A MODEL PERFORMANCE INDEX IMPROVEMENT

by

LUIZ CLAUDIO GOULART COUTINHO MARQUES

B.S., Military Institute of Engineering (Brazil) (1969)

SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE
DEGREE OF
MASTER OF SCIENCE
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
(February 1977)

Signature of Author

Department of Aeronautics and Astronautics

Certified by

Thesis Advisor

Accepted by

Chairman, Departmental Graduate Committee

Aero

FEB 28 1977
A Model Performance Index Improvement

by

Luiz Claudio Goulart Coutinho Marques

Submitted to the Department of Aeronautics and Astronautics on January 21, 1977 in partial fulfillment of the requirements for the Degree of Master of Science.

ABSTRACT

An investigation of the quality of matching between a pre-selected model and systems chosen through the lower limit formulation was carried out. From this analysis, a new functional relationship of the states was developed which not only permits one to select analytically the optimum value for lower limit but allows the direct selection of the optimum set of free parameters. The capabilities of the new formulation were tested through a series of examples, in which we tried to focus the case in which zeroes show up in the transfer functions, a situation not properly handled by any previous work.

Thesis Supervisor:  Henry Philip Whitaker
Title:  Professor of Aeronautics and Astronautics
ACKNOWLEDGMENTS

The author wishes to express his deepest admiration and respect to Professor H. Philip Whitaker who, as Thesis Advisor, provided motivation and continuous constructive criticism throughout this work.

Special appreciation must also be given to my wife, Terezinha, for her encouragement and patience during the course of this work.
## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONE</td>
<td>INTRODUCTION</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>1.1 Historical Summary of Problem Area</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>1.2 Previous Work done by Rediess and Palsson</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>1.3 Purpose of this Work</td>
<td>10</td>
</tr>
<tr>
<td>TWO</td>
<td>REVIEW AND ANALYSIS OF REDIESS, PALSSON</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>AND HO'S WORK</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.1 Rediess' Formulation</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>2.2 Palsson's Formulation</td>
<td>19</td>
</tr>
<tr>
<td></td>
<td>2.3 Ho's Formulation</td>
<td>24</td>
</tr>
<tr>
<td>THREE</td>
<td>THEORETICAL DEVELOPMENT</td>
<td>26</td>
</tr>
<tr>
<td>FOUR</td>
<td>CONCLUSION</td>
<td>35</td>
</tr>
<tr>
<td></td>
<td>REFERENCES</td>
<td>49</td>
</tr>
</tbody>
</table>
### List of Principal Nomenclature and Notation

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n )</td>
<td>order of the system</td>
</tr>
<tr>
<td>( m )</td>
<td>number of system zeros</td>
</tr>
<tr>
<td>( a_i )</td>
<td>system characteristic polynomial coefficient of ( s^i )</td>
</tr>
<tr>
<td>( b_i )</td>
<td>system transfer function numerator polynomial coefficient of ( s^i )</td>
</tr>
<tr>
<td>( y )</td>
<td>system output variable</td>
</tr>
<tr>
<td>( u )</td>
<td>system input variable</td>
</tr>
<tr>
<td>( y_{ss} )</td>
<td>steady state value of the output variable</td>
</tr>
<tr>
<td>( x )</td>
<td>system transient response variable</td>
</tr>
<tr>
<td>( s )</td>
<td>laplace operator</td>
</tr>
<tr>
<td>( ' )</td>
<td>indicates derivative with respect to time</td>
</tr>
<tr>
<td>( x_0 )</td>
<td>system's pseudo initial condition vector</td>
</tr>
<tr>
<td>( x )</td>
<td>system's transient response vector</td>
</tr>
<tr>
<td>( \dot{x} )</td>
<td>system's extended transient response vector</td>
</tr>
<tr>
<td>( a )</td>
<td>system's coefficient vector</td>
</tr>
<tr>
<td>( \hat{a} )</td>
<td>system's extended coefficient vector</td>
</tr>
<tr>
<td>MPI</td>
<td>Model Performance Index</td>
</tr>
<tr>
<td>( (\cdot)^1 )</td>
<td>indicates transpose of a vector</td>
</tr>
<tr>
<td>(</td>
<td></td>
</tr>
<tr>
<td>( r )</td>
<td>pseudo initial condition weighting factor</td>
</tr>
<tr>
<td>( \hat{a} )</td>
<td>model's coefficient vector</td>
</tr>
<tr>
<td>( \hat{\hat{a}} )</td>
<td>model's extended coefficient vector</td>
</tr>
<tr>
<td>( \hat{\hat{x}}_{mo} )</td>
<td>model's extended pseudo initial condition vector</td>
</tr>
<tr>
<td>Symbol</td>
<td>Definition</td>
</tr>
<tr>
<td>--------</td>
<td>------------</td>
</tr>
<tr>
<td>( l )</td>
<td>order of model</td>
</tr>
<tr>
<td>( I )</td>
<td>Identity matrix</td>
</tr>
<tr>
<td>( 0 )</td>
<td>null matrix</td>
</tr>
<tr>
<td>( \tilde{W} )</td>
<td>extended pseudo initial condition weighting matrix</td>
</tr>
<tr>
<td>( d(t) )</td>
<td>instantaneous distance between trajectories</td>
</tr>
<tr>
<td>PIC</td>
<td>Pseudo Initial Condition</td>
</tr>
<tr>
<td>( \hat{x} )</td>
<td>desired response state vector</td>
</tr>
<tr>
<td>( \hat{x}_0 )</td>
<td>desired pseudo initial condition vector</td>
</tr>
<tr>
<td>( \Delta x )</td>
<td>error state vector</td>
</tr>
<tr>
<td>( -A )</td>
<td>system's coefficient matrix</td>
</tr>
<tr>
<td>( \hat{A} )</td>
<td>model's coefficient matrix</td>
</tr>
<tr>
<td>( \Delta x_0 )</td>
<td>initial error state vector</td>
</tr>
<tr>
<td>( (\ )^T )</td>
<td>transpose of a matrix</td>
</tr>
<tr>
<td>( i(t) )</td>
<td>error input excitation function</td>
</tr>
<tr>
<td>( \alpha_i )</td>
<td>model's characteristic polynomial of ( s^i )</td>
</tr>
<tr>
<td>( \beta_i )</td>
<td>model's transfer function numerator polynomial coefficient of ( s^i )</td>
</tr>
<tr>
<td>( p )</td>
<td>indicates a functional relationship of a particular set of free parameters</td>
</tr>
<tr>
<td>( G(s) )</td>
<td>the closed loop transfer function of a system</td>
</tr>
<tr>
<td>( \hat{G}(s) )</td>
<td>the closed loop transfer function of a model</td>
</tr>
<tr>
<td>( \tilde{G}(s) )</td>
<td>a modified transfer function in Palsson's approach</td>
</tr>
<tr>
<td>LL</td>
<td>a particular value of lower limit</td>
</tr>
<tr>
<td>( A(t) )</td>
<td>area under ( d(t) ) curve</td>
</tr>
</tbody>
</table>
CHAPTER ONE

