READ(5,50)(YCOMN(K,I),I=1,LDC)
 50 FORMAT(8F10.5)
 60 FORMAT('0',6X,'INITIAL COMPENSATOR COEFFICIENTS')
     WRITE(6,60)
     DO 72 K=1,KCHNL
     LNC=KNR(K)
     LDC= KDR(K)
     WRITE(6,62)K
 62 FORMAT('0',5X,'CHANNEL NO.',I2,2X,'COMPENSATORS - FACTORED FORM')
     WRITE(6,68)
 68 FORMAT('0',5X,'NUMERATOR COEFFICIENTS')
     WRITE(6,70) (XCOMN(K,I),I=1,LNC)
     WRITE(6,69)
 69 FORMAT('0',5X,'DENOMINATORS COEFFICIENTS')
 72 WRITE(6,70) (YCOMN(K,I),I=1,LDC)
 70 FORMAT(' ',5X,10F10.5)
C MODIFICATION OF FREQ. RESP. INFOR. BY CONTANT COMPENSATOR
     DO 135 J=1,KIFM
135 READ(5,140) (OMEGA(I),GRA(J,I),GIA(J,I),I=1,KPOINT)
140 FORMAT(9F8.5)
```

```
      IF(KIFM.GE.KCHNL)GO TO 150
      K= KIFM + 1
      DO 148 J=K,KCHNL
      DO 148 I=1,KPOINT
      XV= OMEGA(I) * 6.2831853
      GRA(J,I)= -OMEGA(I) * XV * GIA(J-1,I)
  148 GIA(J,I)= OMEGA(I) * XV * GRA(J-1,I)
  150 CONTINUE
      DO 149 J=1,KCHNL
      DO 149 I=1,KPOINT
      GRA(J,I)= GRA(J,I) * GAIN(J)
  149 GIA(J,I)= GIA(J,I) * GAIN(J)
  190 CONTINUE
      DATA STEP,KHOP,SML2,PSQL,SBC2/1.0E-02,0,0.0,1.0E+20,0.0/
      I1=0
      I2=0
      DO 195 K=1,KCHNL
      I1=   KNR(K) + I1
  195 I2=   KDR(K) + I2
      LOX= KSTART
  200 CONTINUE
      LPRESV=KVARY
      NM=0
C EVALUATION OF VARIABLE COMPENSATOR AT CHOSEN FREQS.
      DO 210 K=1,KPOINT
      GR(K)=0.0
      GI(K)=0.0
      XV=OMEGA(K) *6.2831853
      DO 209 I=1,KCHNL
      KCOMP= NUMC(I)
      LNOT=0
      KNOT=0
      GCR(I,K)= 1.0
      GCI(I,K)= 0.0
      DO 208 J=1,KCOMP
      NTR= NRATOR(I,J)+1
      NTD= NDENOM(I,J)+1
      DO 204 M=1,NTR
  204 CNUM(M)= XCOMN(I,M+KNOT)
      DO 205 M=1,NTD
  205 CDOM(M)= YCOMN(I,M+LNOT)
      KNOT= KNOT + NTR
      LNOT= LNOT + NTD
      K2= NTR-1
      K3= NTD-1
      CALL POLFV(CNUM,K2,XV,CNR,CNI)
      CALL POLFV(CDOM,K3,XV,CDR,CDI)
      CD= CDR**2 + CDI**2
      ACR=GCR(I,K)
      ACI= GCI(I,K)
      ACOMR= (CNR * CDR + CNI * CDI)/CD
      ACOMI=(-CNR * CDI + CNI * CDR)/CD
```

```
      GCR(I,K)= ACR * ACOMR - ACI * ACOMI
208 GCI(I,K)= ACR * ACOMI + ACI * ACOMR
      GCOMR(I,K)= GRA(I,K)*GCR(I,K) - GIA(I,K)*GCI(I,K)
      GCOMI(I,K)= GRA(I,K)*GCI(I,K) + GIA(I,K)*GCR(I,K)
      GR(K)= GR(K) + GCOMR(I,K)
209 GI(K)= GI(K) + GCOMI(I,K)
210 CONTINUE
C DETERMINATION OF GAIN MARGINS POINTS BETWEEN F1 AND F2
      CALL GAINMG(GR,GI,KPOINT,NM,F10,F11,KPTS,STBM,OMEGA)
      NGMS=NM
C SETTING DESIRED STABILITY RADII OF G.M.'S
      KPM=NM+1
      IF(NM.EQ.0)GO TO 213
      DO 212 I=1,NM
      TYPE(I)= 'G'
      KWHICH=KPTS(I)
      FREHZ=OMEGA(KWHICH)
      IF(FREHZ.LE.F1)RQ(I)=R1
      IF(FREHZ.GT.F1)RQ(I)=R2
      IF(FREHZ.GE.F2)RQ(I)=R3
212 CONTINUE
213 CONTINUE
C DETERMINATION OF P.M. BETWEEN F3 AND F4
      CALL PHASEM(GR,GI,KPOINT,NM,F12,F13,KPTS,STBM,OMEGA)
      IF(NM.LT.KPM)GO TO 215
C SETTING DESIRED STABILITY RADII OF P.M.'S
      DO 214 I=KPM,NM
      TYPE(I)= 'P'
      KWHICH=KPTS(I)
      FREHZ=OMEGA(KWHICH)
      IF(FREHZ.LE.F3)RQ(I)=R4
      IF(FREHZ.GT.F3)RQ(I)=R5
      IF(FREHZ.GE.F4)RQ(I)=R6
214 CONTINUE
      KPM=NM+1
215 CONTINUE
      IF(NM.EQ.0)GO TO 221
      KLAST=NM
      DO 220 I=1,KLAST
      IF((I.LE.NGMS).AND.(INCGMS.EQ.1))GO TO 219
      IF((I.GT.NGMS).AND.(INCPMS.EQ.1))GO TO 219
      GO TO 220
219 KPM=KPM+1
      NM=NM+1
      KPTS(NM)=KPTS(I)
      STBM(NM)=STBM(I)
      RQ(NM)=RQ(I)
      TYPE(NM)='S'
220 CONTINUE
221 CONTINUE
      KSTBM=KPM
      RPT=1.0
```

