NASA CR-755

3 APPLICATIONS OF ARTIFICIAL INTELLIGENCE TECHNIQUES TO A SPACECRAFT CONTROL PROBLEM 6

By J. M. Mendel

Distribution of this report is provided in the interest of information exchange. Responsibility for the contents resides in the author or organization that prepared it.

Prepared under Contract No. NAS 12-23 by
DOUGLAS AIRCRAFT COMPANY, INC.
Santa Monica, Calif.

for Electronics Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

PREFACE

The work reported on herein was conducted under Contract NAS 12-23. The study director for NASA-ERC was F. Noonan, Control and Information Systems Laboratory, Electronics Research Center, Cambridge, Massachusetts. It was administered under the direction of R. W. Hallet, Jr., Director of Research and Development, Douglas Missile and Space Systems Division. Dr. J. M. Mendel was the principal investigator; J. J. Zapalac was the associate investigator; Dr. M. J. Abzug and S. Viglione were technical advisors. In addition, J. Feather contributed substantially to the study effort.

Under the contract, Douglas was to perform the following:

- Item 1--Identify areas on the basis of the existing state-of-the-art in artificial intelligence techniques, where new and novel design procedures for space vehicle control systems may lead to long lifetimes, invariance performance, and reduced dependence on critical elements.

- Item 2--Based on the needs identified under Item 1, undertake with concurrence of ERC technical monitor, preliminary designs and studies of control systems which may verify feasibility of the concepts.

- Item 3--Provide preliminary evaluation of such newly evolved concepts and system designs to determine specific performance specifications.

The work performed under Item 1 is separated from the work performed under Items 2 and 3, in this Report. Part 1 of this Report, entitled "Survey of Learning Control Systems for Space Vehicle Applications," summarizes the work performed under Item 1. Part 2, entitled "Fine Attitude Control of a Spacecraft Operating in a Partially Known Environment," summarizes the work performed under Items 2 and 3.

Part 1 of this Report was presented to the 1966 Joint Automatic Control Conference, University of Washington, Seattle, Washington, August 17-20, 1966.

ABSTRACT

The results of a 14 month study of the application of artificial intelligence techniques to a spacecraft control problem are presented. In the first part of the report, a survey of the present technology of self-organizing control systems and a discussion of potential space-vehicle applications for such systems are presented. Two types of self-organizing (learning) control systems are outlined: (1) on-line-learning control systems and (2) off-line-learning control systems. An extensive bibliography is included for topics related to self-organizing systems and for each of the two types of such systems mentioned above.

The second part of the report presents a preliminary design of a fine attitude, single-axis controller for an almost cylindrically symmetrical spacecraft that operates in a partially known environment. The design proceeds in two stages. First, a nominal controller is designed. All available information about the plant and the environment are incorporated into this design. The nominal controller is such that if the system and the on-line environment are as assumed off-line, the on-line attitude errors are contained to within ±0.20 arcsec (as required). The nominal controller is "nominal" with respect to the overall on-line controller, which is provided with a capability for updating the nominal controller, when and if updating is necessary.

The on-line controller designed during the second stage is an on-line learning controller. This controller utilizes the system's past experience in attempting to improve the system's present performance. Learning is accomplished, for the most part, through inclusion of a memory into the on-line controller.

The complete design embraces the concepts of stochastic-optimal control theory, combined state and parameter estimation, sensitivity analysis, control situations, on-line cost functions, on-line learning algorithms (error-correcting and reinforcing) and pattern recognition.

CONTENTS

PART 1

SURVEY OF LEARNING CONTROL SYSTEMS FOR SPACE VEHICLE APPLICATIONS

PRECEDING PAGE BLANK NOT FILMED.

PRECEDING PAGE BLANK NOT FILMED.

CONTENTS

Section 1
INTRODUCTION

Artificial intelligence techniques have recently been applied to the design of self-organizing control systems. Two types of such control systems have been postulated: (1) on-line-learning control systems, in which the self-organizing controller learns to control a system whose inputs and/or plant are incompletely specified and/or known; and (2) off-line learning control systems in which the controller learns to control a system whose actual control law is incompletely specified. Each of these systems possesses one or more of the following attributes, making it particularly attractive for space vehicle applications: (1) ability to maintain satisfactory performance in the face of random, unpredictable environments; (2) ability to prolong satisfactory performance in the face of hostile environments that cause progressive component failure; and (3) ability to provide opitmum control of complex plants for which present analytical methods are inadequate for deriving optimum control laws. Before discussing these particular control systems, a general treatment of self-organizing systems is given in the following paragraphs to permit discussion of learning control systems within a common frame of reference.

Section 2
SELF-ORGANIZING SYSTEMS

A self-organizing system is one that changes its basic structure as a function of its experience and/or environment. Its general aim is to evolve toward some desired output state or mode of behavior, in spite of some degree of ignorance of process, inputs, or controls. Since its structure changes as a function of experience, the self-organizing system can be said to "learn". This is consistent with most definitions of learning--provided that such a system improves its future performance by analyzing its past performance. A lucid discussion of "learning" as it is related to control systems is available in the literature (ref. 1).

A controller that is also a self-organizing system is called a "self-organizing controller." Such a controller contains three basic subsystems: (1) sensors, (2) learning network, and (3) a goal circuit (fig. 1).

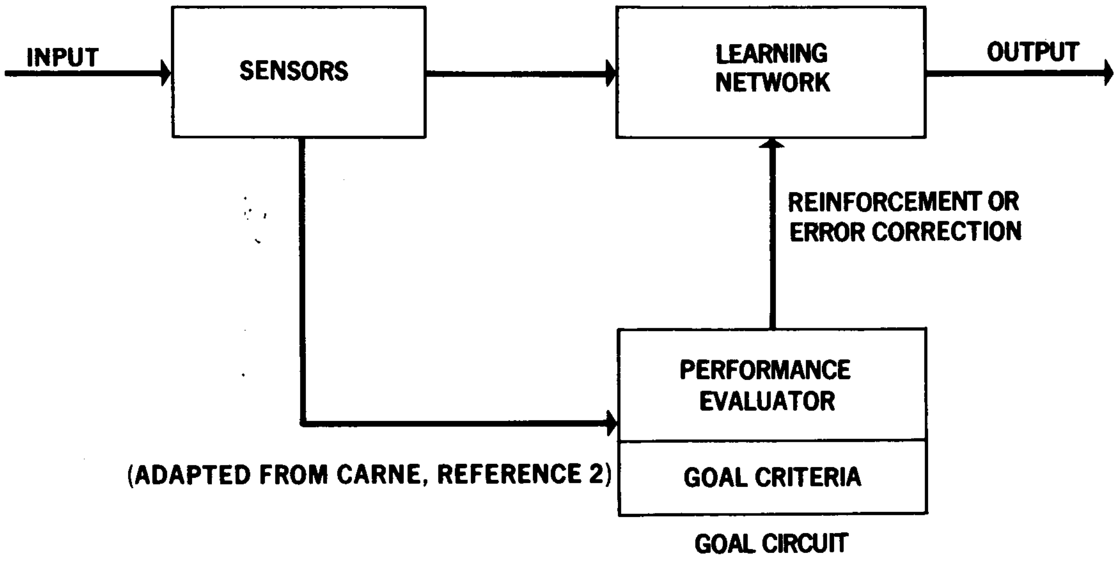


Figure 1. Subsystems Which Comprise a Self-Organizing Controller

The sensors (accelerometers, rate gyros, and horizon scanners) observe the local environment and provide descriptive data to the learning network and the goal circuit. The learning network consists of decision elements which operate on data input from the sensors and which render a desirable output response. Output data from the learning network are supplied to the system being controlled. The goal circuit directs the system organization toward a specific objective and provides information on the degree of success attained by each trial in terms of the specific objective. This is usually accomplished by one of the following techniques:

(1) Reinforcement. If present performance is an improvement upon recent past performance, the goal circuit generates a "reward" signal to the learning network, indicating that improvement has occurred. On the other hand, if present performance is worse than recent past performance, a "punishment" signal is generated, notifying the learning network of that fact. In effect, the reward signal reinforces those states of the network that contribute to improvement, while the punishment signal reverses the states that produced improper behavior.

(2) Error Correction. A signal is generated by the goal circuit only if present performance is worse than recent past performance. This signal reverses those states that produced improper behavior. Improved performance is not rewarded.

In the following paragraphs, a system with a self-organizing controller is referred to as a "self-organizing" or "learning control system."

Section 3

ON-LINE-LEARNING CONTROL SYSTEMS

An "on-line-learning" control system is one in which the inputs and plant may not be known a priori; the controller is self-organizing and learns to control the system properly on-line. A representative on-line-learning control system is shown in fig. 2.

Learning occurs with the self-organizing controller, $\mathcal{S}$ embedded in the control system during real-time operation of the overall system. Learning that occurs when the performance history of the overall system--over a sequence of trials--indicates a trend toward improved performance (ref. 1) automatically improves the control law through the following functions:

(1) Evaluation of results of control choices made by the self-organizing controller for a given situation and according to a prescribed criterion.

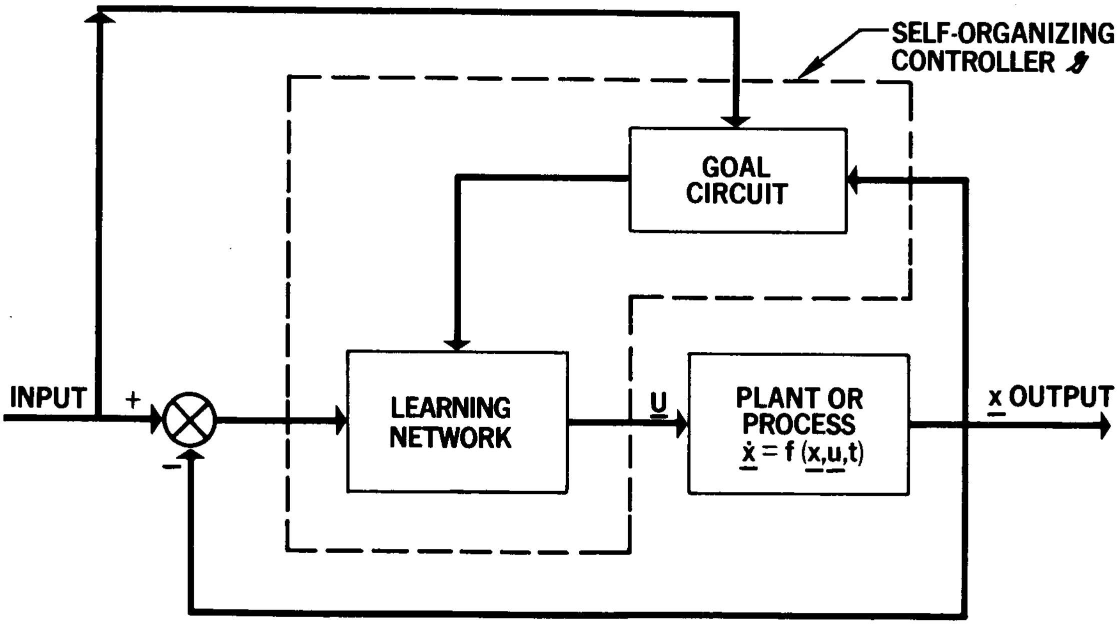

SELF-ORGANIZING CONTROLLER $\mathcal{S}$
GOAL CIRCUIT
INPUT
+
−
LEARNING NETWORK
$\underline{u}$
PLANT OR PROCESS
$\dot{\underline{x}} = f(\underline{x},\underline{u},t)$
$\underline{x}$ OUTPUT

Figure 2. Representative On-Line-Learning Ccntrol System

(2) Modification of the controller's memory store of parameters or its logic, so that subsequent control choices reflect the evaluation.

Two systems illustrate the on-line-learning philosophy. The first system, shown in fig. 3, is an adaptive type.

Parameters a and b are variable. Each contains two components, $a = \alpha + K_1$ and $b = \beta + K_2$; α and β represent random variations of a and b, respectively; and K_1 and K_2 represent the controls to offset the random variations in a and b, respectively. The random variations in a and b are assumed to change sufficiently often so that purely adaptive action could not optimize the system during the periods of constant α and β. Thus, a learning capability (fig. 4) was included (ref. 1).

The system shown in fig. 4 is subjected to a fixed-amplitude square-wave input, r(t), to facilitate the optimization of the performance index, PI, with respect to K_1 and K_2. Constants α and β are constrained to remain constant over two periods of the square-wave input; and at each occurrence of an (α, β) pair, four computations of the PI are carried out, one at each transition of the square wave. The PI is optimized on the computer. Specifically, after each computation of the PI, K_1, and K_2 are adjusted by a two-dimensional hill-climbing technique. The best current values of K_1 and K_2 for a given pair (α_i, β_i), the directions of K_1 and K_2 adjustment, and the best current value of PI are stored in the computer memory.

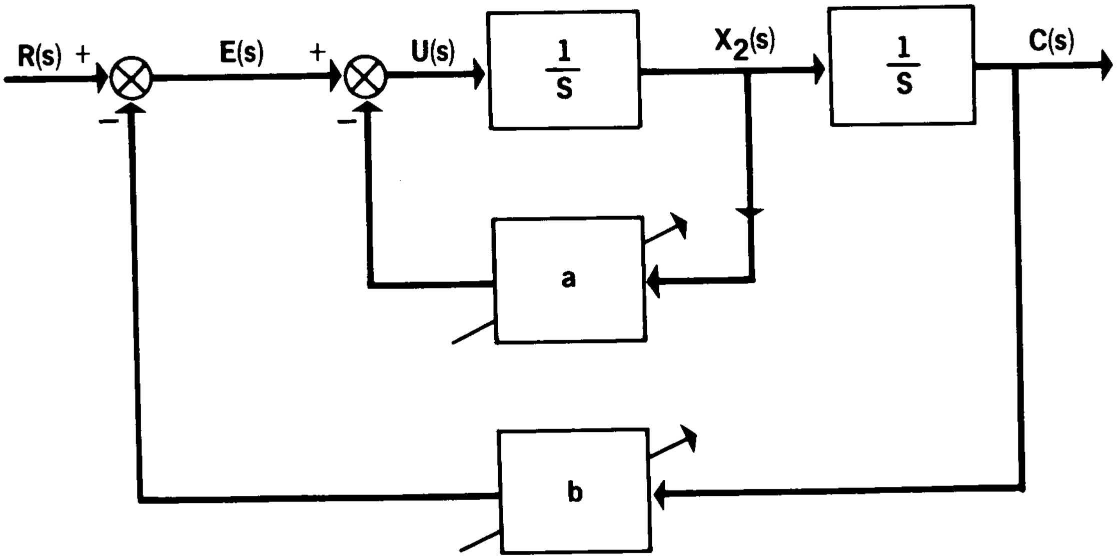


Figure 3. Second-Order System with Two Variable Parameters, Without a Learning Capability

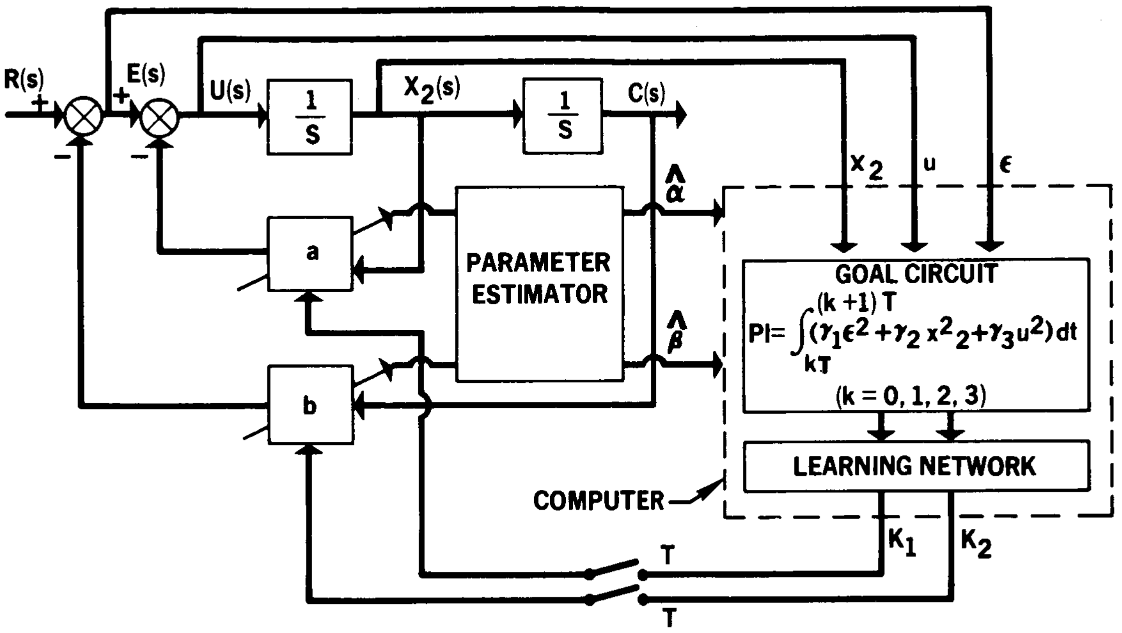
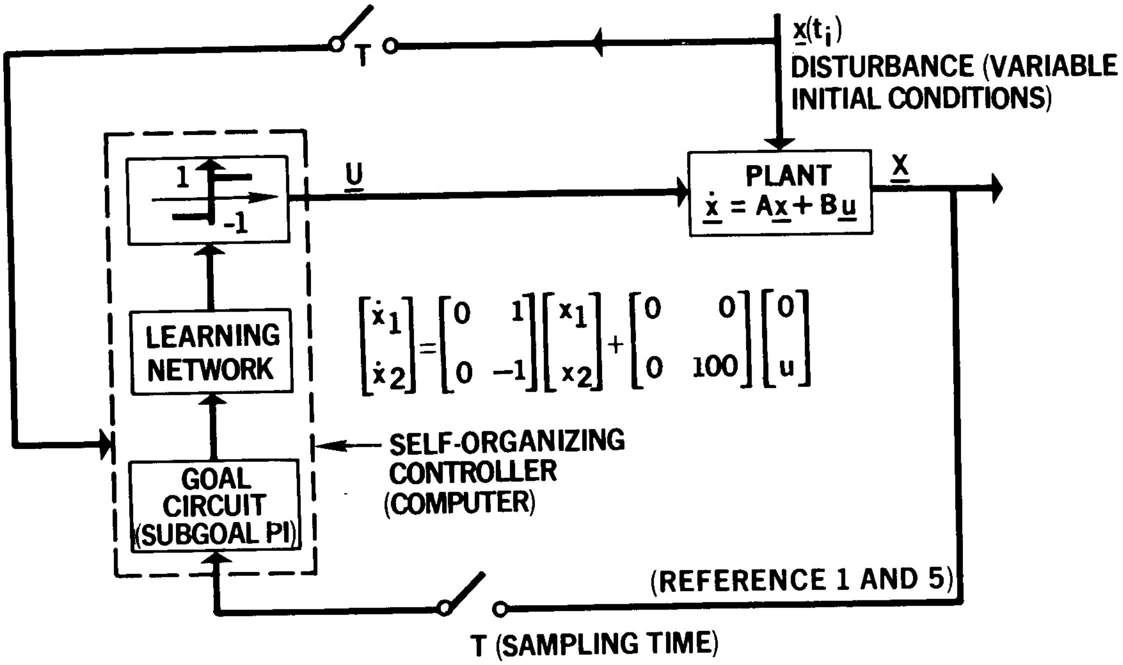


Figure 4. Second-Order System with Two Variable Parameters with a Learning Capability

The first time the system "sees" the pair (α_i, β_i) it reacts as a conventional adaptive-control system. When the pair (α_i, β_i) reoccurs, however, the best values of K_1 and K_2 are set from memory, and the best directions to increment K_1 and K_2 are known. Adaption then proceeds, and better values of K_1, K_2, direction, and PI replace the old values for (α_i, β_i) in the memory. In this way, the system's past experience is incorporated into the machinery responsible for future control choices (K_1 and K_2). The random state variable (RSV) learning strategy discussed by Barron (refs. 3 and 4) utilizes a random search technique for obtaining K_1 and K_2. The system begins by making a random experimental change in K_1 and K_2. If system performance is improved as a consequence of this experiment (as determined in the goal circuit), the new values for K_1 and K_2 are retained; otherwise, the initial changes are discarded, and a new random experiment centered about the original values of K_1 and K_2 is tried. Learning proceeds in this fashion to those values of K_1 and K_2 that provide the best performance.

This first on-line-learning control system has a greater capability than a conventional adaptive system because it recognizes similarly recurring control situations (combinations of α and β) and uses and improves the best previously obtained values of K_1 and K_2 for each control situation.

A second on-line-learning control system is shown in fig. 5.


x(t_i)
DISTURBANCE (VARIABLE INITIAL CONDITIONS)
T
PLANT
ẋ = Ax + Bu
X
U
1
-1
LEARNING NETWORK
GOAL CIRCUIT (SUBGOAL PI)
SELF-ORGANIZING CONTROLLER (COMPUTER)
(REFERENCE 1 AND 5)
T (SAMPLING TIME)

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix}=\begin{bmatrix}0 & 1\\ 0 & -1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix}+\begin{bmatrix}0 & 0\\ 0 & 100\end{bmatrix}\begin{bmatrix}0\\ u\end{bmatrix}$$

Figure 5. Bang-Bang System Studied by Waltz

In this system, the controller learns to drive the state vector $\underline{x}$ from any set of initial conditions to within a distance δ of the origin in the state space in a way that approaches the optimum as defined by the system PI. (This was chosen to be

$$PI = \sum_{r=1}^{n} r\, x_1^2\,(rT)$$

where n is the sampling instant when $\underline{x}$ arrives to within δ of the origin.) Learning occurs through a set stimulus-response relationship between elements of the state space and the control-choice space, which, in this case, contains only the elements, +1 and -1. The approach is first to design a controller that partitions the state space into sets called control situations, and then to learn the best control choice for each situation.

To be specific, the state space is partitioned into circular sets. (For higher-order systems, the state space is partitioned into hyperspherical sets.) This space, according to Waltz, is "partitioned (into control situations) by constructing circular sets of prespecified, fixed radius D. A given measurement vector (state vector) is considered a member of the set which it is closest to, providing the distance between the set vector (center of the set) and the measurement vector is less than D. If this distance is greater than D, a new set is established, and its set vector is equal to the measurement vector. Initially there are no sets and sets are only established in the vicinity of observed measurement vectors. Thus, memory is not wasted in establishing sets in regions where measurements never occur (ref. 1). "

Each circular set (control situation) has either a +1 or a -1 control choice. All state vectors within the same circular region are assumed to have the same control choice. Initially, the probability of either a +1 or a -1 control choice is assumed to be the same. In this case, learning is by reinforcement of the probability that either +1 or -1 will be chosen for a given control situation. This reinforcement is partially based on the optimization of a quadratic subgoal PI every T seconds. (The control choice is fixed over each sampling interval.) In effect, the optimization leads to either a positive or a negative reinforcement of the probability that the control choice for a control situation is +1 or -1. Finally, as learning proceeds, the probability approaches unity for one of the control choices in each control situation.

In this case, the PI does not enter directly into the design of the controller (which contrasts with the preceding system); although, as stated by Gibson and Fu (ref. 1), it does enter into the choice of a proper subgoal PI. (Usually, the PI has an integral form over a number of sampling periods. On the other hand, the subgoal PI is defined over each sampling period.) Nevertheless, the PI in this case, as well as in the preceding adaptive system, may be used as an indicator of learning on a learning curve.

A learning curve is a plot of performance as a function of time, or the number of practice trials used to measure learning. A typical learning curve for an arbitrary PI is shown in fig. 6.

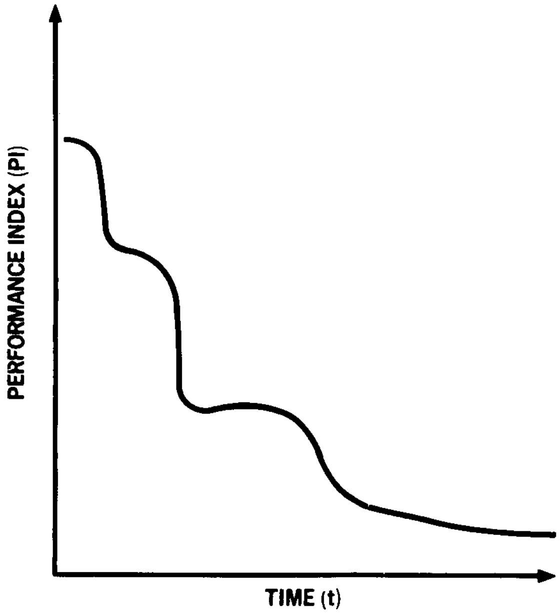


Figure 6. A Representative Learning Curve

Improved performance in this case is demonstrated by a reduction in the PI as a function of increasing values of time. One difficulty in the design of either of the preceding systems is the choice of meaningful PI's and subgoal PI's. This is pointed out by Gibson and Fu (ref. 1) who note that the PI's chosen should have a unique minimum that will be sought out by the system. In terms of learning curves, this means that after sufficient time or practice trials, the PI should remain constant. An interesting choice for an index of performance is made by Connelly (ref. 6) and Barron (ref. 4), who demonstrate the feasibility of using stability criteria in the goal circuit.

The objective of Connelly's study was to design a "bang-bang" controller that would maintain stable operation for the plant $K/s(s + a)$ in the face of plant changes, controller changes, and controller deterioration. A Lyapunov function, $V(\underline{x})$, is defined and evaluated in the goal circuit. The goal circuit then rewards those control choices for which $\dot{V}(\underline{x}) < 0$ and punishes those for which $\dot{V}(\underline{x}) > 0$. The system is similar in many respects to the system studied by Waltz (fig. 5), except that the learning network consists of statistical switches which learn under the influence of the goal circuit to provide the proper control choice. Barron claims better control when using $\ddot{V}(\underline{x})$ instead of $\dot{V}(\underline{x})$. This means that his goal circuit generates a reward signal if $\ddot{V}(\underline{x}) < 0$ and a punish signal if $\ddot{V}(\underline{x}) > 0$.

The ability of a system to improve its performance to a recurrent situation typifies the behavior of an on-line-learning system. For example, if the system in fig. 5 is subjected to the disturbance $\underline{x}_A(t_i)$, then $\underline{x}_B(t_i)$, and then $\underline{x}_A(t_i)$ again, it does not behave as if it had never encountered $\underline{x}_A(t_i)$. In short, its adaptation to $\underline{x}_B(t_i)$ does not destroy its previous adaption to $\underline{x}_A(t_i)$.

Theoretically, the on-line-learning controller can be trained during the real-time operation of the overall control system in which it is embedded. In practice, however, this training is performed before the entire system is operable. In this way, overall system performance is brought up to an acceptable level during what might be called on-line training. Starting at this level, the on-line-learning controller continues to adjust to improve performance during the real-time system operation.

For on-line learning to improve a system's performance while on-line, new information must be made available to the on-line-performance assessor. This means that information not available a priori must be utilized by the performance assessor in making the decisions as to how or if the controls should be modified. If no new information becomes available on-line, the controller could have been designed ahead of time; and there would be no need for an on-line-learning capability.

Space vehicle applications for the on-line-learning concept seem abundant. On-line-learning control appears most suited to unmanned applications, such as attitude control of orbiting vehicles, solar probes, and atmospheric-entry vehicles. Unmanned geophysical research and weather satellites are required to maintain attitude control with respect to Earth for long durations.

For greater pointing accuracies than afforded by gravity-gradient techniques, a long-life, active system is needed. This could be a system with current-carrying coils that are powered by solar energy and that interact with Earth's magnetic field. On-line learning could minimize energy use and attitude perturbations during seasonal changes in radiation and atmospheric external torques.

On-line learning could also be used to provide fine attitude control such as would be required by a laser communication satellite in Mars or Earth orbit. In these applications, the overall goal would be to keep a number of state variables (attitude errors) within a region centered at the origin of the state space when the satellite is subject to random disturbances (for example, solar and atmospheric perturbation torques) and/or component deterioration, including progressive failures. A second application in which the on-line-learning concept would cope with progressive failures is a solar probe to distances of less than one-half an astronomical unit (AU). This application suggests the use of fluid control rather than electronic components and is contingent upon advances in the survivability of other subsystems as well, particularly communications. Another potential application would be the atmospheric-entry attitude-control system for an unmanned planetary probe. This control system may deteriorate on the long journey to another planet. On-line learning could commence during the actual entry and achieve the best possible performance with the degraded system.

Finally, an application of on-line learning for the distant future would be the logical organization of automata to implement exploration and experimentation on other planets. On-line learning could minimize energy use in the performance of assigned tasks.

Section 4

OFF-LINE-LEARNING CONTROL SYSTEMS

An "off-line-learning" control system is one in which the inputs and plant are known a priori; the actual controller is incompletely specified (partially known) and is replaced by a self-organizing system which learns (is trained) to control the system properly off-line.

A representative off-line-learning control system is shown in fig. 7.

Learning occurs with the switch S in position ①; the self-organizing system, $\mathscr{S}$ is shown representative problems and their solutions. For each sample, certain internal modifications to the learning network increase its proficiency. Training requires no external intervention and is systematically convergent toward a learned state. At this point, internal modification of the learning network stops, and it now behaves in a conventional deterministic fashion; i. e., its responses to problems are not based upon any statistical phenomena or past history but are dependent upon its present static internal

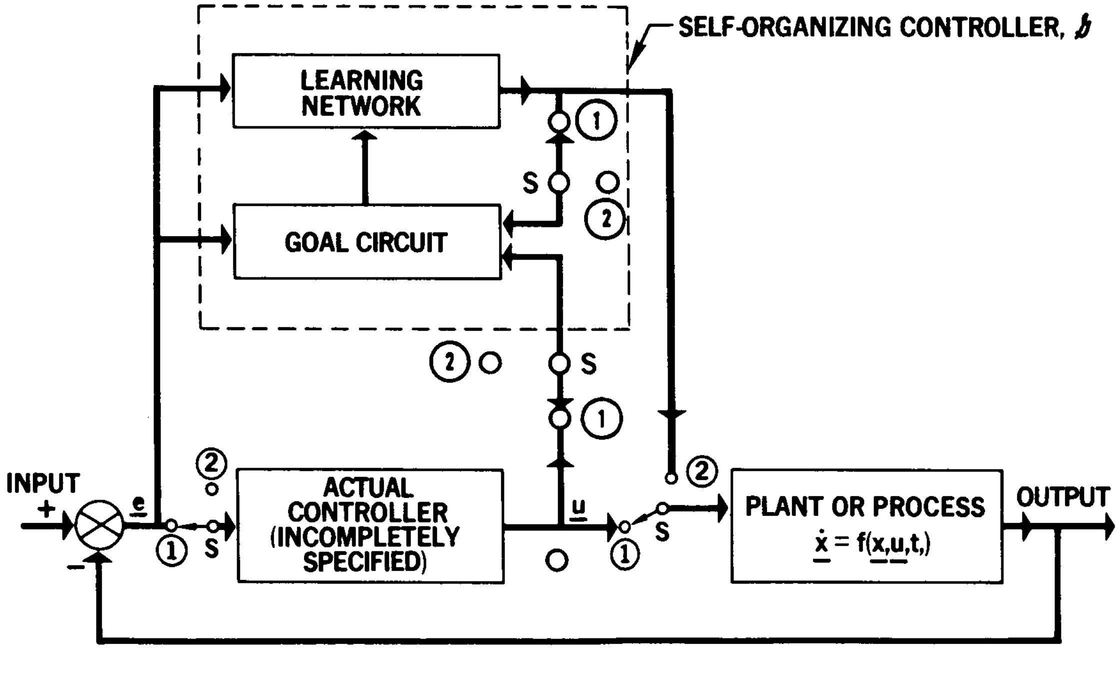

SELF-ORGANIZING CONTROLLER, $\mathscr{S}$
LEARNING NETWORK
GOAL CIRCUIT
S
①
②
INPUT
+
−
e
ACTUAL CONTROLLER (INCOMPLETELY SPECIFIED)
u
PLANT OR PROCESS
$\dot{\underline{x}} = f(\underline{x},\underline{u},t,)$
OUTPUT

Figure 7. Representative Off-Line-Learning Control System

state. Its performance, therefore, will be exactly repeatable. Furthermore, if well trained, the learning network will be able to generalize problems not encountered in training and obtain solutions corresponding to the most similar problem encountered in training. The generalization ability of the learning network allows the incompletely specified actual controller to be bypassed during the real-time control of the plant. With the switch S in position ②, however, the system shown fig. 7 behaves like a conventional nonadaptive control system: learning does not occur.

In the control-system applications of off-line learning discussed in the literature (refs. 7 through 10), the self-organizing system is of the type shown in fig. 8; for clarity, $\underline{e}$ is assumed to be a two-dimensional state vector and u is assumed to be a scalar.

This type of system--without the input encoder--has been described variously as a learning machine, an adaptive pattern recognizer or classifier, an adaptive majority-vote taker, an adaptive linear neuron, and an adaptive linear-threshold element (ADALINE). In the sequel, the complete system is referred to as an adaptive computer. The only other adaptive computers reported utilize an Artron (ref. 11) and a Neurotron (ref. 12).

The e_1 and e_2 inputs to the adaptive computer in fig. 8 are each divided into m quanta. The V's are binary signals having the value of +1 or -1; V_o is a fixed threshold input set at + 1. The set of 2m inputs $V_1, \ldots, V_{2m}$ is often referred to as an input pattern (ref. 9).

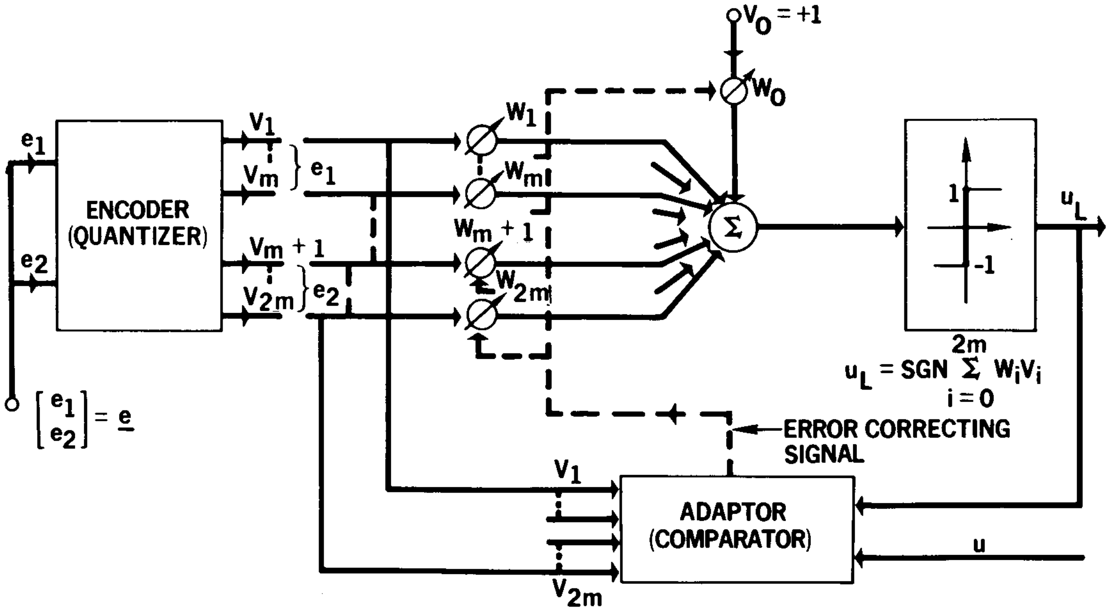


Figure 8. Adaptive Computer, $\mathcal{G}$

The weights, W_i, are learning parameters. The learning feature of $\mathcal{L}$ is provided by the adaptor which adjusts weight values through an iterative training procedure (learning algorithm) to minimize the error between the output, u_L, of the adaptive computer and some desired output, u.

During the training process, the adaptive computer changes its weights only upon the basis of the input pattern present and the desired output pattern. Weights are adjusted by error-correcting techniques known to lead to convergent methods of adaptation for adaptive computers (ref. 13); reinforcement techniques may lead to divergent methods of adaption. Weights are changed only when an input pattern gives an output, u_L, opposite the desired output, u; they are all changed by equal increments.

During training, storage of input patterns or calculations involving more than one input pattern at a time is not necessary. Instead, the input patterns are presented to the adaptive computer sequentially, several times, until all those in the training set are being correctly classified, or until the number of classification errors have reached some steady-state value. During the training procedure, the input patterns may be presented in any order (ref. 9).

Often, a PI, such as the sum of the squares of the errors between the desired and actual outputs of the adaptive computer, is plotted versus the total number of input patterns adapted to by the computer. Such a curve indicates the learning progress of the adaptive computer during training; however, it is meaningful only when switch S (fig. 7) is in position ①. As previously noted, learning does not occur when switch S is in position ②.

An overall PI and a subgoal PI for the adaptive computer are distinguishable (see the second Purdue system discussed previously). The subgoal of the adaptive computer is correct classification of each input pattern. This subgoal directs the updating of the weights, W_i. The overall PI is that the adaptive computer correctly classify all such input patterns. The overall goal determines how often the sequence of input training patterns has to be applied to the adaptive computer before complete learning occurs.

The most attractive feature of the adaptive computer is its ability to generalize input patterns not encountered in training and to obtain solutions corresponding to the most similar input patterns encountered in training. Generalization relates to the concepts of linear separability (ref. 10) and projectability (ref. 7). Moreover, the adaptive computer remains very reliable despite component failures (the weights, W_i) and is particularly well suited to resolve problems for which no analytical solutions can be found.

In fig. 8, u_L is limited to two values, ±1; thus, the desired output, u, must also be limited to ±1; otherwise the adaptive computer could not learn to simulate the actual controller. The adaptive computer is especially useful, therefore, in those situations where the actual controller is the bang-bang type; hence, almost all applications of the adaptive computer in off-line-learning control systems have been limited to the time-optimal control of linear systems where the actual controller is bang-bang. These studies involved (with modifications resulting from individual treatment by each author) the following (see fig. 9):

(1) Quantization of the state space in region of interest and identification of the resulting hypercubes with linearly independent codes.

(2) Selection of a subset I from the complete set of initial conditions $\{\underline{x}(0)\}$

(3) Computation of the open-loop optimal-control, u*(t), and trajectory, $\underline{x}^*(t)$, for each element of I and for a specific plant.

(4) Identification of the controls for the hypercubes through which the trajectories pass; let these hypercubes and their controls constitute the training set S.

(5) Training an off-line learning controller by means of the set S in 4.

These five steps constitute a nonparametric training procedure, which is fairly well established in the discipline of pattern recognition (ref. 13).

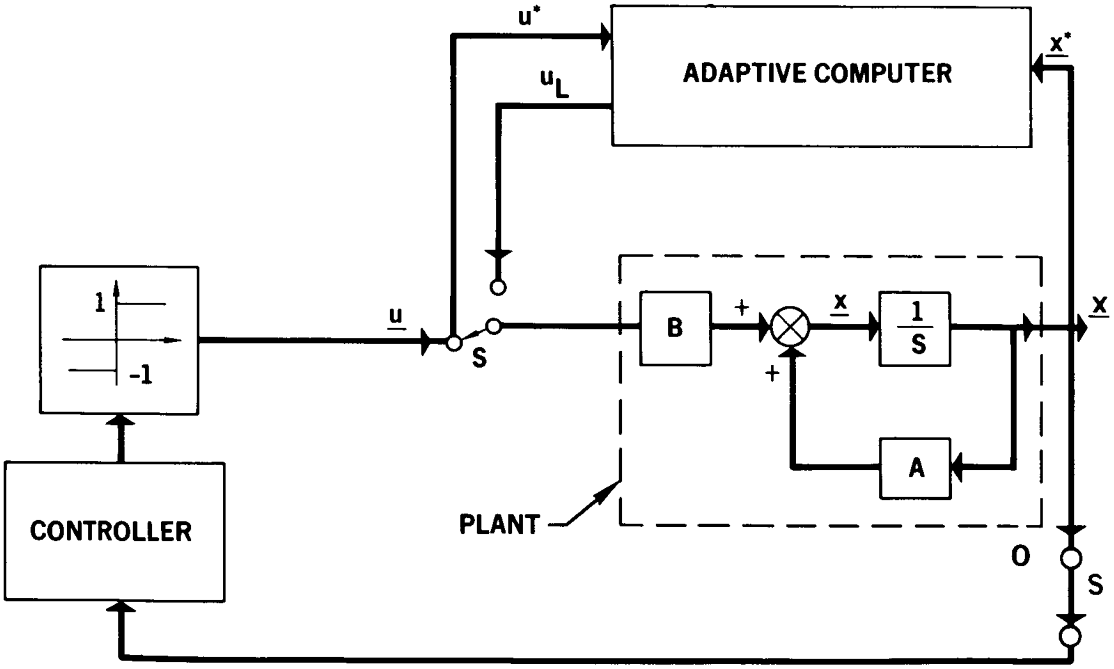


Figure 9. Adaptive Computer for Time-Optimal Control of a Linear Plant

A significant difference in the studies reported in the literature (ref. 8 through 10, and 14) is the selection of the subset I from the complete set of initial conditions $\{\underline{x}(0)\}$ sufficient for complete learning. Van Nortwick (ref. 10) and Zapalac (ref. 14) train the adaptive computer on a set of input patterns from every square in the quantized state space. F. W. Smith (ref. 9) trains the adaptive computer on a set of input patterns from every hypercube in the quantized state space that borders the optimal switching surface. In this case, the optimal switching surface is known a priori. Both of these approaches are practical only for low-order plants. However, F. B. Smith (ref. 8) considers high-order systems for which the exact form of the optimal switching surface, as a function of the state variables, is not known. In these systems, the adaptive computer is trained on a set of input patterns uniformly distributed in the quantized state space. For example, in the control of a fourth-order rigid vehicle, an arbitrary region of the four-space is defined and quantized into 524, 288 four-dimensional cubes. The adaptive computer is trained on a set of 400 initial conditions chosen at uniform intervals in the arbitrary region. This represents only 0. 076% of the total input patterns. According to F. B. Smith, "If the sample set I is sufficiently representative, then the controller obtained will provide control with desirable characteristics for a much wider class of inputs than the sample set" (ref. 8). More recently, Smith has extended his study to include the time-optimal control of systems with variable plants.

At present, the off-line-learning concept, which represents a control system application of pattern recognition, serves as a useful design tool by achieving practical realizations for closed-loop (sub-) optimal control laws for systems where analytical solutions are not feasible. In addition, off-line-learning controllers are extremely reliable in the sense that failures of one or more weights in the adaptive computer do not markedly deteriorate system performance. The full potential of off-line-learning control systems has not been realized, however. At present, the only self-organizing system apparently utilized in these systems is an adaptive computer with a single, linear-threshold element. This necessarily restricts the application of this type of computer to the class of bang-bang controllers. Although this includes the important application of time-optimal control, it eliminates problems that frequently occur in space-vehicle applications, such as minimum-fuel control, minimum-energy control, and any controller requiring multilevel outputs. Finally, all time-optimal switching surfaces are not necessarily realizable with single linear-threshold elements. The surface usually must be projectable to permit this realization (ref. 7 and 14). By including more threshold devices in the adaptive computer, it should be possible to make them applicable to control problems other than time-optimal problems. For example, it has been shown that two linear-threshold elements can realize the three states, -1, 0, and +1. It is likely, therefore, that an adaptive computer containing two threshold devices would be useful in the minimum-fuel problem for realizing the closed-loop optimal control law, where, as is well known, the optimal control law (for linear systems) is of the on-off-on (relay with dead-zone) variety (ref. 15). An interesting space-vehicle application is one in which the off-line-learning controller is trained to act as a backup mode for man during a specific mission, such as re-entry. In this application, man supplies the required training samples through on-ground simulations. This can be considered analogous to optimization theory which supplies the open-loop optimal controls (training samples) when the off-line-learning controller is used, for example, to realize closed-loop time-optimal control.

Section 5

CONCLUDING REMARKS

To combine the advantages of on-line and off-line systems, one could provide the off-line-learning control system with an on-line learning capability. A block diagram of such a system is shown in fig. 10 (Reference 16).

With switches S_1 in position ① and S_2 in position ②, the system reduces to the off-line-learning control system shown in fig. 7. On the other hand, with switches S_1 in position ② and S_2 in position ①, the system reduces to the on-line-learning control system shown in fig. 2.

While the combined system is conditioned on all available a priori information about the plant and environment, as in an off-line learning controller, its main advantage is that it would be able to reorganize if the system experienced a partially known or unknown environment, or if components deteriorated, as in an on-line-learning controller. Such a combined system is described in Part 2 of this report, in connection with the fine attitude control of a laser communication satellite in Mars orbit. Nominal controls are designed with a priori information about variations in plant parameters and disturbance torques. On line, the nominal controls may be updated to compensate for incorrect or incomplete information about actual disturbance torques.

To demonstrate another possible advantage of an off-line system with an on-line learning capability over off-line- and on-line-learning control systems, a hypothetical PI is assumed which is meaningful for both off-line and on-line learning. (A major problem is to find a PI meaningful for both on-line and off-line learning.) Fig. 11 presents a comparison of the three control situations: (1) training plus on-line learning; (2) training and no on-line learning; and (3) on-line learning with no prior training.

Learning occurs in (2) only during the training period, as is evident from the constancy of the training and no on-line learning curve during on-line operation. Note that, until time τ, the performance of (2) is better than the performance of (3). For $t > \tau$, however, (3) swiftly overtakes the performance of (2) and eventually reaches the minimum PI, whereas the performance of (2) remains unchanged. Also, (1) reaches the minimum PI sooner than (3) because when on-line learning begins, (1) starts out with a lower PI than (3).

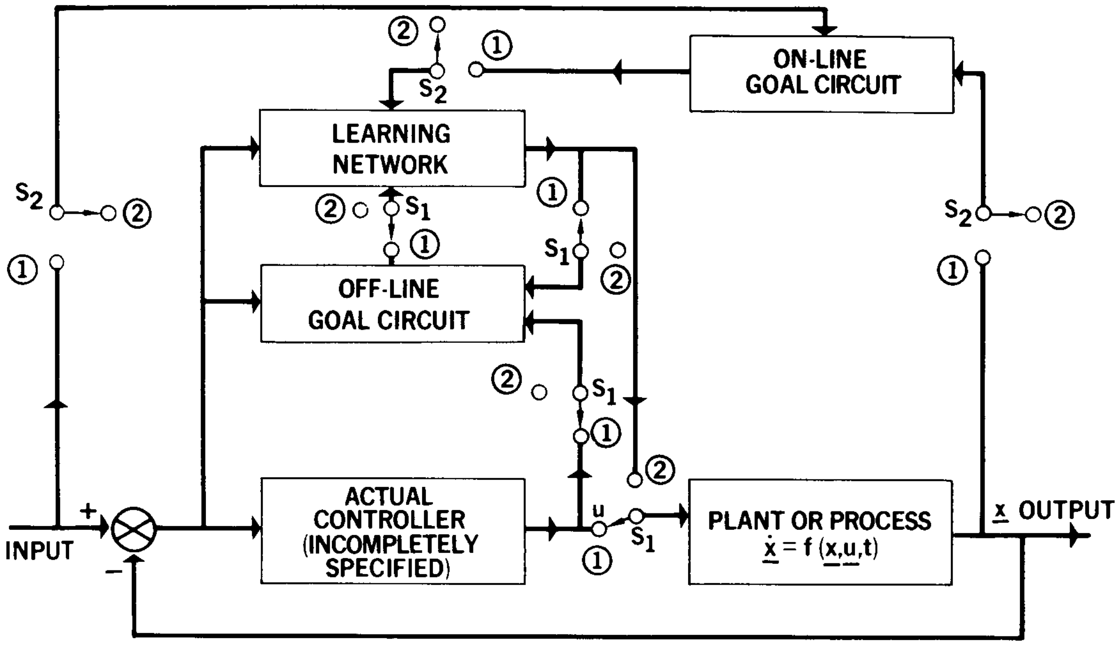


Figure 10. Off-Line-Learning Control System with On-Line-Learning Capabilities

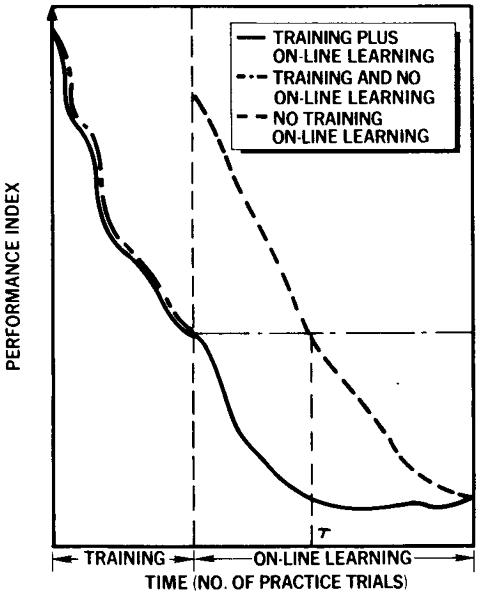


Figure 11. Learning Curves (Hypothetical)

REFERENCES, PART I

1. Fu, K. S.; Gibson, J. E.: et al: Philosophy and State of the Art of Learning Control Systems. Rep TR-EE63-7 (AF Report AFOSR 5144). Control and Information Systems Lab., School of Electrical Engineering, Purdue University, Nov. 1963.

2. Carne, E. B.: Self-Organizing Models - Theory and Techniques. Paper Presented at National Aerospace Electronics Conference, Dayton, Ohio, 1962.

3. Barron, R. L.; Davies, J. M.; Schalkowsky, S.; and Snyder, R. F.: Self-Organizing Adaptive Systems for Space Vehicle Attitude Control. Presented at the SAE-18 Committee Meeting, Miami Beach, Florida, Dec. 1964.

4. Barron, R. L.; Davies, J. M.; Schalkowsky, S.; and Snyder, R. F.: Self-Organizing Adaptive Systems for Space Vehicle Attitude Control. Presented at the AIAA/ION Guidance and Control Conference, Minneapolis, Minnesota, Aug. 1965.
   (NOTE: Content of this paper is different from that of Reference 3.)

5. Fu, K. S.; and Waltz, M. D.: A Computer-Simulated Learning Control System. 1964 IEEE International Convention Record, Part 1, pp. 190-201.

6. Connelly, E. M.; Mirabelli, R. E.; and Worthen, J. H.: Feasibility Studies on Use of Artrons as Logic Elements in Flight Control Systems. FDL-TDR-64-23, Wright-Patterson AFB, Ohio, Feb. 1964.

7. Berkovec, J. W.; and Epley, D. L.: On Time-Optimal Control with Threshold Logic Units. Preprint 4.1, WESCON, Aug. 1964.

8. Smith, F. B., Jr.: A Logical Net Mechanization for Time-Optimal Regulation. NASA TN D-1678, 1962.

9. Smith, F. W.: Contactor Control by Adaptive Pattern-Recognition Techniques. Rep. No. 6762-1, Stanford Electronics Laboratories, Stanford University, April 1964.

10. Van Nortwick, K. G.: An Adaptive Computer Applied to a Second-Order Non-Linear Control System. Boeing Report No. D2-90192-7, May 1963.

11. Fuhr, W. H.: A Study of the Feasibility of Using Artificial Neurons to Develop More Reliable Flight Control Systems. ASD-TDR-63-143, Wright-Patterson AFB, Ohio, April 1963.

12. Lee, R. J.; and Snyder, R. L.: Functional Capability of Neuromime Networks for Use in Attitude Stabilization Systems. Report No. ASD-TDR-63-549, Wright-Patterson AFB, Ohio, Sept. 1963.

13. Nilsson, N. J.: Learning Machines: Foundations of Trainable Pattern-Classifying Systems. McGraw-Hill Book Co., Inc., 1965.

14. Zapalac, J. J.: Self-Organizing Control Systems, Vol. 1. On Adaptive Computers. Douglas Report SM-47857, July 1965.

15. Mendel, J. M.; and Zapalac, J. J.: Self-Organizing Control Systems, Vol. 3. Off-Line Training of Time-Optimal, Fuel-Optimal, and Minimum-Energy Controllers. Douglas Report SM-51975, Feb. 1966.

16. Mendel, J. M.: On Applications of Biological Principles to the Design of Feedback Control Systems. Douglas Report SM-47772, Nov. 1964.

BIBLIOGRAPHY

The following is not intended to be a fully annotated bibliography. For clarity and ease of reference, the works are divided into three sections: On-Line-Learning Control Systems; Off-Line-Learning Control Systems; and Topics Related to Self-Organizing Systems (in some cases, a single work may be mentioned in more than one section).

On-Line-Learning Control Systems

Barron, R. L.; Davies, J. M.; Schalkowsky, S.; and Snyder, R. F.: Self-Organizing Adaptive Systems for Space Vehicle Attitude Control. Paper presented at the SAE-18 Committee Meeting, Miami Beach, Florida, Dec. 1964.

Probability-state-variable and random-state-variable methods are outlined; background material for on-line-learning systems is also presented.

Barron, R. L.; Davies, J. M.; Schalkowsky, S.; and Snyder, R. F.: Self-Organizing Adaptive Systems for Space Vehicle Attitude Control. Paper presented at the AIAA/ION Guidance and Control Conference, Minneapolis, Minnesota, Aug. 1965.

Probability-state-variable method is discussed and experimental results are presented.

Barron, R. L.; Davies, J. M.; Schalkowsky, S.; and Snyder, R. F.: Self-Organizing Spacecraft Attitude Control. Report No. AFFDL-TR-65-141, Wright-Patterson AFB, Ohio, Aug. 1965.

Probability-state-variable method is applied to the attitude control and stabilization of a low-order model for a spacecraft; goal circuits based upon Lyapunov functions are investigated.

Connelly, E. M.; Mirabelli, R. E.; and Worthen, J. H.: Feasibility Studies on Use of Artrons as Logic Elements in Flight Control Systems. Report No. FDL-TDR-64-23, Wright-Patterson AFB, Ohio, Feb. 1964.

Use of Artrons as learning controllers is discussed, and control systems in which goal circuits are based upon Lyapunov functions are investigated.

Fu, K. S.; Gibson, J. E.; et al: Philosophy and State of the Art of Learning Control Systems. Report No. TR-EE63-7 (AF Report AFOSR 5144), Control and Information Systems Laboratory, School of Electrical Engineering, Purdue University, Nov. 1963.

Contains much pertinent material, including some background on learning.

Fu, K. S.: Learning Control Systems. Computer and Information Sciences (J. T. Tou and R. H. Wilcox, editors), Spartan Books, Inc. Washington, D.C., 1964, Chapter 13, pp. 318-343.

Pattern recognition and learning theory are applied to control-system designs.

Fu, K. S.; and Waltz, M. D.: A Computer-Simulated Learning Control System. IEEE International Convention Record, 1964, Part 1, pp. 190-201.

Pattern recognition (control situations) is applied to control-system designs.

Fu, K. S.; and Waltz, M. D.: A Learning Control System. Preprint I-1, Joint Automatic Control Conference, June 1964.

Is quite similar to preceding reference.

Fu, K. S.; Hill, J. D.; and McMurtry, G. J.: A Computer-Simulated On-Line Experiment in Learning Control Systems. AFIPS Conference Proceedings, Spring Joint Computer Conference, Vol. 25, 1964, pp. 315-325.

Adaptive control system with memory is investigated.

Fu, K. S.; and Hill, J. D.: A Learning Control System Using Stochastic Approximation for Hill-Climbing. Preprint XIV-2, Joint Automatic Control Conference, June 1965.

Stochastic approximations and learning rates are discussed.

Fu, K. S.; and McMurtry, G. J.: A Study of Stochastic Automata as Models of Adaptive and Learning Controllers. Report No. TR-EE65-8, School of Electrical Engineering, Purdue University, June 1964.

Models of automata operating in a random environment are applied in a controller of an adaptive learning control system.

Fu, K. S. ; and Waltz, M. D. : A Heuristic Approach to Reinforcement Learning Control Systems. IEEE Transactions on Automatic Control, Vol. 10, No. 4, Oct. 1965, pp. 390-398.

Pattern recognition (control situations) is applied to control-system designs.

Fuhr, W. H. : A Study of the Feasibility of Using Artificial Neurons to Develop More Reliable Flight Control Systems. Report No. ASD-TDR-63-143, Wright-Patterson AFB, Ohio, April 1963.

Self-organization is discussed in relation to improved reliability.

Gibson, J. E.: Adaptive Learning Systems. (Available from ASTIA as AD 292 796), Jan. 1963.

Difference between learning and adaptive control systems is discussed.

Hill, J. D.: An On-Line Learning Control System Using Modified Search Techniques. Ph. D. Thesis, Purdue University, Jan. 1965.

Adaptive control system with memory is investigated; stochastic approximations and learning rates are discussed; and pattern recognition is applied to an index of performance.

Hsu, J. C.; and Meserve, W. E.: Decision-Making in Adaptive Control Systems. IRE Transactions on Automatic Control, Vol. 7, No. 1, Jan. 1962, pp. 24-32.

Decision theory used to link the problems of identification and control when only inexact measurements are available.

Krug, G. K.; and Netushil, A. V.: Automatic Systems with Learning Elements. Paper presented at Second IFAC Meeting, Basle, Switzerland, 1963.

On-line-learning systems that use approximating polynomials are discussed.

Lee, R. J.; and Snyder, R. F.: Functional Capability of Neuromime Networks for Use in Attitude Stabilization Systems. Report No. ASD-TDR-63-549, Wright-Patterson AFB, Ohio, Sept. 1963.

Neurotrons as learning controllers are discussed and learning based upon repeated application of the environment is investigated.

Mendel, J. M.: On Applications of Biological Principles to the Design of Feedback Control Systems. Douglas Report No. SM-47772, Nov. 1964.

Off-line-and on-line-learning control systems are surveyed.

Mosteller, H. W.: Learning Control Systems. Electronics Laboratory, General Electric, 1963.

Off-line- and on-line-learning control systems, reinforcement control, and learning model studies are surveyed.

Raible, R. H.: A Learning Control System Based on Adaptive Principles. Ph. D. Thesis, Purdue University, Aug. 1964.

Learning control is considered in a stepwise and continuously changing environment--the study is entirely experimental.

Taylor, W. K.: A Pattern Recognizing Adaptive Controller. Paper presented at Second IFAC Meeting, Basle, Switzerland, 1963.

Pattern recognition is applied to control system design (similar to the Purdue studies).

Tou, J. T.: System Optimization Via Learning and Adaptation. International Journal of Control, Vol. II, No. I, July 1965, pp. 21-32.

Approach for designing adaptive and learning systems for achieving optimal control is discussed; decision theory and dynamic programming are used.

Waltz, M. D.: A Study of Learning Control Systems Using a Reinforcement Technique. Ph.D. Thesis, Purdue University, Aug. 1964.

Pattern recognition (control situations) is applied to control-system-designs; also discussed are reinforcement learning operators used for adjusting the probabilities of particular control choices for given control situations.

Off-Line-Learning Control Systems

Berkovec, J. W.; and Epley, D. L.: On Time-Optimal Control with Threshold Logic Units. Preprint 4. 1, WESCON, Aug. 1964.
Concept of projectability is discussed and related to the number of linear threshold elements required for a realization of a switching surface.

Charters, R. L.: Applications of Learning Machines to Control Systems. Paper presented at National Aerospace Conference, Dayton, Ohio, 1964.
Boeing's efforts in the field of off-line-learning control are reviewed.

Mendel, J. M.: On Applications of Biological Principles to the Design of Feedback Control Systems. Douglas Report No. SM-47772, Nov. 1964.
Off-line- and on-line-learning control systems are surveyed.

Mendel, J. M.: Self-Organizing Control Systems, Vol. 2, Open-Loop Time-Optimal Control of a Stable Maneuverable Re-Entry Vehicle. Douglas Report No. SM-47904, June 1965.
Off-line learning for closed-loop time-optimal control is discussed.

Mendel, J. M.; and Zapalac, J. J.: Self-Organizing Control Systems, Vol. 3, Off-Line Training of Time-Optimal, Fuel-Optimal and Minimum-Energy Controllers. Douglas Report SM-51975, Feb. 1966.
Adaptive computers trained to provide approximations of time-and fuel-optimal switching surfaces are discussed; experimental and analytical results are presented.

Mosteller, H. W.: Learning Control Systems. Electronics Laboratory, General Electric, 1963.
Off-line- and on-line-learning control systems, reinforcement control, and learning model studies are surveyed.

Smith, F. B., Jr.: A Logical Net Mechanization for Time-Optimal Regulation. NASA TN D-1678, 1962.
Closed-loop, time-optimal control of third-and fourth-order plants are investigated with off-line training.

Smith, F. B., Jr.; Lee, J. F. L.; Butz, A. R.;and Prom, G. J.: Trainable Flight Control System Investigation. FDL-TDR-64-89, Wright-Patterson AFB, Ohio, Aug. 1964.
Closed-loop, time-optimal adaptive control with off-line training is discussed.

Smith, F. W.: Contactor Control by Adaptive Pattern-Recognition Techniques. Report No. 6762-1, Stanford Electronics Laboratories, Stanford University, April 1964.
Applications of ADALINES to control systems are investigated.

Smith, F. W.; and Widrow, B.: Pattern-Recognizing Control Systems. Computer and Information Sciences (J. T. Tou and R. H. Wilcox, editors), Spartan Books, Inc., Washington, D.C., 1964, Chapter 12, pp. 288-317.
ADALINES, MADALINES, and concept of generalization summarized; applications including closed-loop, time-optimal control, are discussed.

Van Nortwick, K. G.: An Adaptive Computer Applied to a Second-Order Non-Linear Control System. Boeing Report No. D2-90192-7, May 1963.
Closed-loop, time-optimal control of a second-order plant with off-line training is investigated.

Zapalac, J. J.: Adaptive Processes in Decision Making. Douglas Report No. SM-45921, April 1964.
Discussions on pattern recognition, analysis of linear threshold elements, learning algorithms, and learning machines are presented.

Zapalac, J. J.: Self-Organizing Control Systems--Volume I, On Adaptive Computers, Douglas Report SM-47857, July 1965.
Theory of linear threshold devices is summarized, and application of single-element adaptive computer to a time-optimal switching curve is detailed.

Topics Related to Self-Organizing Systems

Aizerman, M. A.: Automatic Control Learning Systems (in the Light of Experiments on Teaching the Systems to Pattern Recognition). Paper presented at Second IFAC Meeting, Basle, Switzerland, 1963.
Discusses problem of teaching automatic systems using pattern recognition.

Andrews, A. M.: Learning in Control Systems. Control, Vol. 2, Sept. 1960, pp. 99-103.
Self-optimization and learning are discussed.

Andrews, A. M.: Self-Optimizing Control Mechanisms and Some Principles for More Advanced Learning Machines. Paper presented at First IFAC Meeting, Moscow, Russia, 1960.
Discusses learning and related subgoals for control systems.

Andrews, A. M.: Pre-Requisites of Self-Organization. Computer and Information Sciences (J. T. Tou and R. H. Wilcox, editors), Spartan Books, Inc., Washington, D. C., 1964, Chapter 16, pp. 381-391.
Discusses self-organization (this work is, in part, quite similar to the two immediately preceding).

Butsch L. M.; and Ostgaard, M. A.: Adaptive and Self-Organizing Flight Control Systems. Aerospace Engineering, Sept. 1963, pp. 80-116.
Self-organizing systems are proposed as natural extensions of adaptive control systems.

Carne, E. B.: Self-Organizing Models--Theory and Techniques. Paper presented at National Aerospace Electronics Conference, Dayton, Ohio, 1962.

Gives a useful definition of self-organizing systems.

Chichinadze, V. K.: Logical Design Problems of Self-Optimizing and Learning-Optimizing Control Systems Based on Random Searching. Paper presented at First IFAC Meeting, Moscow, Russia, 1960.

Self-optimization, random searching, and learning are discussed.

Elsey, J. C.; Fong, H. S. ; and Meyers, N. H.: Re-entry Guidance by Threshold Network Storage of Precomputed Optimum Commands. Paper (66-52) presented at AIAA Third Aerospace Sciences Meeting, New York, New York, 1966.

Trainable threshold networks are applied to the realization of optimum guidance surfaces.

Fu, K. S.; and McLaren, R. W.: Synthesis of Learning Systems Operating in an Unknown Random Environment. Paper presented at the International Conference on Microwave, Circuit Theory, and Information Theory, Tokyo, Japan, 1964.

Probabilistic automata and stochastic approximations are applied to the design of a learning system's learning section.

Fu, K. S.; and McLaren, R. W.: An Application of Stochastic Automata to the Synthesis of Learning Systems. Report No. TR-EE65-17, School of Electrical Engineering, Purdue University, Sept. 1965.

Stochastic automaton learning models are investigated in great detail; linear and nonlinear, continuous and discrete models are also considered.

Gerhardt, L. A.; Goerner, J. G.; and Powell, F. D.: The Application of Error Correcting Learning Machines to Linear Dynamic Systems. Paper presented at the National Electronics Conference, Chicago, Illinois, 1965.

Learning machines are used to model or invert an unknown system.

Gibson, J. E.: Adaptive and Learning Control Systems. System Engineering Handbook (R. E. Machol, W. P. Tanner, Jr., and S. N. Alexander, editors), McGraw-Hill Book Co., Chapter 30, 1965.

Adaptive and learning systems are compared; learning concepts, such as pattern recognition, the simple reinforcement learning model and the concept of a learning curve are discussed.

Ho, Y. C.: Adaptive Design of Feedback Controllers for Stochastic Systems. IEEE Transactions on Automatic Control, Vol. 10, No. 3, July 1965, pp. 367-368.

Discusses pattern classification for design of a controller in a stochastic system.

Ivakhnenko, A. G.: Engineering Cybernetics - USSR (English translation of the Russian-language monograph, Technicheskoya kibernetika Sistemy automaticheskago upravleniya s prisposoblenigen kharakteristik). Office of Technical Services, 1961.

Learning systems and self-organizing systems are discussed.

Ivakhnenko, A. G.: Self-Organizing Systems with Positive Feedback Loops. IEEE Transactions on Automatic Control, Vol. AC-8, No. 3, July 1963, pp. 247-254.

Discusses a self-organizing system.

Ivanenko, V. I.; and Tou, J. T.: On the Design of Learning Systems for Control. Computer and Information Sciences (J. T. Tou and R. H. Wilcox, editors), Spartan Books, Inc., Washington, D. C., 1964, Chapter 22, pp. 519-536.

Presents the theoretical background for learning control systems.

Kaplan, K. R.; and Sklansky, J. : Analysis of Markov Chain Models of Adaptive Processes. Report No. AMRL-TR-65-3, Wright-Patterson AFB, Ohio, Jan. 1965.

Techniques for the analysis of synchronous and asynchronous Markov chains are developed; emphasis is placed upon problems encountered in the use of these chains as models of adaptive processes.

Lendaris, G. G.: On the Definition of Self-Organizing Systems. Proceedings IEEE, Vol. 52, No. 3, March 1964, pp. 324-325.

Presents theoretical definition of a self-organizing system.

McLaren, R. W.: A Markov Model for Learning Systems Operating in an Unknown Environment. Paper presented at National Electronics Conference, Chicago, Illinois, 1964.

Outlines stochastic automata theory and applies it to the design of the learning section of a learning system.

Mesarovic, M. D.: Self-Organizing Control System. IEEE Transactions on Application and Industry, Vol. 83, No. 74, Sept. 1964, pp. 265-269.

Self-organizing systems are discussed from general and abstract points of view.

Miller, R. W.: Process Identification by Pattern Recognition Techniques. Ph.D. Thesis, Rensselaer Polytechnic Institute, Jan. 1964.

Pattern-recognition concepts are applied to process identification.

Minsky, M.: Steps Toward Artificial Intelligence, Proceedings of IRE, Vol. 49, No. 1, Jan. 1961, pp. 8-30.

Discusses learning systems (list 95 references).

Narendra, K. S.; and Streeter, D. N.: A Self-Organizing Control System Based on Correlation Techniques and Selective Reinforcement. Technical Report No. 359, Cruft Laboratory, Harvard University, July 1962.

Self-organizing control system based upon the principles of selective reinforcement investigated.

Nilsson, N. J.: Learning Machines: Foundations of Trainable Pattern-Classifying Systems. McGraw-Hill Book Co., Inc., 1965.
Discusses introductory pattern recognition concepts.

Shen, D. W. C.: Artificial Intelligence in a Hierarchy of Nonlinear Systems. Report No. 63-18, Moore School of Electrical Engineering, University of Pennsylvania, March 1963.
Learning systems are defined.

Sklansky, J.: Adaptation and Feedback. Paper presented at Joint Automatic Control Conference, New York, New York, June 1962.
Learning is discussed.

Sklansky, J.: Adaptation, Learning, Self-Repair, and Feedback. IEEE Spectrum, Vol. 1, No. 5, May 1964, pp. 172-174.
Discusses learning as it is related to self-repair, stability, and reliability.

Sklansky, J.: Learning Systems for Automatic Control (abridged version). IEEE International Convention Record, Part 6, 1965, pp. 117-122.
Discusses relation of decision theory, trainable threshold logic, stochastic approximations, and Markov chain theory to learning control systems.

Sklansky, J.: Learning Systems for Automatic Control. IEEE Transactions on Automatic Control, Vol. 11, No. 1, Jan. 1966, pp. 6-19.
Discusses recent developments in learning systems for automatic control from the point of view of pattern recognition; decision theory, trainable threshold logic, stochastic approximation, and Markov chain theory are elaborated upon; includes an extensive bibliography.

