AIAA 80-0050R

Controllers for Aircraft Motion Simulators

William R. Sturgeon*
Systems Analysis and Control, Ridgecrest, Calif.

Two techniques applicable to the design of motion simulator controllers have been developed. The first, a linear optimal controller synthesized with a quadratic performance index, provides all of the response characteristics previously obtained by classical designs, and consists only of constant gain feedforward and feedback loops around the actuators. The resulting closed-loop system is, therefore, relatively simple as no additional filters or compensators are required. The second technique involves the use of accelerometers and rate gyros, located at the pilot station, to close outer feedback loops around the simulator. This causes the fidelity to be dependent on the feedback (sensors) rather than the simulator, and results in a lower sensitivity to parameter variations and nonlinearities. These techniques, when applied to a planar model of the Vertical Motion Simulator, which had the synergistic, overcontrolled, and nonlinear properties of the actual system, provided a controller with the desired response characteristics.

Nomenclature

A	= output error weighting matrix
B	= control weighting matrix
$(\bar{B})^{1/2}$	= Cholesky decomposition of B
C	= feedback gain matrix
C_c	= feedforward gain matrix of input commands
C_e	= feedback gain matrix of output errors
D	= state weighting matrix
F	= system matrix
G	= control distribution matrix
H	= output matrix
I	= identity matrix
J	= performance index
ℓ	= fixed length
M	= output error matrix of states
N	= output error matrix of controls
s	= Laplace operator
T	= time constant
u	= control vector
$w(t)$	= zero mean white noise
x	= state vector
y	= output vector
y_c	= input command vector
y_e	= washout error vector
y_y, y_z	= specific forces in simulator coordinate system
y_ϕ	= angular velocity
z_i	= displacement of ith actuator
Γ	= disturbance distribution matrix
ζ	= damping ratio
σ_i	= integral of ith actuator displacement
ϕ	= simulator roll angle
ω	= undamped natural frequency

Superscripts and Subscripts

$(\ )^T$	= transpose
$(\ )^{-1}$	= inverse
$(\dot{\ })$	= time derivative
$(\frown)$	= transformed
a	= augmented system
c	= commanded or input command model
cl	= closed-loop
m	= simulator model
s	= simulator

Presented as Paper 80-0050 at the AIAA 18th Aerospace Sciences Meeting, Pasadena, Calif., Jan. 14-16, 1980; submitted Jan. 17, 1980; revision received May 12, 1980. This paper is declared a work of the U.S. Government and therefore is in the public domain.

*Control System Engineer. Member AIAA.

Introduction

IN fixed-based flight simulators, the pilot manipulates a set of flight controls and receives visual and instrument cues corresponding to actual flight. Although these simulators are satisfactory for certain types of training or evaluations, it is sometimes desirable to provide a more realistic simulation by the addition of motion cues. Even though the resulting "moving base" or motion simulator offers advantages over fixed-base simulators, it does not have perfect fidelity. The major difficulty is that its motion must be limited. Traditionally, simulator controllers have consisted of washout filters, that transform the simulated vehicle motion into achievable simulator motion commands, lead compensators, and a servo actuator drive system that moves the cockpit. Although the operational requirements are considered in the design of the washout, lead, and servo drive systems, they are distinct systems and are traditionally designed separately. The washout filters take advantage of human sensory imperfections and attempt to maximize the apparent fidelity of the impressed motion cues while maintaining the simulator within its motion limits. The outputs of the washout filters are passed through lead compensators and the resulting signals are used as servo actuator commands. The lead compensators are used to extend the system bandwidth, when required, beyond that of the servo actuators.

The design of washout filters is quite complicated for simulators with many degrees-of-freedom (DOF) due to the coupling between translation and rotation. This coupling normally requires the use of coordinated washout such as developed by Schmidt and Conrad.[1,2] This method has been applied to simulators with one independent actuator for each DOF[1] and to synergistic systems,[3,4] that is, ones requiring a combination of actuator extensions to provide motion in any one DOF. Several efforts have been made to simplify and systematize the design of coordinated washout systems through the use of optimal control. Applications to systems having individual actuators for each DOF or high-speed servo actuators of much higher bandwidth than the simulated aircraft are presented in Refs. 5-8.

In the past, simulators typically had one independent servo actuator for each DOF of the cockpit. Recently, the hexapod motion system[9] has become popular. This system consists of a cockpit mounted on six linear hydraulic actuators that provide six DOF. If desired, the translational motion capability may be increased by mounting the hexapod assembly on a cart or carriage. This resulting system presents additional complication in the design of washout filters as it is synergistic and overcontrolled in the sense that there are more actuators than DOF of the cockpit. As a result, the design of a simulator controller is now substantially more difficult.

The objective of this research is to develop a relatively simple and systematic method of applying linear optimal control theory to the design of controllers for complex motion simulators. Specifically, the methods are to be applicable to synergistic, overcontrolled, and nonlinear simulators such as the carriage mounted hexapod system mentioned previously. The techniques developed complement and extend previous work in this area by considering the servo actuator dynamics in the controller design, and by using feedback from accelerometers and rate gyros located in the cockpit at the pilot station.

