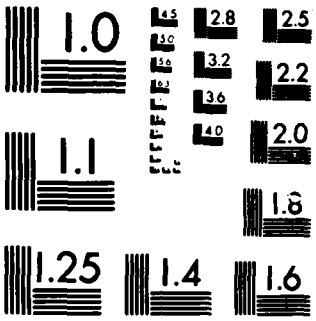
AD-A163 945 PARTICLE BEAM TRACKER FOR AN ACCELERATING TARGET(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING L C JAMERSON DEC 85 AFIT/GE/ENG/85D-22 1/2

UNCLASSIFIED F/G 20/7 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD-A163 945

DTIC
ELECTE
FEB 1 2 1986
S D
D

PARTICLE BEAM TRACKER FOR AN

ACCELERATING TARGET

THESIS

Lawrence C. Jamerson
Second Lieutenant, USAF

AFIT/GE/ENG/85D-22

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

DTIC FILE COPY

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY
AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

86 2 11 132

AFIT/GE/ENG/85

DTIC
ELECTE
FEB 1 2 1986
S D
D

PARTICLE BEAM TRACKER FOR AN ACCELERATING TARGET

THESIS

Lawrence C. Jamerson
Second Lieutenant, USAF

AFIT/GE/ENG/85D-22

Approved for public release; distribution unlimited

AFIT/GE/ENG/85D-22

PARTICLE BEAM TRACKER FOR AN ACCELERATING TARGET

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Electrical Engineering

Accesion For	
NTIS CRA&I	☑
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and / or Special
A-1	

QUALITY INSPECTED 3

Lawrence C. Jamerson, B.S.

Second Lieutenant, USAF

December 1985

Approved for public release; distribution unlimited

Preface

This thesis was under the sponsorship of the Air Force Weapons Laboratory at Kirtland Air Force Base. It is a continuation of previous efforts by Captain David Meer in his Doctoral Dissertation and Lieutenants William Zicker and Bill Moose in their Master's theses. Specifically, the study involves a parameter sensitivity and robustness analysis for the corrected PI controller originally conceived by Moose in addition to a robustness study which incorporates a first order Gauss-Markov acceleration target model.

I would like to thank my thesis advisor, Dr. Peter S. Maybeck, who provided guidance and technical support so that this thesis effort would be complete.

Lawrence C. Jamerson

Table of Contents

	Page
Preface	ii
List of Figures	v
List of Tables	vii
Abstract	viii
I. INTRODUCTION	1-1
I.1 MOTIVATION AND BACKGROUND	1-1
I.2 PROBLEM STATEMENT	1-2
I.3 SCOPE	1-5
I.4 APPROACH	1-6
I.5 SUMMARY OF REMAINING CHAPTERS	1-7
II. MEER FILTER	2-1
II.1 PARTICLE BEAM MODEL	2-1
II.2 SNYDER-FISHMAN FILTER	2-4
II.3 MULTIPLE MODEL ADAPTIVE ESTIMATOR	2-7
II.3.1 GENERAL MMAE	2-7
II.3.2 PARTICLE BEAM MMAE	2-11
II.4 WEIGHTING FACTORS	2-16
II.5 LIMITING THE HYPOTHESIS TREE	2-19
II.5.1 THE BEST HALF METHOD	2-19
II.5.2 MERGE METHOD	2-22
II.6 SUMMARY	2-23
III. PI CONTROLLER	3-1
III.1 PI REGULATOR	3-2
III.2 TRACKER MODEL AND KALMAN FILTER	3-8
III.3 PI TRACKER	3-14
III.4 SUMMARY	3-18
IV. ANALYSIS TOOLS AND METHODS	4-1
IV.1 ANALYSIS TOOLS	4-1
IV.2 ANALYSIS METHOD	4-5
IV.3 SENSITIVITY ANALYSIS	4-7
IV.4 ROBUSTNESS ANALYSIS	4-9
IV.5 SUMMARY	4-12

Page

V. RESULTS 5-1

V.1 FIRST ORDER GAUSS-MARKOV POSITION MODEL 5-2

V.1.1 CHOICE OF WEIGHTING MATRICES 5-2
V.1.2 PARAMETER SENSITIVITY 5-5
V.1.3 ROBUSTNESS TO DISTURBANCE INPUTS 5-10
V.1.4 GENERAL COMMENTS 5-14

V.2 FIRST ORDER GAUSS-MARKOV ACCELERATION MODEL 5-14

V.2.1 CHOICE OF WEIGHTING MATRICES 5-14
V.2.2 ROBUSTNESS TO PARAMETER VARIATIONS . . . 5-16

V.3 SUMMARY. 5-27

VI. CONCLUSIONS AND RECOMMENDATIONS 6-1

VI.1 CONCLUSIONS 6-1
VI.2 RECOMMENDATIONS FOR FURTHER WORK 6-2

References R-1

VITA V-1

List of Figures

Figure		Page
1-1	Performance Analysis for PI Controller	1-8
2-1	Multiple Model Adaptive Estimator	2-10
2-2	Hypotheses After First Event	2-13
2-3	Hypotheses After Second Event	2-15
2-4	Best Half Method of Hypothesis Limiting	2-20
2-5	Merge Method of Hypothesis Limiting	2-24
3-1	General Structure of PI Controller	3-5
3-2	Riccati Equation Solution	3-9
3-3	First Order Gauss-Markov Target Position Model	3-10
3-4	First Order Gauss-Markov Target Acceleration Model	3-12
3-5	Filter/Controller Structure	3-19
5-1	Controller Response with Heavy Weighting on Integral Term	5-3
5-2	Controller Response with Proper Weighting of Integral and Error Terms	5-4
5-3	Sensitivity Analysis: Tracker RMS Errors vs Target Model Driving Noise	5-6
5-4	Sensitivity Analysis: Tracker RMS Errors vs Target Measurement Noise	5-8
5-5	Sensitivity Analysis: Tracker RMS Errors vs Target Position Time Constant	5-9
5-6	Robustness Analysis: Tracker RMS Error vs Constant Unmodelled Disturbance Magnitude	5-11
5-7	Robustness Analysis: Tracker RMS Error vs Frequency of Unmodelled Sinusoidal Disturbances	5-13
5-8	Robustness Analysis: Tracker RMS Error vs Magnitude of Unmodelled Sinusoidal Disturbances	5-15

Figure		Page
5-9	Robustness Analysis: Tracker RMS Error vs True Target Model Driving Noise	5-19
5-10	Robustness Analysis: Tracker RMS Error vs True Target Model Measurement Noise	5-21
5-11	Robustness Analysis: Tracker RMS Error vs True Target Time Constant (First)	5-23
5-12	Robustness Analysis: Tracker RMS Error vs True Target Time Constant (Second)	5-25
5-13	PSD Curve and RMS Acceleration	5-28
5-14	Robustness Analysis: Tracker RMS Error vs True Target Time Constant (Third)	5-29

List of Tables

Table		Page
5-1	Average RMS Tracking Errors with Variations in Q_T of Target	5-18
5-2	Average RMS Tracking Errors with Variations in R of Target	5-18
5-3	Average RMS Tracking Errors with Variations in τ of Target (#1)	5-24
5-4	Average RMS Tracking Errors with Variations in τ of Target (#2)	5-24
5-5	Average RMS Tracking Errors with Variations in τ of Target (#3)	5-26

AFIT/GE/ENG/85D-22

Abstract

The purpose is to use a PI controller to point the centroid of a particle beam at an intended target. Multiple Model Adaptive Estimator is used to estimate the centroid of a one-dimensional Gaussian shaped source of photo-electron events. "Merge Method" of filter pruning is used to limit the size of this filter. A standard Kalman filter is used to estimate the position of the target where the target is initially modelled as a first-order Gauss-Markov position process and later as a first-order Gauss-Markov acceleration process. A PI controller is designed using LQG methods, and true states are replaced by their best estimates by invoking the principle of assumed certainty equivalence. With a target position model within the software, a parameter sensitivity analysis is performed as well as a robustness study where unmodelled constant and sinusoidal disturbances are added to the system. With a target acceleration model within the software, a robustness study is carried out where target parameters are varied within the truth model without telling the filter. In an attempt to recover full state feedback robustness qualities, Loop Transmission Recovery tuning is attempted.

PARTICLE BEAM TRACKER FOR AN ACCELERATING TARGET

I. INTRODUCTION

I.1 MOTIVATION AND BACKGROUND

There is presently a great deal of research within the military and industrial communities aimed at determining the feasibility of the proposed Strategic Defense Initiative (SDI). At the heart of this program is a technology known as Directed Energy Weapons, and one of the primary classes of these weapons involves particle beams. In addition to the problems of generating and propagating the beams, there is also the issue of beam control (Ref. 1). Not only must the beam be regulated to reject undesired inputs such as wind gusts and other physical disturbances, it must also be able to track an intended target. In order to achieve regulation and tracking characteristics, the controller must contain some type of feedback to enhance system performance and to promote stability robustness--maintaining stability in the face of real-world changes to the basic system. However, before control can be attempted, some method of measuring beam location must be achieved.

One proposal for measuring the particle beam's location is to illuminate it using an incident laser. The transfer of energy from the laser to individual particles within the particle beam results in a spontaneous emission of photons as electrons decay from their excited state to a ground state. These photons appear as individual

photo-electron events on a detector array, whereby some knowledge of the beam's location can be inferred.