PART 2

FINE ATTITUDE CONTROL OF A SPACE CRAFT OPERATING IN A PARTIALLY KNOWN ENVIRONMENT

CONTENTS

NOMENCLATURE

$A(t)$	plant matrix, abbreviated A
$A_a(t)$	augmented plant matrix, abbreviated A_a
$a(t)$	plant parameter
a	specific value of $a(t)$
$\underline{b}$	control distribution vector
b	component of $\underline{b}$
$\underline{b}_a$	augmented control distribution vector
$C(t)$	covariance matrix
C	normalization constant
C^j	polar moment of inertia for j^{th} inertia wheel
C_d	discrete disturbance-noise covariance matrix
$\overline{C}_d$	design value for C_d
C_0	rejection coefficient
C_1	detection coefficient; design constant
$C_2, \ldots, C_5$	design constants
c	dummy variable; speed of light
$\underline{d}$	disturbance distribution vector
d	component of $\underline{d}$
$\underline{d}(k)$	mean value vector
d_B	differential as seen by an observer fixed to main-body axes
$f(T)$	weighting function
$f_1(\underset{\sim}{a}, \underset{\sim}{\alpha}, \underset{\sim}{\rho})$, $f_2(\underset{\sim}{a}, \underset{\sim}{\alpha}, \underset{\sim}{\rho})$	sensitivity coefficients
$\underline{f}_a$	augmented disturbance distribution vector

$g_j\ (\underline{P}_{A_a})$	j^{th} discriminant
$H(s)$	transfer function
I	moment of inertia (with appropriate subscripts); identity matrix
K	universal gravitational constant; design constant
$\overline{K}^{*}(n)$	sub-optimal gain matrix
$K^{*}(n)$	optimal gain matrix
k_m	maximum number of iterations for m^{th} entry in a control situation
$\underline{L}$	augmented control distribution vector
$L_i(t)$	torque on i^{th} body axis
$\ell(t)$	external perturbation disturbance torque
$\ell_i(t)$	external perturbation disturbance torque acting on i^{th} body axis
M	observation matrix; bound
M^{*}	augmented observation matrix
$M(\underset{\sim}{a}),\ M(\underset{\sim}{\alpha})$	control situation bounds
$m_i(k)$	conditional expected value
$\underline{n}(t)$	white measurement noise vector
$P(t)$	gain matrix (in Riccati equation) for optimal control
$P(n\mid n)$	covariance matrix
$\underline{P}_{A_a}(t)$	augmented plant parameter vector
$\underset{\sim}{P}^{\ell}_{A_a}$	augmented plant parameter vector associated with center of ℓ^{th} control situation
$\hat{\underline{P}}'_{A_a}(t\mid t)$	pseudo-estimate of $\underline{P}_{A_a}(t)$
$\underline{P}^{i,k}_{A_a}$	one of m_i augmented plant parameter vectors $(k = 1, \ldots, m_i)$ whose control is $\underline{P}^{i}_{\Lambda}$
$\underline{P}_{\Lambda}(t)$	feedback parameter vector
$\underline{P}^{\ell}_{\Lambda}$	nominal feedback parameter vector associated with ℓ^{th} control situation
$\underline{P}^{*\ell}_{\Lambda}(k;t)$	k^{th} iterate of on-line-optimal feedback parameter vector associated with ℓ^{th} control situation

$\underline{P}_{\Lambda}^{*\ell}(k)$	amplitude of $\underline{P}_{\Lambda}^{*\ell}(k;t)$ for $kT < t < (k + 1)T$
$p_i(k)$	probability of being in ith state during kth interval of time
$p_{ij}(t)$	components of P(t)
p_{ℓ}	on-line or off-line probability that $\lvert\ell(t)\rvert \leq 10^{-4}$ lb - ft
PI(k, ℓ ; m)	on-line sub-goal performance index
$\overline{PI}$ (k, ℓ; m)	overall goal or performance index
$\tilde{PI}$	last measured value of PI for which $\lambda_i^{*\ell} = \tilde{\lambda}_i$
Q	weighting matrix
Q(n)	plant matrix in augmented a priori estimation equation
Q*(n)	plant matrix in augmented a prior error equation
q	component of Q
$\underline{q}(t)$	expanded state vector
R(t)	positive definite matrix
R_d	discrete measurement-noise covariance matrix
$\overline{R}_d$	design value for R_d
r	weighting factor; number of control choices; function of ρand a
r_{11}, r_{22}	components of R
r_{d11}, r_{d22}	components of R_d
$S_{\underline{P}}$	space of parameter vectors (with appropriate subscripts and superscripts)
$S_{x_1x_1}$	power spectrum of $x_1(t)$
$S_{\nu\nu}$	power spectrum of $\nu(t)$
s	complex variable; radius of hypersphere
T	decision time
T_s	iteration time in discrete estimation procedure
T(c)	auxiliary function used to indicate whether or not on-line optimizations for the cth control situation have been concluded
T(k)	transformation matrix
t	real time (only used in Appendix A)

t'	scaled time denoted t for notational simplicity
$u^*(t)$	on-line optimal control defined in Figure 3, Part 2
$u_1(t)$, $u_1^*(t)$	control and optimal control defined on and for the system in Figure 1, Part 2
$u_2(t)$, $u_2^*(t)$	control and optimal control defined on and for the system in Figure 2, Part 2
u_{2F}^*	u_2^* with loop closed around filter
u_{2P}^*	u_2^* with loop closed around plant
$u_3^*(t)$	optimal control for noise-free regulator problem
$u_{-1}(t)$	unit step function
$u_j(t)$	control acceleration acting on j^{th} inertia wheel
V	stochastic cost function; positive multiplier in disturbance noise covariance, in the stationary case
$V(t)$	positive function in disturbance noise covariance
V_d	component of C_d
$W(t)$	positive definite matrix
W_1	weighting matrix
$\underline{w}(t)$	vector white noise process
w_{ji}	weights for linear threshold element
$\underline{x}(t)$	state vector, abbreviated $\underline{x}$
$\underline{x}_a(t)$	augmented state vector, abbreviated $\underline{x}_a$
x_1, x_2	components of $\underline{x}(t)$
x_{1FP}	component of $x_1(t)$ due to $(u_{2F}^* - u_{2P}^*)$
$\underline{y}(t)$	observation vector
$y(k)$	decision function
$Z(k,\ell;m)$	normalized cost function
$\underline{z}(t)$	measurement vector

$\alpha(t)$	disturbance torque parameter
α	specific value of $\alpha(t)$
$\underline{\beta}(k)$	pseudo-gradient vector
$\beta_i(k)$	components of $\underline{\beta}(k)$
$\Gamma(t)$	parameter matrix
γ	weighting factor in linear reinforcement average
$\gamma_1(t), \gamma_2(t)$	components of $\Gamma(t)$
γ_i	limiting value for $E \{p_i(k)\}$
$\underline{\Delta}(n+1, n)$	discrete control distribution vector
$\delta(t-\tau)$	impulse function
$\delta(\epsilon)$	prespecified function of allowable attitude error
δ_B	perturbation of an independent variable as seen by an observer fixed to main-body axes
$\underline{\delta}_i$	uniformly distributed feedback-parameter perturbation vector
ϵ	allowable attitude error
$\epsilon'(k)$	threshold function
ϵ'	constant in $\epsilon'(k)$
ζ	damping ratio
$\zeta_i(t)$	perturbation rotation of i^{th} body axis from reference axes
$\eta_i(t)$	orientation of i^{th} reference axis with respect to orbit-plane axes
$\underline{\Theta}(t)$	vector white noise process
θ	learning parameter
$\Lambda(t)$	feedback gain matrix
$\Lambda^{*\ell}(t)$	on-line-optimal feedback gain matrix for ℓ^{th} control situation
$\lambda_1, \lambda_2, \lambda_3$	components of Λ
$\lambda_1^{*\ell}, \lambda_2^{*\ell}, \lambda_3^{*\ell}$	components of $\Lambda^{*\ell}$
$\tilde{\lambda}_i$	most recent value of $\lambda_i^{*\ell}$ that is different from $\lambda_i^{*\ell}(k)$

M	positive definite matrix
$\underline{\mu}$	random vector
μ_{d11}, μ_{d22}	components of C_d
$\nu(t)$	scalar white noise process
$\xi(t)$	disturbance state
$\underline{\pi}(k)$	k^{th} change in $\underline{P}^{*\ell}_{\Lambda}(k; t)$
$\tilde{\underline{\pi}}$	most recent value of $\underline{\pi}$ that is different from $\underline{\pi}(k)$
ρ	weighting factor
σ	distance of origin of main-body axes from origin of inertial axes
$\sigma(t)$	covariance matrix for optimal estimator
σ_{ℓ}, σ_{ξ}, σ_{x_1}	standard deviations
$\Phi(n+1, n)$, $\Phi^*(n+1, n)$	transition matrices
$\underline{\phi}^*(t_1)$	augmented vector white noise process
$\underline{\phi}^*_d(n+1, n)$	discrete vector random process
ω	circular orbital frequency
$\omega^j(t)$	angular velocity of j^{th} inertia wheel with respect to main-body axes
ω_n	natural frequency

$\hat{\bullet}(t_1 \mid t)$	estimate of ●
$\tilde{\bullet}(t_1 \mid t)$	error in estimate of ●
$\underset{\sim}{\bullet}$	value of ● associated with nominal control for a control situation
$\bullet^{(k)}$	k^{th} realization of the random variable ●
$\bullet^{(1)}(k)$	realization of the random process ● (k)
$(\quad)^T$	transpose of ( )

$(\quad)^{-1}$	inverse of ( )
$\vert(\quad)\vert$	absolute value of ( )
$\Vert(\quad)\Vert$	Euclidean norm of ( )
$\int_0^{\infty}(\quad)\,dt$	definite integral of ( )
$E\{(\quad)\}$	expected value of ( )
$E\{\ \mid\ \}$	conditional expected value
max ( )	maximum of ( )
sgn ( )	signum function
$\sum$	summation
∞	infinity
$\delta(\quad)$	variation
$\frac{\partial}{\partial\xi}$	partial derivative
$\Pr\{\ \}$	probability
$\{\quad\}_{j=1}^{J}$	set of J elements, indexed j = 1, ..., J
$(\quad \vdots \quad)^{T}$	partitioned vector
$\left(\begin{array}{c\vert c} - & - \end{array}\right)$	partioned matrix

Section 1

INTRODUCTION

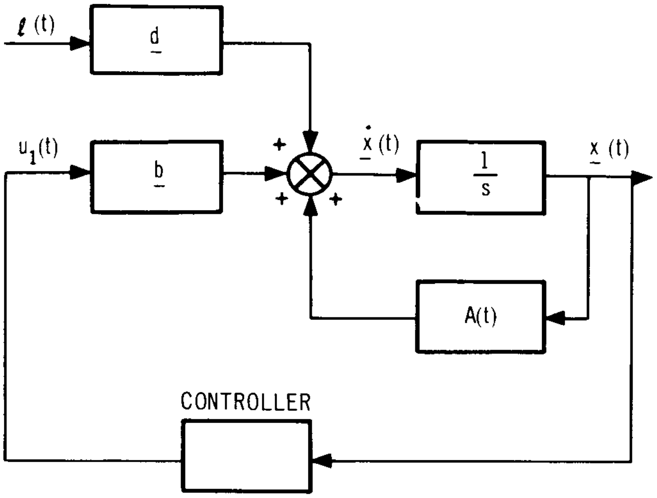
The purpose of this report is to describe the results of a study in which artifical intelligence techniques are applied during the preliminary design of a spacecraft attitude control system. Answers to questions about the feasibility of designing an attitude control system using the present state of technology in artificial intelligence techniques are sought in the context of a specific spacecraft application. New artificial intelligence techniques are not developed, since it is not within the scope of the present study to do so.

The artificial intelligence techniques considered are: (1) on-line learning, which embraces the concepts of: (a) control situations, (b) on-line cost functions, and (c) on-line learning algorithms; and (2) off-line learning, which embraces pattern recognition. The unique feature of the present study is that all of these techniques are used simultaneously.

The specific spacecraft application considered is the fine attitude control of an almost cylindrically symmetrical satellite in Earth or Mars orbit, such as a laser communication satellite in Mars orbit. By fine attitude control is meant that mode of control in which attitude errors are brought from within ±M arcsec (about 80 arcsec) to within $\pm\epsilon$ arcsec (e.g., ±0.20 arcsec for a laser communication satellite). It must be noted that 1 arcsec is equivalent to 4.85×10^{-6} radians. The distinctions between coarse and fine attitude control and fine-pointing control are found in Appendix A (at this point it is recommended that Appendix A be read in its entirety). The property of cylindrical symmetry initially permits the study of the control of a second-order system. This system is linear, time-varying and, in addition, is excited by a disturbance torque $\ell(t)$ which, because of a lack of or an incomplete knowledge about it, is assumed to be a poorly defined stochastic process.

The system is controlled by a controller (inertia wheel) with an output signal that is proportional to the states of the system (fig. 1). The overall goal of the controller is to keep attitude errors (x_1) very small, less than ±0.20 arcsec. Since $\ell(t)$ is a stochastic process, $\underline{x}(t)$ is a stochastic process; hence, one way to properly control the system might be to choose $u_1(t)$ such that

$$E\left\{\int_0^\infty \left[\underline{x}^T W_1 \underline{x} + u_1^2\right] dt\right\}$$



(a)

$$\dot{\underline{x}}(t) = A(t)\,\underline{x}(t) + \underline{b}\,u_1(t) + \underline{d}\,\ell(t)$$

$$\underline{x}(t) = (x_1, x_2)^T$$

$u_1(t)$ AND $\ell(t)$ ARE SCALARS

$$A(t) = \begin{pmatrix} 0 & 1 \\ -a(t) & 0 \end{pmatrix}$$

$$\underline{b} = (0 \quad b)^T$$

$$\underline{d} = (0 \quad d)^T$$

(b)

Figure 1. Second-Order System Disturbed by Perturbation Torques $\ell(t)$. (a) Block Diagram, (b) Mathematical Relationships

is minimized. If $\ell(t)$ and a (t) were known ahead of time precise descriptions of them could be incorporated into this optimization problem; the resulting optimal control, $u_1^*(t)$, would then not only be optimal with respect to the model of the second-order system but would also be optimal with respect to the actual system, since the two would be one and the same.

Since $\ell(t)$ is not known ahead of time an approximation of it is used in the optimization procedure. Specifically, $\ell(t)$ is approximated by a first-order Markov process (see fig. 2). In this case, the resulting optimal control $u_2^*(t)$ [to be distinguished from $u_1^*(t)$], while optimal with respect to the model in fig. 2, is not necessarily optimal with respect to the actual system in fig. 1 in which it will ultimately be used. In short, $u_2^*(t)$ is sub-optimal with respect to the actual system; hence, even though $u_2^*(t)$ may cause the system in fig. 2 to respond satisfactorily it may not sufficiently reduce the attitude errors in the actual system. A poor response may occur in this case as a result of discrepancies between $\ell(t)$ and its first-order model, discrepancies which, due to a lack of a priori knowledge about $\ell(t)$, cannot be adjusted until the system is in actual operation (in orbit around Mars, for example). These adjustments are accomplished through on-line learning.

Basic premise: For on-line learning to improve a system's performance during the actual operation of the system new information must be made available to an on-line performance assessor; that is to say, information which was not available a priori (off-line) must be utilized by an on-line performance assessor in making the decisions how or if the sub-optimal controls should be modified.

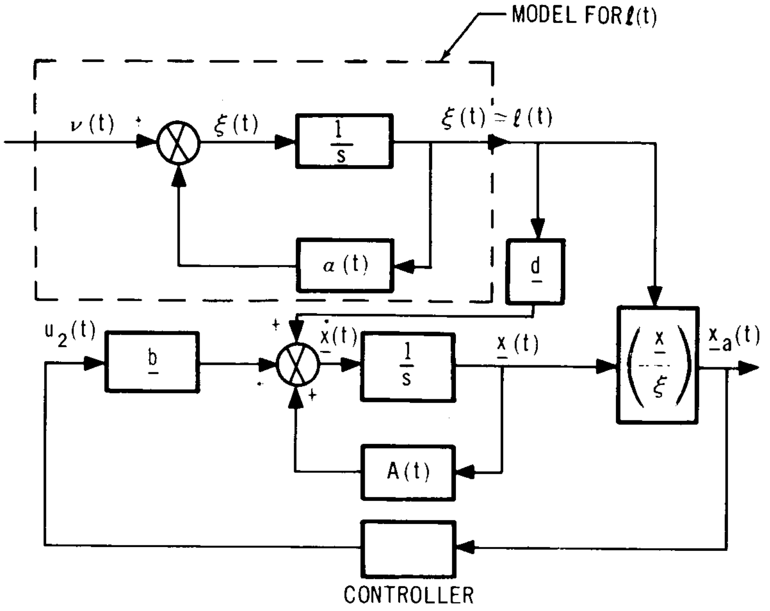
An on-line-learning controller is depicted in fig. 3 for the actual system shown in fig. 1. The on-line-learning controller represents an extension of the controller depicted in fig. 2; an on-line goal circuit (performance assessor) is now available for making decisions as to whether or not the on-line-optimal control, $u^*(t)$, should be updated. The design of the on-line-optimal control proceeds in two steps:

(1) The sub-optimal controls $u_2^*(t)$ (fig. 2) are designed for different combinations of the augmented plant parameters, a(t) and $\alpha(t)$. These controls use all of the available information about the system and its actual environment. In the sequel, these controls are referred to as nominal controls.

(2) The nominal controls are updated on-line if updating is necessary. This updating is accomplished through on-line learning.

Initially $u^*(t) = u_2^*(t)$. The new information which serves as the basis for the decisions made on-line by the on-line goal circuit (fig. 3) comes from measurements of the actual attitude errors. These measurements provide information about the partially known environment. After the measurements are made they are processed in the form of an on-line cost function which is then used in an on-line-learning algorithm for updating the control.

It was mentioned above that the on-line-learning controller represents an extension of the controller depicted in fig. 2. Unfortunately, the model in



(a)

$$\dot{\underline{x}}_a(t) = A_a(t)\,\underline{x}_a(t) + \underline{b}_a\, u_2(t) + \underline{f}_a\, \nu(t)$$

$$\underline{x}_a(t) = (x_1, x_2, \xi)^T$$

$u_2(t)$ and $\nu(t)$ ARE SCALARS

$$A_a(t) = \left(\begin{array}{c|c} A(t) & \underline{d} \\ \hline 0 & \alpha(t) \end{array}\right)$$

$$\underline{b}_a = (\underline{b} \mid 0)^T$$

$$\underline{f}_a = (0 \mid 1)^T$$

$$E\{\nu(t)\} = 0 \text{ AND } E\{\nu(t)\nu(\tau)\} = V(t)\delta(t-\tau)$$

(b)

Figure 2. Second-Order System Where Actual Disturbance Torque is Modeled by a First-Order Markov Process. (a) Block Diagram, (b) Mathematical Relationships

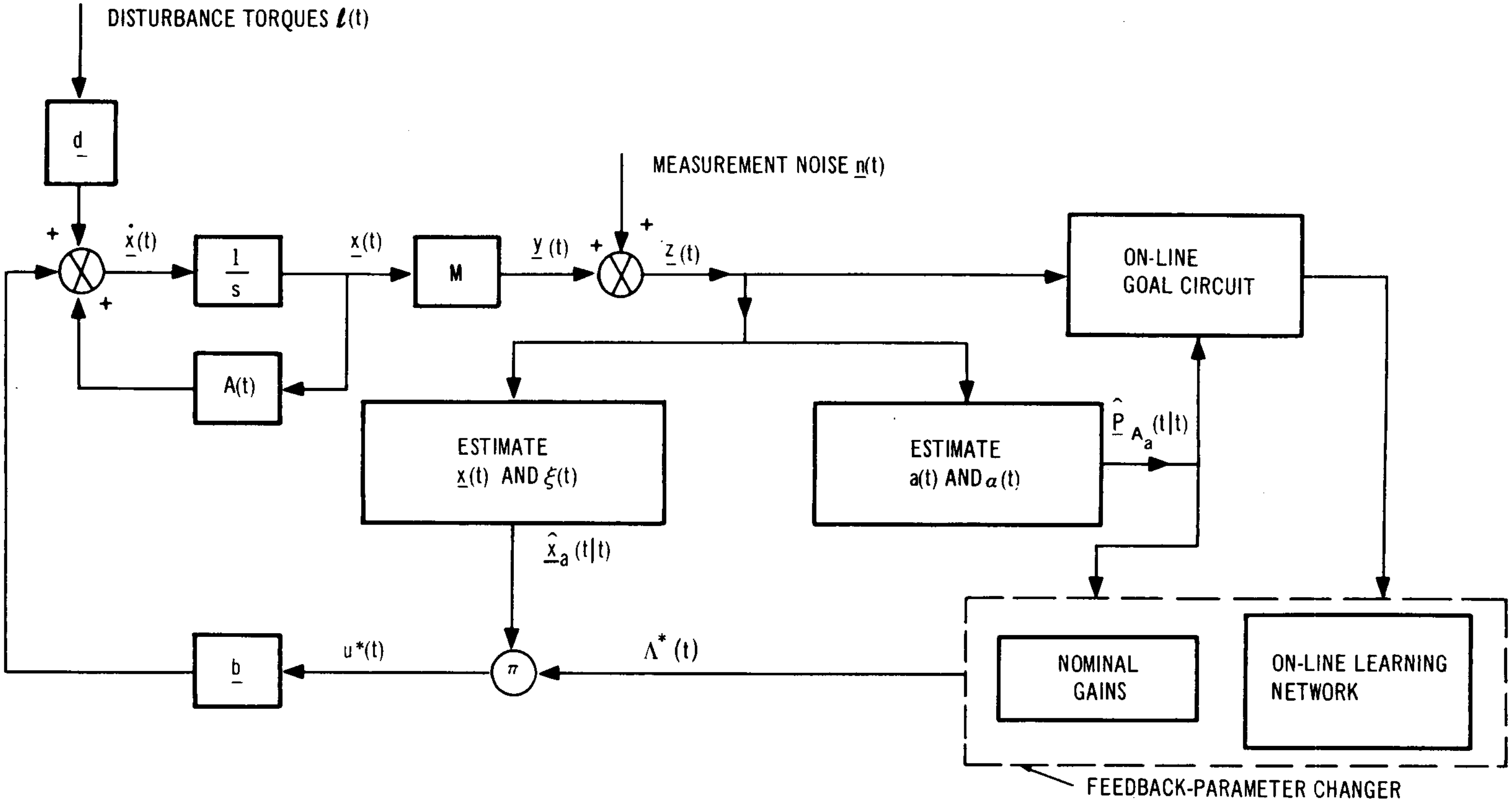


Figure 3. On-Line-Learning Control System

fig. 2 is an oversimplification of the real situation. In reality, it is not possible to observe $\xi(t)$; hence, $\xi(t)$ must first be estimated and then the estimated value of $\xi(t)$ is fed back as part of $u^*(t)$ [fig. 3; or $u_2(t)$ in fig. 2] In addition, it may not be possible to measure $x_1(t)$ and $x_2(t)$ exactly (if, indeed, both states are measurable); that is to say, the measurements may be corrupted with noise. This seems likely in the laser communication satellite application, since $x_1(t)$ and $x_2(t)$ are quite small. $x_1(t)$ and $x_2(t)$ may, therefore, also have to be estimated, in which case the estimated values are fed back as part of $u^*(t)$ (the assumption that it is possible to separate the estimation and control problems is discussed in later sections of this report).

Hence, in order to mechanize the control $u^*(t)$ estimates of $x_1(t)$, $x_2(t)$ and $\xi(t)$ must be obtained from noisy measurements. In addition, estimates of the parameters a(t) and $\alpha(t)$, which characterize the plant and disturbance torque, respectively, are obtained. They are obtained for two reasons: (1) to provide the initial choice for $u^*(t)$, since $u_2^*(t)$ is precomputed for different combinations of a(t) and $\alpha(t)$, and (2) to implement a control situation concept which is used during on-line-learning.

For the mechanization of the on-line-learning system in fig. 3, the following sub-problems are considered in the sequel in greater detail:

(1) Evaluation of nominal controls

(2) Estimation of states and parameters

(3) Development of an on-line-learning procedure

The above sub-problems are discussed in Sections 2, 3, and 4 of this part, respectively. Experiments which test the proposed on-line-learning procedure are described also in Section 4 of this part. Conclusions and suggestions for further work appear in Section 5 of this part.

Section 2

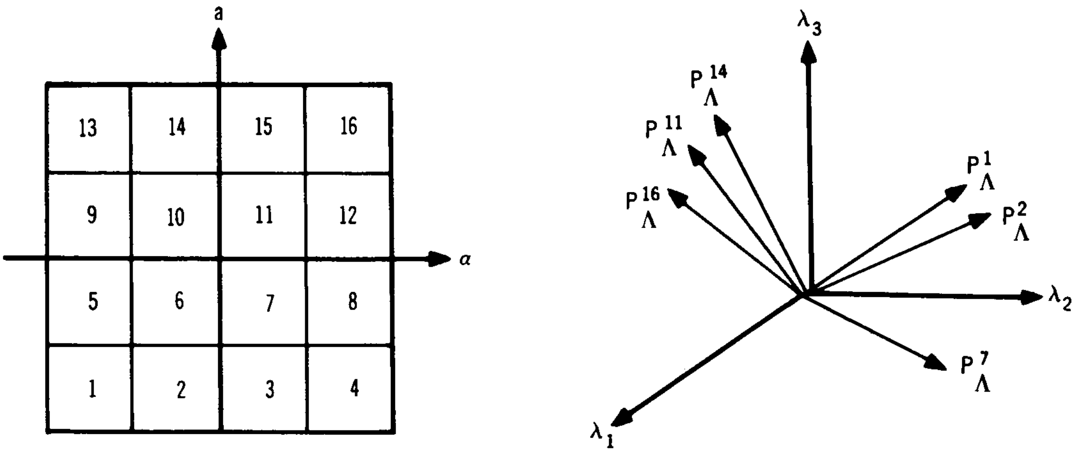
NOMINAL CONTROLS AND NOMINAL CONTROL SITUATIONS

Introduction

Controls, in the design of which all the available information about the system and the environment is incorporated are referred to as nominal controls. The nominal controls for the system in fig. 3 are obtained by formulating and solving an optimal control problem for the system in fig. 2 in which all of the available information about the system and the associated environment has been included. These controls are proportional to the augmented state vector $\underline{x}_a(t)$, as required.

The details of the design of the nominal controls, for specific values of the plant parameters, appear in Appendix C. It is recommended that Appendix C be read in its entirety at this point. For completeness in the present exposition, the results of the design and the major assumptions made during the design are presented in the next subsection. There the nominal controls are obtained by freezing the plant parameters a(t) and $\alpha(t)$ [specific values of a(t) and $\alpha(t)$ are denoted a and α, respectively], after assuming that ranges for both parameters are known a priori. Since it is not very feasible to evaluate the nominal controls for every combination of a and α, a practical realization of the nominal controls must be found. The usual approach is to partition the augmented plant parameter space (the a-α space) into squares (hypercubes in a higher dimensional augmented plant parameter space) and to associate a single control with each square. Such as association, or mapping, is depicted in fig. 4. The partitions, loosely speaking, relate to the concept of control situations. This is elaborated upon in the subsection entitled "Partitioning of Augmented Parameter Space." A simulation of the system in fig. 2 is discussed in the subsection titled "Simulation Studies." The purpose of the simulation was to observe the effects of the numerous assumptions made during the design and realization of the nominal control on the attitude error (which must be kept within ±0.20 arcsec).

The controller designed in this section is an adaptive controller in that feedback gains are changed if plant parameters change. In addition, the feedback gains for the time-varying system are time-varying, even though the nominal gains are not. This is a result of the adaptivity of the nominal controller.



(a)

(b)

Figure 4. (a) Partition of Augmented Plant Parameter Space and (b) Nominal Feedback Parameter Vectors † $\underline{P}^i_\Lambda$, Each Associated With an ith Partition in (a), i = 1, ..., 16.

Summary of Design

In Appendix C, nominal controls are designed for the system in fig. 2, when a(t) and α(t) are held constant, by optimizing a cost function which is quadratic in the attitude error and control. The optimal control problem is a stochastic optimal control problem, since the attitude error, $x_1(t)$, is a stochastic process. Rather than solve the stochastic problem directly, the solution is obtained in two steps:

(1) $\underline{x}_a(t)$ is estimated by filtering theory (discussed in Section 3 and Appendix B)

(2) The optimal control for the noise-free regulator problem $\left[u_3^*(t)\right]$ is obtained.

$u_2^*(t)$, in fig. 2, is then obtained from items 1 and 2 according to the Separation Theorem (ref. C1). Three assumptions are made in this solution for $u_2^*(t)$:

(1) Nominal controls for the time-varying system can be found by employing a design procedure in which the time-varying parameters are frozen initially at specific values

†$\underline{P}_\Lambda = (\lambda_1\ \lambda_2\ \lambda_3)^T$; see eq. (C8).

(2) The Separation Theorem may be applied when states and parameters are estimated using a nonlinear estimator.

(3) Ranges on a(t) and α(t) are known a priori.

As shown in Appendix C,

$$u_2^*(t) = \Lambda \, \hat{\underline{x}}_a (t|t) \tag{1}$$

where

$$\Lambda = \underline{P}_\Lambda^T = (\lambda_1 \, \lambda_2 \, \lambda_3) \tag{2}$$

$$\lambda_1 = -\left(a - \sqrt{a^2 + \rho}\right) \tag{3}$$

$$\lambda_2 = \sqrt{2\left(\sqrt{a^2 + \rho} - a\right)} \tag{4}$$

$$\lambda_3 = \frac{-1000\left[\alpha\sqrt{2\left(\sqrt{a^2+\rho} - a\right)} - \sqrt{(a^2+\rho)} + a\right]}{\alpha^2 + \sqrt{(a^2+\rho)} - \alpha\sqrt{2\sqrt{(a^2+\rho)} - a}} \tag{5}$$

and ρ is a weighting factor that is directly proportional to control effort and inversely proportional to attitude error. This weighting factor is designed for a worst-case design in which the disturbance noise $\nu(t)$ is replaced by $\pm 3\sqrt{V}$ (see fig. 2) and attitude errors are constrained to remain less than or equal to $\pm\epsilon$ (± 0.2 arcsec) in the face of this step-type disturbance. The following inequality results for ρ:

$$r^4 + 2C_2 r^3 + (2C_1^2 + C_2^2) r^2 + 2C_1^2 (C_2 - 2\alpha^4) r + C_1^4 \le 0 \tag{6}$$

where

$$\rho = r - a^2 \tag{7}$$

$$C_1 = \frac{3000\sqrt{V}}{\epsilon} \left|\frac{\alpha^2 + a}{\alpha}\right| \tag{8}$$

and

$$C_2 = \alpha^2 (2a + \alpha^2) - 2C_1 \tag{9}$$

The solutions for ρ in eq. (6) are obtained as described in Appendix C. As these solutions are for specific values of a and α, and since this section is primarily interested in a single value of ρ for regions in the augmented plant parameter space, due to the time-varying nature of a and α, these solutions shall not be elaborated here. Instead, the section shall indicate how eq. (6) can be used to obtain a single value of ρ, and subsequently a single feedback parameter vector, for regions in the $a - \alpha$ space, as in fig. 4.

Control Situations for Nominal Controls

Here the nominal controls are obtained for specific control situations.

A *control situation* is defined as *a* region in the augmented plant parameter space for which a single control (feedback parameter vector) leads to satisfactory performance for all points contained therein.

One might assume that a control situation can be constructed quite arbitrarily about the point at which the nominal control (for all points within the control situation) is calculated. In fig. 5, for example, the nominal control is computed at $(a, \alpha) = (\underset{\sim}{a}, \underset{\sim}{\alpha})$ and either a square or circle is constructed, each centered at $(\underset{\sim}{a}, \underset{\sim}{\alpha})$. It is incorrect to assume that these figures are control situations, in the sense of the definition above, since there is no guarantee that all points within these figures will have a satisfactory performance associated with them. In general, an *admissible region* for variations in a and α about $(\underset{\sim}{a}, \underset{\sim}{\alpha})$ must first be determined and then a control situation within the admissible region must be constructed. The concept of an admissible region relates the phrase satisfactory performance, in the definition of a control situation, to the boundary of a control situation.

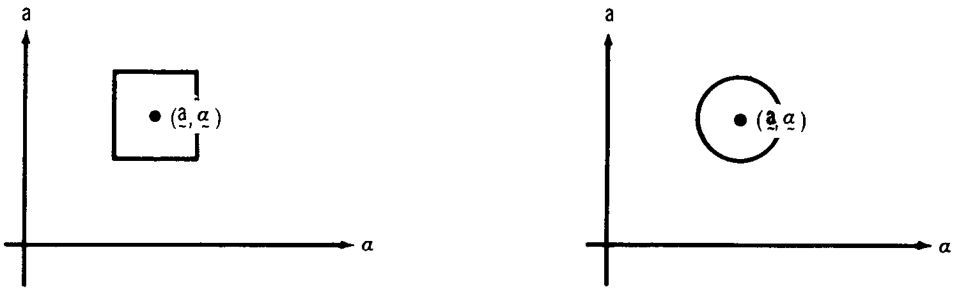

a
$(\underset{\sim}{a}, \underset{\sim}{\alpha})$
α
a
$(\underset{\sim}{a}, \underset{\sim}{\alpha})$
α

Figure 5. Arbitrary Partitions in Augmented Plant-Parameter-Space

An admissible region is defined as the region in the augmented plant parameter space (subject to constraints on a and α) for which the system performs satisfactorily under the action of a single feedback parameter vector (control). A control situation, in general, is a sub-set of an admissible region. It follows, therefore, that before control situations can be constructed the admissible regions must be determined.

Admissible Regions. --The concept of an admissible region relates to satisfactory system performance. Here, the following is assumed:

Satisfactory performance is achieved if, when $\nu(t)$ is replaced by $\pm 3\sqrt{V}$, as in the section titled the "Worst-Case Design for Weighting Factor ρ," in Appendix C,

$$\left| x_1 \ (\infty) \right| \leq \epsilon = 0.20 \text{ arcsec} \qquad (10)$$

As pointed out in the preceding section, eq. (10) is equivalent to the inequality in eq. (6). For the purposes of the present analysis, it is convenient to let

$$C_1 = \frac{C_3}{\alpha} \qquad (11)$$

and

$$C_2 = \frac{C_4}{\alpha}, \qquad (12)$$

where

$$C_3 = K(a + \alpha^2) \qquad (13)$$

$$K = \frac{-3000\sqrt{V}}{\epsilon} \qquad (14)$$

and

$$C_4 = \alpha^3 \ (2a + \alpha^2) - 2\,C_3. \qquad (15)$$

In the derivations of eqs. (11) to (14), it was assumed that $\alpha<0$ and that $(a + \alpha^2)>0$. The results obtained below are easily generalized to include the case when $(a + \alpha^2)<0$. Upon substitution of eqs. (11) and (12) into eq. (6), the latter equation becomes

$$\alpha^4 r^4 + 2C_4\alpha^3 r^3 + \alpha^2 \ (2C_3^2 + C_4^2)\ r^2 + 2\alpha C_3^2 \ (C_4 - 2\alpha^5)\ r + C_3^4 \leq 0 \quad (16)$$

This inequality is the starting point in the determination of admissible regions.

Let $(\underset{\sim}{a}, \underset{\sim}{\alpha})$ be the values of a and α at which the nominal control for a control situation is computed. Associated with this nominal control is a value of ρ, designated $\underset{\sim}{\rho}$, which is found from eq. (16), with $a = \underset{\sim}{a}$ and $\alpha = \underset{\sim}{\alpha}$; that is to say, $\underset{\sim}{\rho}$ is a solution of the inequality

$$\underset{\sim}{\alpha}^4 \underset{\sim}{r}^4 + 2\underset{\sim}{C}_4 \underset{\sim}{\alpha}^3 \underset{\sim}{r}^3 + \underset{\sim}{\alpha}^2 (2\underset{\sim}{C}_3^2 + \underset{\sim}{C}_4^2) \underset{\sim}{r}^2 + 2\underset{\sim}{\alpha}\underset{\sim}{C}_3^2 (\underset{\sim}{C}_4 - 2\underset{\sim}{\alpha}^5) \underset{\sim}{r} + \underset{\sim}{C}_3^4 \le 0 \quad (17)$$

where $\underset{\sim}{C}_3$ and $\underset{\sim}{C}_4$ are evaluated by replacing a and α with $\underset{\sim}{a}$ and $\underset{\sim}{\alpha}$ in eqs. (13) and (15), respectively.

Within each control situation*

$$a = \underset{\sim}{a} + \delta a \qquad -M_1(\underset{\sim}{a}) \le \delta a \le M_2(\underset{\sim}{a}) \quad (18)$$

and

$$\alpha = \underset{\sim}{\alpha} + \delta\alpha \qquad -M_3(\underset{\sim}{\alpha}) \le \delta\alpha \le M_4(\underset{\sim}{\alpha}) \quad (19)$$

It must be observed, that the bounds on δa and $\delta\alpha$ are, in general, dependent upon the specific values of $\underset{\sim}{a}$ and $\underset{\sim}{\alpha}$; however, as there is no apparent procedure for choosing these different bounds, it is assumed that

$$M_1(\underset{\sim}{a}) = M_2(\underset{\sim}{a}) = M_3(\underset{\sim}{\alpha}) = M_4(\underset{\sim}{\alpha}) = M \quad (20)$$

An additional assumption is that M is small; hence, δa and $\delta\alpha$ are small variations in a and α. Eq. (20) represents the constraints on a and α referred to in the definition above of an admissible region.

Now, for all points within an admissible region (and, hence, a control situation) eq. (16) must be satisfied. For these points, if one neglects second- and higher-order effects (δa and $\delta\alpha$ are small), r, C_3 and C_4 become:

$$r = \underset{\sim}{r} + 2\underset{\sim}{a}\,\delta a \quad (21)$$

$$C_3 = \underset{\sim}{C}_3 + K(\delta a + 2\underset{\sim}{\alpha}\,\delta\alpha) \triangleq \underset{\sim}{C}_3 + \delta C_3 \quad (22)$$

$$C_4 = \underset{\sim}{C}_4 + 2\underset{\sim}{\alpha}^3 \delta\alpha + (5\underset{\sim}{\alpha}^4 + 6\underset{\sim}{a}\underset{\sim}{\alpha}^2)\,\delta\alpha - 2\,\delta C_3 \quad (23)$$

Upon substitution of eqs. (21) to (23) and powers of these equations, in which second- and higher-order effects are neglected, into eq. (16), one obtains the following inequality for δa and $\delta\alpha$:

$$f_1(\underset{\sim}{a}, \underset{\sim}{\alpha}, \underset{\sim}{\rho})\,\delta a + f_2(\underset{\sim}{a}, \underset{\sim}{\alpha}, \underset{\sim}{\rho})\,\delta\alpha + \underset{\sim}{C}_5 \le 0 \quad (24)$$

*To be more precise, the notation $\underset{\sim}{a}^{\ell}$ and $\underset{\sim}{\alpha}^{\ell}$, which denotes the ℓth control situation, should be used; however, for notational simplicity, this will be assumed, but not used.

where

$$\begin{aligned} f_1\,(\underset{\sim}{a}, \underset{\sim}{\alpha}, \underset{\sim}{\rho}) = {} & (8\underset{\sim}{a}\,\underset{\sim}{\alpha}^4 + 4\underset{\sim}{\alpha}^6 - 4K\,\underset{\sim}{\alpha}^3)\,\underset{\sim}{r}^3 + (12\underset{\sim}{a}\,\underset{\sim}{\alpha}^3\,\underset{\sim}{C}_4 + 4\underset{\sim}{\alpha}^2\,\underset{\sim}{C}_3K + \\ & 4\underset{\sim}{\alpha}^5\,\underset{\sim}{C}_4 - 4\underset{\sim}{\alpha}^2\,\underset{\sim}{C}_4K)\,\underset{\sim}{r}^2 + (8\underset{\sim}{a}\,\underset{\sim}{\alpha}^2\,\underset{\sim}{C}_3^{\,2} + 4\underset{\sim}{a}\,\underset{\sim}{\alpha}^2\,\underset{\sim}{C}_4^{\,2} + 4\underset{\sim}{\alpha}^4\,\underset{\sim}{C}_3^{\,2} + \\ & 4\underset{\sim}{\alpha}\,\underset{\sim}{C}_3\,\underset{\sim}{C}_4K - 8\underset{\sim}{\alpha}^6\,\underset{\sim}{C}_3K - 4\underset{\sim}{\alpha}\,\underset{\sim}{C}_3^2K)\,\underset{\sim}{r} + (4\underset{\sim}{a}\,\underset{\sim}{\alpha}\,\underset{\sim}{C}_3^2\,\underset{\sim}{C}_4 - \\ & 8\underset{\sim}{a}\,\underset{\sim}{\alpha}^6\,\underset{\sim}{C}_3^2 + 4\,\underset{\sim}{C}_3^{\,3}K) \end{aligned} \tag{25}$$

$$\begin{aligned} f_2\,(\underset{\sim}{a}, \underset{\sim}{\alpha}, \underset{\sim}{\rho}) = {} & 4\underset{\sim}{\alpha}^3\,\underset{\sim}{r}^4 + (6\underset{\sim}{\alpha}^2\,\underset{\sim}{C}_4 + 10\underset{\sim}{\alpha}^7 + 12\underset{\sim}{a}\,\underset{\sim}{\alpha}^5 - 8\underset{\sim}{\alpha}^4K)\,\underset{\sim}{r}^3 + \\ & (8\underset{\sim}{\alpha}^3\,\underset{\sim}{C}_3K + 10\,\underset{\sim}{C}_4\,\underset{\sim}{\alpha}^6 + 12\underset{\sim}{a}\,\underset{\sim}{\alpha}^4\,\underset{\sim}{C}_4 + 4\underset{\sim}{\alpha}\,\underset{\sim}{C}_3^{\,2} + 2\underset{\sim}{\alpha}\,\underset{\sim}{C}_4^{\,2} - \\ & 8\underset{\sim}{\alpha}^3\,\underset{\sim}{C}_4K)\,\underset{\sim}{r}^2 + (10\underset{\sim}{\alpha}^5\,\underset{\sim}{C}_3^{\,2} + 12\underset{\sim}{a}\,\underset{\sim}{\alpha}^3\,\underset{\sim}{C}_3^{\,2} - 20\underset{\sim}{\alpha}^5\,\underset{\sim}{C}_3^{\,2} + 2\underset{\sim}{C}_4\,\underset{\sim}{C}_3^{\,2} + \\ & 8\underset{\sim}{\alpha}^2\,\underset{\sim}{C}_3\,\underset{\sim}{C}_4K - 4\underset{\sim}{\alpha}^5\,\underset{\sim}{C}_3^{\,2} - 16\underset{\sim}{\alpha}^7\,\underset{\sim}{C}_3K - 8\underset{\sim}{\alpha}^2\,\underset{\sim}{C}_3^{\,2}K)\,\underset{\sim}{r} + 8\underset{\sim}{\alpha}\,\underset{\sim}{C}_3^{\,3}K \end{aligned} \tag{26}$$

and

$$\begin{aligned} \underset{\sim}{C}_5 = {} & \underset{\sim}{\alpha}^4\,\underset{\sim}{r}^4 + 2\underset{\sim}{C}_4\,\underset{\sim}{\alpha}^3\,\underset{\sim}{r}^3 + \underset{\sim}{\alpha}^2\,(2\underset{\sim}{C}_3^{\,2} + \underset{\sim}{C}_4^{\,2})\underset{\sim}{r}^2 + \\ & 2\underset{\sim}{\alpha}\,\underset{\sim}{C}_3^{\,2}(\underset{\sim}{C}_4 - 2\underset{\sim}{\alpha}^5)\,\underset{\sim}{r} + \underset{\sim}{C}_3^{\,4} \end{aligned} \tag{27}$$

Equation (24) together with the facts that $|\delta a| \leq M$ and $|\delta\alpha| \leq M$ define the admissible regions in the augmented plant parameter space.* The four possible regions are summarized in Table I. From these results one observes the following:

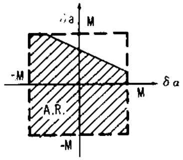
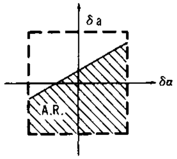
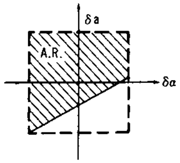
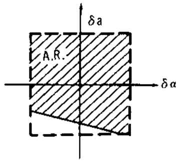
(1) The point $(\underset{\sim}{a}, \underset{\sim}{\alpha})$, at which the nominal control (for a control situation) is computed, is usually not at the center of the admissible region. In fact, if $\underset{\sim}{\rho}$ is computed from the equality in eq. (17), $\underset{\sim}{C}_5 = 0$ and the point $(\underset{\sim}{a}, \underset{\sim}{\alpha})$ is on the boundary of the admissible region.

(2) Once the admissible region is determined a square or circular control situation can be constructed within it. Illustrations of such constructions appear in fig. 6. It must be observed that $(\underset{\sim}{a}, \underset{\sim}{\alpha})$ does not even have to be contained within the control situation.

*Observe that $\underset{\sim}{C}_5 \leq 0$. This fact is made use of in Table I.

Table I

SUMMARY OF ADMISSIBLE REGIONS

$f_1(\underset{\sim}{a},\underset{\sim}{\alpha},\underset{\sim}{\rho})$	$f_2(\underset{\sim}{a},\underset{\sim}{\alpha},\underset{\sim}{\rho})$	Inequality in $\delta\alpha$ and δa[1]	Admissible Region (A. R.)[2]
Positive	Positive	$\delta a \le \left\|\frac{\underset{\sim}{C}_5}{f_1}\right\| - \left\|\frac{f_2}{f_1}\right\| \delta\alpha$	
Positive	Negative	$\delta a \le \left\|\frac{\underset{\sim}{C}_5}{f_1}\right\| + \left\|\frac{f_2}{f_1}\right\| \delta\alpha$	
Negative	Positive	$\delta a \ge - \left\|\frac{\underset{\sim}{C}_5}{f_1}\right\| + \left\|\frac{f_2}{f_1}\right\| \delta\alpha$	
Negative	Negative	$\delta a \ge - \left\|\frac{\underset{\sim}{C}_5}{f_1}\right\| - \left\|\frac{f_2}{f_1}\right\| \delta\alpha$	

[1] For notational convenience, $f_1(\underset{\sim}{a}, \underset{\sim}{\alpha}, \underset{\sim}{\rho})$ and $f_2(\underset{\sim}{a}, \underset{\sim}{\alpha}, \underset{\sim}{\rho})$ are written here as f_1 and f_2.

[2] When $\delta a = 0$, $a = \underset{\sim}{a}$, and when $\delta\alpha = 0$, $\alpha = \underset{\sim}{\alpha}$.

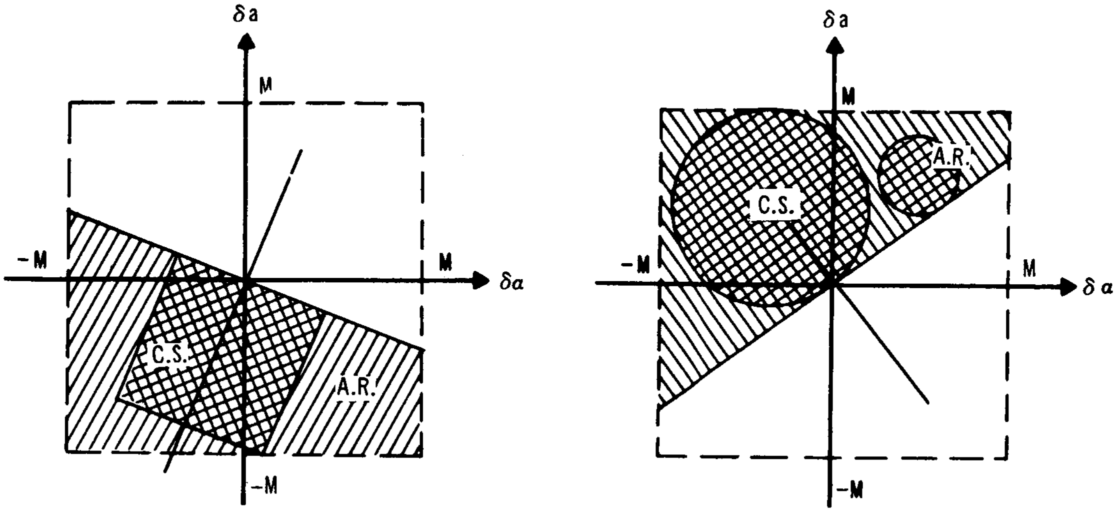


Figure 6. Construction of Square and Circular Control Situations (C.S.) Within the Admissible Regions (A.R.)

Control Situations. --Next, the conditions under which it is possible to center a square or a circular control situation at $(\underset{\sim}{a}, \underset{\sim}{\alpha})$ are investigated. It is assumed here that $\underset{\sim}{\rho}$ is evaluated from the inequality in eq. (17). The results below apply for each of the cases depicted in Table I; they are illustrated in figs. 7 and 8 only for the third case.

Circular Control Situation Centered at $(\underset{\sim}{a}, \underset{\sim}{\alpha})$: Using relatively straightforward geometric arguments, it is possible to show that for the condition in eq. (28) or (30), eq. (29) or (31), respectively, defines circular control situations that are centered at $(\underset{\sim}{a}, \underset{\sim}{\alpha})$. Two cases are depicted in fig. 7.

$$\text{(A) If } \left|\frac{\underset{\sim}{C}_5}{f_1}\right| < M\sqrt{1+\left(\frac{f_2}{f_1}\right)^2} \quad \triangleq \text{ CIRCLE} \qquad (28)$$

$$(\delta a)^2 + (\delta \alpha)^2 \leq \left(\frac{\underset{\sim}{C}_5}{\sqrt{f_1^{\,2} + f_2^{\,2}}} \right)^2 \qquad (29)$$

(B)

$$\text{If} \quad \left| \frac{\underset{\sim}{C}_5}{f_1} \right| \geq M \sqrt{1 + \left(\frac{f_2}{f_1} \right)^2} \qquad (30)$$

$$(\delta a)^2 + (\delta \alpha)^2 \leq M^2 \qquad (31)$$

Square Control Situation Centered at $(\underset{\sim}{a}, \underset{\sim}{\alpha})$: Again, using simple geometric arguments, it is possible to show that for the condition in eqs. (32) or (34), eqs. (33) or (35), respectively, defines square control situations that are centered at $(\underset{\sim}{a}, \underset{\sim}{\alpha})$. Two cases are depicted in fig. 8.

(A)

$$\text{If} \quad \left| \frac{\underset{\sim}{C}_5}{f_1} \right| < M \left(1 + \left| \frac{f_2}{f_1} \right| \right) \triangleq \text{SQUARE} \qquad (32)$$

$$\left| \delta a \right|, \left| \delta \alpha \right| \leq \frac{\left| \underset{\sim}{C}_5 \right|}{\left| f_1 \right| + \left| f_2 \right|} \qquad (33)$$

(B)

$$\text{If} \quad \left| \frac{\underset{\sim}{C}_5}{f_1} \right| \geq M \left(1 + \left| \frac{f_2}{f_1} \right| \right) \qquad (34)$$

$$\left| \delta \alpha \right|, \ \left| \delta \alpha \right| \leq M \qquad (35)$$

Discussion: It must be observed that, since $M\sqrt{1 + \left(\frac{f_2}{f_1}\right)^2} < M \left(1 + \left|\frac{f_2}{f_1}\right|\right)$, it is always possible to construct the maximum circular control situation (of radius M) before it is possible to construct the maximum square control situation (of length 2M). The maximum square control situation equals the admissible region, whereas the maximum circular situation excludes the corners of the admissible region; hence, there is a tradeoff between circular and square control situations.

Partitioning of Augmented Plant Parameter Space. --Here the above theory is applied to partition the augmented plant parameter space into square control situations which are centered at $(\underset{\sim}{a}, \underset{\sim}{\alpha})$ and which have a single value of $\underset{\sim}{\rho}$ associated with each of them. The specific ranges for a(t) and α(t) are:

$$-0.40 \leq a \leq 0.40 \qquad (36a)$$

$$-1.0 \leq \alpha \leq -0.10 \qquad (36b)$$

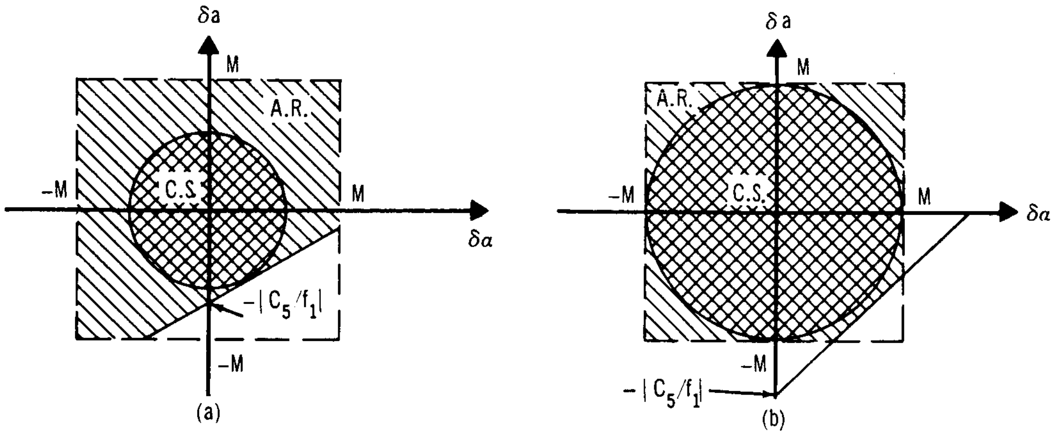


Figure 7. Construction of Circular Control Situations Centered at (a α), for Cases (A) and (B) in Eqs. (28) and(30), Respectively

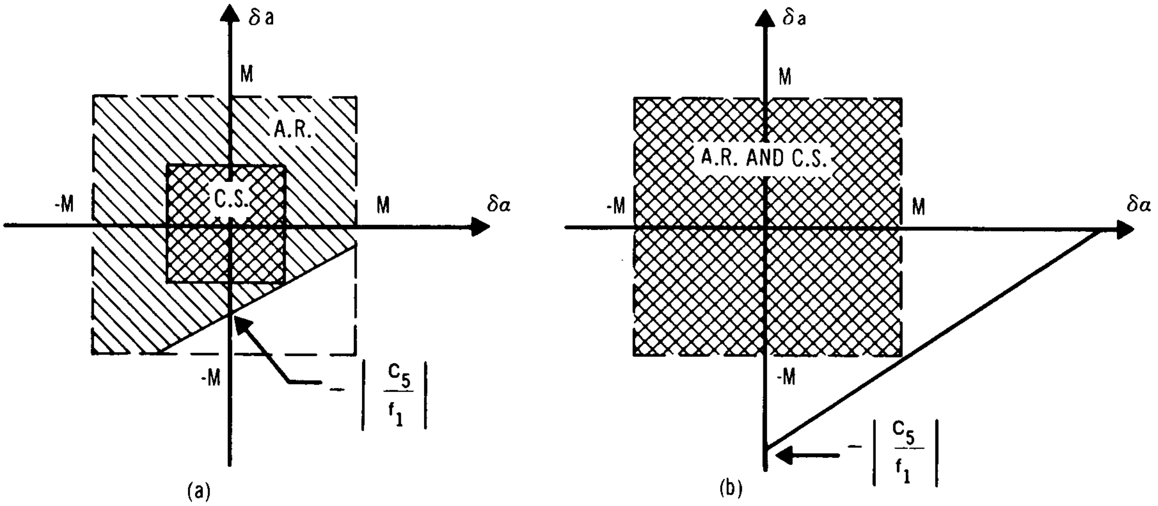

δa
M
A.R.
C.S.
-M
M
δa
-M
$-\left|\frac{C_5}{f_1}\right|$
(a)
δa
M
A.R. AND C.S.
-M
M
δa
-M
$-\left|\frac{C_5}{f_1}\right|$
(b)

Figure 8. Construction of Square Control Situations Centered at (a, α), for Cases (A) and (B) in Eqs. (32) and (34), Respectively

As discussed above, once $\underset{\sim}{a}$ and $\underset{\sim}{\alpha}$ are fixed, a range of values for ρ must be determined from eqs. (17) or (6). Eq. (6) was simulated on the IBM 7094 for over 200 combinations of $\underset{\sim}{a}$ and $\underset{\sim}{\alpha}$. It was observed in every case that eq. (6) has two real roots which are very close to one another; that $\underset{\sim}{C}_1$ always lies between these real roots; and, that

$$\text{left-hand side of eq. (17)} \Bigg|_{\underset{\sim}{r} \,=\, \underset{\sim}{C}_1} < 0.$$

In the procedure described below for partitioning the augmented plant parameter space (a - α space), it is initially assumed, therefore, that $\underset{\sim}{\rho} = \underset{\sim}{C}_1$. In addition, it seems highly desirable to partition the a - α space into squares that are all of the same size; hence, eq. (34) must be satisfied for all values of $\underset{\sim}{a}$ and $\underset{\sim}{\alpha}$ (sufficient to partition the a - α space). The following procedure was used for partitioning the a - α space:

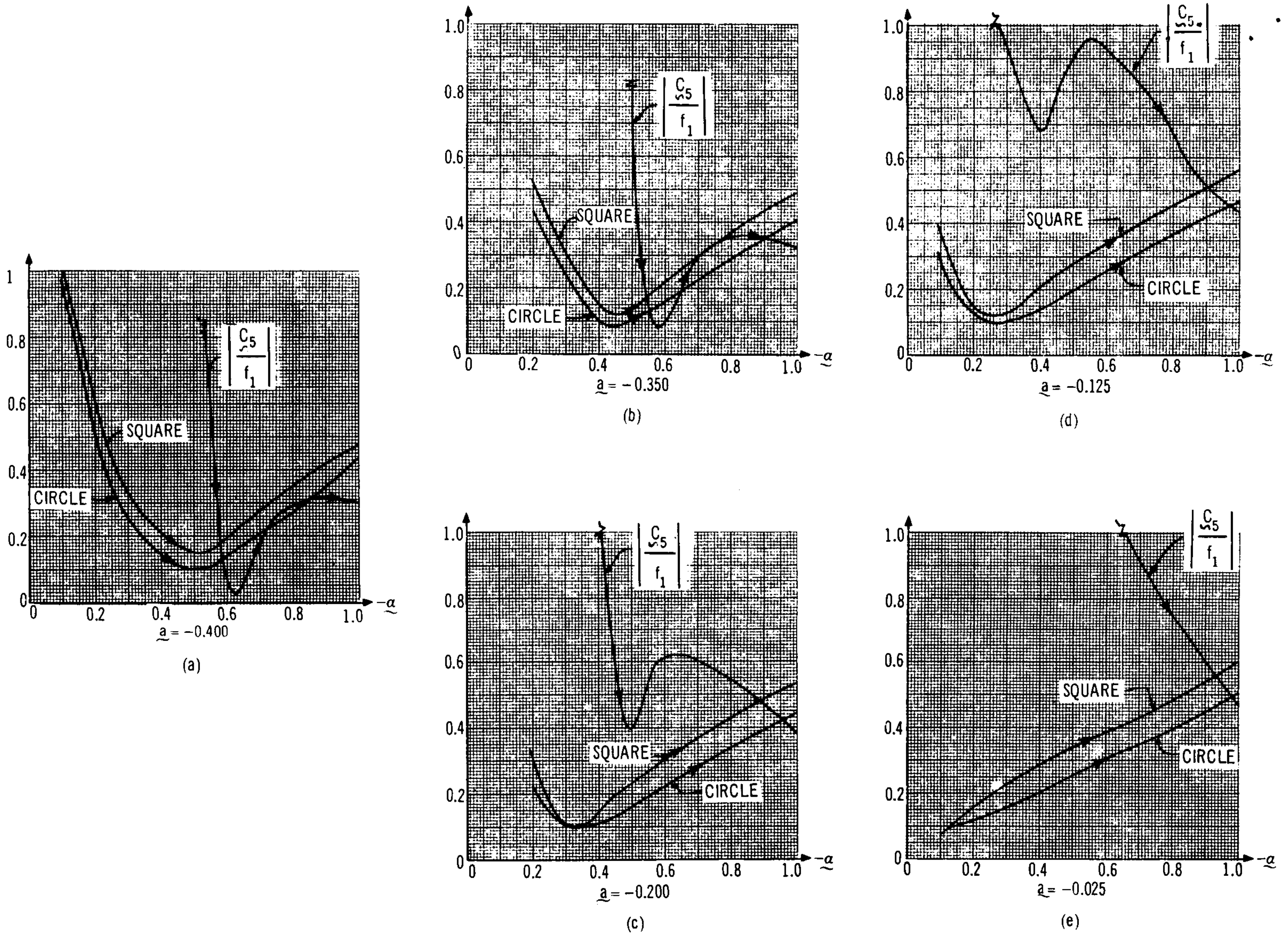
1. Choose $\underset{\sim}{a}$ and $\underset{\sim}{\alpha}$ and set M = 0.10 (for all combinations of $\underset{\sim}{a}$ and $\underset{\sim}{\alpha}$)
2. Evaluate $f_1\ (\underset{\sim}{a}, \underset{\sim}{\alpha}, \underset{\sim}{C}_1)$, $f_2\ (\underset{\sim}{a}, \underset{\sim}{\alpha}, \underset{\sim}{C}_1)$, $\left|\frac{\underset{\sim}{C}_5}{f_1}\right|$ and SQUARE [eq. (32)]
3. Perform the test in eq. (34).

If eq. (34) is true for M = 0.10, $\underset{\sim}{\rho} = \underset{\sim}{C}_1$ and for all values of $\underset{\sim}{a}$ and $\underset{\sim}{\alpha}$ (sufficient to partition the a - α space), the design is complete.* If, on the other hand, eq. (34) does not hold for some combinations of $\underset{\sim}{a}$ and $\underset{\sim}{\alpha}$, any one of the following design refinements may be tried:

1. Set M to a smaller value, keeping $\underset{\sim}{\rho} = \underset{\sim}{C}_1$
2. Choose a different value for $\underset{\sim}{\rho}$, keeping M = 0.10
3. Choose different values for $\underset{\sim}{\rho}$ and M
4. Choose a different set of plant parameters at which to center the control situations.

Design results for the case when M = 0.10 and $\underset{\sim}{\rho} = \underset{\sim}{C}_1$ are summarized in fig. 9. Note that both CIRCLE [in eq. (28)] and SQUARE are plotted on the figures, in order to demonstrate that in all cases circular control situations impose a less severe constraint than square control situations. Observe, also, that for large negative values of $\underset{\sim}{a}$ $\left|\frac{\underset{\sim}{C}_5}{f_1}\right|$ goes below SQUARE, which means that in those cases eq. (34) does not hold; hence, a square control situation of dimension 0.20 x 0.20 cannot be constructed for certain combinations of $\underset{\sim}{a}$ and $\underset{\sim}{\alpha}$. As $\underset{\sim}{a}$ becomes more positive the design becomes more successful. The design was modified by setting M equal to 0.05 and, in

*Larger values of M could be tried.



(a) $\underset{\sim}{a} = -0.400$

(b) $\underset{\sim}{a} = -0.350$

(c) $\underset{\sim}{a} = -0.200$

(d) $\underset{\sim}{a} = -0.125$

(e) $\underset{\sim}{a} = -0.025$

Figure 9. $\left|\frac{\underset{\sim}{C}_5}{f_1}\right|$, Circle, and Square Versus $\underset{\sim}{\alpha}$ for Different Values of $\underset{\sim}{a}$ (M = 0.10).

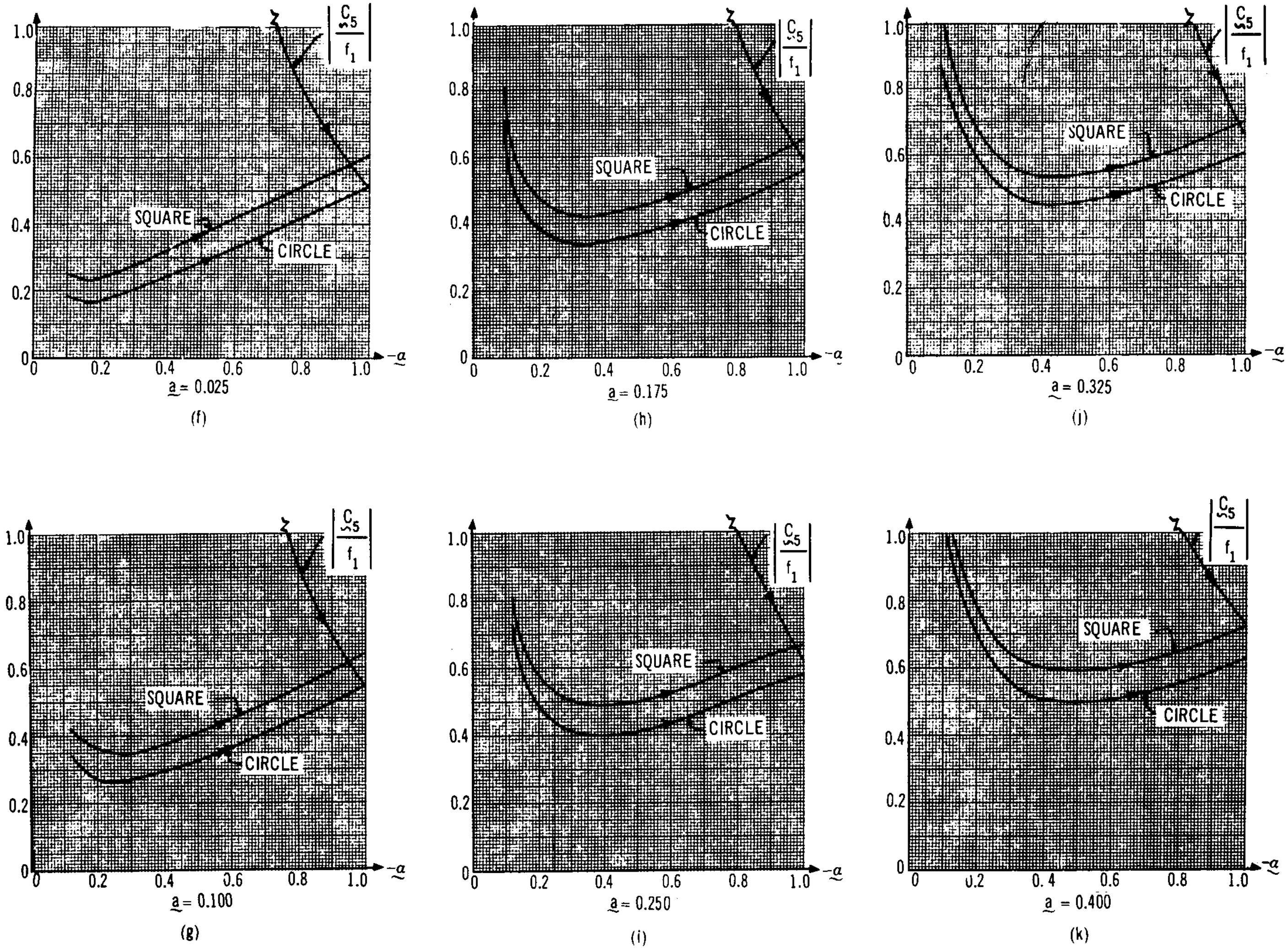


Figure 9. $\left|\frac{\underset{\sim}{C}_5}{f_1}\right|$, Circle, and Square Versus $\underset{\sim}{\alpha}$ for Different Values of $\underset{\sim}{a}$ (M = 0.10).

those few remaining cases where eq. (34) still did not hold [e. g., a = -0.40 and α = -0.60], by rechoosing $\underset{\sim}{\rho}$. The final partition of the a - α space is depicted in fig. 10. There the control situations are labelled 1 through 90.

The weighting factor associated with each nominal control situation is tabulated in Table II. In addition, the components of the nominal feedback parameter vector are tabulated in Table III. These components were obtained from eqs. (3) to (5) and the values for ρ in Table II. Observe, in Table III, that λ_3 is insensitive to a and α; that λ_2 is not very sensitive* to a and α; and, that λ_1 is rather sensitive to a and α. These observations are utilized in the on-line learning experiments which are described in Section 4 of this part.

Note, finally, that the nominal controls in Table III represent the initial controls (feedback parameter vectors) for the on-line-learning control system in fig. 3 of this part.

Simulation Studies

In order to check the performance of the system under the action of the nominal controller, the system in fig. 2 was simulated (on the IBM 7094). The experiments that are described below were performed in order to observe the effects of the following on the attitude errors:

1. A time-varying parameter $[\alpha(t)]$

2. An incorrect knowledge of the disturbance torques.

The simulated system is depicted in fig. 11. The assumptions made during the simulations are listed below.

1. All states are available, which means that in eq. (1) the following is true:

$$\hat{\underline{x}}_a (t|t) = \underline{x}_a (t) \tag{37}$$

2. The plant parameters are generated, which means that**

$$\hat{\underline{P}}_{A_a} (t|t) = \underline{P}_{A_a} (t) \tag{38}$$

*λ_2 in eq. (4) appears, at first, to be only a function of a; however, as demonstrated in fig. C-4 to C-6, ρ is a function of a and α. Thus λ_2 is a function of both a and α.

**$\underline{P}_{A_a}(t)$, the augmented plant parameter vector is given as $[a(t), \alpha(t)]^T$. It contains the time-varying elements in $A_a(t)$. The hat notation in eq. (38) denotes the estimated value of $\underline{P}_{A_a}$ (t) - (see Section 3 of this part, and Appendix B).

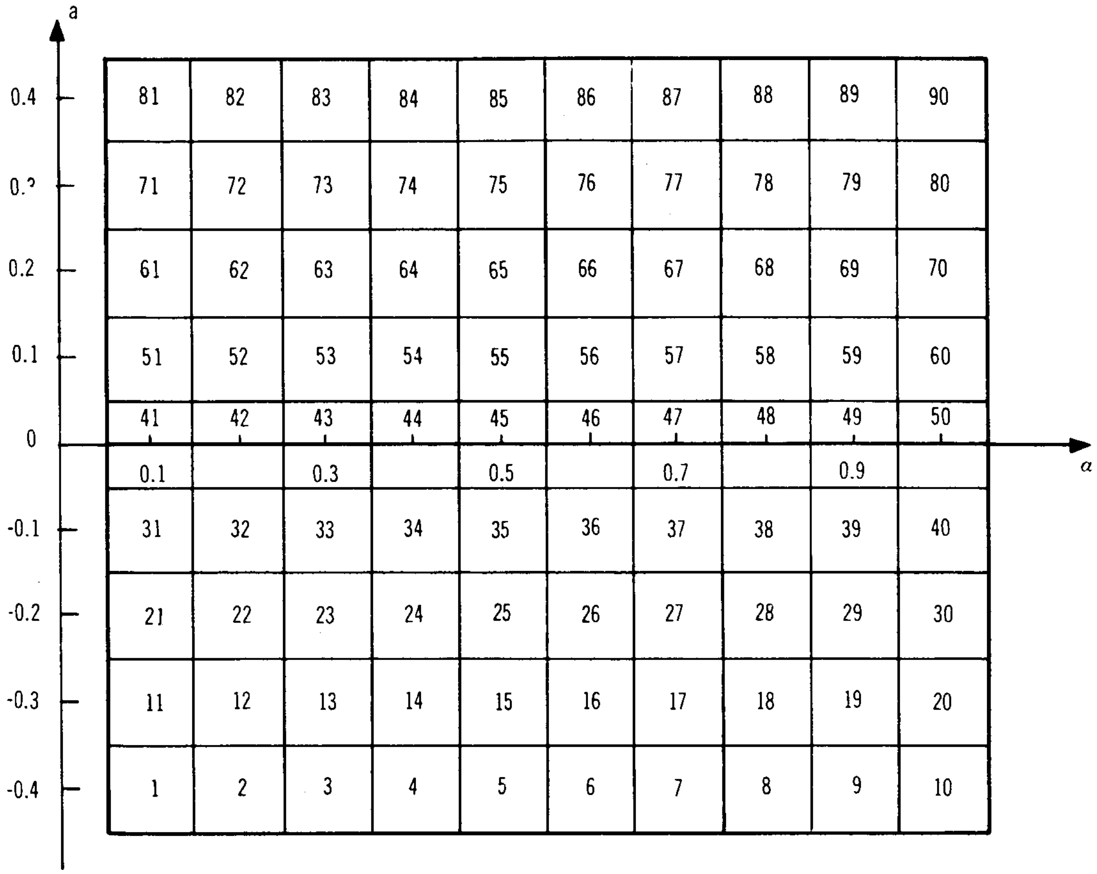


Figure 10. Nominal Control Situations

TABLE II

NOMINAL WEIGHTING FACTORS FOR PARTITIONED a - α SPACE*

Control Situation	$\underset{\sim}{\rho} \times 10^{-5}$	Control Situation	$\underset{\sim}{\rho} \times 10^{-5}$	Control Situation	$\underset{\sim}{\rho} \times 10^{-5}$
1	2.6216	31	0.6050	61	1.4116
2	1.7112	32	0.2852	62	1.1408
3	1.2031	33	0.0388	63	1.1255
4	0.8066	34	0.2017	64	1.2099
5	0.4509	35	0.4509	65	1.3528
6	0.1040	36	0.7135	66	1.5368
7	0.2280	37	0.9909	67	1.7531
8	0.5400	38	1.2834	68	1.9963
9	0.8800	39	1.5909	69	2.2631
10	1.2200	40	1.9131	70	2.5508
11	1.9494	41	0.0672	71	2.0838
12	1.2358	42	0.1901	72	1.6161
13	0.8150	43	0.3493	73	1.5136
14	0.4705	44	0.5378	74	1.5461
15	0.1503	45	0.7515	75	1.6534
16	0.1647	46	0.9879	76	1.8112
17	0.4827	47	1.2449	77	2.0071
18	0.8080	48	1.5210	78	2.2340
19	1.1420	49	1.8149	79	2.4872
20	1.4300	50	2.1257	80	2.7634
21	1.2772	51	0.7394	81	2.7560
22	0.7605	52	0.6654	82	2.0914
23	0.4269	53	0.7374	83	1.9017
24	0.1344	54	0.8739	84	1.8822
25	0.1503	55	1.0522	85	1.9540
26	0.4391	56	1.2624	86	2.0856
27	0.7368	57	1.4990	87	2.2612
28	1.0457	58	1.7587	88	2.4717
29	1.3668	59	2.0390	89	2.7112
30	1.7006	60	2.3383	90	2.9759

*It was assumed that $P_r \left\{ |\ell| \le 10^{-4} \text{ lb-ft} \right\} = 0.96$ in the design of $\underset{\sim}{\rho}$ (see Appendix C).

TABLE III

COMPONENTS OF NOMINAL FEEDBACK PARAMETER VECTOR, FOR PARTITIONED a - α SPACE

Control Situation	Lambda (1)	Lambda (2)	Lambda (3)
1	0.51241577E 03	0.32012990E 02	0.10007569E 04
2	0.41406672E 03	0.28777308E 02	0.10009669E 04
3	0.34725755E 03	0.26353654E 02	0.10011329E 04
4	0.28440732E 03	0.23849835E 02	0.10012890E 04
5	0.21274444E 03	0.20627382E 02	0.10015031E 04
6	0.10238117E 03	0.14309520E 02	0.10024892E 04
7	0.15139722E 03	0.17400990E 02	0.10010523E 04
8	0.23277934E 03	0.21576809E 02	0.10001820E 04
9	0.29704821E 03	0.24374093E 02	0.99967864E 03
10	0.34968521E 03	0.26445612E 02	0.99926999E 03
11	0.44182020E 03	0.29726090E 02	0.10006524E 04
12	0.35183960E 03	0.26526952E 02	0.10008534E 04
13	0.28578220E 03	0.23907413E 02	0.10010244E 04
14	0.21721033E 03	0.20842760E 02	0.10012219E 04
15	0.12289727E 03	0.15677836E 02	0.10017799E 04
16	0.12863585E 03	0.16039691E 02	0.10011468E 04
17	0.22000455E 03	0.20976394E 02	0.10002387E 04
18	0.28455357E 03	0.23855966E 02	0.99977802E 03
19	0.33823504E 03	0.26009038E 02	0.99940665E 03
20	0.37845353E 03	0.27511944E 02	0.99904704E 03
21	0.35757940E 03	0.26742453E 02	0.10005277E 04
22	0.27597171E 03	0.23493476E 02	0.10007252E 04
23	0.20681568E 03	0.20337929E 02	0.10009287E 04
24	0.11613119E 03	0.15240157E 02	0.10014169E 04
25	0.12279706E 03	0.15671443E 02	0.10009323E 04
26	0.20974722E 03	0.20481564E 02	0.10001985E 04
27	0.27164067E 03	0.23308396E 02	0.99980772E 03
28	0.32357291E 03	0.25439061E 02	0.99948130E 03
29	0.36990263E 03	0.27199362E 02	0.99917414E 03
30	0.41258336E 03	0.28725715E 02	0.99887190E 03
31	0.24606750E 03	0.22184116E 02	0.10003626E 04
32	0.16897868E 03	0.18383617E 02	0.10005921E 04
33	0.62389726E 02	0.11170472E 02	0.10014710E 04
34	0.14212116E 03	0.16859488E 02	0.10004326E 04
35	0.21244408E 03	0.20612816E 02	0.10000485E 04
36	0.26721423E 03	0.23117709E 02	0.99976745E 03
37	0.31488565E 03	0.25095245E 02	0.99950411E 03
38	0.35834574E 03	0.26771094E 02	0.99924097E 03
39	0.39896089E 03	0.28247509E 02	0.99897253E 03
40	0.43748999E 03	0.29580061E 02	0.99869679E 03
41	0.81975605E 02	0.12804343E 02	0.99987989E 03
42	0.13787675E 03	0.16605828E 02	0.99999998E 03
43	0.18689569E 03	0.19333685E 02	0.99994592E 03

Table III (Page 2 of 2)

Control Situation	Lambda (1)	Lambda (2)	Lambda (3)
44	0.23190515E 03	0.21536256E 02	0.99982427E 03
45	0.27413500E 03	0.23415166E 02	0.99966316E 03
46	0.31430876E 03	0.25072246E 02	0.99947442E 03
47	0.35283140E 03	0.26564314E 02	0.99926426E 03
48	0.39000000E 03	0.27928480E 02	0.99903640E 03
49	0.42601643E 03	0.29189602E 02	0.99879330E 03
50	0.46105314E 03	0.30366203E 02	0.99853680E 03
51	0.27181912E 03	0.23316051E 02	0.99959890E 03
52	0.25785350E 03	0.22709183E 02	0.99961232E 03
53	0.27145111E 03	0.23300262E 02	0.99959142E 03
54	0.29551801E 03	0.24311232E 02	0.99951856E 03
55	0.32427634E 03	0.25466697E 02	0.99940030E 03
56	0.35520270E 03	0.26653431E 02	0.99924593E 03
57	0.38706923E 03	0.27823344E 02	0.99906293E 03
58	0.41926858E 03	0.28957506E 02	0.99885672E 03
59	0.45145288E 03	0.30048390E 02	0.99863111E 03
60	0.48345972E 03	0.31095328E 02	0.99838892E 03
61	0.37551271E 03	0.27404843E 02	0.99944512E 03
62	0.33755737E 03	0.25982970E 02	0.99940784E 03
63	0.33528478E 03	0.25895358E 02	0.99936919E 03
64	0.34763622E 03	0.26368019E 02	0.99929948E 03
65	0.36760435E 03	0.27114732E 02	0.99919390E 03
66	0.39182046E 03	0.27993587E 02	0.99905507E 03
67	0.41850042E 03	0.28930967E 02	0.99888747E 03
68	0.44659977E 03	0.29886444E 02	0.99869533E 03
69	0.47552055E 03	0.30838954E 02	0.99848225E 03
70	0.50485449E 03	0.31775918E 02	0.99825098E 03
71	0.45618668E 03	0.30205518E 02	0.99932536E 03
72	0.40170757E 03	0.28344579E 02	0.99925373E 03
73	0.38875024E 03	0.27883695E 02	0.99919745E 03
74	0.39290489E 03	0.28032299E 02	0.99912289E 03
75	0.40632032E 03	0.28506853E 02	0.99902048E 03
76	0.42528206E 03	0.29164432E 02	0.99888908E 03
77	0.44770679E 03	0.29923462E 02	0.99873065E 03
78	0.47235218E 03	0.30736043E 02	0.99854807E 03
79	0.49841845E 03	0.31572724E 02	0.99834418E 03
80	0.52538059E 03	0.32415447E 02	0.99812145E 03
81	0.52457634E 03	0.32390626E 02	0.99922382E 03
82	0.45691844E 03	0.30229735E 02	0.99912533E 03
83	0.43568504E 03	0.29518978E 02	0.99905342E 03
84	0.43344347E 03	0.29442944E 02	0.99897194E 03
85	0.44164090E 03	0.29720057E 02	0.99886892E 03
86	0.45628387E 03	0.30208736E 02	0.99874088E 03
87	0.47512093E 03	0.30825993E 02	0.99858805E 03
88	0.49676210E 03	0.31520219E 02	0.99841206E 03
89	0.52029199E 03	0.32258084E 02	0.99821500E 03
90	0.54511824E 03	0.33018729E 02	0.99799901E 03

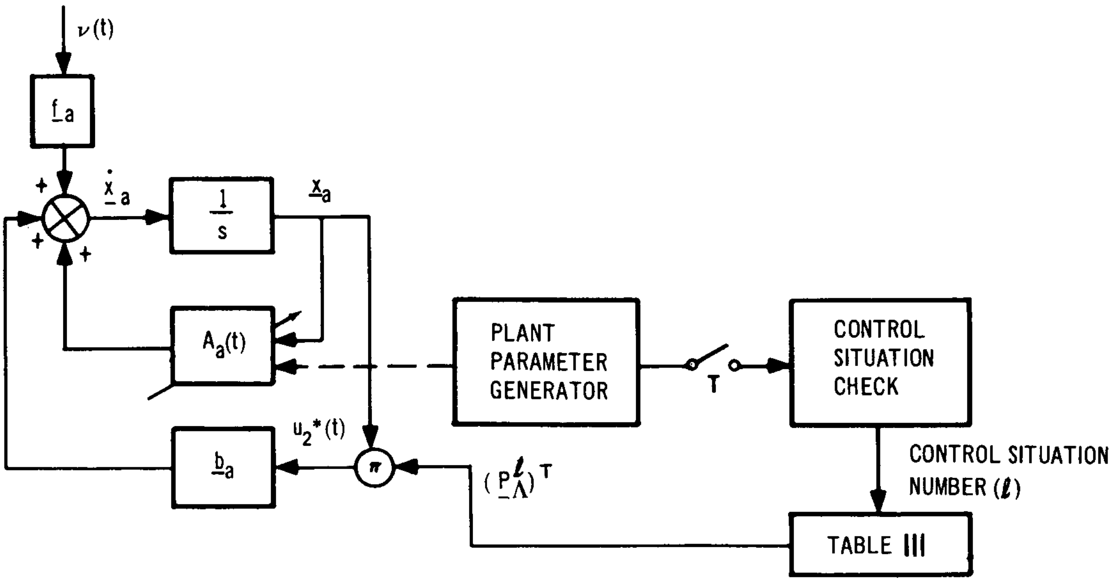


Figure 11. System Simulation for Nominal Control Study

3. Every T units of time, α (t) is checked in order to determine whether the system is in the same control situation. If it is in the same situation, no changes are made in the control gains; on the other hand, if it is in a different control situation the control gains are changed according to Table III.