Optimal Controller Synthesis

Simulator Model

Assume that the simulator actuators are operated as independent position servos, for which the low-frequency dynamics are adequately represented by a second-order transfer function:

$$\frac{z(s)}{z_c(s)} = \frac{\omega^2}{s^2+2\zeta\omega s+\omega^2} \tag{1}$$

A simulator model is obtained by combining the transfer functions of all the actuators and expressing the resulting equations in state variable form:

$$\dot{x}_m = Fx_m + Gu_m \quad \text{(dynamics)} \tag{2a}$$

$$y_m = H_{1m}x_m + H_{2m}\dot{x}_m \quad \text{(kinematics)} \tag{2b}$$

where

x_m = actuator displacements and velocities
u_m = actuator displacement commands
y_m = specific forces and angular velocities in the cockpit body axis system

This model is formed such that only the actuator dynamics are contained in the state equations, and the kinematics, which relate actuator motion to motion in the cockpit body axes, are contained in the output equations. Since the highest derivative of actuator displacement in the state vector is velocity, the outputs y_m, that contain acceleration information, are functions of both the state vector and its derivative.

Since the input commands, that is, the specific forces and angular velocities of the simulated aircraft, are functions of the pilot's control motions, aircraft dynamics, and wind gust dynamics and are not known a priori, they are modeled as a stochastic process. Selection of the stochastic models may be approached on at least two levels as discussed in Ref. 5. One approach is to derive an approximate model from the power spectral density of flight data obtained for a specific task. A second, and significantly more complicated approach is to derive a complete model that accounts for the aircraft, pilot, and wind gust dynamics. In any event, obtaining a valid input model is a significant problem that is not addressed here.

For illustrative purposes, it was assumed that the input commands were represented as outputs of linear first-order systems, driven by white noise. Thus, the input model is

$$\dot{y}_c = F_c y_c + \Gamma_c w(t) \tag{3}$$

where

y_c = input command vector, specific forces and angular velocities in body axis system of simulated aircraft
$w(t)$ = zero mean white noise

The simulator model is augmented with the input model and a controller is designed for the resulting system. The augmented system is

$$\dot{x}_a = F_a x_a + G_a u_m + \Gamma_a w(t) \tag{4}$$

where

$$x_a = [x_m^T y_c^T]^T \qquad F_a = \begin{bmatrix} F & 0 \\ 0 & F_c \end{bmatrix}$$

$$G_a = [G^T 0^T]^T \qquad \Gamma_a = [0^T \Gamma_c^T]^T$$

An important measure of the augmented system performance is the difference between the outputs of the simulator and those of the simulated aircraft. This difference is the motion eliminated through washout effects and is referred to as washout error. This error is defined as

$$y_e = y_m - y_c \tag{5}$$

and is obtained by substituting Eqs. (2-4) into Eq. (5) to yield

$$\begin{aligned} y_e &= [(H_{1m} \ -I) + (H_{2m} \ 0)F_a]x_a + (H_{2m} \ 0)G_a u_m \\ &= Mx_a + Nu_m \end{aligned} \tag{6}$$

Performance Index

A logical approach to the design of a system that should exhibit both washout and following characteristics is to minimize a performance index (PI) of the form

$$J = \frac{1}{2}\int_0^\infty (y_e^T A y_e + x_a^T D x_a + u_m^T B u_m)\,dt \tag{7}$$

It can be shown[10] that the problem of minimizing Eq. (7) given Eqs. (4) and (6) is equivalent to minimizing

$$J = \frac{1}{2}\int_0^\infty (x_a^T \bar{A} x_a + \bar{u}^T I \bar{u})\,dt \tag{8a}$$

given

$$\dot{x}_a = \bar{F}x_a + \bar{G}\bar{u} \tag{8b}$$

where

$$\bar{F} = F_a - G_a\bar{B}^{-1}N^T AM \qquad \bar{A} = M^T(A - AN\bar{B}^{-1}N^T A)M + D$$

$$\bar{B} = B + N^T AN \qquad \bar{G} = G_a(\bar{B}^{1/2})^{-1}$$

and

$$\bar{u} = (\bar{B})^{1/2}(u_m + \bar{B}^{-1}N^T AMx_a) \tag{9}$$

If the control that minimizes Eq. (8a) is

$$\bar{u} = \bar{C}x_a$$

then the control law for the original system, found using Eq. (9), is

$$u_m = ((\bar{B}^{1/2})^{-1}\bar{C} - \bar{B}^{-1}N^T AM)x_a = C_a x_a = [C \quad C_c]\begin{bmatrix} x_m \\ y_c \end{bmatrix} \tag{10}$$

This control law defines the actuator displacement commands u_m in terms of the simulator model states x_m and the input commands y_c. The simulator states, which are the actuator displacements and velocities, can be measured directly, and the input commands are obtained from the motion of the simulated aircraft.

Closed-Loop System Characteristics

Substitution of the control law, Eq. (10), into Eq. (2a) yields the closed-loop simulator model

$$\dot{x}_m = (F+GC)x_m + GC_c y_c = F_{cl}x_m + G_{cl}y_c \tag{11}$$

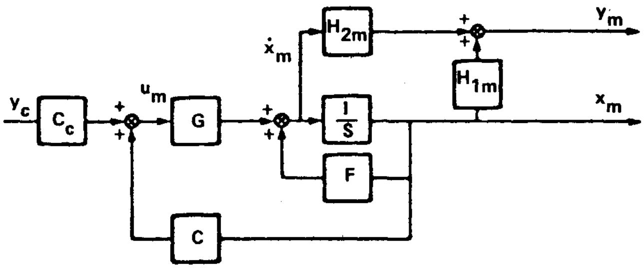

Fig. 1 Block diagram of simulator model with optimal controller.