```
      NSG=1
      FQMIN=0.0
C DETERMINATION OF STABILITY MARGINS
      CALL SRMINS(GR,GI,KPOINT,NM,RPT,NSG,FQMIN,KPTS,STBM,OMEGA)
C SETTING DESIRED STABILITY MARGINS
      IF(NM.LT.KPM)GO TO 216
      DO 230 I=KPM,NM
      TYPE(I)= 'S'
      KWHICH=KPTS(I)
      FREHZ=OMEGA(KWHICH)
      IF(FREHZ.LE.F5)RQ(I)=R7
      IF(FREHZ.GT.F5)RQ(I)=R8
      IF(FREHZ.GE.F6)RQ(I)=R9
  230 CONTINUE
C CHECKING TO SEE IF ANY P.M.'S, G.M.'S, OR S.M.'S ARE EQUAL
C IF THERE RESULTS SOME THAT ARE EQUAL ONLY THE FIRST IS RETAINED.
      DO 228 LB=2,KSTBM
      DO 228 I=KSTBM,NM
      IF(KPTS(LB-1).NE.KPTS(I))GO TO 228
      NM=NM-1
      DO 226 L=I,NM
      KPTS(L)= KPTS(L+1)
      STBM(L)= STBM(L+1)
      RQ(L)= RQ(L+1)
  226 TYPE(L)= TYPE(L+1)
  228 CONTINUE
      KPM=NM+1
  216 CONTINUE
      KMIN=NM
      RPT=0.0
      NSG=-1
      FQMIN=FMIN
C DETERMINATION OF ATTENUATION MARGINS
      CALL SRMINS(GR,GI,KPOINT,NM,RPT,NSG,FQMIN,KPTS,STBM,OMEGA)
C SETTING DESIRED ATTEN. MARGINS
      IF(NM.LT.KPM)GO TO 217
      DO 232 I=KPM,NM
      TYPE(I)= 'A'
      KWHICH=KPTS(I)
      FREHZ=OMEGA(KWHICH)
      IF(FREHZ.LE.F7)RQ(I)=R10
      IF(FREHZ.GT.F7)RQ(I)=R11
      IF(FREHZ.GE.F8)RQ(I)=R12
  232 CONTINUE
  217 CONTINUE
      SBC1=R1
C DETERMINING SMALLEST STABILITY MARGINS OF PRESENT ITER. AND ALL ITER.
      SML1= 100.0
      DO 290 I=1,KMIN
      IF(STBM(I).GT.SML1)GO TO 288
      SML1= STBM(I)
  288 CONTINUE
```

```
      IF(STBM(I).GT.SBC1)GO TO 290
      SBC1= STBM(I)
  290 CONTINUE
      IF(SBC2.GE.SBC1)GO TO 298
      SBC2= SBC1
      IBEST= LOX
      DO 292 K=1,KCHNL
      LNC= KNR(K)
      LDC= KDR(K)
      DO 291 I=1,LNC
  291 BCOMN(K,I)= XCOMN(K,I)
      DO 292 I=1,LDC
  292 BCOMD(K,I)= YCOMN(K,I)
  298 CONTINUE
C CHECKING SATISFACTION OF SYSTEM REQUIREMENTS
      DO 320 I=1,NM
      PORM= 1.0
      IF(I.GT.KMIN)PORM=-1.0
  310 IF((STBM(I)-RQ(I))*PORM)350,320,320
  320 CONTINUE
      WRITE(6,330)
  330 FORMAT('0',15X,'***** ALL SYSTEM REQUIREMENTS HAVE BEEN MET *****'
     1)
  340 CONTINUE
      CALL OTPT1(STBM,OMEGA,KPTS,NM,XCOMN,YCOMN,KMIN,RQ,LOX,KCHNL,NUMC,
     1    NRATOR,NDENOM,PRX,PRY,I1,I2)
      WRITE(6,341) IBEST
  341 FORMAT('0',5X,'***** BEST COMPENSATORS WITH RESPECT TO STABILITY *
     1****'//6X,'OCCURRED ON STEP',I4,2X,'AND THEIR COEFFICIENTS ARE:')
      CALL MULOUT(KCHNL,NUMC,NRATOR,NDENOM,KNR,KDR,BCOMN,BCOMD)
      WRITE(6,345) SBC2
  345 FORMAT('0',21X,'SMALLEST STABILITY MARGIN FOR THE BEST COMPENSATOR
     / =',F10.8)
      WRITE(6,347)
  347 FORMAT('0',5X,'***** COMPENSATORS AND COMPENSATED FREQUENCY RESPON
     1SE AT THE LAST ITERATION PERFORMED ARE AS FOLLOWS *****')
      CALL MULOUT(KCHNL,NUMC,NRATOR,NDENOM,KNR,KDR,XCOMN,YCOMN)
      WRITE(6,346)
  346 FORMAT('0',9X,'COMPENSATED FREQUENCY RESPONSE'//10X,'FREQUENCY',
     1    2X,'MAGNITUDE',3X,'ANGLE')
      DO 349 I=1,KPOINT
      GMTE= SQRT(GR(I)**2 + GI(I)**2)
      AGLE= ATAN2(GI(I),GR(I))*57.3
      WRITE(6,348) OMEGA(I),GMTE,AGLE
  348 FORMAT(' ',7X,F10.5,1X,F10.5,1X,F10.5)
  349 CONTINUE
      STOP
  350 CONTINUE
C STEP SIZE SELECTING
      IF(LOX.EQ.KQUIT)WRITE(6,351)
  351 FORMAT('0',5X,'***** TERMINATION REASON - MAXIMUM ITERATIONS *****
     1')
```