INTRODUCTION

1.1 Historical Summary of Problem Area

Wiener (1) and Kolmogorff (2) are the first formulation of filter design as an optimization problem. Their problem is referred to as the "free-configuration" that selects the filter from the class of all possible linear filters that minimizes the mean squared error between the actual and desired signals. Later Hall (3) and Phillips (4) formulated the "fixed-configurations," using a mean squared error Performance Index. In designing, the left the characteristic frequencies, time constants and gains unspecified, to be determined by the optimization process. Phillips derived a procedure for evaluating the mean squared error over an infinite time interval as an explicit nonlinear function of the free design parameters. The only work is to select the free parameter values that correspond to a minimum point of the function.

Hall and Phillips's method was that the resulting design could allow excessive signal magnitudes within the system that may exceed the range of assumed linearity or saturate. Newton, Gould and Kaiser (5) proposed constraining any signal magnitude by adjoining the mean squared value of that signal to the original Performance Index by a Lagrange Multiplier. The
augmented Performance Index can be evaluated as a function of the free design parameters and Lagrange Multiplier using tabulated integrals. The design selected is the one in which the values of the free parameters minimize this augmented Performance Index while requiring the signal magnitude constraint to be satisfied.

Aizerman (6) proposed a new concept for representing the desired system response within a Performance Index. Rather than using the squared error between the desired and actual response, Aizerman used a linear combination of the square of the actual transient response and its derivatives. The relative weighting of these squared variables in the Performance Index was chosen so that the absolute minimum value of the Performance Index would correspond to a system design with a transient response identical to the desired response. In general, the absolute minimum value can be obtained if one has complete freedom in selecting the closed-loop system design. Since the feedback configuration and design parameter values are usually constrained due to practical requirements, it is not usually possible to obtain the desired response identically. However, minimizing Aizerman's Performance Index would tend to force the system's response to be similar to the desired response, at least for a certain class of system and types of desired response. Aizerman's concept could provide a significant computational advantage in the optimization process over model-referenced performance indices in that the
model's response never enters the computational problem. Aizerman's Performance Index is rather limited in application but represented a distinctly different philosophy in analytical design.

1.2 Previous Work done by Rediess and Palsson

In 1968, Rediess (7) developed a new concept for Model Performance Index. It is based on a geometrical representation for matching a dynamical system or the model by another actual system. In other words, it is an interpretation of the relationship between the transfer function representation of a system and the geometrical representation by its characteristic plane and Pseudo Initial Condition vector. The basic form of the resulting "Model Performance Index" is the same as that of quadratic function frequently appearing in modern control theory. When the Model Performance Index is minimized, the system response becomes closer to or identical to the specified model response. By response is meant not only the displacement response but also the velocity, acceleration, etc. Since the Model Performance Index not only weights the displacement but also their higher derivatives, because the zeros in the system transfer function effect the error response, Rediess added a quadratic term in the initial error state to the Performance Index.

By choosing models whose excess of polies over zeros is no greater than that of the system, Palsson (8) apparently
eliminated the need for initial errors in the Model Performance Index. He also defined the "error excitation source" in both state space and transfer function representation, and found that MPI is the infinite integration of squaring this scalar error excitation source.

Chang Ho, in 1976, developed the concept of the lower limit which consists in beginning the integration in an instant larger than zero. This formulation allows the selection of a set of parameters with good conditions of matching but it is a completely arbitrary process.