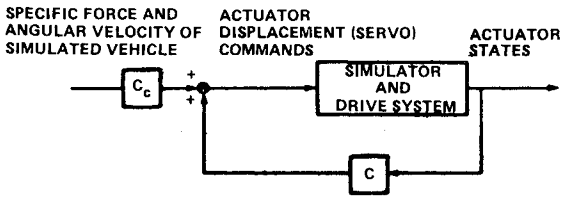

Fig. 2 Block diagram of optimal controller.

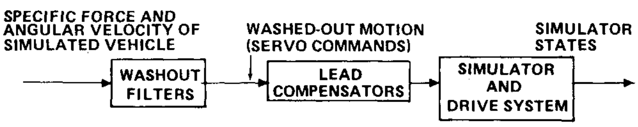

Fig. 3 Block diagram of classical controller.

A block diagram of the closed-loop simulator is shown in Fig. 1. Note that the input model has been eliminated, as it was only a design tool used to define the commands, and the system is driven directly by the input commands y_c.

An interesting property of the closed-loop system is that it exhibits both washout and following characteristics without the use of an explicit washout filter. This becomes apparent if Fig. 1 is simplified as shown in Fig. 2 and the latter compared with the classical controller shown in Fig. 3. The washout filter and lead compensator have been replaced by constant feedforward and feedback gain matrices.

Selection of Performance Index Weighting Matrices

The remaining step in the design of an optimal controller is the selection of the PI weighting matrices. This is normally an iterative process involving: 1) selection of initial weights, 2) solution for corresponding control gains, 3) evaluation of resulting closed-loop system response, and 4) making of appropriate changes in the weights. This process may be simplified by using previously established techniques and by considering the desired response characteristics of a motion simulator.

The first step is to determine which variables should, or must, be weighted in the PI [Eq. (7)]. All the controls must be weighted to ensure that the actuators are not commanded beyond their displacement limits. Note that it is unnecessary to weight the actuator displacements as they are governed by the displacement commands through the servo equations. (The opposite approach, of weighting just the actuator displacements, would allow unrealistically high velocities and accelerations to be obtained by commanding infinite actuator displacements over very short periods of time.) Although the actuator velocities are not weighted initially, they may be weighted later if it is determined that the velocity limits are reached under normal operating conditions. The washout error must be weighted to provide the system following characteristics. It is sufficient to weight only the controls and washout error, and not the states, as all system modes are observable with the washout error.

For simplicity, only diagonal weighting matrices are considered. The diagonal elements are initially selected using Bryson's Rule,[11] that is, the weight on a particular variable is the inverse square of the maximum permissible value of that variable. The corresponding controller gains can now be calculated using methods presented previously. Normally, at this point in the design, the closed-loop response is evaluated in order to determine appropriate changes in the weighting matrices, and the iterative process mentioned previously is followed. This process may be simplified by eliminating the need for the system response calculation, and using the system's "steady-state" response for preliminary evaluations. This requires an understanding of desirable response characteristics that make full use of the simulator's motion capability.

Due to the closed-loop system structure, the actuators have steady-state displacements in response to constant input commands y_c. These steady-state displacements and the corresponding outputs are found using Eqs. (2b) and (11) with $\dot{x}_m = 0$,

$$x_m = -F_{cl}^{-1} G_{cl} y_c \qquad y_m = H_{1m} x_m \tag{12}$$

The desired response to each individual input command is determined by considering classical washout logic, most importantly the idea of obtaining specific force components through the use of residual tilts. A detailed discussion of the desired responses is given in Ref. 10. It is assumed that the expected range of input commands, for a particular task, has "somehow" been related to a set of maximum constant values. Admittedly, constant inputs are unrealistic; however, the estimated values are only being used to aid in the design.

The design procedure, using the desired steady-state responses, is as follows: 1) determine initial weighting matrices using Bryson's Rule; 2) calculate corresponding gain matrices; 3) solve for steady-state response using Eq. (12) and the maximum expected constant inputs; 4) compare the actual and desired steady-state responses and change the weighting matrices accordingly (note that this only involves changes in weights on washout errors and actuator displacement commands); 5) repeat steps 2-4 until an acceptable steady-state response is obtained; 6) evaluate the system time response to determine required weights, if any, on the actuator velocities; and 7) repeat steps 2 and 6 until an acceptable time response is obtained. This procedure does not ensure that all design requirements will be satisfied; in cases involving translational-rotational coupling a compromise is usually required between the desired motions resulting from individual input commands. Also, in following this procedure, the designer should consider the effects of simultaneous inputs on the steady-state actuator displacements.

It should be noted that the steady-state displacements may be exceeded during the transient response, depending on the closed-loop pole and zero locations. However, a well-designed system should have minimal overshoot in response to a constant input.

Integral Control

If the closed-loop simulator is displaced from its nominal position at equilibrium, due to a constant input command, it is not able to take advantage of its full motion capability in response to subsequent commands. This undesirable characteristic may be modified by the application of integral control. The simulator states, presently actuator displacements and velocities, are augmented with the integral of the actuator displacements, that are then weighted in the PI [Eq. (7)] and fedback in Eq. (10). This results in a closed-loop system that reaches equilibrium at its nominal position (i.e., no steady-state displacements) when subjected to constant input commands. Note that the addition of integral control does not change the form of the previously derived control law.