```
      IF(LOX.EQ.KQUIT)WRITE(6,400)STEP
      IF(LOX.EQ.KQUIT)GO TO 340
      IF(LOX.EQ.KSTART)GO TO 354
      ADD=0.0
      MAD=0
      PORM=1.0
      DO 355 I=1,NM
      IF(I.GT.KMIN)PORM=-1.0
      IF(PORM*(STBM(I)-RQ(I)).GE.0.0)GO TO 355
      DO 352 J=1,NML
  352 IF(KPTS(I).EQ.KACT(J))GO TO 353
      MAD=MAD+1
      GO TO 355
  353 CONTINUE
C IF IT IS DESIRED TO HAVE ALL CONSTRAINTS TO BE IMPROVED AT EVERY
C ITERATION REMOVE THE C FROM COLUMN 1 OF THE FOLLOWING CARD
C     IF(PORM*(STBM(I)-SML(I)).LT.-1.0E-05)GO TO 360
      ADD=ADD+PORM*(STBM(I)-SML(J)).
  355 CONTINUE
      IF(MAD.EQ.NML)ADD=1.0
      IF(ADD.LE.0.0)GO TO 360
  354 CONTINUE
      GO TO 371
  360 STEP= STEP/2.0
      IF(STEP.LT.STPMIN )WRITE(6,365)STPMIN
  365 FORMAT('0',5X,'***** TERMINATION REASON - STEP SIZE IS LESS THAN '
     1 ,F10.5,2X,'*****')
      IF(STEP.LT.STPMIN )GO TO 340
      LOX= LOX - 1
      GO TO 450
  371 STEP=1.41416 * STEP
  373 CONTINUE
      SML2=SML1
      IF(STEP.GT.STPMAX)STEP= STPMAX
C OUPUT CONTROL
      IF(KHOP.GT.1)GO TO 410
      KHOP=KPRINT
      WRITE(6,400) STEP
  400 FORMAT('0', 15X, 'PRESENT STEP SIZE =',F10.7)
      CALL OTPT1(STBM,OMEGA,KPTS,NM,XCOMN,YCOMN,KMIN,RQ,LOX,KCHNL,NUMC,
     1    NRATOR,NDENOM,PRX,PRY,I1,I2)
      GO TO 420
  410 KHOP= KHOP - 1
  420 CONTINUE
C SELECTING ACTIVE CONSTRAINTS
      K=0
      DO 411 I=1,NM
      IF(I-1.EQ.KMIN)KMIN=K
      PORM=1.0
      IF(I.GT.KMIN)PORM=-1.0
      IF(PORM*STBM(I).GT.PORM*RQ(I))GO TO 411
      K=K+1
```

```
      KPTS(K)= KPTS(I)
      TYPE(K)= TYPE(I)
      SML(K)= STBM(I)
      KACT(K)= KPTS(I)
  411 CONTINUE
      NM=K
      NML=NM
C CALCULATION OF GRADIENTS OF ACTIVE CONSTRAINTS
      RPT=1.0
      CALL PARCLT(XCOMN,YCOMN,GR,GI,OMEGA,NM,NRATOR,NDENOM,
     1   KCHNL,NUMC,KONT,GCOMR,GCOMI,G,PPT,LSN,KPARC,KPTS,KNR,KDR)
C SET DOT PRODUCT VECTOR
      DO 422 K=1,NM
  422 WEIGHT(K)=1.0
C CALCULUTE DIRVECTIONAL VECTOR
      LRE=0
      KRE=0
  423 IF(NM.GT.LPRESV)WRITE(6,415)
      IF(NM.GT.LPRESV)GO TO 340
      CALL DIRVEC(G,NM,KPARC,DV,WEIGHT)
  415 FORMAT('0',5X,'***** TERMINATION REASON - NO, OF ACTIVE CONSTRAINT
     1S IS GREATER THAN THE NO. OF ALLOWABLE VARIABLES *****')
      DO 426 I=1,I1
  426 PRX(I)= DV(I)
      DO 427 I=1,I2
  427 PRY(I)= DV(I1+I)
      IF(KRE.EQ.1)GO TO 433
C CKECKING POSSIBLE NEGATIVENESS OF ANY COMPENSATOR COEF.
      IF(LRE.GE.I1+I2)GO TO 433
      LRE=LRE+1
      K2=0
      K3=0
      DO 431 K=1,KCHNL
      LNC=KNR(K)
      LDC=KDR(K)
      DO 429 I=1,LNC
      K2=K2+1
      IF(XCOMN(K,I).GT.1.0E-05)GO TO 429
      IF(PRX(K2).GE.0.0)GO TO 429
      LPRESV=LPRESV-1
      KRE=1
      DO 428 J=1,NM
  428 G(J,K2)=0.0
  429 CONTINUE
      DO 431 I=1,LDC
      K3=K3+1
      IF(YCOMN(K,I).GT.1.0E-05)GO TO 431
      IF(PRY(K3).GE.0.0)GO TO 431
      LPRESV=LPRESV-1
      KRE=1
      DO 430 J=1,NM
  430 G(J,I1+K3)=0.0
```

```
431 CONTINUE
    IF(KRE.EQ.1)GO TO 423
433 CONTINUE
    PSQ= 0.0
    DO 438 I=1,I1
    PX(I)= PRX(I)
438 PSQ= PSQ + PX(I)**2
    DO 440 I=1,I2
    PY(I)=PRY(I)
440 PSQ= PSQ + PY(I)**2
    PMG= SQRT(PSQ)
    PSQL= PSQ
    DEL= STEP/PMG
    DO 462 K=1,KCHNL
    LNC= KNR(K)
    DO 462 I=1,LNC
462 COTN(K,I)= XCOMN(K,I)
    DO 464 K=1,KCHNL
    LDC= KDR(K)
    DO 464 I=1,LDC
464 COTD(K,I)= YCOMN(K,I)
    GO TO 465
450 DEL= DEL/2.0
    DO 467 K=1,KCHNL
    LNC= KNR(K)
    DO 467 I=1,LNC
467 XCOMN(K,I)= COTN(K,I)
    DO 468 K=1,KCHNL
    LDC= KDR(K)
    DO 468 I=1,LDC
468 YCOMN(K,I)= COTD(K,I)
465 CONTINUE
    KKK=0
    DO 470 K=1,KCHNL
    LNC= KNR(K)
    DO 470 I=1,LNC
    KKK= KKK+1
    XCOMN(K,I)= XCOMN(K,I) + DEL * PX(KKK)
470 IF(XCOMN(K,I).LT.0.0)XCOMN(K,I)=0.0
    KKK=0
    DO 490 K=1,KCHNL
    LDC= KDR(K)
    DO 490 I=1,LDC
    KKK= KKK+1
    YCOMN(K,I)= YCOMN(K,I)+ DEL * PY(KKK)
490 IF(YCOMN(K,I).LT.0.0)YCOMN(K,I)=0.0
500 CONTINUE
    LOX= LOX + 1
    GO TO 200
    END
```