These photo-electron events can be well modelled as Poisson space-time point processes. Snyder and Fishman (Ref. 15) developed an estimator which specifically handled the case of measurements that appear as Poisson point processes. However, their filter assumed that all measurements were signal-induced, thereby eliminating the practicality of a single such filter being used in an environment where there might be noise-induced events such as "dark current"--detecting a signal when there is no signal present (Ref. 9). In 1982, Captain David Meer, in his doctoral dissertation (Ref. 9), accomplished the design of an estimator which could be specifically applied to the particle beam problem. His final result was a multiple model adaptive estimator (MMAE) which contained a bank of Snyder-Fishman filters. Each filter contained a different assumption as to the sequence of signal and noise events, i.e., whether each event in the sequence was signal-induced or noise-induced. Meer also developed a method of calculating the weighting factors necessary to create the overall adaptive estimate as the weighted sum of the outputs of each filter. In this manner he was able to avoid the inadequacy of a single Snyder-Fishman filter based on a noise-free model, plus account for variations of certain parameters within the models.

I.2 PROBLEM STATEMENT

With this capability of particle beam detection and position estimation in hand, the problem of beam control can now be addressed. In 1983, 1Lt William Zicker, in his master's thesis (Ref. 18), approached this problem with a proportional gain controller. A deterministic

optimal LQ controller (based on a linear system model and using a quadratic performance index for optimization) was designed assuming full state feedback, and then these states were replaced with their best estimates as the principle of assumed certainty equivalence (Ref. 4: Vol. 3) was applied to synthesize a stochastic controller. These state estimates were provided by the Meer filter which modelled the beam centroid location as the output of a linear system model driven by white Gaussian noise with a one dimensional state vector--position along one axis. The Kalman filter which provided estimates of the target's state, incorporated a similar simplified model. Simplification was reasonable because Zicker's work was a feasibility study to see if Meer's filter could be used in a beam controller.

Another problem studied by Zicker was how to limit the size of the Meer filter, which grows with each new realization on the photodetector array. Meer suggested a method of "pruning" the MMAE while Zicker showed the superiority of "merging", a technique originally conceived by Weiss at MIT (Ref. 17). Both these techniques will be discussed more fully within the next chapter.

In 1984, Captain William Moose (Ref. 10) conducted a similar study using a proportional plus integral (PI) controller and compared it to Zicker's work. The PI controller possesses several characteristics which make it ideal for the tracking problem. First, control is based on tracking errors which have occurred in the past as well as those which are occurring at the present time. Second, type-1 systems are capable of tracking a constant input with zero steady state error even in the face of unmodelled constant disturbances. The design was achieved by

again employing the principle of assumed certainty equivalence in conjunction with the simplified design models noted previously. The final result was a tracking system which displayed superior performance (relative to Zicker's proportional gain controller) yet exhibited sluggishness due to the additional integration term in the loop transmission. Although Moose's results exhibited enhanced performance, there was a significant flaw in his software for minimizing the quadratic cost function. When the goal of an optimal controller is to minimize the error between a controlled state and a desired target state, off-diagonal terms appear in the weighting matrix. Unfortunately, Moose neglected to include these terms, thereby computing controller gains which were not optimal for this problem.

The work of Meer, Zicker, and Moose showed that effective estimation of particle beam states is possible and that control of the beam can be synthesized using the principle of assumed certainty equivalence. However, the feasibility was demonstrated at the cost of simplified models; therefore, the next logical question to ask is whether practical designs can be achieved. By increasing the descriptive adequacy and size of the filter models and including such effects as actuator dynamics, the problem characteristics become closer to what might actually be expected in the real world. Another practical consideration is whether the controller can maintain stability robustness in the face of real world plant variations. When the controller is designed, certain models and conditions are assumed. However, when this system is implemented on an actual plant, the model parameters might vary substantially (as well as in a small perturbation sense) from the assumed design conditions. When

these perturbations occur, it is essential to maintain stability, and it is desirable to maintain good performance in the closed loop system. These attributes must be verified.

I.3 SCOPE

This project begins by embedding a new target model into the Kalman filter used to estimate the position of a target. The new Kalman filter contains a three-state acceleration model for the single axis case, letting target acceleration be modelled as a first order Gauss-Markov process. Note that the Meer filter remains unchanged. With the estimators equipped for more practical applications, controller design is begun. Two areas which receive a great deal of attention during this thesis are the robustness of the controller and the proper selection of cost function weighting matrices used in evaluating controller gains. To examine robustness, certain important parameters are varied in the truth model, and the resulting controller response is examined. If the response is not suitable, Loop Transmission Recovery (LTR) techniques (Refs. 6 and 16) are employed in order to retune the Meer Filter. If these techniques are not successful, adaptive estimation of the uncertain parameters will be considered. As far as selection of the cost function weighting matrices is concerned, Moose's software is amended to include off diagonal terms as discussed previously. Through the use of the revised software, alternate weighting matrices are examined in order to yield the desired responses. Since weighting matrix choice is also tied to the robustness issue, the desire to maintain stability and performance despite changes in the real-world system from assumed design conditions will influence our choices.

I.4 APPROACH

The approach to designing a controller of this type is a three-step process. First, the PI controller is designed using optimal control techniques. The models for the beam and target systems are assumed to be linear and a quadratic cost function (performance index) is the optimizing criterion. This is the standard LQ technique where the controller gains can be calculated using a backward running Riccati difference equation (the system is discretized). The controller is designed assuming complete access to all the states of the beam and target. The second step in the process is to form state estimators for both the beam and the intended target. The Meer filter provides position state estimates for the beam while a standard Kalman filter is used to estimate the target's states. The third and final step is to replace the full state feedback assumption with the best estimates of these states as supplied by our filters. Because this is not an LQG problem (due to the Meer filter), we invoke the principle of assumed certainty equivalence to allow this final step.

As was done in previous theses, system performance is studied using computer simulation. Because the particle beam problem does not involve a purely linear system and linear filter/controller, a standard covariance analysis cannot be used for data analysis. Therefore, system performance is evaluated using Monte Carlo simulations in order to yield the necessary statistical data. Because we are dealing with a Poisson point process, each simulation entails approximately 200 runs each instead of the usual 15 or so typical of many aerospace applications.

The adequacy of this number of runs was shown in Zicker's thesis (Ref. 18) and need not be repeated here.

A Monte Carlo simulation can be divided into two separate parts. First, there is the "truth model", which is a highly detailed representation of real world dynamics. This model is the result of careful testing of the real world system. Second, there is the filter/controller combination which is designed using a reduced order model and the selected weighting matrices noted earlier. The "truth model" is used to drive the filter/controller configuration in order to provide realistic inputs so that system performance can be evaluated. This structure is displayed in Figure 1-1. In the figure, the "truth model" uses the true measurement noise $\underline{V}_t$, true process noise $\underline{w}_t(t)$, and feedback controls $\underline{u}(t)$ to form the true states $\underline{x}_t(t)$. These true states indicate where the beam and target are actually located. The truth model provides noise corrupted measurements to the Meer and Kalman filters which in turn attempt to estimate these states ($\hat{\underline{x}}(t)$). By invoking the principle of assumed certainty equivalence, the state estimates can be used with the PI controller to provide the control inputs to the truth model. The error term $\underline{e}(t)$ provides a method for analyzing state estimation accuracy. However, $\underline{x}_t(t)$ and $\underline{u}(t)$ are the important quantities to observe when the performance analysis of a proposed controller is desired.

I.5 SUMMARY OF REMAINING CHAPTERS

The remaining chapters begin with a discussion of the Meer filter in Chapter II. A complete derivation is provided, including the Snyder-Fishman filter equations, the MMAE configuration, weighting factor calculations, and filter "pruning" techniques. Chapter III develops