Integral control should be implemented such that the simulator returns relatively slowly to its nominal position after an initial response similar to that of the original system without integral control. This relatively slow return motion, which should be below the pilot's detection thresholds, is accomplished by proper selection of the displacement integral weights in the PI. If the weights on the displacement integrals are significantly less than those on the displacement commands, then the integral control modes will be relatively very slow and, therefore, essentially decoupled from the remaining modes. As a result, the initial response is not significantly affected by the addition of integral control. Furthermore, since the initial response characteristics are determined by the PI weights selected for the original system, they should not require change due to the addition of integral control. The initial weights on the displacement integrals may be determined using Bryson's Rule or chosen to be several orders of magnitude less than the weights on the corresponding displacement. The system time response is then evaluated to determine required changes of the displacement integral weights.

Accelerometer and Rate Gyro Feedback

An optimal controller has been designed using a linear simulator model (Fig. 1). If the model is sufficiently accurate, the closed-loop response of the actual simulator will be essentially as desired; however, significant modeling errors may lead to unacceptable responses. Considering the hexapod motion system described previously, one such error results from the assumption that the outputs are linear functions of the actuator states and their derivatives, Eq. (2). Actually, the output equations are nonlinear due to the linkage kinematics. These equations must be linearized about the nominal actuator displacements in order to obtain the linear model required for the controller design. As a result, large excursions from the nominal displacements, which significantly change the output equations, introduce errors in the output. One approach to this problem is to calculate the instantaneous output matrices on-line and make compensating gain changes. This is not trivial for the spatial linkage of the hexapod system. Closed-form solutions are impractical, hence previous efforts have made use of numerical iterative techniques.[12]

An alternate approach is based on the use of accelerometers and rate gyros to measure the simulator outputs at the pilot station. The accelerometer and rate gyro outputs are compared with their desired values and the resulting *output errors* used to close a feedback loop around the nonlinearities. This loop attempts to compensate for the nonlinearities by reducing the output error, i.e., driving the simulator outputs toward their desired values. Because the system input commands are the accelerations and angular velocities of the simulated aircraft, there is a one-to-one correspondence with the output errors, and the two signals may be easily combined to close the outer loop. The desired values of the measured outputs are not available directly, but may be obtained from an on-line simulation of the linear closed-loop simulator model, Eq. (11). This model is driven by the same input commands as the actual simulator and therefore provides the desired outputs. A block diagram of the entire system is shown in Fig. 4.

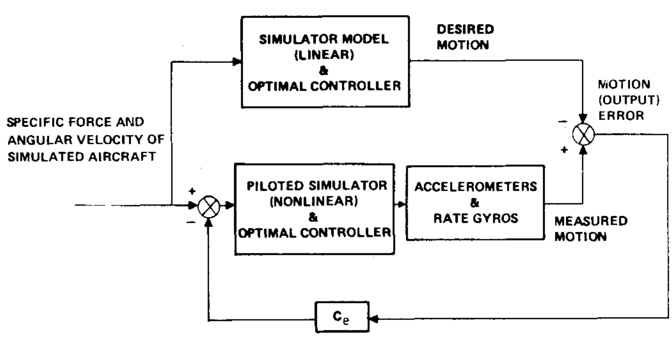

Fig. 4 Block diagram of accelerometer and rate gyro feedback.

The closed-loop equations for the entire model-simulator system are obtained as follows. Given the closed-loop simulator model, Eqs. (2b) and (11), and the simulator equations

$$\dot{x}_s = Fx_s + Gu_s$$
$$u_s = Cx_s + C_c[y_c - C_e(y_s - y_m)]$$
$$y_s = H_{1s}(x_s)x_s + H_{2s}(x_s)\dot{x}_s \tag{13}$$

substitute Eqs. (2b) and (11) into Eq. (13) and combine with Eq. (11) to obtain

$$\begin{bmatrix} \dot{x}_m \\ \dot{x}_s \end{bmatrix} = F_{ms} \begin{bmatrix} x_m \\ x_s \end{bmatrix} + G_{ms}y_c \tag{14}$$

where

$$F_{ms} = \begin{bmatrix} F_{cl} & 0 \\ K^{-1}G_{cl}C_e(H_{1m} + H_{2m}F_{cl}) & K^{-1}(F_{cl} - G_{cl}C_eH_{1s}) \end{bmatrix}$$

$$G_{ms} = \begin{bmatrix} G_{cl} \\ K^{-1}G_{cl}(I + C_eH_{2m}G_{cl}) \end{bmatrix}$$

$$K = I + G_{cl}C_eH_{2s}$$

and the notation associated with the output nonlinearities has been dropped.