The first assumption allows concentration on the system without the state estimator. The assumption is made since, in a simulation, all signals are available. It is not necessary to estimate the augmented plant parameters a(t) and α(t), by virtue of the second assumption. a, which varies very very slowly, is fixed for each run, while α(t) is generated by means of a digital function generator (program). The third assumption provides the rule for updating the nominal controls in order to account for the time-varying nature of the system. It provides the nominal controller with the adaptivity referred to previously.

Five experiments were performed; the experiments are summarized in Table IV. The first four were concerned with the effects of different variations and rates of variations in α(t) on the attitude error; the fifth was concerned with the effects of poor a priori knowledge about the disturbance torques on the attitude error. Each experiment was performed for the following five values of the plant parameter a: a = 0.40, 0.20, 0, -0.20 and -0.40. (For a = 0.40 and 0.20 the open-loop system, that is, the system where $\underline{P}_\Lambda^\ell \triangleq \underline{0}$, is stable; for a = 0, -0.20 and -0.40 the open-loop system is unstable.)

TABLE IV

NOMINAL CONTROLLER EXPERIMENTS

Parameter	Experiment Number				
	1	2	3	4	5
$\alpha(t)$[1]	$-0.55 - 0.45 \sin 2\pi t$	$-0.55 - 0.45 \sin 0.2\pi t$	As in fig. 12 (d)	As in fig. 12 (c)	$-0.55 - 0.45 \sin 2\pi t$
$a(t)$	0, 0.2, 0.4, -0.2, -0.4	As in 1	As in 1	As in 1	As in 1
$\Pr\{\lvert\ell\rvert \leq 10^{-4}\}$[3]	0.96	0.96	0.96	0.96	0.10
T[2]	0.05	0.20	0.05	0.05	0.05
Running Time	100T	100T	100T	100T	100T

[1] Time histories of the disturbance state, $\xi(t)$, are depicted in fig. 12(a) to (e) for the different $\alpha(t)$'s used in Experiments 1 to 5, respectively. There, $\xi(t)$ is depicted for one period in the variation of $\alpha(t)$ [$\alpha(t)$ is periodic in all cases]. In order to demonstrate the behavior of $\xi(t)$ for more than one period of $\alpha(t)$, fig. 13 is presented. This figure extends $\xi(t)$ in fig. 12(a) through four periods of variations in $\alpha(t)$ and demonstrates that, due to the time-varying nature of $\alpha(t)$, $\xi(t)$ may be quite different from one period in $\alpha(t)$ to the next.

[2] $\underline{x}_a(0) = (60 \text{ arcsec}, 0, 0)^T$ for all 5 experiments.

[3] It is useful to distinguish between an off-line probability and an on-line probability that $\lvert\ell\rvert \leq 10^{-4}$ lb-ft. The off-line probability is used during the design of the nominal controller ($p_\ell = 0.96$); the on-line probability is associated with the actual disturbances as seen by the spacecraft or during a simulation. In general, it may differ from the off-line probability. For convenience, p_ℓ is used to denote both the off-line or the on-line probability that $\lvert\ell\rvert \leq 10^{-4}$ lb-ft. Values of V equal to 0.48×10^{-8} and 0.1024×10^{-5} were used when $p_\ell = 0.96$ and $p_\ell = 0.10$, respectively.

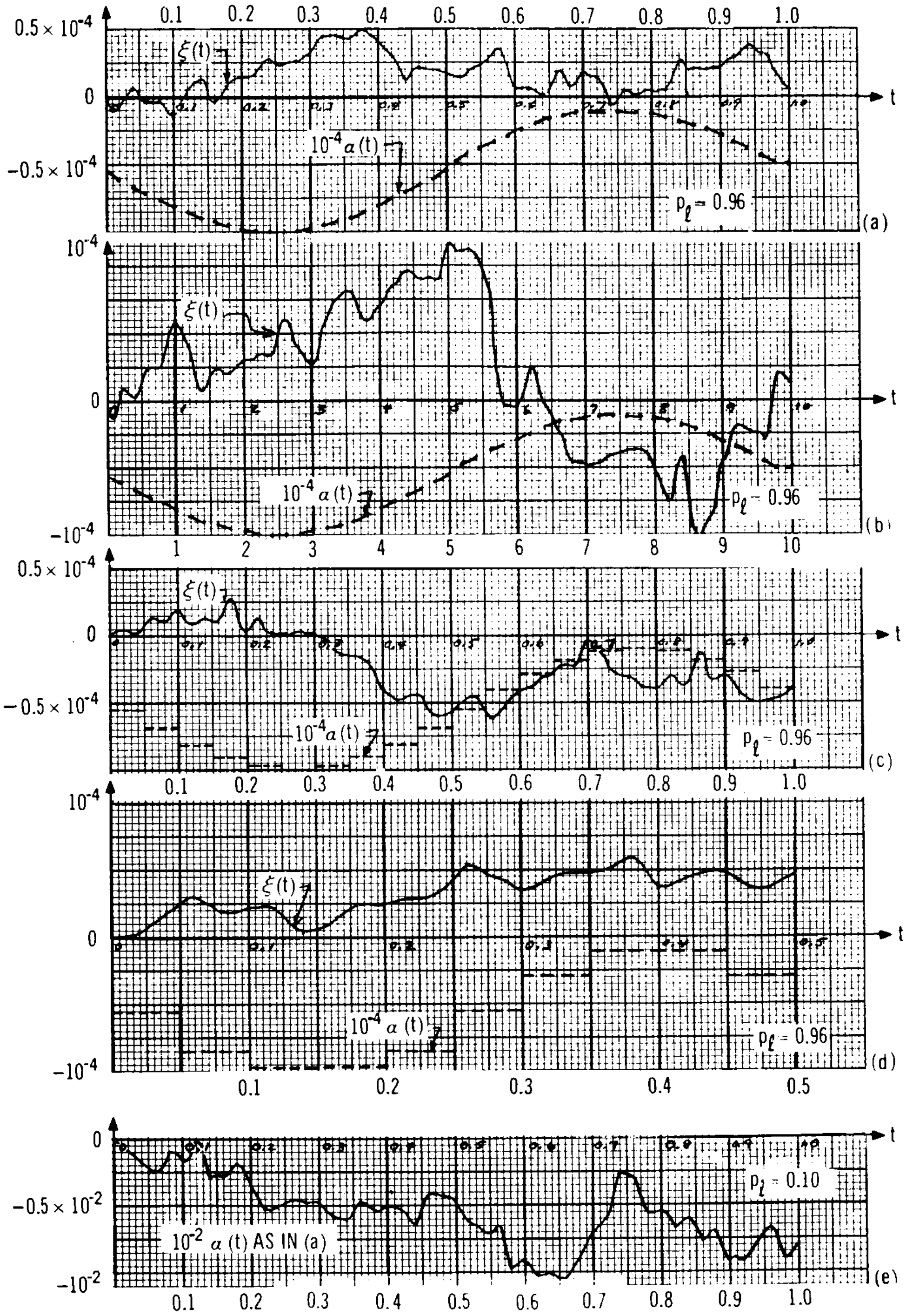


Figure 12. Disturbance State, $\xi(t)$, for Different $\alpha(t)$'s

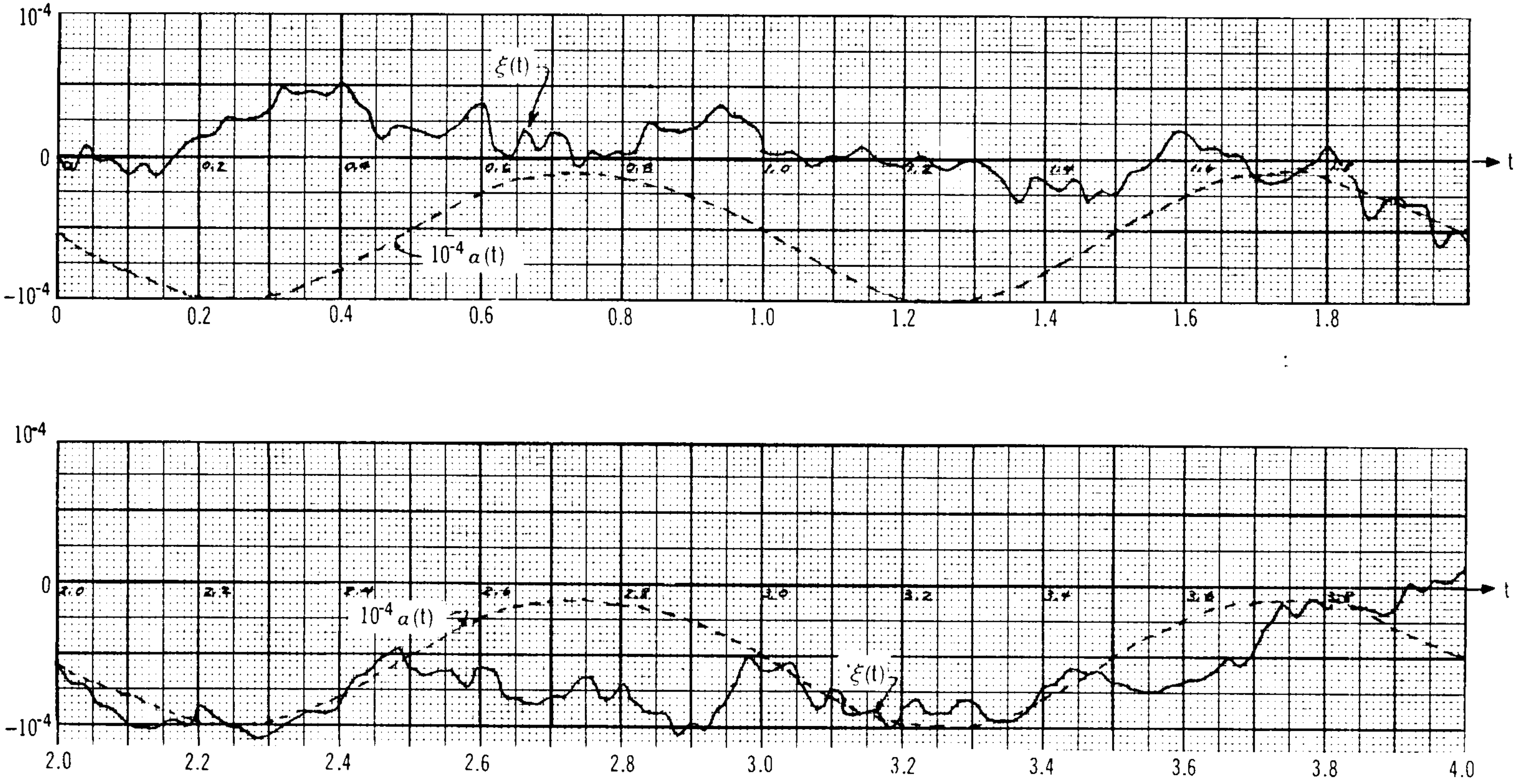


Figure 13. Disturbance State, $\xi(t)$, for $\alpha(t) = -0.55 - 0.45 \sin 2\pi t$

The attitude error results for each experiment are summarized in fig. 14 to 18, respectively. In every experiment the system performs better for the more stable open-loop system. The poorest attitude error time histories occur for a = -0. 40; the best histories occur for a = 0. 40. The results for the first four experiments are very similar. The experiments show that, regardless of how $\alpha(t)$ varies (continuously or discontinuously) or whether $\alpha(t)$ varies slowly or quickly (0. 2π rad/sec to 2π rad/sec), attitude errors are contained for values of a >0. On the other hand, for values of a ≤0 the attitude error may exceed ±0. 20 arcsec over isolated intervals of time; however, the excursions from ±0. 2 arcsec never exceed ±0. 5 arcsec. In many cases the interval of time was very short, the largest being about 0. 7 units of time. The overall conclusion is that the nominal controller performs well under the action of different types of α's provided the on-line probability* agrees with the off-line probability that $|\ell| \leq 10^{-4}$ lb-ft.

If, on the other hand, the on-line probability differs from the off-line probability that $|\ell| \leq 10^{-4}$ lb-ft, attitude errors may exceed ±0. 20 arcsec for long periods of time (if not for all values of time). This is demonstrated in fig. 18, for Experiment 5, where the on-line probability that $|\ell| \leq 10^{-4}$ lb-ft. was assumed to be 0. 10. It is clear from fig. 18 that $x_1(t)$ does indeed exceed $\pm 0.97 \times 10^{-6}$ radians for large intervals of time. It is worthwhile noting, however, that in the worst case (when a = -0. 40) $|x_1|$ never exceeds 4 arcsec, and in the better cases (when a > 0) $|x_1|$ never exceeds 1. 5 arcsec.

It is of interest to observe the time-varying behavior of the components of the feedback gain vector. Observe, from fig. 11, that

$$u_2^*(t) = \underline{P}_\Lambda^T(t)\,\underline{x}_a(t) = \sum_{k=0}^{\infty} \underline{P}_\Lambda^{\ell T}\, u_{-1}(t - kT)\,\underline{x}_a(t) \qquad (39)$$

where $\underline{P}_\Lambda^{\ell}$ is the feedback parameter vector associated with the ℓ^{th} control situation.**

$\underline{P}_\Lambda(t)$ is seen, from eq. (39), to be a piecewise-constant function of time. Typical variations for each of the components of $\underline{P}_\Lambda(t)$ are depicted in figs. 19, 20 and 21. The functions $\lambda_1(t)$, $\lambda_2(t)$ and $\lambda_3(t)$ are shown for only one period in the variation of $\alpha(t)$; however, since $\alpha(t)$ is periodic, $\lambda_1(t)$, $\lambda_2(t)$ and $\lambda_3(t)$ are also periodic, each with the same period as $\alpha(t)$.

In general, $\underline{P}_\Lambda(t)$ is stepped to values which depend on the magnitudes of a and $\alpha(t)$, the rate at which $\alpha(t)$ varies and the decision time T (see fig. 11). $\underline{P}_\Lambda(t)$, quite obviously, is different for different choices of the decision time. From figs 19, 20, and 21 it must be noted that the components of $\underline{P}_\Lambda(t)$, in most cases, look like zero-order approximations to sinusoidal- or rectified-sinusoidal-type functions (unequal lobes).

*See third footnote under Table IV.

**If the feedback gains were fixed (perhaps at some average values) then, according to the preceding sensitivity analysis, attitude errors would exceed ±0. 20 arcsec.

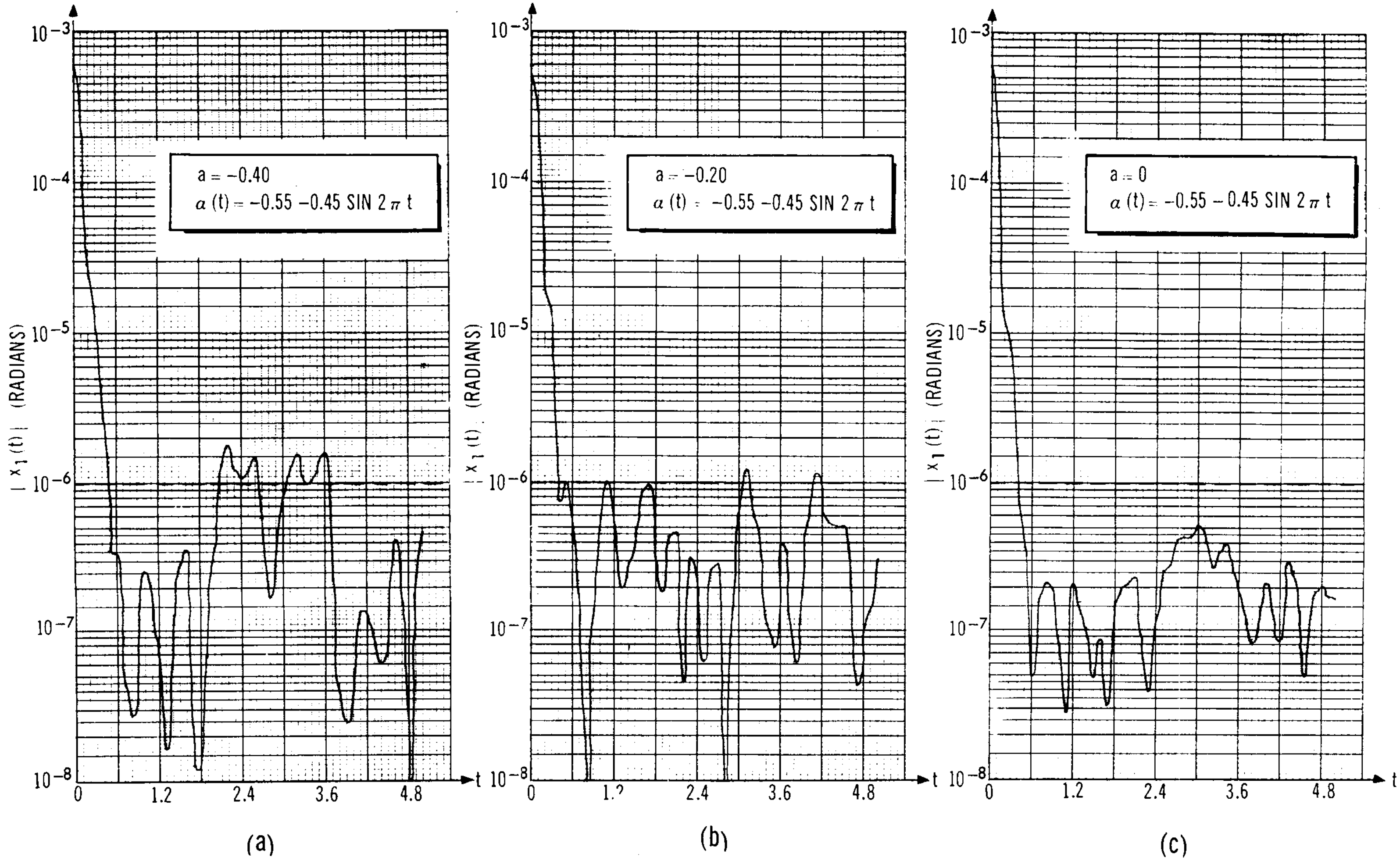
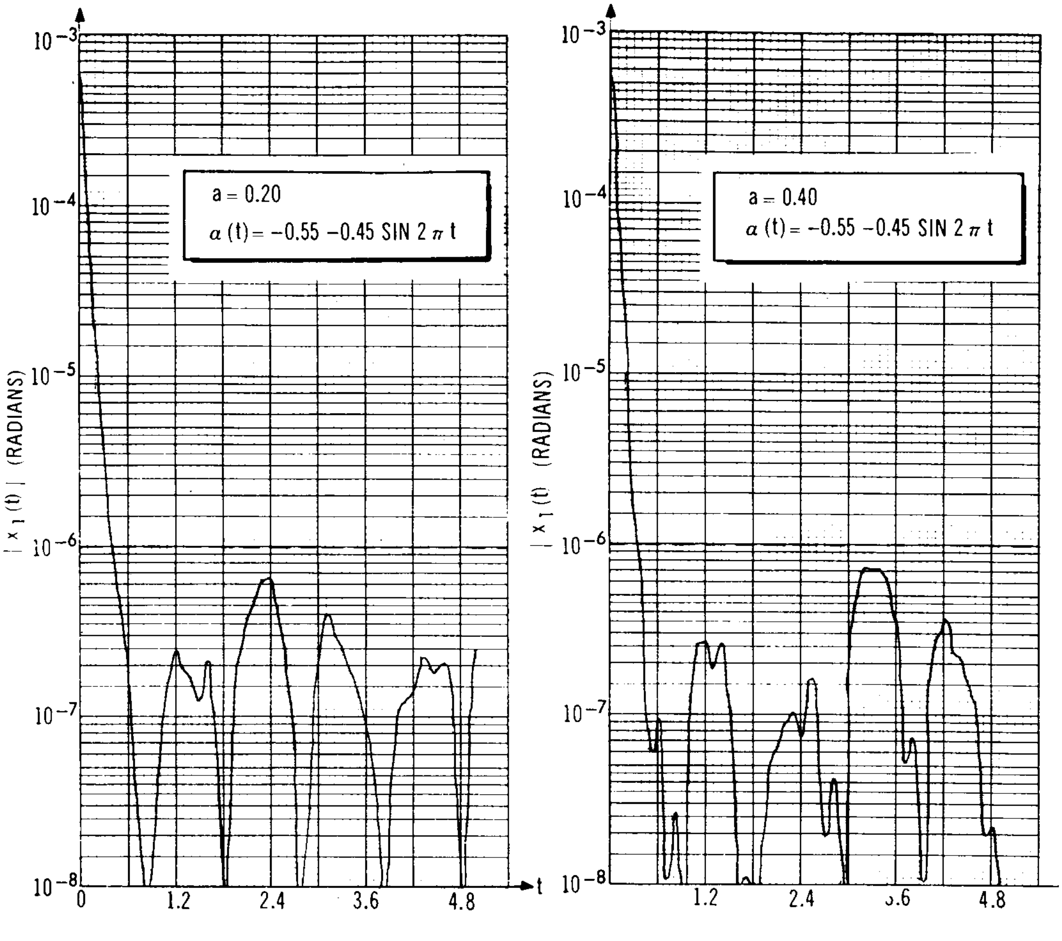


Figure 14. | x_1 (t) | for 5 Values of a (Experiment 1)



(d)

(e)

Figure 14. $|x_1(t)|$ for 5 Values of a (Experiment1)

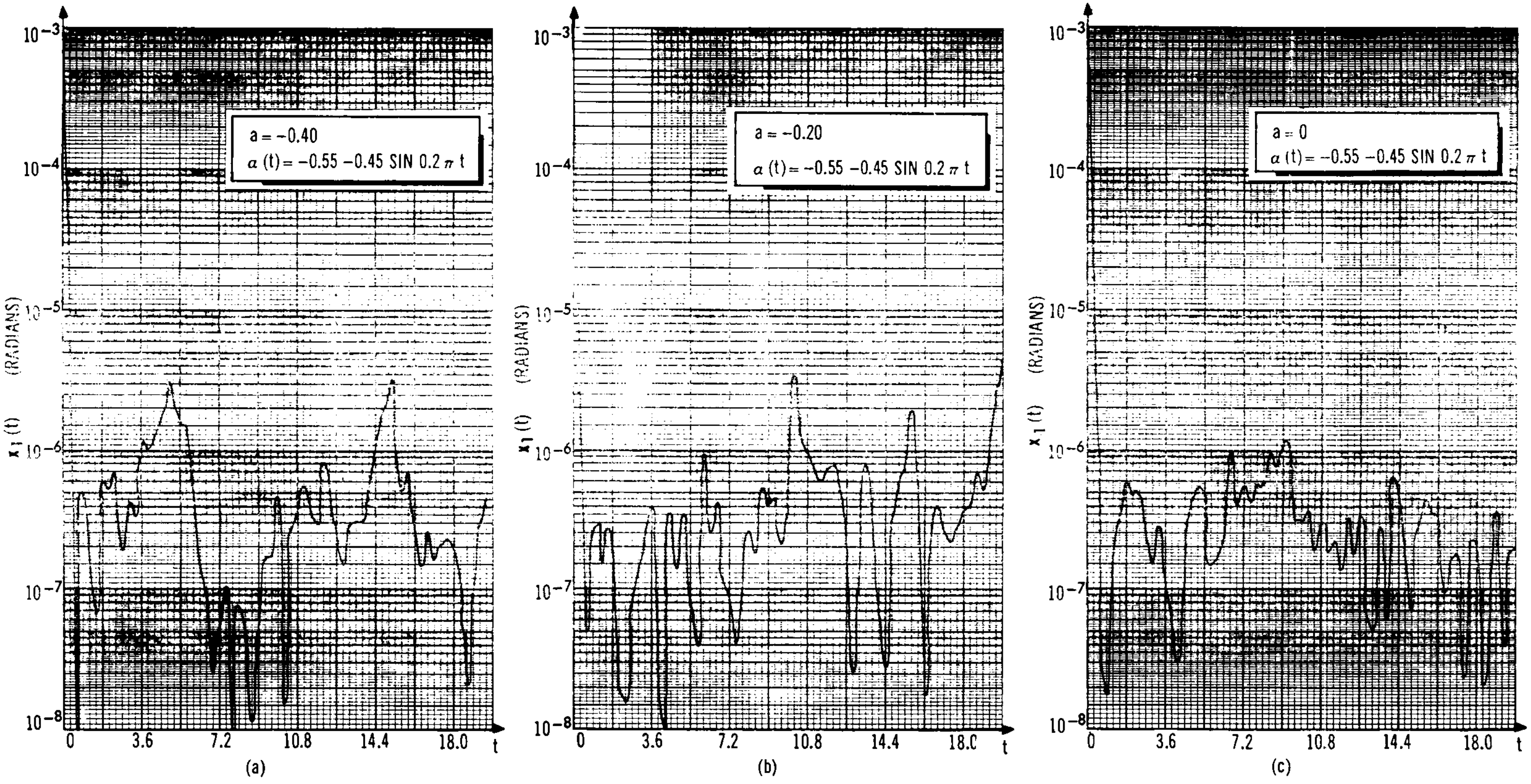
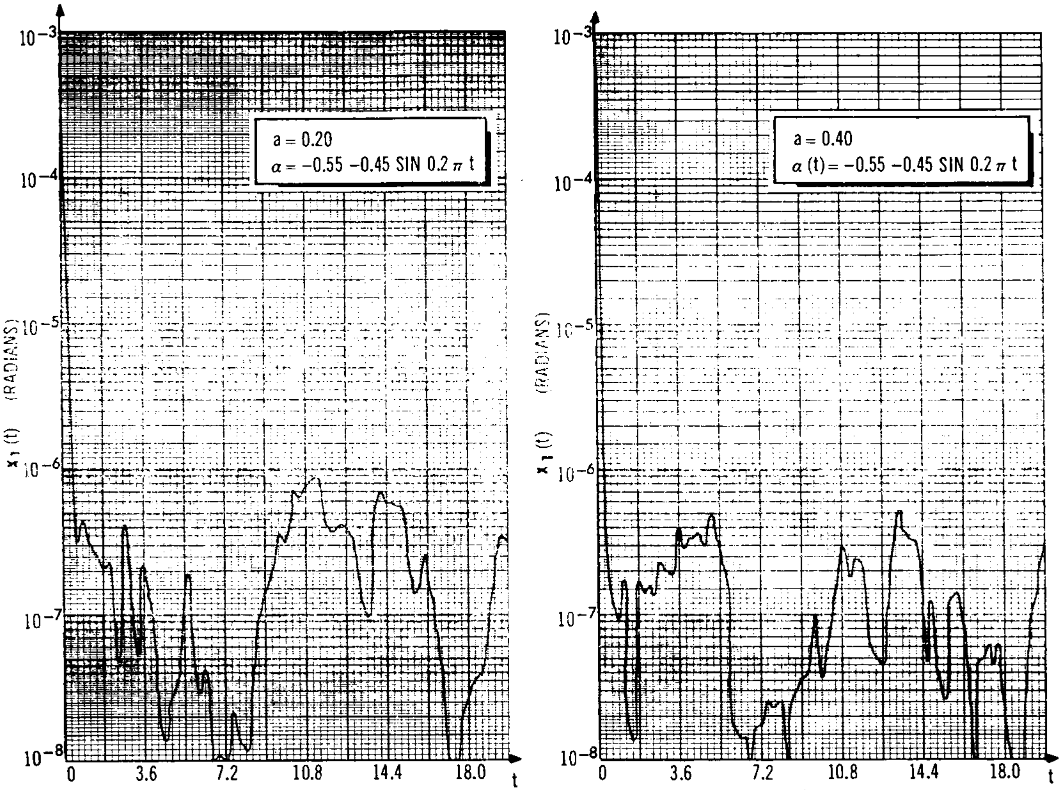


Figure 15. $|x_1(t)|$ for 5 Values of a (Experiment 2)



(d)

(e)

Figure 15. $|x_1(t)|$ for 5 Values of a (Experiment 2)

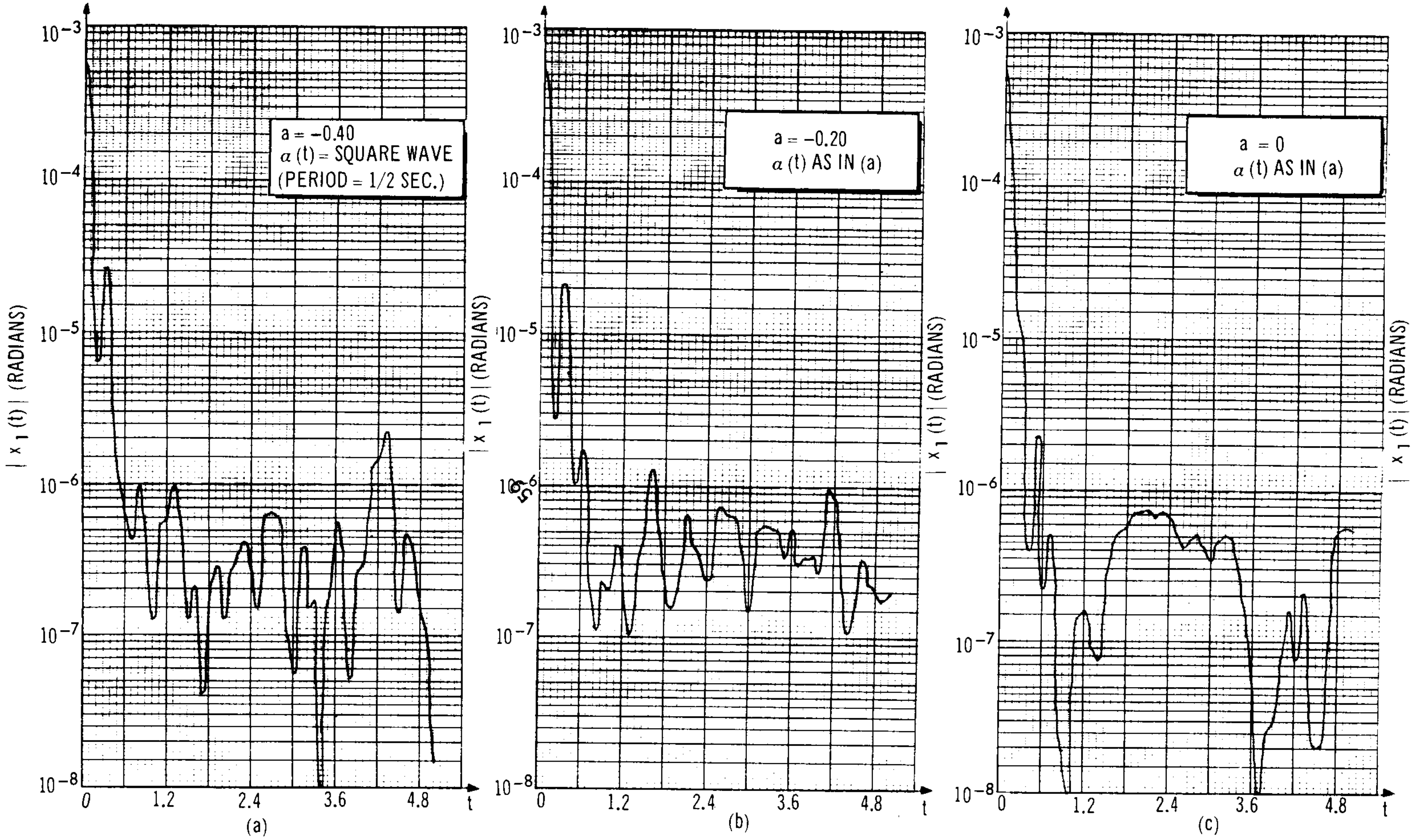


Figure 16. $|x_1(t)|$ for 5 Values of a (Experiment 3)

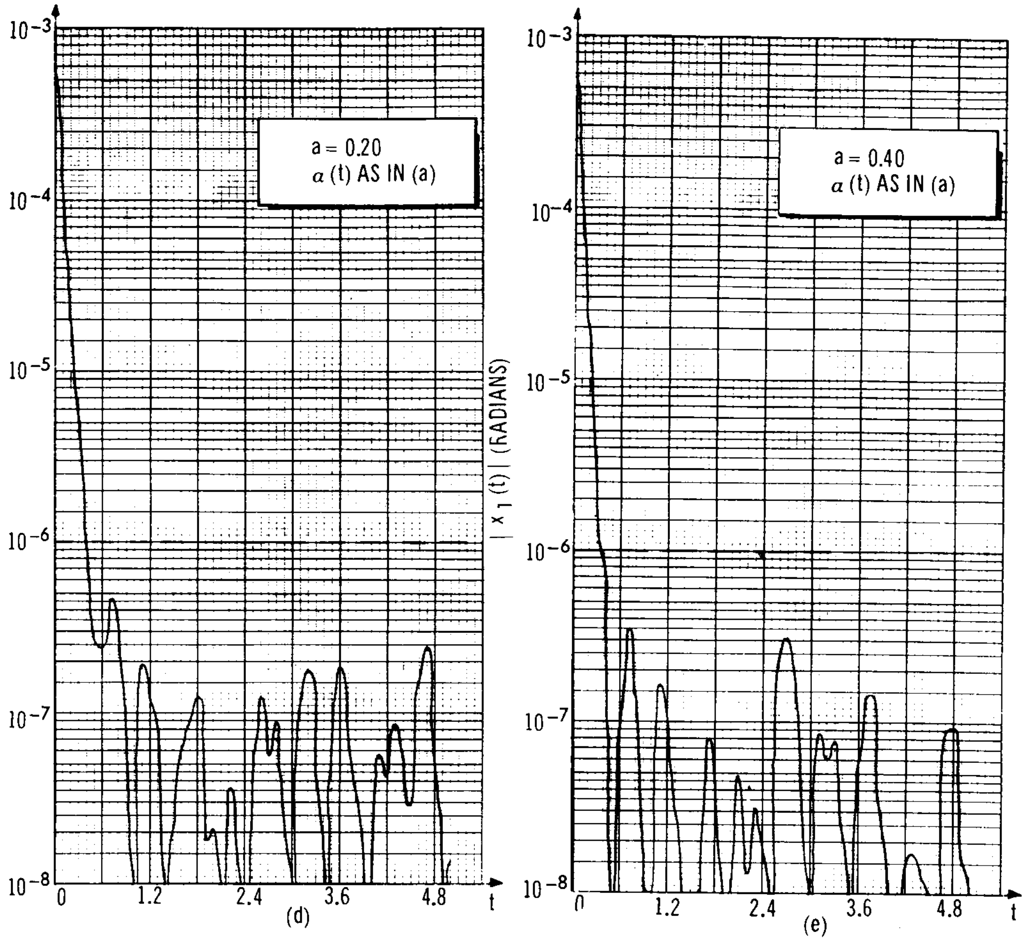

10^-3
10^-4
10^-5
10^-6
10^-7
10^-8
a = 0.20
α (t) AS IN (a)
0
1.2
2.4
3.6
4.8
t
(d)
| x_1 (t) | (RADIANS)
10^-3
10^-4
10^-5
10^-6
10^-7
10^-8
a = 0.40
α (t) AS IN (a)
0
1.2
2.4
3.6
4.8
t
(e)

Figure 16

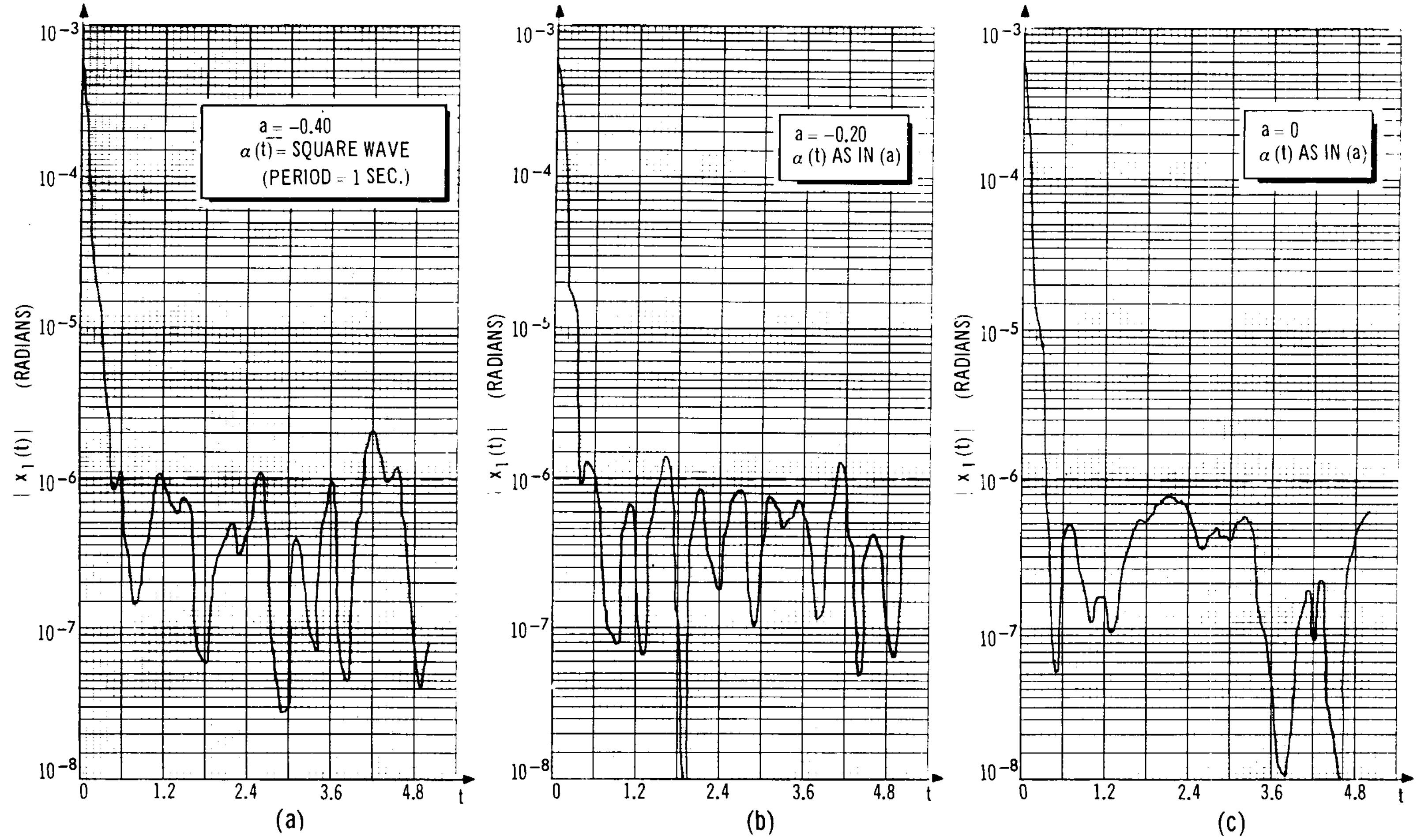


Figure 17. | $x_1(t)$ | for 5 Values of a (Experiment 4)

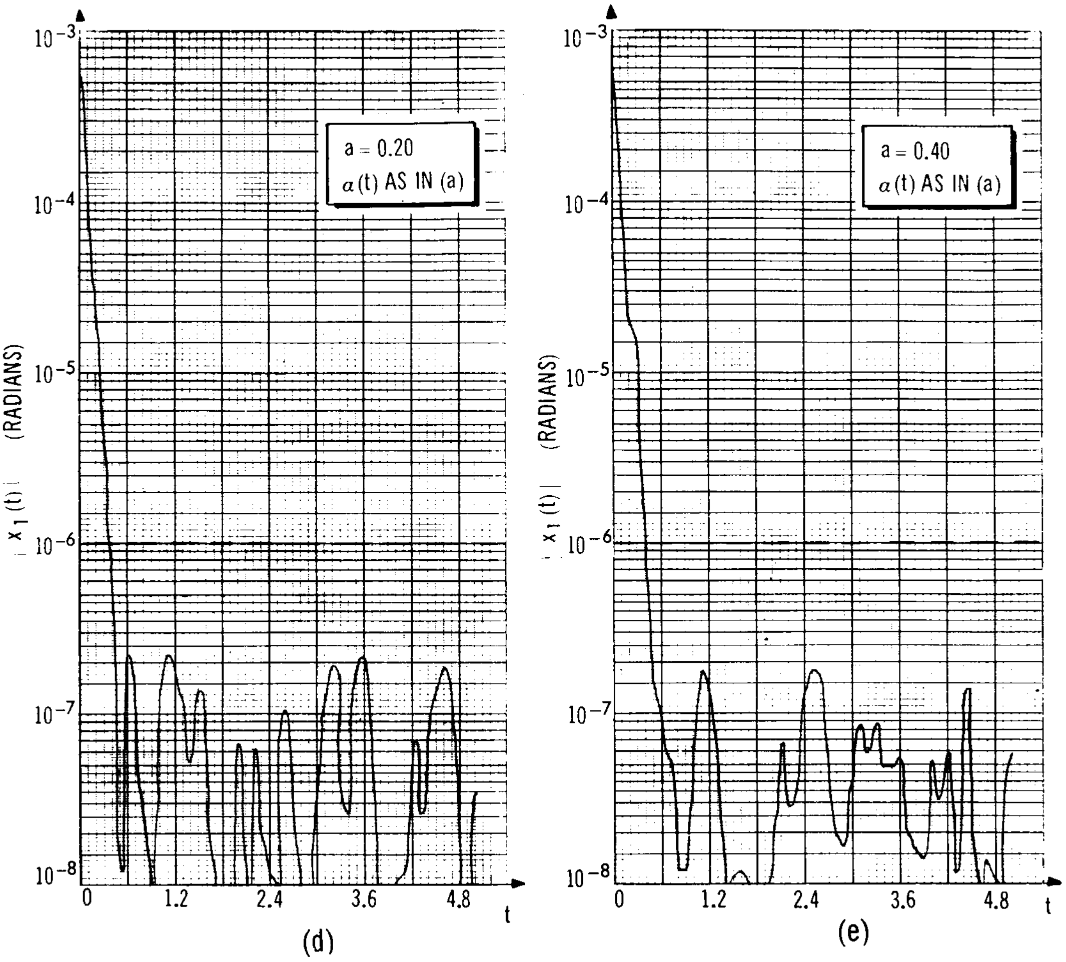


Figure 17. $|x_1(t)|$ for 5 Values of a (Experiment 4)

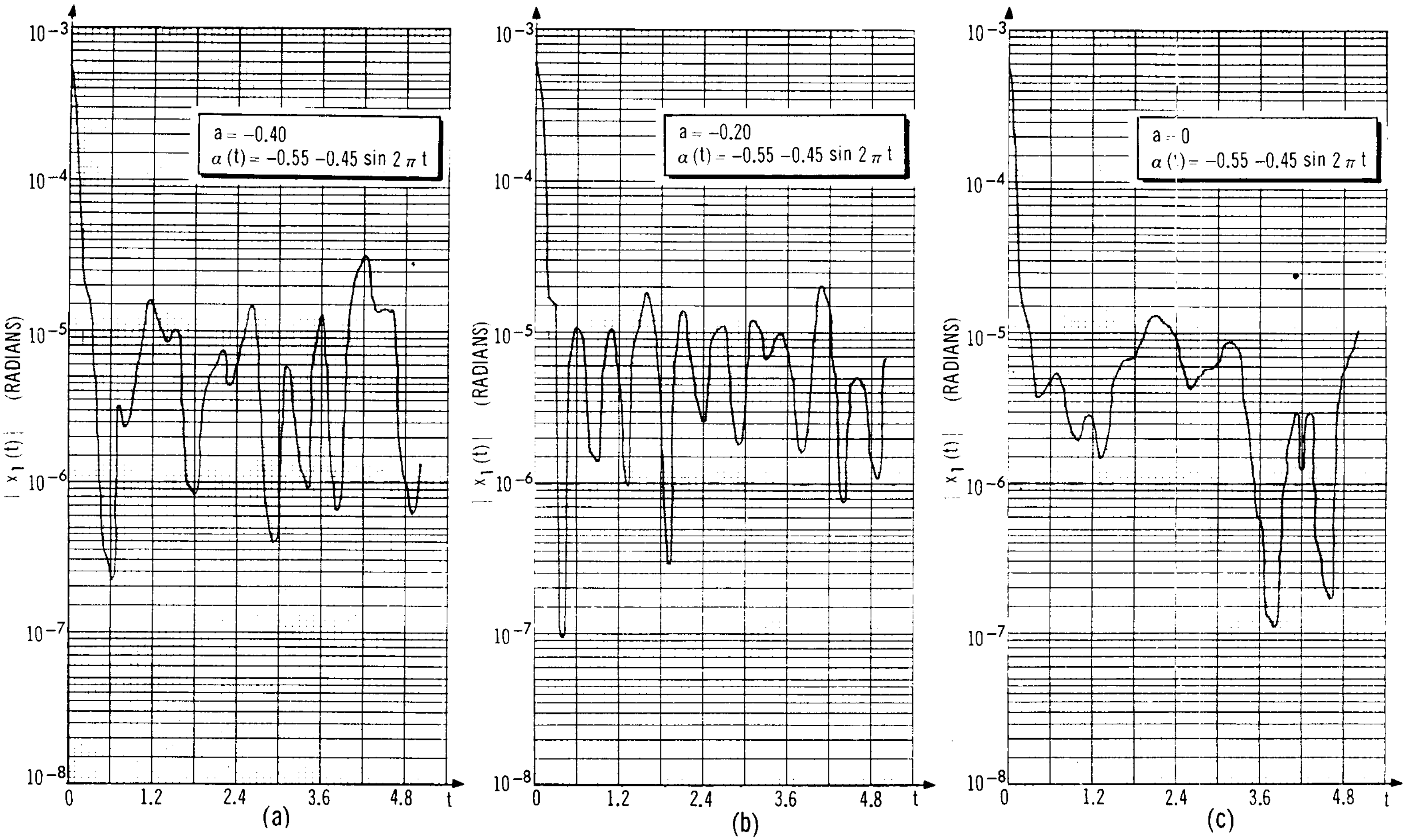
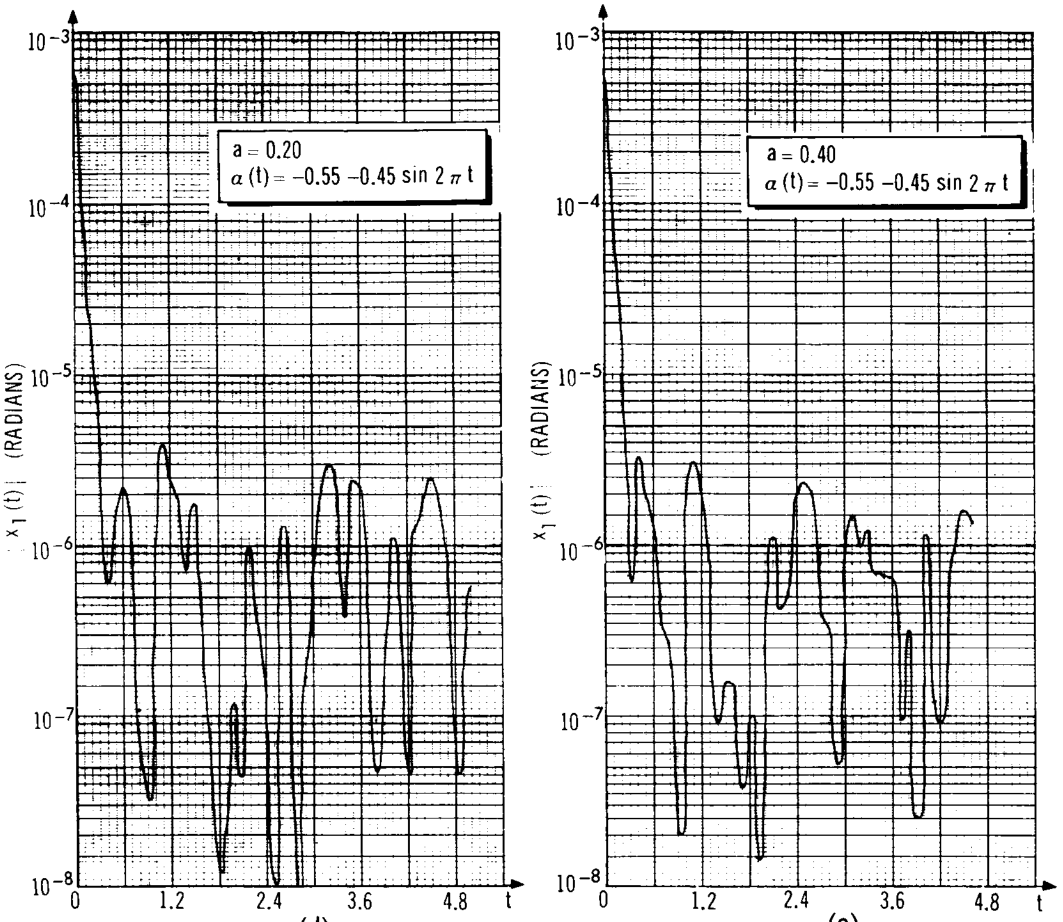


Figure 18. | $x_1(t)$ | for 5 Values of a (Experiment 5)


a = 0.20
α (t) = −0.55 −0.45 sin 2 π t
a = 0.40
α (t) = −0.55 −0.45 sin 2 π t
$|x_1(t)|$ (RADIANS)
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{-8}
0
1.2
2.4
3.6
4.8
t

(d) (e)

Figure 18.

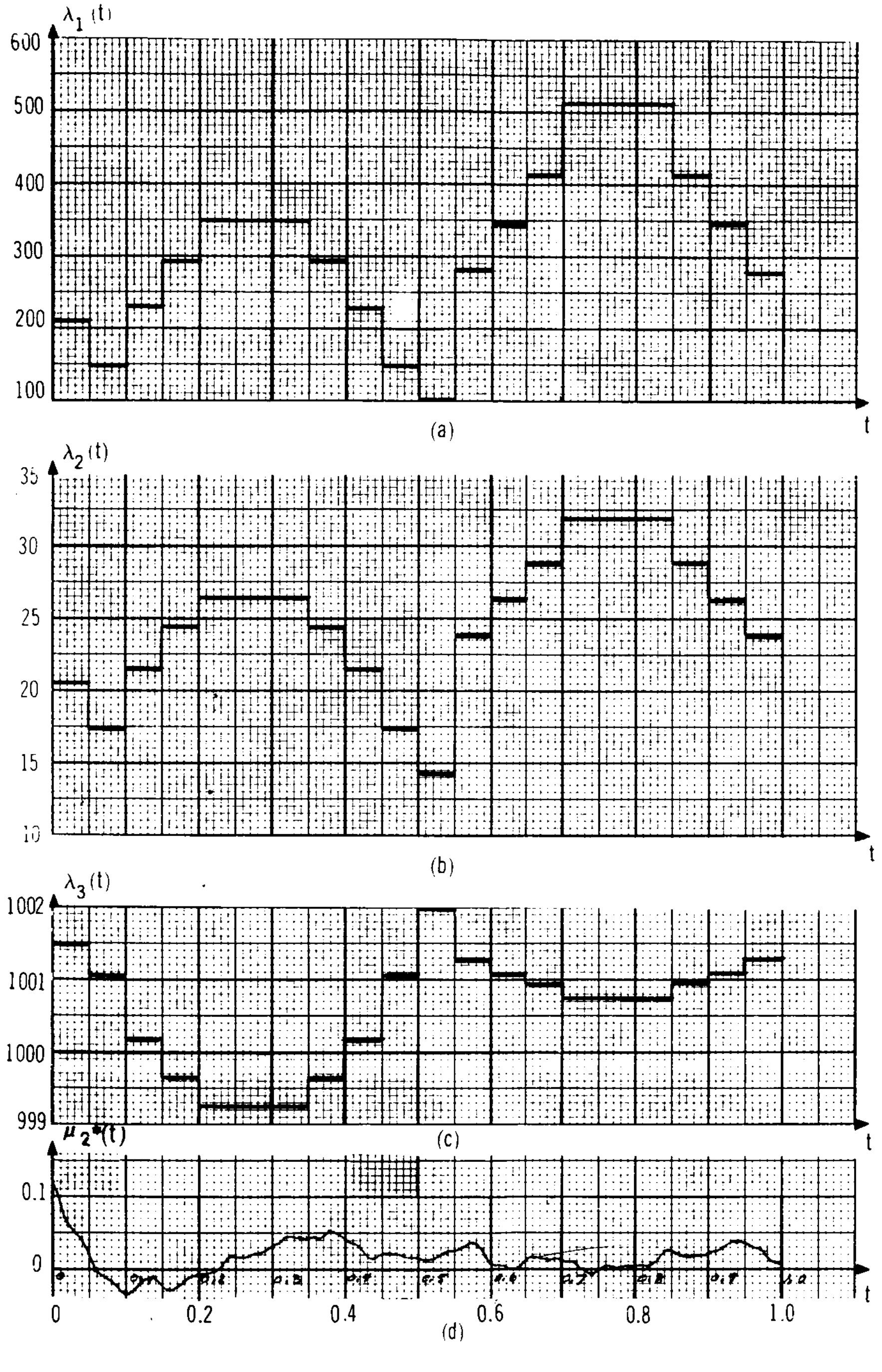


Figure 19. Nominal Feedback Gains and Control as a Function of Time – a = –0.40, $\alpha(t) = -0.55 - 0.45 \sin 2\pi t$ and $p_\ell = 0.96$

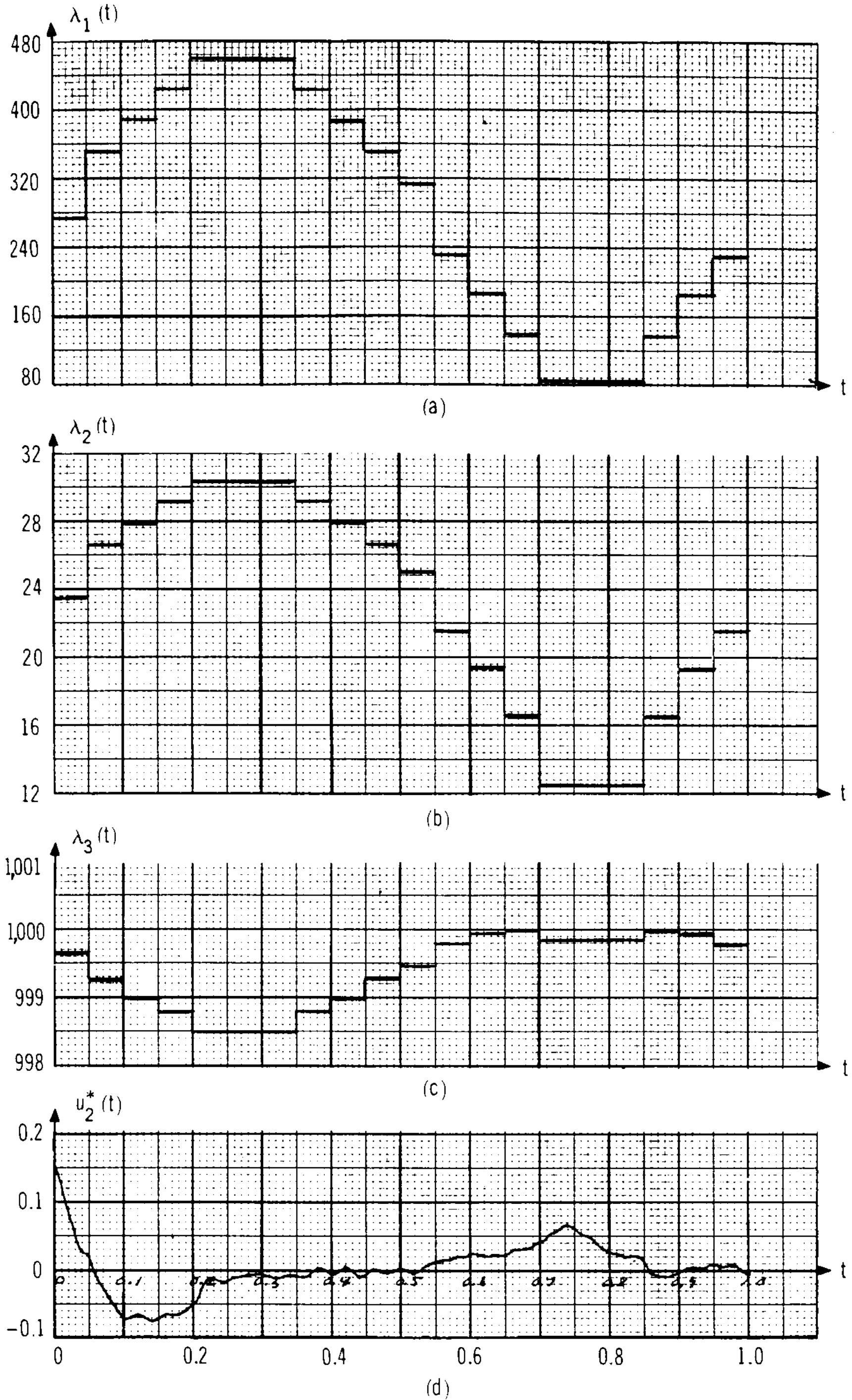
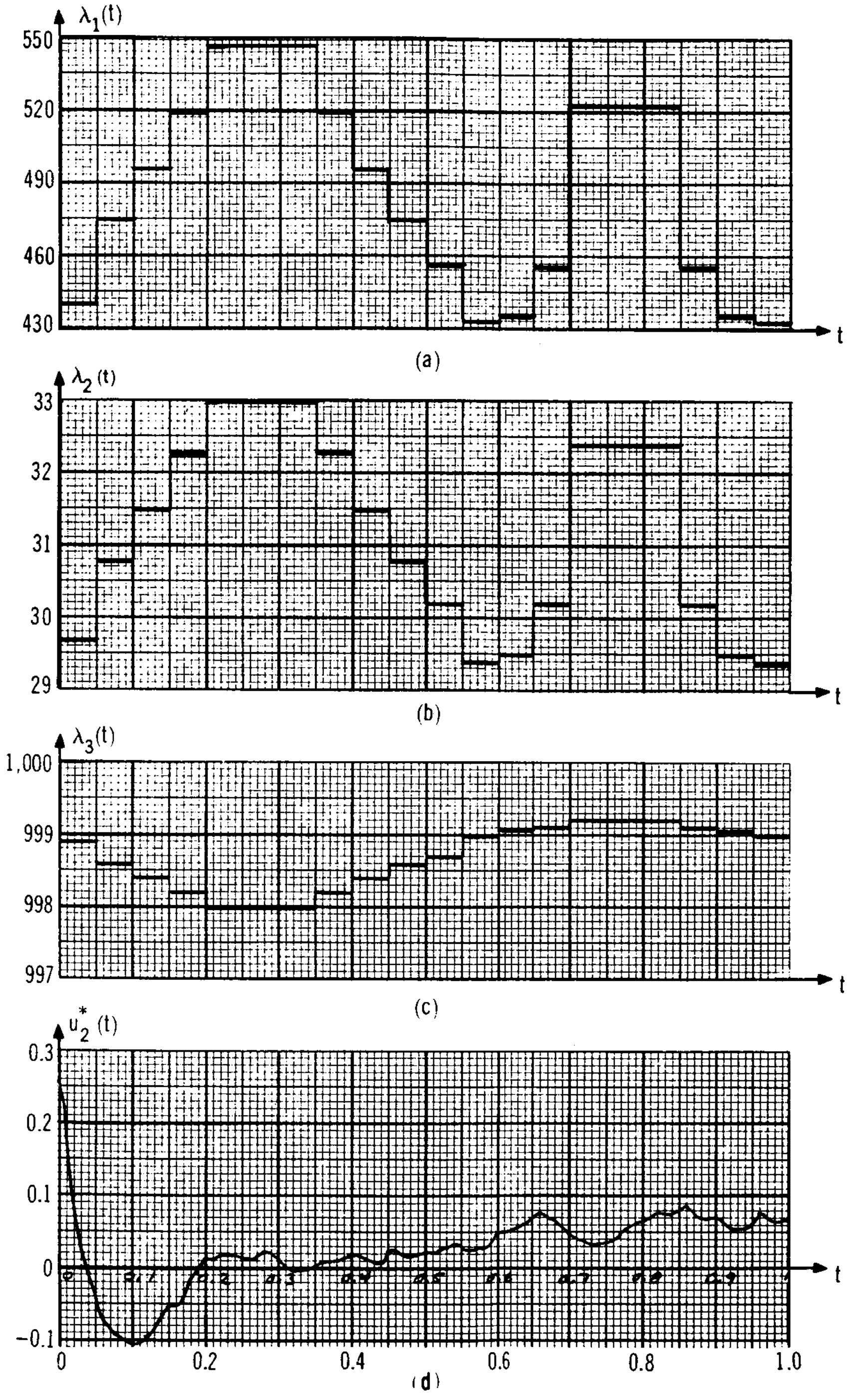


Figure 20. Nominal Feedback Gains and Control as a Function of Time – $a = 0$, $\alpha(t) = -0.55 - 0.45 \sin 2\pi t$ and $p_\ell = 0.96$



(a)

(b)

(c)

(d)

Figure 21. Nominal Feedback Gain and Control as a Function of Time – a = 0.40, $\alpha(t) = -0.55 - 0.45 \sin 2\pi t$ and $p_\ell = 0.96$

The optimal control $u_2^*(t)$ is also depicted on figs. 19, 20, and 21. Observe that after a short initial time-interval $u_2^*(t)$ becomes quite random. This is expected, since

$$u_2^*(t) \simeq \lambda_3(t)\,\xi(t) \tag{40}$$

if the system is performing satisfactorily, that is, driving attitude errors to within ±0. 20 arcsec. Note also that if $\xi(t)$ is bounded u_2^* will also be bounded.

A typical phase-plane plot is depicted in fig. 22. The four plots depict the trajectory of x_2 as a function of x_1 for the following: (a) $0 \le t \le 0.20$, (b) $0.22 \le t \le 0.44$, (c) $0.46 \le t \le 0.80$, and (d) $0.82 \le t \le 1.72$. Different scales are used for $x_1(t)$ and $x_2(t)$ on each plot. The predicted limit cycle behavior is evident from (d) in fig. 22. It must be observed that for $0.82 \le t \le 1.72$ the limit cycle is contained to within $\pm 0.97 \times 10^{-6}$ radians for $x_1(t)$, as desired. From fig. 14(a), which presents $|x_1(t)|$ for the case depicted in fig. 22, one observes that for $2<t<2.7$ and $3<t<3.7$ $|x_1(t)|>0.97 \times 10^{-6}$; hence, for these intervals of time the limit cycle in (d) would exceed ±0. 20 arcsec.

Summary and Conclusions

An adaptive controller has been designed. It is designated nominal controller, since all of the available information about the plant and its environment has been incorporated into the design. The design proceeded in three steps: (1) for fixed values of plant parameters, nominal gains were obtained using stochastic optimal-control theory; (2) for specific ranges of the plant parameters, which were denoted control situations, nominal gains were obtained using the results from Step (1) and sensitivity analyses; and (3) for the controller designed in Step (2), simulations were performed in order to observe the effects of some of the assumptions made during the design on the attitude error.

Although parameters and states were not estimated during these simulations, the following were observed: (1) a time-varying disturbance parameter $[\alpha = \alpha(t)]$ degrades attitude error more when the open-loop system is unstable than when it is stable and, in both cases, the degradation is not very appreciable; and (2) disturbance levels greater than those anticipated during the design of the nominal controller may cause attitude errors to exceed ±0. 20 arcsec, depending upon whether or not the open-loop system is stable or unstable.

The effects of different types of estimation errors on the attitude error, as controlled by the nominal controller designed in this section, are discussed in Section 3.

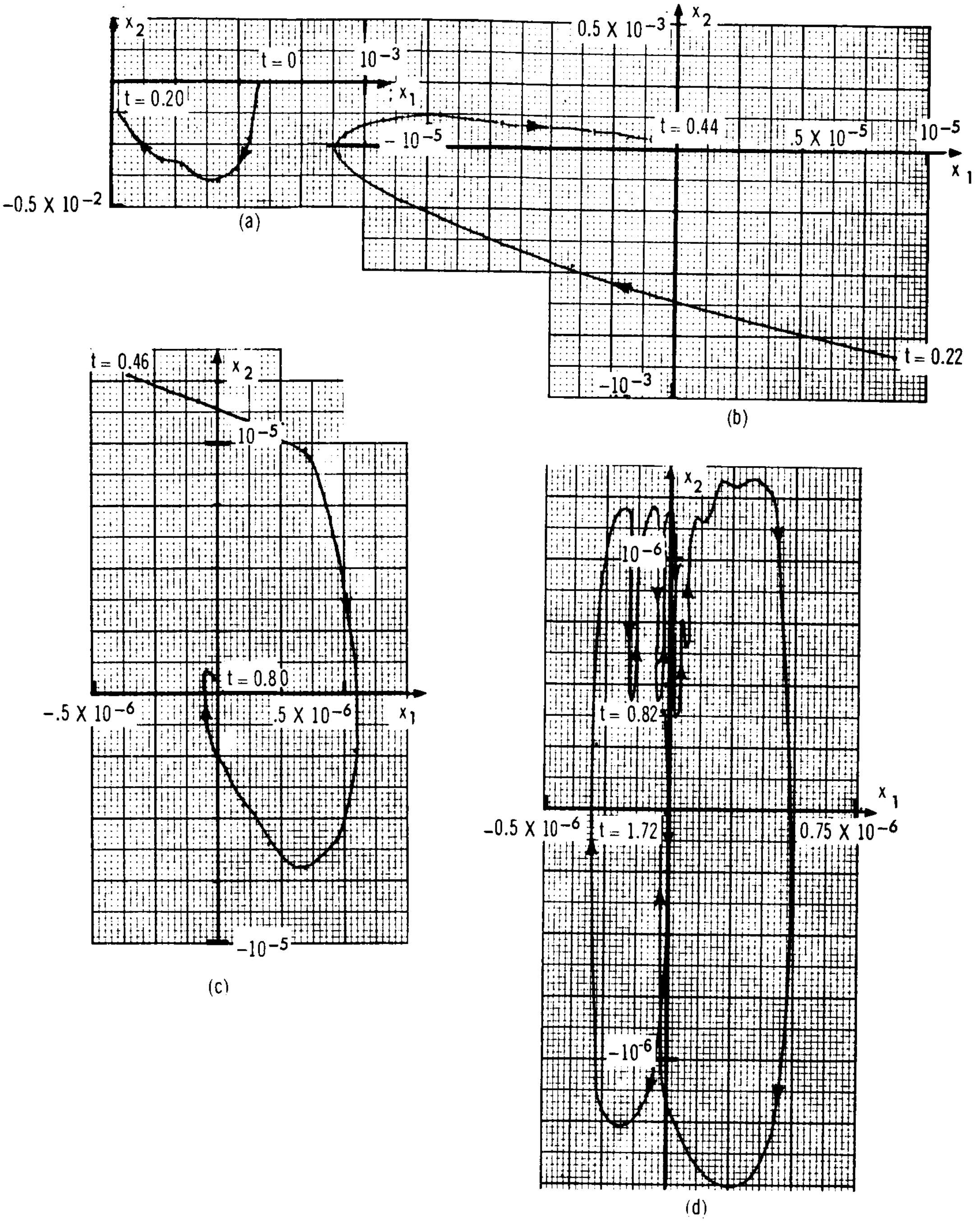


Figure 22. Typical Phase-Plane Portraits--a = -0.40, $\alpha(t)$ = -0.55-0.45 sin $2\pi t$ And p_ℓ = 0.96

The nominal control system serves two purposes in relation to the on-line-learning control system shown in fig. 3. First, it is used as a system with which the performance of the on-line-learning control system may be compared, and a comparison of this type is important if the on-line-learning control system is to be judged fairly. Second, simulations of the system under the control of the nominal controller point up the reasons, if any, for wanting to include an on-line-learning capability. Hence, the nominal controller not only serves to provide the justifications for on-line-learning, but also provides a reference for statements made about the performance of the on-line-learning control system.

Section 3
STATE AND PARAMETER ESTIMATION FOR CONTROL

Introduction

It was mentioned in Section 1 and discussed in Section 2 that estimates of the three states $x_1(t)$, $x_2(t)$, and $\xi(t)$ are needed in order to implement the control $u^*(t)$ (see fig. 3). It was also pointed out in those sections that estimates of the plant parameters, a(t) and $\alpha(t)$ are needed for the following reasons: (1) to provide the initial choice for $u^*(t)$ in fig. 3, since the nominal control is prestored for different regions in the a-α space (nominal control situations), and (2) to implement a control situation concept which is utilized during on-line learning (discussed in section 4).

A technique for estimating the three states and two parameters simultaneously is described in Appendix B. This technique represents an extension of Kalman filtering to the combined estimation problem, that is, the problem of estimating states and parameters simultaneously. It results in a nonlinear estimator.

The purpose of the present section is twofold. First, results are presented in connection with our specific combined estimation problem, followed by a description of the experiments and results for the system under the action of the nominal controller that was designed in the preceding section but with the loop closed around the filter instead of the plant.

Combined Estimation

The structure of the combined state and parameter estimator is depicted in fig. B-2 (at this point it is recommended that Appendix B be read in its entirety). For convenience, the equations which describe the estimator are summarized below, since many of them are referred to throughout this paragraph.

Summary of filter equations is as follows:

$$\hat{\underline{q}}_\epsilon(n|n) = K^*(n)\left[\underline{z}(n) - M^*\hat{\underline{q}}(n|n-1)\right] \qquad (41)$$

$$\hat{\underline{q}}(n|n) = \hat{\underline{q}}(n|n-1) + \hat{\underline{q}}_\epsilon(n|n) \qquad (42)$$

$$\hat{\underline{q}}\,(n+1|n) = \Phi(n+1,n)\,\hat{\underline{q}}\,(n|n) + \underline{\Delta}\,(n+1,n)\,u\,(n) \tag{43}$$

$$K^*(n) = P\,(n|n-1)\,M^{*T}\left[M^*P(n|n-1)\,M^{*T} + R_d\right]^{-1} \tag{44}$$

$$P\,(n|n) = P\,(n|n-1) - K^*(n)\,M^*P\,(n|n-1) \tag{45}$$

$$P\,(n+1|n) = \Phi^*\,(n+1,n)\,P\,(n|n)\,\Phi^{*T}\,(n+1,n) + C_d \tag{46}$$

where M^*, $\Phi\,(n+1,n)$, $\underline{\Delta}\,(n+1,n)$ and $\Phi^*\,(n+1,n)$ are defined in eqs. (B42), (B36), (B37) and (B38), respectively. In addition, the discrete measurement noise covariance, R_d, and the discrete disturbance noise covariance, C_d, are as follows:

$$R_d = RT_s = \begin{pmatrix} r_{11}T_s & 0 \\ 0 & r_{22}T_s \end{pmatrix} \triangleq \begin{pmatrix} r_{d11} & 0 \\ 0 & r_{d22} \end{pmatrix} \tag{47}$$

$$C_d = CT_s \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & VT_s & 0 & 0 \\ 0 & 0 & 0 & \mu_{11}T_s & 0 \\ 0 & 0 & 0 & 0 & \mu_{22}T_s \end{pmatrix} \triangleq \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & V_d & 0 & 0 \\ 0 & 0 & 0 & \mu_{d11} & 0 \\ 0 & 0 & 0 & 0 & \mu_{d22} \end{pmatrix} \tag{48}$$

If the noise and disturbance covariances are known a priori then the filter in fig. B-2 is the optimal filter, since the covariances called for in the derivation of the optimal filter are the true ones. In the present application, however, measurement noise-levels were not prespecified and the disturbance noise-levels which appear in the equation for $\underline{P}_{A_a}(t)$ [eq. (B17)] were not fixed ahead of time. The only noise level which is fixed, in a certain sense (as discussed in Appendix C) is the disturbance torque noise-level. The approach adopted was to consider the four covariances r_{d11}, r_{d22}, μ_{d11} and μ_{d22} as design parameters for the combined estimator; hence, the filter is not optimal in the derived sense, it is sub-optimal in a design sense.

In order to differentiate the sub-optimal and optimal filters, let $\bar{R}_d$ and $\bar{C}_d$ be the design covariances, whereas R_d and C_d are the true (but unknown) covariances. For the latter covariances, $K^*(n)$ is as defined in eq. (44), whereas for the former $K^*(n)$ becomes $\bar{K}^*(n)$, a sub-optimal gain.