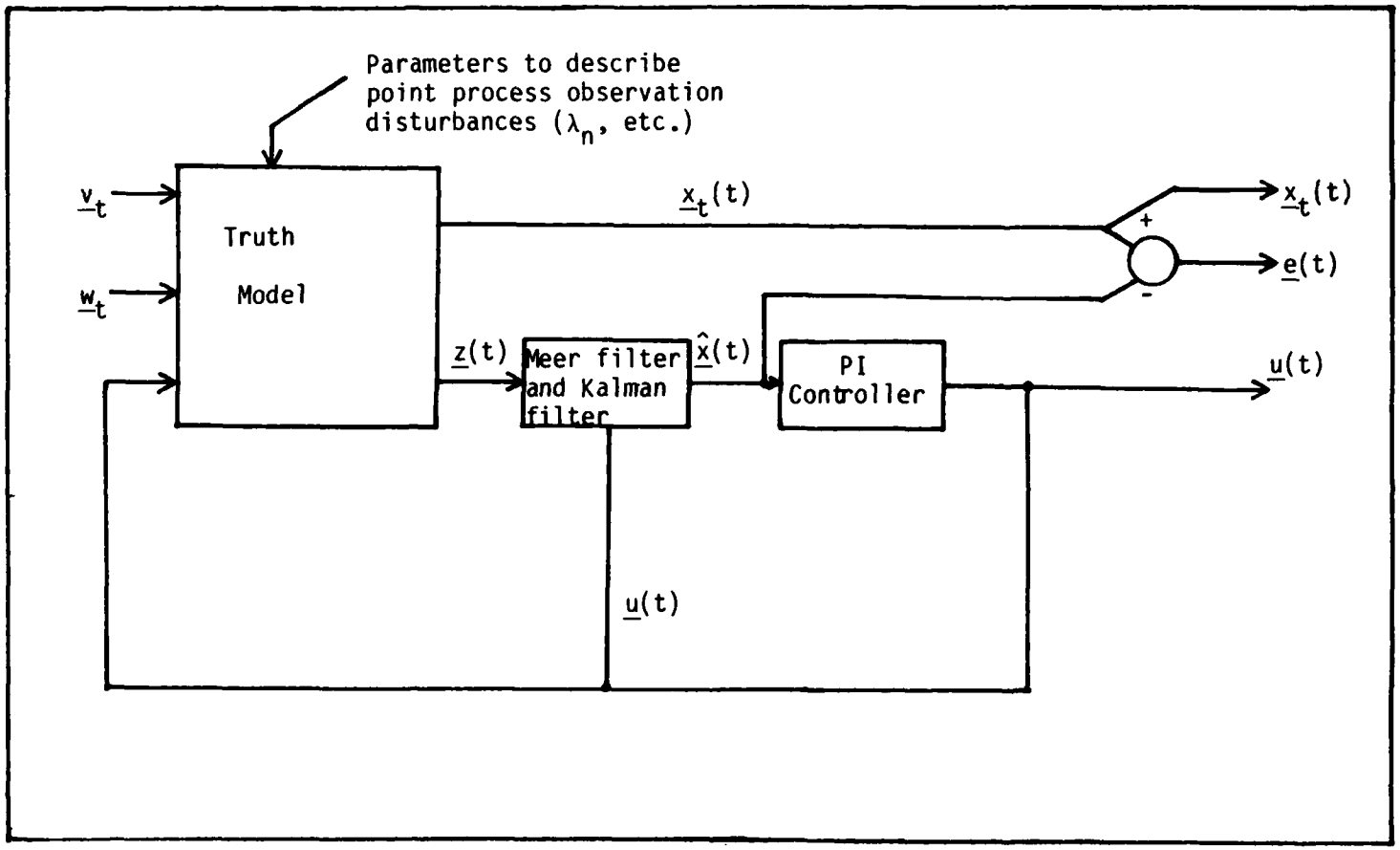


Figure 1-1. Performance Analysis for PI Controller

the PI controller to be used in this thesis. This controller incorporates a first order Gauss-Markov acceleration model for the target. Chapter IV describes the analysis techniques used, while Chapter V is a presentation of results. Chapter VI provides conclusions and recommendations for further study.

II. MEER FILTER

The purpose of this chapter is to present a detailed description of the Meer filter (Refs. 5 and 9). Recall that this is the estimator which was specifically created for the particle beam pointing and tracking problem. As with all filters, there must first be a design model upon which the algorithms are based. Once the model is chosen, the filter structure can be developed. This development includes justification for the MMAE configuration as well as a discussion of the component Snyder-Fishman filters and weighting matrix equations. The chapter concludes with a presentation of the two methods for limiting the size of the Meer filter, which is imperative if computational loading is to be considered. Throughout the chapter, mathematical rigor is employed so that this section might be understood without the use of reference aids. However, these references will be provided for the reader who desires additional background.

II. 1 PARTICLE BEAM MODEL

Estimation of the particle beam's location begins by shining a laser at the beam and observing the individual photo-electron events as the particles transition from an excited energy state to their ground state. These events can be recorded on a photo-detector array; however, the regularity of measurements is neither continuous nor at predetermined discrete time intervals. Rather, Snyder and Fishman noted that the recorded measurements appear as Poisson space-time point processes (Ref. 15:692). That is to say, each measurement has associated with it

a particular time of occurrence t and spatial location $\bar{r}$ in the m-dimensional vector space (i.e., $\bar{r}_i \epsilon R^m$); for a two-dimensional detector array, m=2. A history of observations might therefore appear as

$$[(t_1,r_1),(t_2,r_2), \ldots\ldots,(t_{N(t)},r_{N(t)})]$$

where N(t) is the total number of events observed.

In order to progress into the work of Snyder and Fishman (Ref. 15), each of these observations is assumed to be purely signal-induced. With the nature of the observations firmly in hand, the next step is to examine signal rates. For this problem, the signal rate parameter which describes rates of occurrence can be assumed to be spatially distributed in a Gaussian manner, with the equation as a function of spatial variable $\bar{r}$ given as (Ref. 9:18):

$$\lambda_s(t,\bar{r},\underline{\bar{x}}(t)) = \Lambda(t)\exp\{-[\bar{r}-\underline{H}(t)\underline{\bar{x}}(t)]^T\underline{R}^{-1}(t)[\bar{r}-\underline{H}(t)\underline{\bar{x}}(t)]/2\} \quad (2\text{-}1)$$

where

Λ is the maximum value of the gaussian curve.

$\underline{R}(t)$ is symmetric positive definite matrix which depicts the spread of the curve.

$\underline{H}(t)\underline{\bar{x}}(t)$ represents the assumed location of the beam centroid in the m-dimensional detector space.

$\underline{H}(t)$ is an m x n projection matrix which relates $\underline{\bar{x}}(t)$ to the beam centroid.

$\underline{\bar{x}}(t)$ is a stochastic process (as noted by the under tilde) which represents the state dynamics of the beam centroid.

The state process can be modelled as the Gaussian output of the linear stochastic differential equation:

$$d\bar{\underset{\sim}{x}}(t) = \underline{F}(t)\underset{\sim}{x}(t)dt + \underline{G}(t)d\bar{\underset{\sim}{\mu}}(t) \qquad (2\text{-}2)$$

$$\bar{\underset{\sim}{x}}(t_0) = \bar{\underset{\sim}{x}}_0$$

where $\bar{\underset{\sim}{\mu}}(t)$ is a Wiener process (Brownian motion) with a diffusion $\underline{Q}(t)=\underline{I}$, and $\bar{\underset{\sim}{x}}_0$ is a Gaussian random vector with mean $\bar{x}_0$ and covariance $\underline{P}_0$. There is no loss of generality in letting this be a unit-diffusion process, since, if we let $\underline{G}=\underline{G}'Q$ so that $\underline{GG}^T=\underline{G}'\underline{Q}'\underline{G}'^T$, non-unit diffusions can be readily addressed.

Note that the equations for the signal rate parameter and the state variable $\bar{\underset{\sim}{x}}(t)$ are specific to the particle beam problem. $\lambda_s(t,\bar{r},\bar{\underset{\sim}{x}}(t))$ need not be Gaussian, nor the shaping filter linear, in order to develop the MMAE for the general estimator. However, these particular choices do lend themselves well to estimators which have forms similar to those of Kalman filters. These filters were developed by Snyder and Fishman (Ref. 15) and will be discussed in the next section.

Similar to signal-induced events, noise-induced events appear as Poisson space-time point processes, but this time with rate parameter $\lambda_n(t,\bar{r})$. Generally, this parameter is assumed to depend on the random process $\underset{\sim}{\theta}$ that can depict characteristics of the noise. Therefore, $\lambda_n(t,\bar{r})$ itself can be taken as a random process dependent on $\underset{\sim}{\theta}$. However, in this work $\lambda_n(t,\bar{r})$ is deterministic, time invariant and constant-valued across the entire detector array (Ref. 9:21-22).

The signal and noise events are independent and additive. Therefore, the probability density function of the sum is equal to the convolution of the individual densities, or the characteristic function of the sum equals the product of the individual characteristic functions

(Ref. 18:12-13). As shown in Papoulis (Ref. 13:155), a characteristic function is simply the Fourier transform of a particular probability density function. Therefore, for the signal process, the equation for the conditional characteristic function for the signal process, conditioned on a given value of $\underset{\sim}{\bar{x}}(t)$, is

$$\phi_s(\omega,\bar{r},\underset{\sim}{\bar{x}}(t)) = \exp\{\lambda_s(t,\bar{r},\underset{\sim}{\bar{x}}(t))(e^{j\omega}-1)\} \tag{2-3}$$

and for the noise

$$\phi_n(\omega,\bar{r}) = \exp\{\lambda_n(t,\bar{r})(e^{j\omega}-1)\} \tag{2-4}$$

where $j=\sqrt{-1}$ and ω represents frequency in radians/sec. Multiplying these together yields a total rate parameter which looks like the sum of the signal and noise rate parameters.

$$\lambda(t,\bar{r},\underset{\sim}{\bar{x}}(t)) = \lambda_s(t,\bar{r},\underset{\sim}{\bar{x}}(t)) + \lambda_n(t,\bar{r}) \tag{2-5}$$

II.2 SNYDER-FISHMAN FILTER

Snyder and Fishman (Ref. 15) proposed an estimate of the beam's centroid position (assuming the absence of noise sources) as $\underline{H}(t)\hat{\underset{\sim}{\bar{x}}}(t)$, where $\hat{\underset{\sim}{\bar{x}}}(t)$ is the Bayesian estimate:

$$\hat{\underset{\sim}{\bar{x}}}(t) = E\{\hat{\underset{\sim}{\bar{x}}}(t)|Z^{N(t)}\} = \int_{R^m} \bar{\xi}(t) f(\bar{\xi}(t)|Z^{N(t)}) d\bar{\xi} \tag{2-6}$$

This equation is interpreted as the expected value of the beam state $\underset{\sim}{\bar{x}}(t)$, conditioned upon all measurements made through time t. The function $f(\bar{\xi}|Z^{N(t)})$ is the conditional probability density function of the state $\underset{\sim}{\bar{x}}(t)$ (letting $\bar{\xi}(t)$ be the dummy variable of integration associated with $\underset{\sim}{\bar{x}}(t)$), conditioned on $Z^{N(t)}$. $Z^{N(t)}$ represents the measurement

history shown on page 2-2. R^m indicates that the integration is being performed across the entire detector array. In the absence of noise events, the estimator can be described by the following differential equations:

$$d\hat{\underset{\sim}{x}}(t) = \underline{F}(t)\hat{\underset{\sim}{x}}(t)dt + \int_{R^m} \underline{K}(t)[\bar{r}-\underline{H}(t)\hat{\underset{\sim}{x}}(t)]N(dt \times d\bar{r}) \qquad (2\text{-}7)$$

$$d\underline{P}(t) = \underline{F}(t)\underline{P}(t)dt + \underline{P}(t)\underline{F}^T(t)dt + \underline{G}(t)\underline{G}^T(t)dt$$

$$- \int_{R^m} \underline{K}(t)\underline{H}(t)\underline{P}(t)N(dt \times d\bar{r}) \qquad (2\text{-}8)$$

$$\underline{K}(t) = \underline{P}(t)\underline{H}^T(t)[\underline{H}(t)\underline{P}(t)\underline{H}^T(t) + \underline{R}(t)]^{-1} \qquad (2\text{-}9)$$

starting from the initial conditions:

$$\hat{\underset{\sim}{x}}(t_o) = \bar{x}_o$$

$$\underline{P}(t_o) = \underline{P}_o \qquad (2\text{-}10)$$

In these equations, $\hat{\underset{\sim}{x}}(t)$ is the estimate of beam state, $\underline{H}(t)\hat{\underset{\sim}{x}}(t)$ is the estimate of the beam centroid location, $\underline{P}(t)$ is the filter-computed estimate error covariance (i.e., an indication of error), and $\underline{K}(t)$ is the filter gain as computed in Equation (2-9). $\int_{R^m} f(t,\bar{r})N(dt \times d\bar{r})$ is known as a counting integral and is defined below.