In order to facilitate analysis, assume that the simulator output equations have been linearized about a set of actuator displacements other than those used to obtain the original linear model [Eqs. (2b) and (11)]. Since the simulator output matrices are now assumed constant, the entire model-simulator system can be analyzed as a linear system. Equation (14) indicates that x_s is a function of x_m but not vice versa; therefore the "separation of eigenvalues" property applies, and the closed-loop eigenvalues are those of the closed-loop simulator model, Eq. (11), and those of

$$[I + G_{cl}C_eH_{2s}]^{-1}[F_{cl} - G_{cl}C_eH_{1s}] \tag{15}$$

The addition of an on-line simulator model has doubled the closed-loop system order. Therefore, the system response is dependent not only on the poles and zeros of the model and optimal controller, but also on those resulting from the increased system order. These latter poles and zeros are the eigenvalues of Eq. (15) and the zeros of the transfer function matrix between the outputs y_s and inputs y_c. Using Eq. (14), this transfer function matrix is expressed as

$$y_s(s) = [[0 \quad H_{1s}][sI - F_{ms}]^{-1} + [0 \quad H_{2s}][F_{ms}(sI - F_{ms})^{-1} + I]]G_{ms}y_c(s) \tag{16}$$

Insight into the significance of the additional poles and zeros may be obtained through the analysis of simple single-DOF simulators, as presented in Ref. 10. Briefly, it was found that the dominant poles and zeros are those of the closed-loop model [Eq. (11)]. The remaining poles, that are of relatively low-frequency, are nearly cancelled by closed-loop zeros. It should be remembered that this analysis uses a linear approximation to the simulator output equations. The validity of this analysis is dependent upon the characteristics of the nonlinearity. Although the linear analysis is not exact, it does provide insight into the system operation.

Selection of the output error gain C_e results from a compromise between fidelity, implying high gains, and noise attenuation and stability, implying low gains.

Example: Vertical Motion Simulator

The techniques discussed previously are applied to a model of the NASA Ames Vertical Motion Simulator[9] (VMS) (Fig. 5). The VMS is composed of a hexapod motion system, called the Small-Motion Simulator (SMS), mounted on a carriage, called the Large-Motion Simulator (LMS), which increases the vertical and horizontal motion capability. A controller is designed and evaluated for a planar model of the VMS. This model (Fig. 6) has all the significant characteristics of the actual system in that it is synergistic, over-controlled, and nonlinear. The simulator model is obtained by combining the servo actuator models to form the state equations, and by linearizing the output equations about the nominal actuator displacements. This simulator model is augmented with the integrals of the actuator displacements (integral control) and with first order input models. Next, a PI [Eq. (7)] is used to design an optimal controller for the augmented system. Effects of the nonlinearities are determined by comparing the response of the linear and nonlinear closed-loop models. Feedback from accelerometers and a rate gyro is modeled to determine its effectiveness in compensating for the nonlinearities.

Actuator Models

All of the actuators are operated as independent position servos and use extensive compensation and multiloop feedback. This feedback significantly increases the system order and, therefore, makes the use of a complete model undesirable for use in initial controller design. A suitable low-frequency model is obtained by assuming that each servo provides consistent response to position commands. As a result, much of the high-frequency and actuator-load dynamics are neglected, and the servo transfer functions are obtained from the closed-loop frequency response.

The nominal servo characteristics for the SMS[13] and LMS actuators are, respectively,

$$\frac{z(s)}{z_c(s)} = \frac{100}{s^2+12s+100} \tag{17}$$

and

$$\frac{z(s)}{z_c(s)} = \frac{900}{s^2+42s+900} \tag{18}$$

The operating limits of each type of actuator are given in Table 1. An unusual aspect of this system is that the LMS has nearly equivalent acceleration and six times the velocity capability of the SMS. This is also reflected in the higher bandwidth of the LMS. Since the LMS has the larger displacement along with the higher bandwidth, the simple division of the desired motion into low-frequency high-amplitude and high-frequency low-amplitude commands to the LMS and SMS, respectively, is not appropriate.

An alternative approach is to use the SMS to provide only rotational motion and to use the LMS to provide all translational motion. Although this approach would simplify the control problem, it was considered too inefficient to have the LMS provide relatively small translations that could easily be supplied by the SMS.

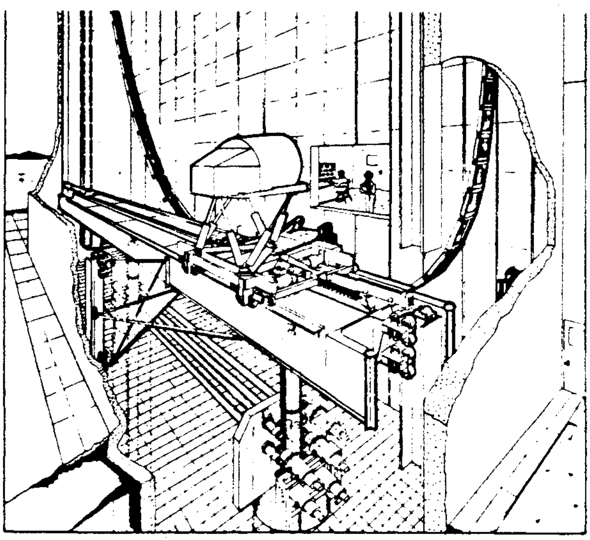

Fig. 5 Vertical motion simulator.

Simulator Model

A planar model of the VMS is shown in Fig. 6. The SMS actuator displacements, z_3, z_4, z_5, are by Eq. (17) and the

Table 1 Approximate actuator operating limits[a]

	SMS	LMS
Displacement	±0.5	±10.0 m
Velocity	0.5 m/s	3.0 m/s
Acceleration	10.5 m/s^2	8.0 m/s^2

[a] Note relatively large velocity and acceleration limits of LMS.