It is also possible to view the disturbance covariance, V_d, as a design parameter, initially, even though it is known. This is a result of the following theorem, the proof of which is found in ref. 1.

Theorem. -- The Kalman filter is not dependent on the absolute values of the disturbance and measurement noise-levels, but only on the ratio of the two. For example,

$$\overline{K}^* (n;\overline{R}_d, \overline{C}_d) = \overline{K}^* (n;c\overline{R}_d, c\overline{C}_d) \tag{49}$$

where c is an arbitrary positive constant.

The overall approach to the design of a working estimator was to first design a sub-optimal state estimator and then to design the combined estimator. The parameters a(t) and α(t) were frozen and were assumed known during the design of the sub-optimal state estimator. The covariances r_{d11} and r_{d22} obtained during this design were then fixed during the design of the combined estimator; $\overline{\mu}_{d11}$ and $\overline{\mu}_{d22}$ were varied during the latter design.

Good results were obtained for the state estimator, even though an uncontrollable state [ξ(t)] was estimated; however, it was not possible to obtain satisfactory estimates of a(t) and α(t) using the combined estimator. The design of and some experimental results for the sub-optimal state estimator are described below. Some analytical as well as experimental reasons why the combined estimator failed to provide satisfactory parameter estimates are also discussed below.

State Estimation.* -- The following steps were performed as part of the design of the sub-optimal state estimator:

(1) $\overline{V}_d$ was set equal to V_d and the elements of $\overline{R}_d$ were varied until good estimates for x_1(t) and x_2(t) were obtained [regardless of the quality of the estimates of ξ (t)].

(2) $\overline{R}_d$ was fixed from the results of Step 1 and $\overline{V}_d$ was varied until good estimates for all three states were obtained.

(3) $\overline{V}_d$ from Step 2 was scaled to the actual V_d and, as a result of the theorem stated in the preceding paragraph $\overline{R}_d$ was similarly scaled.

*The filter in fig. B-2 reduces to a state estimator when μ_{d11} and μ_{d22} are both set equal to zero, and when γ_1 and γ_2 in eqs. (B29) and (B30) are also set equal to zero.

Fifty-eight different filters were designed. NASA requested an informal tabulation of the qualities of each design, those that were failures as well as those that were successes. This information has been transmitted to NASA and is not included in the present report. The values of $\overline{r}_{d11}$ and $\overline{r}_{d22}$ which resulted from this study are*

$$\overline{r}_{d11} = 7.70 \times 10^{-6} \tag{50a}$$

$$\overline{r}_{d22} = 7.70 \times 10^{-12} \tag{50b}$$

In addition,

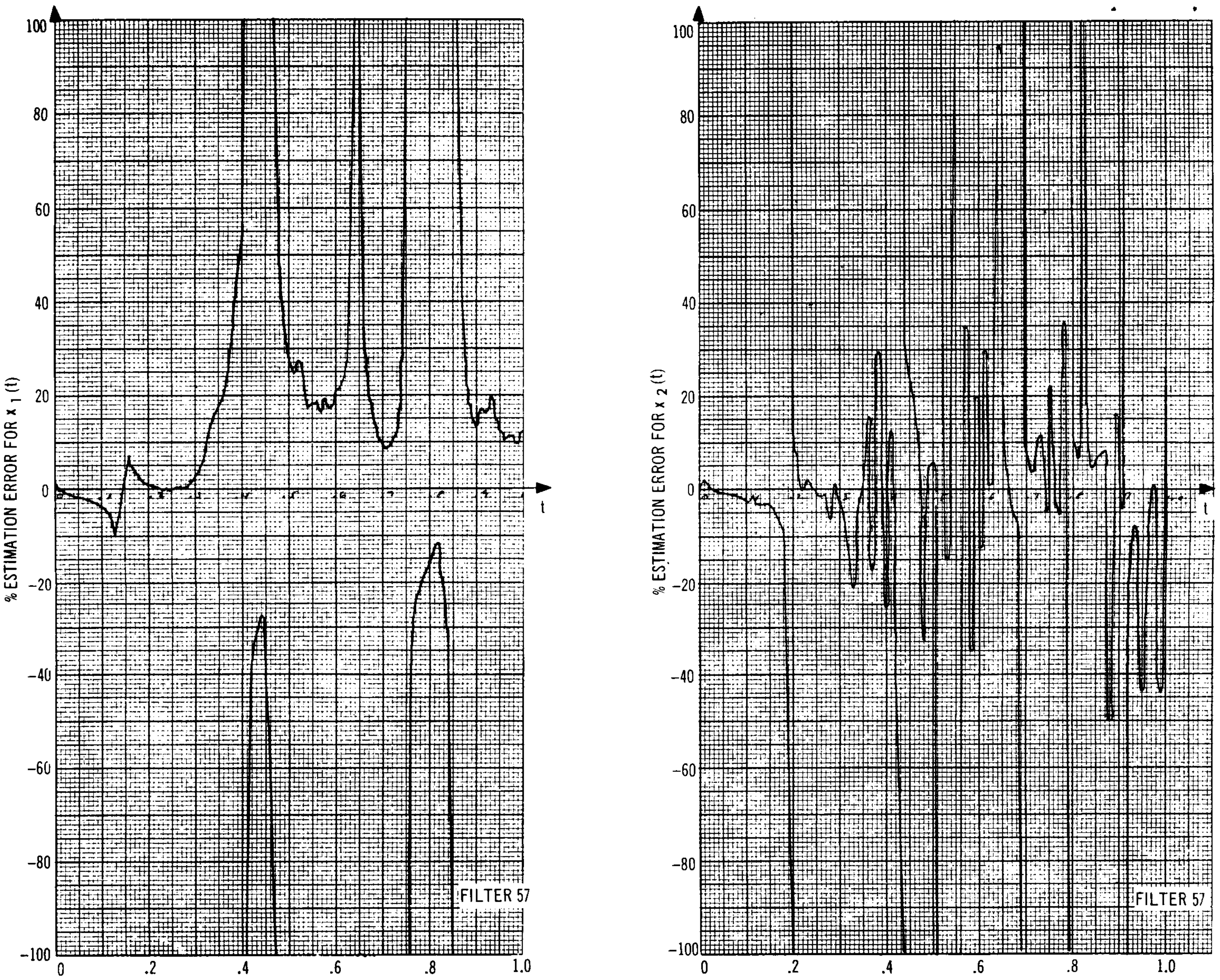
$$V_d = 3.84 \times 10^{-12} \tag{50c}$$

During the design of the sub-optimal state estimator, decisions were made concerning the relative qualities of the 58 different designs. These decisions were made, for the most part, by observing the data output from the IBM 7094 simulation. The data consisted of the actual states, estimated states, estimation errors, and percent estimation errors; the data was printed out every 0.01 units of time [T_s in eqs. (47) and (48) was 0.001 in these designs] during each unit of time. At the time that the Final Report was written, a concise way of summarizing the performance of each filter was discovered. It is especially useful for making decisions concerning the relative qualities of different filters, and is discussed, briefly, next.

First, the percent estimation errors for attitude, rate, and $\xi(t)$ are plotted as, for example, in fig. 23 for Filter 57. The covariances of Filter 57 are those given in eq. (50a, b, c). The discontinuous nature of these plots is due to the fact that the percent estimation error grows large when the actual function passes through zero (changes sign). Next, the cumulative time for which the percent error is contained within a specific range is plotted, as, for example, in fig. 24. The cumulative frequency function (which is an approximation of the probability distribution for | percent estimation error |) provides a concise means for comparing the different estimates from a single filter, and, in addition, provides a concise means for comparing estimates from different filters (fig. 25).

In fig. 24, observe that Filter 57 estimates $x_1(t)$ and $x_2(t)$ about the same, whereas it estimates $\xi(t)$ much better. For example, 20%-or-less estimation errors occur 90% of the time for $\xi(t)$ while they occur only 66% and 69% of the time for $x_1(t)$ and $x_2(t)$, respectively. It is also interesting that such similar plots occur for the cumulative frequencies associated with percent estimation errors for $x_1(t)$, $x_2(t)$, and $\xi(t)$, as compared with the rather diverse percent estimation error plots in fig. 23.

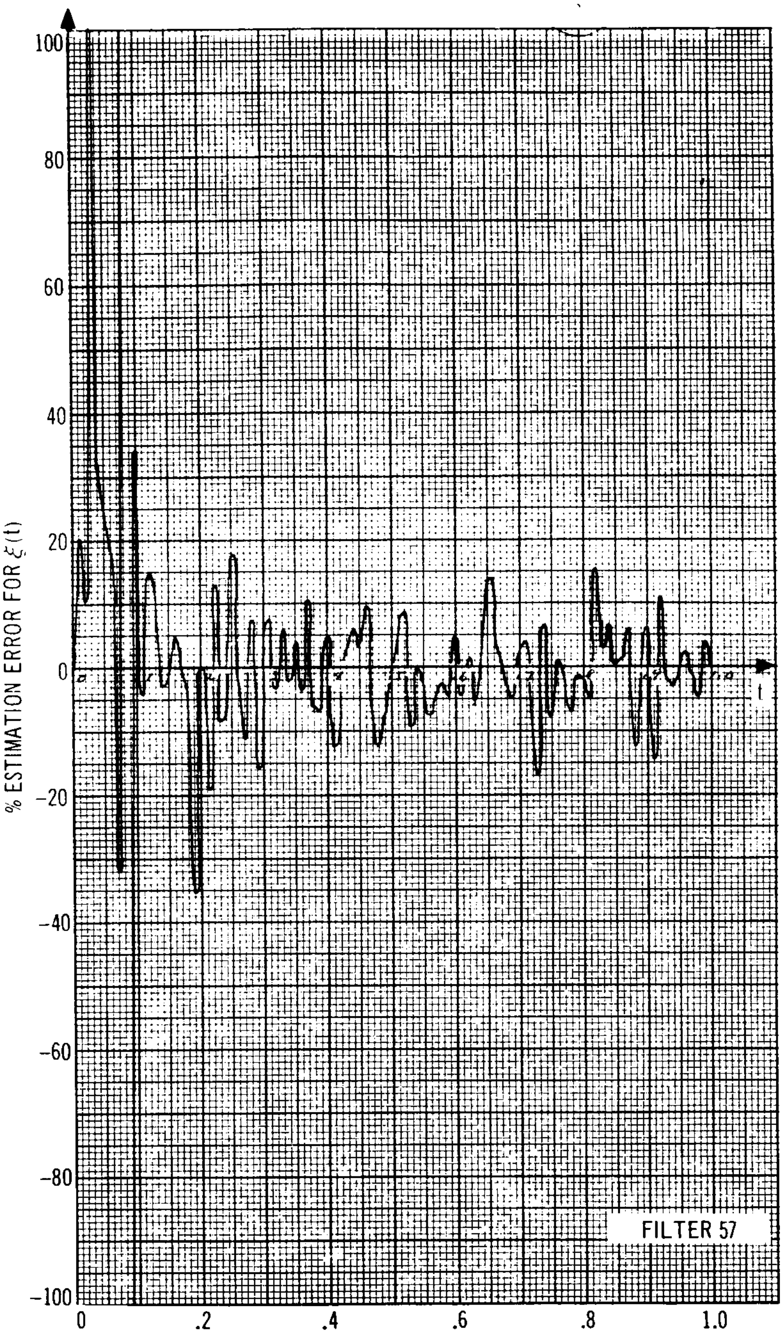
*a and α were set equal to 0.40 and -0.80, respectively, during the study.



(a)

(b)

Figure 23. Percent Estimation Errors



(c)

Figure 23. Percent Estimation Errors

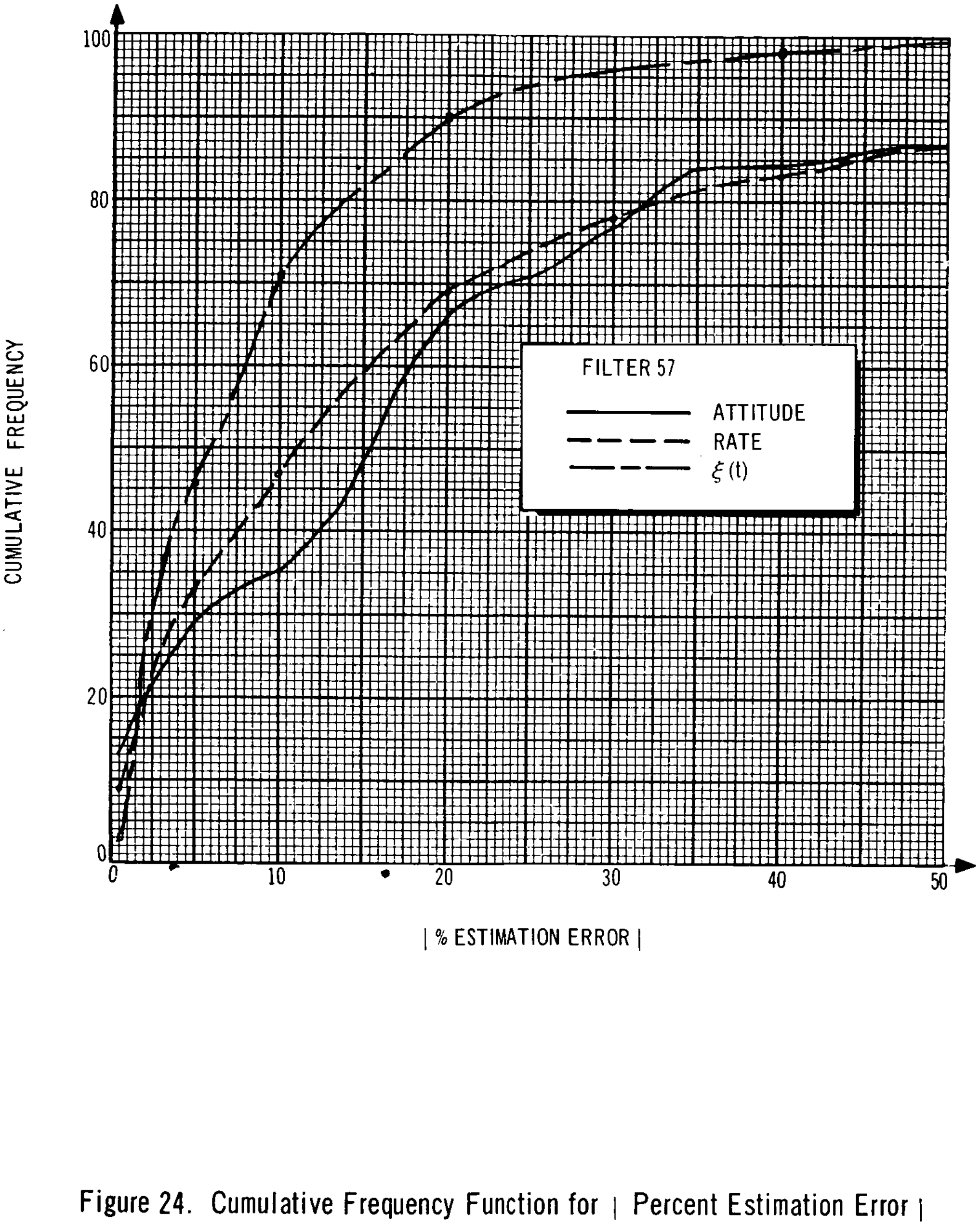


Figure 24. Cumulative Frequency Function for | Percent Estimation Error |

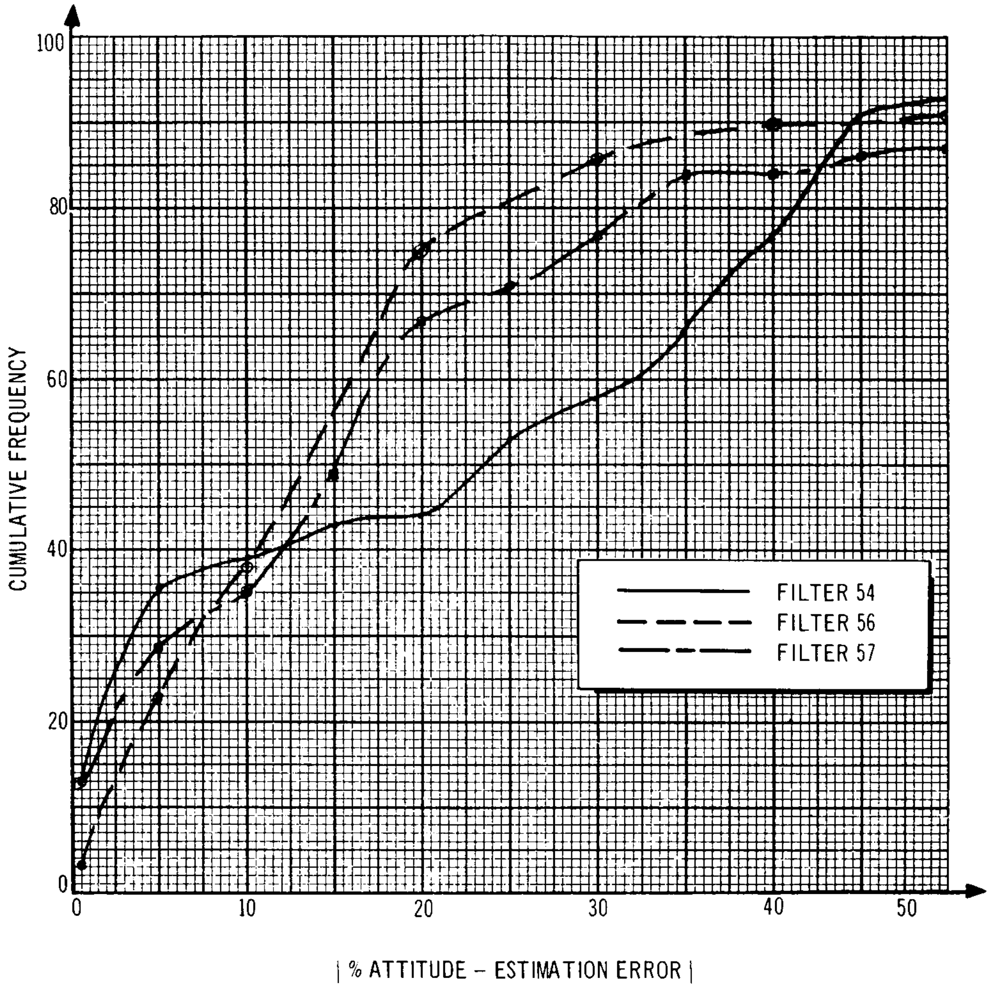


Figure 25. Cumulative Frequency Function for | Percent Attitude-Estimation Error |

Three different filters are compared in fig. 25. During the design, each one of these filters was judged good; however, Filters 56 and 57 were judged better than Filter 54. Observe that for values of|percent attitude - estimation error | less than 10% this judgement is wrong; Filter 54 is better than the other two filters. A wrong judgement occurs also for values of | percent attitude - estimation error | between 45 and 50%. On the other hand, the judgement is correct for values of | percent attitude - estimation error | between 15 and 40% ; hence, for large ranges of | percent attitude - estimation error | the cumulative frequency plots substantiate earlier judgements regarding the relative qualities of different filters.

State and Parameter Estimation. -- The combined estimator depicted in fig. B-2was simulated on the IBM 7094. The covariances in eq. (50)were used during this study; different experiments were performed in an effort to choose values for μ_{d11} and μ_{d22} in eq. (48).

In one experiment α was assumed known* and a was estimated. In another experiment a was assumed known** and α was estimated. The values of μ_{d11} and μ_{d22} were varied over wide ranges during these experiments (μ_{d11}in the first experiment and μ_{d22} in the second experiment); however, estimates for a and α were not achieved†. Some results from the experiment in which a was estimated are summarized in fig. 26. For each run (identified by a different filter number) the actual value of a was 0.40 and the initial estimate of a was chosen to be 0.30. For very small values of μ_{d11} the steady-state estimate, $\hat{a}_{s.s.}$, did not differ from the initial estimate. For larger values of μ_{d11}, however, the steady-state estimate was worse than the initial estimate. Time did not permit a more extensive study; one, for example, in which the effects of different initial estimates on the steady-state estimate are studied (since the combined estimator is nonlinear, initial conditions are important).

It was also observed that, regardless of how poorly the filter estimated a(t) and α(t), it still estimated x_1(t), x_2(t) and ξ(t) relatively well. This suggests that the state estimator is relatively insensitive to a and α. That this is indeed the case was demonstrated in an experiment during which wrong values for a and α were used in the state estimator. Some results from this experiment are tabulated in Table V for attitude errors. Similar results were obtained, in each case, for x_2(t) and ξ(t).

*μ_{d22} was set equal to zero.

**$\mu_{d11}$was set equal to zero.

†γ_1(t) and γ_2(t), in eq. (B17)were set equal to zero during these experiments.

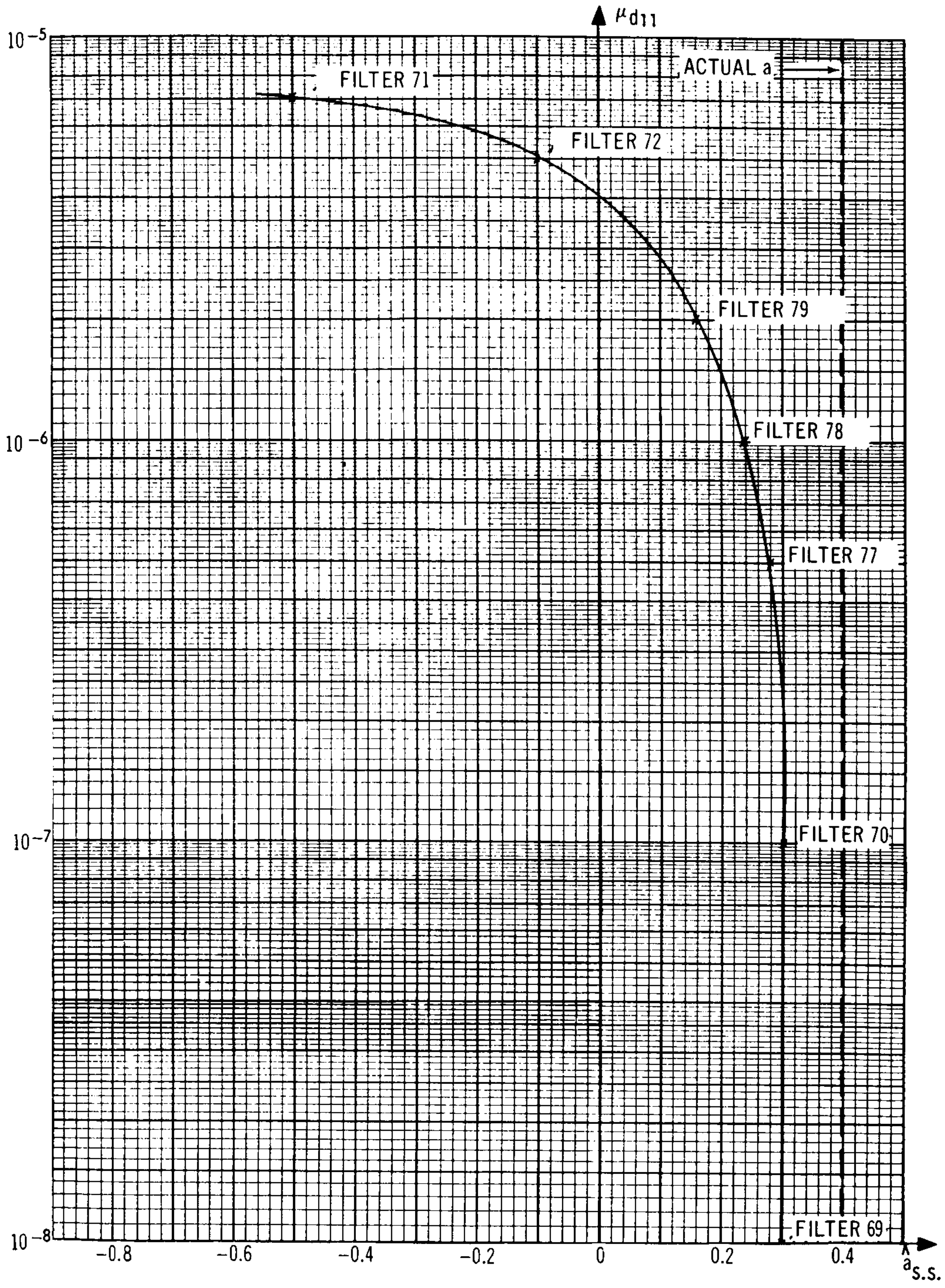


Figure 26. Steady-State Value of $\hat{a}$, $\hat{a}_{s.s.}$, For Different Filters

TABLE V

ESTIMATES OF ATTITUDE AS A FUNCTION OF TIME -- FOR VARIOUS MISMATCHES IN a AND α[*], [**]

t		a = 0.40 α = -0.80	a = -0.10 α = -0.80	a = 0.80 α = -0.80	a = 0.40 α = -0.20	a = 0.80 α = -0.20	a = 0.40 α = -1.50	a = -0.10 α = -1.50
0	$x_1 \cdot 10^{-4}$	5.84	5.84	5.80	5.80	5.80	5.80	5.80
0.05	$x_1 \cdot 10^{-4}$	3.72	3.72	3.71	3.72	3.72	3.72	3.73
0.10	$x_1 \cdot 10^{-5}$	9.21	9.23	9.19	9.20	9.18	9.21	9.23
0.15	$x_1 \cdot 10^{-5}$	-1.56	-1.57	-1.55	-1.58	-1.59	-1.53	-1.57
0.20	$x_1 \cdot 10^{-5}$	-2.74	-2.74	-2.72	-2.75	-2.76	-2.72	-2.76
0.25	$x_1 \cdot 10^{-5}$	-1.34	-1.36	-1.33	-1.35	-1.34	-1.33	-1.36
0.30	$x_1 \cdot 10^{-6}$	-4.48	-4.89	-4.87	-4.82	-4.80	-4.95	-4.96
0.35	$x_1 \cdot 10^{-7}$	-1.47	-1.30	-1.57	-0.49	-0.45	-2.47	-2.20
0.40	$x_1 \cdot 10^{-6}$	3.98	4.06	3.91	4.11	4.09	3.84	4.00
0.45	$x_1 \cdot 10^{-7}$	5.97	6.53	5.53	6.81	6.68	5.02	6.00

[*] The attitude initial conditions for these runs were $x_1(0) = 5.82 \times 10^{-4}$ and $\hat{x}_1(0|-1) = 3.88 \times 10^{-4}$; $T_s = 0.005$.

[**] The actual plant parameters, for the runs in this table, are a = 0.40 and α = -0.80. The values of a and α listed in the table were used in the equations for the state estimator.

TABLE V (Continued)*,**

t		$a = 0.40$ $\alpha = -0.80$	$a = -0.10$ $\alpha = -0.80$	$a = 0.80$ $\alpha = -0.80$	$a = 0.40$ $\alpha = -0.20$	$a = 0.80$ $\alpha = -0.20$	$a = 0.40$ $\alpha = -1.50$	$a = -0.10$ $\alpha = -1.50$
0.50	$x_1 \cdot 10^{-7}$	-1.86	-2.20	-1.59	-1.64	-1.59	-2.10	-2.79
0.55	$x_1 \cdot 10^{-6}$	-4.37	-4.42	-4.32	-4.34	-4.33	-4.39	-4.50
0.60	$x_1 \cdot 10^{-6}$	-2.94	-2.93	-2.95	-2.84	-2.84	-3.05	-3.04
0.65	$x_1 \cdot 10^{-6}$	-3.57	-3.65	-3.51	-3.50	-3.49	-3.64	-3.79
0.70	$x_1 \cdot 10^{-6}$	-7.73	-7.77	-7.69	-7.60	-7.59	-7.87	-7.96
0.75	$x_1 \cdot 10^{-6}$	-4.60	-4.63	-4.58	-4.42	-4.41	-4.81	-4.87
0.80	$x_1 \cdot 10^{-6}$	-2.03	-2.02	-2.06	-1.80	-1.80	-2.29	-2.28
0.85	$x_1 \cdot 10^{-6}$	1.92	1.95	1.89	2.16	2.16	1.64	1.70
0.90	$x_1 \cdot 10^{-6}$	-1.14	-1.15	-1.13	-0.94	-0.94	-1.36	-1.39
0.95	$x_1 \cdot 10^{-6}$	-3.03	-3.01	-3.04	-2.80	-2.80	-3.29	-3.27
1.00	$x_1 \cdot 10^{-7}$	9.65	9.63	9.68	11.77	11.78	7.23	7.13

*The attitude initial conditions for these runs were $x_1(0) = 5.82 \times 10^{-4}$ and $\hat{x}_1(0|-1) = 3.88 \times 10^{-4}$; $T_s = 0.005$.

**The actual plant parameters, for the runs in this table, are $a = 0.40$ and $\alpha = -0.80$. The values of a and α listed in the table were used in the equations for the state estimator.

TABLE V (Continued)*

t		$a = -0.30$ $\alpha = -0.40$	$a = 0.40$ $\alpha = -0.80$	$a = -0.10$ $\alpha = -0.80$	$a = 0.80$ $\alpha = -0.80$	$a = 0.40$ $\alpha = -0.20$	$a = 0.80$ $\alpha = -0.20$	$a = 0.40$ $\alpha = -1.50$	$a = -0.10$ $\alpha = -1.50$
0	$x_1 \cdot 10^{-4}$	5.80	5.80	5.80	5.80	5.80	5.80	5.80	5.80
0.05	$x_1 \cdot 10^{-4}$	4.80	4.80	4.80	4.80	4.80	4.80	4.79	4.80
0.10	$x_1 \cdot 10^{-4}$	2.75	2.75	2.75	2.74	2.75	2.74	2.74	2.75
0.15	$x_1 \cdot 10^{-4}$	1.08	1.07	1.07	1.07	1.07	1.06	1.07	1.07
0.20	$x_1 \cdot 10^{-6}$	7.97	7.51	7.80	7.28	7.23	6.80	7.84	7.82
0.25	$x_1 \cdot 10^{-5}$	-2.99	-3.01	-3.00	-3.02	-3.03	-3.05	-2.99	-3.00
0.30	$x_1 \cdot 10^{-5}$	-3.45	-3.46	-3.46	-3.47	-3.47	-3.47	-3.46	-3.47
0.35	$x_1 \cdot 10^{-5}$	-2.34	-2.35	-2.35	-2.34	-2.33	-2.33	-2.35	-2.36
0.40	$x_1 \cdot 10^{-6}$	-7.23	-7.33	-7.32	-7.34	-7.12	-7.07	-7.57	-7.50
0.45	$x_1 \cdot 10^{-7}$	-9.95	-10.88	10.70	-10.99	-8.80	-8.35	-13.24	-12.37

*The actual plant parameters, for the runs in this table, are $a = -0.30$ and $\alpha = -0.40$.

TABLE V (Continued)*

t		a = -0.30 α = -0.40	a = 0.40 α = -0.80	a = -0.10 α = -0.80	a = 0.80 α = -0.80	a = 0.40 α = -0.20	a = 0.80 α = -0.20	a = 0.40 α = -1.50	a = -0.10 α = -1.50
0.50	$x_1 \cdot 10^{-6}$	1.97	1.95	1.89	1.99	2.08	2.16	1.80	1.74
0.55	$x_1 \cdot 10^{-6}$	-3.09	-3.10	-3.17	-3.04	-2.98	-2.96	-3.22	-3.34
0.60	$x_1 \cdot 10^{-6}$	-4.90	-5.01	-5.02	-5.00	-4.83	-4.83	-5.22	-5.24
0.65	$x_1 \cdot 10^{-6}$	-6.74	-6.79	-6.89	-6.71	-6.62	-6.60	-6.98	-7.17
0.70	$x_1 \cdot 10^{-5}$	-1.33	-1.34	-1.35	-1.33	-1.31	-1.31	-1.37	-1.39
0.75	$x_1 \cdot 10^{-5}$	-1.20	-1.22	-1.23	-1.22	-1.18	-1.81	-1.26	-1.28
0.80	$x_1 \cdot 10^{-6}$	-8.64	-8.94	-8.96	-8.91	-8.45	-8.43	-9.49	-9.55
0.85	$x_1 \cdot 10^{-6}$	-1.64	-2.01	-2.00	-2.01	-1.45	-1.44	-2.64	-2.62
0.90	$x_1 \cdot 10^{-6}$	-1.21	-1.54	-1.56	-1.53	-1.00	-0.99	-2.16	-2.20
0.95	$x_1 \cdot 10^{-6}$	-3.44	-3.82	-3.81	-3.82	-3.25	-3.24	-4.47	-4.46
1.00	$x_1 \cdot 10^{-7}$	8.64	4.97	4.88	4.98	10.58	10.60	-1.55	-1.73

*The actual plant parameters, for the runs in this table, are a = -0.30 and α = -0.40.

TABLE V (Continued)*

t		$a = 0$ $\alpha = -0.40$	$a = 0.40$ $\alpha = -0.80$	$a = -0.10$ $\alpha = -0.80$	$a = 0.80$ $\alpha = -0.80$	$a = 0.40$ $\alpha = -0.20$	$a = 0.80$ $\alpha = -0.20$	$a = 0.40$ $\alpha = -1.50$	$a = -0.10$ $\alpha = -1.50$
0	$x_1 \cdot 10^{-4}$	5.80	5.80	5.80	5.80	5.80	5.80	5.80	5.80
0.05	$x_1 \cdot 10^{-4}$	4.74	4.73	4.74	4.73	4.74	4.73	4.73	4.74
0.10	$x_1 \cdot 10^{-4}$	2.61	2.60	2.61	2.60	2.61	2.60	2.61	2.62
0.15	$x_1 \cdot 10^{-5}$	9.41	9.38	9.42	9.35	9.35	9.31	9.41	9.42
0.20	$x_1 \cdot 10^{-8}$	2.68	-10.87	11.65	-28.48	-37.60	-75.69	21.62	13.18
0.25	$x_1 \cdot 10^{-5}$	-3.19	-3.18	-2.18	-3.19	-3.21	-3.23	-3.16	-3.19
0.30	$x_1 \cdot 10^{-5}$	-3.28	-3.28	-3.28	-3.28	-3.28	-3.29	-3.28	-3.29
0.35	$x_1 \cdot 10^{-5}$	-2.05	-2.05	-2.06	-2.05	-2.04	-2.04	-2.06	-2.07
0.40	$x_1 \cdot 10^{-6}$	-4.91	-5.02	-5.01	-5.04	-4.18	-4.77	-5.26	-5.17
0.45	$x_1 \cdot 10^{-7}$	1.66	0.53	0.76	0.34	2.49	2.85	-1.70	-0.75

*The actual plant parameters, for the runs in this table, are $a = 0$ and $\alpha = -0.40$.

TABLE V (Continued)*

t		$a = 0$ $\alpha = -0.40$	$a = 0.40$ $\alpha = -0.80$	$a = -0.10$ $\alpha = -0.80$	$a = 0.80$ $\alpha = -0.80$	$a = 0.40$ $\alpha = -0.20$	$a = 0.80$ $\alpha = -0.20$	$a = 0.40$ $\alpha = -1.50$	$a = -0.10$ $\alpha = -1.50$
0.50	$x_1 \cdot 10^{-6}$	2.21	2.16	2.11	2.19	2.28	2.31	2.02	1.97
0.55	$x_1 \cdot 10^{-6}$	-3.27	-3.01	-3.37	-3.26	-3.21	-3.19	-3.42	-3.54
0.60	$x_1 \cdot 10^{-6}$	-5.07	-5.18	-5.19	-5.18	-5.01	-5.01	-5.37	-5.40
0.65	$x_1 \cdot 10^{-6}$	-6.77	-6.84	-6.94	-6.76	-6.68	-6.66	-7.02	-7.21
0.70	$x_1 \cdot 10^{-5}$	-1.31	-1.32	-1.33	-1.32	-1.30	-1.29	-1.35	-1.34
0.75	$x_1 \cdot 10^{-5}$	-1.16	-1.18	-1.19	-1.18	-1.15	-1.15	-1.22	-1.24
0.80	$x_1 \cdot 10^{-6}$	-8.28	-8.57	-8.60	-8.55	-8.10	-8.09	-9.09	-9.15
0.85	$x_1 \cdot 10^{-6}$	-1.51	-1.86	-1.85	-1.87	-1.33	-1.32	-2.46	-2.44
0.90	$x_1 \cdot 10^{-6}$	-1.42	-1.74	-1.76	-1.72	-1.23	-1.22	-2.31	-2.35
0.95	$x_1 \cdot 10^{-6}$	-3.80	-4.15	-4.14	-4.16	-3.62	-3.61	-4.76	-4.76
1.00	$x_1 \cdot 10^{-7}$	4.24	0.77	0.71	0.81	6.03	6.06	-5.29	-5.47

*The actual plant parameters, for the runs in this table, are $a = 0$ and $\alpha = -0.40$

The reason for this insensitivity is that a(t) and α(t) are very small compared to the feedback gains λ_1, λ_2, and λ_3; hence, the closed-loop response is insensitive to variations in a and α. This can also be verified, in part, analytically. For example, $\underline{\hat{q}}(n|n)$ in eq. (42) can be written as

$$\underline{\hat{q}}(n|n) = \overline{K}^*(n)\underline{z}(n) + \left[I-\overline{K}^*(n)M^*\right]. \left[\Phi(n,n-1) + \underline{\Delta}(n,n-1)\Lambda^*\right]\underline{\hat{q}}(n-1|n-1). \qquad (51)$$

Eq. (51) follows from eqs. (41) and (43) and the fact that

$$u(n) = \Lambda^*\underline{\hat{q}}(n-1|n-1) \qquad (52)$$

where

$$\Lambda^* = (\lambda_1, \lambda_2, \lambda_3, 0, 0). \qquad (53)$$

It is relatively straightforward to show that in the matrix

$$\left[\Phi(n,n-1) + \underline{\Delta}(n,n-1)\Lambda^*\right]$$

the terms due to $\hat{a}(n-1|n-1)$ and $\hat{\alpha}(n-1|n-1)$ are negligible compared with the terms due to Λ^*, which substantiates the above conjecture.

The fact that the combined estimator did not provide estimates for a(t) and α(t) in this case does not mean that the estimator cannot be used in other applications. This estimator was used with success by Waymeyer, et al in ref. 1, and is currently being used with success at Douglas in connection with an adaptive controller for a maneuverable re-entry vehicle. In both of these applications, however, the estimated parameters are either of the same order of magnitude or larger than the feedback gains.

Nominal Controller Simulation Studies

In this subsection (which is a continuation of the Simulation Studies in Section 2), the performance of the system under the action of the nominal controller is, once again, investigated. The main difference between the simulations described below and those described in Section 2 is, however, that here the loop is closed around the output of the state estimator (described in the preceding subsection), whereas in Section 2 the loop was closed around the plant. The experiments that are described below were performed in order to observe the effects of the following on the attitude errors:

(1) State-estimation errors.

(2) State-estimation errors <u>and</u> parameter-mismatch errors.

Seven experiments were performed; they are summarized in Table VI. The first four were concerned with the effects of state-estimation errors on the attitude error. The simulated system for these experiments is depicted in fig. 27 (compare with fig. 11). The last three experiments were concerned with the combined effects of state- and parameter-estimation errors. Since it was not possible to obtain estimates for a(t) and $\alpha(t)$ using the combined estimator described in the preceding subsection, and since time did not permit other means for estimating the parameters to be studied, pseudo-estimation errors were assumed. The simulated system for these experiments is depicted in fig. 28.

Observe the difference between figs. 27 and 28. In fig. 28 the actual plant parameters are both corrupted in some statistical fashion. The corrupted plant parameter vector, or, pseudo-estimated plant parameter vector, denoted $\hat{\underline{P}}'_{A_a}(t|t)$, is

$$\hat{\underline{P}}'_{A_a}(t|t) = \underline{P}_{A_a}(t) + \tilde{\underline{P}}'_{A_a}(t), \qquad (54)$$

where $\tilde{\underline{P}}'_{A_a}(t)$ is a (2x1) pseudo-estimation error vector. In Experiments 10 and 11 the components of $\tilde{\underline{P}}'_{A_a}(t)$ are uniformly and normally distributed, respectively. In these experiments

$$E\left\{\tilde{\underline{P}}'_{A_a}(t)\right\} = \underline{0}.$$

The density functions for the two distributions are depicted in fig. 29. In Experiment 12 the components of $\tilde{\underline{P}}'_{A_a}(t)$ are normally distributed and

$$E\left\{\tilde{\underline{P}}'_{A_a}(t)\right\} \neq \underline{0}.$$

Both the interval of the uniformly distributed error and the variance of the normally distributed error were chosen so that $(\hat{a}', \hat{\alpha}')^T$ could lie in a control situation different from the one in which the actual parameters are in. For example, if a = o and α = -0. 50, so that the plant is really in control situation 45 (fig. 10), $(\hat{a}', \hat{a}')^T$ may be in either control situation 45, 44, 46, 54, 55, 56, 34, 35, or 36 (assuming that $E\left\{\tilde{\underline{P}}'_{A_a}(t)\right\} = \underline{0}$).

Table VI
NOMINAL CONTROLLER EXPERIMENTS (CONTINUED)[1]

Experiment #	6 (Figure 27)	7 (Figure 27)	8 (Figure 27)	9 (Figure 27)	10 (Figure 28)	11 (Figure 28)	12 (Figure 28)
$\alpha(t)$	$-0.55 - 0.45 \sin 2\pi t$	As in 6	As in 6	As in 6	As in 6	As in 6	As in 6
a(t)	$0 \vert 0.2 \vert 0.4 \vert -0.2 \vert -0.4$	$0 \vert 0.4 \vert -0.4$	0	0	As in 6	As in 6	-0.20
$\Pr\{\lvert \ell \rvert \le 10^{-4}\}$[2, 3]	0.96	0.96	$V = 0.48 \times 10^{-8}$, 0.48×10^{-10}, 0.48×10^{-14}, 0.48×10^{-18}	$V = 0.48 \times 10^{-12}$	0.96	0.96	0.96
T[4]	0.05	0.05	0.05	0.05	0.05	0.05	0.05
Running time	100T	100T	100T	100T	100T	100T	100T
Pseudo parameter-estimation error	None	None	None	None	Uniformly distributed[5] $E\{\tilde{a}'\} = E\{\tilde{\alpha}'\} = 0$	Normally distributed[6] $E\{\tilde{a}'\} = E\{\tilde{\alpha}'\} = 0$	Normally distributed $E\{\tilde{a}'\} = E\{\tilde{\alpha}'\} =$ $\pm 0.05, \pm 0.10, \pm 0.15$
Measurement Noise	None	$r_{d11} = r_{d22} = 10^{-16}$	$r_{d11} = r_{d22} = 10^{-16}$	$r_{d11} = r_{d22} = r$, $r = 10^{-14}, 10^{-10}, 10^{-6}$	None	None	None

[1] The numbering for the experiments in this Table is continued from Table IV.

[2] See the third footnote under Table IV.

[3] A value of $V = 0.48 \times 10^{-8}$ was chosen for $p_\ell = 0.96$.

[4] $\underline{x}_a(0) = (60 \text{ arcsec}, 0, 0)^T$ for all 7 experiments.

[5] See figure 29a.

[6] See figure 29b.

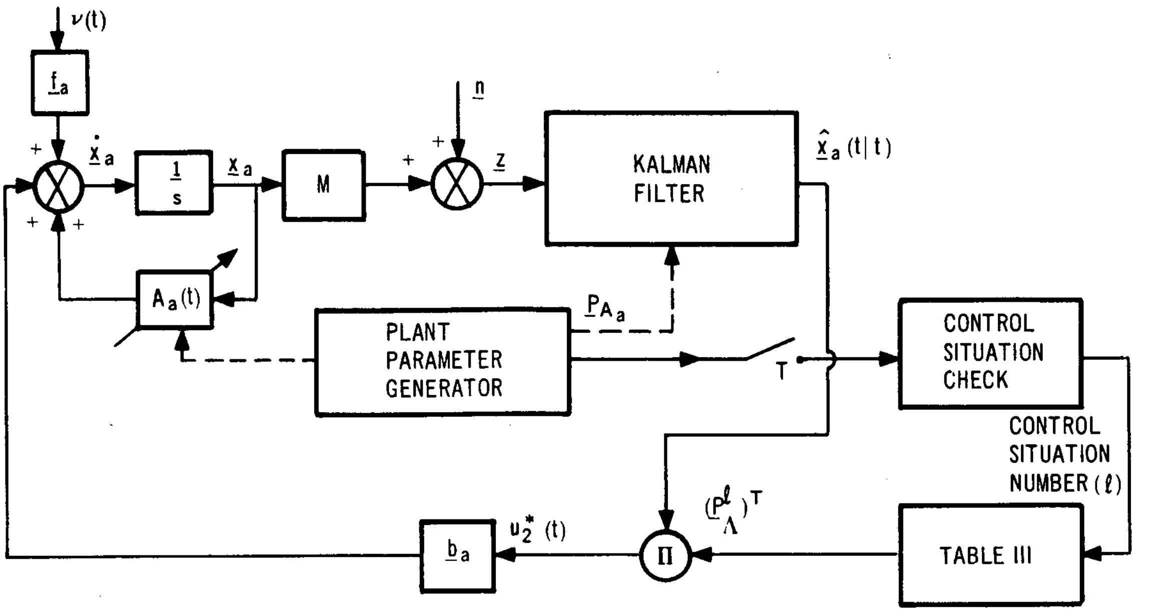


Figure 27. System Simulation for Nominal Control Study --Loop Closed Around Filter

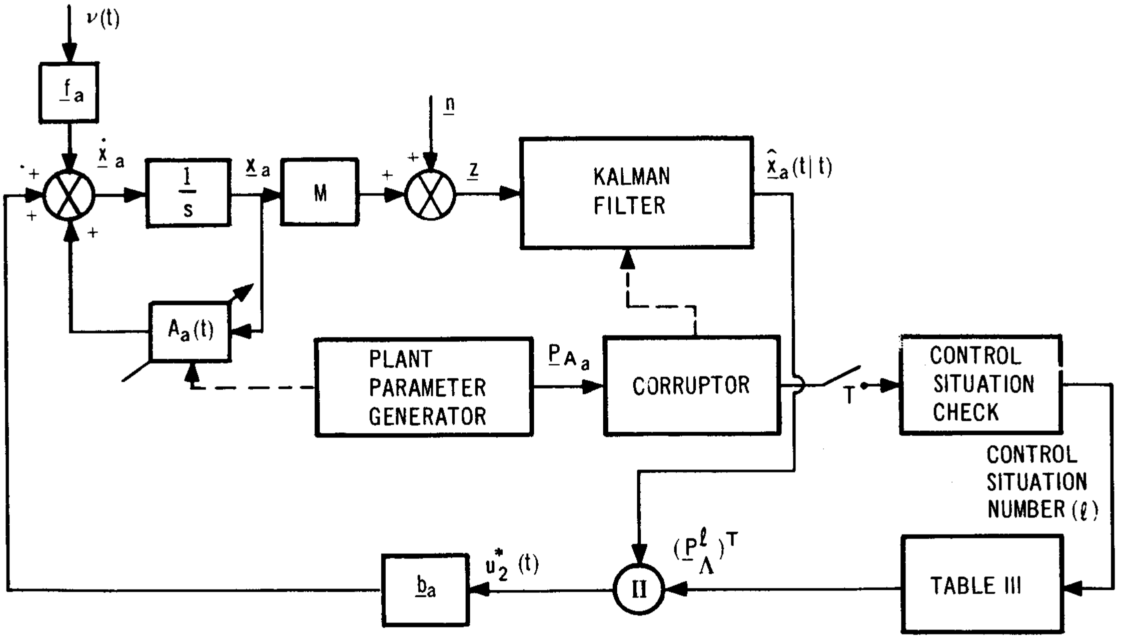


Figure 28. System Simulation for Nominal Control Study--Loop Closed Around Filter and Parameters Mismatched

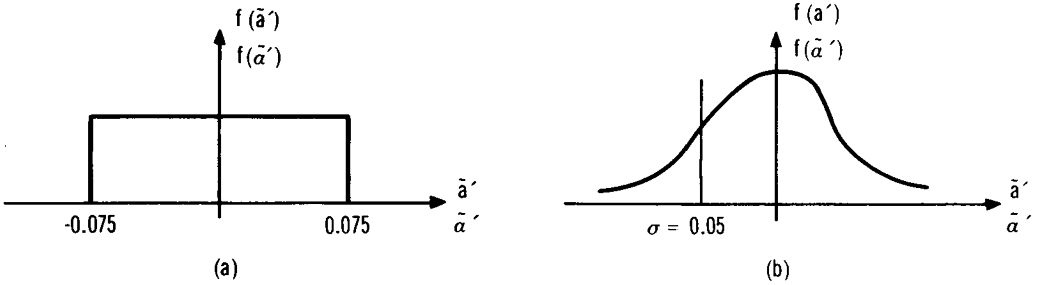


Figure 29. Density Functions for Elements of Pseudo-Estimation Vector

In Experiments 6 to 9 the actual plant parameters (generated) are utilized in both the Kalman filter and the mechanism for updating the control gains. In Experiments 10 to 12, on the other hand, corrupted (mismatched) plant parameters are utilized in both of these places, as would occur if a(t) and $\alpha(t)$ were actually estimated.

The attitude error results for Experiment 6 are summarized in fig. 30 (the results for Experiment 7, which differs from Experiment 6 in that a small amount of measurement noise is introduced in the former, are omitted, since they are similar to the results in fig. 30). These results should be compared with those in fig. 14. Apparently, state estimation-errors degrade attitude error by from one to two orders of magnitude. This represents the effect of neglecting the estimation errors during the design of the weighting factor, ρ, in Appendix C [eq. (C13)].

In Section 2, it was shown that, for the system with the loop closed around the plant, $u_2^*(t) \rightarrow \lambda_3(t)\,\xi(t)$ after a short initial transient. For the system with the loop closed around the filter this approximation is no longer valid, due to estimation errors. In the random-limit-cycle mode, $u_2^*(t)$ is proportional not only to $\hat{\xi}(t|t)$ but also to $\hat{x}_1(t|t)$, and $\hat{x}_2(t|t)$, since $\hat{x}_1(t|t)$, $\hat{x}_2(t|t)$ and $\hat{\xi}(t|t)$ each settle to the same order of magnitude.

The next few paragraphs shall distinguish between the nominal control with the loop closed around the plant, $u_{2P}^*(t)$, and the nominal control with the loop closed around the filter, $u_{2F}^*(t)$. It shall be shown, for a typical case, that the control error signal $(u_{2F}^* - u_{2P}^*)$ does indeed degrade attitude error from an order of magnitude of 10^{-6} radians to an order of magnitude of about 10^{-5} radians; that is to say, this section will demonstrate that $(u_{2F}^* - u_{2P}^*)$ leads to an attitude error of approximately $10^{-5} - 10^{-6} = 0.000009$ radians.

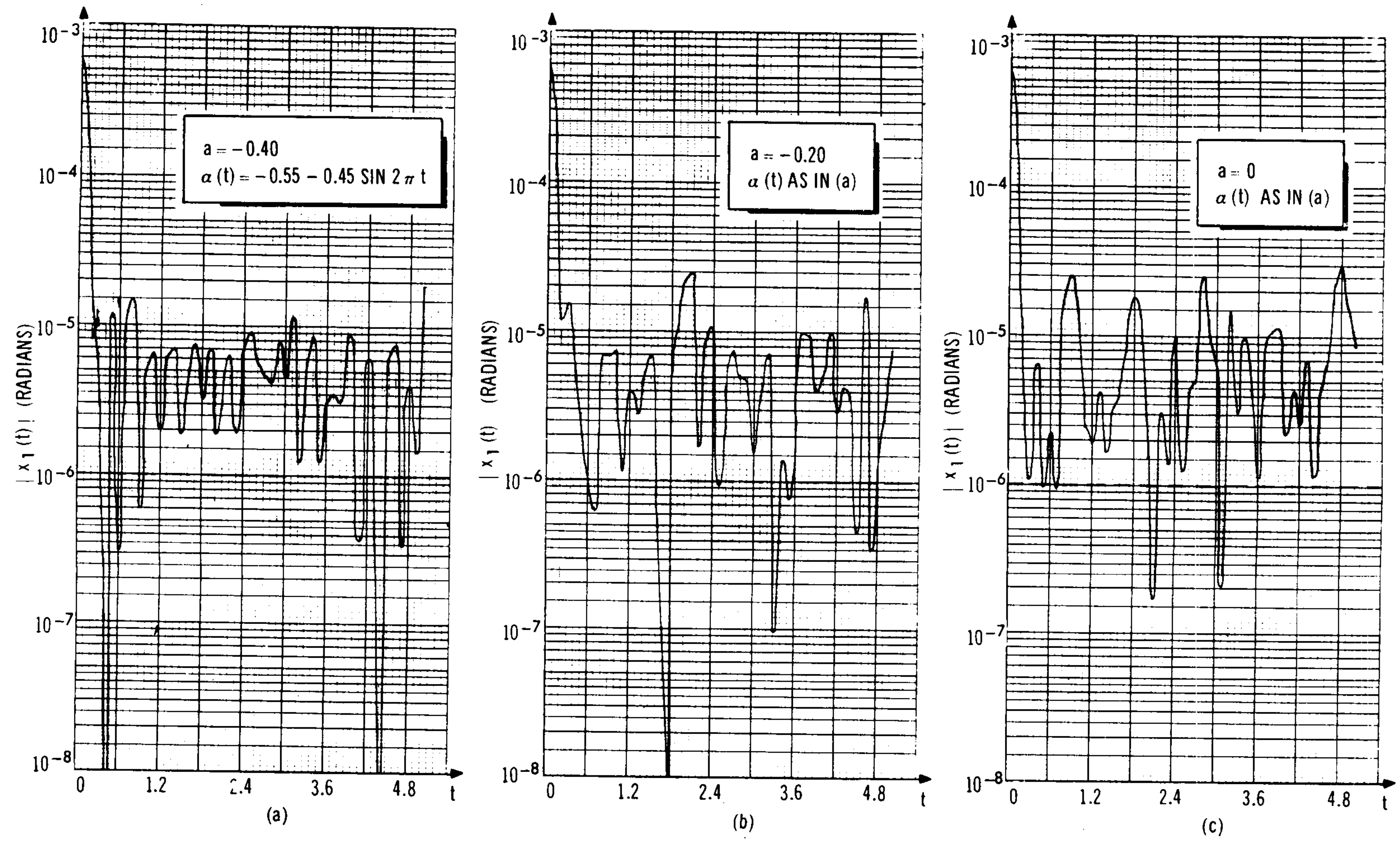


Figure 30. $|x_1(t)|$ for 5 Values of a (Experiment 6)

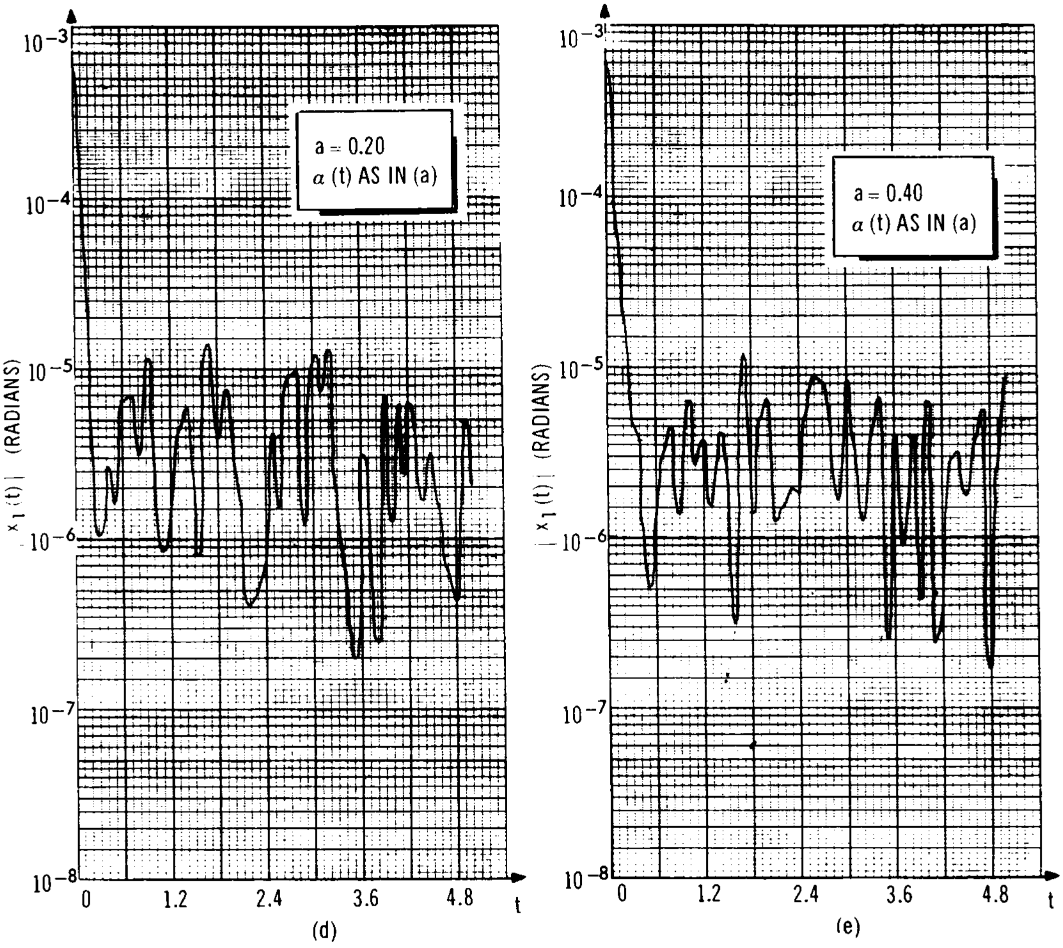


Figure 30. $|x_1(t)|$ for 5 Values of a (Experiment 6)

The functions u^*_{2F} and u^*_{2P} are plotted in fig. 31 for the case when the open-loop system is a double integrator. u^*_{2P} was obtained from a simulation of the system in fig. 11, whereas u^*_{2F} was obtained from a simulation of the system in fig. 27. The initial conditions, integration times, and random sequences used in both simulations were identical. Observe the close similarity between u^*_{2F} and u^*_{2P} and observe, also, the small amplitude and oscillatory behavior of the control error signal, which is also plotted in fig. 31.

In order to demonstrate that $(u^*_{2F} - u^*_{2P})$ leads to an attitude error degradation of 9 x 10^{-6} radians, it is first approximated as[†]

$$(u^*_{2F} - u^*_{2P}) \sim \pm 0.005 \sin [2\pi (3.5)t]. \tag{55}$$

Denoting the component of the attitude error due to $(u^*_{2F} - u^*_{2P})$ as x_{1FP}, it follows that

$$x_{1FP} \sim \pm \iint 0.005 \sin 21.98\tau \, d\tau^2 \tag{56}$$

$$x_{1FP} \sim \pm 10.35 \times 10^{-6} \sin 21.98t. \tag{57}$$

The close agreement between the amplitude of x_{1FP} in eq. (57) and the attitude error degradation of 9 x 10^{-6} radians must be noted.

Experiments 8 and 9 represent disturbance and measurement-noise level sensitivity studies, respectively, and are closely related. The purpose of Experiment 8 was to determine the disturbance level for which attitude errors are just contained to within ±0.20 arcsec, with the loop closed around the filter. As expected, by lowering the disturbance noise level $[V = 0.48 \times 10^{-8}$, 0.48×10^{-10}, 0.48×10^{-14} and 0.48×10^{-18} correspond to disturbances of the order of magnitude of 10^{-4}, 10^{-5}, 10^{-7} and 10^{-9} lb-ft., respectively$]$ it is possible to contain attitude errors, even though state-estimation errors exist (fig. 32). The disturbance noise level for which attitude errors just begin to remain within ±0.20 arcsec is between 0.48×10^{-10} and 0.48×10^{-14}. A value of $V = 0.48 \times 10^{-12}$ was assumed for Experiment 9, the purpose of which was to determine how much measurement noise could be introduced before attitude errors would once again become greater in magnitude than 0.20 arcsec.

[†] The frequency of 3.5 cps was obtained by counting the number of positive peaks of $(u^*_{2F} - u^*_{2P})$ in fig. 31, and dividing this number by 5 units of time. The amplitude of 0.005 is an average value.

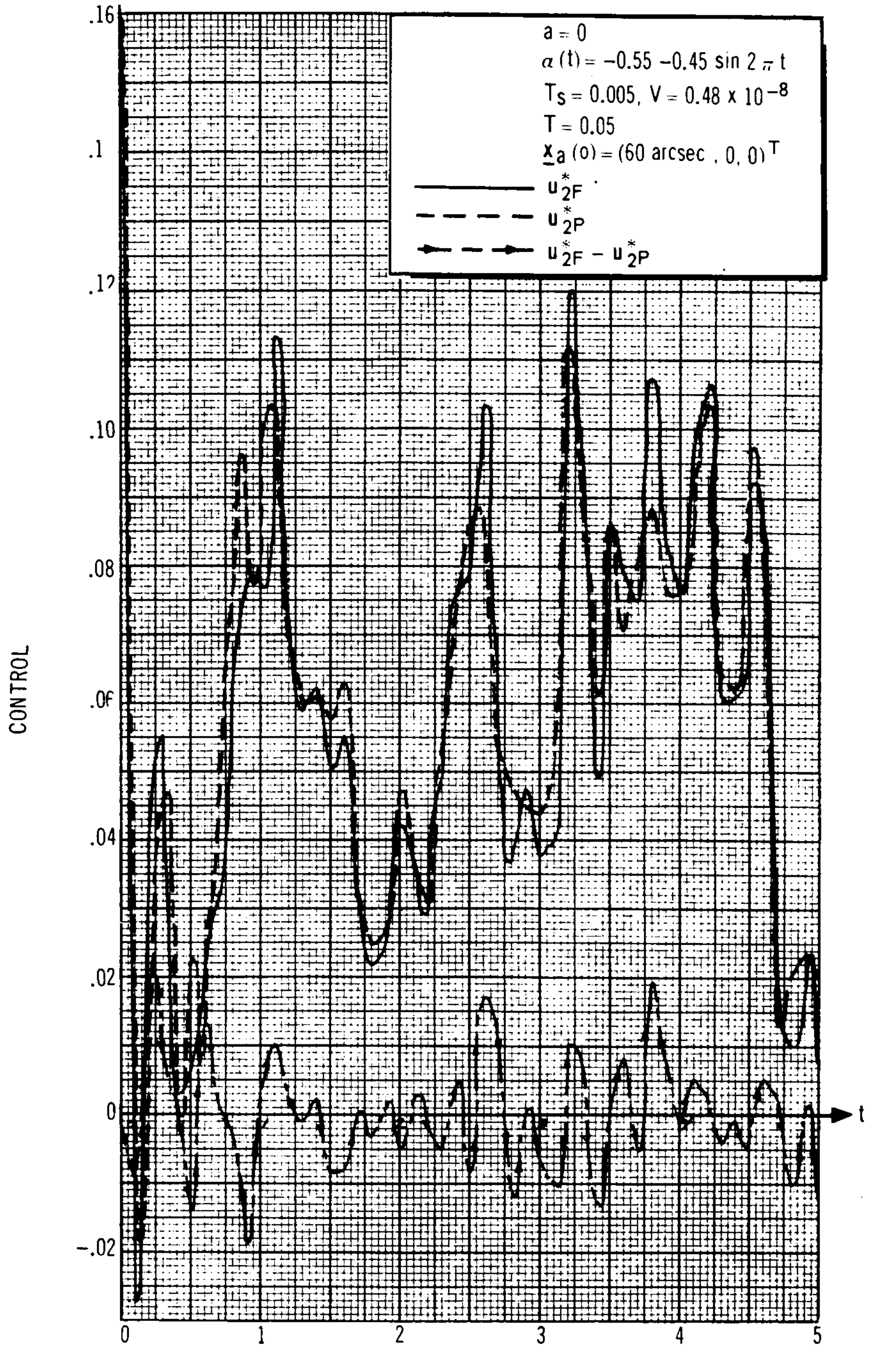


Figure 31. Control Versus Time for System with Loop Closed Around Filter (u^*_{2F}) and Closed Around Plant (u^*_{2P})

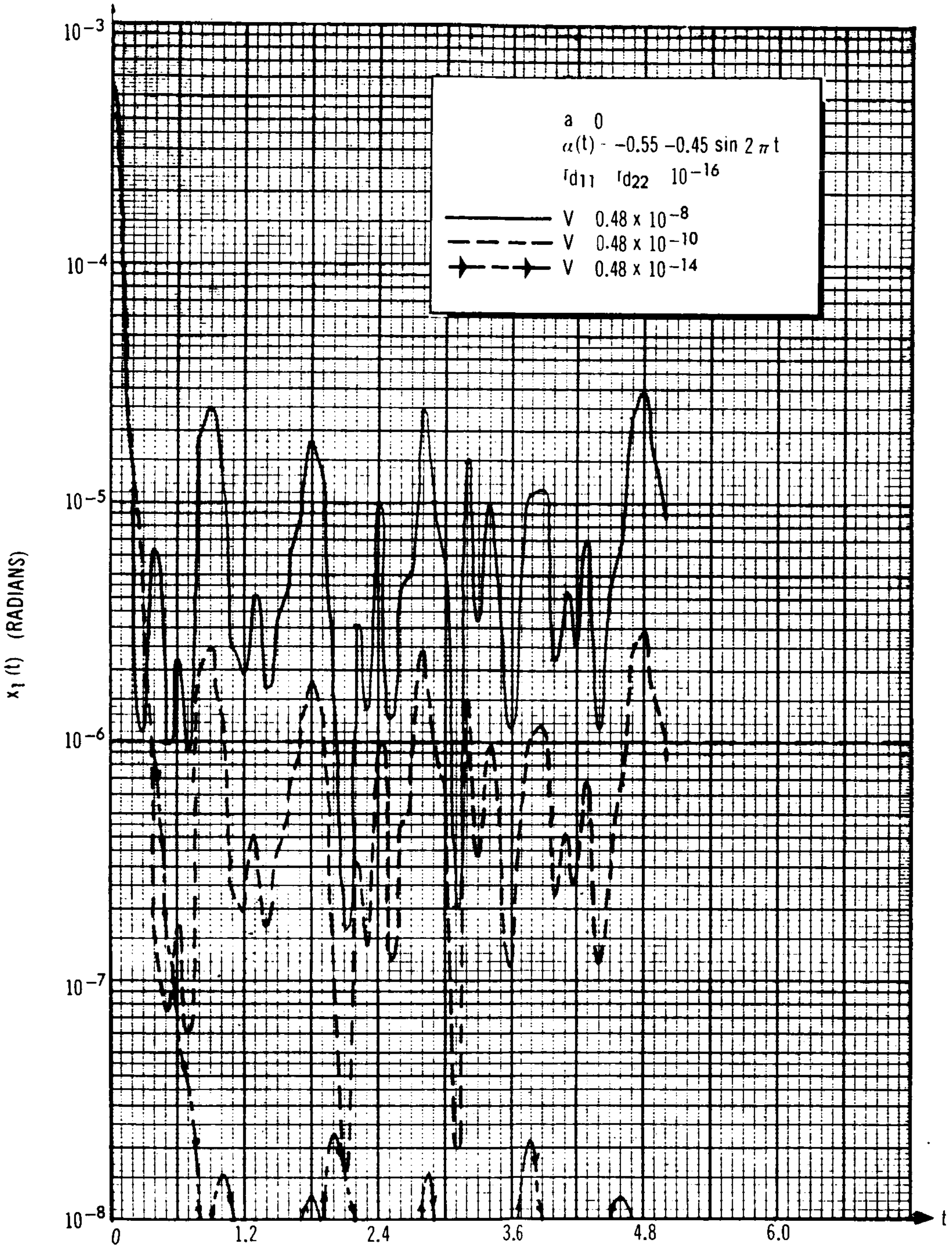


Figure 32. $|x_1(t)|$ for Three Different Disturbance Levels (Experiment 8)

As the measurement noise level increased, it was observed that the estimates of x_1, x_2, and ξ became poorer. This was especially true for the estimates of ξ, which were wrong by from three to four orders of magnitude when $r_{d11} = r_{d22} = 10^{-6}$. In this case, the control was much larger than it was when $r_{d11} = r_{d22} = 10^{-10}$ or 10^{-14}, larger by two orders of magnitude. The measurement noise level for which attitude errors just begin to exceed ±0. 20 arcsec is seen in fig. 33 to be between 10^{-10} and 10^{-14}.

The results for Experiment 10 are summarized in fig. 34. Due to the close similarity between the results obtained for Experiment 11 and those in fig. 34, the results for that experiment are omitted. Observe that zero mean parameter mismatches do not further degrade attitude errors from the degradation incurred by closing the loop around the filter. The purpose of Experiment 12 was to investigate the effect of non zero-mean parameter mismatches on the attitude errors. Some results from this experiment are depicted in fig. 35a. Note that even with fairly large parameter mismatches a further degradation in attitude error does not occur. Apparently, $|x_1(t)|$ is rather insensitive to variations in a(t) and $\alpha(t)$. An interesting experiment which would point up the effects of parameter mismatch (as distinguished from Experiment 12, where the effects of parameter mismatch and state-estimation errors are blended), is one in which the loop is closed around the plant, as in fig. 11, and various non zero-mean normally-distributed pseudo parameter-estimation errors are introduced. Time did not permit this experiment to be performed, however.

Due to the randomness of the plant parameter pseudo-estimation errors, the nominal feedback gains $\lambda_1(t)$, $\lambda_2(t)$ and $\lambda_3(t)$ in Experiments 10 to 12 are nonperiodic random sequences. This is illustrated in fig. 35b and 35c for Experiments 10 and 11, respectively. In Experiments 6 to 9, on the other hand, the feedback gains are periodic [with period equal to that of $\alpha(t)$] and vary exactly the same as the gains in Experiment 1 (figs. 19, 20 and 21 for example).

Summary and Conclusions

A state estimator, for use in the nominal controller, has been designed. It is sub-optimal, since the covariances which are used by the estimator do not correspond, in general, to the true noise and disturbance covariances (unknowns in this study). The state estimator was shown to be relatively insensitive to variations in a(t) and $\alpha(t)$, because of the largeness of the feedback gains as compared to the smallness of a(t) and $\alpha(t)$. Due to this insensitivity, it was not possible to obtain estimates of a(t) and $\alpha(t)$ from the combined estimator.