$$\int_{R^m} f(t,\bar{r})N(dt \times d\bar{r}) = \begin{cases} 0 & N_t=0 \\ \sum_{i=1}^{N(t)} f(t_i,\bar{r}_i)\delta_{t,t_i} & N_t \geq 1 \end{cases} \qquad (2\text{-}11)$$

where δ_{t,t_i} is the Kronecker delta. Stated simply, if no signal is detected by the array, the integral assumes a value of $\emptyset$. If an event is detected during the time interval between t and t + dt, the integral equals $f(t_i,\bar{r}_i)$ for that time t_i.

The estimator equations are still not in readily implementable form; however, Santiago and Maybeck (Refs. 7 and 14) showed that Equations (2-8) through (2-10) could be separated into propagation and update equations. The propagation equations have the form:

$$d\hat{\underset{\sim}{x}}(t) = \underline{F}(t)\hat{\underset{\sim}{x}}(t)dt \tag{2-12}$$

$$d\underline{P}(t) = \underline{F}(t)\underline{P}(t)dt + \underline{P}(t)\underline{F}^T(t)dt + \underline{G}(t)\underline{G}^T(t)dt \tag{2-13}$$

while the update equations are written as

$$\hat{\underset{\sim}{x}}(t_i^+) = \hat{\underset{\sim}{x}}(t_i^-) + \underline{K}(t_i)[\bar{r} - \underline{H}(t_i)\hat{\underset{\sim}{x}}(t_i^-)] \tag{2-14}$$

$$\underline{P}(t_i^+) = \underline{P}(t_i^-) - \underline{K}(t_i)\underline{H}(t_i)\underline{P}(t_i^-) \tag{2-15}$$

t_i^- represents time immediately prior to the observed event, and t_i^+ is the time immediately following the event. Note that these equations are very similar to standard Kalman filter equations except for one major difference. With the Snyder-Fishman filter, updates are performed whenever a photo-electron event happens to be detected, whereas Kalman filter updates are performed at predetermined intervals.

This section has completely described the estimation of $\bar{\underset{\sim}{x}}(t)$ using a Snyder-Fishman filter. In the absence of noise events, or with a complete knowledge of which events are noise and which are signal, these equations could be implemented with a great degree of confidence in the

output estimates. However, in actual practice, the detector cannot differentiate between the two sources, and, as Santiago points out (Ref. 14), accuracy is severely degraded. Therefore, some type of configuration is needed in order to "filter out" the noise in hope of properly estimating the beam's location. To accomplish this task, Captain Meer designed the MMAE configuration (Refs. 5 and 9) which is discussed in the next section.

II.3 MULTIPLE MODEL ADAPTIVE ESTIMATOR

This section gives an overall picture of the Meer filter. First, the general MMAE scheme is discussed in order to generate an understanding of its uses and the equations involved. Second, the specific application to the Meer filter is presented. This latter section includes such topics as the hypothesis sequence and its relation to the MMAE as well as the need to limit the size of the resulting hypothesis tree. References are cited in case a more thorough derivation is desired.

II.3.1 GENERAL MMAE. In order to develop the basic equations for an MMAE, we assume that minimal mean squared error is the optimality criterion to be used for the estimate of beam location (Ref. 4:232, Vol. I). This being the case, we have the same equation for an optimal $\hat{\underset{\sim}{x}}(t)$ as in the previous section:

$$\hat{\underset{\sim}{x}}(t) = E\{\underset{\sim}{x}(t)|Z^{N(t)}\} = \int_{R^m} \bar{\xi}(t) f(\bar{\xi}(t)|Z^{N(t)}) d\bar{\xi} \qquad (2\text{-}16)$$

$Z^{N(t)}$ is again the history of measurements through time t (i.e., $Z^{N(t)} = [\underset{\sim}{z}(t_1), \underset{\sim}{z}(t_2), \ldots, \underset{\sim}{z}(t_{N(t)})]$), and $\bar{\xi}$ is the dummy variable to represent the beam state within the Riemann integral. $f(\bar{\xi}|Z^{N(t)})$ is the

conditional probability density function of the state $\bar{\xi}$ based upon the history of measurements $Z^{N(t)}$. Remember that the $Z^{N(t)}$ represents Poisson space-time point process observations; however, this has no bearing on the general development. Also, $f(\bar{\xi}|Z^{N(t)})$ within the integral must exist in order for the estimate to be evaluated.

The next step is to assume that either the state process $\underset{\sim}{\bar{x}}(t)$ or the measurement process $\underset{\sim}{\bar{z}}(t)$ is not adequately described by a single model, making $f(\bar{\xi}|Z^{N(t)})$ difficult to evaluate. However, $\underset{\sim}{\bar{x}}(t)$ or $\underset{\sim}{\bar{z}}(t)$ can be described by a finite number of models chosen from an appropriately defined set. Assume that there are J+1 models which can be used to describe $\underset{\sim}{\bar{x}}(t)$ or $\underset{\sim}{\bar{z}}(t)$. Each model is represented by h_i where $i=0,1,2,\ldots,J$. We can depict h_i as

$$h_i \in H \tag{2-17}$$

where H is the set of possible models. Note that the collection of models need not be finite for the general MMAE development, yet we will continue this assumption due to the nature of the particle beam estimator.

Turning back to the equation for the optimal estimate, we find that the Riemann integral can be evaluated as:

$$\underset{\sim}{\hat{\bar{x}}}(t) = \int_{R^m} \bar{\xi}(t) \int_H f(\bar{\xi}(t),h|Z^{N(t)})dh d\bar{\xi} \tag{2-18}$$

The joint density function is needed because the marginal function is not adequately modelled by one h. By using Bayes' rule (Ref. 13:161), Equation (2-18) can be expressed as

$$\hat{\underset{\sim}{x}}(t) = \int_{R^m} \bar{\xi}(t) \int_H f(\bar{\xi}|h,Z^{N(t)}) f(h|Z^{N(t)}) dh d\bar{\xi} \tag{2-19}$$

The number of models has been limited so that

$$f(h|Z^{N(t)}) = \sum_{j=0}^{J} Pr(h_j \text{ correct}|Z^{N(t)}) \delta(h-h_j) \tag{2-20}$$

where $Pr(\cdot)$ is probability and $\delta(\cdot)$ is the Dirac Delta function. Substituting Equation (2-20) into (2-19) and applying the sifting property (Ref. 1:185, Vol. II) of $\delta(\cdot)$, we find:

$$\hat{\underset{\sim}{x}}(t) = \int_{R^m} \bar{\xi}(t) \sum_{j=0}^{J} f(\bar{\xi}(t)|h_j,Z^{N(t)}) Pr(h_j|Z^{N(t)}) d\bar{\xi} \tag{2-21}$$

By shifting the order of the summation and integral signs we see:

$$\hat{\underset{\sim}{x}}(t) = \sum_{j=0}^{J} Pr(h_j|Z^{N(t)}) \int_{R^m} \bar{\xi}(t) f(\bar{\xi}(t)|h_j,Z^{N(t)}) d\bar{\xi} \tag{2-22}$$

$$= \sum_{j=0}^{J} Pr(h_j|Z^{N(t)}) E\{\underset{\sim}{x}(t)|h_j,Z^{N(t)}\} \tag{2-23}$$

$$= \sum_{j=0}^{J} Pr(h_j|Z^{N(t)}) \hat{\underset{\sim}{x}}_j(t) \tag{2-24}$$

In Equation (2-24), $\hat{\underset{\sim}{x}}_j(t)$ represents the estimate of $\underset{\sim}{x}(t)$ assuming model h_j is correct, and $Pr(h_j|Z^{N(t)})$ is the probability of model h_j being correct given that the measurement sequence $Z^{N(t)}$ was observed. Equation (2-24) can be represented by the block diagram in Figure 2-1 and is the general equation for an MMAE state estimate.