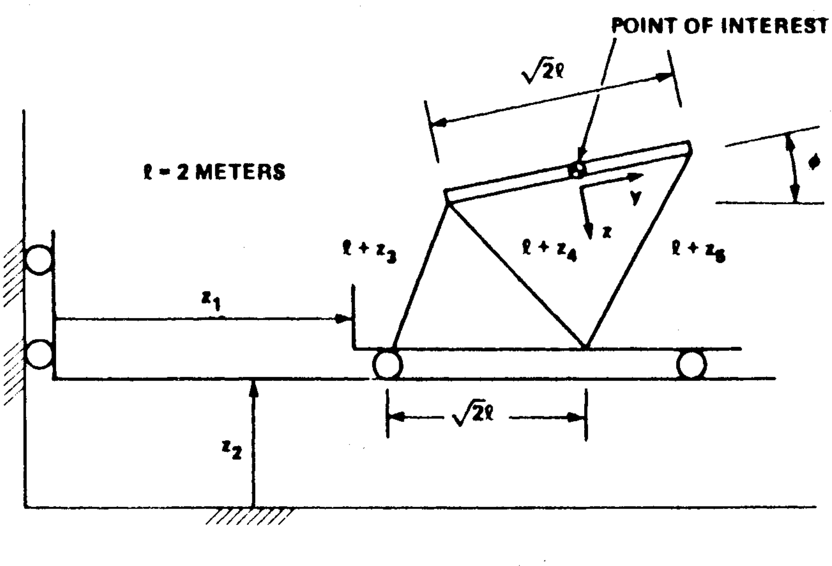

Fig. 6 Planar model of Vertical Motion Simulator.

LMS actuator displacements are governed by Eq. (18). The motion of the center of the rigid link on top of the SMS is of interest. The outputs are the specific forces and angular velocity at this point, expressed in a coordinate system fixed to the link. Since the SMS kinematics lead to nonlinear output equations, a controller is designed using nominal output matrices.

The equations of the augmented linearized system are

$$\dot{x}_a = F_a x_a + G_a u_m + \Gamma_a w(t) \qquad y_m = H_{1m} x_a + H_{2m} \dot{x}_a \tag{19}$$

and y_e is defined by Eq. (6), where

$$x_a = [z_1 \quad \dot{z}_1 \quad z_2 \quad \dot{z}_2 \quad z_3 \quad \dot{z}_3 \quad z_4 \quad \dot{z}_4 \quad z_5 \quad \dot{z}_5 \quad \sigma_1 \quad \sigma_2 \quad \sigma_3 \quad \sigma_4 \quad \sigma_5 \quad y_{cy} \quad y_{cz} \quad y_{c\phi}]^T$$

$$F_a = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-900 & -42 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -900 & -42 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -100 & -12 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -100 & -12 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -100 & -12 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -7 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -7 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -7
\end{bmatrix}$$

$$G_a = \begin{bmatrix}
0 & 900 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 900 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 100 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}^T$$

$$u_m = [z_{1c} \quad z_{2c} \quad z_{3c} \quad z_{4c} \quad z_{5c}]^T \qquad \Gamma_a = [0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 1 \; 1 \; 1]^T$$

$$y_m = [y_y \quad y_x \quad y_\phi]^T$$

$$H_{1m} = \begin{bmatrix}
0 & 0 & 0 & 0 & -5 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -0.5 & 0 & 0 & 0 & 0.5 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}$$

$$H_{2m} = \begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0.707 & 0 & -0.707 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0.707 & 0 & 0.707 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}$$

in consistent units of meters and seconds.

The open-loop eigenvalues (in rad/s) of Eq. (19) are

$$s_{1\text{-}4} = -21 \pm 21\,i \quad \text{(actuators-LMS)}$$

$$s_{5\text{-}10} = -6 \pm 8\,i \quad \text{(actuators-SMS)}$$

$$s_{11\text{-}15} = 0 \quad \text{(integral displacement modes)}$$

$$s_{16\text{-}18} = -7 \quad \text{(input model)}$$

where the subscripts indicate conjugate and/or repeated eigenvalues.

Optimal Controller

The selection of the weighting matrices is complicated slightly, as all SMS actuators respond to each of the three input commands. Use of the diagonal weighting matrices

$$A_{\text{diag}} = [20 \quad 20 \quad 500]$$

$$B_{\text{diag}} = [0.4 \quad 1 \quad 14.1 \quad 14.1 \quad 14.1]$$

$$D_{\text{diag}} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0.0004 \quad 0.001 \quad 0.0141 \quad 0.0141 \quad 0.0141 \quad 0 \quad 0 \quad 0]$$

in the PI, [Eq. (7)], yields a closed-loop model with the following eigenvalues (in rad/s)

$$\left.\begin{aligned} s_1 &= -420 \\ s_2 &= -0.65 \end{aligned}\right\} \quad \text{(SMS rotation)}$$

$$s_{3\text{-}6} = -6.1 \pm 8.1\,i \quad \} \quad \text{(SMS translation)}$$

$$\left.\begin{aligned} s_{7,8} &= -0.35 \pm 0.56\,i \\ s_{9,10} &= -0.33 \pm 0.33\,i \end{aligned}\right\} \quad \text{(LMS translation)}$$

$$\left.\begin{aligned} s_{11\text{-}14} &= -0.032 \\ s_{15} &= -0.0037 \end{aligned}\right\} \quad \text{(integral control modes)}$$

Note that the eigenvalues of the input model have been omitted.