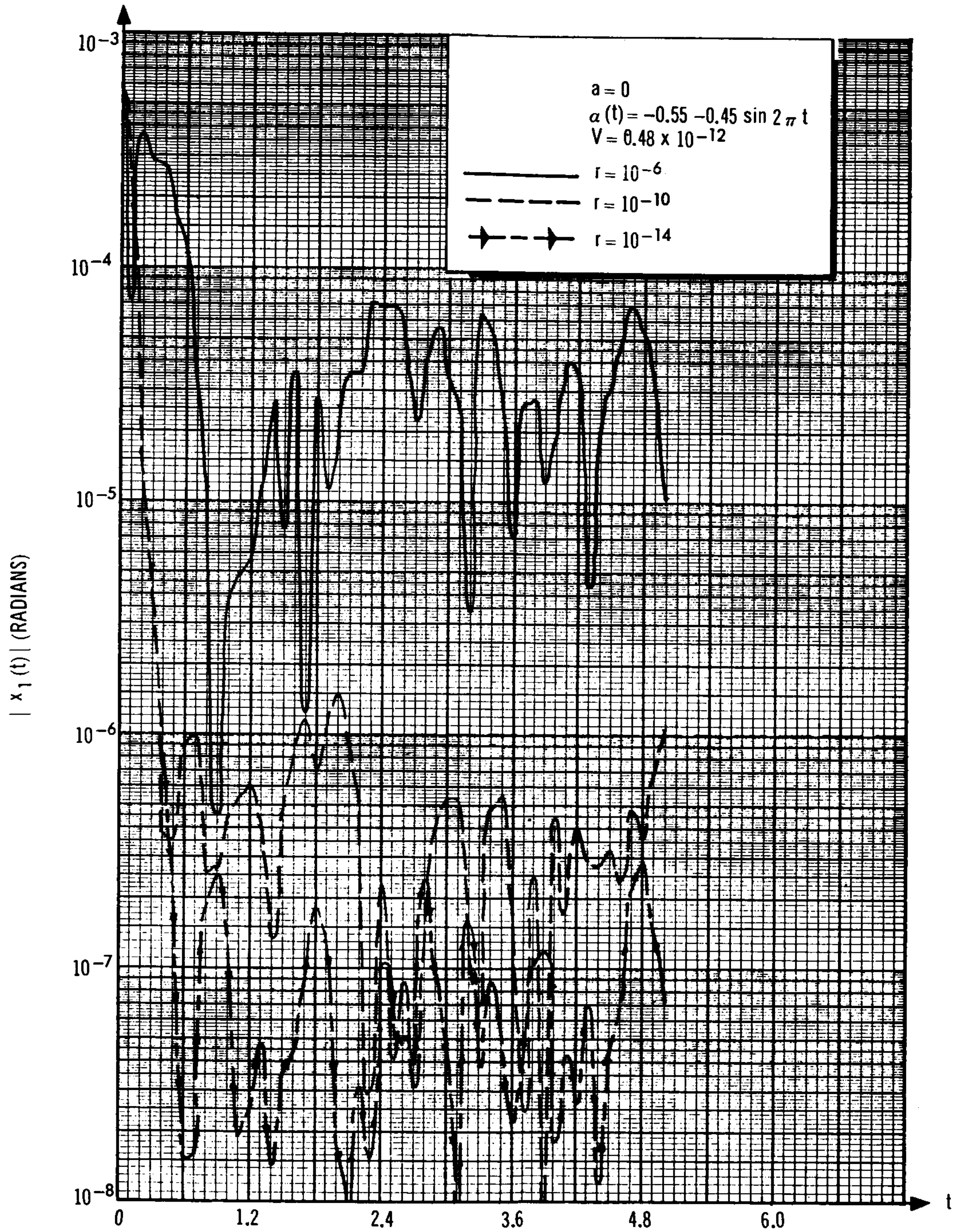


Figure 33. $|x_1(t)|$for Three Different Measurement Noise Levels (Experiment 9)

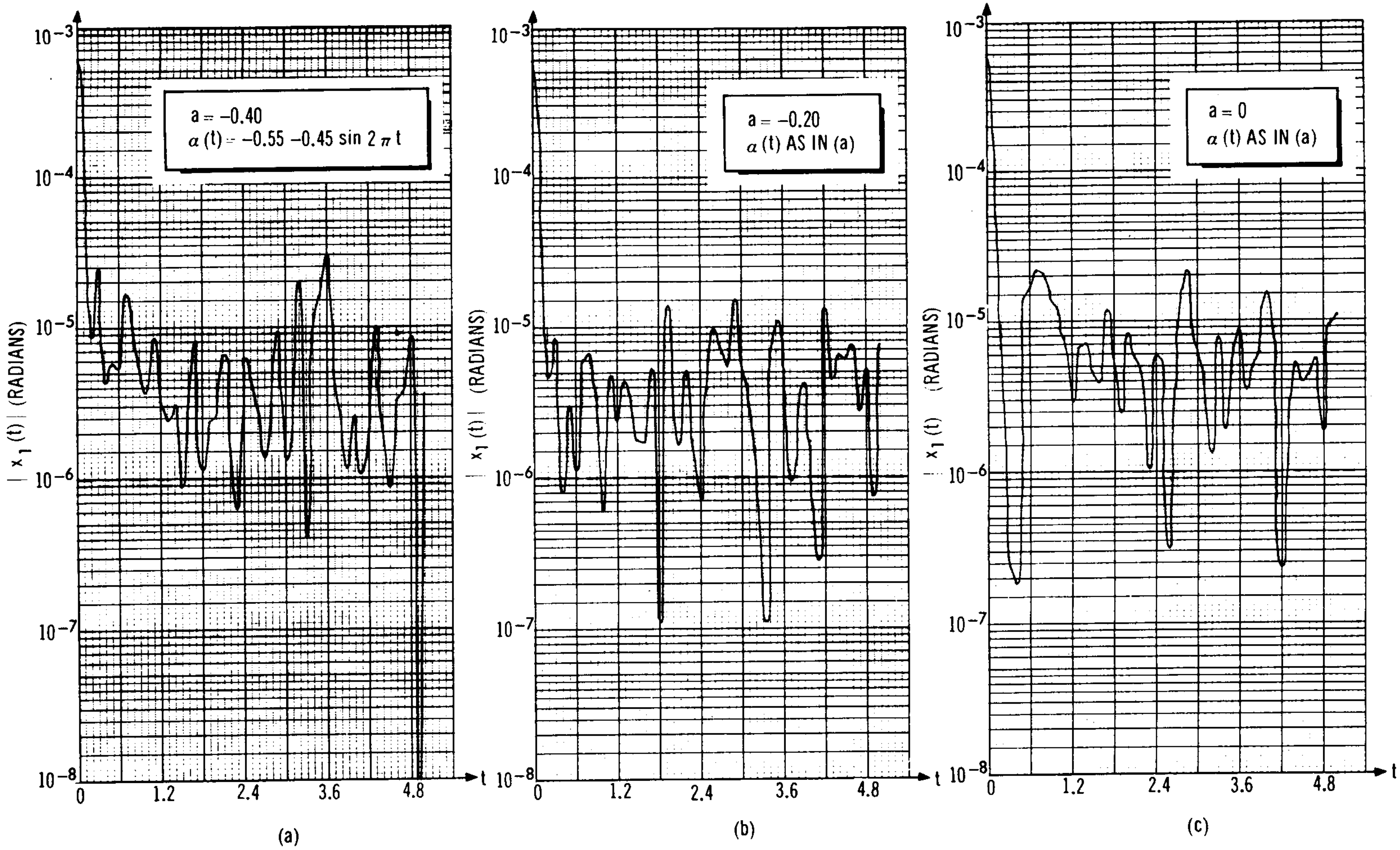


Figure 34. $|x_1(t)|$ for 5 Values a, with Parameter Mismatches (Experiment 10)

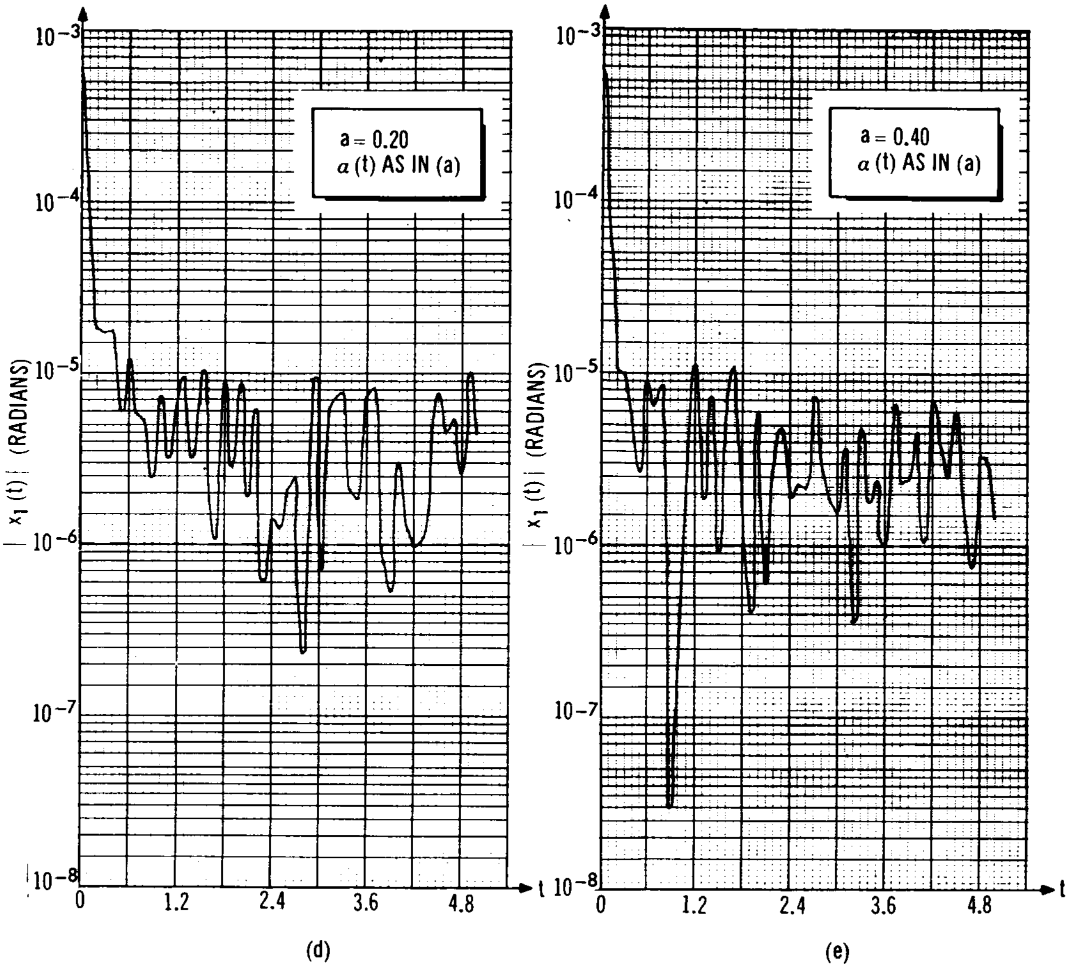


Figure 34. $|x_1(t)|$ for 5 Values a, with Parameter Mismatches (Experiment 10)

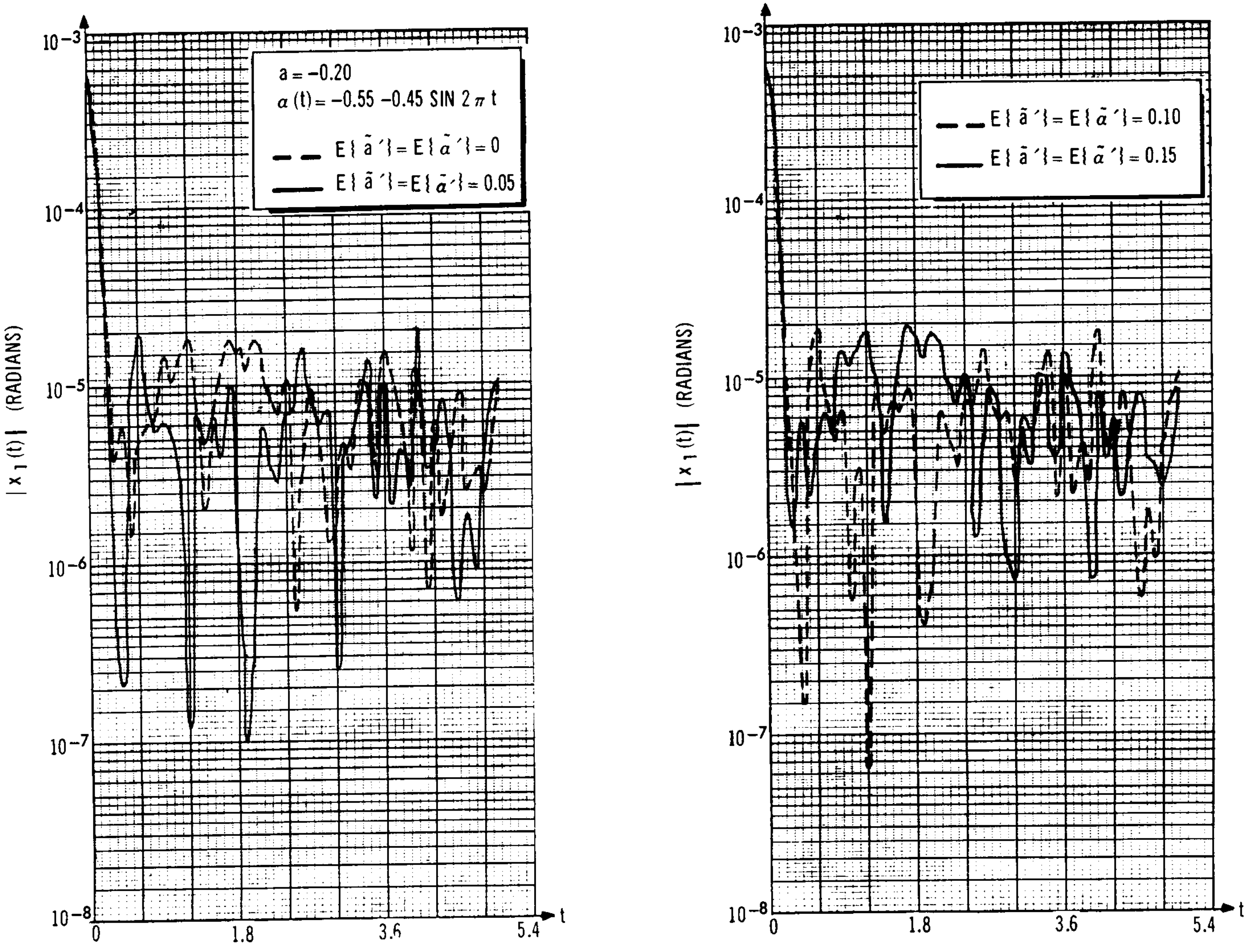


Figure 35a. | $x_1(t)$ | for Four Cases of Parameter Mismatch (Experiment12)

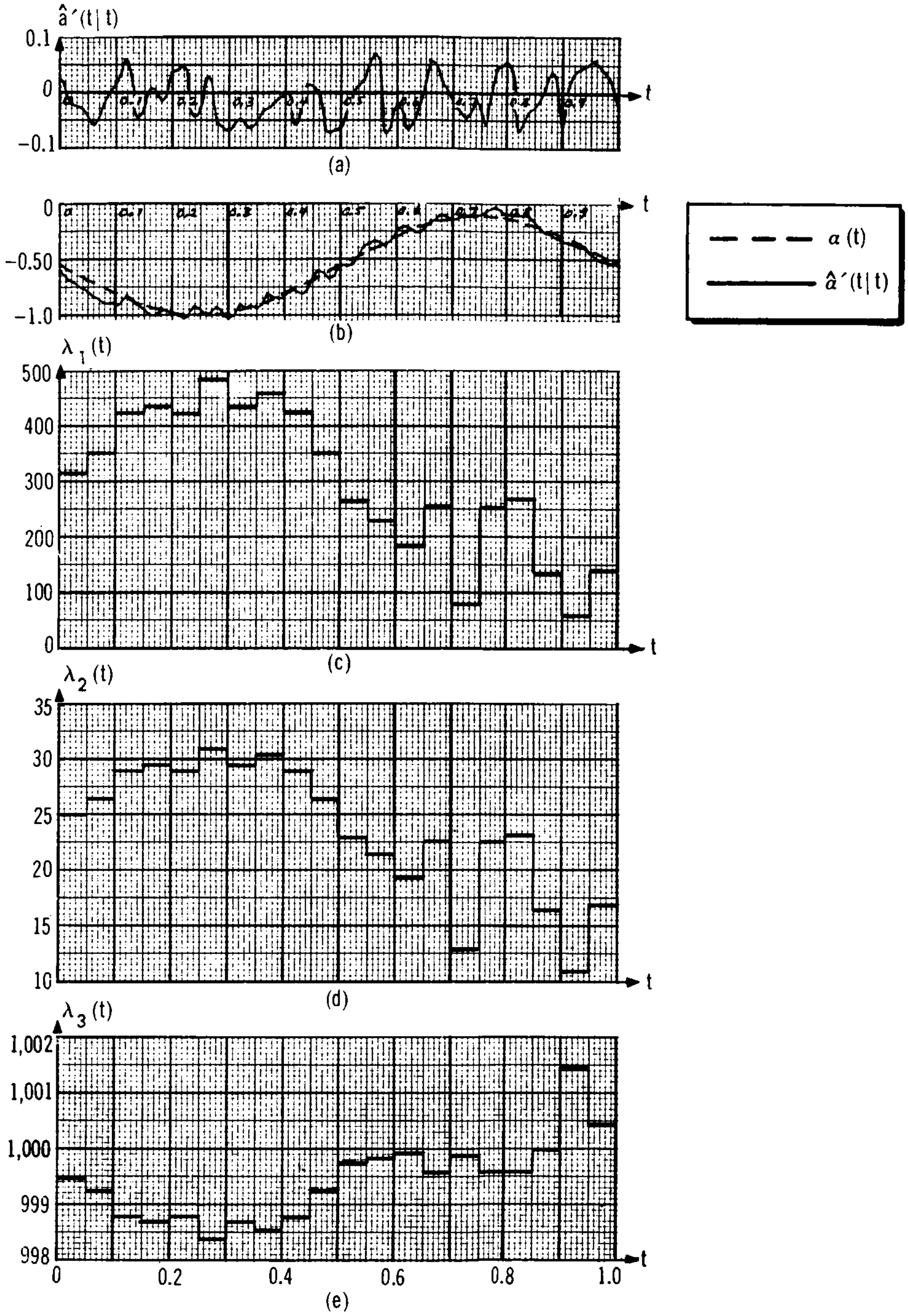


Figure 35b. Pseudo-Estimates of (a) a(t) and (b) α(t) [$\tilde{a}$ and $\tilde{\alpha}$ Uniformly Distributed] and Nominal Feedback Gains – a = 0, α(t) = –0.55 – 0.45 sin 2πt, and p_ℓ = 0.96

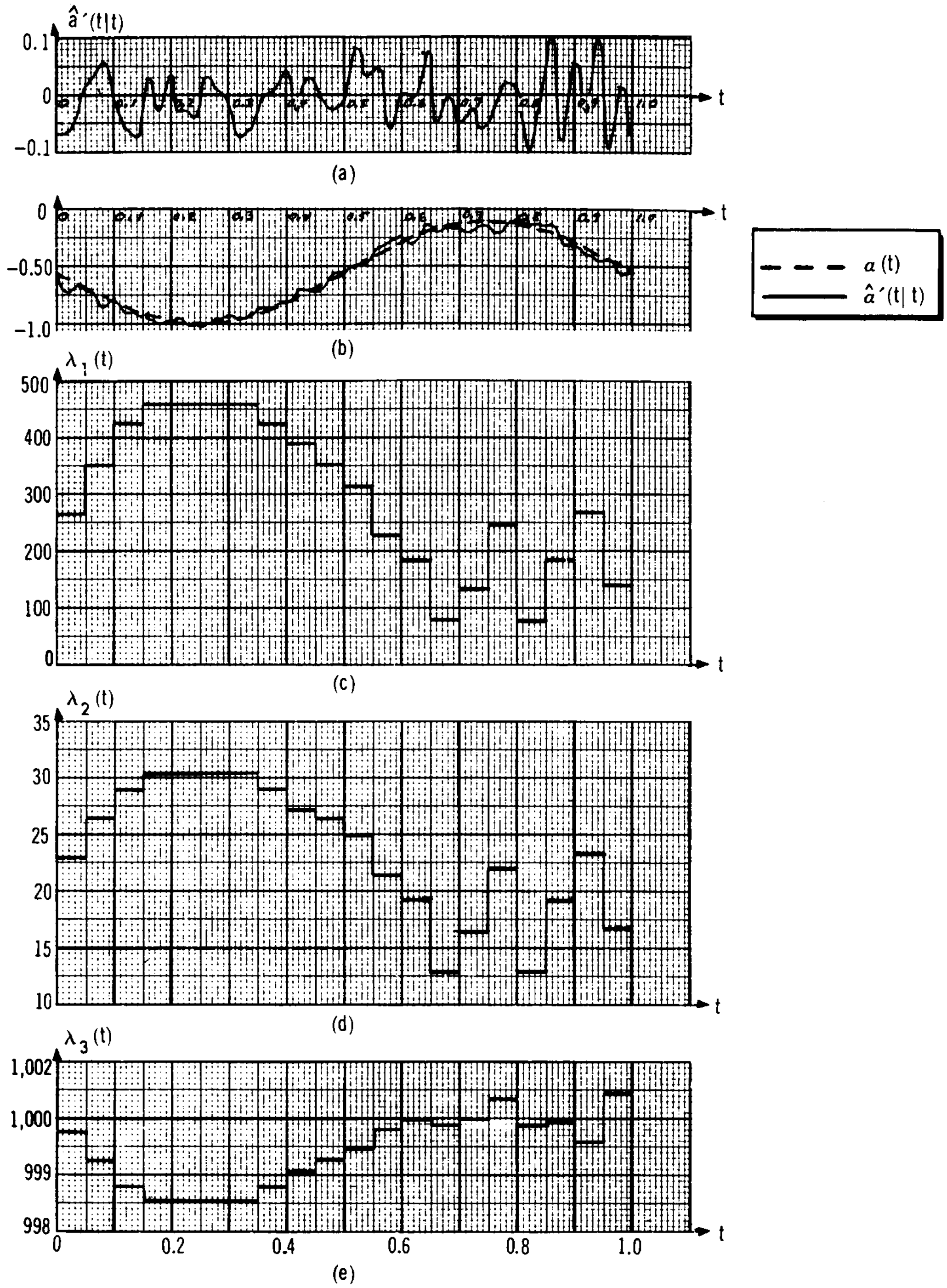


Figure 35c. Pseudo-Estimates of (a) a(t) and (b) $\alpha(t)$ [$\tilde{a}$ and $\tilde{\alpha}$ Normally Distributed] and Nominal Feedback Gains – $a = 0$, $\alpha(t) = -0.55 - 0.45 \sin 2\pi t$, and $p_\ell = 0.96$

Simulations involving the nominal controller, in which the loop was closed around the output of the state estimator, were performed. The simulations demonstrated that: (1) attitude errors are degraded by between one and two orders of magnitude due to state-estimation errors when disturbance torques are of the order of magnitude of 10^{-4} lb -ft ; (2) if disturbance torques are only of the order of magnitude of 10^{-6}lb-ft, attitude errors will remain less than ±0 20 arcsec, even though estimation errors occur (provided measurement-noise level is small); (3) large measurement-noise levels degrade attitude errors further; and (4) attitude errors are not degraded further when parameter mismatch (uniformly or normally distributed) occurs.

State-estimation errors can be compensated for during the design of the nominal controller, since their presence and effect (on attitude error) can now be anticipated, off-line. For example, since attitude errors are degraded by between one and two orders of magnitude, ϵ in Appendix C could be chosen [specifically in eqs. (C22), (C23) and (C26)] to be between one and two orders of magnitude smaller than 0. 20 arcsec, its present value. C_1 would then be larger; hence, the weighting factor associated with a nominal control situation, ρ, would also be larger (in Section 2, $\underset{\sim}{\rho} = \underset{\sim}{C}_1$ for most of the nominal control situations). This means that the entries in table III would be modified; a new adaptive nominal controller results. This design modification remains to be verified.

Section 4

AN ON-LINE LEARNING PROCEDURE

Introduction

It has been shown in the preceding sections that, if disturbance torques exceed their a priori design amplitudes attitude errors may exceed ±0.20 arcsec. The overall objective is to achieve fine attitude control (see Appendix A); hence, it may be necessary to modify the nominal controller designed in Section 2 if that controller cannot contain attitude errors to within ±0.20 arcsec. The system depicted in fig. 3 has been proposed for updating the nominal controller, when and if modifications are necessary.

The present section is concerned with the following aspects of the on-line-learning control system in fig. 3: 1) on-line goal circuit, 2) on-line learning algorithm, and 3) simulations of the overall system.

The on-line goal circuit processes measured data in the form of on-line cost functions. The next subsection distinguishes between an overall performance index and a sub-goal performance index. The overall performance index is related directly to the statement that attitude errors must be kept to values less than or equal to ±0.20 arcsec. The sub-goal performance index is related to the overall performance index through an average and is used in the on-line learning algorithm.

The on-line learning algorithm is heuristic and error-correcting. Error-correcting is used to mean that decisions to update the feedback gains are made only if the present performance is worse than the preceding performance. Heuristic is used to mean that no convergence theorem is provided for the algorithm. Control situations are utilized in the on-line learning procedure. A distinction between off-line and on-line control situations is also made in this section.

Three experiments involving the on-line-learning control system are described in the subsection titled Simulation Studies. Their objectives are to demonstrate the error-correction algorithm derived in the subsection titled Error Correction Learning Procedure and to determine some of the design parameters which enter into the algorithm. These experiments were performed at the very end of the study and due to insufficient time are incomplete.

On-Line Cost Functions

The overall objective is, as pointed out above, to contain attitude errors to within ±0. 20 arcsec. In connection with the laser communication satellite application, it was not specified by NASA whether attitude errors must be contained at every instant of time or over intervals of time, in an absolute sense or in some probabilistic sense. Due to the stochastic nature of the attitude errors, it was felt that it is more meaningful to look at the behavior of attitude errors over an interval of time (T) rather than at discrete values of t; hence, it was assumed that the overall objective is to contain attitude errors in an absolute sense over fixed intervals of time.

In Section 1 a Basic Premise was stated in connection with the necessity for on-line learning. It is, that for on-line learning to improve a system's performance on-line, new information must be made available to an on-line performance assessor. In this study it was assumed that the new information is contained in the measured attitude error, $z_1(t)$; for, truly, the effects of the actual disturbance torques are observable in $z_1(t)$.

It is desirable to weight positive and negative attitude errors alike; hence, in the cost functions discussed below, the squared measured attitude error, $z_1^2(t)$, is used, since it is the simplest function of the measured attitude error which is never negative and thus does not have the disadvantage of having positive attitude error contributions cancelled by negative attitude error contributions.

The overall goal or performance index is denoted $\overline{PI}$ (k,ℓ, m). It provides the cost for having to iterate the control $u^*(t)$ in fig. 3 the m^th time the ℓth control situation is re-entered. It is assumed, first, that it is not very likely that the iterations of $u^*(t)$ will be completed the first time the ℓth control situation is entered and, second, that the maximum number of iterations associated with the first, second, ..., m^th, ..., etc. entries into the ℓth control situation is k_1, k_2, ..., k_m, ..., etc., respectively. The expression for $\overline{PI}$ (k, ℓ; m) is

$$\overline{PI}\ (k,\ell;m) = \int_{kT}^{(k+1)T} z_1^2(k,\ell;t)dt \tag{58}$$

where the special notation used for z_1 is control situation oriented, and is elaborated upon below.

In keeping with the overall objective of containing attitude errors in an absolute sense over fixed intervals of time, the following theorem is proved for the case when there is zero measurement noise. The more general results are discussed in the final paragraph of this section.

Theorem

If $n_1(t) = 0$ and $\bar{PI}(k, \ell;m) \leq \delta(\epsilon)$, where $\delta(\epsilon) = \epsilon^2 T$, then $|x_1(k, \ell;t)| \leq \epsilon$ for $kT < t < (k+1)T$ and $k = 0, 1, \ldots$.

Proof: When $n_1(t) = 0$ $z_1(k, \ell;t) = x_1(k, \ell;t)$ and

$$\overline{PI}(k, \ell;m) = \int_{kT}^{(k+1)T} x_1^2(k, \ell;t)dt \tag{59}$$

It is assumed that x_1 is bounded during each interval of time; that is,

$$|x_1(k, \ell;t)| \leq x_1(k) \quad \text{for } kT < t < (k+1)T \tag{60}$$

The above assumption is felt to be reasonable for the application. It follows, that

$$\bar{PI}(k, \ell;m) \leq x_1^2(k)T \quad \text{for } kT < t < (k+1)T \tag{61}$$

In eq. (61), if $x_1(k) = \epsilon$, such that $\delta(\epsilon) \triangleq \epsilon^2 T$, then in eq. (60)

$$|x_1(k, \ell;t)| \leq \epsilon \quad \text{for } kT < t < (k+1)T \tag{62}$$

which completes the proof.

Due to large initial attitude errors, which may result from a poor initial choice of u*(t), that is to say, a poor choice for $\underset{\sim}{P}^{\ell}$ (the nominal feedback parameter vector associated with the ℓth control situation), it is probably more useful to weight more recent observations of the performance index $\bar{PI}$ more heavily than older observations. To this end, the sub-goal performance index PI(k,ℓ;m), defined below, is used in the decision making procedure for updating the on-line control gains, and not the overall goal. As is shown, however, the control gains are modified only when the overall goal is violated.

The sub-goal performance index is given recursively as

$$PI\ (k+1,\ell;m) = \gamma PI\ (k,\ell;m) + \frac{(1-\gamma)}{T}\ \overline{PI}\ (k+1,\ell;m) \quad k = 0, 1 \ldots \qquad (63)$$

where

$$PI\ (0,\ell;m) = \frac{1}{T}\ \overline{PI}\ (0,\ell;m) \qquad (64)$$

and

$$0 \le \gamma \le 1. \qquad (65)$$

From eqs (63) and (64), it is straightforward to show that

$$PI\ (k,\ell;m) = \frac{\gamma^k}{T}\ \overline{PI}\ (0,\ell;m) + \frac{(1-\gamma)}{T} \sum_{j=1}^{k} \gamma^{k-j}\ \overline{PI}\ (j,\ell;m) \qquad (66)$$

where $1 \le k \le k_m$

It must be observed, that two observations of $\overline{PI}$ $(j,\ell;m)$ made at j=q and j=q+s are weighted, over any iteration interval $kT < t < (k+1)T$, in the following proportion:

$$\frac{(1-\gamma)}{T}\gamma^{k-s-q} \Big/ \frac{(1-\gamma)}{T}\ \gamma^{k-q} = \gamma^{-s} \ .$$

By suitable choices for γ , the appropriate emphasis may be placed on recent and past observations. The name linear reinforcement averaging is associated with the averaging process in eq. (66), when $0 < \gamma < 1$. When $\gamma = 0$ PI $(k+1,\ell;m)$ is proportional to $\overline{PI}$ $(k+1,\ell;m)$, and, when $\gamma = 1$ PI $(k+1,\ell;m)$ is proportional only to the initial cost $\overline{PI}$ $(0,\ell;m)$. In addition, if $\gamma = 1 - \frac{1}{(k+2)}$ then eq. (66) reduces to a standard averaging process. The term linear reinforcement averaging is borrowed from the so-called linear reinforcement learning model which is used in learning theory (ref. 2, for example). The linear reinforcement learning model is of the same form as in eq. (66). Discussions on suitable choices for γ are continued in the subsection titled "Simulation Studies."

Error-Correction Learning Procedure

On-Line-Optimal Feedback Gains. --In order to distinguish between the nominal controller designed in Section 2 and the learning controller designed in this section, the following notation and definitions must be introduced:

1. $u^{*}(t)$ is the on-line-optimal control, and is initially equal to the nominal control, $u_{2}^{*}(t)$, designed in Section 2.

$$u^{*}(t) = \Lambda^{*}(t)\hat{\underline{x}}_{a}(t|t) \tag{67}$$

2. $\Lambda^{*}(t)$, the on-line-optimal feedback gain matrix, is a function of which control situation the system is in; for the ℓth control situation,

$$\Lambda^{*}\backslash t) \rightarrow \Lambda^{*\ell}(t) . \tag{68}$$

3. It is more convenient to work with the on-line-optimal feedback parameter vector $\underline{P}^{*\ell}_{\Lambda}(t)$ than with $\Lambda^{*\ell}(t)$; however,

$$\left(\underline{P}^{*\ell}_{\Lambda}(t)\right)^{T} = \Lambda^{*\ell}(t) \tag{69}$$

from which it follows, that for the ℓ^{th} control situation,

$$u^{*}(t) = \left(\underline{P}^{*\ell}_{\Lambda}(t)\right)^{T} \hat{\underline{x}}_{a}(t|t) . \tag{70}$$

4. In the on-line-learning procedure, discussed shortly, $\underline{P}^{*\ell}_{\Lambda}(t)$ is obtained iteratively. It is incremented once every T units of time. The k^{th} iterate of $\underline{P}^{*\ell}_{\Lambda}(t)$, $\underline{P}^{*\ell}_{\Lambda}(k;t)$, is given as

$$\underline{P}^{*\ell}_{\Lambda}(k;t) = \sum_{i=0}^{k} \underline{\pi}(i)U_{-1}(t-i\,T) \tag{71}$$

and is a piecewise continuous function as illustrated in fig. 36 for one of the components of $\underline{P}^{*\ell}_{\Lambda}(k;t)$.

5. Since the on-line-optimal control is initially equal to the nominal control,

$$\underline{\pi}(0) = \underline{P}^{\ell}_{\Lambda} \quad , \tag{72}$$

where $\underline{P}^{\ell}_{\Lambda}$ denotes the nominal gains, designed in Section 2, for the ℓth control situation.

In connection with these definitions, the following observations are made. First, note that time variations in $\Lambda^{*}(t)$ occur in two distinct modes. $\Lambda^{*}(t)$ varies according to eq. (71) for as long as the system remains in the ℓth control situation. Here the variations are due to the on-line optimizations which are performed individually for each control situation. On the other

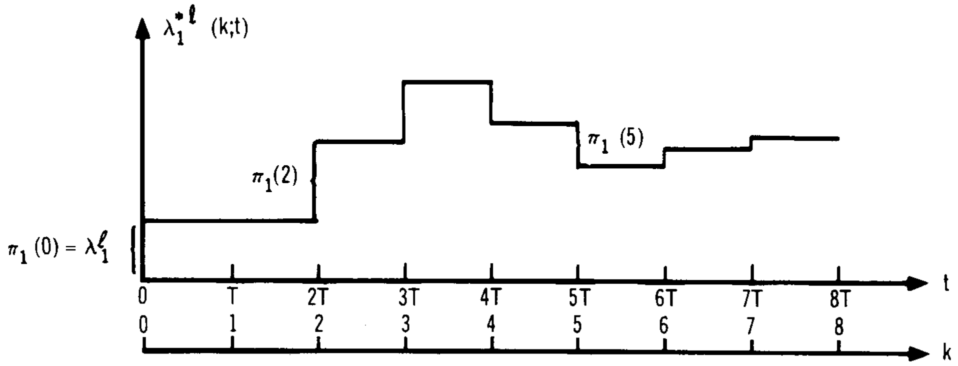


Figure 36. Variations in One Element of $\underline{P}_\Lambda^{*\ell}(k;t)$

hand, $\Lambda^*(t)$ also varies as the system moves from one control situation into another. This movement is reflected as variations in the index ℓ. The second type of variation in the on-line-optimal feedback gain matrix is identical to that observed for the nominal gains (figs. 19, 20, and 21, for example). The first type of variation, on the other hand, represents a transient phenomenon which is associated with each control situation. It persists until satisfactory performance is achieved for all points within the control situation, at which time $\underline{P}^{*\ell}_{\Lambda}(k;t)$ is constant and remains constant for all time thereafter, unless the performance of the system becomes unsatisfactory once again.

Note, also, that since Λ^* is a function of ℓ, k, and t, $\underline{x}_a$ is also a function of these three parameters; hence, the measured attitude error, $z_1(t)$, which is used in eq. (58), is also a function of ℓ, k, and t; that explains the notation used in eq. (58). This functional dependence of the states on ℓ and k is used only for the purposes of clarification; otherwise, the dependence is assumed.

On-Line-Learning Algorithm. --The on-line-learning procedure is embodied in an algorithm for updating $P^{*\ell}_{\Lambda}(k;t)$ in eq. (71). The algorithm relates the sub-goal performance index, discussed previously, to decisions as to when and if $P^{*\ell}_{\Lambda}(k;t)$ should be changed (updated). These changes occur only if $|x_1(t)|>0.20$ arcsec; hence, the learning algorithm is referred to as an error-correction learning algorithm. The algorithm is described below for the ℓ^{th} control situation.

The first time the ℓ^{th} control situation is entered, m in eq. (66) is unity, and the expression for the sub-goal performance index is

$$PI(k,\ell;1) = \frac{\gamma^k}{T}\int_0^T z_1^2(0,\ell;t)dt + \frac{(1-\gamma)}{T}\sum_{j=1}^{k}\gamma^{k-j}\int_{jT}^{(j+1)T} z_1^2(j,\ell;t)dt$$

$$k = 0, 1, \ldots, k_1 \tag{73}$$

The system responds for the first T units of time under the action of the nominal controller [eq. (72)]. The value of $PI(0,\ell;1)$ is computed; however, it serves merely as a reference value for the sub-goal performance index during the next iteration interval. In the present error-correction algorithm, there is no basis for a change in the feedback gains during the interval $T<t<2T$.

For $T<t<2T$, $\underline{P}^{*\ell}_{\Lambda}(k;t) = \underline{P}^{*\ell}_{\Lambda}(1;t) \triangleq \underline{P}^{*\ell}_{\Lambda}(0;t)$; hence,

$$\underline{\pi}(1) = \underline{0} \tag{74}$$

The above ensures that the system operates under the action of the nominal controller for the second T units of time as well as for the first T units of time.

The value of $PI(1,\ell;1)$ is computed and compared with $PI(0,\ell;1)$. If $PI(1,\ell;1) \leq PI(0,\ell;1) - \epsilon'(0)$, $\underline{\pi}(2)$ in eq. (71) is set equal to $\underline{0}$, which is indicative of a satisfactory choice for $\underline{P}^{*\ell}_{\Lambda}(1;t)$. On the other hand, if $PI(1,\ell;1) > PI(0,\ell;1) - \epsilon'(0)$, which is indicative of a degradation in performance, $\underline{\pi}(2)$ is chosen different from $\underline{0}$. The exact rule for choosing $\underline{\pi}(2)$ is deferred to a later discussion, so as not to break the continuity of the present discussion. $\epsilon'(0)$ is a threshold function which is a design parameter and is elaborated upon in the next subsection.

As k (and, subsequently, t) evolves, the on-line-optimal feedback parameter vector in eq. (71) evolves, until, for $k=k_1$

$$\underline{P}^{*\ell}_{\Lambda}(k_1;t) = \sum_{i=0}^{k_1} \underline{\pi}(i)\underline{U}_{-1}(t-iT) \tag{75}$$

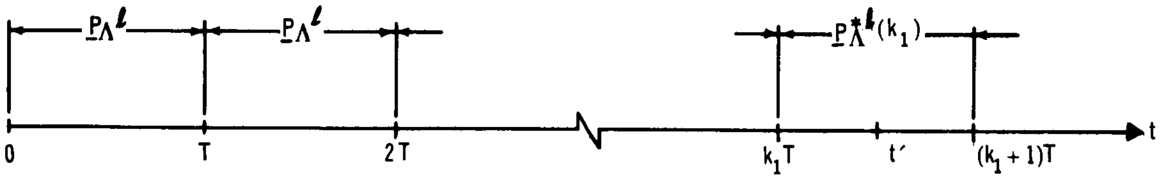
The system is assumed to leave the ℓ^{th} control situation at some instant of time t', where $k_1T < t' < (k_1+1)T$ (fig. 37), at which time the amplitude of the on-line-optimal feedback parameter vector is $\underline{P}^{*\ell}_{\Lambda}(k_1)$, where

$$\underline{P}^{*\ell}_{\Lambda}(k_1) = \sum_{i=0}^{k_1} \underline{\pi}(i) \tag{76}$$

This value is stored in a memory compartment which is associated with the ℓ^{th} control situation for use as the first value of $\underline{P}^{*\ell}_{\Lambda}(k;t)$ the second time the ℓ^{th} control situation is entered.

At $t=t'$ the measured attitude error is $z_1(k_1,\ell;t')$. The second time the ℓ^{th} control situation is entered the initial (with respect to the second entry) measured attitude error is not $z_1(k_1,\ell;t')$. This is due to variations in z_1 which occur while the system is in other control situations; hence, it is meaningless to continue eq. (73) out past k_1 or to base decisions on choices for $\pi(i)$, the second time the ℓ^{th} control situation is entered, on $PI(k_1,\ell;1)$. [Hill (Ref. 3), in a different context of course, carries over his performance index from one entry into the ℓ^{th} control situation to another. As pointed out by Mr. Frank Noonan (NASA), there is no guarantee that Hill's procedure for updating his feedback gains will converge, due to the discontinuities which occur from one entry to another.]

The second time the system enters into the ℓ^{th} control siutation m=2 in eq. (66), and



$\underline{P}_\Lambda^{\ell}$
$\underline{P}_\Lambda^{\ell}$
$\underline{P}_\Lambda^{*\ell}(k_1)$
0
T
2T
k_1T
t′
$(k_1+1)T$
t

Figure 37. Amplitudes of On-Line-Optimal Feedback Parameter Vector During First Entry into ℓth Control Situation

$$\mathrm{PI}\,(k,\ell;2) = \frac{\gamma^k}{T}\int_0^T z_1^2(0,\ell;t)dt + \frac{(1-\gamma)}{T}\sum_{j=1}^{k}\gamma^{k-j}\int_{jT}^{(j+1)T} z_1^2(j,\ell;t)dt \qquad k = 0, 1, \ldots, k_2 \tag{77}$$

It is assumed that whatever the instant of time the ℓ^{th}control situation is re-entered is labeled as t=0. Iterations then proceed from 0 to T, T to 2T, . . . , and k_2T to t'', where t'' is the time at which the system again leaves the ℓ^{th} control situation $\left[k_2T < t'' < (k_2+1)T\right]$.

For $0 < t < T$, $\underline{P}_\Lambda^{*\ell}(k;t) = \underline{P}_\Lambda^{*\ell}(0;t) = \underline{P}_\Lambda^{*\ell}(k_1)$ which, of course, is available from the preceding entry. This is in contrast to the value chosen for $\underline{P}_\Lambda^{*\ell}(0;t)$ the first time entry was made into the ℓ^{th} control situation, which was the nominal control $\underline{P}_\Lambda^{\ell}$. It is at this point that the learning controller is utilizing the past experience of the system in order to achieve on-line learning. The algorithm for updating $\underline{P}_\Lambda^{*\ell}(k;t)$ on the second entry is identical to the algorithm discussed above for updating $\underline{P}_\Lambda^{*\ell}(k;t)$ on the first entry. It is worth mentioning again, that once again there is no basis for a new decision during the interval of time $T < t < 2T$; hence, $\underline{P}_\Lambda^{*\ell}(1;t) = \underline{P}_\Lambda^{*\ell}(0;t) = \underline{P}_\Lambda^{*\ell}(k_1)$.

For purposes of exposition, it is assumed that eventually the system will enter and remain in the ℓ^{th} control situation long enough for the attitude error to be driven to values less than or equal to ±0. 20 arcsec, as required. It is assumed, for example, that this is achieved during the p^{th} entry, for $k=k_p$. The amplitude of the on-line-optimal feedback parameter vector is, then,

$$\underline{P}_{\Lambda}^{*\ell}(k_p) = \sum_{i=0}^{k_p} \underline{\pi}(i) \tag{78}$$

where

$$\underline{\pi}(0) = \underline{P}_{\Lambda}^{*\ell}(k_{p-1}) \tag{79a}$$

and

$$\underline{\pi}(1) = \underline{0}. \tag{79b}$$

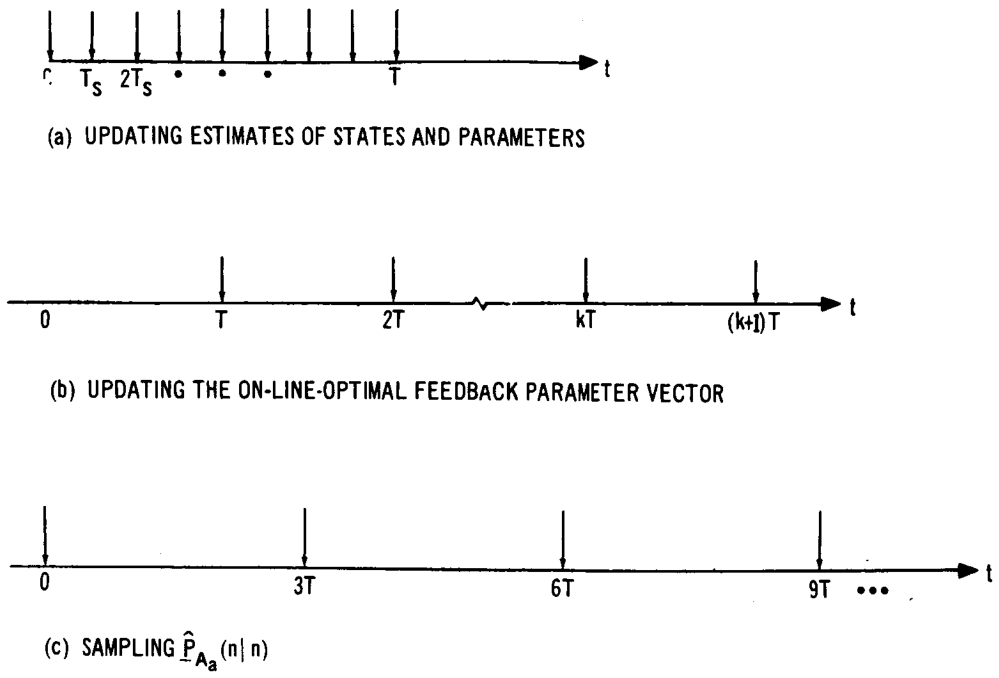
The feedback gain in eq. (78) is associated with the ℓ^{th} control situation; henceforth, whenever the ℓ^{th} control situation is entered $\underline{P}_{\Lambda}^{*\ell}$ (k_p) is applied and no further iterations are required unless, of course, $\overline{PI}$, which is always computed and monitored, exceeds $\delta(\epsilon)$. If $\overline{PI} > \delta(\epsilon)$ attitude errors are excessive and the on-line-optimal feedback parameter vector is once again updated until attitude errors are driven to values less than or equal to ±0. 20 arcsec.

The Error-Correction Learning Algorithm is summarized in fig. 38. Note that every 3T units of time the on-line optimizations are preceded by a control situation check. The factor of 3, while quite arbitrary, does allow at least two decisions to be made about the quality of the on-line-optimal feedback parameter vector, per entry into a control situation.

In the next subsection discussions are presented on a number of topics directly related to the Error-Correction Learning Algorithm.

Topics Related to the Error-Correction Learning Algorithm. --This subsection begins with a brief summary of the timing problem, that is the problem of when states and parameter estimates are updated, when these estimates are sampled for use in the on-line-learning procedure and when the control gains are updated during the on-line-learning phase. A rule for updating the control gain-change vector $\underline{\pi}(k+1)$ and the associated effects on the learning algorithm are discussed in the subsection titled "Updating the Control Gains." The time-varying threshold, $\epsilon'(k)$, is discussed in the subsection titled "Threshold Functions." Finally, the possibility of and reasons for distinguishing between on-line and off-line control situations are discussed in the subsection titled "Control Situations."

The Timing Problem: The relationships between the times at which estimates of states and parameters are updated (using the estimator in fig. B-2), on-line-optimal control gains are updated, and samples of the estimated plant parameters are taken are depicted in fig. 39. The estimates of states and



(a) UPDATING ESTIMATES OF STATES AND PARAMETERS

(b) UPDATING THE ON-LINE-OPTIMAL FEEDBACK PARAMETER VECTOR

(c) SAMPLING $\hat{\underline{P}}_{A_a}(n|n)$

Figure 39. Timing Relationships

parameters are updated every T_s units of time, where T_s is some fractional multiple of T, the decision interval. The estimated plant parameters are sampled every 3T.

The decision interval should be chosen small enough, that

$$\|\hat{\underline{P}}_{A_a}\left[(n+3)T|(n+3)T\right] - \hat{\underline{P}}_{A_a}(nT|nT)\| \le M \qquad (80)$$

for n=0, 3, 6. . . , where M is related to the size of a square control situation (fig. 8b). This means that the on-line optimization interval, 3T, should not be sc large that on-line optimizations are assumed to be in progress for one control situation when in reality they are in progress for another (due to the system moving out of the first control situation, during 3T, by an appreciable amount). Eq. (80) provides a means for bounding the decision interval, from abov

Updating the Control Gains: Numerous possibilities exist for updating the on-line-optimal feedback parameter vector. The procedure described below is intimately connected with the error-correction technique. Two other procedures are described in Appendices E and F. Both of these procedures result in reinforcement-type learning algorithms in that the feedback gains are updated not only when the overall performance deteriorates but also when performance improves. The stochastic automata approach, described in

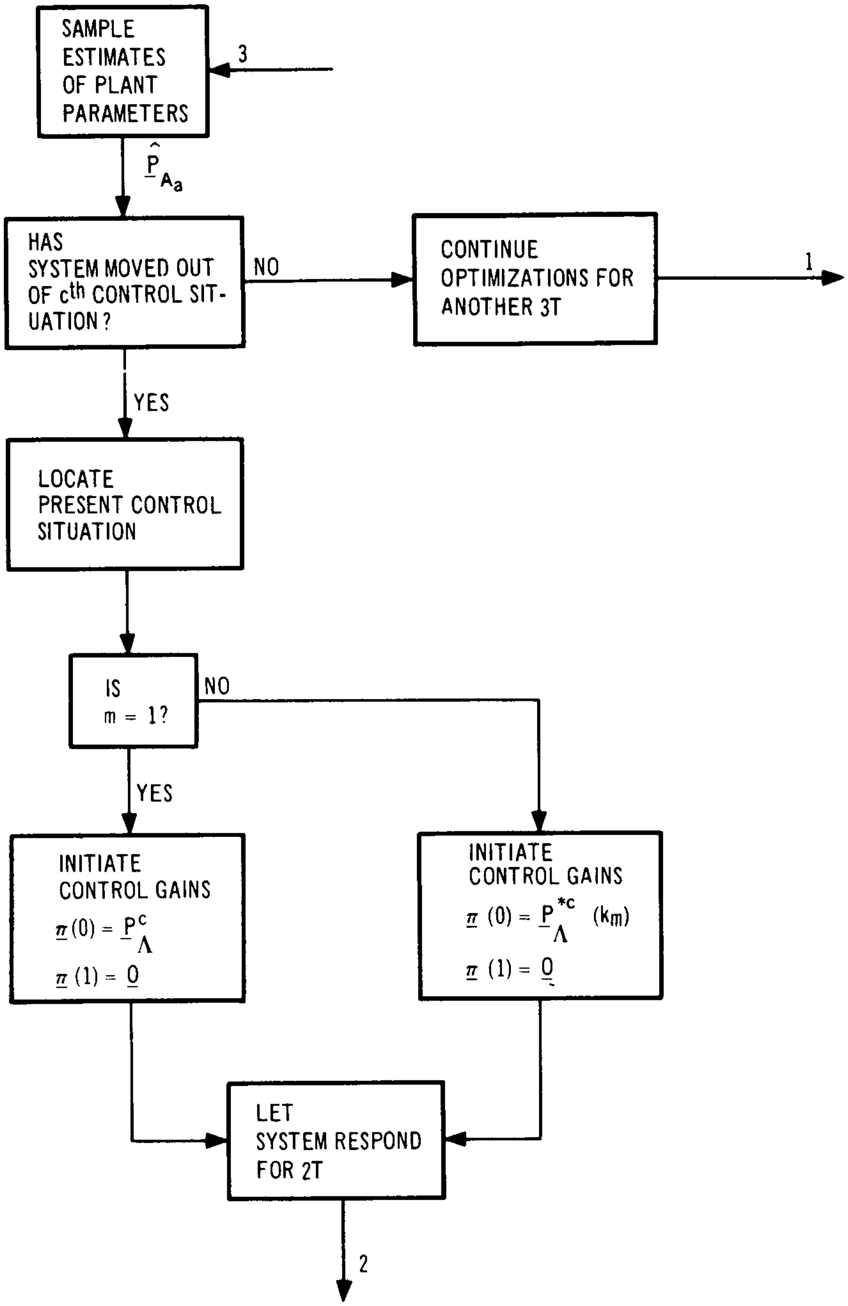

SAMPLE ESTIMATES OF PLANT PARAMETERS
3
$\hat{\underline{P}}_{A_a}$
HAS SYSTEM MOVED OUT OF c^{th} CONTROL SITUATION?
NO
CONTINUE OPTIMIZATIONS FOR ANOTHER 3T
1
YES
LOCATE PRESENT CONTROL SITUATION
IS m = 1?
NO
YES
INITIATE CONTROL GAINS
$\underline{\pi}(0) = \underline{P}^{c}_{\Lambda}$
$\underline{\pi}(1) = \underline{0}$
INITIATE CONTROL GAINS
$\underline{\pi}(0) = \underline{P}^{*c}_{\Lambda}(km)$
$\underline{\pi}(1) = \underline{0}$
LET SYSTEM RESPOND FOR 2T
2

Figure 38. Flow Diagram for Error-Correction Learning Algorithm

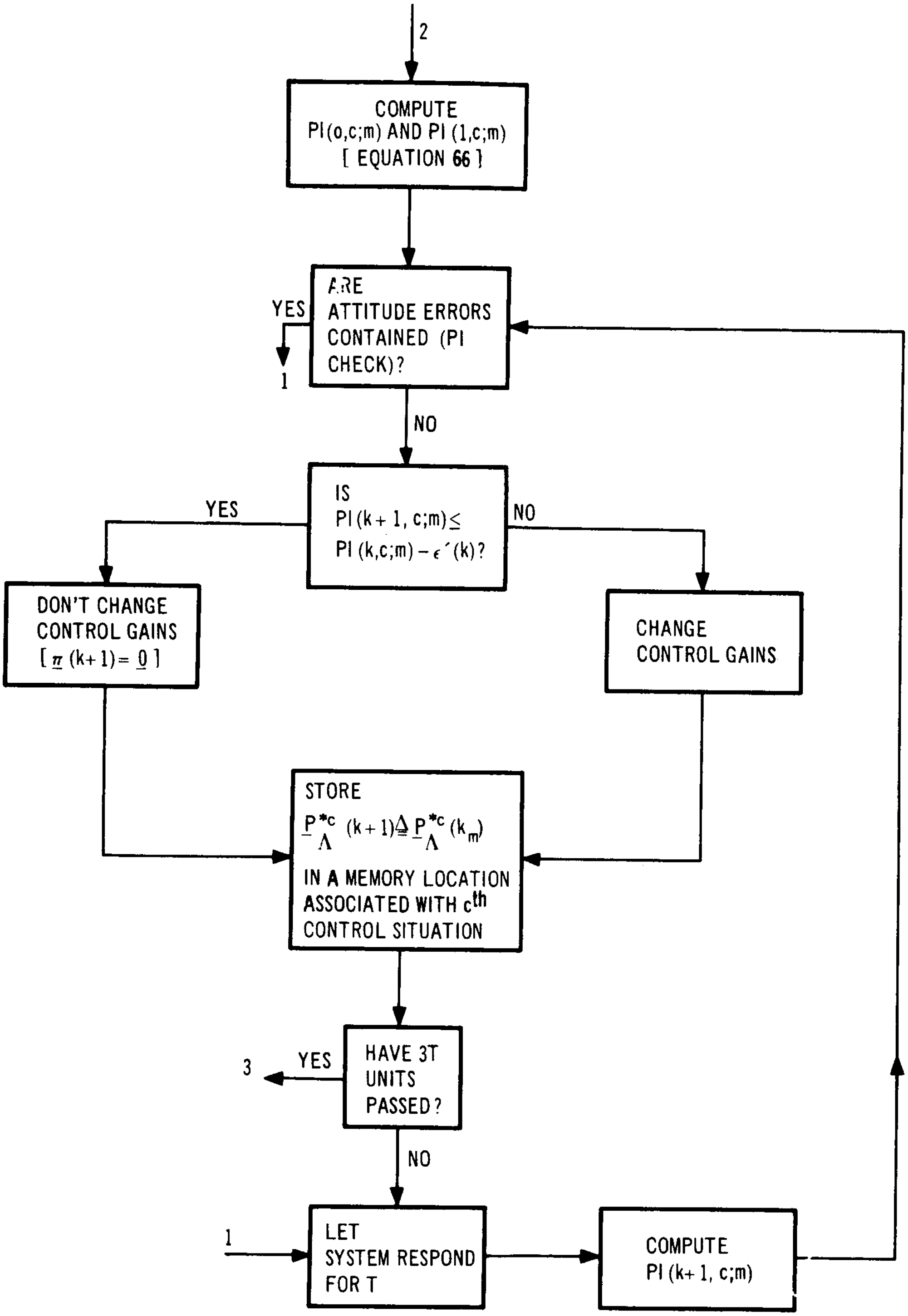


Figure 38. (Continued)

Appendix E, while interesting, does not appear to be applicable to this problem. The adaptive, random-optimization approach, described in Appendix F, may be applicable; however, time did not permit the algorithm described in that appendix to be simulated.

The rest of this subsection is concerned with a specific rule for updating the control gain-change vector $\underline{\pi}(k+1)$ in eq. (71), since

$$\underline{P}_{\Lambda}^{*\ell}(k+1) = \underline{P}_{\Lambda}^{*\ell}(k) + \underline{\pi}(k+1) \qquad k=1,2,\dots, \tag{81}$$

which follows from eq. (71). It is not necessary to obtain $\underline{\pi}(0)$ and $\underline{\pi}(1)$, since they are both known a priori, as discussed previously in this section [eqs. (79a) and (79b)].

Here $\underline{\pi}(k+1)$ is updated according to the following rule:

$$\underline{\pi}(k+1) = \left[\frac{y(k)-1}{f(T)}\right]\underline{\beta}(k) \qquad k=1,2,\dots \tag{82}$$

y(k) is a <u>decision function</u> which takes on the values of zero or unity according to the following rule:

$$y(k) = \begin{cases} 0 \text{ if } PI(k,\ell;m) > PI(k-1,\ell;m) - \epsilon'(k) \\ \\ 1 \text{ if } PI(k,\ell;m) \le PI(k-1,\ell;m) - \epsilon'(k) \end{cases} \tag{83}$$

for $k=1,2,\dots$.

f(T) is a <u>weighting function,</u> defined as

$$f(T) = 1 + [\text{Number of successive intervals for which } y(k) = 1] \tag{84}$$

$\underline{\beta}(k)$ is a vector which is proportional to $\underline{P}_{\Lambda}^{*\ell}(k)$; that is,

$$\beta_i(k) = C_1 \lambda_i^{*\ell}(k) \operatorname{sgn}\left[\frac{PI(k,\ell;m)-\widetilde{PI}}{\lambda_i^{*\ell}(k)-\tilde{\lambda}_i}\right] \qquad i=1,2,3 \tag{85}$$

where:

C_1 is a proportionality constant which is a design parameter,*

$\tilde{\lambda}_i$ is the most recent value of $\lambda_i^{*\ell}$ that is different from $\lambda_i^{*\ell}(k)$, and

$\widetilde{PI}$ is the last measured value of PI for which $\lambda_i^{*\ell} = \tilde{\lambda}_i$ (i=1, 2, 3)

Some observations are in order at this point.

When y(k) is unity $\underline{\pi}(k+1) = \underline{0}$, as required in the Error-Correction Learning Algorithm. When this case occurs it is not necessary to compute f(T) and $\underline{\beta}$ (k); hence, these computations are bypassed when y(k) = 1.

f(T) is a weighting function that causes $\underline{\pi}$ (k+1) to be updated by a smaller amount the longer the overall performance of the system continues to improve. In the worst case, when a choice for $\underline{\pi}(k+1)$ does not lead to immediately improved performance, f(T) is unity.

$\underline{\beta}$ may be viewed as a pseudo-gradient. Directional information is provided by the argument of the signum function. The numerator of this argument is negative if the performance for the present control choice is better than the performance for the preceding control choice; it is positive if the performance for the present control choice is worse than the performance for the preceding choice. Magnitude information is provided by the product $C_1\lambda_i^{*\ell}(k)$ which is interpreted as a fractional change in $\lambda_i^{*\ell}(k)$. Choices for C_1 are discussed in the subsection titled "Experiment A".

The nature of the on-line optimization procedure is depicted in fig. 40. for one component of $\underline{P}_{\Lambda}^{*\ell}(k)$. In connection with $\tilde{\lambda}_i$ and $\widetilde{PI}$, the following assumptions are made:

1. $\tilde{\lambda}_i$ is initially equal to zero (i=1, 2, 3),

2. $\widetilde{PI}$ is initially equal to PI(0,ℓ;m) when a control situation is entered for the first time or is re-entered.

Observe, in fig. 40, that eq. (82) provides a capability for either increasing or decreasing the preceding control choice. Four changes in $\lambda_1^{*\ell}(k)$ are illustrated in that figure (indicated by numbers), one each at k = 2, 3, 9 and 10. In connection with these (and other) changes in a component of $\underline{P}_{\Lambda}^{*\ell}(k)$ a more detailed discussion of the rule for updating $\lambda_1^{*\ell}$ from the k^{th} to the $(k+1)^{st}$ iteration is now in order.

*In the most general case, C_1 would be different for each component of $\underline{\beta}$.

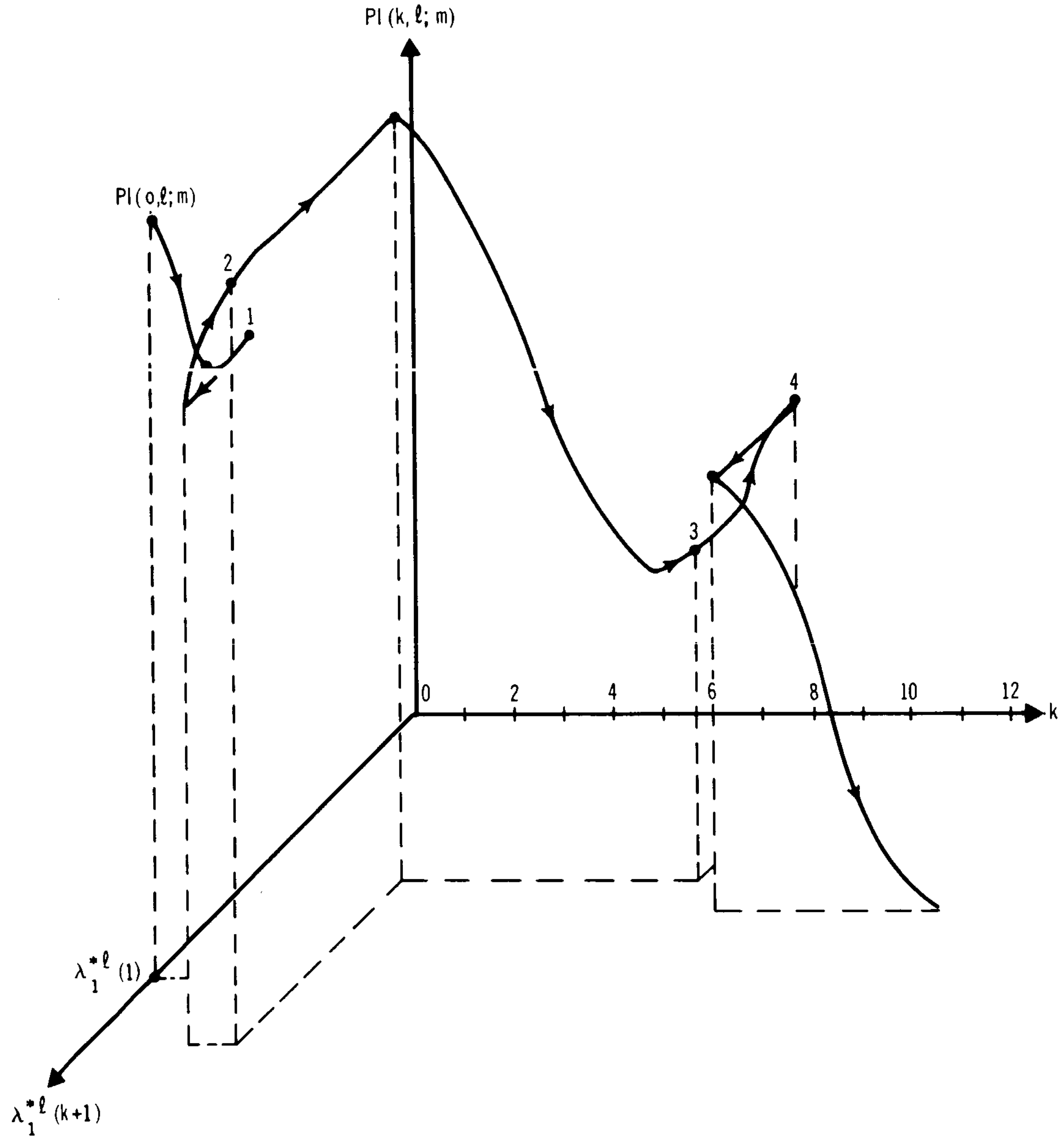


Figure 40. Optimization Trajectory

$\lambda_1^{*\ell}(k)$ changes onhy when $y(k) = 0$ in eq. (82), in which case it follows, from eqs. (81), (82), and (85), that

$$\lambda_1^{*\ell}(k+1) = \left\{1 - \frac{C_1}{f(T)} \operatorname{sgn}\left[\frac{PI(k,\ell;m) - \widetilde{PI}}{\lambda_1^{*\ell}(k) - \tilde{\lambda}_1}\right]\right\} \lambda_1^{*\ell}(k) \tag{86}$$

$$k=1, 2, \ldots .$$

Now, $y(k) = 0$ in either of the following ways:

$$\text{1.}\quad \begin{aligned} &PI(k,\ell;m) > PI(k-1,\ell;m) - \epsilon'(k) \text{ and} \\ &PI(k,\ell;m) < PI(k-1,\ell;m) \end{aligned} \tag{87}$$

or

$$\text{2.}\quad \begin{aligned} &PI(k,\ell;m) > PI(k-1,\ell;m) - \epsilon'(k) \text{ and} \\ &PI(k,\ell;m) > PI(k-1,\ell;m) \end{aligned} \tag{88}$$

In eq. (87), $PI(k,\ell;m)$ is going in the proper direction but hasn't gone far enough during one iteration. In eq. (88), on the other hand, $PI(k,\ell;m)$ is going in the wrong direction. In order to distinguish between the two modes in eqs. (87) and (88), let

$$PI(k,\ell;m) - \widetilde{PI} = g_i \tag{89}$$

where

$$i = 1 \text{ if } y(k) = 0 \text{ as in eq. (87)}$$

$$i = 2 \text{ if } y(k) = 0 \text{ as in eq. (88)}$$

From eqs. (87) and (88), it follows that

$$g_1 < 0 \text{ and } g_2 > 0 \tag{90}$$

In order to distinguish between the cases when $\lambda_1^{*\ell}(k) - \tilde{\lambda}_1$, in eq. (86), is positive or negative, let

$$\lambda_1^{*\ell}(k) - \tilde{\lambda}_1 = h_j \tag{91}$$

where

$$j = 1 \text{ if } \lambda_1^{*\ell}(k) > \tilde{\lambda}_1$$

$$j = 2 \text{ if } \lambda_1^{*\ell}(k) < \tilde{\lambda}_1$$

Upon substitution of eqs. (91) and (89) into eq. (86) the latter equation becomes

$$\lambda_1^{*\ell}(k+1) = \left[1 - \frac{C_1}{f(T)} \operatorname{sgn}\left(\frac{g_i}{h_j}\right)\right] \lambda_1^{*\ell}(k) \qquad (92)$$

$$k = 1, 2, \ldots .$$

The behavior of $\lambda_1^{*\ell}(k+1)$ is tabulated below for i, j = 1, 2. Table VII was obtained by analyzing eq. (92) for all combinations of i and j. It provides the starting point in our inquiry as to how $\lambda_1^{*\ell}$ is updated from one iteration to the next.

TABLE VII

SUMMARY OF BEHAVIOR OF $\lambda_1^{*\ell}(k+1)$

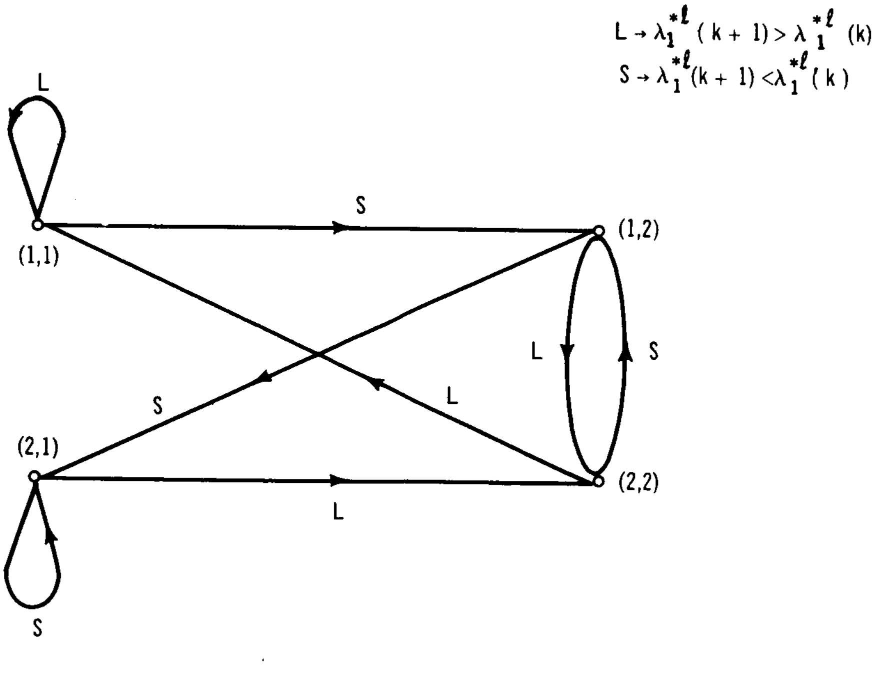
i	1	2
j	(PI improved, but not sufficiently)	(PI Worse)
1	$\lambda_1^{*\ell}(k+1)$ increases in the direction of the preceding value of $\lambda_1^{*\ell}$, $\lambda_1^{*\ell}(k)$, which tended to improve PI	$\lambda_1^{*\ell}(k+1)$ reverses direction and becomes smaller than $\lambda_1^{*\ell}(k)$
2	$\lambda_1^{*\ell}(k+1)$ decreases in the direction of the preceding value of $\lambda_1^{*\ell}$, $\lambda_1^{*\ell}(k)$, which tended to improve PI	$\lambda_1^{*\ell}(k+1)$ reverses direction and becomes larger than $\lambda_1^{*\ell}(k)$

The transition in $\lambda_1^{*\ell}$ from $\lambda_1^{*\ell}(k)$ to $\lambda_1^{*\ell}(k+1)$ may be conveniently viewed as the transition from one of four states in a finite state machine. The four states are (i, j) = (1, 1), (1, 2), (2, 1), and (2, 2). The possible transitions are depicted in fig. 41 which was obtained by analyzing Table VII for all combinations of (i, j).

From fig. 41, one observes that six different limit cycles may occur during the updating of $\lambda_1^{*\ell}$. The first is the self-loop at the node associated with the state (1, 1). In this first case $\lambda_1^{*\ell}$ continually increases. The second limit cycle appears as a self-loop at node (2, 1). In this second case $\lambda_1^{*\ell}$ continually decreases. The third limit cycle is associated with the transitions from state (1, 2) to (2, 2) to (1, 2) etc. In this third case $\lambda_1^{*\ell}$ oscillates in magnitude. The fourth limit cycle is associated with the transitions from state (1, 1) to (1, 2), to (2, 2), to (1, 1), etc. Here, $\lambda_1^{*\ell}$ decreases during the transition from (1, 1) to (1, 2) but increases during the transitions from (1, 2) to(2, 2), and (2, 2) to (1, 1). The fifth limit cycle is associated with the transitions from state (2, 1) to (2,2), to (1,2) to (2, 1), etc. Over this fifth limit cycle $\lambda_1^{*\ell}$ increases once and decreases twice. The sixth limit cycle is associated with the transitions from state (1, 1) to (1,2) to (2, 1) to (2,2) to (1, 1), etc. Observe, also, that it is impossible to have transitions from certain states to others [(1, 1) to (2, 1), for example] and that no self-loops are possible at states (1,2) and (2, 2).

The above limit cycles were observed during the simulations. The interesting point to be made here is that using the transition diagram concept all limit cycles can be predicted ahead of time.

Threshold function, $\epsilon'(k)$: In this paragraph the time-varying nature of the threshold function ϵ' in eq. (83) is discussed. First, it is demonstrated that ϵ' should be time-varying; hence, $\epsilon' \triangleq \epsilon'(k)$. Then, an explicit expression for $\epsilon'(k)$ is discussed.



$L \rightarrow \lambda_1^{*\ell}(k+1) > \lambda_1^{*\ell}(k)$
$S \rightarrow \lambda_1^{*\ell}(k+1) < \lambda_1^{*\ell}(k)$
L
S
(1,1)
(1,2)
L
S
S
L
(2,1)
(2,2)
L
S

Figure 41. Transition Diagram for $\lambda_1^{*\ell}(k)$ to $\lambda_1^{*\ell}(k+1)$, k = 0,1,...

The discussion below, in connection with the time-varying nature of ϵ', is presented in the context of a straight average* for $\overline{PI}$. It has been previously established that when $\gamma = 1 - 1/(k+2)$ the average in eq. (66) reduces to a straight average. In this case, it is straightforward to show that

$$PI(k+1,\ell;m) = \frac{(k+1)}{(k+2)} PI(k,\ell;m) + \frac{1}{(k+2)T} \overline{PI}(k+1,\ell;m) \qquad (93)$$

$$k = 0, 1, \ldots$$

where

$$PI(0,\ell;m) = \frac{1}{T} \overline{PI}(0,\ell;m). \qquad (94)$$

The inequality

$$PI(k+1,\ell;m) \leq PI(k,\ell;m) - \epsilon' \qquad k = 0, 1, \ldots \qquad (95)$$

which is used in the decisions for updating the control gain-change vector, reduces to

$$\frac{\overline{PI}(k+1,\ell;m)}{T} \leq PI(k,\ell;m) - (k+2)\epsilon' \qquad (96)$$

$$k = 0, 1, \ldots$$

when eq. (93) is substituted into it. If the on-line-learning algorithm is functioning properly, $PI(k,\ell;m)$ will be getting smaller as k gets larger; however, $(k+2)\epsilon'$ will be getting larger as k increases. This means that unless ϵ' is a function of k there will be values of k for which eq. (96) will not be satisfied, since $\overline{PI}$ must remain positive. Actually, if $k = k_{max}$ is the largest positive integer (or zero) for which the right-hand side of eq. (96) is positive then as $PI(k,\ell;m) \to \epsilon' \; k_{max} \to 0$. It must be concluded, therefore, that ϵ' must be functionally dependent upon k.

Various rules were tried for $\epsilon'(k)$, such as $\epsilon'/(k+2)$, an exponential rule and, finally, the rule

$$\epsilon'(k) = C_2 \; PI(k,\ell;m) \qquad k = 0, 1, \ldots . \qquad (97)$$

*The generalization to the linear reinforcement average is straightforward. The argument is somewhat different, however, than the one presented below for straight averaging.

It was found (through simulations) that the rule $\epsilon'/(k+2)$ was too sensitive, i.e., using this threshold the feedback paiameter vector was continually changed due to the fact that for a specific value of k $\epsilon'/(k+2)$ was usually too big a change to be required of PI(k,ℓ;m). The exponential rule, on the other hand, was not sensitive enough; $\epsilon'(k)$ got very small so quickly that extremely small changes in PI(k,ℓ;m) were considered successful. The constant C_2 in eq. (97) is a design parameter and may be interpreted as a fractional change in PI(k,ℓ;m). Choices for C_2 are discussed in a subsequent subsection.

Control Situations: Generally, two types of control situations are distinguished: off-line or nominal control situations, which are based upon a priori ranges for the plant parameters (as in Section 2); and on-line control situations, which are based upon the actual on-line trajectories taken by the parameters in the augmented plant parameter space. In certain applications it may even be expeditious to reorganize the nominal control situations into on-line control situations which are geometrically different from the nominal situations.

Suppose, for example, a(t) and α(t) vary as shown in fig 42a. In this example the nominal control situations might be restructured as in fig. 42b in order to make use of the on-line trajectory information and, also, to permit finer refinements in the control gains to be made than might have been possible for the nominal control situations. While it is relatively straightforward to mechanize the construction of circular or hyperspherical regions (it is not an easy matter to mechanize hypersquares) on-line, it is

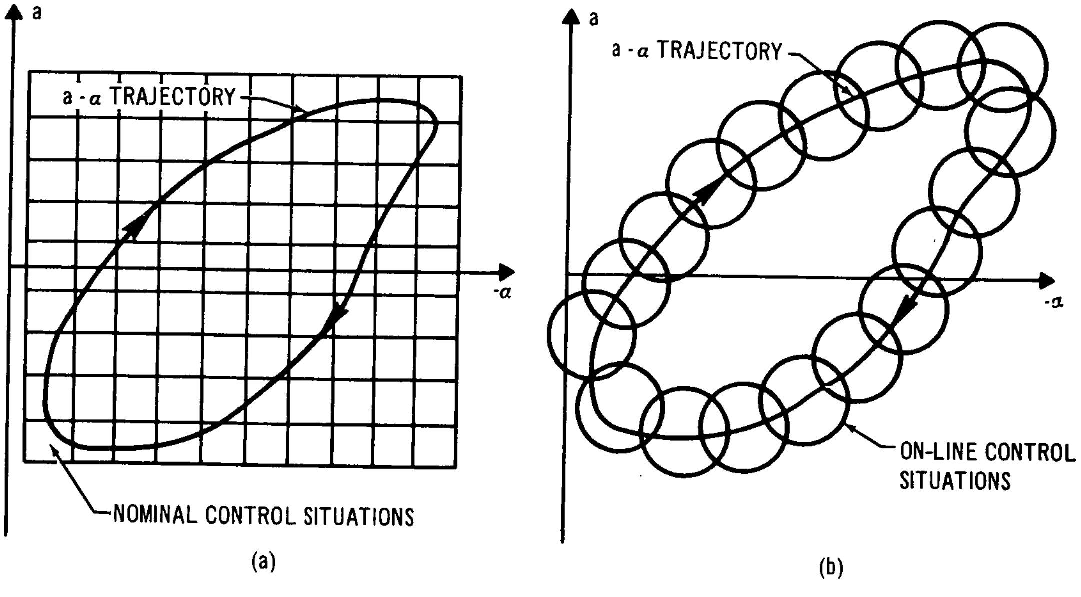


Figure 42. An Hypothetical Plant Parameter Trajectory in Relation to (a) Nominal Control Situations and (b) On-Line Control Situations

not an easy matter to construct hyperspherical control situations in the sense of the control situation definition in Section 2. For the above reason and, also, because it is doubtful that distinct trajectories like the one in fig. 42a will occur in the present application, the nominal control situations designed in Section 2 are also used as on-line control situations. Distinct trajectories are unlikely because a(t) varies rather slowly.