1.3 Purpose of this Work

The purpose of this work was to obtain a way in which one could select the lower limit value of Ho's formulation without the present trial and error procedure. Besides, we should establish a way of quantifying the quality of matching, which is an aspect not treated in previous works. During this analysis, we were able to develop a functional relationship of the states which led to a new performance index which proved to have very definite advantages over the previous formulations.
CHAPTER TWO

REVIEW AND ANALYSIS OF REDIESS, PALSSON AND HO'S WORKS

In order to allow a better understanding of the problem in hand, of the difficulties found and of the different formulations tried for its solution, we are going to follow briefly the works done by Rediess, Palsson and Ho. Where necessary, specific examples will be presented and comments added in a suitable way to clarify the reader.

2.1 Rediess Formulation

Herman Rediess (ref. 7), in 1968, based in a geometric "criterium" for the approximation of a dynamic system (the model) by another (the actual system), developed a new performance index which allows a direct interpretation in terms of a certain model of the desired response. This index has received the title "Model Performance Index" or "Model P.I."

Let a system be described by the differential equation below

\[ y^n(t) + a_{n-1}y^{n-1}(t) + \ldots + a_1y'(t) + a_0y(t) = b_mu^m(t) + \ldots + b_1u(t) + b_0u(t) \]  

(2-1)

where: \( y \) is the output

\( u \) is the input

\( m \leq n-1 \)

initial condition assumed to be zero
The corresponding transfer function is

$$y(s) = \frac{b_m s^m + \ldots + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + \ldots + a_2 s^2 + a_1 s + a_0} \quad (2-2)$$

If we assume the existence of a finite steady state value for the output, \(y_{ss}\), in response to a step input, one can write the transient portion of the response as

$$x(t) = y(t) - y_{ss}.$$ 

The laplace transform of the transient response can now be obtained from (2-2), in the form below:

$$x(s) = \frac{b_m s^{(m-1)} + \ldots + b_3 s^2 + b_2 s + b_1}{s^{(n)} + a_{(n-1)} s^{(n-1)} + \ldots + a_2 s^2 + a_1 s + a_0} + \frac{b_0}{a_0} \left( \frac{s^{(n-1)} + a_{n-1} s^{(n-2)} + \ldots + a_2 s + a_1}{s^n + a_{n-1} s^{n-1} + \ldots + a_2 s^2 + a_1 s + a_0} \right) \quad (2-4)$$

Defining a set of hypothetical initial conditions here called "pseudo initial conditions," capable of producing a response identical to that of a unit step, made possible the inclusion of the zeros of the transfer function in the index to be developed. Therefore, the transient response become described by the homogeneous differential equation

$$x^{(n)}(t) + a_{n-1} x^{(n-1)}(t) + \ldots + a_1 \dot{x}(t) + a_0 x(t) = 0 \quad (2-5)$$

subject to the pseudo initial conditions.
\begin{align*}
  x(0) &= x_0 \\
  x'(0) &= x'_0 \\
  &\vdots \\
  x(n) &= x_n
\end{align*}

These initial values are then determined in a way that assures the already mentioned equivalence and are given by

\[
x_0 = -\frac{b_0}{a_0} \quad \text{for } i > m
\]

\[
x_0 = 0 \\
  b_1 - \sum_{j=n-m}^{n-i-1} a_j x_j^{(j)} \quad \text{for } i = 1, 2, \ldots, m
\]  

(2.6)

In this expression, the summation is zero when \( n-m < i-1 \).

The form of equation (2.5) suggested an extention of the state space in one dimension to \((n+1)\) order. In this space, referred as the "extended state space," the equation might be interpreted as defining a hyper-plane perpendicular to the vector \( \tilde{a} \), that's to say

\[
\tilde{x} \quad \tilde{a} = 0
\]

where \( \tilde{x} = (x, x', \ldots, x^{(n-2)}, x^{(n-1)}, x^{(n)}) \)

(2.7)

\[
\tilde{a} = (a_0, a_1, \ldots, a_{n-2}, a_{n-1}, 1)
\]

The trajectory of \( \tilde{x} \) as a function of time, which is the same as the transient response and its first \( n \) derivatives, is completely contained in this hyper-plane. Any other system which might generate a trajectory also contained in this hyper-plane, would differ from the first one only by the pseudo initial
conditions. We then conclude that the hyper-plane contains the trajectories of all systems possessing the same characteristics equation. In other words, a linear invariant system is completely described by this plane and corresponding set of pseudo initial conditions. Rediess called this plane the "characteristic plane."

We must stress that the representation of a system by its characteristic plane and pseudo initial conditions does not bring any new information from the mathematical point of view. However, it brings a very useful way for visualizing the process of approximation of a system by another, using the Model Performance Index.

If both system and model could be represented by its characteristic planes and pseudo initial conditions, it would then be possible to establish a geometrical "criterium" for evaluation of its approximation. The Model Performance Index developed by Rediess is one of these criteria. Its basic form is a generalized measure of the distance between system's trajectory and model's characteristic plane, as can be seen below.

\[
\text{MPI} = \int_0^\infty \frac{\| \ddot{X}(t) \|^2}{\| \ddot{Z} \|^2} \, dt
\]  

(2.8)

where: \( \ddot{z} = (\alpha_0, \alpha_1, \alpha_2, ..., \alpha_{n-2}, \alpha_{n-1}) \) and refers to the characteristic plane of the model. The subscript ' means a transpose matrix or vector.
The vector \( \tilde{a} \) is a function of design parameters and the equation (2.5) allows one to determine for each set of these parameters, the vector \( \tilde{x}(t) \).