```
      SUBROUTINE PARCLT(XCOMN,YCOMN,GOR,GOI,OMEGA,NFREQ,NRATOR,NDENOM,
     1   KCHNL,NUMC,KONT,GCOMR,GCOMI,G,PPT,LSN,NPARC,KPTS,KNR,KDR)
C
C PROGRAM FOR CALCULATING THE CHANGE OF A FREQUENCY RESPONSE WITH
C RESPECT TO A CONPENSATOR COEFFICIENTS
C
C DEFINITIONS OF  I/O VARIABLES
C
C XCOMN(I,J)   -NUMERATOR COEFS. OF COMPENSATOR IN I-TH CHANNEL
C YCOMN(I,J)   -DENOM. COEFS. OF COMPENSATOR IN  I-TH CHANNEL
C GOR(I)       -I-TH REAL PART OF OPEN LOOP FREQ. RESP.
C GOI(I)       -I-TH IMAG. PART OF OPEN LOOP FREQ. RESP.
C OMEGA(I)     -I-TH FREQUENCY RESPONSE POINT
C NFREQ        -NUMBER OF MARGINS TO BE IMPROVED
C NRATOR(I,J)  -NUM. ORDER OF J-TH COMP. IN I-TH CHANNEL
C NDENOM(I,J)  -DEN. ORDER OF J-TH COMP. IN I-TH CHANNEL
C KCHNL        -NUM. OF CHANNELS
C NUMC(I)      -NUM. OF COMPS. IN  I-TH CHANNEL
C KONT(I)      -GAIN CONTROL NUM. FOR I-TH CHANNEL
C GCOMR(I,J)   -REAL PART OF J-TH CHANNEL COMP. FREQ. RESP. AT J-TH FREQ.
C GCOMI(I,J)   -IMAG. PART OF J-TH CHNL. COMP. FREQ. RESP. AT J-TH FREQ.
C G(I,J)       -J-TH PARTIAL OF I-TH FREQ.
C Z            -NEG. OF POINT FOR WHICH PARTIALS ARE DESIRED
C L            -NO. OF POINTS TO TREAT AS STABILITY MARGINS(THE REMAINING
C                ARE CONSIDERED AS ATTENUATION MARGINS)
      DIMENSION C(10),D(10),E(10),GR(50),GI(50),OMEGA(999),Y(10),X(10),
     1   NUMC(20),KONT(10),G(20,99),GOR(999),GOI(999),NRATOR(10,20),
     2   NDENOM(10,20),GCOMR(5,999),GCOMI(5,999),PFX1(5,50),
     3   PFY1(5,50), KPTS(1),XCOMN(10,50),YCOMN(10,50),KNR(1),KDR(1)
      DOUBLE PRECISION G
      IMPLICIT REAL*8(A-F,P-W)
      REAL*4 X,Y,XV,CNR,CNI,XCOMN,YCOMN,CDR,CDI
      DIMENSION PPT(4), LSN(4)
      COMMON TYPE(50)
      INTEGER TYPE
      COMPLEX P,PPT
      DO 140 J=1,NFREQ
      IF(TYPE(J).EQ.'G')P=-PPT(1)
      IF(TYPE(J).EQ.'P')P=-PPT(2)
      IF(TYPE(J).EQ.'S')P=-PPT(3)
      IF(TYPE(J).EQ.'A')P=-PPT(4)
      IF(TYPE(J).EQ.'G')SGN= LSN(1)
      IF(TYPE(J).EQ.'P')SGN= LSN(2)
      IF(TYPE(J).EQ.'S')SGN= LSN(3)
      IF(TYPE(J).EQ.'A')SGN= LSN(4)
      KWHICH= KPTS(J)
      XV= OMEGA(KWHICH) * 6.2831853
      DO 130 L=1,KCHNL
      NCOMD= NUMC(L)
      KNOT=0
      LNOT=0
      IOP= KONT(L)
```

```
      DO 130 N=1,NCOMD
      IF(N.GT.1)IOP=2
      N1= NRATOR(L,N) + 1
      M1= NDENOM(L,N) + 1
      DO 5 LP=1,N1
  5 X(LP)= XCOMN(L,LP+KNOT)
      DO 6 LP=1,M1
  6 Y(LP)= YCOMN(L,LP+LNOT)
      K2=N1-1
      K3=M1-1
      CALL POLFV(X,K2,XV,CNR,CNI)
      CALL POLFV(Y,K3,XV,CDR,CDI)
      RD= CNR**2 + CNI**2
      RR= (CDR*CNR+CDI*CNI)/RD
      RI=(-CDR*CNI+CDI*CNR)/RD
      GR(J)= GCOMR(L,KWHICH)*RR - GCOMI(L,KWHICH)*RI
      GI(J)= GCOMR(L,KWHICH)*RI + GCOMI(L,KWHICH)*RR
      A= REAL(P)+GOR(KWHICH)-GCOMR(L,KWHICH)
      B= AIMAG(P)+GOI(KWHICH)-GCOMI(L,KWHICH)
      FREQ=1.0
      KSKIP=1
      DO 40 I=1,N1
      KUL1=(-1)**((I+1)/2)
      KUL2=(-1)**((I+2)/2)
      FUL1=KUL1
      FUL2=KUL2
      IF(KSKIP-1)20,20,30
 20 C(I)=-GR(J)*FREQ*FUL2
      D(I)=-GI(J)*FREQ*FUL1
      KSKIP=2
      GO TO 40
 30 C(I)=-GI(J)*FREQ*FUL2
      D(I)=-GR(J)*FREQ*FUL1
      KSKIP=1
 40 FREQ= FREQ*OMEGA(KWHICH)*6.2831853
      FREQ= 1.0
      DO 50 I=1,M1
      KMUL=(-1)**((I+1)/2)
      EMUL=KMUL
      E(I)= -FREQ * EMUL
 50 FREQ= FREQ * OMEGA(KWHICH)*6.2831853
      FNA1=0.0
      FNA2=0.0
      DO 60 I=1,N1
      FNA1=FNA1+C(I)*X(I)
 60 FNA2=FNA2+D(I)*X(I)
      FD2=0.0
      KI= 2 * ((K3+1)/2)
      DO 70 I=2,KI,2
 70 FD2=FD2+E(I)*Y(I)
      FD1=0.0
      KE= 2 * ((K3+2)/2) - 1
```