Simulation Studies

Several experiments were performed in order to evaluate the Error-Correction Learning Algorithm and to determine suitable values for some of the design parameters which are utilized by the algorithm. These parameters are γ, C_1, and C_2 which are associated with the reinforcement averaging [eq. (63)], control gain-changing [eq. (85)] and threshold setting [eq. (97)], respectively.

Despite the fact that the experiments described below are of the simplest variety, difficulties were experienced. These difficulties are similar to those usually experienced during the solution of a multivariable optimization problem with an heuristic algorithm. They are compounded here by the numerous design parameters which enter into the algorithm, and by the dynamic nature of the on-line optimization problem.

The following three types of experiments were performed:

1. Experiment A--Correct nominal controls and (on-line) $p_\ell = 0.96$*
2. Experiment B--Incorrect nominal controls and (on-line) $p_\ell = 0.96$
3. Experiment C--Correct nominal controls and (on-line) $p_\ell \neq 0.96$

The purpose of Experiment A was twofold. First, the experiment checked the error-correction algorithm to see if the algorithm updated the control gains unnecessarily; and, second, it investigated different choices for the design parameters γ, C_1 and C_2.

Experiment B may be interpreted in either of the following ways. First, it may be viewed as a test of the on-line learning procedure as a design tool. Second it may be viewed as a test of the error-correction algorithm to properly update the control gains in the face of plant parameter errors, if such errors degrade attitude errors.

In Experiment C on-line learning was used to update the control gains in order to compensate for erroneous a priori information about disturbance torques.

*In Experiments A and B the on-line p_ℓ = off-line p_ℓ; in Experiment C, on the other hand, the actual disturbance torques differed from those assumed during the nominal controller design.

In each experiment the following assumptions were made:

1. The actual disturbance, $\ell(t)$, is a first-order Markov process. In Experiments A and B $\ell(t) = \xi(t)$; however, $\ell(t) \neq \xi(t)$ in Experiment C, the difference being that $\sigma_\ell \neq \sigma_\xi$ [see Assumption 3, below, and Appendix C,].

2. There is no measurement noise and all states are observable; hence, the loop is closed around the plant and $\hat{\underline{x}}_a(t|t) = \underline{x}_a(t)$ and $\underline{z}(t) = \underline{x}_a(t)$ in fig. 3.

3. The augmented plant parameters are completely known because the parameters are generated. Hence, $\hat{\underline{P}}_{A_a}(t|t) = \underline{P}_{A_a}(t)$ in fig. 3. In addition, a (t) and α(t) are fixed at constant values for each Experiment.

A simplified version of fig. 3, one that describes these experiments, is shown below in fig. 43. This system was simulated on the IBM 7094.

A total of 51 different runs were performed: 7 for Experiment A, 34 for Experiment B, and 10 for Experiment C. The following values of a and α were used in one or more of these runs: (a,α) = (0. 40, -0. 80), (0, -0. 40), (-0. 30, -0. 40) and (-0. 30, -0. 80). For a = 0. 40, 0, and -0. 30 the open-loop system is stable, has a double pole at the origin, and is unstable, respectively. A decision interval equal to 0. 05 was used in every run (time did not permit the decision time to be considered as a design parameter). Experimental results for each experiment are presented and discussed next.

Experiment A. --During this experiment, different rules for the threshold function $\epsilon'(k)$ were investigated. Discussions on three of these appeared previously in Section 4. The final choice for $\epsilon'(k)$ is given in eq. (97). This rule was simulated for different values of C_2 [0. 05, 0. 10, and 0. 20], which was finally set at 0. 20; hence, it was required that the $(k+1)^{st}$ sub-goal performance index, $PI(k+1,\ell;m)$, be 20 percent less than $PI(k,\ell;m)$. Within this experiment, this 20 percent rule worked well. It did not work so well, however, in the other experiments (see subsection titled "Experiment C").

Different values for the design parameter C_1 in eq. (85) were investigated in relation to the rate at which $x_1(t)$ was driven less than or equal to 0. 20 arcsec. Values of C_1 = 0. 05, 0. 10, and 0. 20 were tried. The final choice for C_1 was 0. 10; hence, the amplitude of the on-line-optimal feedback parameter vector was updated by 10 percent of its preceding value.

Different values for the learning parameter, γ, in the linear reinforcement average were tried (γ = 0. 10, 0. 20, 0. 40 and 0. 60). A value of γ = 0. 10 was finally chosen.

The choice for a suitable value of γ is complicated by a paradox, which became apparent after a few runs. Suppose, for example, that the system's performance is initially poor; that is to say, attitude error is increasing and $PI(k+1,\ell;m) > PI(k,\ell;m)$ for k = 0,... , K_1. Assume next that the system's

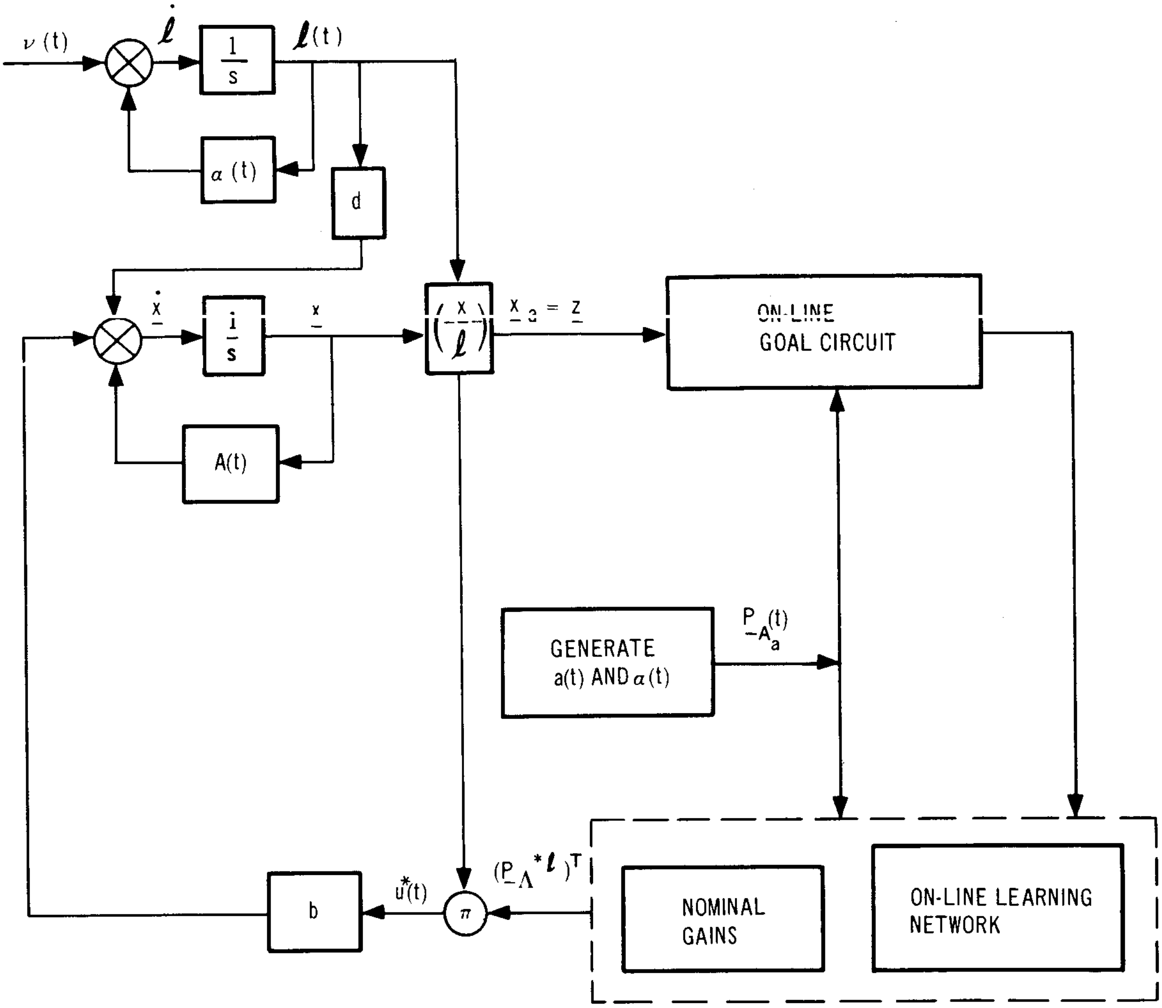


Figure 43. On-Line-Learning Control System for Experiments A, B, and C

performance improves during the (K_1+1)st time interval; i. e., $\overline{PI}(K_1+1,\ell;m) < \overline{PI}(K_1,\ell;m)$. It would certainly not seem desirable to change $\underline{P}^{*\ell}_{\Lambda}$ at this point; however, if γ is too large (usually, > 0. 20) the past poor performance is too heavily weighted and, as a result

$$PI(K_1+1,\ell;m) \nless PI(K_1,\ell;m) - \epsilon'(K_1).$$

Therefore $P^{*\ell}_{\Lambda}$ is once again changed. In this situation, a small value for γ is desirable so that when the system's performance goes from bad to good the present is weighted more heavily than the past. Assume, next, that the system performs well for $k = K_1+1, \ldots, K_2$, which means that $\overline{PI}(k+1,\ell;m) < \overline{PI}(k,\ell;m)$ for $k = K_1+1, \ldots, K_2$. Now, since $x_1(t)$ is a stochastic process it may easily occur that, for $K_2T<t<(K_2+1)T$, $x_1(t)$ is larger than it was for $(K_2-1)T<t<K_2T$ and is just larger than ±0.20 arcsec; hence, $\overline{PI}(K_2+1,\ell;m) > \overline{PI}(K_2,\ell;m)$ which means that the control gains must be changed. With a small value of γ, the degraded performance is weighted much heavier than all good previous performance, even though the system was performing well during many iterations. In this situation a larger value for γ is desirable. Obviously, unless γ is adaptive, the requirements for small and large γ can not be satisfied simultaneously.

Experiment B. --During this experiment the system was initially controlled by an incorrect nominal controller (the nominal controller designed in Section 2 is the correct nominal controller). The incorrect nominal controls were obtained in two ways. In one approach, an incorrect value for the weighting function, ρ, was specified during the computation of the nominal gains λ_1^ℓ, λ_2^ℓ, and λ_3^ℓ. Since the three gains are functions of ρ [eqs. (3) to (5)], each gain was incorrectly specified in this first approach. In another approach only λ_1^ℓ was specified incorrectly; it was chosen different enough from the true value that attitude errors exceeded ±0. 20 arcsec.

Regardless of how poorly ρ or λ_1^ℓ were chosen, the performance of the system improved, initially; that is attitude errors decreased, initially. In these cases, however, a limit cycle larger in magnitude than 0. 20 arcsec occured for $x_1(t)$; hence, on-line learning was required to reduce the magnitude of the limit cycle to within 0.20 arcsec.

After a number of unsuccessful runs, during which the three components of $\underline{P}^{*\ell}_{\Lambda}$ were updated simultaneously, it was decided to hold λ_2^ℓ and λ_3^ℓ at the initial values and to update λ_1^ℓ only. The justification for this stems in part from the results of the nominal control design. There (Table III) it is observed that λ_1^ℓ is most sensitive to a and α. As soon as this approach* was

*Varying λ_1, while holding λ_2 and λ_3 fixed, represents an incomplete relaxation-method approach to the solution of the on-line optimization problem.

adopted the error-correction algorithm functioned as expected and many successful runs were obtained. The details for one run are discussed next.

In fig. 44 the magnitude of the attitude error is depicted when a = -0. 30 and α = -0. 80 for two conditions. The dashed curve depicts the behavior of $|x_1(t)|$ under the action of the correct nominal controller [from Table III and fig. 10 $\lambda_1^{18} \cong 284.55$, $\lambda_2^{18} \cong 23.86$ and $\lambda_3^{18} \cong 999.78$]. The solid curve depicts $|x_1(t)|$ under the action of an incorrect controller [$\lambda_1 = 100$, $\lambda_2 = \lambda_2^{18}$, and $\lambda_3 = \lambda_3^{18}$]. Note the destabilizing effect when λ_1 is incorrect. Before proceeding, it should be mentioned that the random sequences used in generating the disturbance torque were identical in both cases; hence, the system was subjected to the same random disturbances in both cases. This sequence was also utilized in the learning runs and was available by virtue of the way the sequence was generated and programmed for the 7094. The disturbance state, which remains unchanged during learning, since it is an uncontrollable state, and which reflects this random sequence, is depicted as the dashed curve in fig. 45. It must be observed that correlation between the large values of $|\xi(t)|$ and the values of $|x_1(t)|$ that are greater than 0. 20 arcsec exists (fig. 44).

The behavior of $|x_1(t)|$ during on-line learning is depicted in fig. 45 (solid curve). In addition, the behavior of $\lambda_1^*(t)$ is shown directly above the plot of $|x_1(t)|$. The time scales are the same for both plots. Observe that there was always a lag of at least 0. 05 units of time before the learning controller sensed that the attitude errors exceeded ±0. 20 arcsec. This is as expected, since the decision time T = 0. 05. It must also be observed, that $\lambda_1^*(t)$ was updated only when $|x_1| > 0.20$ arcsec, as postulated (the stopping rule which is embodied in the theorem presented in the subsection entitled "On-Line Cost Functions" was used to ensure this behavior); hence, the algorithm was functioning in an error-correcting mode, as designed.

At first sight, it does not appear as though the system's performance was improved during on-line learning. Looking closer at $|x_1(t)|$ in fig. 45 and fig. 44 one notes, however, that the behavior of $|x_1(t)|$ in the on-line learning case is superior to $|x_1(t)|$ when $\lambda_1 = 100$. For example, the time interval during which $|x_1(t)| > 0.20$ arcsec is always smaller in the on-line learning situation. This is because of the attempted improvements (this may not always be the case however). Note also that $\lambda_1^*(t)$ is moved in the proper direction by the algorithm, although there are many detours, the most notable of which occurred at the end of the run. These "detours" and other movements of $\lambda_1^*(t)$ corroberate the limit cycles discussed in connection with fig. 41. For example, the sideward movement for $4.2 < t < 4.5$ is indicative of the limit cycle from state (1, 2) to (2, 2) to (1, 2), etc. The strange behavior for $10.35 < t < 10.5$ is predictable and is indicative of the self-loop limit cycle at state (2, 1).

It must also be pointed out, that for $7.65 < t < 7.75$ the attitude error exceeds 0. 20 arcsec, but for only a very short time interval. The algorithm ignores the good performance of the system over the preceding 35 iteration intervals (T = 0. 05) and changes $\lambda_1^*(t)$. Fortunately, the change was made in the proper direction and the attitude error was swiftly contained.

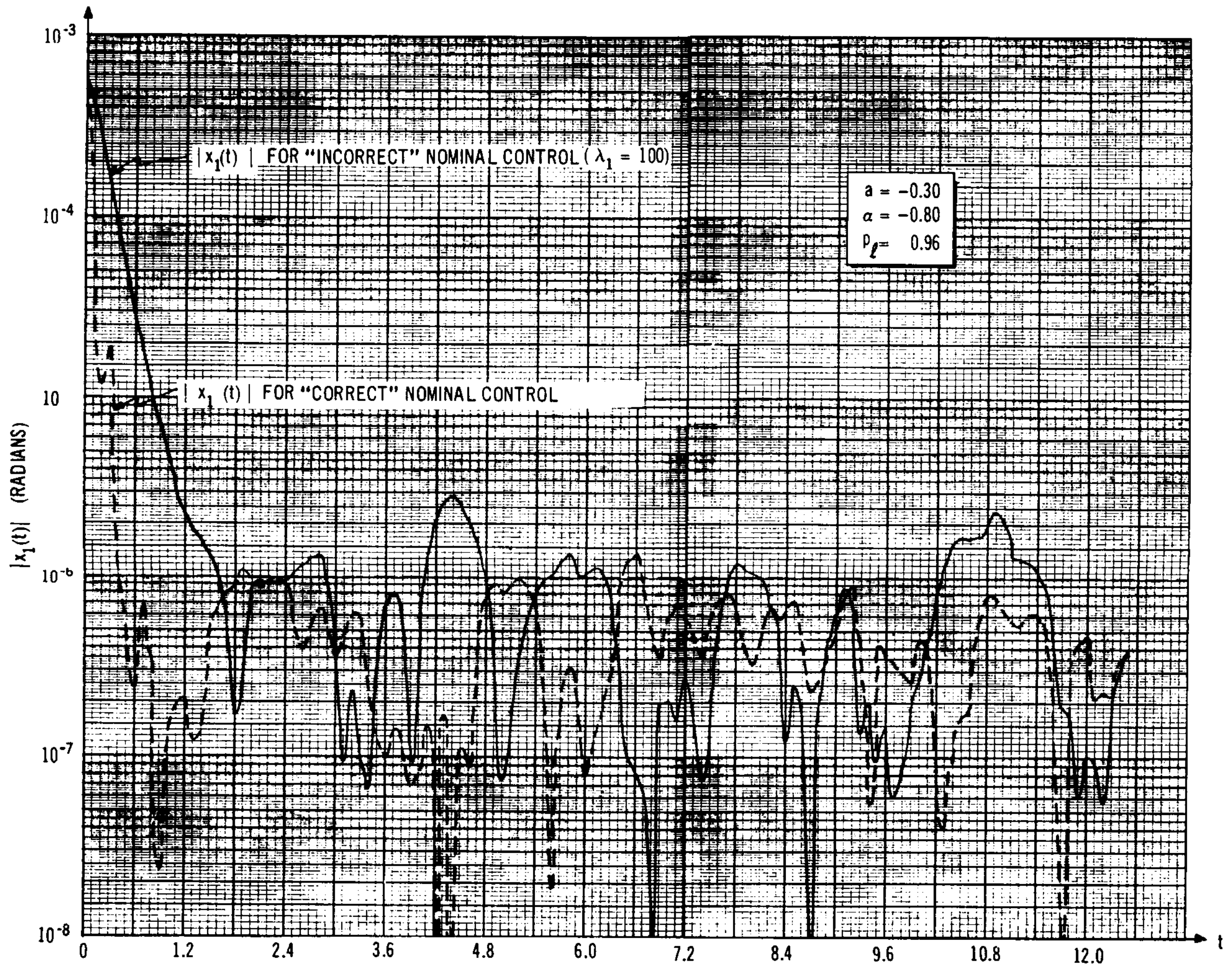


Figure 44. x_1 (t) For "Correct" and "Incorrect" Nominal Control

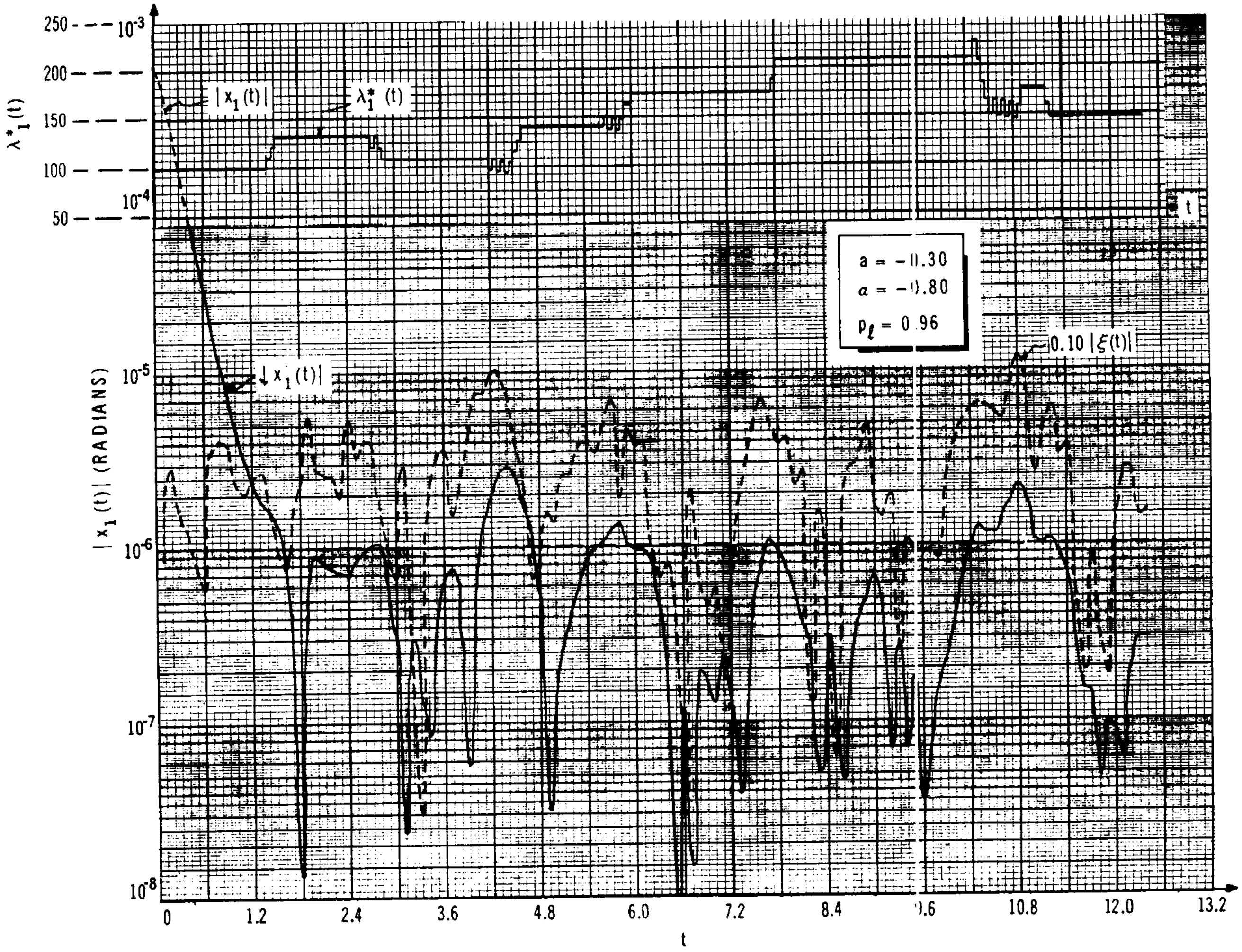


Figure 45. $|x_1(t)|$, $|\xi(t)|$ and $\lambda^*_1(t)$ for One Run From Exper ment B

If a larger value of γ had been used, one which took into account the satisfactory performance over the 35 preceding iteration intervals the changes in $\lambda_1^*(t)$ might not have been made because they might not have been necessary (note that $|\xi(t)|$ is decreasing).*

Experiment C. --During this experiment the system was initially controlled by the nominal controller, which was designed under the a priori assumption that the on-line $\Pr\{|\ell|\leq 10^{-4}$ lb-ft$\} = 0.96$. The system was disturbed on-line, however, by larger disturbances; that is to say, during this experiment $p_\ell \neq 0.96$. The nominal controls are correct when $p_\ell = 0.96$. They are incorrect when $p_\ell \neq 0.96$; hence, the need for on-line learning during the present experiment.

It was observed that, for the system under the control of the nominal controller, attitude errors are contained when the on-line p_ℓ was as low as 0.50. Runs were made when $p_\ell = 0.10$ and 0.40. Some were successful, others were not. The results from one of the runs, where a = 0.40 and $\alpha = -0.80$, are depicted in fig. 46. Once again, observe the correlation between large values of $|\xi(t)|$ and large value of $|x_1(t)|$. Although only three changes in $\lambda_1^*(t)$ are necessary in order to maintain $|x_1(t)| \leq 0.20$ arcsec more changes may be required should attitude errors again exceed ±0.20 arcsec for values of time greater than t = 3.6.

In some of the runs, C_2, in the rule for $\epsilon'(k)$, was observed to be too large. Generally, too large a threshold has a deleterious effect on $\lambda_1^*(t)$. For example (Refer to Table VII and fig. 41), if the system is initially in states (1, 2), or (2, 2), and moves to state (2, 1) [in which case $\lambda_1^*(t)$ decreases] it is probable, due to too large a threshold, that it will remain in that state for many iterations. Generally, $\lambda_1^*(t)$ should increase if attitude errors are to be reduced [eq. (C. 23), for example]; however, $\lambda_1^*(t)$ decreases for as long as the system remains in state (2, 1). This condition represents, in effect, an instability in the learning algorithm. Usually, though, it is not a permanent instability, since the performance of the system deteriorates rapidly as $\lambda_1^*(t)$ becomes smaller. Eventually the system moves out of state (2, 1) and $\lambda_1^*(t)$ begins to increase.

Summary

An on-line learning procedure has been discussed and simulated. It is termed error-correcting since it leads to control-gain changes only when the present performance is worse than the preceding performance. The Error-Correction Learning Algorithm, as the learning procedure is called, has been discussed in detail. Although no mathematical proof for its convergence exists, it is felt that the algorithm should update the control gains in the proper directions. This was verified, in part, during the experiments which were performed. In addition, it was demonstrated, through the use

*There is a relationship between larger values of γ and the threshold $\epsilon'(k)$. For example, if $C_2 = 0.20$, then increasing γ may lead to larger values of PI for which a 20 percent reduction in the sub-goal performance index during one iteration interval may no longer be practical. The dependence of $\epsilon'(k)$ on γ requires further study.

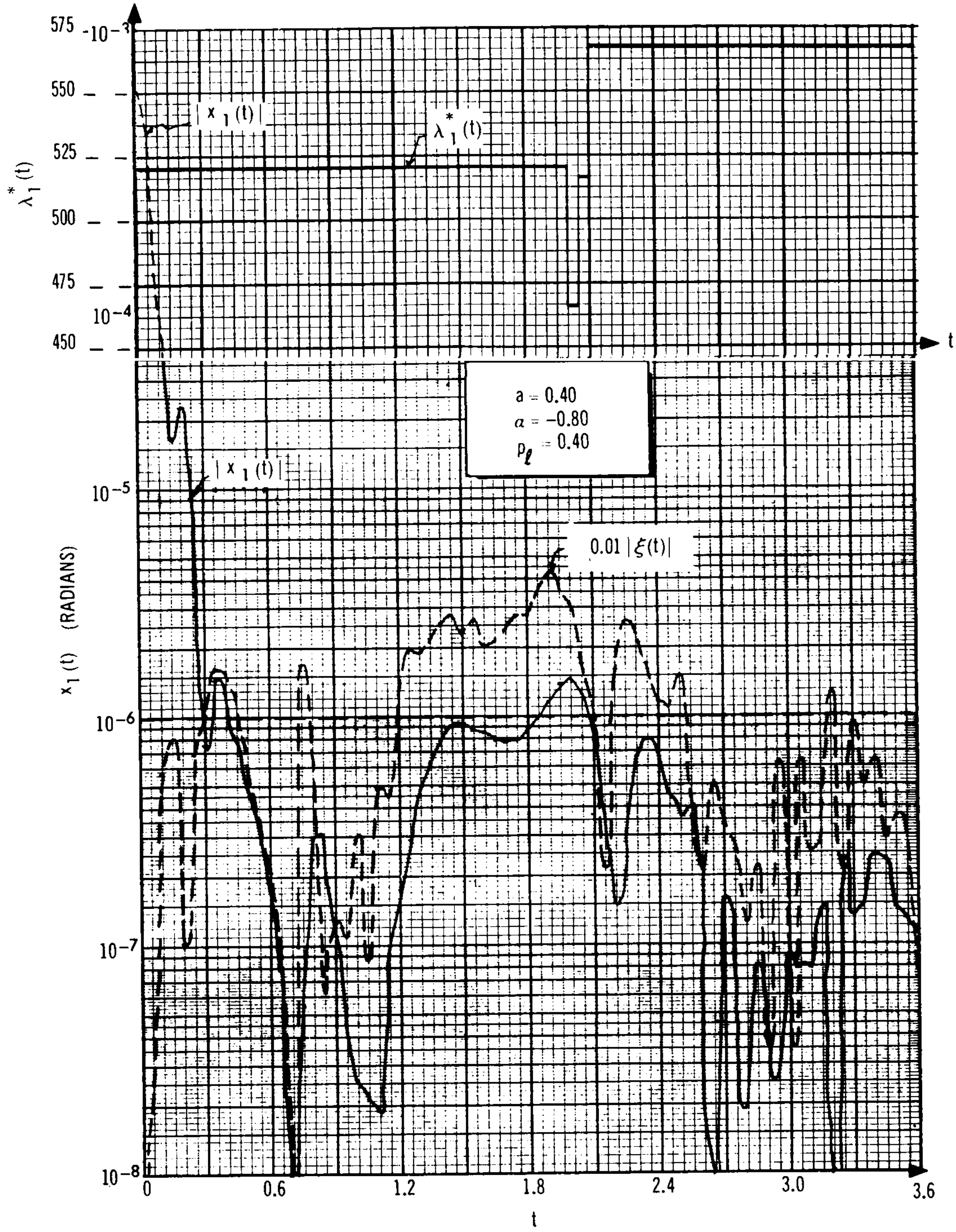


Figure 46. $|x_1(t)|$, $|\xi(t)|$ and λ_1^* (t) For One Run From Experiment C

of a state transition diagram, that six limit cycles may occur during the updating of the control gains. These limit cycles were predicted and, in addition, each was observed experimentally.

The simulation studies were for a simplified system, one in which the loop was closed around the plant and not around the state estimator derived in Section 3. The reason for this is that the study described in this section was in progress in conjunction with the study in the preceding section. Time did not permit the nominal controller to be redesigned in order to compensate for estimation errors; hence the simulation in the present section could not be performed for the system with the loop closed around the filter. In the simplified system no measurement noise was assumed and, for that case, a theorem which relates the overall performance index to the containment of attitude errors over an interval of time was proved. The extension of the theorem to the case when measurement noise is present is possible if it is assumed that the system is discrete--as it is when it is simulated digitally. The white measurement noise becomes a Gaussian random sequence and is mathematically tractable in the inequality analysis involved in the proof of the more general theorem.

Results from the experiments demonstrate that on-line learning has the potential for improving a system's performance, on-line. Much more work needs to be done, however, in connection with the algorithm used in the on-line learning procedure before a point is reached where more significant experiments can be considered. An example of these experiments is one where a and α are time-varying and the loop is closed around the estimator.

Section 5

CONCLUSIONS

A preliminary design of a fine-attitude, single-axis controller for an almost cylindrically symmetrical spacecraft, operating in a partially known environment, has been performed. The design proceeded in two stages.

First, a nominal controller was designed. This design incorporated all available information about the plant and the plant environment. If the plant and the on-line environment are as assumed off-line, the on-line attitude errors are contained to within ±0. 20 arcsec, (as required). The nominal controller is nominal with respect to the overall on-line controller, which was provided with a capability for updating the nominal controller, when and if updating is necessary.

The on-line controller designed during the second stage is an on-line-learning controller. This controller utilizes the system's past experience in attempting to improve the system's present performance. Learning is accomplished, for the most part, through inclusion of a memory into the on-line controller.

The nominal control system serves two purposes in relation to the on-line-learning control system. First, it is used as a system with which the performance of the on-line-learning control system may be compared; and a comparison of this type is important if the on-line-learning control system is to be judged fairly. Second, simulations of the system under the action of the nominal controller point up the reasons, if any, for wanting to include an on-line learning capability. Hence, the nominal controller not only serves to provide the justifications for on-line learning, but also provides a reference for statements made about the performance of the on-line-learning control system.

The nominal controller, which is a linear combination of estimated states, is an adaptive controller, in that feedback gains are changed when plant parameters change. On-line estimates of the states and time-varying parameters (two) are required, in order to implement this controller.

An augmented Kalman filter was investigated for obtaining combined, on-line estimates of the states and parameters. Due to the insensitivity of the combined estimator to plant parameter variations (±100%), satisfactory estimates of these parameters were not obtained, although good estimates were obtained for the states.

The design of the gains for the nominal controller proceeded in three steps: (1) for fixed values of plant parameters, nominal gains were obtained using stochastic optimal-control theory; (2) for specific ranges of the plant parameters, which were denoted control situations, nominal gains were obtained using the results from Step 1 and sensitivity analyses; and (3) for the gains designed in Step 2, simulations were performed in order to observe the effects of some of the assumptions made during the design. A continuation of this procedure to higher-order systems, using off-line training, is elaborated upon in Appendix D.

For the system under the control of the nominal controller, the following was observed:

(1) Disturbance levels greater than those anticipated during the design of the nominal controller may cause attitude errors to exceed ±0.20 arcsec;

(2) Attitude error is degraded by between one and two orders of magnitude due to state-estimation errors, when disturbance torques are of the order of magnitude of 10^{-4} lb-ft;

(3) If disturbance torques are only of the order of magnitude of 10^{-6} lb-ft, attitude error will remain less than ±0.20 arcsec, even though estimation errors occur (provided measurement-noise level is low);

(4) Large measurement-noise levels degrade attitude error further, due to larger state-estimation errors;

(5) Neither the time-varying nature of the plant parameters, nor parameter mismatches (at least as large as ±15%), degrade attitude error further.

Once the effects of state-estimation error, measurement noise and parameter mismatch (or parameter-estimation error) are observed, compensation should be possible through the modification of the nominal controller. On-line learning is not required for the compensation of these effects, since the effects are known a priori (before the spacecraft is in orbit).

Different disturbance levels, on the other hand, may necessitate on-line corrections of the nominal controls. One approach is to update the nominal gains only when the system's performance becomes unacceptable (attitude errors exceed ±0.20 arcsec), or if the performance deteriorates. A second approach is to update the nominal gains regardless of whether the system's performance improves or worsens. The first approach is referred to as error-correcting, and the second as reinforcing.

Two reinforcement, on-line optimization techniques were formulated for this study. The first utilizes concepts from stochastic automata theory, while the second utilizes concepts from random optimization theory. The stochastic automata approach, while interesting, does not appear to be applicable to the fine attitude control problem. This is because of the dynamic nature of this optimization problem, the time-varying behavior of the system and the difficulty in finding a meaningful stationary expected value function for the problem.

The adaptive, random-optimization approach appears promising; however, it remains to be tested (through simulations) in the context of the fine-attitude control problem. Adaptive, random optimization has, however, been used with success at Douglas in the context of a 58-dimensional stationary, calculus, hill-climbing problem; excellent convergence properties have been observed.

An error-correction technique was formulated and tested for this problem. Although no mathematical proof for the techniques convergence exists, it is felt that the technique should update the control gains in the proper directions. This was verified, in part, in a number of experiments. In addition, six limit cycles in the control gains were predicted, using a state transition diagram. Each limit cycle was observed experimentally. Unfortunately, time did not permit a complete study of the error-correction technique.

Attention should be devoted to optimum choices for the many design parameters which are a part of the Error-Correction Learning Algorithm; or, the Algorithm should be modified, to reduce the numbers of these parameters. The application of the Algorithm to the multivariable optimization problem requires further investigation. Finally, on-line cost functions which ensure desirable relative stability properties should be developed. Using the present on-line cost functions, one is only able to achieve a form of asymptotic stability for attitude errors (to a random limit cycle).

Results from the on-line learning experiments demonstrated that on-line learning has the potential for improving a system's performance. The on-line-learning controller appears, however, to be very complex, requiring an on-board digital computer in order to mechanize the on-line-learning algorithm. No doubt, the tradeoffs between an on-line-learning control system and an over-designed nominal controller (overdesigned, perhaps, by assuming $|\ell(t)| \leq 10^{-3}$ or 10^{-2} lb-ft instead of 10^{-4} lb-ft) should be investigated. The overdesigned nominal controller would, for example, use more control effort but would be less complex than the on-line-learning control system. The learning control system, on the other hand, would be able to cope with the unexpected.

A final remark about what a learning control system can and cannot be expected to do is in order. Much confusion exists in the minds of engineers on this point. This study adopted the following ground rule for on-line learning control systems. Ground Rule: An on-line-learning controller must be able to satisfactorily correct any degradation in performance that could have been treated satisfactorily had its cause been anticipated.

At the present state-of-technology in learning control systems, however, the supernatural should not be expected from these systems. For example, if an inertia wheel falls apart it should not be expected that present learning control systems learn how to repair the wheel or learn how to function without the wheel. On-line-learning control systems are only as good as the on-line learning strategies. These strategies are, at present, decided upon by the control system engineer and are, for the most part still in the very early stages of development.

Section 6

REFERENCES

1. Waymeyer, W. K.; and Blackburn, T. R.: Bending Feedback Final Report. Douglas Report DAC-59275, June 1966.

2. Bush, R. R.; and Mosteller, F.: Stochastic Models for Learning. John Wiley and Sons, New York, 1955.

3. Hill, J. D: An On-Line Learning Control System Using Modified Search Techniques, Ph.D. Thesis, Purdue Univ., Jan. 1965.

Appendix A

SYSTEM DYNAMICS

The equations of motion for a satellite in Earth or Mars orbit are presented in this section. It is assumed that fine attitude control of the satellite is required, as in the case, for example, of a pair of laser communication satellites; the satellite telescopes must be pointed very accurately in order to maintain the communication links between the satellites (fig. A-1).

Before presenting the equations of motion for the satellite, a distinction must be made between coarse and fine attitude control and fine-pointing control. These distinctions will provide the proper frame of reference for the specific problem discussed in this document.

By course attitude control is meant, that mode of control which brings attitude errors to within ±M arcsec (for example, 80 arcsec). Fine attitude control (fig. A-2) refers to that mode of control which brings attitude errors from within ±M arcsec to within $\pm\epsilon$ arcsec (for example, 0.2 arcsec for a laser communication satellite). Once fine attitude control is achieved, that mode of control which points the telescope is fine-pointing control; thus, unless attitude errors can first be maintained less than or equal to $\pm\epsilon$ there is no hope of pointing the telescope properly. In this study it has been assumed that these modes of control are distinct.

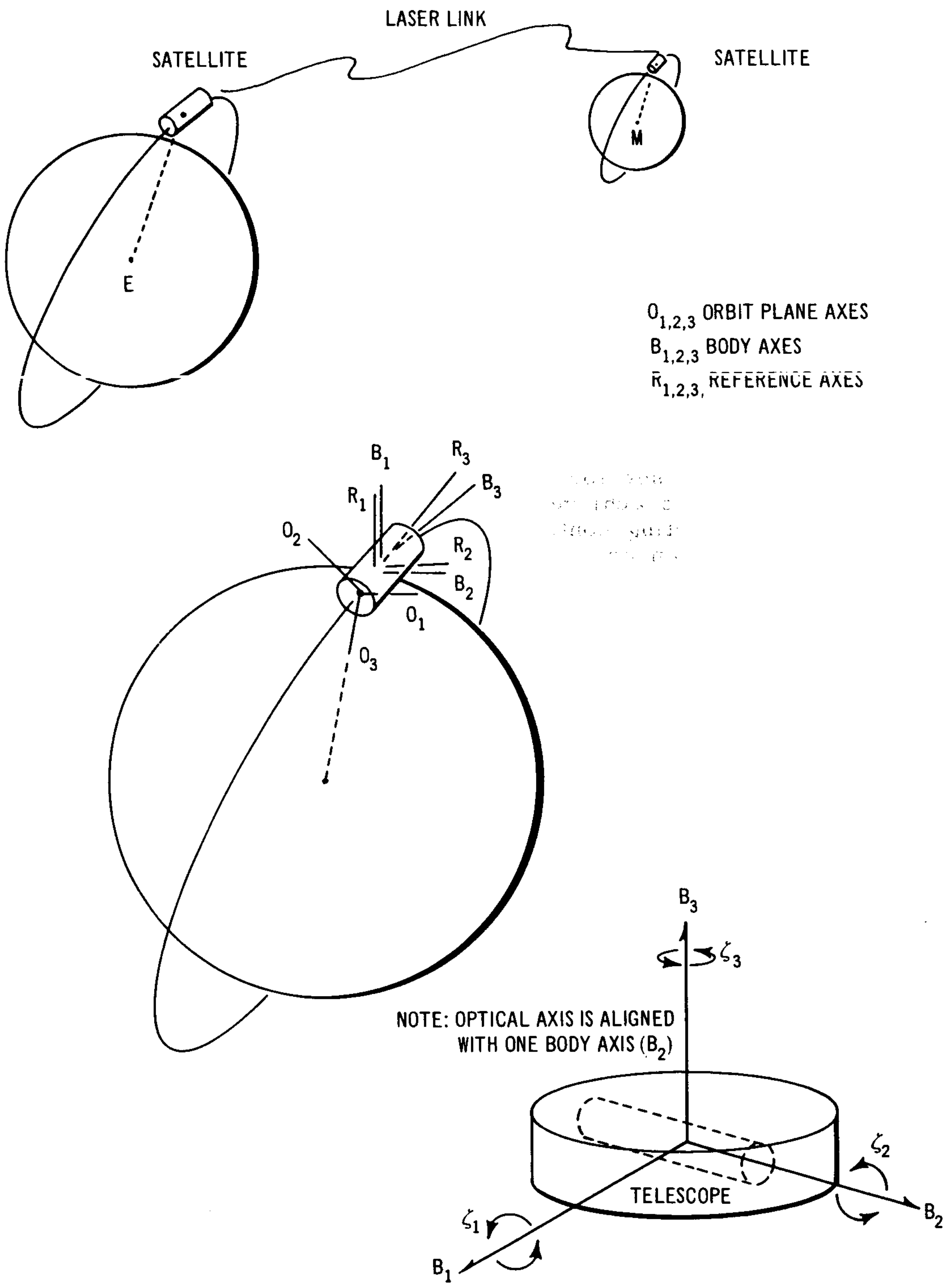
Equations of Motion

The equations of motion for a satellite are derived in ref. A1. The following assumptions are made in this document:

(1) Attitude error perturbations are small.

(2) All aerodynamic and incident radiation torques except for torques due to attitude changes about the same axis are lumped into perturbation torque terms.

(3) Gravity torque contributions of planetary oblateness and higher-order potential terms are lumped into perturbation torque terms.



NOTE: OPTICAL AXIS IS ALIGNED WITH ONE BODY AXIS (B_2)

Figure A-1. Laser Satellite and Axis System

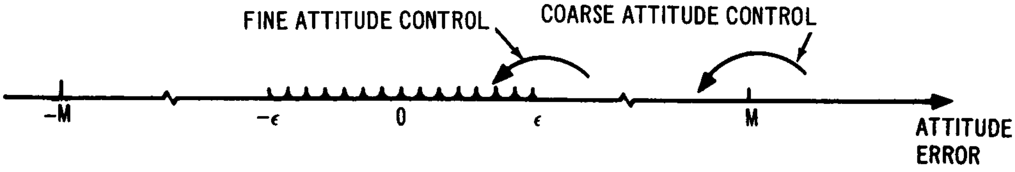


Figure A-2. Regions for Coarse and Fine Attitude Control and Fine-Pointing Control (Shaded Region)

(4) All reference axis angular velocities are neglected*.

(5) Linear perturbation motions (translation-rotation coupling) are neglected.

*Reference axis angular velocities are mainly due to velocity aberration of the distant source. The maximum angular velocities associated with this are given by $(\omega)_{max} = (K/c)\sigma^2$ where c is the speed of light. Numerical values for $(\omega)_{max}$ for four cases are tabulated below, and are seen to be quite small from the point of view of inertial coupling. Reference axis angular velocities due to parallax should be entirely negligible.

Central Body	Orbital Altitude, nmi	$(\omega)_{max}$, rad/sec.
Earth	100	0.0233×10^{-6}
Earth	19,300 (synchronous)	0.000717×10^{-6}
Mars	100	0.00878×10^{-6}
Mars	1,000	0.00442×10^{-6}

(6) Reference axes are at arbitrary inclinations to orbit axes, but are fixed in inertial space.

(7) Only moving parts are three axially symmetric inertia wheels rotating on axes fixed to the vehicle.

(8) Each wheel axis is aligned with a vehicle principal axis (defined on the composite vehicle, including wheels).

(9) Inertia wheels are stopped when attitude errors are all zero.

(10) Satellite is cylindrically symmetrical.

(11) Control is proportional to the system's state variables.

These assumptions permit us to linearize the equations of motion, lump all perturbation disturbance torques into a single term, and decouple the equation of motion for one of the three body axes (fig. A-1) from the complete set of equations.

The equations of motion can be shown to reduce to the following set:

$$I_1 \frac{d^2\zeta_1}{dt^2} + c_{44}(t)\zeta_1(t) + c_{45}(t)\zeta_2(t) + c_{46}(t)\zeta_3(t) = - C^1 d_B \dot{\omega}^1(t) + \delta_B L_1(t)$$

$$I_2 \frac{d^2\zeta_2}{dt^2} + c_{54}(t)\zeta_1(t) + c_{55}(t)\zeta_2(t) + c_{56}(t)\zeta_3(t) = - C^2 d_B \dot{\omega}^2(t) + \delta_B L_2(t)$$

$$I_3 \frac{d^2\zeta_3}{dt^2} + c_{64}(t)\zeta_1(t) + c_{65}(t)\zeta_2(t) + c_{66}(t)\zeta_3(t) = - C^3 d_B \dot{\omega}^3(t) + \delta_B L_3(t) \qquad \text{(A1)}$$

The nomenclature used in eq. (A1) is that found in ref. A1 and is defined in the list of symbols at the beginning of this report. In addition, the parameters $c_{44}(t)$, $c_{45}(t)$, . . . , and $c_{66}(t)$ are tabulated in table A-I.

For notational convenience, eq. (A1) is rewritten as

$$\ddot{\zeta}_1(t) + a_1(t)\zeta_1(t) + a_2(t)\zeta_2(t) + a_3(t)\zeta_3(t) = b_1 d_B \dot{\omega}^1(t) + d_1 \ell_1(t)$$

$$\ddot{\zeta}_2(t) + a_4(t)\zeta_1(t) + a_5(t)\zeta_2(t) + a_6(t)\zeta_3(t) = b_2 d_B \dot{\omega}^2(t) + d_2 \ell_2(t)$$

$$\ddot{\zeta}_3(t) + a_7(t)\zeta_1(t) + a_8(t)\zeta_2(t) + a_9(t)\zeta_3(t) = b_3 d_B \dot{\omega}^3(t) + d_3 \ell_3(t) \qquad \text{(A2)}$$

TABLE A-I

PARAMETERS IN EQUATION (A1)

Symbol	Formula*
$c_{44}(t)$	$-\frac{\partial L_1}{\partial \zeta_1} + (I_2 - I_3)\frac{3K}{\sigma 3}\cos 2\eta_1(t)\cos^2\eta_2(t)$
$c_{55}(t)$	$-\frac{\partial L_2}{\partial \zeta_2} + (I_3 - I_1)\frac{3K}{\sigma 3}\left[\sin^2\eta_2(t) - \cos^2\eta_1(t)\cos^2\eta_2(t)\right]$
$c_{66}(t)$	$-\frac{\partial L_3}{\partial \zeta_3} + (I_1 - I_2)\frac{3K}{\sigma 3}\left[\sin^2\eta_1(t)\cos^2\eta_2(t) - \sin^2\eta_2(t)\right]$
$c_{45}(t)$	$(I_3 - I_2)\frac{3K}{\sigma 3}\sin\eta_1(t)\sin\eta_2(t)\cos\eta_2(t)$
$c_{56}(t)$	$(I_1 - I_3)\frac{3K}{\sigma 3}\sin\eta_1(t)\cos\eta_1(t)\cos^2\eta_2(t)$
$c_{64}(t)$	$(I_2 - I_1)\frac{3K}{\sigma 3}\cos\eta_1(t)\sin\eta_2(t)\cos\eta_2(t)$
$c_{46}(t)$	$(I_2 - I_3)\frac{3K}{\sigma 3}\cos\eta_1(t)\sin\eta_2(t)\cos\eta_2(t)$
$c_{54}(t)$	$(I_3 - I_1)\frac{3K}{\sigma 3}\sin\eta_1(t)\sin\eta_2(t)\cos\eta_2(t)$
$c_{65}(t)$	$-(I_1 - I_2)\frac{3K}{\sigma 3}\sin\eta_1(t)\cos\eta_1(t)\cos^2\eta_2(t)$

* $\left(\frac{\partial L_i}{\partial \zeta_i}\right)$ are aerodynamic and incident radiation torque derivatives which can be pre-estimated.

where

$$\ell_i(t) = \delta_B L_i(t) \qquad i = 1,\ 2,\ 3 \qquad \text{(A3)}$$

and the parameters $a_1(t), \ldots, a_9(t), b_1, b_2, b_3, d_1, d_2$, and d_3 are tabulated in table A-II. Observe that for a specific vehicle configuration, ranges on $a_1(t), \ldots,$ and $a_9(t)$ can be obtained from the expressions for these quantities in table A-II.

TABLE A-II

PARAMETERS IN EQUATION (A2)

Symbol	Formula	Symbol	Formula
$a_1(t)$	$c_{44}(t)/I_1$	$a_9(t)$	$c_{66}(t)/I_3$
$a_2(t)$	$c_{45}(t)/I_1$	b_1	$-C^1/I_1$
$a_3(t)$	$c_{46}(t)/I_1$	b_2	$-C^2/I_2$
$a_4(t)$	$c_{54}(t)/I_2$	b_3	$-C^3/I_3$
$a_5(t)$	$c_{55}(t)/I_2$	d_1	$1/I_1$
$a_6(t)$	$c_{56}(t)/I_2$	d_2	$1/I_2$
$a_7(t)$	$c_{64}(t)/I_3$	d_3	$1/I_3$
$a_8(t)$	$c_{65}(t)/I_3$		

The control variables $u_1(t)$, $u_2(t)$, and $u_3(t)$ are taken as the relative (to the vehicle) angular accelerations of the three inertia wheels. This implies the existence of tight closed-loop control of the relative angular accelerations of the wheels.

$$u_1(t) = d_B \dot{\omega}^1(t)$$

$$u_2(t) = d_B \dot{\omega}^2(t)$$

$$u_3(t) = d_B \dot{\omega}^3(t) \qquad \text{(A4)}$$

Next, note from tables A-I and A-II and eq. (A1) that if $I_1 = I_2$, $I_2 = I_3$ or $I_1 = I_3$ (see assumption 10) then one of the axes (ζ_3, ζ_1 and ζ_2, respectively) is completely uncoupled from the remaining axes. In these cases, instead of investigating the control of a completely coupled sixth-order system, the control of a second-order system and a coupled fourth-order system can be investigated. For example, if $I_1 = I_2$, eq. (A2) becomes

$$\ddot{\zeta}_1(t) + a_1(t)\zeta_1(t) + a_2(t)\zeta_2(t) + a_3(t)\zeta_3(t) = b_1u_1(t) + d_1\ell_1(t)$$

$$\ddot{\zeta}_2(t) + a_4(t)\zeta_1(t) + a_5(t)\zeta_2(t) + a_6(t)\zeta_3(t) = b_2u_2(t) + d_2\ell_2(t)$$

$$\ddot{\zeta}_3(t) + a_9(t)\zeta_3(t) = b_3u_3(t) + d_3\ell_3(t) \qquad \text{(A5)}$$

Hence this document shall concentrate initially (for the present study) on the second-order system

$$\ddot{\zeta}_3(t) + a_9(t)\zeta_3(t) = b_3u_3(t) + d_3\ell_3(t) \qquad \text{(A6)}$$

Before this equation is written in state space notation, values for b_3 and d_3, a range for $a_9(t)$, and the order of magnitude for the perturbation disturbance torques, $\ell(t)$, are determined. These enable the formulation of a scaled version of eq. (A6) in state space notation. The scaled version is more useful than the unscaled version, especially for numerical calculations and computer simulations.

Ranges for Certain Parameters in the Satellite Equations of Motion

As an initial step, the vehicle configuration is identified further. For the purpose of obtaining aerodynamic and solar radiation torque parameters, a cylindrically symmetrical external configuration is assumed. The mass distribution can also be assumed to be such that the transverse moments of inertia I_1 and I_2 are almost equal. Douglas studies of laser communication satellites have resulted in configurations that are close to cylindrical symmetry. Solar cells, which are arrayed on paddles on the Orbiting Astronomical Observatory (OAO), destroying cylindrical symmetry, are mounted on the sides and ends of the spacecraft shell in these studies. Except for a relatively small radar antenna, these configurations have been symmetrical.

Ranges on the Parameter d_3. --The parameter $d_3 = 1/I_3$ is the inverse of the axial moment of inertia. Fig. A-3 shows the ranges of weight and moment of inertia expected for the laser satellite. The weight ranges shown are taken from the study that resulted in Douglas Document Number SM-48730P, Proposal to Perform a Preliminary Design Study for an Optical Technology Apollo Extension System. At the low end of the range, the laser satellite weighs 900 lb and carries a single telescope with a 3-ft aperture. The upper end of the range is a 15,000-lb satellite carrying 3- and 5-ft aperture telescopes. The earlier study was not detailed enough to result in moment of inertia estimates. The correlation in fig. A-3, which includes the Mercury, OAO, and MORL space vehicles was used to provide this information. The resultant range on the parameter d_3 is 10^{-4} to 1.1×10^{-2} 1/slug-ft^2. A nominal value of 10^{-3} 1/slug-ft^2 is suggested, corresponding to a laser satellite weighing 4,000 lb.

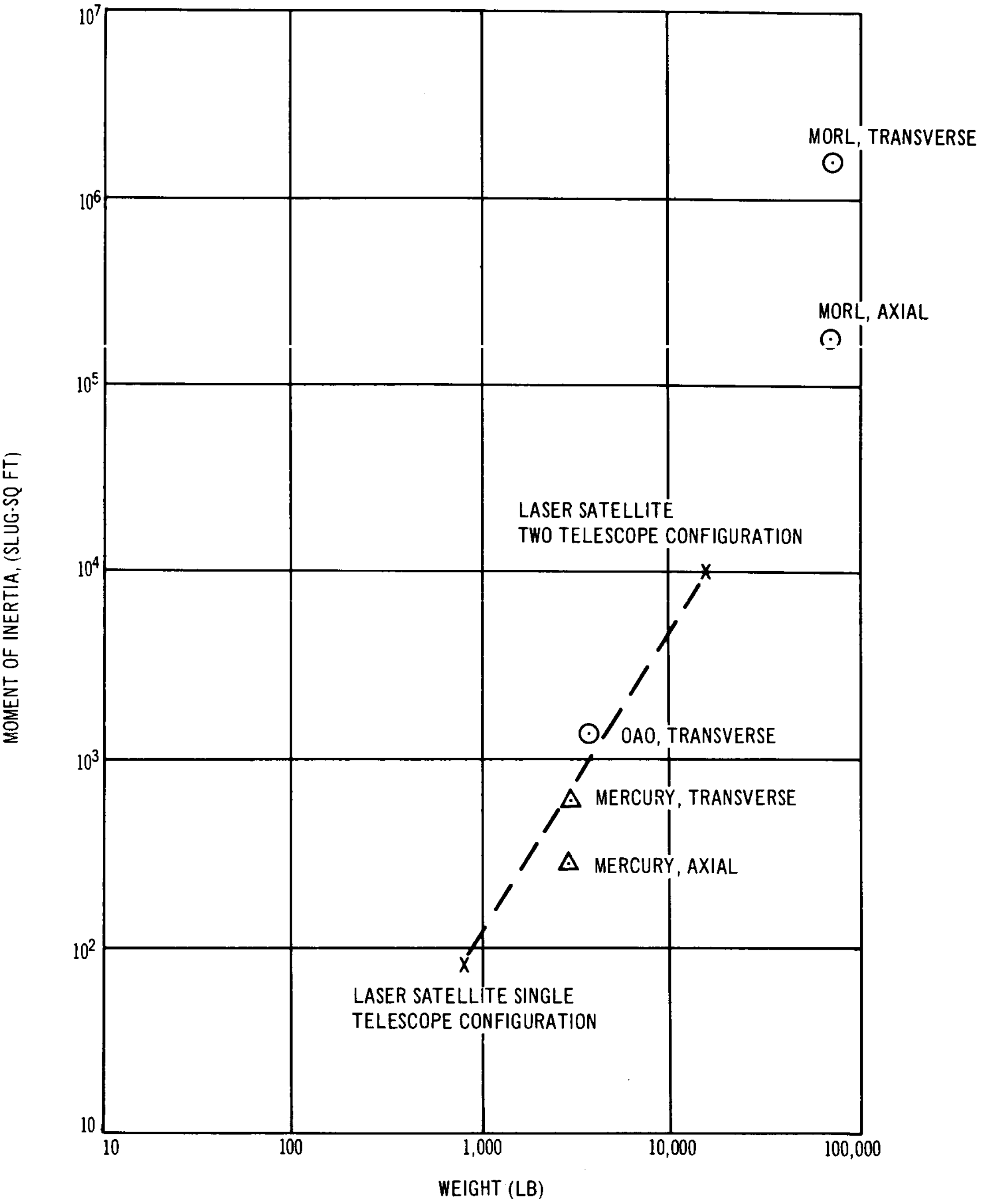


Figure A-3. Weight and Moment-of-Inertia Ranges for Laser Satellite

Ranges on the Parameter b_3. --The parameter $b_3 = -C^3/I_3$ is the ratio of moment of inertia for the fine-pointing inertia wheel that controls the attitude error $\zeta_3(t)$, to the moment of inertia about that axis. The maximum range of this parameter is -10^{-7} to -10^{-5}. The smaller number would apply for extremely precise control. A more reasonable value for fine pointing, which is suggested for the current study, is -10^{-6}. Papers written by a number of authors were consulted in arriving at these results, including DeBra (ref. A2), Cannon (ref. A3) and Adams (ref. A4).

Order of Magnitude for Perturbation Disturbance Torques $\ell(t)$. --The peak values of total (not perturbation) torques for the nominal 4000-lb laser satellite are shown in fig. A-4, as a function of altitude above the surface of Mars. These results show clearly that gravitational torques are dominant in the altitude range considered. The aerodynamic and solar radiation torque data given in fig. A-4 were derived from 0A0 data prepared by Evans (ref. A5). The atmospheric density variations with Mars altitude were obtained by reducing the Earth density values given by Harris (ref. A6) by two orders of magnitude. This procedure was suggested by Dr. L. R. Koenig of the Douglas Space Sciences Department.

It was assumed that a suitable range for $\ell(t)$ is ±10% of the total gravitational torques; hence, $\ell(t)$ is of the order of magnitude of 10^{-4} lb-ft.

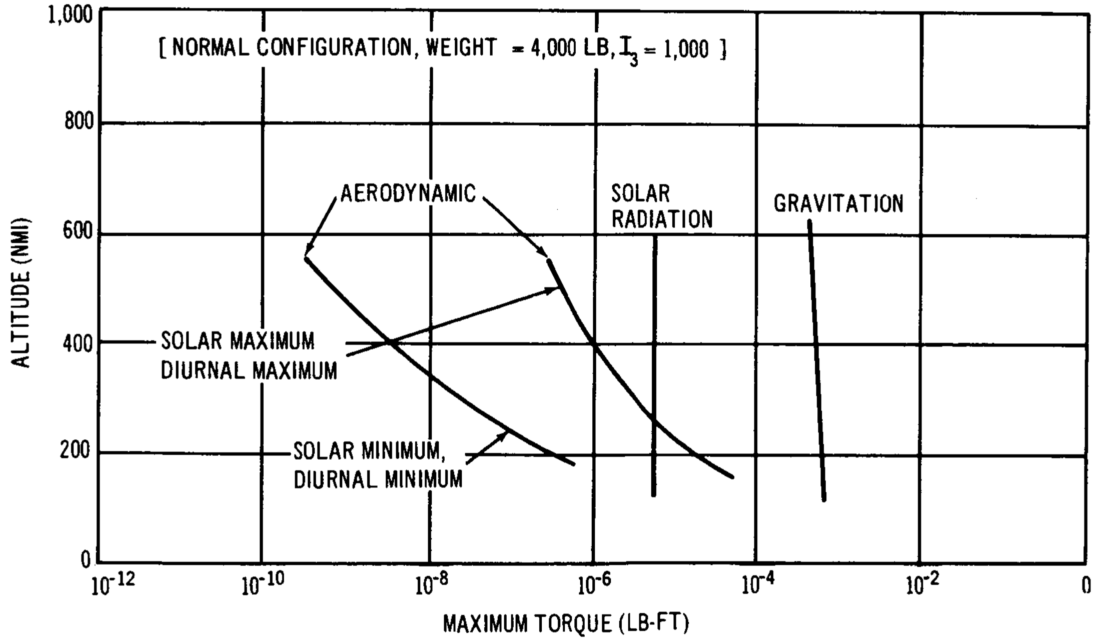


Figure A-4. Maximum Torques Applied to Laser Satellite in Mars Orbit

Range on the Parameter a_9 (t). --The expression for $c_{66}(t)$ in table A-I contains two terms. The first term $\partial L_3/\partial \zeta_3$, is identically zero for external cylindrical symmetry. The second term is identically zero for the assumed mass distribution which makes $I_1 = I_2$. However, under practical circumstances, there will be perturbation torque gradients about the B_3 axis which would appear in the equations of motion as the parameter $c_{66}(t)$ or $a_9(t)$. These would result from the slight departures from external and inertial cylindrical symmetry noted previously, such as would be caused by a radar antenna. [If these departures are significant the equations of motion are once again completely coupled. Here, however, it is assumed that the equation about the B_3 axis is uncoupled and, also, that $a_9(t) \neq 0$. In this way, a solution can be presented for a system that is characterized by both a torque parameter and a plant parameter (see the section on State Space Formulation).]

An upper limit on the gravitational torque gradient, $\partial L_3/\partial \zeta_3$, was calculated. For both the total and gradient gravitational torques, an inertial difference $I_1 - I_2$ of 1/3 I_3 was assumed. Fig. A-5 shows the orbital frequencies for circular orbits around Mars, which vary from a maximum of 0.860 x 10^{-3} rad/sec at sea level to 0.0707 x 10^{-3} rad/sec at the synchronous altitude of 9,000 mi. The trigonometric functions in $a_9(t)$ vary at twice this frequency.

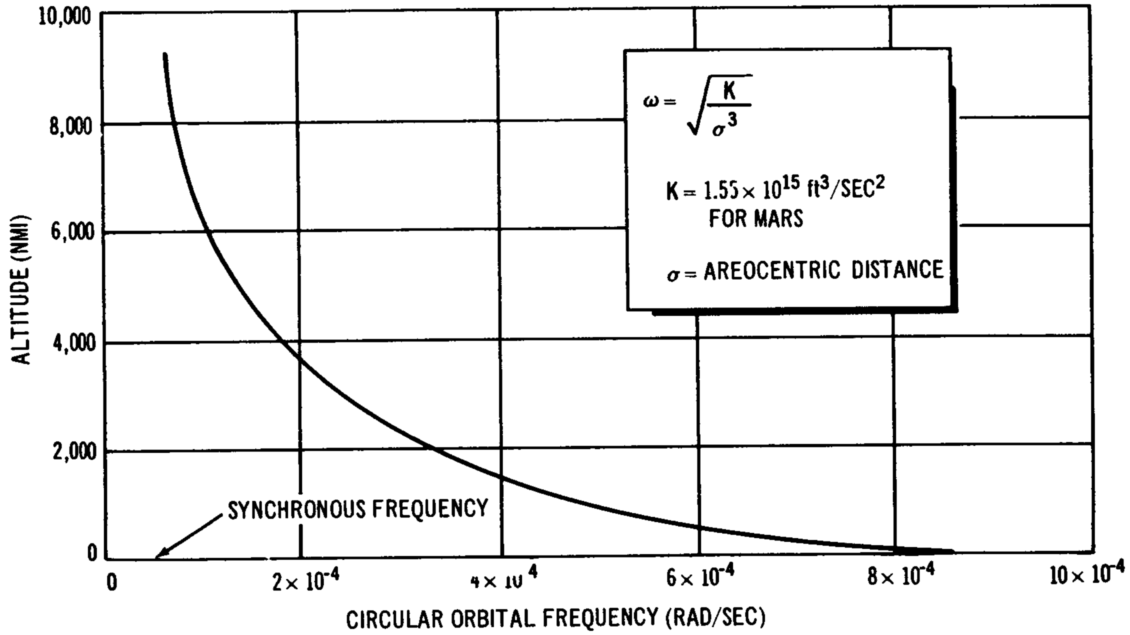


Figure A-5. Orbital Frequencies for Circular Orbits Around Mars

This study used a peak value of $a_9(t)$ of 4×10^{-7} $\sec^{-2}$ varying sinusoidally at 2.0×10^{-4} cps. This is an upper limit for the orbital altitude of 500 nmi. The lower limit of $a_9(t)$ is, of course, zero which corresponds to exact external and inertial symmetry about B_3. The upper limit on $a_9(t)$ is independent of satellite size and is a function only of orbital altitude. Fig. A-4 shows that there is no appreciable increase in gravitational torque below 500 mi. Lower altitudes would be undesirable for operational reasons. The expected range of da_9/dt is twice the frequency values shown in fig. A-5, or from 0.23×10^{-4} to 2.7×10^{-4} cps. The values for $a_9(t)$, b_3, and d_3 that are used in numerical calculations are summarized in table A-III.

TABLE A-III

SUMMARY OF PARAMETER VALUES

Parameter	Units	Value
$a_9(t)$	$\left(\sec^2\right)^{-1}$	$4 \times 10^{-7} \sin\left(4\pi \times 10^{-4}t\right)$
b_3	---	-10^{-6}
c_3	$\left(\text{slug - ft}^2\right)^{-1}$	10^{-3}

Scaling of Equation (A6)

Upon substitution of the values for $a_9(t)$, b_3, and d_3 (in table A-III) into eq. (A6), that equation becomes

$$\ddot{\zeta}(t) + \left[4 \times 10^{-7} \sin\left(4\pi \times 10^{-4}t\right)\right]\zeta(t) = -10^{-6}u(t) + 10^{-3}\boldsymbol{\ell}(t) \qquad \text{(A7)}$$

To avoid small numbers during numerical calculations and during simulations (for example, 4×10^{-7} or 10^{-6}), eq. (A7) is scaled by defining a new time t', where

$$t' = 10^3 t \qquad \text{(A8)}$$

Upon substitution of eq. (A8) into eq. (A7), the latter equation becomes

$$\frac{d^2\zeta(t')}{dt'^2} + \left[0.40 \sin\left(4\pi \times 10^{-7}t'\right)\right]\zeta(t') = -u(t') + 10^3\boldsymbol{\ell}(t') \qquad \text{(A9)}$$

In all discussions in the sequel and in the main body of this report, t' is the time referred to; however, for notational convenience, the prime notation is dropped. In addition, ($\dot{}$) is used for d/dt'. [If $\ell(t)$ is modelled from available physical data the difference between real time, t, and scaled time, t', must be accounted for accordingly.] Eq. (A9) is written, therefore, as

$$\ddot{\zeta}(t) + \left[0.40 \sin (4\pi \times 10^{-7} t)\right] \zeta(t) = -u(t) + 10^{3} \ell(t) \qquad \text{(A10)}$$

and is the basis for the system studied in the main body of this report.

State Space Formulation

Based on the preceding discussions, the document shall concentrate here on the following second-order system

$$\ddot{\zeta}(t) + a(t)\zeta(t) = bu(t) + d\ell(t) \quad , \qquad \text{(A11)}$$

which is equivalent to the scaled version of eq. (A6), eq. (A10) (see table A-IV). Eq. (A11) can be expressed in vector-matrix notation as

$$\underline{\dot{x}}\,(t) = A(t)\,\underline{x}(t) + \underline{b}\,u(t) + \underline{d}\,\ell(t) \qquad \text{(A12)}$$

where $\underline{x}(t)$ is the (2 x 1) state vector[*] $(x_1, x_2)^T = (\zeta, \dot{\zeta})^T$, u and ℓ are scalars,

$$A(t) = \begin{pmatrix} 0 & 1 \\ -a(t) & 0 \end{pmatrix} \qquad \text{(A13)}$$

$$\underline{b} = (0 \quad b)^T \qquad \text{(A14)}$$

and

$$\underline{d} = (0 \quad d)^T \qquad \text{(A15)}$$

Observe that the plant matrix, A(t), varies very slowly in time (table A-IV).

[*] Often, for notational convenience, the explicit dependence of $\underline{x}(t)$, u(t), and $\ell(t)$ on t is omitted; i.e., $\underline{x} = \underline{x}(t)$, etc.

TABLE A-IV

VALUES FOR PARAMETERS IN EQUATION (A11)

Parameter	Value
$a(t)$	$0.40 \sin (4 \times 10^{-7} t)$
b	-1.0
d	1000

It is assumed that $\ell(t)$ [which, from assumptions 2 and 3 in the section on Equations of Motion, is a combination of second-order aerodynamic, incident radiation and gravity gradient effects] is either unknown or partially known. It represents the unknown or partially known environment which acts to disturb the satellite and, as such, is assumed to be a random perturbation disturbance torque. More specifically, $\ell(t)$ is assumed to be a colored (non-white) stochastic process.

For the purposes of analysis, design and synthesis $\ell(t)$ is modelled. Here it is assumed that $\ell(t)$ is modelled by a first-order Markov process; that is to say, $\ell(t) \simeq \xi(t)$ where $\xi(t)$ is described by the following first-order differential equation

$$\dot{\xi}(t) = \alpha(t)\, \xi(t) + \nu(t) \qquad \text{(A16)}$$

where $\nu(t)$ is a white noise process with

$$E\{\nu(t)\} = 0 \qquad \text{(A17)}$$

and

$$E\{\nu(t)\nu(\tau)\} = V(t)\,\delta(t-\tau) \qquad \text{(A18)}$$

With $\ell(t) \simeq \xi(t)$, eq. (A16) and eq. (A12) can be combined; that is to say, eq. (A16) can be augmented to eq. (A12). The resulting <u>augmented state equation</u> is

$$\dot{\underline{x}}_a(t) = A_a(t)\, \underline{x}_a(t) + \underline{b}_a\, u(t) + \underline{f}_a\, \nu(t) \qquad \text{(A19)}$$

where the augmented state vector $\underline{x}_a(t) = (x_1, x_2, \xi)^T$ and

$$A_a(t) = \left(\begin{array}{c|c} A(t) & \underline{d} \\ \hline 0 & \alpha(t) \end{array}\right) \qquad \text{(A20)}$$

$$\underline{b}_a = (\underline{b} \mid 0)^T \qquad \text{(A21)}$$

and

$$\underline{f}_a = (\underset{(2 \times 1)}{\underline{0}} \;\vdots\; 1)^T \qquad \text{(A22)}$$

Finally, it is assumed (see the section on Equations of Motion) that the control u(t) is proportional to the augmented state vector (or to estimates of the augmented state vector)

$$u(t) = \Lambda(t)\underline{x}_a(t) \qquad \text{(A23)}$$

where

$$\Lambda(t) = \left(\lambda_1(t), \lambda_2(t), \lambda_3(t)\right) \qquad \text{(A24)}$$

is a feedback matrix, with components dependent upon the parameters a(t) and $\alpha(t)$.

As discussed in the opening section, fine attitude control is used to bring attitude errors, $x_1(t)$, to within $\pm\epsilon$. Plant changes and excessive disturbance torques could cause $x_1(t)$ to become excessive. The problem is, therefore, to choose $\Lambda(t)$ in eq. (A23) such that $x_1(t)$ remains within its prescribed limits. Solutions to this problem which utilize artificial intelligence techniques are discussed in the main body of this report.

References

A1. Abzug, M. J. : Active Satellite Attitude Control. Guidance and Control of Aerospace Vehicles, ch. 8, C. T. Leondes, Ed., McGraw-Hill Book Co., Inc., 1963, pp. 331-426.

A2. De Bra, D. B. ; and Cannon, R. H. : Momentum Vector Considerations in Wheel-Jet Satellite Control System Design. Paper presented at the ARS Guidance and Control Conf. (Stanford, California), August 1961.

A3. Cannon, R. H. : Basic Response Relations for Some Complete Systems. Control System Mechanization and Analysis, vol. II, WADD TR-60-643, 1960.

A4. Adams, J. J. ; and Chilton, R. G. : A Weight Comparison of Several Attitude Controls for Satellites.. NASA Memo 12-30-58L, 1959.

A5. Evans, W. J. : Aerodynamic and Radiation Disturbance Torques on Satellites Having Complex Geometry. Paper (ARS No. 62-49) presented at the ARS Goddard Memorial Symp. (Washington, D. C.), March 1962.

A6. Harris I. ; and Priester, W. : Relation Between Theoretical and Observational Models of the Upper Atmosphere. NASA X-640-63-145, 1963.

Appendix B

ESTIMATION OF STATES AND PARAMETERS (FEATURE EXTRACTION)

A technique used to obtain sequential estimates of the three states $x_1(t)$, $x_2(t)$, and $\xi(t)$ as well as the two augmented plant parameters a(t) and $\alpha(t)$ is described in this section. The technique represents an application of Kalman filtering to a combined state and parameter estimation problem. While it is not the only technique used to estimate the states and to identify the parameters, it has an advantage over most other techniques in that it is able to directly provide the bootstrapping effects from the parameter identifications to the state estimations, and vice versa. The development below is similar to that given by Kumar (ref. B1) as well as Kopp and Orford (ref. B2); hence many of the details are omitted.

The system of interest in this section is depicted in fig. B-1 and is described by the vector differential equation (fig. 2, Part 2).

$$\dot{\underline{x}}_a(t) = A_a(t)\underline{x}_a(t) + \underline{b}_a u(t) + \underline{f}_a \nu(t) \tag{B1}$$

where

$$\underline{x}_a(t) = (x_1, x_2, \xi)^T \tag{B2}$$

$$A_a(t) = \begin{pmatrix} 0 & 1 & 0 \\ -a(t) & 0 & d \\ 0 & 0 & \alpha(t) \end{pmatrix} \tag{B3}$$

$$\underline{b}_a = (0 \quad b \quad 0)^T \tag{B4}$$

$$\underline{f}_a = (0 \quad 0 \quad 1)^T \tag{B5}$$

and $\nu(t)$ is a scalar white noise process with

$$E\left|\nu(t)\right| = 0 \quad \text{and} \quad E\left|\nu(t)\nu(\tau)\right| = V\,\delta(t - \tau) \tag{B6}$$

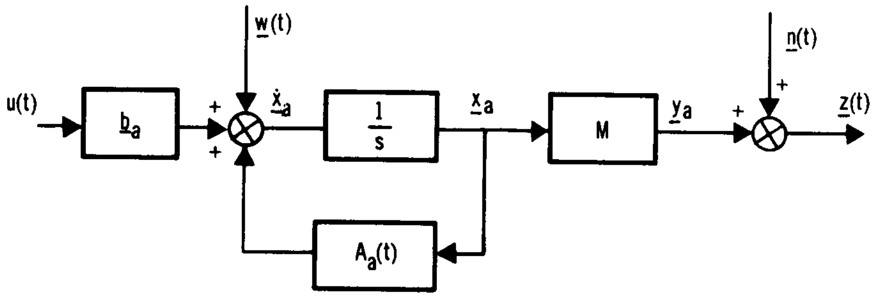


Figure B-1. System with Measurements

and u(t) is a scalar [u(t) = $u_2(t)$ for the system in fig. 2, Part 2 and u(t) = $u^*(t)$ for the on-line-learning control system in fig. 3, Part 2].