In the case in which system and model do not have zeroes in their transfer function, Rediess proved that a necessary and sufficient condition for a perfect matching of the trajectories was the perfect matching of the characteristic planes. However, the functional relationship \( \tilde{a} = \tilde{a}(p) \) in most cases rules out the possibility of a perfect matching of trajectories. In this case, however, it is possible to consider that the best matching condition corresponds to the closest proximity of system's trajectory and model's characteristic plane, which means the lowest possible value for Model Performance Index.

In the case in which zeroes do show up in the transfer function, Rediess realized that, to obtain a correct solution, it would be necessary to add to the performance index a term representing the difference between the pseudo initial conditions of model and system. Without this correction, we might get very close characteristic planes but with trajectories well apart. With this modification, the performance index becomes:

\[
\text{MPI} = r \left\| \tilde{x}_o - \tilde{x}_{mo} \right\|_W^2 + \int_0^\infty \frac{||\tilde{x}'(t)\tilde{x}||}{||\tilde{x}||} \, dt
\]

(2.9)

Where: \( || \tilde{x} ||_W \) indicates here \((\tilde{x}_o - \tilde{x}_{mo})W(\tilde{x}_o - \tilde{x}_{mo})\)
and \( \hat{W} \) is a square, positive matrix \[
\begin{bmatrix}
I & 0 \\
0 & I
\end{bmatrix}
\]
of order \( l \) (\( l \leq n \)).

In this expression, the scalar quantity 'r' must be selected by the designer in order to establish the relative weight between the matching of the pseudo initial conditions and of the characteristic planes.

This last comment should be examined with care. Would it be enough to make 'r' equal to 1 in (2.9)? The experience has shown that this is not enough and, to make some points clear, we reproduce here the example 3-3 of ref. (7).

Example

Consider the system shown in the figure below, where \( k_1 \) and \( k_2 \) are free design parameters. The objective here is to select \( k_1 \) and \( k_2 \) in a way that the closed-loop step response become close to that of the model.
The autonomous representation of the transient response in closed loop form is
\[ x + \frac{k_1 + k_2}{1 + k_2} x + \frac{k_1}{1 + k_2} x = 0 \]

The corresponding model performance index is
\[ MPI = r \left( \frac{k_1}{1 + k_2} - \sqrt{2} \right)^2 + \int_0^\infty \left( x + 2 \dot{x} + \ddot{x} \right)^2 \, dt \] (2.10)

The values of the free parameters which minimize the MPI are presented, for two values of the scalar 'r', in Table 1 below:

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|c|c|}
\hline
r & $x_1$ & $x_2$ & $\tilde{a}$ & $\tilde{x}_0'$ \\
\hline
0.0025 & 1.59 & 0.48 & 1.08/ 1.40/ 1.0 & -1.0/ 1.08/ -0.45 \\
0.25 & 1.82 & 0.32 & 1.38/ 1.60/ 1.0 & -1.0/ 1.38/ -0.86 \\
\hline
\end{tabular}
\end{table}

Figures (1) and (2) reproduce the solutions for these two values of 'r', respectively.

Some comments
a) The reader probably has noted that Rediess included in (2.10) only the initial value of the first order state. This was done based on the fact that W was a square matrix of order '1' (2 in this case). Apparently Rediess did not pay attention to the fact that we are dealing with pseudo initial conditions and the second order states must be accounted for. In other
words, $W$ should be a $1 \times 1$ matrix. In this particular example, the second order states are different.

b) We are going to use, as an ultimate measure of quality of matching, the instantaneous distance between trajectories, here referred as $d(t)$. Table 2 below summarizes these results, for $t=0$.

TABLE 2

<table>
<thead>
<tr>
<th>States</th>
<th>Model System ($r = 0.0025$) $d(t)$</th>
<th>comp. System ($r = 0.25$) $d(t)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_o$</td>
<td>-1.0</td>
<td>-1.0</td>
</tr>
<tr>
<td>$\dot{x}_o$</td>
<td>1.41</td>
<td>1.08</td>
</tr>
<tr>
<td>$\ddot{x}_o$</td>
<td>1.08</td>
<td>0.33</td>
</tr>
<tr>
<td>$\dddot{x}_o$</td>
<td>-1.0</td>
<td>-0.55</td>
</tr>
</tbody>
</table>

Notice that $d(0)$ is 0.6414 and 0.1432 for $r$ 0.0025 and 0.25, respectively. These results confirm, now based on the new tri-dimensional figure of merit, that $r = 0.25$ gives the best result, at least for the set of values under scrutiny.

c) If we observe the behavior of the integral term in (2.10) for $t=0$, we would see that this term is in fact introducing an excessive weight for the PIC values in MPI calculations. Being more specific, the 0th order state which, by any logical reasoning, ought not contribute to the index (at $t=0$), is being fully charged (we recall that, at $t=0$, $x_o$ and $x_{mo}$ are equal). And more, the first order state receives a weight of 1.4142 which in terms of our figure of merit is distorting the picture of $d(0)$.

These facts suggest that the scalar '$r$' is effectively
releasing the weight unduly introduced through the integral term. This conclusion is important and, as will be shown during the present work, has its counterpart in the 'lower limit' mechanism introduced by Ho in Palsson's formulation.

In order to summarize this presentation, we can say that Rediess' method works pretty well in the case in which there are no zeroes in the transfer functions. When these zeroes show up, the expression (2.9) of MPI can be applied still with success, but the value of the scalar 'r' is selected by trial and error, which is a drawback.