```
      DO 80 I=1,KE,2
   80 FD1=FD1+E(I)*Y(I)
      FN1= FNA1 + FD1 * A - FD2 * B
      FN2= FNA2 + FD2 * A + FD1 * B
      FD=FD1**2+FD2**2
      FN=FN1**2+FN2**2
      FYE= (FD *(A * FN1 + B * FN2) - FN * FD1)/ FD**2
      FYO= (FD*(-B * FN1 + A * FN2) - FN * FD2)/ FD**2
      FX1=FN1/FD
      FX2=FN2/FD
      PFX1(L,KNOT+1)= 0.0
      DO 90 I=1,KE,2
      PFY1(L,I+LNOT)= FYE * E(I) * SGN
   90 CONTINUE
      DO 100 I=2,KI,2
      PFY1(L,I+LNOT)= FYO * E(I) * SGN
  100 CONTINUE
      IF(IOP.EQ.2)PFY1(L,LNOT+1)= 0.0
      DO 110 I=2,N1
      PFX1(L,I+KNOT)=(FX1*C(I) + FX2*D(I)) * SGN
  110 CONTINUE
      KNOT= KNOT + N1
      LNOT= LNOT + M1
  130 CONTINUE
      KLAD=0
      DO 135 IX=1,KCHNL
      KNOT= KNR(IX)
      DO 135 LX=1,KNOT
      KLAD=KLAD+1
  135 G(J,KLAD)= PFX1(IX,LX)
      DO 139 IX=1,KCHNL
      LNOT= KDR(IX)
      DO 139 LX=1,LNOT
      KLAD=KLAD+1
  139 G(J,KLAD)=PFY1(IX,LX)
  140 CONTINUE
      NPARC=KLAD
      DO 150 J=1,NFREQ
      SUM=0.0
      DO 145 I=1,NPARC
  145 SUM=SUM+G(J,I)*G(J,I)
      SUM= DSQRT(SUM)
      DO 149 I=1,NPARC
  149 G(J,I)= G(J,I)/SUM
  150 CONTINUE
      RETURN
      END


      SUBROUTINE PHASEM(GR,GI,KPOINT,NM,FQMIN,FQMAX,KPTS,STBM,OMEGA)
      DIMENSION GR(1),GI(1),KPTS(1),STBM(1),OMEGA(1)
C
```

```
C SUBPROGRAM FOR CALCULATING PHASE MARGINS
C
C   DEFINITIONS OF I/O VARIABLES
C
C   GR     -ARRAY OF OPEN LOOP TRANSFER FUNCTION REAL PARTS
C   GI     -ARRAY OF OPEN LOOP TRANSFER FUNCTION IMAGINARY PARTS
C KPOINT-NO. OF POINTS
C   OMEGA -ARRAY OF FREQS.
C   NM     -COUNTER
C KPTS    -FREQUENCY NOS. WHERE MARGINS OCCUR
C   STBM  -STABILITY MARGINS OF MARGINS
C   FQMIN -LOWER FREQ. FOR MARGIN DETECTION
C   FQMAX - UPPER FREQ. FOR MARGIN DETECTION
      P=1.0
      DO 50 I=1,KPOINT
      SO= GR(I)**2 + GI(I)**2
      S2=SO-1.0
      IF(I.EQ.1)S1=S2
      IF(OMEGA(I).LT.FQMIN)GO TO 40
      IF(OMEGA(I).GT.FQMAX)RETURN
      IF(ABS(S2).LT.1.0E-20)GO TO 30
      SGN=S2/ABS(S2)
      IF(S1*SGN.GT.0.0)GO TO 40
   30 I1=I-1
      IF(ABS(S2).LT.ABS(S1))I1=I
      NM=NM+1
      KPTS(NM)=I1
      S3= (P+GR(I1))**2+GI(I1)**2
      STBM(NM)= SQRT(S3)
   40 S1=S2
   50 CONTINUE
      RETURN
      END


      SUBROUTINE GAINMG(GR,GI,KPOINT,NM,FQMIN,FQMAX,KPTS,STBM,OMEGA)
      DIMENSION GR(1),GI(1),KPTS(1),STBM(1),OMEGA(1)
C
C SUBPROGRAM FOR CALCULATING GAIN MARGINS
C
C   DEFINITIONS OF I/O VARIABLES
C
C   GR     -ARRAY OF OPEN LOOP TRANSFER FUNCTION REAL PARTS
C   GI     -ARRAY OF OPEN LOOP TRANSFER FUNCTION IMAGINARY PARTS
C KPOINT-NO. OF POINTS
C   OMEGA -ARRAY OF FREQS.
C   NM     -COUNTER
C KPTS    -FREQUENCY NOS. WHERE MARGINS OCCUR
C   STBM  -STABILITY MARGINS OF MARGINS
C   FQMIN -LOWER FREQ. FOR MARGIN DETECTION
C   FQMAX - UPPER FREQ. FOR MARGIN DETECTION
      P=1.0
```

```
      DO 50 I=1,KPOINT
      S2=GI(I)
      IF(I.EQ.1)S1=S2
      IF(OMEGA(I).LT.FQMIN)GO TO 40
      IF(OMEGA(I).GT.FQMAX)RETURN
      IF(ABS(S2).LT.1.0E-20)GO TO 30
      SGN=S2/ABS(S2)
      IF(S1*SGN.GT.0.0)GO TO 40
   30 CR= GR(I)
      IF(CR.GE.0.0)GO TO 40
      I1=I-1
      IF(ABS(S2).LT.ABS(S1))I1=I
      NM=NM+1
      KPTS(NM)=I1
      S3= (P+GR(I1))**2+GI(I1)**2
      STBM(NM)= SQRT(S3)
   40 S1=S2
   50 CONTINUE
      RETURN
      END


      SUBROUTINE SRMINS(GR,GI,KPOINT,NM,P,N,FQMIN,KPTS,STBM,OMEGA)
      DIMENSION GR(1),GI(1),KPTS(1),STBM(1),OMEGA(1)
C
C  SUBPROGRAM FOR DETERMINING THE MINMUNS OF THE OPEN LOOP FREQUENCY
C      RESPONSE WITH RESPECT TO THE -1 POINT WHEN GIVEN POINTS ON THE
C      OPEN LOOP REQUENCY RESPONSE
C
C  DESCRIPTION OF I/O VARIABLES
C      GR       - VECTOR OF REAL PARTS OF OPEN LOOP FREQUENCY RESPONSE
C      GI       - VECTOR OF IMAGINARY PARTS OF OPEN LOOP FREQ. RESPONSE
C      KPOINT - NUMBER POINTS OF THE OPEN LOOP FREQ. RESPONSE GIVEN
C      OMEGA  - CORRESPONDING FREQUENCIES OF CHOSEN POINTS
C KPTS     -FREQUENCY NOS. WHERE MARGINS OCCUR
C FQMIN    -MINIMUM FRQ. CONSIDERED
C -P       -POINT W.R.T. A MAX. OR MIN. IS DESIRED
C N        -DETERMINES WHETHER A MAX. OR MIN. IS DETERMINED
      ASN1=0.0
      S1=0.0
      DO 50 I=1,KPOINT
      IF(OMEGA(I).LE.FQMIN)GO TO 50
      S2= (P + GR(I))**2 + GI(I)**2
      ASN2=S2-S1
    5 CONTINUE
      IF(ASN2*N)10,50,10
   10 IF(ASN1*ASN2)20,40,40
   20 IF(ASN1*N)30,40,40
   30 NM=NM+1
      I1= I - 1
      KPTS(NM)=I1
      STBM(NM)= SQRT(S1)
```