Accelerometer and Rate Gyro Feedback

Since the optimal controller was derived using nominal (linearized) output matrices, and the simulator output equations are actually nonlinear (i.e., the output matrices are functions of actuator displacements, velocities, and accelerations), the output of the closed-loop simulator will differ from that of the closed-loop linear model. These differences may be reduced by feeding back the difference between the measured simulator outputs, obtained using accelerometers and rate gyros, and the outputs of an on-line linear model. The eigenvalues of this entire system are those of the closed-loop model, listed previously, and those of the linearized closed-loop simulator. These latter poles, that are dependent on the SMS actuator displacements, contribute little to the linearized system response because they are near closed-loop zeros [Eq. (16)]. Therefore, the dominant poles and zeros (Fig. 7) are those of the closed-loop model. A comparison of these poles with those of simpler examples[10] suggests that the complex poles are associated with the translational motion and the two relatively large real poles are associated with the rotational motion. The poles of the SMS, two complex pairs and a real pair (integral control), associated with translational motion are nearly cancelled by closed-loop zeros. This is due to the relatively large weights on the SMS controls. The remaining SMS poles, associated with rotational motion, are real as a result of commanding angular velocity. The remaining poles, two complex pairs and a real pair (integral control), are associated with the LMS translation.

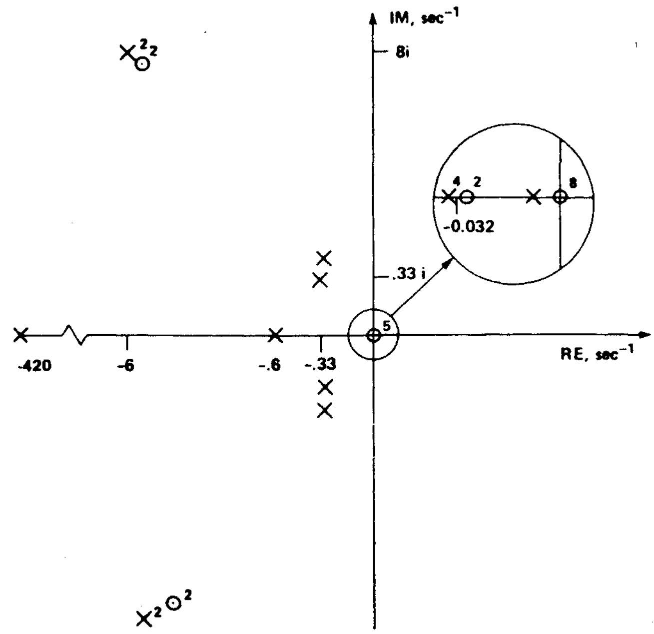

Fig. 7 Input-output transfer function roots of the linearized VMS planar model with optimal controller (superscripts indicate multiple poles or zeros).

The effects of changing the PI weights are complicated by the coupling between translation and rotation. For instance, increasing the weight on specific force error increases the residual tilt angle but decreases the angular displacement due to an angular velocity command because it results in a specific force error. Also, increasing the weight on angular velocity error increases the angular displacement, due to an angular velocity command, but decreases the residual tilt angle, because it is obtained through an angular velocity error. Changing the control weights produces similar coupling effects. The system design, therefore, results in a compromise between the responses to commands of specific force and angular velocity. In general, the controller design should be dependent on the simulated aircraft and on the task to be performed.

System Response

The response to a constant lateral specific force command is shown in Fig. 8 for three systems: the linear model, nonlinear model, and the nonlinear model with accelerometer and rate gyro feedback. The linear model [Eq. (19)], that was used in the controller design, provides the response of an ideal simulator. Since the simulator is nonlinear, its response differs from the ideal response. This difference, which is significant for the specific force, is reduced substantially by the use of accelerometer and rate gyro feedback. Note that it

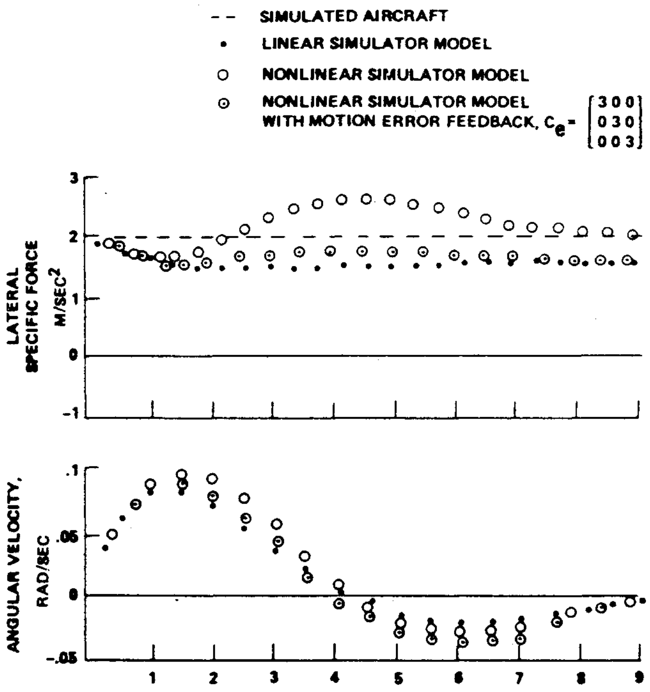

Fig. 8 Response of Vertical Motion Simulator planar model to constant lateral acceleration commands.

is not necessary for the simulator to follow the linear model outputs exactly, as the latter are obtained by washing out the input commands. The system operation should therefore be acceptable as long as the differences between the model and simulator outputs are small relative to those between the model outputs and input commands, and the noise attenuation is adequate. The final acceptability criterion is the fidelity required by the pilot for the specific task being simulated.