2.2 Palsson's Formulation

In 1971, Thorgeir Palsson (ref. 8) introduced his concept of system's error equation. Through the representation of the transient response by the matrix equations below

\[ \dot{x} = Ax \]
\[ x(0) = x_0 \]  
(2.11)

and the desired response by the equations

\[ \dot{\hat{x}} = \hat{A} \hat{x} \]
\[ \hat{x}(0) = \hat{x}_0 \]  
(2.12)

Palsson defined the error equation in the following form

\[ \Delta \dot{x} = \hat{A} \Delta x + \Delta \hat{A} x \]
\[ \Delta x(0) = \Delta x_0 \]  
(2.13)

where: \[ \Delta x = x - \hat{x} \]
\[ \Delta x_0 = x_0 - \hat{x}_0 \]
\[ \Delta \hat{A} = \hat{A} - \hat{\hat{A}} \]
This last equation is valid only for the case where system and model are of same order.

We observe that the homogeneous part of equation (2.13) is identical to the model's equation, with the state of the model replaced by the state of the error. The particular term is expressed only in terms of the states of the system and, being so, the response of the error might very well be imagined as shown in Fig. (3).

Even considering that, in practical situations, the model is of lower order than the system, no comments will be made here on the modifications necessary to take care of the case. Such details would lengthen without reason this work and, when needed, the very symbology will alert the reader for the proper case. We suggest here, to those interested, ref. (2).

The error state equation allows a very simple and objective interpretation of the Model Performance Index, as we will see in the sequel.

In the case of system without zeroes in the transfer function, we can see from expression (2.6) that all the states, except the 0th order, are null for t=0, since in this case m=0. Moreover, this state is equal to the system static sensitivity with a minus signal applied. Therefore, if we assume that system and model have the same value for static sensitivity, the initial state of the error will be zero. We recall that this assumption is reasonable since in most practical situations, the steady state output error due to a unit step
will be zero by a project requirement.

The error equation, written in detailed form, is

\[ \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \vdots \\ \Delta x_n \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & \ldots & 0 \\ 0 & 0 & 1 & 0 & \ldots & 0 \\ \vdots \\ -\alpha_0 & -\alpha_1 & \ldots & -\alpha_{n-1} \end{bmatrix} + \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \vdots \\ (\Delta x_n)(\alpha_0 - a_0) \ldots (\alpha_{n-1} - a_{n-1}) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \]

where: \( a^T = (a_0, a_1, \ldots, a_{n-1}) \)

\( \alpha^T = (\alpha_0, \alpha_1, \ldots, \alpha_{n-1}) \)

are the coefficients of the characteristic equations of system and model respectively.

There are two potential sources of excitation to the error, as can be seen in Fig. (3). First, any initial error will result in an error response. For the case in which the model and system do not have zeroes in the transfer functions, this effect does not need to be considered, since the initial state of the error is zero as already seen. The second source is represented by the scalar

\[ i(t) = \sum_{i=0}^{n-1} (\alpha_i - a_i) x_{i+1} = (\alpha - a)^T x \]

As the matrix of the model, A, is a constant, the response of the error can only be influenced if we change the sources of excitation of the error. It is clear, for instance, that the response of the error is zero for all values of \( t \) when \( i(t) \) is
identically null, since there is no other perturbation to the error equation in this case. Therefore, one obvious manner to reduce the error between the response of the system and the desired response would be a minimization of some measure of the excitation to the error equation which is a scalar when the equations are written in phase variable form.

One of these measures is

\[ J = \int_0^\infty x^T (\alpha - a)(\alpha - a)^T x \, dt \]  \hspace{1cm} (2.15)

It should be noticed that the Index is still a function of the free parameters, since \( a = a(p) \).

In the case in which \( l < n \), we can define \( \hat{\alpha}^T = (\alpha_0, \alpha_1, \alpha_2, \ldots, \alpha_{l-1}, 1, 0, \ldots, 0) \) in such a form that the Index will take the form below:

\[ J = \int_0^\infty x^T \hat{\alpha} \hat{\alpha}^T x \, dt \]  \hspace{1cm} (2.16)

For the case in which zeroes appear in the transfer functions, Palsson suggested an approach that would eliminate the need for including the error initial states in the performance index. He had previously proved the (in the simpler case) validity of the expression below.

\[ i(s) = \frac{G(s)}{(------- - 1) G(s)} u(s) \]  \hspace{1cm} (2.17)

Writing this expression explicitly for the case in which we have zeroes
i(s) = \left( \frac{s^{l+x} + s^{l-1} + \cdots + s^x + b_m s^m + \cdots + b_1 s + b_0}{\beta_k s^k + \cdots + \beta_1 s + \beta_0 \frac{1}{s^n + a_{n-l} s^{n-l} + \cdots + a_1 s + a_0}} \right) \cdot \left( \frac{1}{s^n + a_{n-l} s^{n-l} + \cdots + a_1 s + a_0} \right) (2.18)

If we define a new system whose transfer function contains the zeroes of the model as poles of the system (which means added to the regular poles of the system) in such a way that

\[ G(s) = \frac{\beta_0 (b_m s^m + b_1 s + b_0)}{\beta_k s^k + \cdots + \beta_1 s + \beta_0} \frac{1}{s^n + a_{n-l} s^{n-l} + \cdots + a_1 s + a_0} \] (2.19)

then it would be possible to write expression (2.18) in the same form we had for expression (2.17). Therefore, Palsson shows that it was possible to reach the same performance index as in expression (2.15), without having to add any term representing the initial states, as Rediess did.