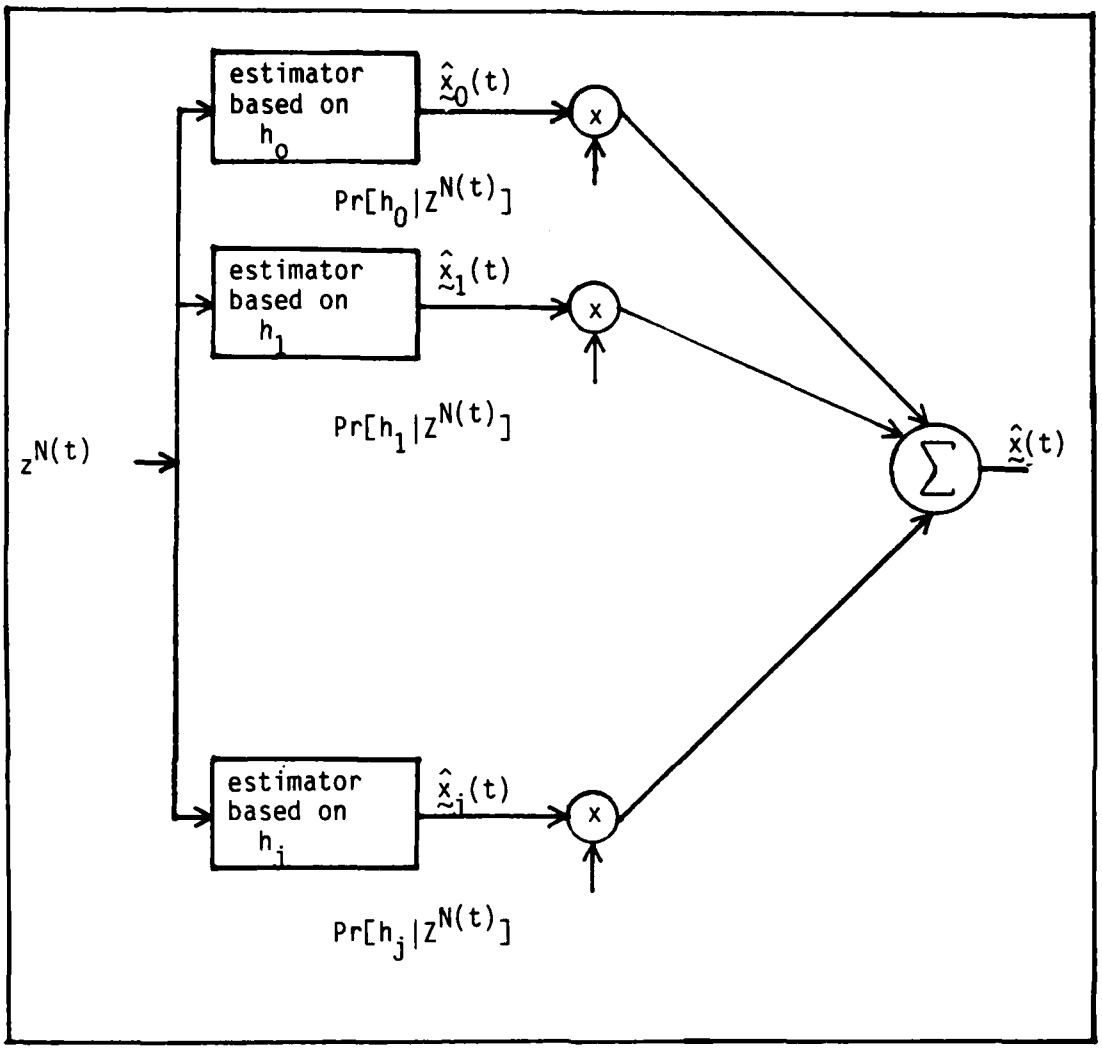


Figure 2-1. Multiple Model Adaptive Estimator

Another variable in which we might be interested is the state error covariance of the estimator. This quantity is not essential for obtaining the estimate $\hat{\bar{\underset{\sim}{x}}}(t)$; however, it is useful for performance analysis. The covariance of the full scale estimator can be written as

$$\underline{P}(t) = E\{[\underset{\sim}{\bar{x}}(t) - \hat{\bar{\underset{\sim}{x}}}(t)][\underset{\sim}{\bar{x}}(t) - \hat{\bar{\underset{\sim}{x}}}(t)]^T | Z^{N(t)}\} \tag{2-25}$$

while that of a particular estimator appears as

$$\underline{P}_j(t) = E\{[\underset{\sim}{\bar{x}} - \hat{\bar{\underset{\sim}{x}}}_j(t)][\underset{\sim}{\bar{x}}(t) - \hat{\bar{\underset{\sim}{x}}}_j(t)]^T | h_j, Z^{N(t)}\} \tag{2-26}$$

From these, the overall covariance can be written as (Ref. 9):

$$\underline{P}(t_i^+) = \sum_{j=0}^{J} Pr(h_j | Z^{N(t)}) \; \{\hat{\underline{P}}_j(t_i^+) + [\hat{\bar{\underset{\sim}{x}}}_j(t_i^+) - \hat{\bar{\underset{\sim}{x}}}(t_i^+)][\hat{\bar{\underset{\sim}{x}}}_j(t_i^+) - \hat{\bar{\underset{\sim}{x}}}(t_i^+)]^T\} \tag{2-27}$$

using the model-specific estimates $\hat{\bar{\underset{\sim}{x}}}_j(t)$, covariances $\underline{P}_j(t_i^+)$, and the overall estimate $\hat{\bar{\underset{\sim}{x}}}(t)$ from Equation (2-24). Note that since this equation depends on $Pr(h_j | Z^{N(t)})$, $\hat{\bar{\underset{\sim}{x}}}_j(t_i^+)$, and $\hat{\bar{\underset{\sim}{x}}}(t_i^+)$, it is not precomputable (Ref. 4:131, Vol. II).

II.3.2 <u>PARTICLE BEAM MMAE</u>. If we are to estimate the particle beam state $\underset{\sim}{\bar{x}}(t)$ effectively, we need two basic equations for our filter model. These are the system dynamics and measurement model equations. As shown in Equation (2-2) earlier in this chapter, the state process $\underset{\sim}{\bar{x}}(t)$ can be modelled as the Gaussian output of a linear stochastic differential equation. This dynamics model led to the Snyder-Fishman filter equations shown in (2-12) through (2-15). Note that these equations assumed that any measurements were purely signal-induced. However, as noted

previously, a photo-electron event can be either signal- or noise-induced. Therefore, we do not yet have an adequate measurement model for the particle beam estimator.

It is known that any observed event on the detector array must be either signal- or noise-induced. Knowing that these are the only possibilities which exist, we can easily construct all the possible sequences which could have yielded the observed sequence. This type of logic leads to a "hypothesis tree" for the estimator, and the discussion which follows closely parallels the thoughts and notation of both Captain Meer and Lieutenant Zicker (Refs. 5, 8, 9 and 18).

The hypothesis tree begins at the initial time t_0 when no events have happened. At some $t_1 > t_0$ an event is observed on the photodetector array at location $\bar{r}_1$. As stated previously, this event must be either signal- or noise-induced which leads to the two hypotheses shown in Figure 2-2. Throughout this discussion, certain conventions are employed to draw the hypothesis trees. First of all, assumed signal-induced events are drawn upward from a previous node, while noise-induced events are displayed downward. The hypothesis sequence is represented by $h_j^{N(t)}$ where $j=\emptyset,1,2,\ldots,2^{N(t)}-1$, and N(t) is the total number of observed events. An entire sequence could be written as

$$h_j^{N(t)} = \{h_j^{N(t)}(1),\ h_j^{N(t)}(2),\ \ldots,\ h_j^{N(t)}(N(t))\} \qquad (2\text{-}28)$$

All signal-induced events are represented by a 1 while noise-induced events are marked by a $\emptyset$. Therefore, for our first measurement in Figure 2-2, $h_0^1(1)$ is defined as $\emptyset$ while $h_1^1(1)$ is defined as 1.

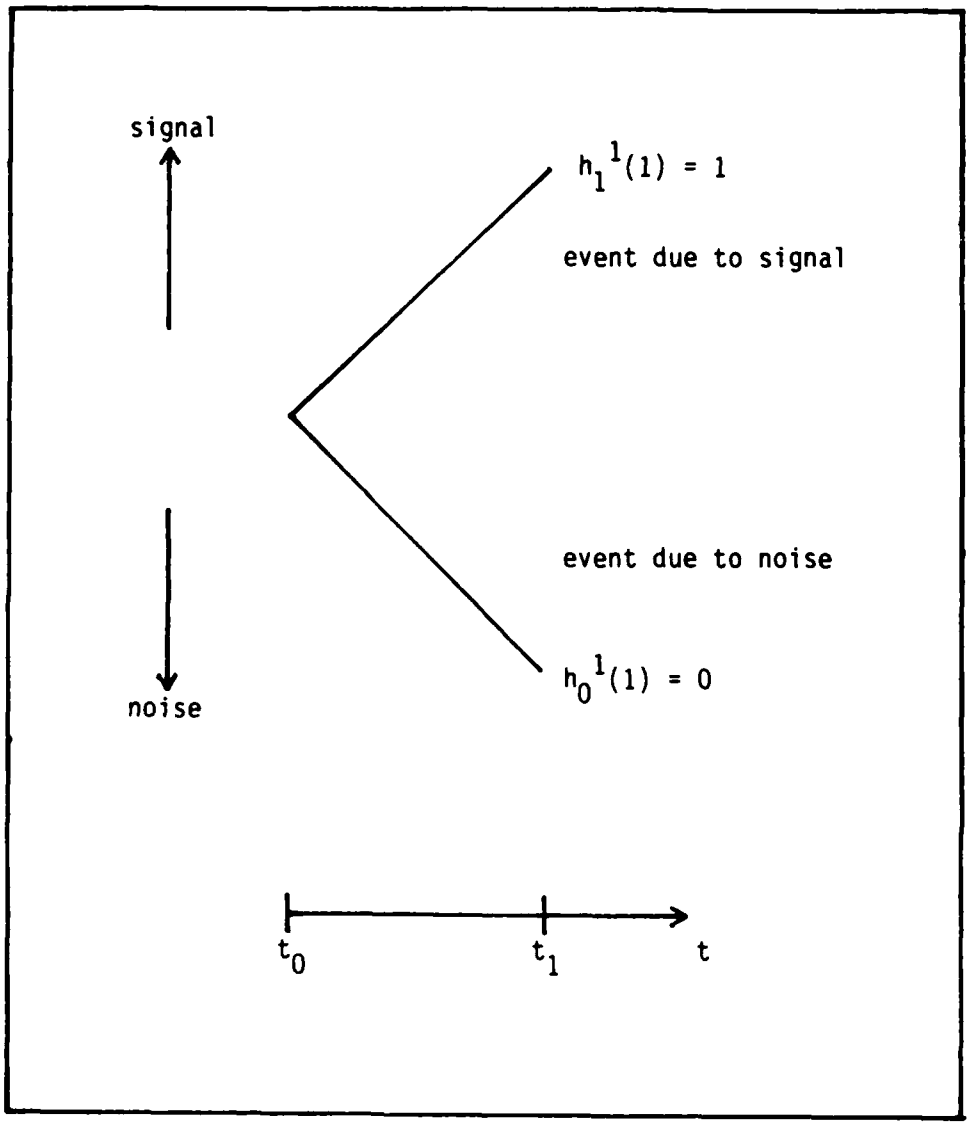


Figure 2-2. Hypotheses After First Event

The benefits of this notation are clearly seen as we progress to more measurements. At time t_2 we detect an event at $\bar{r}_2$. The hypothesis tree expands as shown in Figure 2-3. Using the standard notation we see that the hypothesis sequences which result are as follows:

$$h_0^2 = \{h_0^2(1),\ h_0^2(2)\} = \{0,0\}$$

$$h_1^2 = \{h_1^2(1),\ h_1^2(2)\} = \{0,1\}$$

$$h_2^2 = \{h_2^2(1),\ h_2^2(2)\} = \{1,0\}$$

$$h_3^2 = \{h_3^2(1),\ h_3^2(2)\} = \{1,1\} \qquad (2\text{-}29)$$

As can be seen by looking at the structure of the hypothesis tree, the number of hypotheses doubles with each measurement, yielding $2^{N(t)}$ hypotheses at time $t_{N(t)}$. We therefore expect memory and computational loading problems if this growth is not restricted.

We now have a method of estimating the particle beam state $\bar{\underset{\sim}{x}}(t)$ which incorporates the possibility of noise-induced measurements. Any particular hypothesis corresponds to a specific sequence of signal- and noise-induced events. Knowing the sequence, we can easily employ the Snyder-Fishman equations, updating only where a signal-induced event is hypothesized. If we do this for each possible hypothesis sequence $h_j^{N(t)}$, we have $2^{N(t)}$ estimates of the same state variable $\bar{\underset{\sim}{x}}(t)$. As seen in the general MMAE discussion, we can derive an overall estimate of $\bar{\underset{\sim}{x}}(t)$ by probabilistically weighting the individual estimates. This configuration could easily be represented by the equation:

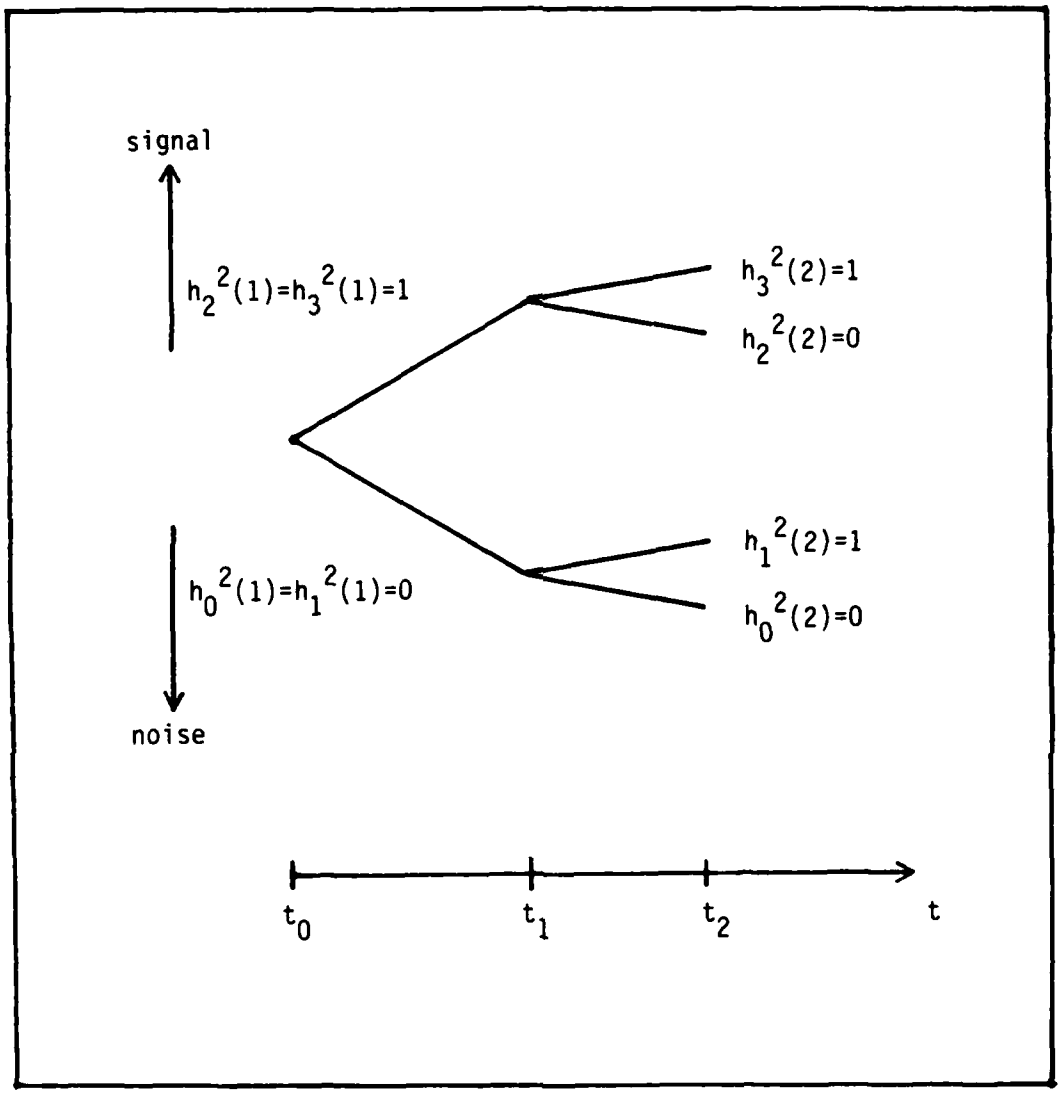


Figure 2-3. Hypotheses After Second Event

$$\hat{\underset{\sim}{x}}(t) = \sum_{j=0}^{2^{N(t)}-1} Pr[h_j|Z^{N(t)}]\hat{\underset{\sim}{x}}_j(t) \tag{2-30}$$

$Pr(h_j|Z^{N(t)})$ is the probability that hypothesis h_j is correct given that the sequence of measurements $Z^{N(t)}$ has been observed. $\hat{\underset{\sim}{x}}_j(t)$ is the estimate of $\bar{\underset{\sim}{x}}(t)$ assuming that the sequence $h_j^{N(t)}$ was true. We could write a similar equation for the covariance of the beam state:

$$\underline{P}(t) = \sum_{j=0}^{2^{N(t)}-1} Pr[h_j|Z^{N(t)}] \{\underline{P}_j(t)$$

$$+ [\hat{\underset{\sim}{x}}_j(t) - \hat{\underset{\sim}{x}}(t)][\hat{\underset{\sim}{x}}_j(t) - \hat{\underset{\sim}{x}}(t)]^T\} \tag{2-31}$$

where $\underline{P}_j(t)$ is the covariance calculated when hypothesis $h_j^{N(t)}(t)$ is assumed correct.

The particle beam MMAE has now been shown in its most general form. There should be no question as to why this configuration was chosen; however, there are two major areas which must be wrung out before our discussion of the Meer filter is complete. First, equations for probabilistic weighting factors $Pr(h_j|Z^{N(t)})$ must be developed if the MMAE is to have any meaning at all. Second, some method of limiting the filter sized must be created in order to implement the filter practically. The next two sections are devoted to these topics.

II.4 WEIGHTING FACTORS

In order to complete the set of equations for the particle beam MMAE, the weighting factor equations must be developed. These equations, in conjunction with the Snyder-Fishman equations of Section II.2 and the

MMAE equations of Section II.3, will allow us to specify the Meer filter completely. The weighting factor equations which were developed by Meer in his dissertation appear as:

$$\Pr[h_j^{N(t)}|Z^{N(t)}] = \frac{\hat{\lambda}_s \text{ or } \hat{\lambda}_n}{\hat{\lambda}} \Pr[h_j^{N(t)-1}|Z^{N(t)-1}] \qquad (2\text{-}32)$$

$\hat{\lambda}_s$, $\hat{\lambda}_n$, and $\hat{\lambda}$ are estimates of the signal, noise, and overall rate parameters respectively (where $\hat{\lambda}=\hat{\lambda}_s + \hat{\lambda}_n$), $\Pr(h_j^{N(t)-1}|Z^{N(t)-1})$ is the probability of $h_j^{N(t)-1}$ before the most recent event was detected. $\hat{\lambda}_s$ appears in the numerator if $h_j^{N(t)}(N(t))=1$ and $\hat{\lambda}_n$ appears if $h_j^{N(t)}(N(t))=\emptyset$. Because of the recursive nature of this equation, an entire history of measurements $Z^{N(t)}$ need not be kept.