For notational convenience, the subscript a is omitted from all vectors and matrices in the rest of this section. Where confusion between the augmented and unaugmented systems is likely to occur, the subscript will be included. Eq. (B1) is written, therefore, as

$$\underline{x}(t) = A(t)\underline{x}(t) + \underline{b}u(t) + \underline{w}(t) \tag{B7}$$

where $\underline{w}(t)$ is a 3 x 1 vector white noise process with

$$E\left|\underline{w}(t)\right| = \underline{0} \quad \text{and} \quad E\left|\underline{w}(t)\underline{w}^T(\tau)\right| = W\delta(t - \tau) \tag{B8}$$

and

$$W = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & V \end{pmatrix} \tag{B9}$$

The observed output, $\underline{y}(t)$ is assumed to be

$$\underline{y}(t) = M\underline{x}(t) \tag{B10}$$

where

$$M = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \tag{B11}$$

that is to say, it is assumed that $x_1(t)$ and $x_2(t)$ are observed. The analysis below is not affected if different sets of observations are made; only M is affected. Finally, it is assumed that the measured values of $x_1(t)$ and $x_2(t)$ are corrupted with measurement noise. The measured output is

$$\underline{z}(t) = \underline{y}(t) + \underline{n}(t) \tag{B12}$$

where $\underline{n}(t)$ is a 2 x 1 vector white noise process, with

$$E\left|\underline{n}(t)\right| = \underline{0} \quad \text{and} \quad E\left|\underline{n}(t)\underline{n}^T(\tau)\right| = R\delta(t - \tau) \tag{B13}$$

and R is the positive definite matrix

$$R = \begin{pmatrix} r_{11} & 0 \\ 0 & r_{22} \end{pmatrix} \tag{B14}$$

The problem is to estimate the elements a(t) and α(t) of the augmented plant matrix A_a(t) and the components of $\underline{x}_a$(t) from a knowledge of the measured output $\underline{z}$(t) and the control u(t). It is assumed in the sequel that the estimation and control problems are separable. The implications of this assumption are discussed in the main body of this report.

Before proceeding with the development of a filter for obtaining the combined estimates, the following definitions are in order:

(1) $\underline{P}_{A_a}(t) \triangleq$ augmented plant parameter vector, where

$$\underline{P}_{A_a}(t) = (a(t), \alpha(t))^T \tag{B15}$$

(2) $\hat{\underline{x}}(t_1|t)\left[\hat{A}(t_1|t);\ \hat{\underline{P}}_{A_a}(t_1|t)\right] \triangleq$ estimated value of $\underline{x}(t)\left[A(t);\ \underline{P}_{A_a}(t)\right]$ at any time $t_1 \geq t$ given the observations $\underline{z}(0), \ldots, \underline{z}(t)$. $\hat{\underline{x}}(t_1|t)$ approximates $\underline{x}(t_1)$ in some suitable sense.

(3) $\tilde{\underline{x}}(t_1|t)\left[\tilde{A}(t_1|t);\ \tilde{\underline{P}}_{A_a}(t_1|t)\right] \triangleq$ error in estimating $\underline{x}\left[A;\ \underline{P}_{A_a}\right]$ at time t_1, where

$$\tilde{\underline{x}}(t_1|t) = \underline{x}(t_1) - \hat{\underline{x}}(t_1|t) \tag{B16}$$

The estimation and error vectors and matrices above are used extensively in the next few paragraphs, where the filter is derived.

A Priori Estimation and Error Equations

Here, equations for $\hat{\underline{x}}_a(t_1|t)$, $\hat{\underline{P}}_{A_a}(t_1|t)$, $\tilde{\underline{x}}_a(t_1|t)$ and $\tilde{\underline{P}}_{A_a}(t_1|t)$ are formulated. The formulation initially is continuous and then the continuous equations are discretized. This is done since the optimal filter, derived in the next paragraph, is discrete.

To begin, equations which describe a(t) and α(t) are augmented to eq. (B7). Using the notation in eq. (B15) these equations combine to yield the following vector differential equation for $\underline{P}_{A_a}(t)$:

$$\dot{\underline{P}}_{A_a}(t) = \Gamma(t)\underline{P}_{A_a}(t) + \underline{\Theta}(t) \tag{B17}$$

where

$$\Gamma(t) = \begin{pmatrix} \gamma_1(t) & 0 \\ 0 & \gamma_2(t) \end{pmatrix} \tag{B18}$$

and $\Theta(t)$ is a 2 x 1 vector white noise process, with

$$E\left|\underline{\Theta}(t)\right| = \underline{0} \text{ and } E\left|\underline{\Theta}(t)\,\underline{\Theta}^T(\tau)\right| = M\,\delta(t-\tau) \tag{B19}$$

The fact that $\underline{\Theta}$(t) is a random process provides a statistical degree of freedom for the estimation of a(t) and α(t).

Next, expressions for $\dot{\hat{\underline{x}}}(t_1|t)$, $\dot{\hat{\underline{P}}}_{Aa}(t_1|t)$, $\dot{\tilde{\underline{x}}}(t_1|t)$, and $\dot{\tilde{\underline{P}}}_{Aa}(t_1|t)$ are obtained. $\hat{\underline{x}}(t_1|t)$ and $\hat{\underline{P}}_{Aa}(t_1|t)$ are unbiased a priori estimates of $\underline{x}(t_1)$ and $\underline{P}_{Aa}(t_1)$, respectively [these estimates do not make use of the measurement $\underline{z}(t_1)$]. From eq. (B7), the unbiased estimate of $\underline{x}(t_1)$ is found according to

$$\dot{\hat{\underline{x}}}(t_1|t) = \hat{A}(t_1|t)\,\hat{\underline{x}}(t_1|t) + \underline{b}u(t_1) \tag{B20}$$

where

$$\hat{A}(t_1|t) = \begin{pmatrix} 0 & 1 & 0 \\ -\hat{a}(t_1|t) & 0 & d \\ 0 & 0 & \hat{\alpha}(t_1|t) \end{pmatrix} \tag{B21}$$

and, from eq. (B17), the unbiased estimate of $\underline{P}_{A_a}(t_1)$ is found according to

$$\dot{\hat{\underline{P}}}_{A_a}(t_1|t) = \Gamma(t_1)\,\hat{\underline{P}}_{A_a}(t_1|t) \tag{B22}$$

Eqs.(B20) and (B22) are the desired equations for $\dot{\hat{\underline{x}}}_a(t_1|t)$ and $\dot{\hat{\underline{P}}}_{A_a}(t_1|t)$, respectively.

To obtain the expression for $\dot{\tilde{\underline{x}}}(t_1|t)$, one proceeds as follows. From eqs. (B16), (B7), and (B20),

$$\dot{\tilde{\underline{x}}}(t_1|t) = \dot{\underline{x}}(t_1) - \dot{\hat{\underline{x}}}(t_1|t)$$

$$\dot{\tilde{\underline{x}}}(t_1|t) = A(t_1)\,\underline{x}\,(t_1) - \hat{A}(t_1|t)\,\hat{\underline{x}}(t_1|t) + \underline{w}(t_1)$$

$$\dot{\tilde{\underline{x}}}(t_1|t) = \left[\tilde{A}(t_1|t) + \hat{A}(t_1|t)\right]\left[\tilde{\underline{x}}\,(t_1|t) + \hat{\underline{x}}\,(t_1|t)\right] - \hat{A}(t_1|t)\,\hat{x}\,(t_1|t) + \underline{w}(t_1)$$

$$\dot{\tilde{\underline{x}}}(t_1|t) = \tilde{A}(t_1|t)\,\hat{\underline{x}}(t_1|t) + \hat{A}(t_1|t)\,\tilde{\underline{x}}(t_1|t) + \underline{w}(t_1) \tag{B23}$$

In eq. (B23), the second-order effect $\tilde{A}(t_1|t)\,\tilde{\underline{x}}(t_1|t)$ has been neglected, and $\tilde{A}(t_1|t)$ is [from eqs. (B16), (B3) and (B21)]

$$\tilde{A}(t_1|t) = \begin{pmatrix} 0 & 0 & 0 \\ -\tilde{a}(t_1|t) & 0 & 0 \\ 0 & 0 & \tilde{\alpha}(t_1|t) \end{pmatrix} \tag{B24}$$

Proceeding in a similar manner for $\dot{\tilde{\underline{P}}}_{A_a}(t_1|t)$, it follows that

$$\dot{\tilde{\underline{P}}}_{A_a}(t_1|t) = \Gamma(t_1)\,\tilde{\underline{P}}_{A_a}(t_1|t) + \underline{\Theta}\,(t_1) \tag{B25}$$

Eqs. (B23) and (B25) are the desired expressions for $\dot{\tilde{\underline{x}}}(t_1|t)$ and $\dot{\tilde{\underline{P}}}_{A_a}(t_1|t)$, respectively.

Next, a new vector $\underline{q}(t)$ is introduced; it is defined as

$$\underline{q}(t) = \left(\underline{x}(t),\ \underline{P}_{A_a}(t)\right)^T \tag{B26}$$

Eqs. (B20) and (B22) combine to give an equation for $\dot{\hat{\underline{q}}}(t_1|t)$ and eqs. (B23) and (B25) combine to give an equation for $\dot{\tilde{\underline{q}}}(t_1|t)$:

$$\dot{\hat{\underline{q}}}(t_1|t) = Q(t_1)\,\hat{\underline{q}}(t_1|t) + \underline{L}\,u(t_1) \tag{B27}$$

and

$$\dot{\tilde{\underline{q}}}(t_1|\, t) \;=\; Q^*(t_1)\, \tilde{\underline{q}}(t_1|\, t) \;+\; \underline{\phi}^*(t_1) \qquad \text{(B28)}$$

Eqs. (B27) and (B28) are the continuous (augmented) a priori estimation and error equations, and, in these equations:

$$Q(t_1) \;=\; \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ -\hat{a}(t_1|\, t) & 0 & d & 0 & 0 \\ 0 & 0 & \hat{\alpha}(t_1|\, t) & 0 & 0 \\ 0 & 0 & 0 & \gamma(t_1) & 0 \\ 0 & 0 & 0 & 0 & \gamma(t_2) \end{pmatrix} \qquad \text{(B29)}$$

$$Q^*(t_1) \;=\; \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ -\hat{a}(t_1|\, t) & 0 & d & -\hat{x}_1(t_1|\, t) & 0 \\ 0 & 0 & \hat{\alpha}(t_1|\, t) & 0 & \hat{\xi}(t_1|\, t) \\ 0 & 0 & 0 & \gamma_1(t_1) & 0 \\ 0 & 0 & 0 & 0 & \gamma_2(t_1) \end{pmatrix} \qquad \text{(B30)}$$

$$\underline{L} \;=\; \begin{pmatrix} 0 & b & 0 & 0 & 0 \end{pmatrix}^T \qquad \text{(B31)}$$

and

$$\underline{\phi}^*(t_1) \;=\; \begin{pmatrix} 0 & 0 & \nu(t_1) & \theta_1(t_1) & \theta_2(t_1) \end{pmatrix}^T \qquad \text{(B32)}$$

Observe that, because of the appearance of the states $\hat{x}_1(t_1\,|\,t)$ and $\hat{\xi}(t_1|t)$ in $Q^*(t_1)$, the a priori error equation is nonlinear; hence, as will be seen in the next paragraph, the optimal filter is nonlinear.

Finally, eqs. (B27) and (B28) are discretized, for use in the next subsection. The resulting discrete (augmented) a priori estimation and error equations are:

$$\underline{\hat{q}}(n+1|n) = \Phi(n+1,\ n)\,\underline{\hat{q}}(n|n) + \underline{\Delta}(n+1, n)\,u(n) \qquad \text{(B33)}$$

and

$$\underline{\tilde{q}}(n+1|n) = \Phi^{*}(n+1,\ n)\,\underline{\tilde{q}}(n|n) + \underline{\varphi}^{*}{}_{d}(n+1,\ n). \qquad \text{(B34)}$$

In these equations:

$$\underline{q}(n+1\ |\ n) \triangleq \underline{q}\left[(n+1)T_s \middle| nT_s\right] \quad n = 0, 1, \ldots, \qquad \text{(B35)}$$

$$\Phi(n+1,\ n) = e^{Q(n)T_s} \qquad \text{(B36)}$$

$$\underline{\Delta}(n+l,\ n) = \int_{nT_s}^{(n+1)T_s} \Phi(n+1,\ \tau)\,\underline{L}\,d\tau \qquad \text{(B37)}$$

$$\Phi^{*}(n+l,\ n) = e^{Q^{*}(n)T_s} \qquad \text{(B38)}$$

and

$$\underline{\varphi}^{*}{}_{d}(n+l,\ n) = \int_{nT_s}^{(n+1)T_s} \Phi^{*}(n+l,\ \tau)\ \underline{\varphi}^{*}(\tau)\ d\tau \qquad \text{(B39)}$$

In the derivations of eqs. (B33) and (B34), it was assumed that $Q(t_l)$ and $Q^{*}(t_l)$ are piecewise constant over each sampling interval.

For the purposes of simulation, it is interesting to observe that $\underline{\varphi}_d^{*}$ (n+1, n), in eq. (B39) is a zero-mean vector random-sequence with a covariance equal to CT_s, where

$$E\{\underline{\varphi}^{*}(t)\ \underline{\varphi}^{*T}(\tau)\} = C\,\delta(t-\tau) \qquad \text{(B40)}$$

and

$$C = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & V & 0 & 0 \\ 0 & 0 & 0 & \mu_{11} & 0 \\ 0 & 0 & 0 & 0 & \mu_{22} \end{pmatrix} \qquad \text{(B41)}$$

The expression for C is derived from eqs. (B32), (B19), and (B6).

A Posteriori Estimation and Error Equations

Here, the measurement taken at $t = t_1$ is incorporated into the estimation equations; the result is an optimal sequential-estimator. The derivation of the optimal estimator is well-documented in the literature (refs. B1 and B2, for example); hence, in this subsection, the optimal estimator is only summarized. Its structure is depicted (fig. B-2), with

$$M^* = \left(M \,\vdots\, \overset{(2x3)}{0}\right) \qquad \text{(B42)}$$

and M is given in eq. (B11)

$$K^*(n) = \left[\Phi^*(n, n-1)\sigma(n-1)\Phi^{*T}(n, n-1) + CT_s\right]M^{*T} \cdot$$
$$\text{(B43)}$$
$$\left[M^*\left[\Phi^*(n, n-1)\sigma(n-1)\Phi^{*T}(n, n-1) + CT_s\right]M^{*T} + RT_s\right]^{-1}$$

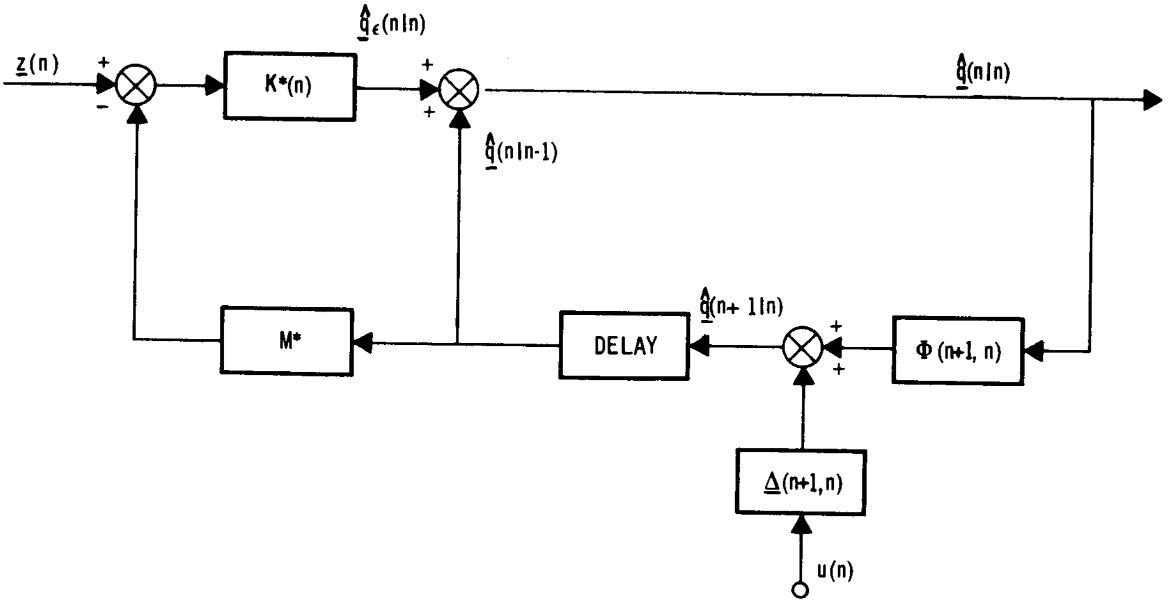
where Φ^*, C and R are given in eqs. (B38), (B41) and (B14), respectively, and the covariance σ is given as

$$\sigma(n) = \left[I - K^*(n)M^*\right]\left[\Phi^*(n, n-1)\sigma(n-1)\Phi^{*T}(n, n-1) + CT_s\right], \text{(B44)}$$

$\hat{\underline{q}}(n+1|n)$ is given in eq. (B33), and $\hat{\underline{q}}(n|n)$ is the optimal a posteriori estimate.

It must be observed, that the system in fig. B-1 ties into the filter in fig. B-2 in two places: $\underline{z}(n)$ and u(n). In addition, the filter is nonlinear, since Q*(n) is a function of $\hat{\underline{q}}(n+1|n)$; hence, the convergence properties of the filter are subject to question. To-date, to the best knowledge of the author of this report, the convergence properties have not been studied. What usually is done is that a linear region of operation is found, through trial and error, and the filter is operated only within this linear region. Any departure from the conditions required to maintain the filter operating in the linear region may lead to instabilitieṣ and, therefore, to a filter which is unacceptable for the new set of operating conditions.

It is not within the scope of the present study to investigate the convergence properties of this or any other nonlinear or linear estimator; hence, the approach was to determine a linear operating region for the filter in fig. B-2, an operating region within which it is possible to obtain estimates for $\underline{x}_a(t)$ and $\underline{P}_{A_a}(t)$. The estimates are required for the on-line-learning control system in fig. 3, Part 2. Simulation results for the filter are presented and discussed in the section on State and Parameter Estimation for Control in Part 2.



$\underline{z}(n)$
+
-
K*(n)
$\hat{\underline{q}}_\epsilon(n|n)$
+
+
$\hat{\underline{q}}(n|n)$
$\hat{\underline{q}}(n|n-1)$
M*
DELAY
$\hat{\underline{q}}(n+1|n)$
+
+
Φ (n+1, n)
Δ(n+1,n)
u(n)

Figure B-2. Discrete Optimal Filter Structure

References

B1. Kumar, K. S. P.: On the Identification of Control Systems. Dissertation, Purdue University, Jan. 1964.

B2. Kopp, R. E.; and Orford, R. J.: Linear Regression Applied to System Identification for Adaptive Control Systems. AIAA Journal, vol. 1, no. 10, Oct. 1963.

Appendix C

DESIGN OF NOMINAL (SUB-OPTIMAL) CONTROLS FOR FIXED PARAMETERS

The sub-optimal control $u_2^*(t)$, which serves as the initial control choice for $u^*(t)$ in the on-line-learning control system (fig. 3, Part 2) is obtained here for the off-line augmented system in fig. C-1 Sub-optimal control $u_2^*(t)$ is found by optimizing a cost function which is quadratic in the attitude error (x_1) and control (u_2) and is obtained for specific combinations of a(t) and $\alpha(t)$. Specific values of a(t) and $\alpha(t)$ are denoted a and α, respectively; hence, for specific values of these parameters $A_a(t) = A_a$ in fig. C-1.

$\underline{x}_a(t)$ is a vector stochastic process, since it depends upon the random process $\nu(t)$. For every motion, a random variable V is defined as

$$V = \frac{1}{2} \int_0^{T_1} \left[q\, x_1^2(t) + r\, u_2^2(t)\right] dt \qquad \text{(C1)}$$

The constants q and r, which are both positive, penalize attitude errors and control effort, respectively. The performance index is now defined to be the expected value of V (with respect to the probability distribution of $\nu(t)$). Control $u_2^*(t)$ is found by minimizing $E\{V\}$.

As stated, this is a problem in stochastic optimal control. An alternate approach to the suggested solution of minimizing $E\{V\}$ is provided by the Separation Theorem (ref. C1) which states, in effect, that the solution to the above problem can be obtained in the following two steps (ref. C2):

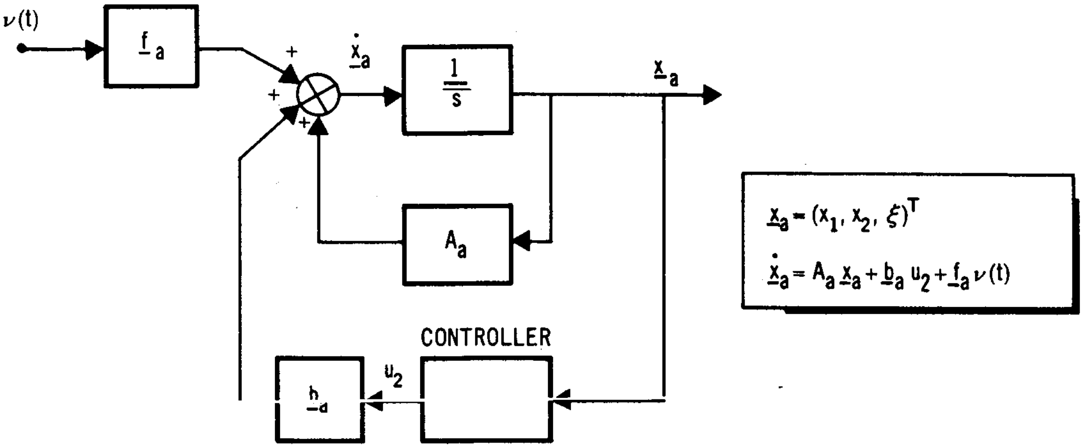
(1) Estimate $\underline{x}_a(t)$ by filtering theory (as in Appendix B).

(2) Obtain the optimal control $u_3^*(t)$ from noise-free regulator theory.

The expression for $u_2^*(t)$ is then found from items 1 and 2 according to the Separation Theorem, as illustrated below.

Optimal Control for Noise-Free Regulator

To find $u_3^*(t)$, the optimal control for the noise-free regulator, one assumes noise effects are absent in fig. C-1 and minimizes V in eq. (C1) For the noise-free regulator problem V and not $E\{V\}$ is the performance



$\nu(t)$
$\underline{f}_a$
+
+
+
$\dot{\underline{x}}_a$
$\frac{1}{s}$
$\underline{x}_a$
A_a
CONTROLLER
u_2
$\underline{b}_a$
$\underline{x}_a = (x_1, x_2, \xi)^T$
$\dot{\underline{x}}_a = A_a \underline{x}_a + \underline{b}_a u_2 + \underline{f}_a \nu(t)$

A_a, $\underline{b}_a$ and $\underline{f}_a$ ARE DEFINED IN APPENDIX A.

Figure C-1. Off-line Augmented System

index, since $\nu(t)$ is zero for all t. The solution for $u_3^*(t)$ is, as is well known (see ref. C2, for example),

$$u_3^*(t) = \Lambda(t)\, \underline{x}_a(t) \qquad \text{(C2)}$$

where the optimal gain matrix, $\Lambda(t)$, is

$$\Lambda(t) = -\frac{1}{r} \underline{b}_a^T P(t) \qquad \text{(C3)}$$

and P(t), which is a positive-definite, symmetric matrix (3x3), is the solution of the matrix Riccati equation

$$-\frac{dP(t)}{dt} = P(t) A_a + A_a^T P(t) - P(t) \underline{b}_a \frac{1}{r} \underline{b}_a^T P(t) + Q \qquad \text{(C4)}$$

where $P(T_1) = 0$ and

$$Q = \begin{pmatrix} q & o & o \\ o & o & o \\ o & o & o \end{pmatrix} \tag{C5}$$

In the sequel, it is assumed that the transient region for P(t) is a small fraction of the total interval of control $0 \le t \le T_1$. In such cases (see refs. C2 and C3, for example), the response of the system for $t \ll T_1$ is determined entirely by the steady-state values of the gains. In addition, when $T_1 \to \infty$ the gains assume their steady-state values throughout the entire interval $0 \le t \le T_1$. Since our ultimate objective is to obtain a time-invariant nominal control, only the steady-state optimal gains are computed, provided, of course, the steady-state gains exist. Since the system is not completely controllable, there is no guarantee that the steady-state gains do exist (see refs. C2 and C4, for example). A direct simulation of eq. (C4), for values of interest, of a(t) and $\alpha(t)$, was performed. For a random sampling from the sets of values for a(t) and $\alpha(t)$, convergence to a steady-state occurred; hence, it is assumed in the sequel that the steady-state gains exist and can be found by setting the right-hand side of eq. (C4) equal to zero. The resulting equations are

$$\left.\begin{aligned}
&-2a\, p_{21} - \frac{b^2}{r} p_{21}^2 + q = 0 \\
&p_{11} - a\, p_{22} - \frac{b^2}{r} p_{21}\, p_{22} = 0 \\
&dp_{21} + \alpha p_{13} - a\, p_{23} - \frac{b^2}{r} p_{21}\, p_{23} = 0 \\
&2p_{21} - \frac{b^2}{r} p_{22}^2 = 0 \\
&dp_{22} + \alpha p_{23} + p_{13} - \frac{b^2}{r} p_{22}\, p_{23} = 0 \\
&2dp_{23} + 2\,\alpha p_{33} - \frac{b^2}{r} p_{23}^2 = 0
\end{aligned}\right\} \tag{C6}$$

For purposes of this paper, it is not necessary to solve for all six unknowns p_{11}, p_{22}, p_{33}, p_{21}, p_{23}, and p_{13}. This is because $\Lambda(t)$ in eq. (C3) is only a function of p_{21}, p_{22}, and p_{23}; that is to say

$$\Lambda = -\frac{b}{r}\left(p_{21}\ p_{22}\ p_{23}\right) \tag{C7}$$

Upon solving eq. (C6) for p_{21}, p_{22}, and p_{23} and substituting eq. (C7) into eq. (C2), $u_3^*(t)$ becomes

$$u_3^*(t) = \lambda_1\ x_1(t) + \lambda_2\ x_2(t) + \lambda_3\ \xi(t) \tag{C8}$$

where

$$\lambda_1 = -\left(a - \sqrt{a^2 + \rho}\right) \tag{C9a}$$

$$\lambda_2 = \sqrt{2\left(\sqrt{a^2 + \rho} - a\right)} \tag{C9b}$$

$$\lambda_3 = \frac{-1000\left[\alpha\sqrt{2\left(\sqrt{a^2 + \rho} - a\right)} - \sqrt{a^2 + \rho} + a\right]}{\alpha^2 + \sqrt{a^2 + \rho} - \alpha\sqrt{2\left(\sqrt{a^2 + \rho} - a\right)}} \tag{C9c}$$

and*

$$\rho = q/r \tag{C10}$$

The nominal control, $u_2^*(t)$, for the on-line-learning control system in fig. 3 (Part 2) is found, from the Separation Theorem, as

$$u_2^*(t) = \Lambda \hat{\underline{x}}_a\ (t\,|\,t). \tag{C11}$$

*The values for b and d in table A-IV were incorporated into the solution of eq. (C6).

To completely determine λ_1, λ_2, and λ_3 in eq. (C9), the weighting factor ρ must be specified. This can be accomplished in a number of ways. For example, ρ can be chosen such that the sensitivity of λ_1, λ_2, or λ_3 to variations in the parameter a (or α) is small. This type of choice for ρ might be useful in the implementation of control situations, since, by choosing ρ suitably, it might be possible to make the nominal control relatively insensitive to variations in a (or α). On the other hand, ρ can be chosen such that for specific values of a and α attitude errors remain within ±0.20 arcsec in some probabilistic sense. This latter approach was adopted and is elaborated upon further in the next section.

Computation of Weighting Factor

It is desired that attitude errors remain within ±0.2 arcsec (see Appendix A) in spite of a certain amount of uncertainty about the perturbation disturbance torques $\ell(t)$. In Appendix A, a range for $\ell(t)$ was specified; that is, $|\ell(t)| \leq 10^{-4}$ lb-ft. The certainty that $\ell(t)$ is in the assumed range can be expressed in terms of the probability.

$$\Pr\{|\ell(t)| \leq 10^{-4} \text{ lb-ft}\} = p_\ell. \tag{C12}$$

If it is very certain that $\ell(t)$ is in the range, p_ℓ is very close to unity. On the other hand, if it is not so certain that $\ell(t)$ is in this range p_ℓ will not be so close to unity. It shall be assumed that at each instant of time, $t = t_1$, $\ell(t_1)$ is normally distributed. The standard deviation, σ_ℓ, of $\ell(t_1)$, is shown in fig. C-2 as a function of p_ℓ [see table 26.5 in ref. C5].

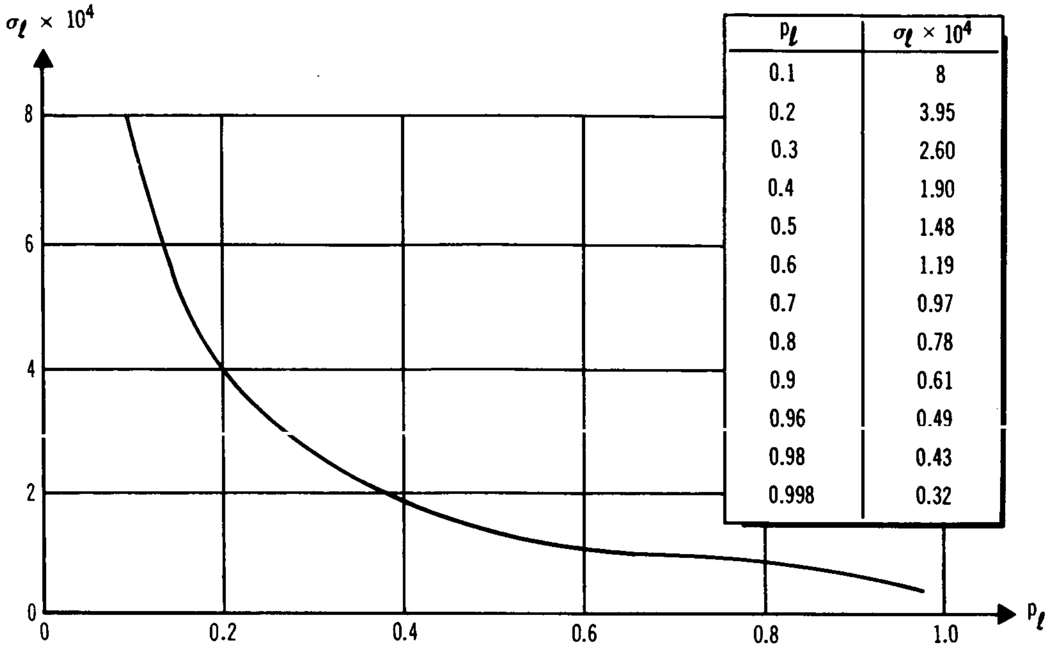
The uncertainty in $\ell(t)$ is reflected in an uncertainty about attitude errors, $x_1(t)$. It is known, for example, that $u_2^*(t)$ drives the expected value of $x_1(t)$ to zero in the fixed parameter case. However, if $x_1(t)$ varies by large amounts about the zero mean, due to poor knowledge about $\ell(t)$, then fine attitude control is not achieved satisfactorily. As a measure of the variation of $x_1(t)$ the standard deviation σ_{x_1} can be computed. To this end, it is assumed that at each instant of time, $t = t_1$, $x(t_1)$ is normally distributed.

In the following section, σ_{x_1} is computed as a first step in the design of the weighting factor ρ. The ultimate objective is to design ρ and then compare $\Pr\{|x_1| \leq 0.2$ arcsec' with $\Pr\{|\ell| \leq 10^{-4}$ lb-ft$\}$ for a number of designs.

Computation of σ_{x_1}.--Here it is assumed that $\alpha<0$ and, in addition†, that $u_2^*(t) = \Lambda \underline{x}_a(t)$; hence, the system in fig. C-1 can be represented as in fig. C-3, where

$$H(s) = \frac{(1000 - \lambda_3)}{(s + |\alpha|)\ [s^2 + \lambda_2 s + (a + \lambda_1)]} \tag{C13}$$

†The effects of state estimation errors are not known a priori; they are experimentally determined in Section 3 (Part 2) and depend upon the specific estimator design.



p_ℓ	$\sigma_\ell \times 10^4$
0.1	8
0.2	3.95
0.3	2.60
0.4	1.90
0.5	1.48
0.6	1.19
0.7	0.97
0.8	0.78
0.9	0.61
0.96	0.49
0.98	0.43
0.998	0.32

Figure C-2. Standard Deviation of Disturbance Torques as a Function of the Certainty that $|\ell(t_1)| \leq 10^{-4}$ lb-ft

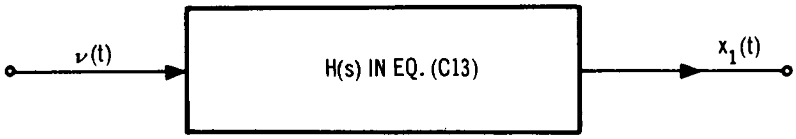


Figure C-3. Relation Between Attitude Error and White Noise Input

For stationary linear systems (as in fig. C-3)

$$\sigma_{x_1}^2 = \frac{1}{2\pi} \int_{-\infty}^{\infty} S_{x_1 x_1}(\omega)\, d\omega \qquad \text{(C14)}$$

where

$$S_{x_1 x_1}(\omega) = S_{\nu\nu}(\omega)\, |H(j\omega)|^2 \qquad \text{(C15)}$$

and

$$S_{\nu\nu}(\omega) = V \qquad \text{(C16)}$$

since $E\{\nu(t)\ \nu(\tau)\} \triangleq V\delta(t - \tau)$. $\sigma_{x_1}^2$ in eq. (C14) is evaluated by a contour integration; the algebra, though lengthy, is straightforward. The resulting expression for $\sigma_{x_1}^2$ is

$$\sigma_{x_1}^2 = \frac{(1000 - \lambda_3)^2 V}{4\,|\alpha|\,\zeta\ \omega_n^3} \cdot \frac{\omega_n^2\,(2\zeta\omega_n + |\alpha| - 4\alpha\zeta^2) + \alpha^3}{\left[(\alpha^2 + \omega_n^2)^2 - (2\zeta\omega_n\alpha)^2\right]} \qquad \text{(C17)}$$

where [see eq. (C13)]

$$\lambda_2 \triangleq 2\zeta\omega_n \qquad \text{(C18)}$$

and

$$\lambda_1 \triangleq \omega_n^2 - a \qquad \text{(C19)}$$

Observe that for small values of a and α, eq. (C17) becomes

$$\frac{\sigma_{x_1}^2}{V} \cong \frac{(1000 - \lambda_3)^2}{2\,|\alpha|\,\omega_n^4} \qquad \text{(C20)}$$

One approach to the design of ρ might be to constrain $\sigma_{x_1}^2$ in eq. (C17); however, due to the very complicated relations between λ_1, λ_2, and λ_3 and ρ [see eq. (C9)] this approach was not adopted. An alternate approach is described in the following section.

Worst Case Design for Weighting Factor ρ.--Here ν(t) in fig. C-3 is replaced by $\pm 3\sqrt{V}$. The resulting expression for $X_1(s)$ is

$$X_1(s) = \frac{\pm 3\sqrt{V}\,(1000 - \lambda_3)}{s(s + |\alpha|)\left[s^2 + \lambda_2 s + (a + \lambda_1)\right]} \qquad \text{(C21)}$$

The magnitude of the steady-state attitude error in eq. (C21) is now constrained to be less than or equal to 0.2 arcsec (ε). This requirement serves as the basis for the design of ρ and results in the following inequality

$$\left| \lim_{s \to o} \; s\, X_1(s) \right| \leq \epsilon \qquad \text{(C22)}$$

or

$$\left| \frac{3\sqrt{V}}{|\alpha|} \cdot \frac{(1000 - \lambda_3)}{(a + \lambda_1)} \right| \leq \epsilon \qquad \text{(C23)}$$

Upon substitution of eqs. (C9a) and (C9c) into eq. (C23), and after a bit of algebra, eq. (C23) becomes

$$r^4 + 2\,C_2\,r^3 + (2C_1^2 + C_2^2)r^2 + 2C_1^2\,(C_2 - 2\alpha^4)\,r + C_1^4 \leq 0 \qquad \text{(C24)}$$

where

$$r = \rho + a^2 \qquad \text{(C25)}$$

$$C_1 = \frac{3000\sqrt{V}}{\epsilon}\left|\frac{\alpha^2 + a}{\alpha}\right| \qquad \text{(C26)}$$

and

$$C_2 = \alpha^2\,(2a + \alpha^2) - 2\,C_1 \qquad \text{(C27)}$$

To solve eq. (C24), V in eq. (C26) is required. Note, also, that V is needed in eq. (C17) for $\sigma_{x_1}^2$; thus, before discussing the solution of eq. (C24), an expression is obtained for V.

Expression for V. --Here V is obtained in terms of σ_ℓ and α. It is assumed that $\ell(t) \simeq \xi(t)$, where

$$\dot{\xi}(t) = \alpha\xi(t) + \nu(t) \tag{C28}$$

and

$$E\{\nu(t)\nu(\tau)\} = V\delta(t - \tau) \tag{C29}$$

and, therefore, that $\sigma_\xi \simeq \sigma_\ell$. From the solution of eq. (C28), it follows that

$$E\{\xi(t_1)\xi(t_2)\} = \frac{V}{2|\alpha|} \quad e^{-|\alpha|\,|t_2 - t_1|} \tag{C30}$$

hence,

$$\sigma_\xi^2 = \frac{V}{2|\alpha|} \tag{C31}$$

which, from the assumption that $\sigma_\xi \simeq \sigma_\ell$, leads to the following expression for V:

$$V = 2|\alpha|\sigma_\ell^2 \tag{C32}$$

Eq. (C32) is the desired result. Note that, from eq. (C32) and fig. C-2 it is possible to relate V to Pr$\{|\ell| \leq 10^{-4}$ lb-ft$\}$.

Solution For Weighting Factor ρ. --The solution for ρ proceeds in two steps:

(1) Selection of σ_ℓ, based upon knowledge of $\ell(t)$,

(2) For various combinations of a and α:

(a) The real roots of eq. (C24) are derived

(b) For the roots in eq. (C24), ρ in eq. (C25) is derived [if there is more than one real root, the smallest root, is retained since control effort is proportional to ρ].

Eq. (C24) was programmed. For all combinations of a and α considered $[|a| \leq 0.40$ and $-1 \leq \alpha \leq -0.10]$ during this study, eq. (C24) had two real roots. One root was slightly smaller than C_1 and the other root was slightly larger than C_1 [observe, from eqs. (C26) and (C27), that for small values of a and α $C_2 \simeq -2C_1$ and, that eq. (C24) reduces approximately to $(r - C_1)^4 = 0$]. Figs. C-4, C-5, and C-6 summarize ρ as a function of a and α for three designs: (1) $p_\ell = 0.75$, (2) $p_\ell = 0.96$, and (3) $p_\ell = 0.998$, respectively. The surface in each of these figures can be viewed as the nominal feedback parameter switching surface, since λ_1, λ_2, and λ_3 are functions only of ρ for given values of a and α. It must be observed that as knowledge of $\ell(t)$ becomes less precise ρ increases; hence, more control effort is exerted as p_ℓ becomes smaller. In addition, for small values of α and large negative values of a a great deal of control effort must be exerted. This is due to the fact that the open-loop system is unstable for negative values of a(t).

Once a value for σ_ℓ is chosen σ_{x_1} and, subsequently, $Pr\{|x_1| \leq 0.2$ arcsec$\}$ can be computed for specific values of a and α. For small values of a and α [see eqs. (C18), (C19), (C9a) and (C9b)], it is straightforward to show that

$$\zeta \simeq 0.707 \quad \text{(C33a)}$$

$$\omega_n \simeq \rho^{1/4} \text{ rad/sec} \quad \text{(C33b)}$$

and that the approximation for $\sigma_{x_1}{}^2$ in eq. (C20) is a good one. After σ_{x_1} is computed from eq. (C20), the expression

$$P_r\{|x_1| \leq 0.2 \text{ arcsec}\} = P_r\{|x_1| \leq 0.97 \times 10^{-6} \text{ radians}\} \quad \text{(C34)}$$

is found from table 26.5 in ref. C5 (see fig. C-7).

An example that illustrates the behavior of $Pr\{|x_1| \leq 0.2$ arcsec$\}$ against $Pr\{|\ell| \leq 10^{-4}$ lb-ft$\}$ appears in fig. C-8. The tradeoff between the three designs must be noted. If it is not very certain that $|\ell| \leq 10^{-4}$ lb-ft., as in design 1, a large amount of control must be used (see fig. C-9; control is proportional to ρ). However, because so much control is used, attitude errors are contained to within ±0.2 arcsec with a probability that is very close to unity, even if the actual disturbance torques are less than or equal to $\pm 10^{-4}$ lb -ft 50% of the time. On the other hand, if it is very certain that $|\ell| \leq 10^{-4}$ lb-ft, as in design 3, about one-third of the previous control is used. However, if the assumption is wrong, that is if the disturbance torques are less than or equal to $\pm 10^{-4}$ lb-ft less than 99.8% of the time, attitude errors increase (in probability) very rapidly. For example, if $p_\ell = 0.80$, $Pr\{|x_1| \leq 0.2$ arcsec$\} = 0.96$; if $p_\ell = 0.50$, $Pr\{|x_1| \leq 0.2$ arcsec$\} = 0.73$.

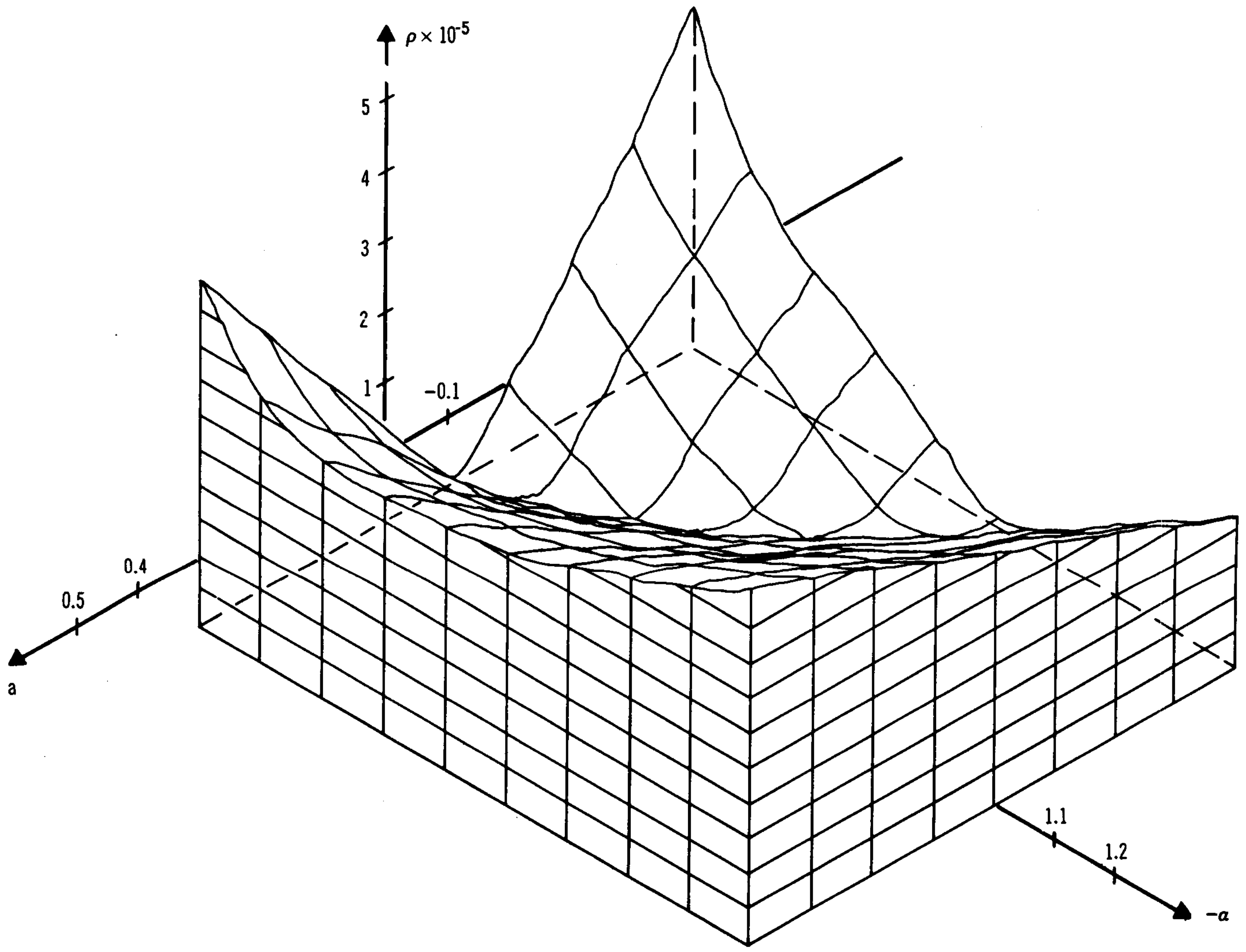


Figure C-4. ρ as a Function of a and α for Pr $\{ |\boldsymbol{\ell}| \leq 10^{-4}$ lb-ft$\} = 0.75$

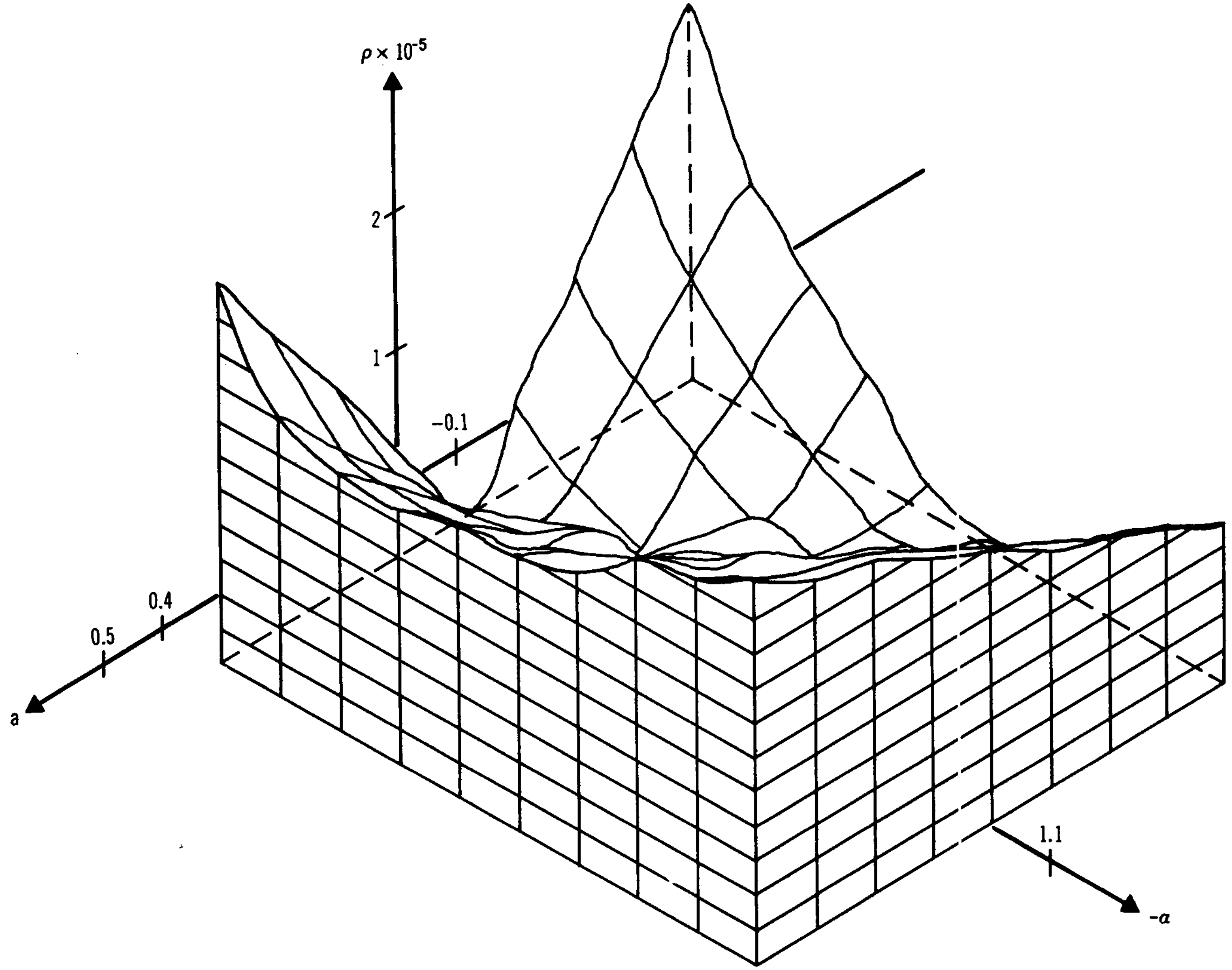


Figure C-5. ρ as a Function of a and α for Pr $\{ \, |\ell| \leq 10^{-4}$ lb-ft $\} = 0.96$

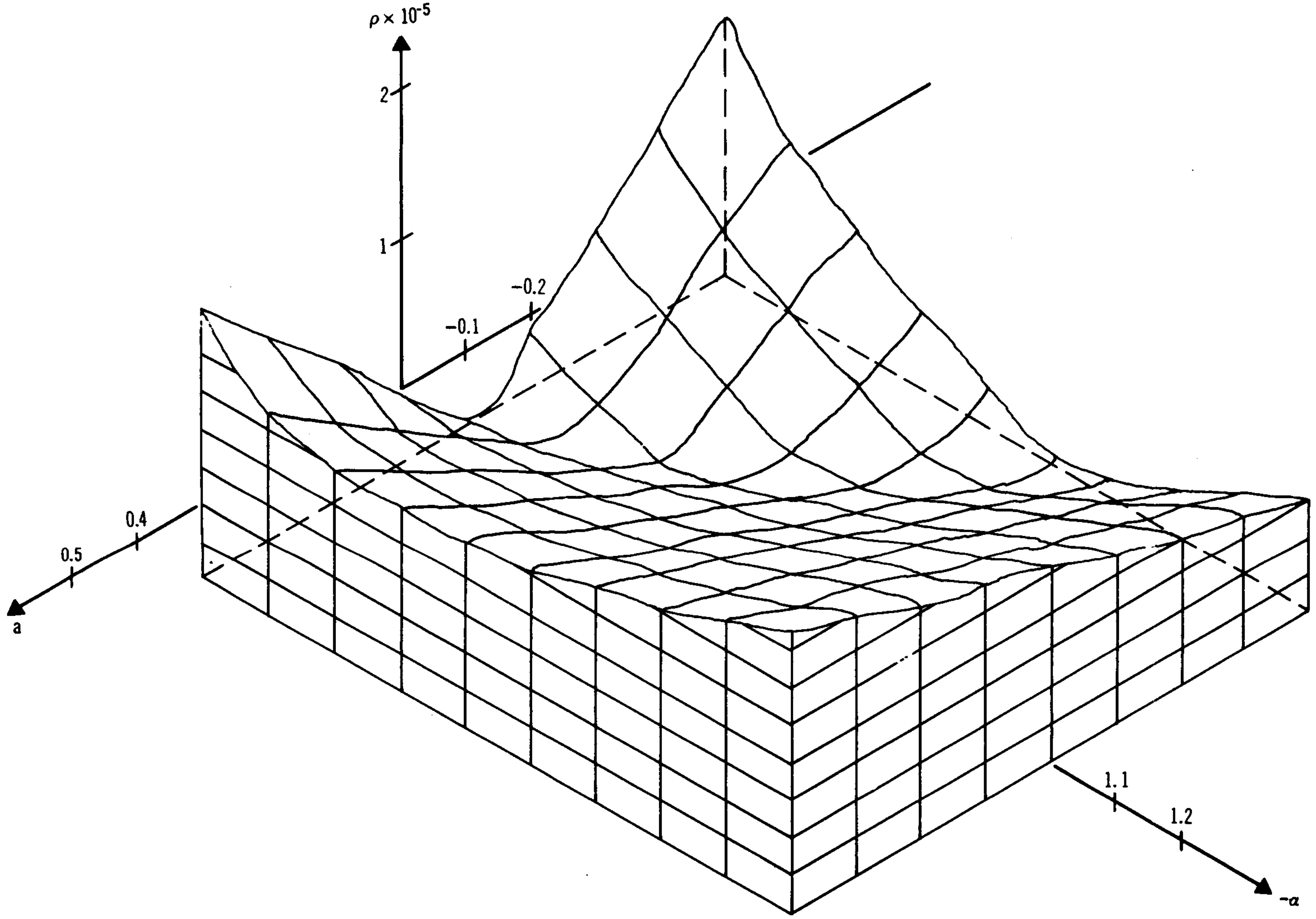


Figure C-6. ρ as a Function of a and α for Pr { $|\ell| \leq 10^{-4}$ lb-ft } = 0.998

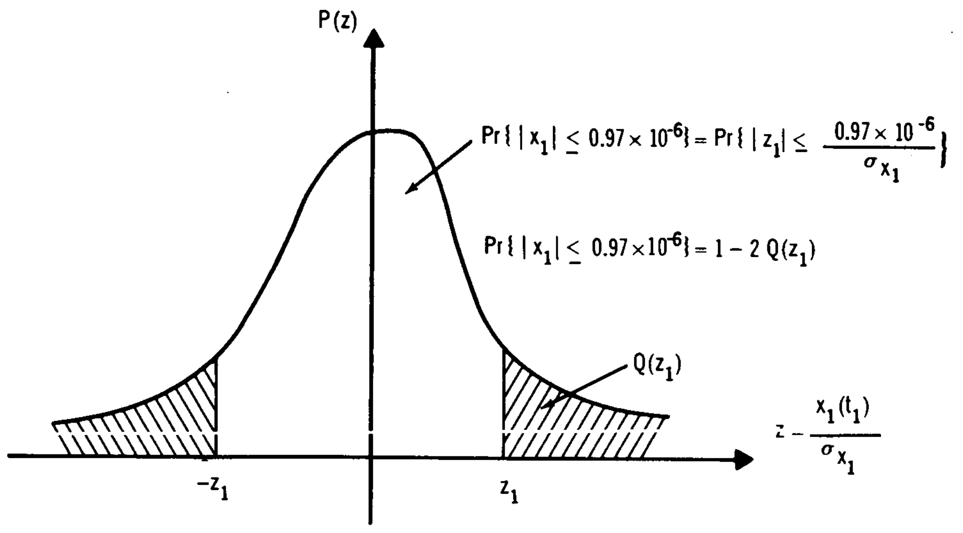

P(z)
Pr{ | x$_1$ | ≤ 0.97 × 10^{-6}} = Pr{ | z$_1$ | ≤ $\frac{0.97 \times 10^{-6}}{\sigma_{x_1}}$}
Pr{ | x$_1$ | ≤ 0.97 × 10^{-6}} = 1 − 2 Q(z$_1$)
Q(z$_1$)
$z - \frac{x_1(t_1)}{\sigma_{x_1}}$
−z$_1$
z$_1$

Figure C-7. Computation of Pr { | x$_1$ | ≤ 0.2 arcsec }

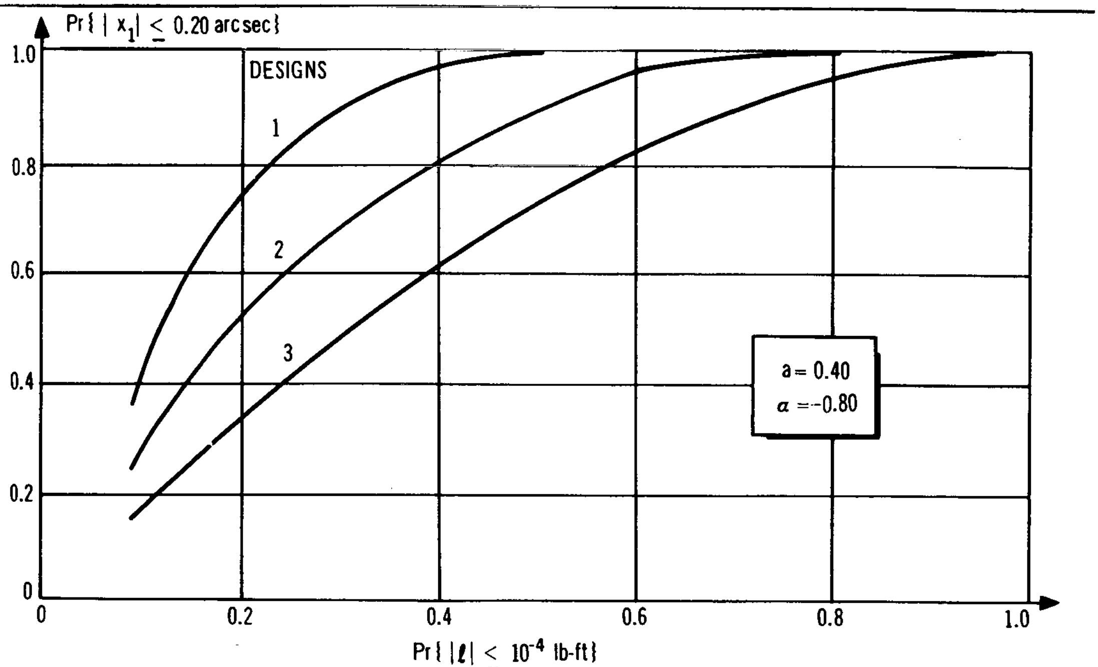


Figure C-8. Probability that Attitude Errors Will be Less Than ± 0.2 arcsec Versus Probability that Disturbance Torques Will be Less Than ± 10^{-4} lb-ft for Three Designs of ρ: (1)p_ℓ = 0.75, (2) p_ℓ = 0.96 and (3) p_ℓ = 0.998, Respectively.

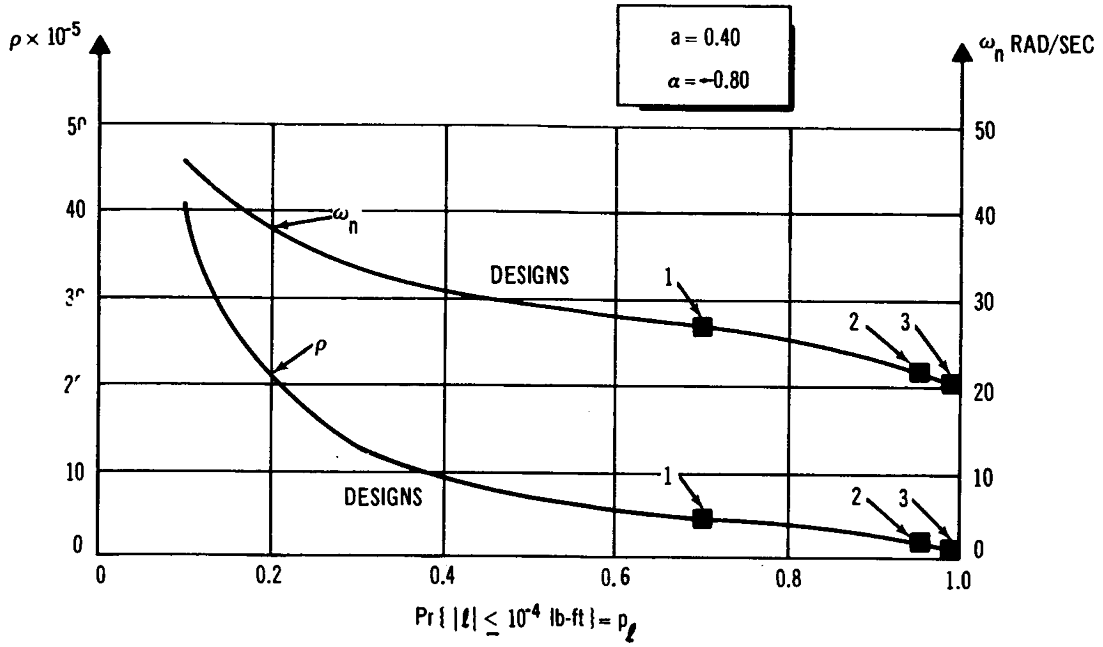


Figure C-9. Weighting Factor ρ and Natural Frequency ω_n as a Function of p_ℓ

The curve for the natural frequency ω_n, in fig. C-9, was obtained using eq. (C33b). As knowledge about $\ell(t)$ increases, ω_n decreases, and the system responds slower. The curve for the weighting factor ρ was obtained, in part, from eq. (C25) and, in part, from the approximation to eq. (C24):

$$r \simeq C_1 \simeq \rho$$

which, as discussed above, holds for small values of a and α.

A continuation of the design of this appendix to systems of higher-order, which makes use of off-line training, is discussed in Appendix D.

References

C1. Potter, J. E.: A Guidance-Navigation Separation Theorem. Paper presented at the AIAA/ION Astrodynamics Guidance and Control Conference. (Los Angeles, Calif.), Aug. 1964.

C2. Kalman, R. E.; Englar, T. S.; and Bucy, R. S.: Fundamental Study of Adaptive Control Systems. ASD-TR-61-27, vol. 1, Wright-Patterson AFB, Ohio, April 1962.

C3. Ellert, F. J.: Indices for Control System Design Using Optimization Theory. Ph. D. thesis, Rensselaer Polytechnic Institute, Jan. 1964.

C4. Potter, J. E.: A Matrix Equation Arising in Statistical Filter Theory. NASA CR-270, 1965.

C5. Abromowitz, M.; and Stegun, L. A., editors: Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables. National Bureau of Standards Applied Math. Series 55.

Appendix D

REALIZATION OF NOMINAL CONTROLS THROUGH OFF-LINE-TRAINING

Theoretically, at least, the nominal feedback parameter vector $\underline{P}_\Lambda$ could be obtained for all possible combinations of a and α, as in Appendix C; however, such an approach would not be practical for plants with more than two variable parameters due to the large computer storage requirements that would result. In addition, for higher-order systems it usually would not be possible to obtain closed-form solutions for the steady-state gains in the matrix Riccati equation; hence, an alternate approach may be required.

One such approach is tc obtain $\underline{P}_\Lambda$ for a representative set of plant parameter vectors $\left\{\underline{P}^j_{A_a}\right\}_{j=1}^{J}$. The set of couples $\left\{\underline{P}^j_{A_a}, \underline{P}^j_\Lambda\right\}_{j=1}^{J}$ is is then used to train an off-line-learning controller.

The function of an off-line-trained controller is to define the complete control law for a plant, with knowledge of the correct control at only a finite number of points. The controller is a pattern recognition device which accepts patterns (plant parameters) as inputs and produces a classification of that pattern (the control) as an output. The relation between input and output is obtained by training a trainable controller with the known finite sample (the training sample). After training, the trained controller (now a deterministic device) provides an implicit definition of the entire control law by virtue of the controller's generalization characteristics. Generalization, of course, is the ability to correctly associate plant situations, and the controls for them, for plant situations which were not included in the original training sample. The ability to do this hinges on the selection of an adequate model for the trainable controller (the functional form of the adaptive mechanism) and a representative training sample.

The trainable controller is organized about discriminators with variable parameters that are adjusted during a training process. Two basic training methods have been distinguished: parametric and non-parametric See Nilsson (ref. D1), for example. The non-parametric methods are applicable when little or no information is available regarding the distribution of patterns in each classification. Parametric methods, on the other hand, are used when information is available. For the present problem the latter are applicable, since it is known that similar plant parameters will have somewhat similar controls.

The realization of an off-line-trained controller may be accomplished in two stages:

(1) A representative training sample consisting of the couples

$$\left\{ \underline{P}^j_{A_a} \quad \underline{P}^j_{\Lambda} \right\}_{j=1}^{J} \quad \text{is obtained.}$$

(2) A self-organizing controller is trained using the above sample.

The above training sample, hereafter designated as S1, is obtained by selecting the points $\underline{P}^j_{A_a}$ uniformly throughout the augmented plant parameter space and obtaining the nominal control, $\underline{P}^j_{\Lambda}$, associated with these points. The idea involved here is that, if training is performed on situations ($\underline{P}^j_{A_a}$) which are dispersed throughout the entire augmented plant parameter space, then all situations will be accounted for in the trained controller. The number of points, J, required in order to have a representative sample is not known a priori; it is selected arbitrarily initially and increased until generalization with the trained controller is acceptable. Generalization is determined using another sample, one with a size also equal to J, and one with elements also uniformly distributed throughout the augmented plant parameter space.

The function of the adaptive computer (self-organizing controller), after training, is to generate the nominal control, $\underline{P}_{\Lambda}$, for any arbitrary plant parameter vector, $\underline{P}_{A_a}$. The adaptive computer organization that is used here has the capacity to provide at most J discrete control choices, where J is the number of elements in the training sample. No provision is made for interpolation; that is to say, points in the augmented plant parameter space that are not in the original training sample are classified into a category corresponding to the most similar element, $\underline{P}^j_{A_a}$ (j = 1, 2, . . . , J), from the training sample. The metric that is used for measuring similarity is the Euclidean norm; hence, classifications are made on the basis of minimum distance.

Two adaptive computers are developed in this section. The first is synthesized directly from the training sample and has the ability to generate J discrete control choices. The second is organized like the first except that it is only able to generate n discrete control choices, where n<J. It is shown that the second adaptive computer has the advantage over the first of smaller size in hardware, or, equivalently, requires less computation time in a computer simulation. The disadvantage of the second computer over the first is that the number of possible control choices is fewer; thus, the selection of controls to affect a minimum distance classification is poorer for the second computer. This deficiency is minimized, however, since the n control choices are selected on the basis of the frequency of occurrence.

Adaptive Computer A

The first adaptive computer utilizes the training sample S1 directly. The classification of an arbitrary point $\underline{P}_{A_a}$ (which may or may not be in the training sample) into one of the J control choices is based on a minimum distance classification. This classification requires the computation of:

$$\left\| \underline{P}_{A_a} - \underline{P}^j_{A_a} \right\| = + \sqrt{\left(\underline{P}_{A_a} - \underline{P}^j_{A_a}\right)^T \left(\underline{P}_{A_a} - \underline{P}^j_{A_a}\right)} \qquad \text{(D1)}$$

for j = 1, 2, ..., J. The control selected for $\underline{P}_{A_a}$ is the one that is associated with the element in $\left\{ \underline{P}^j_{A_a} \right\}_{j=1}^{J}$ for which eq. (D1) is smallest. The discriminant for the minimum distance classifier is developed next.

If both sides of eq. (D1) are squared, the result is:

$$\begin{aligned} \left\| \underline{P}_{A_a} - \underline{P}^j_{A_a} \right\|^2 &= \underline{P}^T_{A_a} \underline{P}_{A_a} - 2 \underline{P}^T_{A_a} \underline{P}^j_{A_a} + \underline{P}^{jT}_{A_a} \underline{P}^j_{A_a} \\ &= \underline{P}^T_{A_a} \underline{P}_{A_a} - 2 \left[\underline{P}^T_{A_a} \underline{P}^j_{A_a} - \frac{1}{2} \underline{P}^{jT}_{A_a} \underline{P}^j_{A_a} \right] \qquad \text{(D2)} \end{aligned}$$

for j = 1, 2, ..., J. Quite obviously eq. (D2) is minimized when the term in brackets is maximized, regardless of the first term (this term is common for all j). The j^{th} discriminant* becomes:

$$g_j(\underline{P}_{A_a}) = \underline{P}^T_{A_a} \underline{P}^j_{A_a} - \frac{1}{2} \underline{P}^{jT}_{A_a} \underline{P}^j_{A_a} \qquad \text{(D3)}$$

for j = 1, 2, ..., J. Observe that $g_j(\underline{P}_{A_a})$ is linear and may be realized with the linear element shown in fig. D-1. The weights w_{j1}, w_{j2}, and w_{j3} are specified on the figure. The complete adaptive computer A is shown in fig. D-2. The operation of the computer is as follows: A plant parameter vector is presented to the machine at the designated input. The outputs of the J discriminants [eq. (D3)]are available as inputs to the maximum

*The relation of discriminants to adaptive computers is discussed in Nilsson (ref. D1) and, therefore, is not elaborated upon here

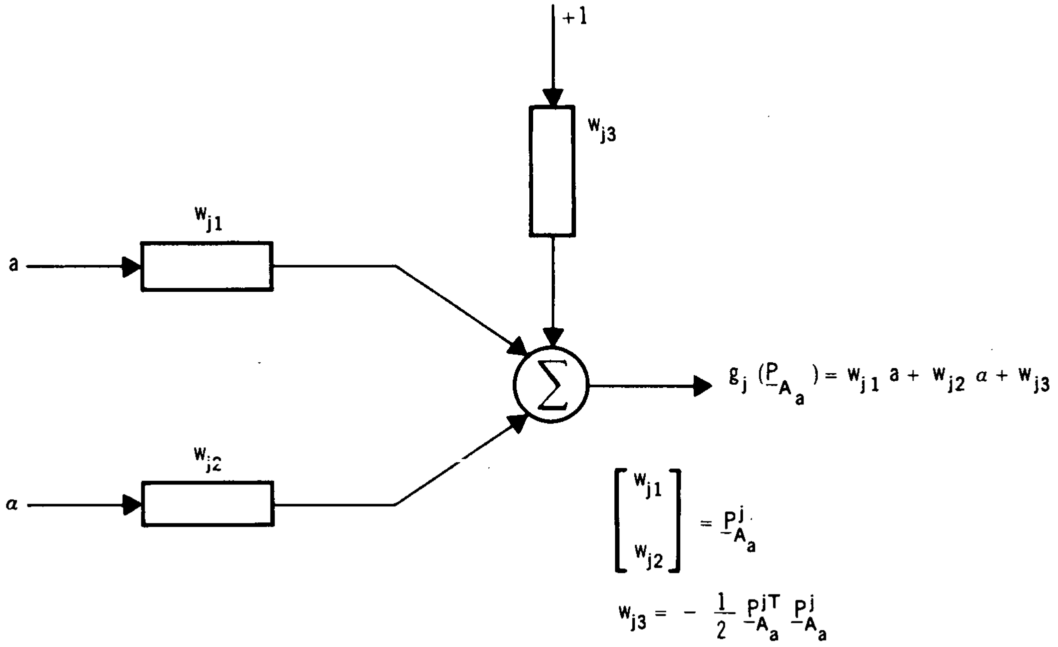

+1
w_{j3}
w_{j1}
a
w_{j2}
α
$g_j(\underline{P}_{A_a}) = w_{j1}a + w_{j2}\alpha + w_{j3}$
$\begin{bmatrix} w_{j1} \\ w_{j2} \end{bmatrix} = \underline{P}^j_{A_a}$
$w_{j3} = -\frac{1}{2}\underline{P}^{jT}_{A_a}\underline{P}^j_{A_a}$

Figure D-1. Realization for Equation (D3), $j = 1, 2, \ldots, J$

selector. This device selects the maximum discriminant and produces, as an output, the control choice associated with this discriminant. The number of discriminators required for this computer is J; that is, J elements in hardware, or J iterations in a computational scheme.

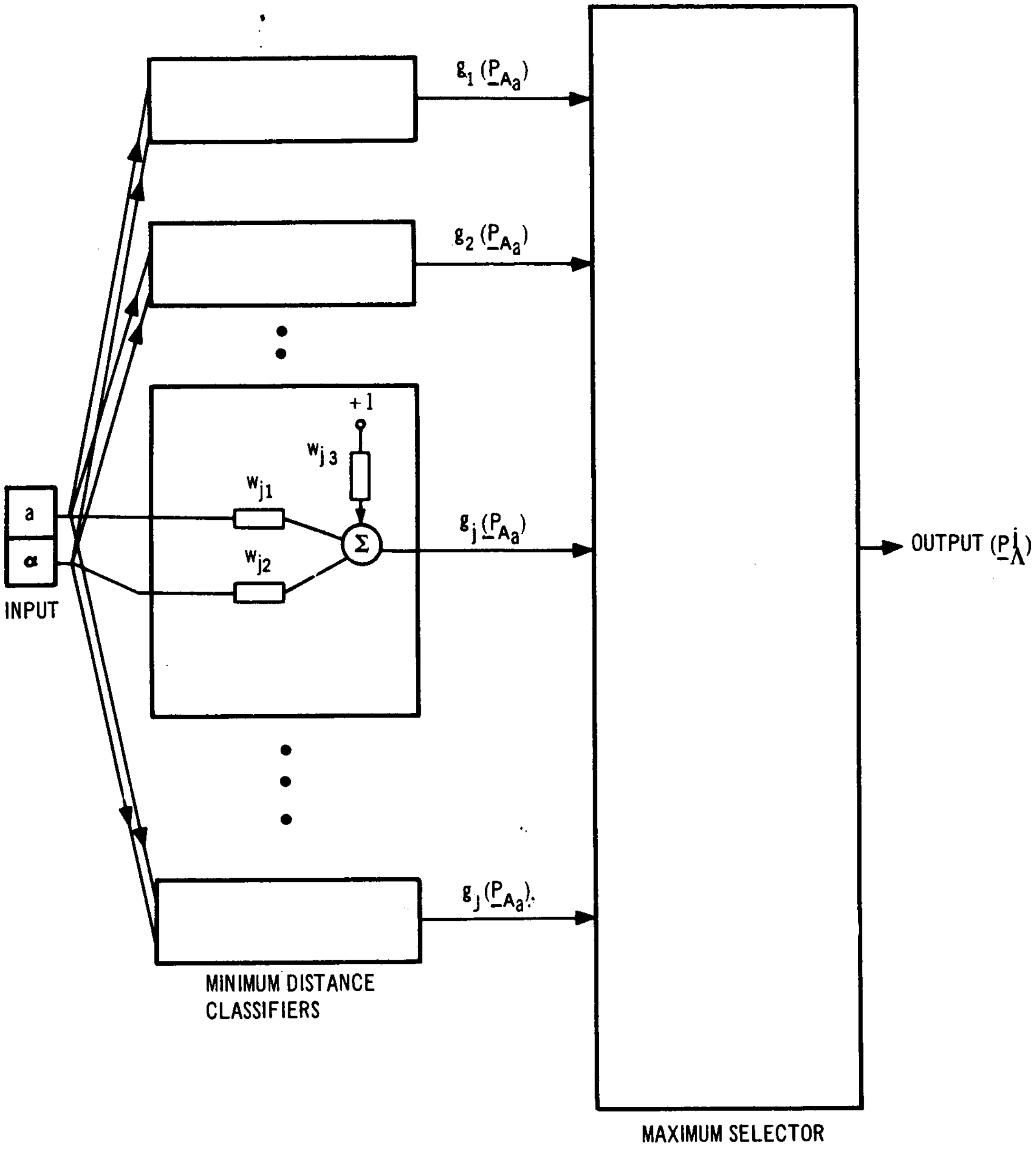
Adaptive Computer B

While excellent results may be obtained with adaptive computer A, the computers disadvantage is the large number of elements (discriminators) required. The development of adaptive computer B is oriented to reduce the number of discriminators and at the same time minimize the loss in proficiency because of this reduction. It involves the following steps:

(1) Modify S1 to S2 by reducing the number of control choices in S1 from J to n ($n < J$).

(2) Specify a computer using S2 and the minimum distance criterion as used for adaptive computer A.

(3) Eliminate all redundant elements in item 2.