To summarize, Palsson's method works pretty well in the case with no zeroes. In the other case, however, if the difference in the initial states was big enough, his mechanism did not work as he thought. It may, at most, be considered satisfactory if and only if there is no emphasis on the behavior around \( t=0 \). In other words, since we accept as unavoidable a certain difference during the initial portion of the transient response. In any way, as Palsson's formulation had some conceptual advantages and as its form really facilitates the computation of the performance index, the efforts then became focused in how to reach a weighting of the initial states similar to Rediess' method.
2.3 Ho's Formulation

In 1967, Chang Ho, after applying Palsson's method to a series of examples in which, due to the particular arrangement of zeroes of model and system, the error input excitation function presented a strong initial spike (Fig. 4), which contributed heavily to the performance index, suggested that this undesired behavior could be screened out the calculations if we began the integration in an instant larger than zero. This method would certainly produce lower performance indexes and it might very well allow the selection of a set of parameters with better conditions of matching. The experience confirmed this suggestion, but the process of choosing the lower limit is still a trial and error one, as in Rediess' formulation.

To summarize the discussion up to this point, we can say that the present difficulties in Model Performance Index Development can be traced to the existence of zeroes in the transfer functions. Both Rediess' method and Palsson's modified lower limit method are arbitrary, a fact that renders the minimization algorithm only partially useful. Both present limitations as follows:

a) Model and system must have the same static sensitivity, which is a reasonable hypothesis, as seen before.

b) The model should have an excess of poles over zeroes lower or at most equal to that of system.

This last restriction was raised with the idea that, in mathematical terms, there is no difference in which transfer
function is associated to the model and which is associated to the system.

The reader certainly has noticed the fact that we constantly used words as "good matching" when speaking about trajectory closeness. And certainly has noticed also that no effort was made to quantify such statements. The search of a function which could be used as a yardstick of the quality of matching is the departure point of the present work.
CHAPTER THREE

THEORETICAL DEVELOPMENT

The most important difficulty affecting Ho's formulation is the process of selecting the lower limit, which is entirely arbitrary. Palsson's performance index has no usefulness now that the index decreases monotonically as we increment the lower limit. It can be verified that this course of action can even lead to trajectory matchings which can be considered very bad. Example 3 (ref. 11), was solved for different values of lower limit. Table 3 below presents a summary of these results.

<table>
<thead>
<tr>
<th>Lower limit values</th>
<th>MPI</th>
<th>Matching quality evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>1.39876</td>
<td>Bad</td>
</tr>
<tr>
<td>0.01</td>
<td>0.556044</td>
<td>Bad</td>
</tr>
<tr>
<td>0.02</td>
<td>0.181143</td>
<td>Good</td>
</tr>
<tr>
<td>0.03</td>
<td>0.048856</td>
<td>Bad</td>
</tr>
<tr>
<td>0.05</td>
<td>0.040300</td>
<td>Bad</td>
</tr>
</tbody>
</table>

NOTE: Figures 5 and 6 present displacement and first derivative for this example. From these figures one could say that lower limit 0.02 certainly is better than the others but can not say a word to answer the question "Is it the best?" nor to try a classification of quality.

In the examples 2a and 2b of ref. (11), Ho tried to show that displacement token by itself was not enough to judge the merits of two values of MPI and matching conditions associated
with it. Ho plotted the first and second derivatives in order to have a better appreciation which, we must say, was very far from being a clear cut one.

Extending these ideas, we could say that as we have the trajectories in a three-dimensional extended state space (in this example), it would be interesting to see how these trajectories match one another directly in the extended space, instead of each state at a time.

Following this line of thought, it was proposed a plotting of the trajectories using a tri-dimensional plotter in the IPC (Calcomp), which is valid for any second order model and particularly suited for the analysis of Ho's example three. Some of the principal points will be aborded here in a brief manner.

Basically, a cataloged procedure allows the plotting (in Calcomp Plotter) of any bi-dimensional array, interpreting the value of each element of the array as the "Z" component and the subscripts as the components "X" and "Y." This procedure is particularly suited for plotting surfaces, the functional relationships of which could be considered as well behaved in the interval of definition of "X" and "Y." The plotter is an off-line device, which means that we cannot control the parameters of the plotting subroutine during the picture producing process.

In our specific case, in which "X," "Y" and "Z" correspond to \(x(t), \dot{x}(t)\) and \(\ddot{x}(t)\) respectively, due to the sinusoidal
behavior of these functions, we have some additional complications, as explained below:

a) as "x" and "y" are not monotonically increasing or decreasing functions of time, the program must provide a way of transforming these coordinates into an array. In other words, the program must sort them to be monotonically increasing or decreasing functions which can then be used by the plotter.

b) as "x" and "y" are interpreted as subscripts of an array, they must be converted into positive, integer quantities before they can be used by the plotter. This means a certain distortion in the picture which, in this case, did not change the qualitative result.

c) the plotting had to be done using the "histogram mode," since the "point to point mode" (an ideal option in plottings) would blur the picture. This means that we are going to see cylindric surfaces instead of curves in the extended state space, making the task of interpretation somewhat difficult. This difficulty has played an important role in the example under scrutiny, and we had to plot each curve at a time, superimposing the pictures afterwards in order to evaluate the results.