Zicker was able to represent Equation (2-32) in a slightly different way (Ref. 18). We can trace Equation (2-32) back to the probability at t_0:

$$\Pr[h_j^0|Z^0] = 1 \qquad (2\text{-}33)$$

The probability is 1 because there are no hypotheses at $t=t_0$. Realizing this initial condition, we find that Equation (2-32) is transformed into:

$$\Pr[h_j^{N(t)}|Z^{N(t)}] = \frac{\prod_{S_j} \hat{\lambda}_s(t_k,\bar{r}_k;\omega) \prod_{N_j} \hat{\lambda}_n(t_k,\bar{r}_k;\omega)}{\prod_{i=1}^{N(t)} \hat{\lambda}} \qquad (2\text{-}34)$$

where S_j represents the set of event times when we hypothesize a signal being detected under hypothesis $h_j^{N(t)}$:

$$S_j = \{k:h_j^{N(t)}(k) = 1\} = \{k_1, k_2, \ldots, k_q\} \tag{2-35}$$

and N_j represents the set of event times when we hypothesize noise being detected under hypothesis $h_j^{N(t)}$:

$$N_j = \{\ell:h_j^{N(t)}(\ell) = \emptyset\} = \{\ell_1, \ell_2, \ldots, \ell_p\} \tag{2-36}$$

Note that $p+q=N(t)$.

The only values remaining to be specified in these weighting equations are the estimated λ_n and λ_s. To accomplish this, Meer postulates an unobservable probability space Ω_s associated with the system state, which maps into an observable space Ω. Ω is subdivided into three subspaces:

$$\Omega = \Omega_1 \times \Omega_2 \times \Omega_3 \tag{2-37}$$

Ω_1, Ω_2, and Ω_3 correspond to the randomness of the signal process, noise process, and the hypothesis sequences respectively. Therefore, $\hat{\lambda}_s$ and $\hat{\lambda}_n$ are generated by calculating the expected value over the appropriate subspace:

$$\hat{\lambda}_s(t,\bar{r};\omega) = E_1\{\lambda_s(t,\bar{r};\omega;\omega_1)|Z^{N(t)}\} \tag{2-38}$$

$$\hat{\lambda}_n(t,\bar{r};\omega) = E_2\{\lambda_n(t,\bar{r};\omega;\omega_2)|Z^{N(t)}\} \tag{2-39}$$

where E_1 represents expectation over $\omega_1 \varepsilon \Omega_1$ and E_2 represents expectation over $\omega_2 \varepsilon \Omega_2$. By assuming Λ, $\underline{R}$, and $\underline{H}$ constant and deterministic, Meer was able to represent the expected value as:

$$\hat{\lambda}_s = \Lambda \exp\{-[\bar{r}-\underline{H}\hat{\underline{x}}_j(t)]^T \underline{R}^{-1} [\bar{r}-\underline{H}\hat{\underline{x}}(t)]/2\} \tag{2-40}$$

Note that $\hat{\underset{\sim}{x}}_j(t)$ is the estimate of the beam state derived from the specific Snyder-Fishman estimate which is about to be weighted. Since the noise rate was assumed deterministic and constant over the entire detector, the noise rate estimate $\hat{\lambda}_n$ is equal to this same constant.

II.5 LIMITING THE HYPOTHESIS TREE

At this point we have all the equations necessary in order to construct a full-scale particle beam estimator. The Snyder-Fishman equations provide the propagation and update cycles for any particular hypothesis sequence. Each estimate $\hat{\underset{\sim}{x}}_j(t)$ based on $h_j^{N(t)}$ is then weighted and summed to yield the particle beam state estimate $\hat{\underset{\sim}{x}}(t)$. Similar comments could be made concerning the beam state error covariance $\underline{P}(t)$. However, it is readily seen that at each new measurement, the number of hypotheses doubles, yielding $2^{N(t)}$ possible sequences at $t_{N(t)}$. It is therefore imperative that we limit the size of the hypothesis tree to a particular depth D. When D=d, there is an upper limit of 2^d possible hypotheses at any time t_j. As noted in Zicker's thesis (Ref. 18), there are two methods for limiting the tree depth to D: the Best Half method and the Merge method. Both techniques will be discussed in the sections which follow.

II.5.1 THE BEST HALF METHOD. The Best Half method is the technique used by Meer in his dissertation (Ref. 9). With this method we let the hypothesis tree grow in the usual fashion until the number of observed events equals D. At this point, turn to the weighting factors to provide insight into the problem. We begin by dividing the hypothesis tree in half. The upper half corresponds to the sequences of branches which originate from a signal-induced event, while the lower half

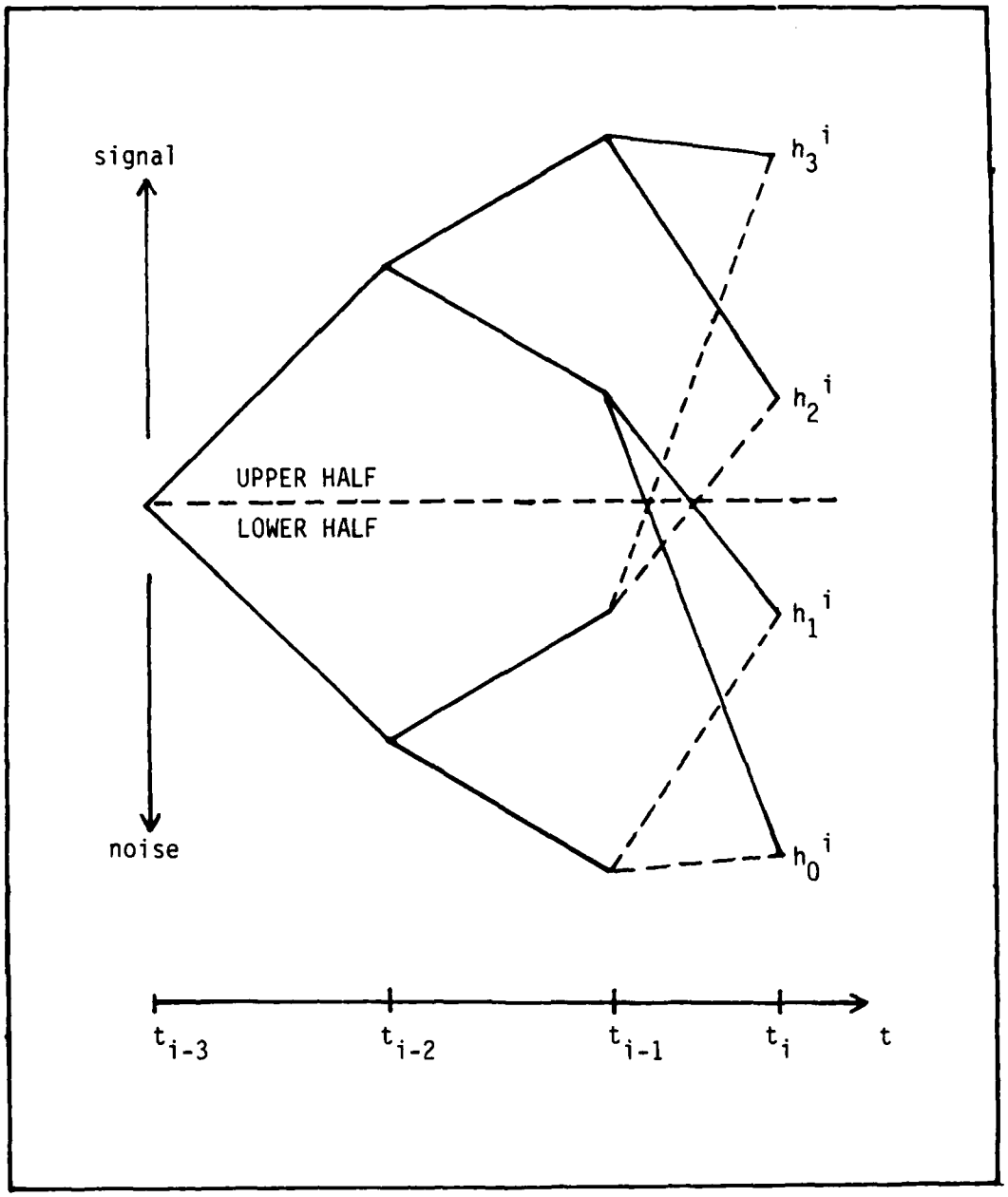


Figure 2-4. Best Half Method of Hypothesis Limiting

corresponds to the sequences of branches which originate from a noise-induced event. The two branches differ only in the hypothesis at the first event time. We then look at the probability that the upper half is correct by summing the weighting factors at time t_D associated with the upper half. Also, the probability of the lower half being correct is calculated by summing the weighting factors at t_D associated with the lower half.

$$\Pr[\text{UPPER HALF}] = \sum_{j=\frac{2^D}{2}}^{2^D-1} \Pr[h_j^{N(t)} | Z^{N(t)}]$$

$$\Pr[\text{LOWER HALF}] = \sum_{j=0}^{\frac{2^D}{2}-1} \Pr[h_j^{N(t)} | Z^{N(t)}] \tag{2-41}$$

Since these two equations must sum to 1, only one summation needs to be computed. If Pr(UPPER HALF) is greater than Pr(LOWER HALF) we expand the upper half as usual while terminating the lower half. If Pr(LOWER HALF) is greater, then the reverse is true. In either case the remaining weighting factors must be renormalized so that their sum is again equal to 1.

An example of the Best Half Method is shown in Figure 2-4 for D=2. The upper and lower halves are divided accordingly depending upon whether the initial event was signal- or noise-induced. When the number of hypotheses equals 4, we must make a decision. The solid lines from t_{i-1} to t_i represent the upper half being more correct while the dashed lines mean that the lower half is preferred. In either case, note that

at t_i we have still limited the total number of hypotheses to 2^D by discarding or "pruning" either the solid or dashed branches of the decision tree. Unfortunately, in order to limit the size of the filter with the Best Half method, we have literally thrown away half the information. Some means must be available for limiting the filter while not discarding half the hypothesis tree. Zicker incorporated the laternative Merge method.