$g_1(\underline{P}_{A_a})$
$g_2(\underline{P}_{A_a})$
+1
w_{j3}
w_{j1}
a
α
Σ
w_{j2}
$g_j(\underline{P}_{A_a})$
OUTPUT $(\underline{P}^{j}_{\Lambda})$
INPUT
$g_J(\underline{P}_{A_a})$
MINIMUM DISTANCE CLASSIFIERS
MAXIMUM SELECTOR

Figure D-2. Adaptive Computer A

The reduction of controls specified in the first step is accomplished by merging similar controls $\underline{P}^j_\Lambda (j = 1, 2, \ldots, J)$ in S1 into n prototype controls $\underline{P}^i_\Lambda$ $(i = 1, 2, \ldots, n)$. Two techniques for affecting this reduction are discussed next.

The first technique is elementary in concept and is easy to apply. The feedback parameter space $S_{\underline{P}_\Lambda}$ is pregridded into n m x m x m cubes; the center of each square (hypercube) then becomes a prototype. The points $\underline{P}^j_\Lambda$ $(j = 1, \ldots, J)$ in this space are merged into the prototype point of the hypercube in which the points lie. The selection of prototypes by this method usually will not be very efficient unless the distribution of points $\underline{P}^j_\Lambda$ $(j = 1, \ldots, J)$ happens to be uniform throughout $S_{\underline{P}_\Lambda}$, for prototypes may be assigned where classifications rarely occur.

A more efficient method is one in which prototypes are created according to the density of the vectors in $S_{\underline{P}_\Lambda}$. This is accomplished by performing a cluster analysis of the feedback parameter space and assigning prototypes to the cluster points found. One method for performing such an analysis has been successfully used by Sebestyen (ref. D2) and is presented next (as our second technique).

Sebestyen's technique is similar to the approach used in on-line learning for creating hyperspherical control situations. Initially the method requires that a cell (a hypersphere of radius D) be created and centered on an arbitrarily selected point $\underline{P}^j_\Lambda (j = 1, 2, \ldots, J)$. The mean of this cell, M_1, is initially set equal to $\underline{P}^j_\Lambda$ and the number of points in the cell, N_1, is initially set equal to 1. If the next pattern falls within the cell (that is, if the distance between M_1 and the pattern is less than D), M_1 is updated by recomputing the mean of the two samples, and N_1 is set equal to 2. When a sample falls sufficiently far outside the first cell, a new cell is created whose center is located at that pattern. If the sample falls outside the first cell by a small amount, then the sample is temporarily stored. The purpose of this step is to create a guard ring which will allow maneuvering room for the cell. As the process continues all vectors are forced into cells. At the conclusion of the process the mean of each cell is the location of a cluster point. The number of points in the cell, N_i $(i = 1, 2, \ldots n)$, divided by J, the total number of points, is the a priori probability, p_i, of the occurrence of the i^{th} cell. Cells with low values of p_i may be combined with adjacent cells if a reduction in the number of prototypes is desired.

The modified training sample S2, obtained by either of the above techniques, results in the formation of n mutually exclusive subsets. The elements of each subset are the points $\underline{P}^j_{A_a}$ $(j = 1, 2, \ldots, J)$ in S2 which have the same control. The designation for the i^{th} subset $(i = 1, 2, \ldots, n)$, which contains m_i elements, all of whose controls are $\underline{P}^i_\Lambda$, is $\left|\underline{P}^{i,k}_{A_a}\right|^{m_i}_{k=1}$. Since there are n mutually exclusive subsets,

$$\sum_{i=1}^{n} m_i = J \qquad \text{(D4)}$$

T'ie elements of each subset define regions of constant control in $S_{\underline{P}_{A_a}}$ Each of these regions, in general, will vary in size and geometry depending on the number and the location of points in the subset. The boundaries which separate these regions may be viewed as decision boundaries which are to be realized by the adaptive computer. The adaptive computer to be used requires n discriminators, one for each control category. As before, the output of the computer is the control associated with:

$$\max_{i = 1, \ldots, n} \left\{ g_i \left(\underline{P}_{A_a} \right) \right\}$$

The computer organization which is used to realize the decision boundaries is shown in fig. D-3. Each discriminator is associated with the boundary that defines the control region of that discriminator.

The discriminator, g_i ($\underline{P}_{A_a}$), whose output is maximum for all points in the i^{th} control region, uses m_i subsidiary discriminators which are defined directly from the i^{th} subset. Each subsidiary discriminator is a minimum distance classifier; hence (compare with weights in fig. D-1)

$$\begin{bmatrix} w_{i1}^{(k)} \\ \\ w_{i2}^{(k)} \end{bmatrix} = \underline{P}_{A_a}^{i,k} \qquad \text{(D5)}$$

and

$$w_{i3}^{(k)} = -\frac{1}{2} \left(\underline{P}_{A_a}^{i,k} \right)^T \underline{P}_{A_a}^{i,k} \qquad \text{(D6)}$$

where $k = 1, \ldots, m_i$, define the appropriate weights of the k^{th} subsidiary discriminator in the i^{th} discriminator. The output of the i^{th} discriminator for an arbitrary input $\underline{P}_{A_a}$ is

$$\max_{k = 1, \ldots, m_i} \left\{ \underline{P}_{A_a}^T \underline{P}_{A_a}^{i,k} - \frac{1}{2} \left(\underline{P}_{A_a}^{i,k} \right)^T \underline{P}_{A_a}^{i,k} \right\} = g_i \left(\underline{P}_{A_a} \right) \qquad \text{(D7)}$$

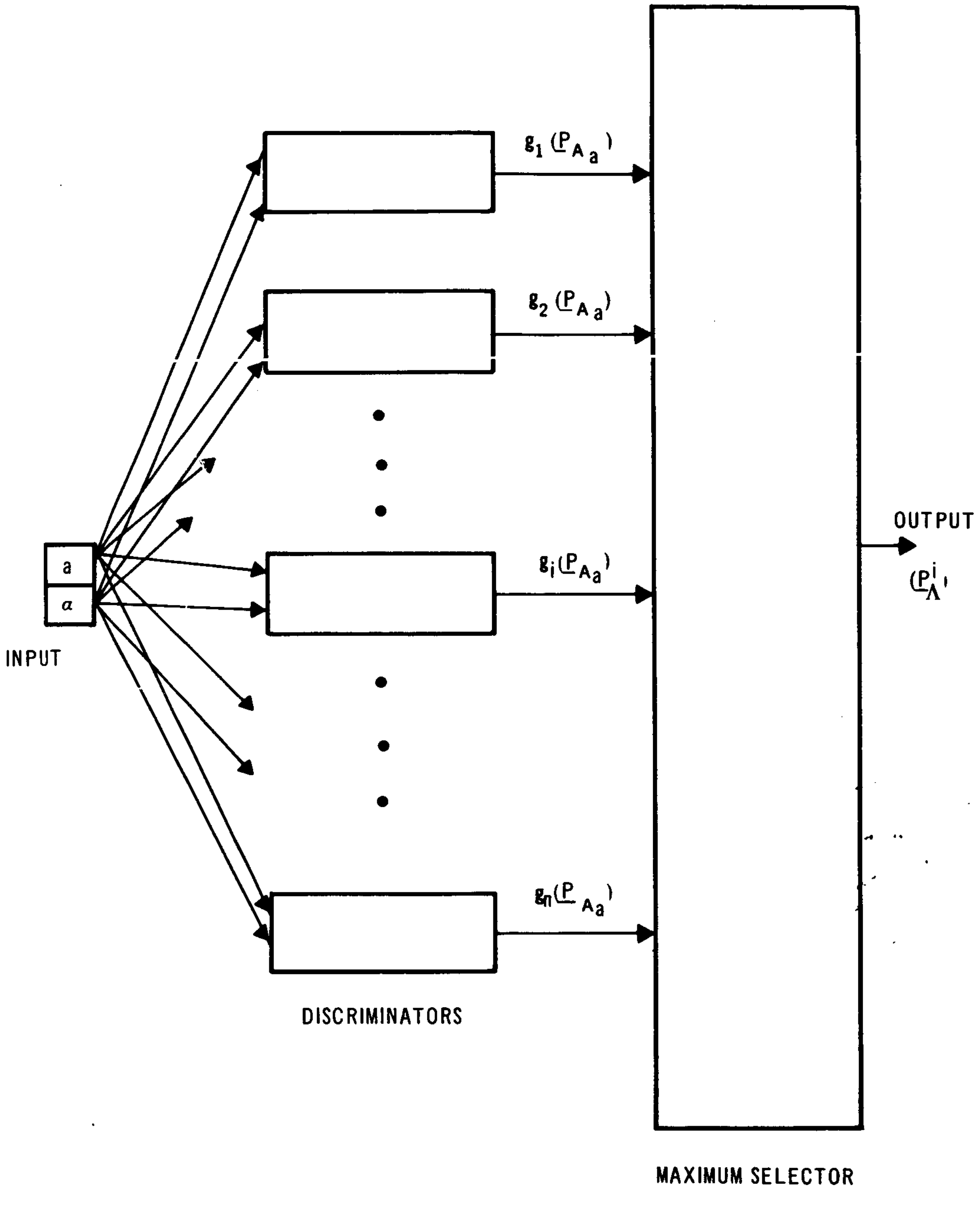

$g_1 (\underline{P}_{A_a})$
$g_2 (\underline{P}_{A_a})$
OUTPUT
$(\underline{P}^i_{\Lambda}$
a
α
INPUT
$g_i(\underline{P}_{A_a})$
$g_n(\underline{P}_{A_a})$
DISCRIMINATORS
MAXIMUM SELECTOR

Figure D-3. Adaptive Computer B

[compare with the discussions for eqs. (D2) and (D3)]. A detailed structure for the i^{th} discriminator appears in fig. D-4.

The decision surfaces which may result from the above process are demonstrated in an example in fig. D-5a. There are three sets of points which define three control regions. If the points are used to specify three discriminators [using eqs. (D5) and (D6)] the indicated decision boundary results. This boundary is defined by segments of straight lines which separate points of different classes. The lines are always drawn equidistant between the closest points of different classes. The lines between points of the same class are not drawn (though they are defined in the computer) because they do not partition different control regions (hence, they are redundant). This point introduces the final step in the development of the adaptive computer, namely, elimination of redundant elements.

In fig. D-5b, it is shown how the exact decision boundary in the above example can be defined using only essential points from fig. D-5a. Each point used corresponds to an essential subsidiary discriminator in the computer. The elimination of redundant elements from the computer may be accomplished with the following procedure:

(1) Arbitrarily remove the subsidiary discriminant $g_i^{(k)}(\underline{P}_{A_a})$ from the computer (see fig. D-4).

(2) Apply the point $\underline{P}_{A_a} = \begin{bmatrix} w_{i1}^{(k)} \\ w_{i2}^{(k)} \end{bmatrix}$ as an input to the computer.

(3) If the classification produced is $\underline{P}_{\Lambda}^{i}$ discard the element; if not reinsert the element.

(4) Repeat the above steps for all subsidiary discriminators.

The above procedure essentially tests every point to see if it is needed to define the decision boundary. This, of course, results in reducing the number of elements in the i^{th} discriminator from m_i to another number, say $\ell_i (\ell_i \leq m_i)$. It is not possible to state the extent of the reduction, as it depends on the number, size and distribution of each subset.

The final test of the resultant computer is its generalization characteristics. As stated previously, generalization is the ability to classify points correctly even though the points were not included in the training sample. This test requires a sample similar to S1; the sample should have points which are uniformly distributed throughout $S_{\underline{P}_{A_a}}$ (but not the exact same points) and sample size should be also about J. This sample is applied to the computer and an evaluation procedure determines the proficiency obtained. On this basis a decision is made concerning the acceptability of the adaptive computer's generalization characteristics. If generalization is found unacceptable, then a new sample is required, one which is larger than S1, and a new adaptive computer is developed with this sample.

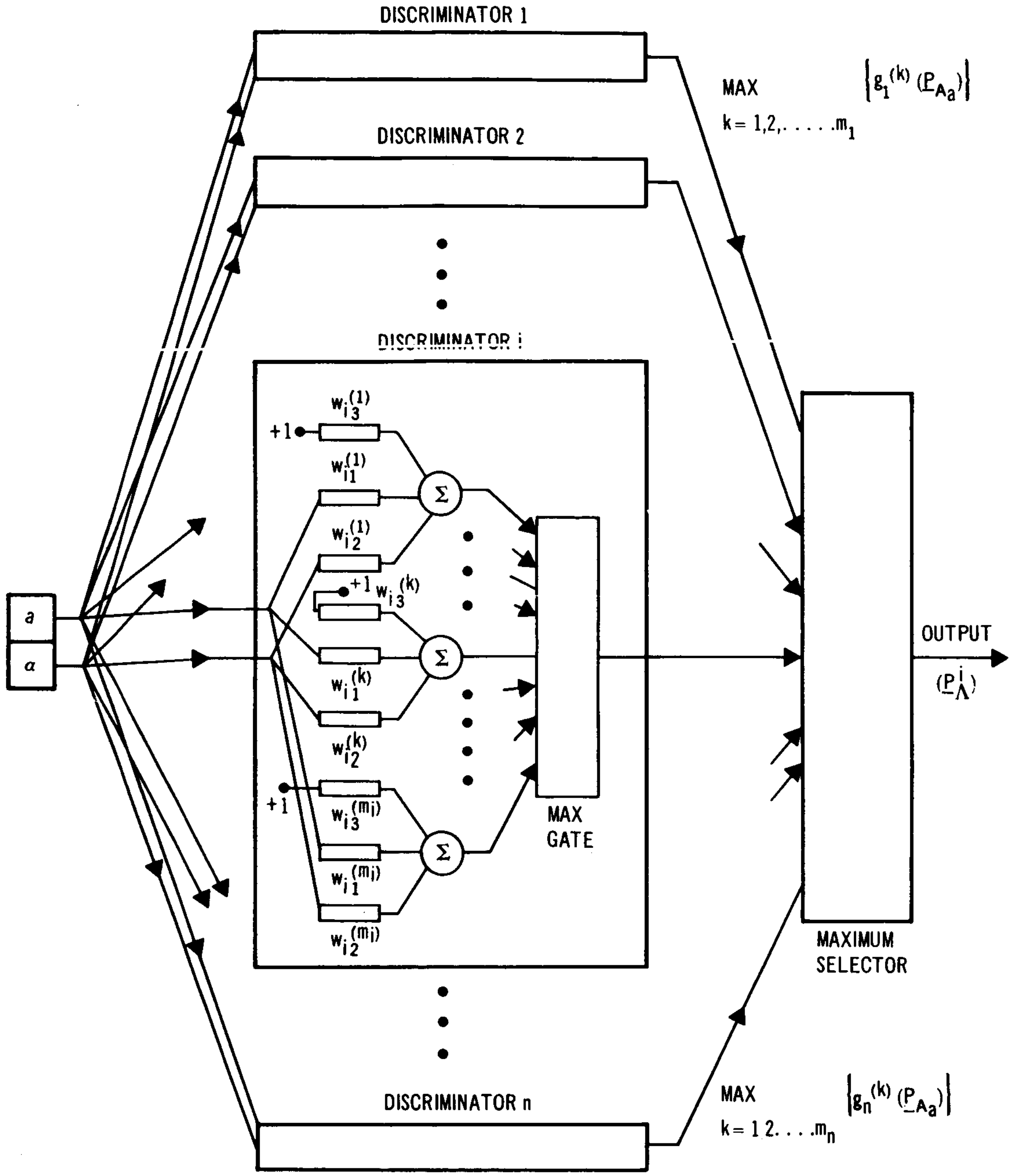
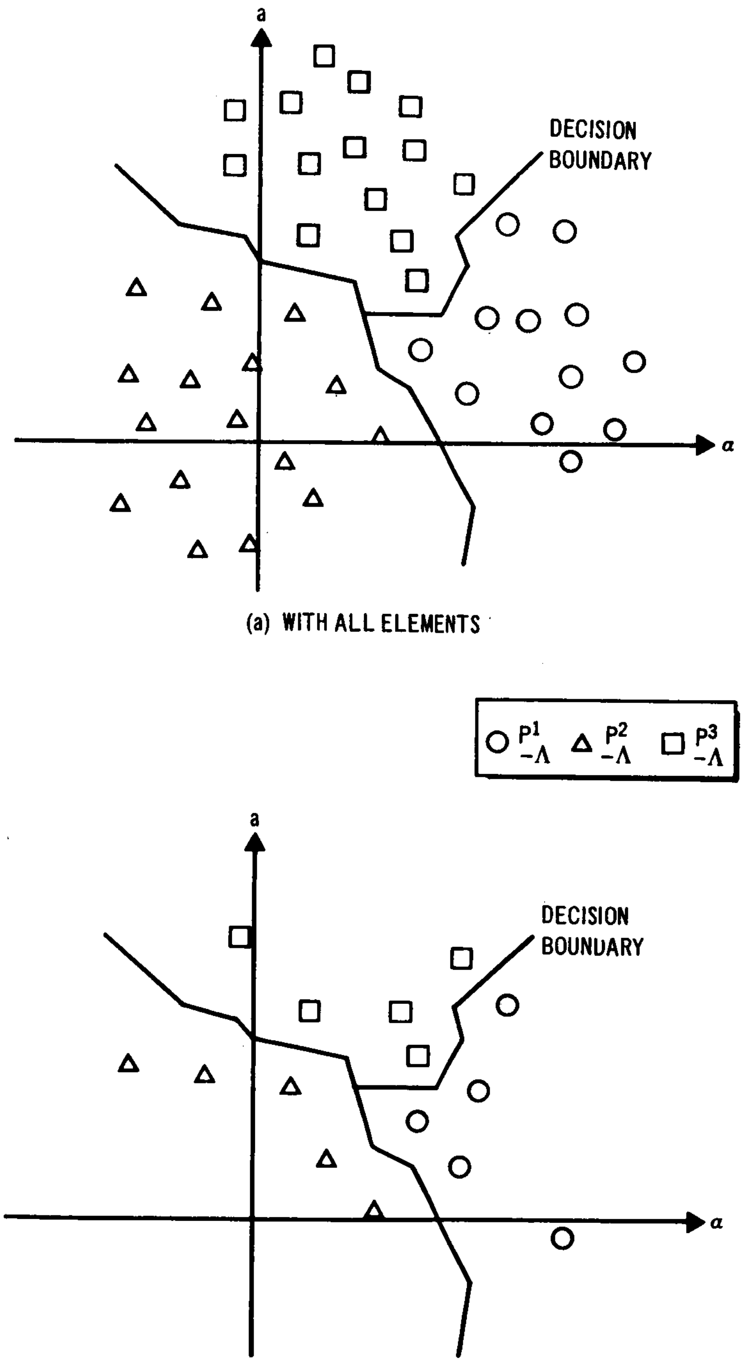


Figure D-4. Adaptive Computer B, Detailed



(a) WITH ALL ELEMENTS

(b) REDUNDANT ELEMENTS REMOVED

Figure D-5. Decision Boundaries Using Minimum-Distance Classification

The final comment is concerned with the implementation of the adaptive computer. Two alternative approaches are available; hardware and simulation. The choice of one over the other depends on various systems considerations that will not be discussed here. The hardware approach results in a small package since each weight [eqs. (D5) and (D6)] may be realized with a resistor and the summation may be performed with miniature solid-state circuits. A computer implementation (simulation) requires that all of the subsidiary discriminants must be evaluated and compared in order to determine the maximum values in eq. (D7).

References

D1. Nilsson, N. J.: Learning Machines: Foundations of Trainable Pattern-Classifying Systems. McGraw-Hill Book Co., Inc., 1965.

D2. Sebestyen, G.; and Edie, J.: Pattern Recognition Research. AFCRL-64-821. Wright-Patterson AFB, Ohio, June 1964.

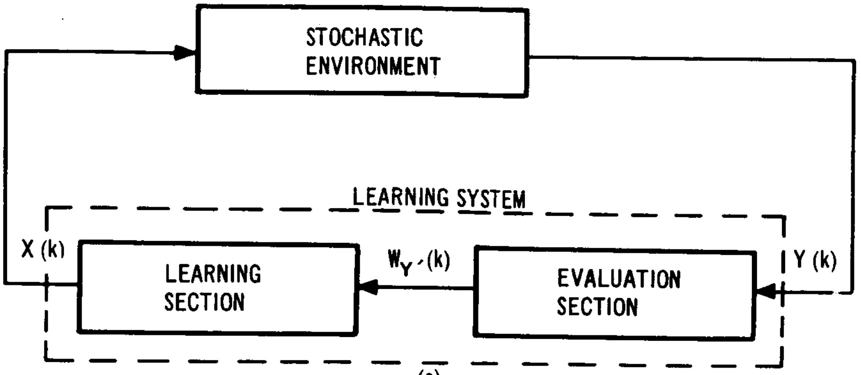
Appendix E
STOCHASTIC AUTOMATA LEARNING ALGORITHM

Recently, Fu and McLaren (ref. E1) have investigated a learning system the synthesis of which is based upon observed interactions between the learning system and its stochastic environment. The similarities between the Fu-McLaren system and the system presented in this document, as well as in the philosophies of solution are revealed in the following pertinent quotes:

> "The learning system will perform its operations iteratively, during successive intervals (or steps) of time. During a given step in time, the output of the learning system is a particular action or set of parameter values resulting from a decision by the learning system. The action or output takes place in the environment. As a result, the environment reacts to this action in a certain way, observable to the learning system. Having observed this, the system evaluates the reaction as to desirability. Based upon this evaluation, the learning system will change its structure or parameters in selecting its action for the next step. The change should be such that the 'average' desirability (or performance) increases (or improves). As structural or parametric changes are repeated step by step, the learning system seeks (or converges toward) the action that would yield the best performance. This step by step process is considered as a learning process--steady improvement of performance based upon past experience. The system learns as it operates in its environment; there is no separation of the operation into training and working phases. The system learns from environmental reaction to its own actions. The environment is assumed, of course, to be random; its statistical properties are, in general, unknown to the designer except, perhaps, for the assumption of stationarity. In order that the learning system be able to evaluate its reactions from the environment, the value of each 'useful' (as a measure of performance) reaction must be specified. Such specifications are bound-up with the purpose or goal of the learning system."

The notational correspondence between the problem in this document and Purdue's is summarized in fig. E-1. [The block diagram in (a) is adapted from fig. 1 of ref. E1.] It is assumed in this section that $\overline{PI}\ (k, \ell; m)$ is not given by eq. (58); $\overline{PI}\ (k, \ell; m)$ is, for the present, an unknown measure of the system's performance [at t = (k+1) T, k = 0, 1, ...]. In Item 9D in fig. E-1 $\overline{PI}\ (k, \ell; m)$ is normalized in order to ensure that

$$0 \leq \frac{\overline{PI}\ (k, \ell; m)}{CT} \leq 1 \qquad \text{(E1)}$$



(a)

PURDUE SYSTEM (P)	DOUGLAS SYSTEM (D)
1P. STOCHASTIC ENVIRONMENT	1D. SPACECRAFT ACTED UPON BY DISTURBANCES
2P. EVALUATION SECTION	2D. MEASURER; STATE AND PARAMETER ESTIMATOR; ON-LINE PERFORMANCE INDEX COMPUTOR
3P. LEARNING SECTION	3D. ON-LINE-LEARNING ALGORITHM
4P. LEARNING SYSTEM	4D. SELF-ORGANIZING CONTROLLER
5P. X (k)	5D. $\underline{P}^{*\ell}_{\Lambda}\ (k;t)$
6P. Y (k)	6D. MEASUREMENTS
7P. Y´(k)[FEATURES OR PROPERTIES]	7D. $\left(\hat{\underline{P}}_{A_a}(t\|t) \mid \hat{\underline{x}}_a(t\|t) \mid z_1(k,\ell;t)\right)^T$
8P. $W_{Y'}(k)$	8D. $\overline{PI}(k,\ell;m)/T$
9P. $Z_{Y'}(k)$	9D. $\overline{PI}(k,\ell;m)/CT$, WHERE $C = \max_{k} \overline{PI}(k,\ell;m)/T$
10P. $\bar{P}_m(k) \simeq E_Z \{ Z_Y(k) \}$	10D. PI (k, ℓ; m) , WHERE $PI(k,\ell;m) = \frac{1}{(k+1)} \sum_{\beta=0}^{k} \frac{\overline{PI}(\beta,\ell;m)}{CT}$

(b)

Figure E-1. Learning System Block Diagram and Correspondences Between the Purdue and the Douglas Systems for all $k \geq 0$. This normalization is required in order to apply stochastic automata theory to the synthesis of the self-organizing controller.

Stochastic Automata Approach

Assume that the ℓ^{th} control situation has been established and that the nominal feedback parameter vector, $\underline{P}_\Lambda^\ell$, has been associated with it. A set of r uniformly distributed feedback vectors, which includes $\underline{P}_\Lambda^\ell$ as an element †, is then constructed. The construction of this set, which is centered about $\underline{P}_\Lambda^\ell$ in a hypersphere of radius s, and which is denoted

$$\left|\underline{P}_i^\ell\right|_{i = 1}^{i = r} = \left|\underline{P}_i^\ell\right|_1^r ,$$

where

$$\underline{P}_r^\ell \triangleq \underline{P}_\Lambda^\ell , \tag{E2}$$

is discussed shortly.

Probabilities $p_1, p_2, \ldots,$ and p_r are assigned to each one of the r possible feedback parameter vectors in eq. (E2). These probabilities are updated every T units of time according to the following rule:

If

$$\underline{P}_\Lambda^{*\ell} (k; t) = \underline{P}_j^\ell$$

then

$$p_j (k + 1) = \Theta p_j (k) + (1 - \Theta) \left[1 - Z (k, \ell; m)\right] \tag{E3a}$$

$$p_i (k + 1) = \Theta p_i (k) + \frac{(1 - \Theta)}{(r - 1)} Z (k, \ell; m) \quad i \neq j = 1, \ldots, r \tag{E3b}$$

for $k = 0, 1, \ldots, k_m$ ††. In eqs. (E3a) and (E3b)

† In general, $\underline{P}_\Lambda^\ell$ need not be included a priori in this set. If it is not included, then eq. (E2) does not hold; that is to say, $\underline{P}_r^\ell$ is chosen in the same way as the other r-1 feedback vectors.

†† The notation, p_i (k+1) for the probability that $P_\Lambda^{*\ell}$ (k+1; t) = $\underline{P}_i^\ell$ does not reflect the conditioning to $\underline{P}_\Lambda^{*\ell}$ (k; t). This notation is used for convenience.

$$0 < \theta < 1 \tag{E4}$$

and

$$Z\,(k,\ \ell;m) \triangleq \frac{\overline{PI}\,(k,\ \ell;m)}{CT} \tag{E5}$$

In general, $Z(k,\ell;m)$ is either a discrete-parameter (k) continuous-valued random process or a discrete-parameter discrete-valued process; hence, the probabilities in eqs. (E3a) and (E3b) are discrete-parameter continuous-valued random processes (random sequences).

Updating the probabilities according to eqs. (E3a) and (E3b) is shown by Fu and McLaren to lead to a control choice which minimizes the expected value of $Z(k,\ \ell;\ m)$, where (ref. E1).

$$E\left|Z\,(k,\ \ell;\ m)\right| = \sum_{i=1}^{r} E\left|Z\,(k,\ell;\ m)\middle|\underline{P}_{\Lambda}^{*\ell}\,(k;t) = \underline{P}_{i}^{\ell}\right| E\left|p_i(k)\right| \tag{E6}$$

Eq. (E6) is written for convenience as

$$E\left|Z\,(k,\ \ell;\ m)\right| = \sum_{i=1}^{r} m_i\ (k)\ E\left|p_i\ (k)\right| \tag{E7}$$

where

$$m_i\ (k) = E\left|\ Z\ (k,\ \ell;\ m)\middle|\ \underline{P}_{\Lambda}^{*\ell}\ (k;\ t) = \underline{P}_{i}^{\ell}\right| \tag{E8}$$

Fu and McLaren prove that in the stationary case, that is, the case when

$$m_i\ (k) = m_i \text{ for all } k\ (i = 1,\ \ldots,\ r) \tag{E9}$$

$$E\left|p_i\ (k)\right| \to \gamma_i$$

monotonically, where

$$\gamma_i = \prod_{\substack{j=1\\ \neq i}}^{r} (1 - m_j) \Bigg/ \sum_{j=1}^{r} \prod_{\substack{q=1\\ \neq j}}^{r} (1 - m_q) \tag{E10}$$

and

$$E\left|Z\ (k+1,\ \ell;\ m)\right| < E\left|\ Z\ (k,\ \ell,m)\right|,\ \text{finite } k \geq 0,$$

and

$$\lim_{k\to\infty}\ E\left|\ Z\ (k,\ \ell;\ m)\ \right| = \sum_{j=1}^{r} m_j\ \gamma_j \tag{E11}$$

which means that E {Z (k, ℓ; m)} converges in a monotonic decreasing manner toward its minimum limit value.

They also demonstrate that eqs. (El0) and (E11) hold, if

$$E\left|\ Z\ (k,\ell;\ m)\ \middle|\ \underline{P}_{\Lambda}^{*\ell}\ (k;\ t)\ =\ \underline{P}_i^{\ell}\right|\ =\ m_i\ (k)\quad i\ =\ 1,\ \ldots,\ r \tag{E12a}$$

$$\lim_{k\to\infty}\ m_i\ (k) \to m_i,\ \text{monotonically,} \tag{E12b}$$

$$\gamma_i\ (k) \to \gamma_i\ \text{monotonically so that} \tag{E12c}$$

$$\gamma_i \geq \frac{1}{r} \Rightarrow \gamma_i\ (k) \geq \frac{1}{r}\ ,\quad \gamma_i < \frac{1}{r} \Rightarrow \gamma_i(k) < \frac{1}{r},\ \text{all } k \tag{E12d}$$

Hence, monotonic convergence is guaranteed even in certain nonstationary cases.

It is important to note that, in general, the values for the conditional expected values $m_i\ |\ m_i\ (k)\ |$ are not available to the designer. If Z(k, ℓ; m) is an ergodic random process, then the expected value of Z(k, ℓ; m) is approximated by the average

$$E\left|Z\ (k,\ \ell;\ m)\right| \simeq \frac{1}{(k+1)}\ \sum_{\beta=o}^{k}\ \overline{PI}\ (\beta,\ \ell;\ m)\ /\ CT \tag{E13}$$

which provides a convenient way of evaluating the expected value on-line. Observe, also, from eq. (E13) and Item 10D in fig. E-1, that for an ergodic process

$$E\left|\ Z\ (k,\ \ell;\ m)\right| \simeq PI\ (k,\ \ell;\ m) \tag{E14}$$

hence, for an ergodic process, the Douglas and Purdue indexes of performance are the same.

One final observation is in order. Suppose that instead of minimizing $E\{Z(k, \ell; m)\}$, it is desired to minimize the expected value of the discrete time average of $Z(k, \ell; m)$; i.e., to minimize

$$E\left\{\frac{1}{(k+1)} \sum_{\beta=o}^{k} Z(\beta, \ell; m)\right\}$$

It is straightforward to show that a sufficient, although not necessary, condition for minimizing this quantity is eq. (E3); hence, for an ergodic process, the solution to the problem of minimizing $E\{Z(k, \ell; m)\}$ is also a solution to the problem of minimizing $E\{PI(k, \ell; m)\}$.

In the next section, it is assumed that either $m_i(k) = m_i$ (for all $k \geq 0$) or the conditions in eq. (E12) are satisfied. Further discussions on this important point appear in the section on Convergence of Algorithm.

Description of On-Line Learning Algorithm

An on-line learning algorithm that is based on the preceding optimization technique is summarized in fig. E-2. It is assumed here that the overall goal is to minimize $E\{Z(k, \ell; m)\}$. [It is not contended here that minimizing $E\{Z(k, \ell; m)\}$ is sufficient for reducing attitude errors to within ±0.2 arc-sec] and that in each control situation on-line learning terminates as this goal is achieved; hence, it is possible to reach a state of complete learning. By definition, complete learning refers to the condition when the optimization problem has been satisfactorily solved for every existing control situation.

The control situation portion of the algorithm in fig. E-2 has been discussed in connection with the error-correction algorithm and, therefore, will not be elaborated upon here. After a new control situation is established, a set of r uniformly distributed vectors $\{\underline{P}_i^c\}_1^r$ is constructed. Before proceeding further, consider the construction of this set.

Construction of Uniformly Distributed Feedback Parameter Vectors. -- A set of vectors $\{\underline{P}_i^c\}_1^r$ must be constructed that are uniformly distributed in a hypersphere of radius s that is centered about $\underline{P}_\Lambda^c$, the nominal control associated with the cth control situation. It is assumed that

$$\underline{P}_r^c = \underline{P}_\Lambda^c \tag{E15}$$

which means that only r-1 vectors are needed. Let

$$\underline{P}_i^c = \underline{P}_\Lambda^c + \underline{\delta}_i \qquad i = 1, \ldots, r-1 \tag{E16}$$

where the components of $\underline{\delta}_i$ [$\underline{\delta}_i = (\delta_{i1}, \delta_{i2}, \delta_{i3})^T$]are uniformly distributed (fig. E-3). It follows from the construction statement above, that

$$\left\| \underline{P}_i^c - \underline{P}_\Lambda^c \right\| \leq s \quad i = 1, \ldots, r - 1 \qquad \text{(E17)}$$

hence,

$$\delta^2_{i1} + \delta^2_{i2} + \delta^2_{i3} \leq s^2 \quad i = 1, \ldots, r - 1 \qquad \text{(E18)}$$

Eq. (E18) must be satisifed for all values of δ_{ij}; hence, it must be satisfied for the worst possible case, when $\delta_{ij} = \Delta$, $j = 1, 2, 3$. In this case, eq. (E18) becomes

$$\Delta \leq \frac{s}{\sqrt{3}} \cdot \qquad \text{(E19)}$$

One concludes, therefore, that choosing

$$\Delta = \frac{s}{\sqrt{3}} \qquad \text{(E20)}$$

is sufficient to satisfy eq. (E18) for all values of δ_{ij}.

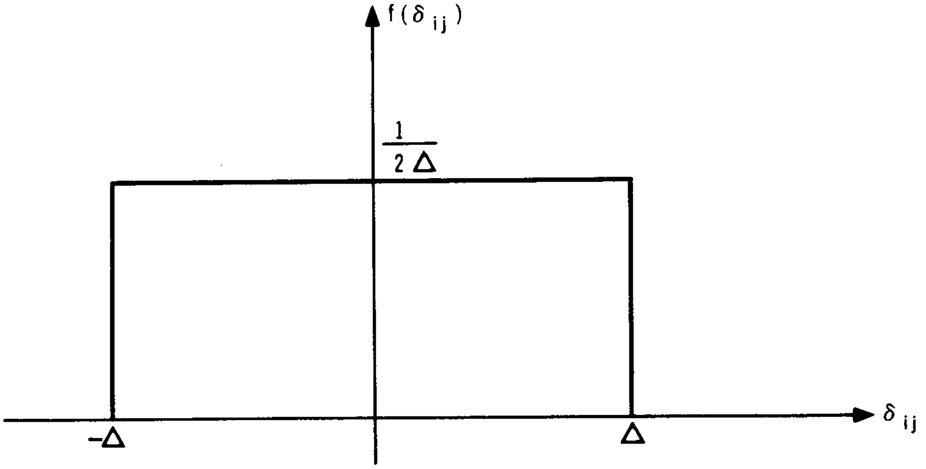


Figure E-3. Uniformly Distributed δ_{ij}'s (j = 1, 2, 3,)

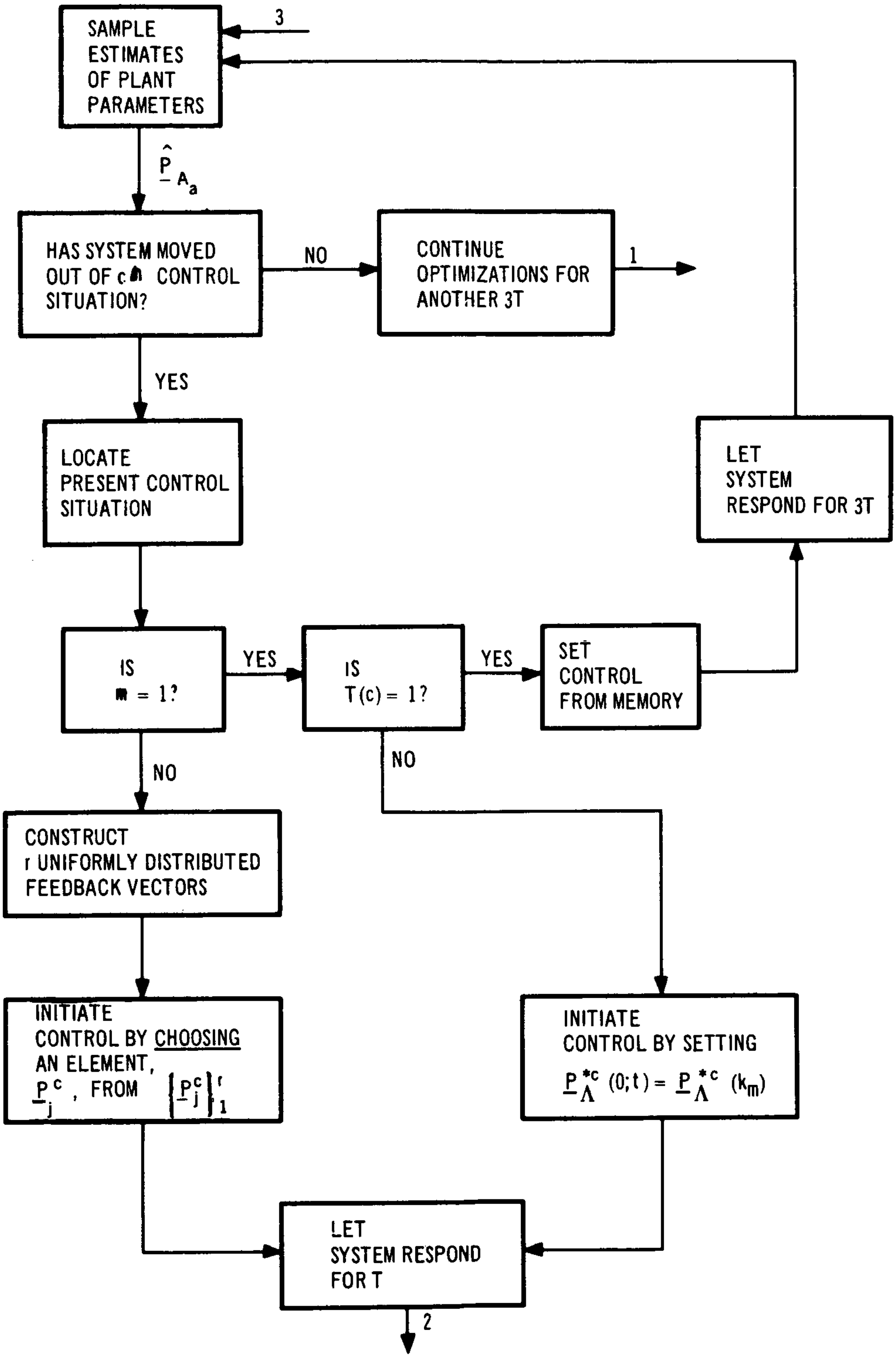

3
SAMPLE ESTIMATES OF PLANT PARAMETERS
$\hat{\underline{P}}_{A_a}$
HAS SYSTEM MOVED OUT OF c m CONTROL SITUATION?
NO
CONTINUE OPTIMIZATIONS FOR ANOTHER 3T
1
YES
LOCATE PRESENT CONTROL SITUATION
LET SYSTEM RESPOND FOR 3T
IS m = 1?
YES
IS T(c) = 1?
YES
SET CONTROL FROM MEMORY
NO
NO
CONSTRUCT r UNIFORMLY DISTRIBUTED FEEDBACK VECTORS
INITIATE CONTROL BY CHOOSING AN ELEMENT, $\underline{P}_j^c$, FROM $\left\{\underline{P}_j^c\right\}_1^r$
INITIATE CONTROL BY SETTING $\underline{P}_\Lambda^{*c}(0;t) = \underline{P}_\Lambda^{*c}(k_m)$
LET SYSTEM RESPOND FOR T
2

Figure E-2. Stochastic Automata Learning Algorithm

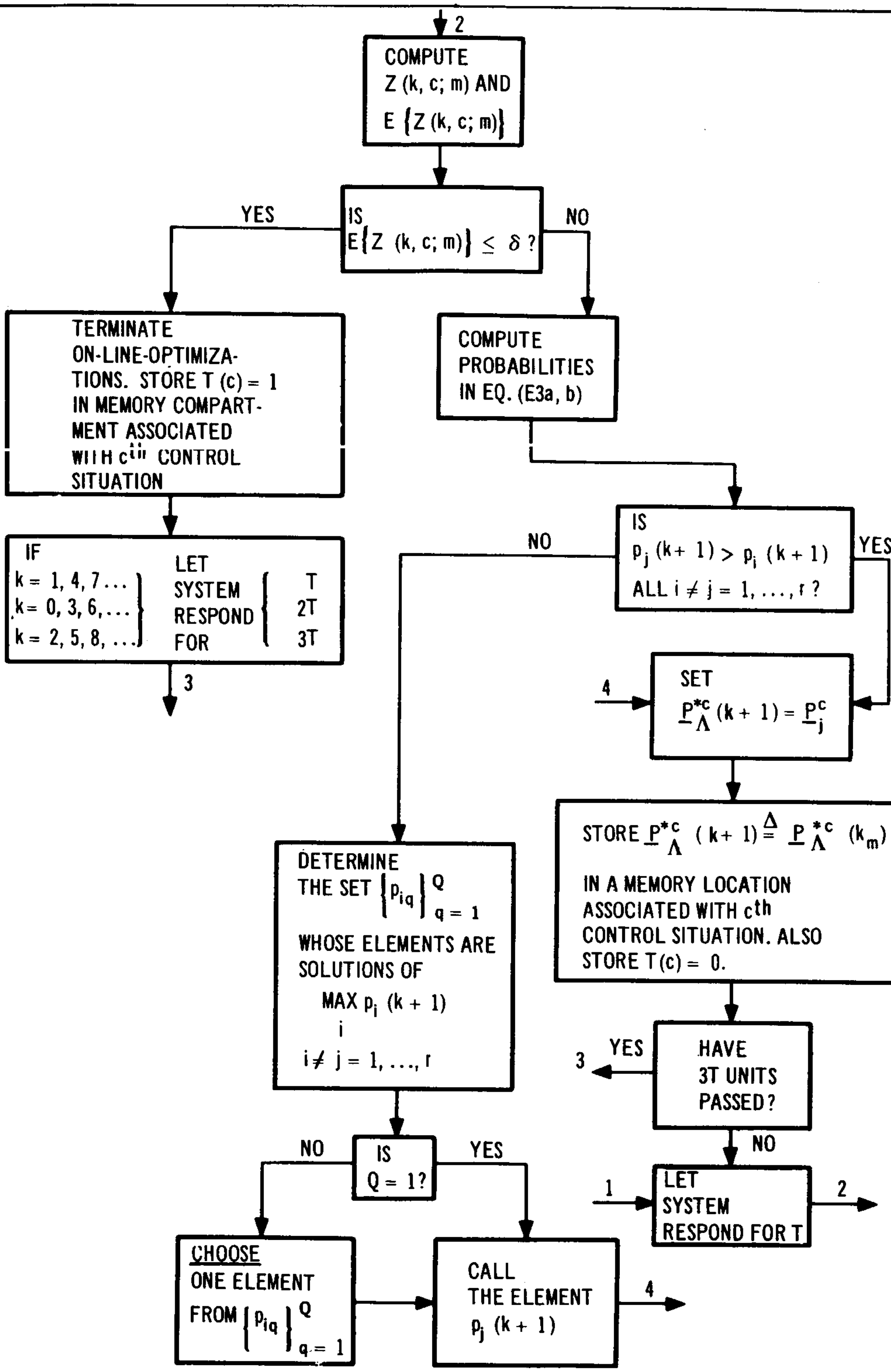

2
COMPUTE Z (k, c; m) AND E {Z (k, c; m)}
IS E{Z (k, c; m)} ≤ δ ?
YES
NO
TERMINATE ON-LINE-OPTIMIZATIONS. STORE T (c) = 1 IN MEMORY COMPARTMENT ASSOCIATED WITH cth CONTROL SITUATION
COMPUTE PROBABILITIES IN EQ. (E3a, b)
IF k = 1, 4, 7... k = 0, 3, 6,... k = 2, 5, 8, ... LET SYSTEM RESPOND FOR T 2T 3T
3
IS $p_j (k+1) > p_i (k+1)$ ALL i ≠ j = 1, . . . , r ?
NO
YES
4
SET $\underline{P}^{*c}_{\Lambda} (k+1) = \underline{P}^{c}_{j}$
STORE $\underline{P}^{*c}_{\Lambda} (k+1) \overset{\Delta}{=} \underline{P}^{*c}_{\Lambda} (k_m)$ IN A MEMORY LOCATION ASSOCIATED WITH cth CONTROL SITUATION. ALSO STORE T(c) = 0.
DETERMINE THE SET $\{p_{iq}\}^{Q}_{q=1}$ WHOSE ELEMENTS ARE SOLUTIONS OF MAX $p_i (k+1)$ i i≠ j = 1, . . ., r
3
YES
HAVE 3T UNITS PASSED?
NO
NO
IS Q = 1?
YES
1
LET SYSTEM RESPOND FOR T
2
CHOOSE ONE ELEMENT FROM $\{p_{iq}\}^{Q}_{q=1}$
CALL THE ELEMENT $p_j (k+1)$
4

Figure E-2. (Continued)

Continuation of Description of Learning Algorithm in Fig. E-2.--After constructing $\{\underline{P}_i^c\}_1^r$, probabilities $p_1(k), \ldots,$ and $p_r(k)$ are associated with the respective feedback vectors. Initially, [fig. E-2]

$$p_1(o) = p_2(o) = \ldots = p_r(o) = \frac{1}{r} \qquad \text{(E21)}$$

In this way, the initial control choice is unbiased; this control choice, which is designated $\underline{P}_j^c$, may be thought of as being obtained by tossing an r-sided coin, where each side is associated with one control choice and is equally probable.

Z(o, c; m) is computed from eq. (E5); C must be specified or determined ahead of time. Next, $E\{Z(o, c; m)\}$ is computed and compared with the threshold, δ. If Z(k, c; m) is ergodic $E\{Z(o, c; m)\}$ can be computed from eq. (E13). Now, if $E\{Z(o, c; m)\} \leq \delta$ the on-line optimizations are terminated in the c^th control situation. If, on the other hand, $E\{Z(o, c;m) > \delta$ then the probabilities $p_1(0), \ldots,$ and $p_r(0)$ are updated to $p_1(1), \ldots,$ and $p_r(1)$ using eqs. (E3a) and (E3b).

Next, a decision is made as to which control is to be applied for $T<t<2T$. In the present algorithm, the control choice is obtained by choosing the feedback vector having the largest probability associated with it; hence, this means, that for some $k \geq K$,

$$\max_{i=1,\ldots,r} p_i(k) = p_j(k) \Leftrightarrow \underline{P}_\Lambda^{*c}(k;t) = \underline{P}_j^c. \qquad \text{(E22)}$$

The meaning of "for some $k \geq K$" will become clear shortly.

If $p_j(1) > p_i(1)$, all $i \neq j = 1, \ldots, r$ then the same control is applied to the system for $T<t<2T$. If, on the other hand, the most probable control is no longer $\underline{P}_j^c$ then either of the following possibilities may occur:

1. $p_i(k+1) > p_j(k+1)$ for one value of $i \neq j = 1, \ldots, r$.

2. $p_i(k+1) > p_j(k+1)$ for some values of $i \neq j = 1, \ldots, r$.

3. $p_i(k+1) > p_j(k+1)$ for all values of $i \neq j = 1, \ldots, r$.

An example that illustrates the three cases is summarized in table E-I. In the first case above, $Q = 1$, whereas in the second and third cases, $Q > 1$ and $Q = r - 1$, respectively (fig. E-2). In the latter cases one element is chosen by assuming $p_{i1}, p_{i2}, \ldots,$ and p_{iQ} are uniformly distributed (which they are). This element, which is denoted $p_j(1)$ may be thought of as being obtained by tossing a Q-sided coin, where each side is associated with an element of $\{p_{iq}\}_1^Q$. For the example in table E-I, $K = 3$; that is to say, K is the first value of k for which Case 1 above holds.

Returning to fig. E-2, the control $\underline{P}_\Lambda^*(1) = \underline{P}_j^c$ is chosen and applied to the system for $T<t<2T$, after which $Z(1, c; m)$ is computed, and the entire procedure repeats.

The sampling of $\hat{\underline{P}}_{A_a}(t|t)$ occurs every 3T units of time; thus, at least three decisions are made for each entry into a control situation.

Discussions

Implications of Eq. (E22).--Here the implications of eq. (E22) are investigated in greater detail in order to gain a better understanding of the Stochastic Automata Learning Algorithm. Suppose, for example, that for $k \geq K$ $\underline{P}_\Lambda^{*\ell}(k;t) = \underline{P}_j^{\ell}$. Under what conditions is $\underline{P}_\Lambda^{*\ell}(k+1;t)$ also equal to $\underline{P}_j^{\ell}$? The answer to this question is summarized in the following theorem, the proof of which is given below.

Theorem:

If $\underline{P}_\Lambda^{*\ell}(k;t) = \underline{P}_j^{\ell}$ and $Z(k, \ell; m) < \frac{r-1}{r}$, then $\underline{P}_\Lambda^{*\ell}(k+1;t) = \underline{P}_j^{\ell}$.

Proof:

$\underline{P}_\Lambda^{*\ell}(k;t) = \underline{P}_j^{\ell}$ means, from eq. (E22), that $p_j(k)>p_i(k)$ for all $i \neq j = 1, \ldots, r$. From eqs. (E3a) and E3b) it follows, therefore, that

$$p_j(k+1)>p_i(k+1) + (1-\theta)\left[1 - \left(\frac{r}{r-1}\right) Z(k, \ell; m)\right]. \qquad \text{(E23)}$$

Now since $Z(k, \ell; m) < \frac{r-1}{r}$, $\mathrm{l}_j(k+1) > p_i(k+1)$ for all $i \neq j = 1, \ldots, r$; hence, $\underline{P}_\Lambda^{*\ell}(k+1;t) = \underline{P}_j^{\ell}$.

An application of this theorem appears in table E-I for $k = 5$. There $Z(4, c; m) = 0.10$ which is less than $\frac{r-1}{r} = 0.75$; hence, $\underline{P}_\Lambda^{*c}(5;t) = \underline{P}_\Lambda^{*c}(4;t) = \underline{P}_2^c$. Observe, also, it does not follow if $Z(k, \ell; m) > \frac{r-1}{r}$ that $p_i(k+1) > p_j(k+1)$ for some $i \neq j = 1, \ldots, r$. This is demonstrated in table E-I for $k = 6$.

This theorem illustrates a disadvantage to selecting the control choice on the basis of eq. (E22). As long as $Z(k, \ell; m) < \frac{r-1}{r}$ a new control choice will not be made; hence, the performance may be deteriorating rapidly, that is $Z(k, \ell; m) \rightarrow \frac{r-1}{r}$, with no change in control possible. This situation is not very desirable; hence, an alternate procedure for choosing $\underline{P}_\Lambda^{*\ell}(k+1;t)$ is desired.

One such alternate approach might be to associate the probabilities $p_1(k), \ldots,$ and $p_r(k)$ with an r-sided coin. In this case, the sides are biased by the probabilities; hence, the control choices are not equally-likely. In addition, the control choice having the highest probability will not always be chosen. Convergence to such a control choice is in probability, which means

TABLE E-I
EXAMPLE ILLUSTRATING TRANSITION OF PROBABILITIES[†]

k	$p_1(k)$	$p_2(k)$	$p_3(k)$	$p_4(k)$	j	Assumed Z(k, c;m)	Comments
0	0.2500	0.2500	0.2500	0.2500	3	0.90	$\underline{P}_\Lambda^{*c}$ (0;t) = $\underline{P}_3^c$, by choice;
1	0.2750	0.2750	0.1750	0.2750	1	0.80	$\underline{P}_\Lambda^{*c}$ (1:t) = $\underline{P}_1^c$, by choice; Illustrates Case 3
2	0.2375	0.2708	0.2208	0.2708	4	0.80	$\underline{P}_\Lambda^{*c}$ (2;t) = $\underline{P}_4^c$, by choice; Illustrates Case 2
3	0.2521	0.2687	0.2437	0.2354	2	0.50	$\underline{P}_\Lambda^{*c}$ (3;t) = $\underline{P}_2^c$; Illustrates Case 1
4	0.2094	0.3844	0.2052	0.2010	2	0.10	$\underline{P}_\Lambda^{*c}$ (4;t) = $\underline{P}_2^c$; Illustrates $p_2(4) > p_i(4)$ i = 1, 3, 4
5	0.1214	0.6422	0.1193	0.1172	2	0.90	$\underline{P}_\Lambda^{*c}$ (5;t) = $\underline{P}_2^c$; Illustrates Theorem
6	0.2107	0.3711	0.2097	0.2086	2		$\underline{P}_\Lambda^{*c}$ (6;t) = $\underline{P}_2^c$ Performance getting worse

[†] $r = 4$ and $\theta = 0.50$; hence, eqs. (E3a) and (E3b) become

$$p_j(k) = 1/2\ p_j(k-1) + 1/2\ [1 - Z(k-1,\ c;\ m)] \tag{E3a'}$$

$$p_i(k) = 1/2\ p_i(k) + 1/6\ Z(k-1,\ c;\ m) \quad \text{all } i \neq j = 1,\ 2,\ 3,\ 4 \tag{E3b'}$$

$$k = 1,\ 2,\ \ldots$$

that for some $k > K_1$ only a small percentage of control choices other than the one having the highest probability will be made.

Convergence of Algorithm.--Unless eq. (E9) or the conditions in eq. (E12) hold, convergence in any sense is not guaranteed when the linear model in eq. (E3) is used as the basis for the stochastic automaton. This poses a major obstacle for the use of this model in our problem, since the system in this document is time-varying and, therefore, is nonstationary. At present, no convenient choice for $Z(k, \ell; m)$ in eq. (E5) has been found; hence, the stochastic automata algorithm is not applicable to the problem in this document although it does provide an interesting approach.

References

E1 Fu, K. S.; and McLaren, R. W.: An Application of Stochastic Automata to the Synthesis of Learning Systems. Rep. TR-EE65-17, School of Electrical Engineering, Purdue Univ., September 1965.

Appendix F

ADAPTIVE, RANDOM-OPTIMIZATION LEARNING ALGORITHM

Introduction

Here the discussion begins with the expression for the on-line-optimal feedback parameter vector, $\underline{P}_{\Lambda}^{*\ell}$ (k+1;t), and indicates heuristic techniques for choosing $\underline{\pi}$(k+1) randomly. Of primary interest is a so-called adaptive, random-optimization technique; however, a brief discussion of simple, random-optimization is in order first. Discussions on simple random-optimization are found in Idelsohn (ref. F1), Brooks (refs. F2 and F3) and Matyas (ref. F4); discussions on adaptive, random-optimization, in the context of a stationary hill-climbing problem, are found in Matyas (ref. F4); discussions in which random and gradient optimization techniques are compared, and which point out the superiority of the random technique in many circumstances, are found in Rastrigin (ref. F5) and Gurin and Rastrigin (ref. F6).

Simple Random Optimization

Simple random optimization of the sub-goal (PI(k,ℓ;m) involves a sequence of trials for $\underline{\pi}$, where $\underline{\pi}$ is now a 3 x 1 random vector with zero mean and unit correlation matrix. The kth realization of $\underline{\pi}$ is denoted $\underline{\pi}^{(k)}$

For the purposes of a simple, random-optimization learning algorithm, eq. (81) is rewritten here as

$$\underline{P}_{\Lambda}^{*\ell}(k+1) = \underline{P}_{\Lambda}^{*\ell}(k-1) + y(k)\,\underline{\pi}^{(k)} + \underline{\pi}^{(k+1)}$$

$$k = 0, 1, \ldots, \qquad \text{(F1)}$$

where

$$\underline{\pi}^{(0)} \triangleq \underline{P}_{\Lambda}^{\ell} = \underline{P}_{\Lambda}^{*\ell}(0) \qquad \text{(F2a)}$$

$$\underline{P}_{\Lambda}^{*\ell}(-1) \triangleq \underline{0} \qquad \text{(F2b)}$$

and y(k) is the decision function that is defined in eq. (83), with

$$y(0) \triangleq 1. \qquad (F3)$$

The effect of the decision function is to remove unsuccessful choices for $\underline{\pi}$ from succeeding iterations of $\underline{P}_\Lambda^{*\ell}$.

A brief description of a heuristic learning algorithm that is based upon a simple, random-optimization procedure follows. The convergence of this algorithm is proved in Matyas (ref. F4) for a stationary, calculus, hill-climbing problem. Questions related to the convergence of the algorithm (and also to the adaptive, random-optimization algorithm discussed in the next paragraph), as applied to our problem, a dynamic optimization problem, have not been answered, since they were not within the scope of the present study.

Observe, from eq. (F2), that the nominal control $\underline{P}_\Lambda^{\ell}$ serves as a starting point for the on-line optimizations. $\underline{P}_\Lambda^{\ell}$ is applied initially for $0<t<T$ and PI(o,ℓ;m) is then computed. A value of $\underline{\pi}$ is chosen and the control

$$\underline{P}_\Lambda^{*\ell}(1;t) = \underline{P}_\Lambda^{\ell} + \underline{\pi}^{(1)} \qquad (F4)$$

is applied to the system for $T<t<2T$. PI(1,ℓ;m) is then computed and compared with PI(o,ℓ;m).

If PI(1,ℓ;m) $\leq$ PI(o,ℓ;m) $-\epsilon'(1)$ then y(1) is unity, which means that the choice for $\underline{\pi}$, $\underline{\pi}^{(1)}$ is satisfactory; hence, $\underline{\pi}^{(1)}$ is retained. A value for $\underline{\pi}$ is chosen and the control

$$\underline{P}_\Lambda^{*\ell}(2;t) = \underline{P}_\Lambda^{\ell} + \underline{\pi}^{(1)} + \underline{\pi}^{(2)} \qquad (F5)$$

is applied to the system for $2T<t<3T$.

If on the other hand, PI(1,ℓ;m) > PI(o,ℓ;m) $-\epsilon'(1)$, y(1) is set equal to zero, which means that the choice for $\underline{\pi}$, $\underline{\pi}^{(1)}$, is unsatisfactory. A new value is chosen for $\underline{\pi}$, and the control

$$\underline{P}_\Lambda^{*\ell}(2;t) = \underline{P}_\Lambda^{\ell} + \underline{\pi}^{(2)} \qquad (F6)$$

is applied to the system for $2T<t<3T$. This procedure is repeated until PI(k,ℓ;m) $\leq \delta(\epsilon)$.

In general, the probability of a successful step using simple, random optimization is 1/2. Matyas (ref. F4) has demonstrated, however, that the convergence can be speeded up by incorporating learning and memory into the above procedure. A modified version of his technique is discussed next.

Adaptive, Random-Optimization

Adaptive, random optimization of $PI(k,\ell;m)$ involves a sequence of trials for $\underline{\pi}(k+1)$, where $\underline{\pi}(k+1)$ is now a discrete stochastic process which has a variable mean value and a variable correlation matrix. These variables are adjusted in a learning process that is based upon past successes and failures. The effect of this is to increase the probability of a successful trial; this is achieved by learning the most favorable direction and step size for an iteration of the feedback parameter vector.

$\underline{\pi}(k+1)$ is now given by the expression

$$\underline{\pi}(0) = \underline{P}_{\Lambda}^{\ell} \tag{F7}$$

$$\underline{\pi}(k+1) = \underline{d}(k+1) + T(k+1)\,\underline{\mu} \qquad k=0, 1, \ldots \tag{F8}$$

where

$\underline{\mu}$ is a (3 x 1) random vector with zero mean and unit correlation matrix, $T(k+1)$ is a transformation matrix whose elements can be selected so as to control the standard deviation of the vector $T(k+1)\,\underline{\mu}$,

and

$\underline{d}(k+1)$ is a mean value vector given by the expression

$$\underline{d}(1) = \underline{0} \tag{F9}$$

$$\underline{d}(k+1) = C_o\Big(y(k)\Big)\,\underline{d}(k) + C_1\Big(y(k)\Big)\,\underline{\pi}^{(1)}(k) \tag{F10}$$

where

C_0 and C_1 are functions of the decision function defined in eq. (83) and in the sequel are assumed given by the relations

$$C_o\Big(y(k)\Big) = \frac{1}{2} + \frac{1}{2}\,y(k) \tag{F11a}$$

and

$$C_1\left(y(k)\right) = \frac{1}{2} y(k). \qquad \text{(F11b)}$$

The general conditions which C_0 and C_1 should satisfy are found in Matyas (ref. F4). Choosing $\underline{d}(1) = \underline{0}$ tends to leave $\underline{\pi}(1)$ unbiased. The realization of $\underline{\pi}(k)$ in eq. (F10), $\underline{\pi}^{(1)}(k)$, follows from eq. (F8). The final expression for $\underline{d}(k+1)$ is

$$\underline{d}(k+1) = \left[\frac{1}{2} + \frac{1}{2}y(k)\right] \underline{d}(k) + \frac{1}{2}y(k)\ T(k)\ \underline{\mu}^{(k)}$$

$$k=1, 2, \dots \qquad \text{(F12)}$$

Before proceeding to a discussion of an adaptive, random-optimization learning algorithm, the implications of choosing C_0 and C_1 as in eqs. (F11a) and (F11b), respectively, are discussed. Two cases are distinguished: (1) y(k)=0--poor choice for $\underline{\pi}(k)$, and (2) y(k)=1--acceptable choice for $\underline{\pi}(k)$.

When $y(k) = 0$ $C_0 = \frac{1}{2}$ and $C_1 = 0$ and $\underline{d}(k+1)$ becomes

$$\underline{d}(k+1) = \frac{1}{2} \underline{d}(k) \qquad \text{(F13)}$$

In this case, the mean value of $\underline{\pi}(k+1)$, d(k+1), can be viewed as being in a direction opposite to the preceding mean value $\underline{d}(k)$, hence, C_0 is called the <u>rejection coefficient</u>.

When $y(k) = 1$ $C_0 = 1$ and $C_1 = \frac{1}{2}$ and $\underline{d}(k+1)$ becomes

$$\underline{d}(k+1) = \underline{d}(k) + \frac{1}{2}\ T(k)\ \underline{\mu}^{(k)} \qquad \text{(F14)}$$

In this case, the mean value of $\underline{\pi}(k+1)$ is chosen in the same direction as the preceding mean value. Observe that for an unsuccessful step $C_1 = 0$ and for a successful step $C_1 > 0$; hence, C_1 is called the <u>detection coefficient</u>.

A Learning Algorithm

An on-line learning algorithm is summarized in fig. F-1. Included on the figure is a control situation algorithm, one which has been discussed in connection with the Error-Correction Learning Algorithm in Section 4, of Part 2. A brief description of the flow diagram follows.

The system is allowed to run for T units of time under the action of $\underline{P}_\Lambda^c$. $\underline{P}_\Lambda^c$ is the nominal control for a new control situation and is the stored value of $\underline{P}_\Lambda^{*c}$ associated with the last entry into an existing control situation. The on-line performance index is evaluated for $0<t<T$. Rather than let the system continue under the same control for $T<t<2T$, as in the Error-Correction Learning Algorithm, $\underline{P}_\Lambda^{*c}$ is updated by $\underline{\pi}(1)$, for $T<t<2T$.

The value of $\underline{\pi}(1)$ is computed from eq. (F8) by choosing $\underline{\mu}\left[\underline{\mu}^{(1)}\right]$ and specifying $\underline{d}(1)$. It is assumed in the sequel that T(k+1) is the identity matrix (3 x 3) [time did not permit the investigation of an algorithm which updates T(k+1)]. If the c^{th} control situation is entered for the first time $\underline{d}(1)$ is set equal to zero [see eq. (F9)]. If, on the other hand, the c^{th} control situation is being re-entered, $\underline{d}(1)$ is set equal to the last value of $\underline{d}$ that is stored in a memory compartment associated with the c^{th} control situation; hence, eq. (F9) should read $\underline{d}(1) = \underline{0}$ when m = 1.

The on-line performance index is computed for $0<t<2T$ and is then compared with the performance for $0<t<T$. If the performance for $0<t<2T$ is better than that for $0<t<T$, y(1) is set equal to unity, $\underline{d}(2)$ is computed from eq. (F12), $\underline{\mu}$ is chosen $\left[\underline{\mu}^{(2)}\right]$, and $\underline{\pi}(2)$ is computed from eq. (F8). Finally, if the performance for $0<t<2T$ is worse than that for $0<t<T$, y(1) is set equal to zero, $\underline{d}(2)$ is computed, $\underline{\mu}$ is chosen $\left[\underline{\mu}^{(2)}\right]$, and $\underline{\pi}(2)$ is computed.

In both of these cases, once $\underline{\pi}(2)$ has been computed, the new $\underline{P}_\Lambda^{*c}$, which is designated $\underline{P}_\Lambda^{*c}(k_m)$, is stored. The on-line performance index is then evaluated for $0<t<3T$. PI(2, c;m) is compared with PI(1, c;m) and the entire procedure repeats.

Here, as in the Error-Correction Learning Algorithm, the estimates of the plant parameters are sampled every 3T units of time.

In conclusion, the following observations are noted: it is possible to distinguish two levels of learning in the proposed on-line-learning controller. The first level is associated with learning the on-line-optimal feedback parameter vectors $\underline{P}_\Lambda^{*\ell}(t)$ for each control situation ($\ell = 1, 2, \ldots$). The second level, which is lower in the hierarchy, is associated with learning optimum step sizes and directions for each control situation. The second level of learning represents, in effect, an attempt to learn <u>how to learn</u> $\underline{P}_\Lambda^{*\ell}(t)$.

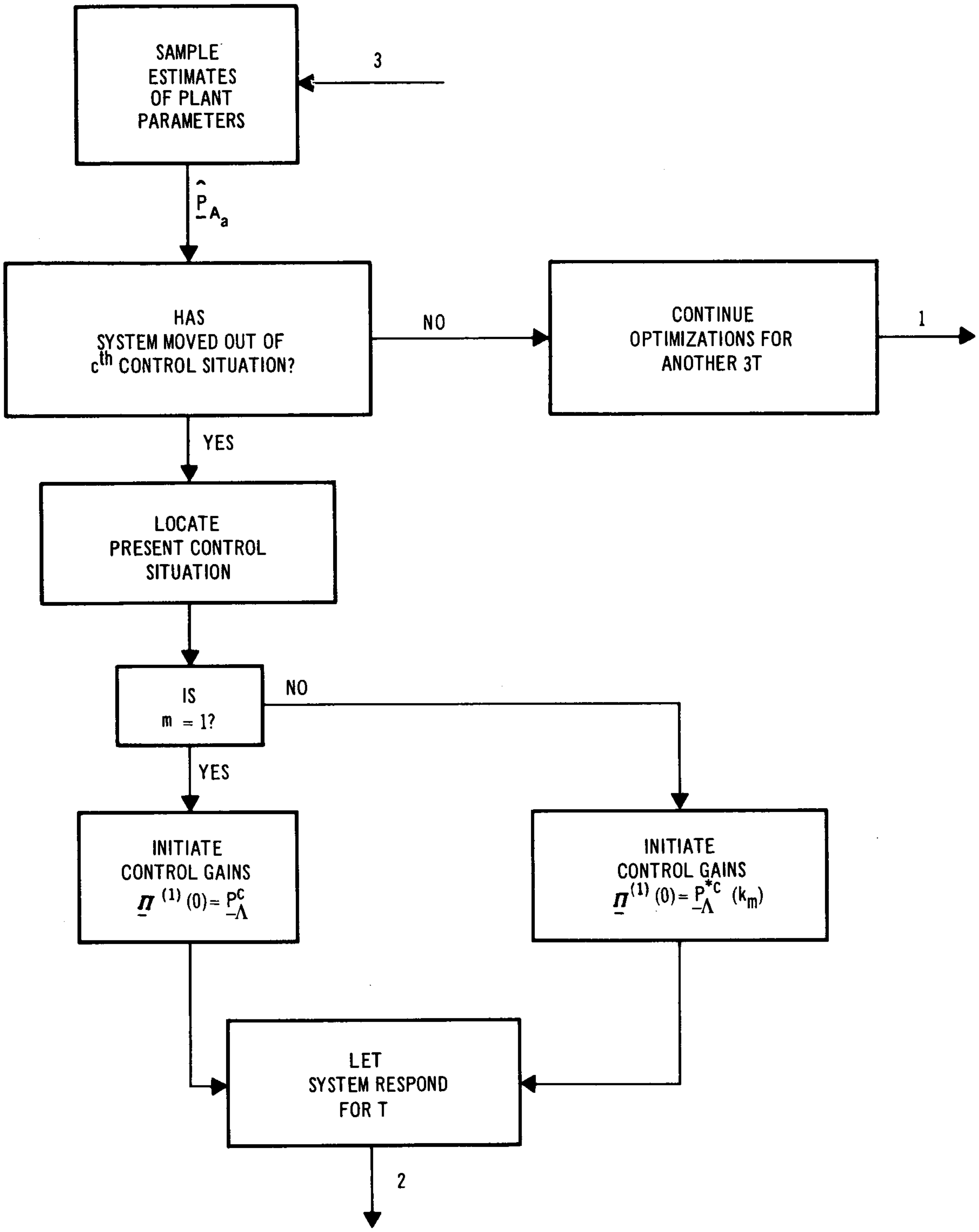

SAMPLE ESTIMATES OF PLANT PARAMETERS
3
$\hat{\underline{P}}_{A_a}$
HAS SYSTEM MOVED OUT OF c^{th} CONTROL SITUATION?
NO
CONTINUE OPTIMIZATIONS FOR ANOTHER 3T
1
YES
LOCATE PRESENT CONTROL SITUATION
IS m = 1?
NO
YES
INITIATE CONTROL GAINS $\underline{\Pi}^{(1)}(0) = \underline{P}^{c}_{\Lambda}$
INITIATE CONTROL GAINS $\underline{\Pi}^{(1)}(0) = \underline{P}^{*c}_{\Lambda}(k_m)$
LET SYSTEM RESPOND FOR T
2

Figure F-1. Flow Diagram for Adaptive, Random-Optimization Learning Algorithm

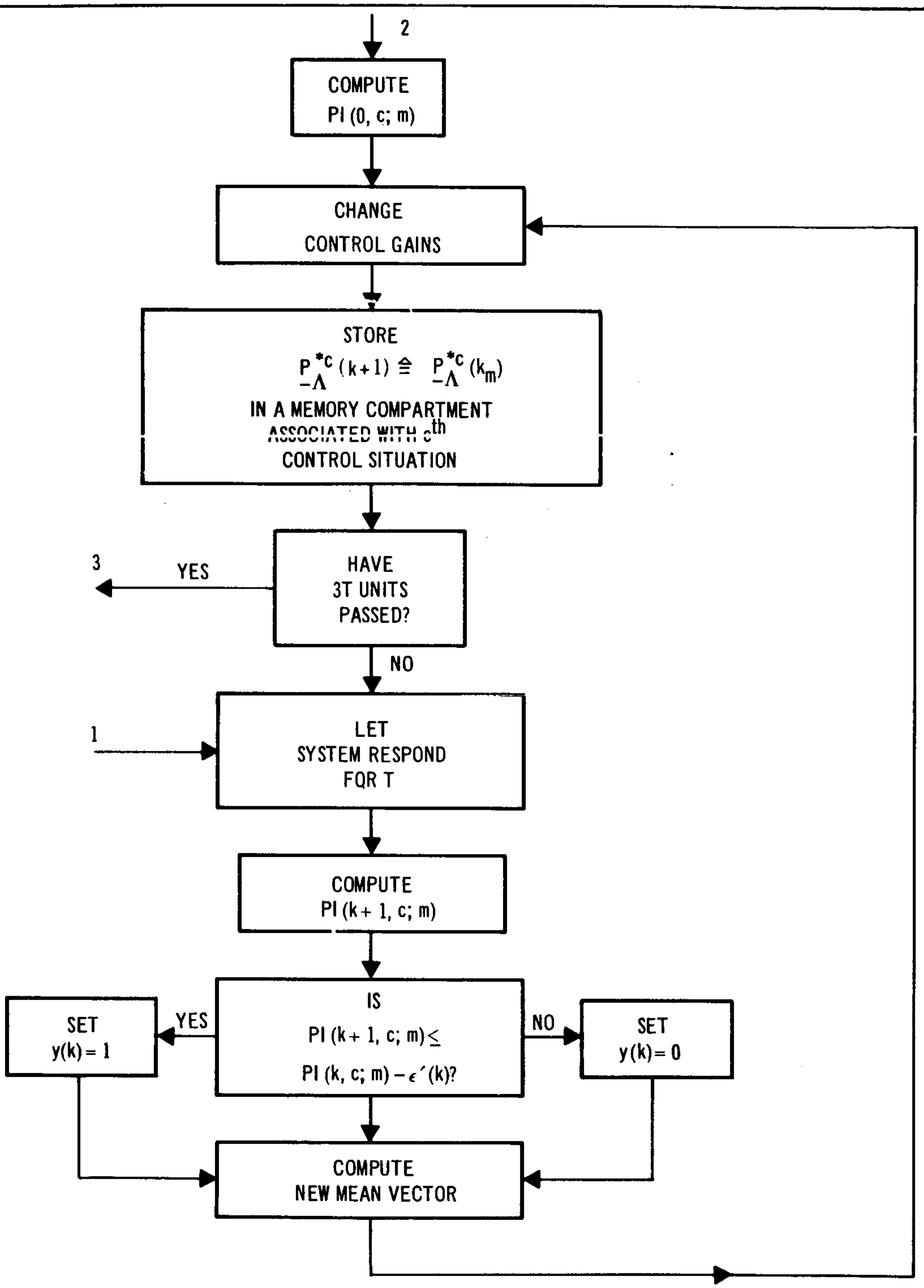


Figure F-1. (Continued)

References

F1. Idelsohn, J. M.: 10 Ways to Find the Optimum. Control Engineering, June, 1964.

F2. Brooks, S. H.: A Comparison of Maximum Seeking Methods. Operations Research, Vol. 7, No. 4, 1959.

F3. Brooks, S. H.: A Discussion of Random Methods for Seeking Maxima. Operations Research, Vol. 6, No. 21, March-April, 1958.

F4. Matyas, J.: Random Optimization. Automation and Remote Control, Vol. 26, No. 2, Feb. 1965, pp. 244-251.

F5. Rastrigin, L. A.: The Convergence of the Random Search Method in Extremal Control of a Many-Parameter System. Automation and Remote Control, Vol. 24, No. 11, Nov. 1963, pp. 1337-1342.

F6. Gurin, L. S., and Rastrigin, L. A.: Convergence of the Random Search Method in the Presence of Noise. Automation and Remote Control, Vol. 26, No. 9, Sept. 1965, pp. 1505-1511.

NASA-Langley, 1967 —— 10 CR-755