```
   40 S1=S2
      ASN1= ASN2
   50 CONTINUE
      RETURN
      END


      SUBROUTINE DIRVEC(G,NM,KPARC,DV,WEIGHT)
C DIRECTIONAL  VECTOR PROGRAM
C
C DEFINITIONS OF I/O VARIABLES
C
C G      -MATRIX WHOSE ROWS CONTAIN THE GRADIENT VECTORS OF THOSE
C          STABILITY MARGINS ONLY CONSIDERED PERTINENT
C NM     -NUMBER OF STABILITY MARGINS CONSIDERED PERTINENT
C WEIGHT-WEIGHTING FACTOR VECTOR
C
      DIMENSION G(20,99), A(30,30), WEIGHT(1),
     /    AI(30,30), X(30), DV(30), Y(30)
      IMPLICIT REAL*8(A-H,O-Z)
      DO 200 K=1,NM
      Y(K)= WEIGHT(K)
      DO 200 J=K,NM
      SUM= 0.0
      DO 150 I=1,KPARC
  150 SUM= SUM + G(J,I) * G(K,I)
      A(J,K)= SUM
      A(K,J)= SUM
  200 CONTINUE
      IF(NM.GT.1)GO TO 230
      AI(1,1)= 1.0/A(1,1)
      X(1)= WEIGHT(1) * AI(1,1)
      GO TO 310
  230 CONTINUE
      CALL MATINV(A,NM,AI,IER)
      IF(IER.EQ.0)GO TO 300
      WRITE(6,250)
  250 FORMAT('0',15X,'THE PARTIALS ARE NOT LINEARLY INDEPENDENT.  THUS T
     /HE PROGRAM IS TERMINATED.')
      STOP
  300 CALL MATMUL(NM,AI,NM,Y,1,X)
  310 CONTINUE
      DO 450 I=1,KPARC
      SUM= 0.0
      DO 400 J=1,NM
  400 SUM= SUM + G(J,I) * X(J)
  450 DV(I)= SUM
  690 RETURN
      END


      SUBROUTINE MATINV(Z,N,Y,IER)
```

```
      DO 50 I=1,KPOINT
      S2=GI(I)
      IF(I.EQ.1)S1=S2
      IF(OMEGA(I).LT.FQMIN)GO TO 40
      IF(OMEGA(I).GT.FQMAX)RETURN
      IF(ABS(S2).LT.1.0E-20)GO TO 30
      SGN=S2/ABS(S2)
      IF(S1*SGN.GT.0.0)GO TO 40
   30 CR= GR(I)
      IF(CR.GE.0.0)GO TO 40
      I1=I-1
      IF(ABS(S2).LT.ABS(S1))I1=I
      NM=NM+1
      KPTS(NM)=I1
      S3= (P+GR(I1))**2+GI(I1)**2
      STBM(NM)= SQRT(S3)
   40 S1=S2
   50 CONTINUE
      RETURN
      END


      SUBROUTINE SRMINS(GR,GI,KPOINT,NM,P,N,FQMIN,KPTS,STBM,OMEGA)
      DIMENSION GR(1),GI(1),KPTS(1),STBM(1),OMEGA(1)
C
C SUBPROGRAM FOR DETERMINING THE MINMUNS OF THE OPEN LOOP FREQUENCY
C     RESPONSE WITH RESPECT TO THE -1 POINT WHEN GIVEN POINTS ON THE
C     OPEN LOOP REQUENCY RESPONSE
C
C DESCRIPTION OF I/O VARIABLES
C     GR      - VECTOR OF REAL PARTS OF OPEN LOOP FREQUENCY RESPONSE
C     GI      - VECTOR OF IMAGINARY PARTS OF OPEN LOOP FREQ. RESPONSE
C     KPOINT - NUMBER POINTS OF THE OPEN LOOP FREQ. RESPONSE GIVEN
C     OMEGA  - CORRESPONDING FREQUENCIES OF CHOSEN POINTS
C KPTS     -FREQUENCY NOS. WHERE MARGINS OCCUR
C FQMIN  -MINIMUM FRQ. CONSIDERED
C -P       -POINT W.R.T. A MAX. OR MIN. IS DESIRED
C N        -DETERMINES WHETHER A MAX. OR MIN. IS DETERMINED
      ASN1=0.0
      S1=0.0
      DO 50 I=1,KPOINT
      IF(OMEGA(I).LE.FQMIN)GO TO 50
      S2= (P + GR(I))**2 + GI(I)**2
      ASN2=S2-S1
    5 CONTINUE
      IF(ASN2*N)10,50,10
   10 IF(ASN1*ASN2)20,40,40
   20 IF(ASN1*N)30,40,40
   30 NM=NM+1
      I1= I - 1
      KPTS(NM)=I1
      STBM(NM)= SQRT(S1)
```

```
C
C      MULTIPLIES (A) * (B)
C      A IS AN NR X N
C      B IS AN N X NC
C      X IS AN NR X NC
C
       DO 4 I=1,NR
     4 X(I) = 0.0
       DO 5 I=1,NR
       DO 5 K=1,NC
       DO 5 J=1,N
     5 X(I)= X(I) + A(I,J) * B(J)
       RETURN
       END



       SUBROUTINE POLFV(FW,K,X,FREAL,FIMAG)
C PROGRAM FOR EVALUATING A POLYNOMIAL AT AN IMAGINARY FREQUENCY
C
C DEFINITIONS OF I/O VARIABLES
C
C  FW     -VECTOR POLYNOMIAL COEFFICIENTS
C  K      -ORDER OF POLYNOMIAL
C  X      -FREQUENCY TO BE EVALUATED AT
C  FREAL  -REAL PART OF FW(JX)
C  FIMAG  -IMAGINARY OF FW(JX)
       DIMENSION FW(1)
       KEVEN=(K+2)/2
       KODD=(K+1)/2
       Y=1.0
       FREAL=0.0
       DO 10 I=1,KEVEN
       L=2*I-1
       FREAL= FREAL + FW(L)*Y
    10 Y=-Y*X*X
       FIMAG=0.0
       IF(KODD.EQ.0)GO TO 30
       Y=X
       DO 20 I=1,KODD
       L=2*I
       FIMAG= FIMAG + FW(L)*Y
    20 Y=-Y*X*X
    30 RETURN
       END