To summarize, in terms of usefulness as a project tool, this approach is limited to second order models, has low efficiency since the program must be adjusted to each case (we recall that some of the parameters in the plotting subroutine must be closely controlled), and the program is expensive even
number of points.

Now considering the results for this example, we can see from Fig. 7, 8, 9, 10, which represent the trajectories for values of lower limit equal to 0.00, 0.01, 0.02 and 0.03 that LL = 0.02, represents the better matching condition, at least for the set of values under scrutiny (we recall that this approach does not provide a method for finding "the best" curve). It is interesting to note that for LL = 0.00 the initial distance between trajectories is very small, while the whole system curve is very far from the model's one. If we observe the behavior for other values of LL, we can see that the initial distance is increasing as we increase LL and, at the same time, we obtain trajectories which are much closer on the whole. This fact is very important because it shows that the lower limit mechanism is different from Rediess' weighting term (we recall that in Rediess' formulation we made the initial distance close in order to get better matching condition).

Due to the difficulties found in the previous approach, we began to consider an alternate solution which consisted in a direct measurement of the distance between trajectories and plotting against time (two dimensions). This was done for the same set of lower limit values and the resultant curves are presented in Fig. 11.

There are some interesting points in this figure which deserve a comment:

a) As we increase the value of the lower limit, in all
curves (except the one for \( LL = 0.01 \)), the initial distance \( d(0) \) assume monotonically increasing values. This confirms that the initial states are not being made closer through the lower limit mechanism (in the form that the initial quadratic term of Rediess used to work); this unexpected result led us to the conclusion that the lower limit method has no correlation with Rediess' formulation.

b) As we increase the values of lower limit, we observe that all the curves cut the one corresponding to \( LL = 0.00 \) in an instant of time \( t = t \) which is each time lower. In other words, it is possible to develop a qualitative relationship as a function of lower limit in the following form:

\[
\tau(LL) \leq \tau(LL + 1)
\]

where \( LL \) is a general value for the lower limit. Besides, it can be seen that the curve corresponding to \((LL+1)\) is kept essentially below the one corresponding to \((LL)\), for all \( t > \tau \).

c) Once more we can see that there exists a trade-off between the behaviors in low and high frequency. This fact was indicated by the tri-dimensional plotting, as we have seen before. In other words, as we increase the value of lower limit, we observe that the state of the initial error becomes bigger and bigger, and, at the same time, a better matching condition becomes evident in the low frequency range. This trade-off suggests that some generalized measure of \( d(t) \) may go through a minimum while we change the value of the lower limit.
d) The curves corresponding to $LL = 0.00$ and $LL = 0.01$ present a very curious behavior in their initial portion -- they grow for a certain time, going even beyond $d(0)$. This fact suggests that Palsson's formulation would have been applying an excessive weight to the pseudo initial conditions (we recall that we reached a similar conclusion when we made the analysis of the integral term of Rediess). Then, the method of lower limit would be releasing this undesired situation and allowing the minimization algorithm to select a better matching condition.

e) It is not possible to tell, by inspection, that a particular curve represents a better matching condition than any other curve. That is to say, even having a function as $d(t)$, which was selected in rather logical grounds, as the ultimate measure of matching quality, it is necessary to find some other generalized function of $d(t)$ or some other function which can describe more effectively the quality of matching.

Following the line exposed in comments c) and e) above, we decided to try the area under the curve $d(t)$ as a generalized measure of matching quality. We calculated this function for the curves corresponding to $LL = 0.00, 0.01, 0.02, 0.03$ and $0.05$. The results are presented in Table 4 below. As can be seen, the area in fact goes through a minimum and this minimum corresponds to $LL = 0.02$, a condition better than any of the values under scrutiny, as we have seen before.
TABLE 4

Area under curve d(t) for some values of lower limit

<table>
<thead>
<tr>
<th>Lower limit</th>
<th>Area</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>3.331800</td>
</tr>
<tr>
<td>0.01</td>
<td>3.156600</td>
</tr>
<tr>
<td>0.02</td>
<td>1.058634</td>
</tr>
<tr>
<td>0.03</td>
<td>1.212800</td>
</tr>
<tr>
<td>0.05</td>
<td>2.078300</td>
</tr>
</tbody>
</table>

It should be noticed that we have only a small sample of the universe of possible values, a fact which keeps us from using the term "best possible condition," at least until now. It should be stated in this point that the function area under d(t) in fact should be submitted to a formal minimization algorithm. This task, however, can't be made within the time allocated to this work.

In any way, considering the actual matching condition for LL = 0.02, particularly in terms of the 2nd order state ("Z" coordinate in Fig. 9), it seems that we have still some margin for improvement. In order to explore this situation, we have chosen an inverted way, which consists in plotting A(t) against LL (supposing that a nice functional relationship really exist between them) (Fig. 12), guessing the LL value which corresponds to the lower value of A(t) and the final use of the present algorithm to verify the true value of A(t). These results are presented below.
TABLE 5
Verification of the behavior of A(t) for LL = 0.0225

<table>
<thead>
<tr>
<th>Lower limit</th>
<th>Area under d(t)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00</td>
<td>3.331800</td>
</tr>
<tr>
<td>0.01</td>
<td>3.156600</td>
</tr>
<tr>
<td>0.02</td>
<td>1.058634</td>
</tr>
<tr>
<td>0.0225</td>
<td>2.3890</td>
</tr>
<tr>
<td>0.03</td>
<td>1.212800</td>
</tr>
<tr>
<td>0.05</td>
<td>2.078300</td>
</tr>
</tbody>
</table>