Conclusions

Two techniques applicable to the design of motion simulator controllers have been developed. The first is a linear optimal controller and the second involves the use of feedback from accelerometers and rate gyros located in the simulator cockpit.

The optimal controller offers two major advantages over classical designs:

1) The optimal controller, with integral control, provides the desired response characteristics, similar to those produced by classical designs, and consists only of constant gain feedforward and feedback loops around the actuators. The resulting closed-loop system is, therefore, relatively simple as no additional filters or compensators are required. This simplification becomes significant for systems with many degrees-of-freedom, such as the Vertical Motion Simulator.

2) The optimal controller results from a relatively simple design procedure, which consists of selecting performance index weights and solving for the resulting gains. The skill required of the designer is less dependent on the system complexity than with classical design, although experience is required to become proficient in relating the performance index weights to specifications.

This controller is designed using an input model whose outputs are assumed to describe motion of the simulated aircraft for a specific task. The simulator performance is, therefore, dependent on the accuracy of this model, and could be limited by the successive execution of significantly different tasks. However, this limitation is inherent in any linear washout scheme. Since only first-order input models have been considered, a logical extension is the consideration of higher order models. This will increase the system complexity, as the input model states or their estimates are required for feedback.

Feedback of the measured specific force and angular velocity of the pilot station causes the accuracy and fidelity to be dependent on the feedback (sensors) rather than the simulator characteristics, which results in a lower sensitivity to parameter variations and nonlinearities. Although the aforementioned method does not eliminate the effects of the nonlinearities, it does reduce them significantly and may eliminate the need for on-line calculations and gain changes, depending on the individual system and its specifications. This method is limited, as the gain selection results from a tradeoff between motion fidelity and adequate noise attenuation.

The foregoing techniques, when applied to a planar model of the Vertical Motion Simulator, which had the synergistic, overcontrolled, and nonlinear properties of the actual system, provided a controller with the desired response characteristics. It should be noted that these techniques are applicable to other motion simulators and are not limited to the type and configuration used in the example. Also, the performance index used in the optimal controller design is applicable to more general control problems, as it produces high-pass or washout effects, provides a method of weighting the time derivative of the system state, and accommodates systems with more controls than degrees-of-freedom or outputs to be controlled.

References

[1] Schmidt, S.F. and Conrad, B., "Motion Drive Signals for Piloted Flight Simulators," NASA CR-1601, May 1970.

[2] Schmidt, S.F. and Conrad, B., "A Study of Techniques for Calculating Motion Drive Signals for Flight Simulators," NASA CR-114345, July 1971.

[3] Parrish, R.V., Dieudonne, J.E., Martin, D.J. Jr., and Copeland, J.L., "Compensation Based on Linearized Analysis for a Six Degree of Freedom Motion Simulator,'" NASA TN D-7349, Nov. 1973.

[4] Parrish, R.V., Dieudonne, J.E., and Martin, D.J. Jr., "Motion Software for a Synergistic Six Degree of Freedom Motion Base," NASA TN D-7350, Dec. 1973.

[5] DeBra, D.B., and Kurosaki, M., Final Report on "Development of a Research Model of the Vertical Motion Simulator," submitted to NASA ARC, NSG-2178, Stanford University Guidance and Control Lab., Dec. 1978.

[6] Friedland, B. and Ling, C.K., "Quasi-Optimum Design of Control Systems for Moving Base Simulators," NASA CR-1613, Oct. 1970.

[7] Friedland, B., Ling, C.K., and Hutton, M.F., "Quasi-Optimum Design of a Six Degree of Freedom Moving Base Simulator Control System," NASA CR-2312, Oct. 1973.

[8] Kosut, R.L., "Nonlinear Optimal Cue-Shaping Filters for Motion Base Simulators," *Journal of Guidance and Control,* Vol. 2, Nov.-Dec. 1979, pp. 486-490.

[9] Dusterberry, J.C., "Vertical Motion Simulator," *Simulation,* Vol. 26, Feb. 1976, p. 64.

[10] Sturgeon, W.R., "Controllers for Complex Motion Simulators," Vol. 26, NASA TM 78529, Feb. 1979, p. 64.

[11] Bryson, A.E., "Control Theory for Random Systems," Invited Lecture, 13th International Congress of Theoretical and Applied Mechanics, Moscow, Aug. 1972.

[12] Dieudonne, J.E., Parrish, R.V., and Bardusch, R.E., "An Actuator Extension Transformation for a Motion Simulator and an Inverse Transformation Applying Newton-Raphson's Method," NASA TN D-7067, Nov. 1972.

[13] Howarth, B.A., "Design Study for the Compact Six Degrees of Freedom Motion System—Final Report," CAE Electronics Ltd., submitted to NASA Ames Research Center, Contract NAS2-8542, May 1975.