       SUBROUTINE OTPT1(STBM,OMEGA,KPTS,NM,XCOMN,YCOMN,KMIN,RQ,LOX,KCHNL,
      1    NUMC,NRATOR,NDENOM,PRX,PRY,I1,I2)
       DIMENSION STBM(1),OMEGA(1),KPTS(1),XCOMN(10,50),YCOMN(10,50),RQ(1)
      1    ,PRX(1),PRY(1),NUMC(1),NRATOR(10,20),NDENOM(10,20)
       DIMENSION TYPE(50)
       COMMON TYPE
```

```
      INTEGER TYPE
C
C SUBPROGRAM FOR THE OUTPUT OF INFORMATION CALCULATED
C
      IOP=1
      WRITE(6,10) LOX
   10 FORMAT('0',25X,'ITERATION NO. ',I4)
      DO 110 I=1,NM
      KWH= KPTS(I)
      FREQ= OMEGA(KWH)
      IF(I.EQ.KMIN+1)GO TO 50
      IF(I.EQ.1)GO TO 70
      GO TO 90
   50 WRITE(6,60)
   60 FORMAT('0',25X,'ATTENUATED FREQUENCY INFORMATION'//)
      GO TO 90
   70 WRITE(6,80)
   80 FORMAT('0',25X,'RELATIVE STABILITY INFORMATION'//)
   90 CONTINUE
      WRITE(6,100) I, STBM(I), FREQ, RQ(I), TYPE(I)
  100 FORMAT(' ',2X,'MARGIN RADIUS NO. ',I2,'=',F10.5,5X,'FREQUENCY=',
     1    F10.5,1X,'HZ',5X,'DESIRED MARGIN=',F10.5,5X,'MARGIN TYPE=',1X,
     2    A1)
  110 CONTINUE
      DO 104 I=1,KCHNL
      WRITE(6,102) I
  102 FORMAT('0',25X,'CHANNEL NO.',I2,' COMPENSATORS')
      L=NUMC(I)
      LX=1
      LY=1
      KX=0
      KY=0
      DO 104 IX=1,L
      KX=KX + NRATOR(I,IX) + 1
      KY=KY + NDENOM(I,IX) + 1
      WRITE(6,106) (XCOMN(I,N),N=LX,KX)
      WRITE(6,107) (YCOMN(I,N),N=LY,KY)
  106 FORMAT('0',10X,'NUMERATOR',8F10.5)
  107 FORMAT('0',8X,'DENOMINATOR',8F10.5)
      LX=KX+1
      LY=KY+1
  104 CONTINUE
      WRITE(6,130) (PRX(I),I=IOP,I1)
      WRITE(6,120) (PRY(I),I=IOP,I2)
  120 FORMAT('0','PARTIALS WITH RESPECT TO Y',8E10.3)
  130 FORMAT('0','PARTIALS WITH RESPECT TO X',8E10.3)
      WRITE(6,160)
  160 FORMAT('0')
      RETURN
      END
```

```
      SUBROUTINE MULOUT(KCHNL,NUMC,NRATOR,NDENOM,KNR,KDR,XCOMN,YCOMN)
      DIMENSION CON(30), COM(30), XCOF(30), XCOMN(10,50), YCOMN(10,50),
     1    NUMC(30), NRATOR(10,20), NDENOM(10,20), KNR(20), KDR(20)
      DO 80 I=1,KCHNL
      CON(1)= 1.0
      N=0
      LAX=1
      NAT= NUMC(I)
      WRITE(6,40) I
   40 FORMAT('0',25X,'CHANNEL NO.',I2,2X,'COMPENSATOR')
      DO 65 J=1,NAT
      M=NRATOR(I,J)
      M1= M + 1
      LAY= LAX + M
      KL=0
      DO 62 K=LAX,LAY
      KL=KL+1
   62 COM(KL)= XCOMN(I,K)
      LAX= LAY + 1
      CALL POLMUL(CON,COM,N,M,XCOF)
      N=N+M
      N1=N+1
      DO 64 K=1,N1
   64 CON(K)= XCOF(K)
   65 CONTINUE
      WRITE(6,67)
   67 FORMAT('0',25X,'NUMERATOR COEFFICIENTS')
      WRITE(6,69) (CON(J),J=1,N1)
   69 FORMAT('0',2X,7E15.5)
      CON(1)=1.0
      N=0
      LAX=1
      DO 75 J=1,NAT
      M= NDENOM(I,J)
      M1= M+1
      LAY= LAX+M
      KL=0
      DO 72 K=LAX,LAY
      KL=KL+1
   72 COM(KL)= YCOMN(I,K)
      LAX= LAY + 1
      CALL POLMUL(CON,COM,N,M,XCOF)
      N=N+M
      N1=N+1
      DO 74 K=1,N1
   74 CON(K)= XCOF(K)
   75 CONTINUE
      WRITE(6,77)
   77 FORMAT('0',25X,'DENOMINATOR COEFICIENTS')
      WRITE(6,69) (CON(J),J=1,N1)
   80 CONTINUE
      RETURN
```