This unexpected result deserves some comments:

a) It is clear that our previous assumption on the behavior of the pair A(t)-LL is wrong, that is to say, this functional relationship is far from being smooth, a fact which render our forecast useless.

b) It is very interesting to note that a small variation in LL had brought such a big departure from optimality. For comparison, we present below the values of the free parameters for LL = 0.02 and 0.0225:

<table>
<thead>
<tr>
<th>Parameter 1</th>
<th>Parameter 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>LL = 0.02</td>
<td>LL = 0.0225</td>
</tr>
<tr>
<td>0.39067</td>
<td>0.51603</td>
</tr>
<tr>
<td>0.79100</td>
<td>0.83061</td>
</tr>
</tbody>
</table>

At this point, as the parameters for LL = 0.225 were obtained using a different minimization algorithm, some extra runs were tried which have indicated 0.49699 and 0.76628 as the new parameters for LL = 0.02. These values are far from the previous ones but MPI(s) are quite the same. Within
the new situation, \( A(t) \) would then select the set correspond-
ing to \( LL = 0.03 \). This seems to show that there is no specific
correlation between \( A(t) \) and \( LL \). However, it does not render
the new function invalid -- it only stresses the necessity for
submitting \( A(t) \) to a minimization algorithm.

In another effort to check the capabilities of the new
function, we did a follow-up of an actual computer run (which
still uses Palsson's formulation through a minimization algo-
rithm), calculating \( A(t) \) for each set of parameters. The
results are presented below:

**TABLE 6**

<table>
<thead>
<tr>
<th>Parameter 1</th>
<th>Parameter 2</th>
<th>MPI value</th>
<th>Area under ( d(t) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.44056</td>
<td>0.86849</td>
<td>0.193162</td>
<td>2.1395</td>
</tr>
<tr>
<td>0.43596</td>
<td>0.83349</td>
<td>0.191412</td>
<td>2.1201</td>
</tr>
<tr>
<td>0.42387</td>
<td>0.83579</td>
<td>0.187253</td>
<td>2.0801</td>
</tr>
<tr>
<td>0.40862</td>
<td>0.81246</td>
<td>0.183488</td>
<td>2.0475</td>
</tr>
<tr>
<td>0.39384</td>
<td>0.79423</td>
<td>0.181372</td>
<td>1.0768</td>
</tr>
<tr>
<td>0.39067</td>
<td>0.79100</td>
<td>0.181143</td>
<td>1.0586</td>
</tr>
</tbody>
</table>

These results show that \( A(t) \) can in fact be used for
selecting directly the best set of free parameters, which in
turn means that the new function \( A(t) \) can be used to define a
new form of Model Performance Index.
CHAPTER FOUR

CONCLUSION

As a conclusion, it is perfectly safe to say that we developed a new performance index based in a generalized measure of the distance between the trajectories of a particular system and a chosen model, in the form below:

\[ \text{MPI} = \int_0^\infty \sqrt{(x(t) - x_{\text{mo}}(t))^2} \, dt \]  

(3.1)

In the case in which \( l < n \), or the model is of an order lower than that of the system, one must project the extended state space of the system over the one of the model, and expression (3.1) is still applicable.

The new formulation has some advantages, if compared to the previous work, as summarized below:

a) Eliminate, as seen, the necessity to use a trial and error method in order to include the effect of the states of the initial error which was a major drawback in the previous formulations.

b) As there is no difference, in mathematical terms, between model and system, we have raised the restriction about the excess of poles over zeroes, which was common to the other formulations. The artifice used until now to take care of the case in which model had a greater poles over zeroes excess consisted in an inversion of the roles played by the transfer
functions of model and system. When the system possessed right half plane zeroes, a case which is very common in lateral control systems of airplanes, this idea brought a divergent performance index (since \( i(p) \) would have right half plane poles) and, therefore, useless.
Figure 1 Geometrical Representation for Example 3.3
Figure 2 Geometrical Representation for Example 3.3 Concluded
Figure 3  System Error Response
Example 2-a
I.C. = 10.1734

Example 2-b
I.C. = 64.9199

Fig. 4  Error Input Excitation Function
HO 3 SYSTEM 0.00

VIEWING ANGLES: Θ = 60.00  PHI = 315.00

COORDINATE RANGES: X = 1 TO 120  BY 2
                    Y = 1 TO 60  BY 2
                    Z = -30 TO 100

Figure 7 HO-3 System 0.00
HO 3 SYSTEM 0.01

VIEWING ANGLES:  THETA = 60.00  :  PHI = 315.00

COORDINATE RANGES:  X = 1 TO 120  BY 2
                    Y = 1 TO 60  BY 2
                    Z = 30 TO 100

Figure 8  HO-3 System 0.01
HO 3 SYSTEM 0.02

VIEWING ANGLES: THETA = 60.00 PHI = 315.00
COORDINATE RANGES: X = 1 TO 120 BY 2
       Y = 1 TO 60 BY 2
       Z = -30 TO 100

Figure 9 HO-3 System 0.02
HO 3 MODEL

VIEWING ANGLES:  THETA = 60.00  PHI = 315.00
COORDINATE RANGES:  X = 1 TO 120 BY 2
                    Y = 1 TO 60  BY 2
                    Z = -30 TO 100

Figure 10 HO-3 Model
REFERENCES