```
      END


      SUBROUTINE POLMUL(CON,COM,N,M,XCOF)
      DIMENSION CON(1), COM(1), XCOF(1), CONA(50), COMRA(50)
C
C     FOR DOUBLE PRECISION REMOVE C FROM FIRST COLUMN OF NEXT CARD.
C     DOUBLE PRECISION CON, COM, XCOF, CONA, COMRA
C
C     THE VECTOR CON IS A VECTOR OF THE COEFFICIENT OF A POLYNOMIAL
C     OF ORDER N.
C     THE VECTOR COM IS A VECTOR OF THE COEFFICIENTS OF A POLYNOMIAL OF
C     ORDER M.
C     THE VECTOR XCOF IS A VECTOR OF THE COEFFICIENTS OF THE PRODUCT OF
C     A POLYNOMIAL OF ORDER N AND A POLYNOMIAL OF ORDER M.  THE
C     POLYNOMIAL OF WHICH THE COEFFICIENTS ARE THE VECTOR XCOF HAS AN
C     ORDER OF M + N.
C
      DO 1 I=1,M
    1 CONA(I)=0.0
      NX=N+1
      DO 2  I=1,NX
      LX=M+I
    2 CONA(LX)=CON(I)
      MX=M+1
      DO 3  I=1,MX
      MY=M+2-I
    3 COMRA(I)=COM(MY)
      DO 4  I=1,N
      NX=M+1+I
    4 COMRA(NX)=0.0
      KY=M+N+1
      KX=KY
      DO 7  K=1,KY
      XCOF(K)=0.0
      DO 5 L=1,KX
    5 XCOF(K)= CONA(L) * COMRA(L)+XCOF(K)
      KX=KX-1
      DO 6 J=1,KX
    6 CONA(J)=CONA(J+1)
    7 CONTINUE
      RETURN
      END
```

REFERENCES

1. G. B. Dantzig, "Recent Advances in Linear Programming," _Management Science_, 2, 1956, pp. 131-144.

2. L. Cooper and D. Steinberg, _Introduction to Methods of Optimization_, W. B. Saunders Co., Philadelphia, Penn., 1970.

3. G. B. Dantzig, _Linear Programming and Its Extensions_, Princeton U. Press, Princeton, N. J., 1963.

4. G. Hadley, _Linear Programming_, Addison-Wesley, Reading, Mass., 1962.

5. S. I. Gass, _Linear Programming: Methods and Applications_, Third Edition, McGraw-Hill, New York, N. Y., 1969.

6. G. Hadley, _Nonlinear and Dynamic Programming_, Addison-Wesley, Reading, Mass., 1964.

7. Jerrel R. Mitchell and Cecil D. Burge, "Computer Aided System Compensation," Interim Report, Contract No. NAS8-21377, NASA, MSFC, Huntsville, Ala.

8. Jerrel R. Mitchell and Willie L. McDaniel, "Automated Compensator Design," _Proceeding of the 1970 Symposium on Adaptive Processes_ (9th), The University of Texas at Austin, pp. XIII.4.1-XIII.4.5.

9. Jerrel R. Mitchell and Willie L. McDaniel, "Compensator Improvement with an Application to a Large Space Vehicle," _Proceeding of the 1971 Conference on Decision & Control_, The University of Florida, Gainsville, Fla., pp. 207-213.

10. Jerrel R. Mitchell, "Extensions of Compensator Improvement to Include Multi-Channel Systems and Factored Compensators," Interim Report, Contract No. NAS8-21377, NASA, MSFC, Huntsville, Ala.

11. P. Wolfe, "The Simplex Method for Quadratic Programming," _Econometricia_, 27, 1959, pp. 382-398.

12. H. O. Hartley and R. R. Hocking, "Convex Programming by Tangential Approximation," _Management Science_, 9.4, 1963, pp. 600-612.

13. J. B. Rosen, "The Gradient Projection Method for Nonlinear Programming, Part I," _Journal of the Society for Industrial and Applied Mathematics_, 8, 1960, pp. 181-217.

14. J. B. Rosen, "The Gradient Projection Method for Nonlinear Programming, Part II," Journal of the Society for Industrial and Applied Mathematics, 9, 1961, pp. 514-532.

15. A. V. Fiacco and G. P. McCormick, "Computational Algorithms for the Sequentially Unconstrained Minimization Techniques for Nonlinear Programming," Management Science, Vol. 10, 1964, pp. 601-617.

16. C. W. Carroll, "The Created Response Surface Technique for Optimizing Nonlinear Restrained Systems," Operations Res., Vol. 9, 1961, pp. 169-184.

17. A. V. Fiacco and G. P. McCormick, Nonlinear Programming: Sequential Unconstrained Minimization Techniques, Wiley, New York, 1968.

18. J. Abadie and J. Carpentier, "Generalization of the Wolfe Reduced Gradient Method to the Case of Nonlinear Constraints," Optimization, edited by R. Fletcher, Academic Press, New York, N. Y.

19. M. D. Canon, C. D. Cullum, and E. Polak, Theory of Optimal Control and Mathematical Programming, New York, N. Y., McGraw-Hill, 1970.

20. H. Halkin and L. W. Neustadt, "General Necessary Conditions for Optimization Problems," Proc. Nat. Acad. Sci. U. S., Vol. 56, 1966, pp. 1066-1071.

21. L. W. Neustadt, "An Abstract Variational Theory With Applications to a Broad Class of Optimization Problems, Part I--General Theory," SIAM J. Contr., Vol. 4, 1966, pp. 505-527.

22. L. W. Neustadt, "An Abstract Variational Theory With Applications to a Broad Class of Optimization Problems, Part II--Applications," SIAM J. Contr., Vol. 5, 1967, pp. 90-131.

23. D. Tabak, "Applications of Mathematical Programming Techniques: A Survey," IEEE Transactions on Automatic Control, Vol. AC-15, No. 6, 1970, pp. 688-690.

24. B. C. Kuo and D. Tabak, "A Nonlinear Programming Approach to Optimal Control of Discrete Systems with Constrained Sampling Times," Proceeding of the 1968 JACC, Michigan State University, Ann Arbor, Michigan, 1968.

25. J. Abadie, "Application of the GRG Algorithm to Optimal Control Problems," The GRG Algorithm and Control Problems, North-Holland Publishing Co., Amsterdam, Netherlands, 1970, pp. 191-211.

26. C. D. Burge, "A Mixed Integer Programming Algorithm for Application to Optimal Control Problems," Ph.D. Dissertation, Mississippi State University, State College, Mississippi, 1970.

27. T. C. Coffey, "Automated Frequency--Domain Synthesis of Multiloop Control Systems," *AIAA Journal*, Vol. 8, No. 10, 1970.

28. J. A. Page, "Automated Design of Feedback Control Systems," Ph.D. Dissertation, University of California, Los Angeles, 1970.

29. Edwin B. Stear and John A. Page, "Automated Design of Multivariable Control Systems," *Proceedings of the 1971 IEEE Decision and Control Conference*, University of Florida, Gainsville, Fla., pp. 192-196.

30. J. Abadie, Editor, *Nonlinear Programming*, North-Holland Publishing Company, Amsterdam, Holland, 1967, pp. 121-123.

31. F. B. Hildebrand, *Methods of Applied Mathematics*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1965.

32. F. B. Hildebrand, *Advanced Calculus for Applications*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1962.