II.5.2 MERGE METHOD. The Merge method was initially proposed by Weiss and co-workers (Ref. 17) so that the information in the entire time history of measurements could be somewhat preserved. With this method we look at pairs of sequences to limit the size of the expanding filter. As before, the tree grows as usual until t_D. At this point we pair all the hypotheses which are equivalent except for the earliest retained event. The hypothesis pairs are represented as h_j^D and h_k^D, where $j=\emptyset,1,\ldots,(2^D/2)-1$ and $k=j+2^D/2$. There are $(2^D/2)$ such pairs. Since they differ in the earliest time event we see that

$$\begin{aligned} h_j^{N(t)}(N(t) - D + 1) &= \{0\} \\ h_k^{N(t)}(N(t) - D + 1) &= \{1\} \end{aligned} \tag{2-42}$$

We then take the weighted sum of the estimates associated with each pair of hypotheses:

$$\hat{x}_j(t) = \{Pr[h_j|Z^{n(t)}]\hat{x}_j(t) + Pr[h_k|Z^{N(t)}]\hat{x}_k(t)\}/a \tag{2-43}$$

$$\begin{aligned} \underline{P}_j(t) = \{&Pr[h_j|Z^{N(t)}](\underline{P}_j(t) + [\hat{\underset{\sim}{x}}_j(t) - \hat{\underset{\sim}{x}}_j(t)][\hat{\underset{\sim}{x}}_j(t) - \hat{\underset{\sim}{x}}_j(t)]^T) \\ &+ Pr[h_k|Z^{N(t)}](\underline{P}_k(t) + [\hat{\underset{\sim}{x}}_k(t) - \hat{\underset{\sim}{x}}_j(t)][\hat{\underset{\sim}{x}}_k(t) - \hat{\underset{\sim}{x}}_j(t)]^T)\}/a \end{aligned} \tag{2-44}$$

$$a = Pr[h_j'|Z^{N(t)}] = Pr[h_j|Z^{N(t)}] + Pr[h_k|Z^{N(t)}] \qquad (2\text{-}45)$$

In Equation (2-45), a is a normalizing factor so that the sum of the probabilities of each new estimate will equal 1. An example of the Merge method is shown in Figure 2-5 for D=2. Note that t_{i-1}' represents an intermediate step where the dashed lines merge. From t_{i-1}' the nodes split as usual and at t_i the total number of hypotheses is still 2^D.

II.6 SUMMARY

This chapter has been a complete discussion of the Meer filter (Refs. 5 and 9) used to estimate the particle beam's centroid. We began by discussing the measurement scheme and assumed state dynamics. These dynamics were used in conjunction with the assumption of signal-induced measurements to form the Snyder-Fishman equations. However, this assumption was unrealistic for actual implementation, forcing Meer to develop an MMAE scheme based on an uncertain measurement model. However, the number of models could easily grow without bound if we did not limit the size with either the Best Half or Merge method, so this is incorporated for a practical filter.

With this state estimator in hand, we now set about designing a PI controller to regulate the beam and to track intended targets. The following chapter reveals how this regulator and tracker will be derived.

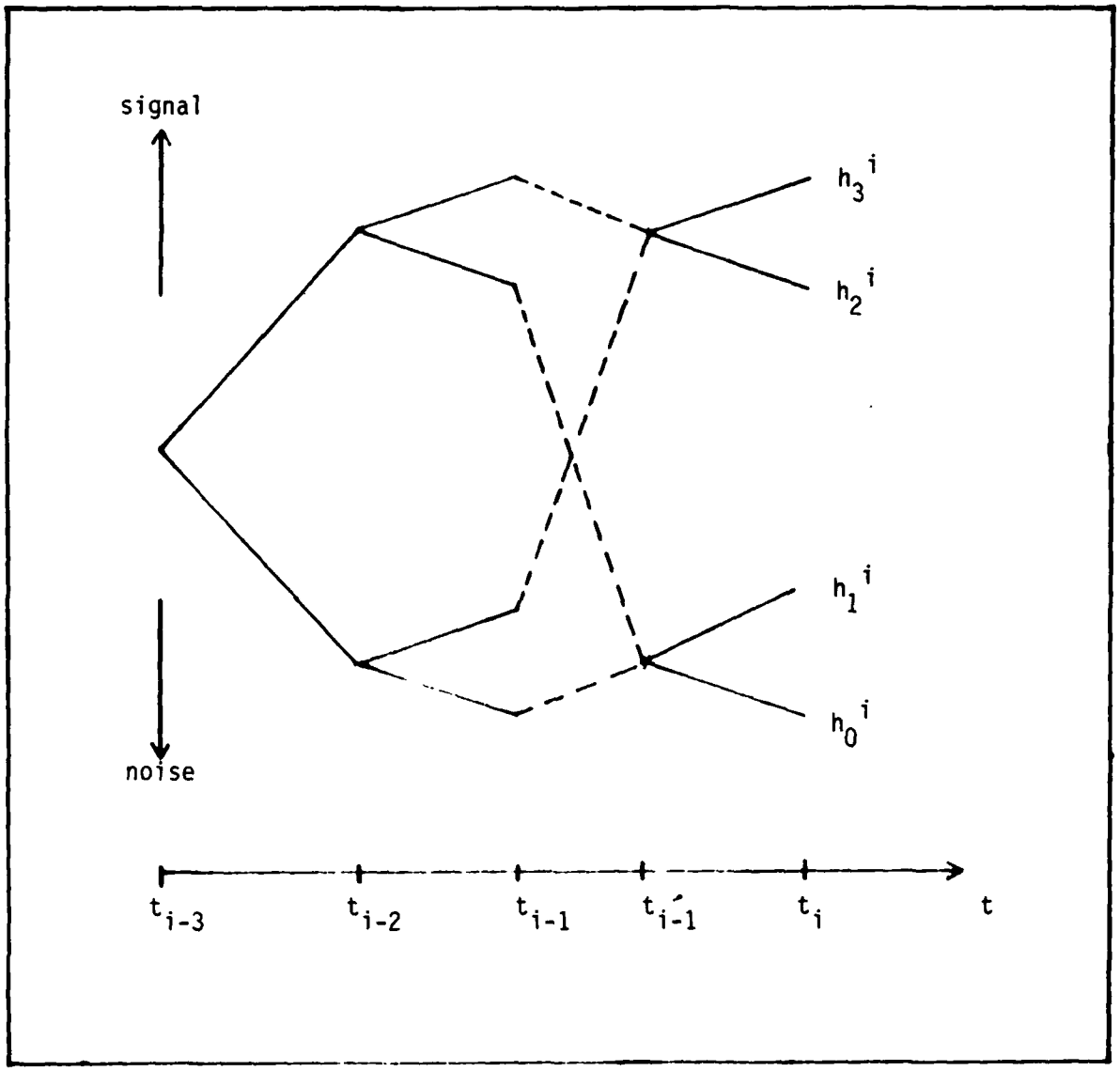


Figure 2-5. Merge Method of Hypothesis Limiting

III. PI CONTROLLER

The purpose of this chapter is to discuss the Proportional plus Integral (PI) controller which will be used to direct the particle beam weapon. In the first section we will look at regulation of the beam itself. Here the controller is trying to maintain the beam at a particular setpoint (usually zero). After looking at the regulator problem, we turn our attention toward the tracking of a particular target. To this end, the second section of this chapter is devoted to developing a three-state acceleration model for the target as well as a Kalman filter which is used to estimate the position of the intended target. With models for both the beam and the target, we turn to Section 3 for a development of the PI tracker.

Both the regulator and the tracker are designed assuming linear models with quadratic cost functions. Using LQ techniques we can derive steady state (constant) controller gains through the use of backward Riccati difference equations. To this point we assume that the controller has perfect access to the true states of the beam and target. We can replace these true states with their best estimates using the principle of assumed certainty equivalence (Ref. 4:Vol. 3). The particle beam state estimate is provided by the Meer filter while the target state comes from the Kalman filter. This step is necessary because it is unrealistic to assume complete access to the true states of the system.

Throughout this chapter the reader is assumed to understand the purpose of the cost function (performance index) as it relates to the field of optimal control, in addition to being familiar with the

principles of Kalman filtering. An in-depth discussion of the quadratic cost function (its meaning and uses) is provided by Reference 4 (Vol. 3). For a rigorous development of performance indices in the field of optimal control, see Kirk's text (Ref. 2). Here the reader will find how variational calculus techniques are used to obtain the optimal state trajectory to satisfy the desired performance index. Kwakernaak and Sivan (Ref. 2) also provide useful reference for this topic. For the theory of Kalman filtering as well as examples of how these estimators are used in the field of guidance and control, Maybeck (Ref. 4:Vol. 1) provides a thorough development.

III.1 PI REGULATOR

The purpose of the PI regulator is to guide the beam state to a particular setpoint y_d. When y_d equals Ø, we are trying to keep the backscatter indication of the beam centroid on the center of the detector array. As discussed in Chapter 2, we are using a single position state model for the beam dynamics:

$$\dot{\underset{\sim}{x}}(t) = -\frac{1}{T}\underset{\sim}{x}(t) + Bu(t) + G\underset{\sim}{w}(t) \tag{3-1}$$

where T is the time constant for the beam state dynamics, $\underset{\sim}{w}(t)$ is zero-mean white Gaussian noise of strength Q, and u(t) is the continuous time control input which will be held constant over each control sample period ($\Delta t = t_{i+1} - t_i$) once the system is discretized. The first order model was chosen because this is a feasibility study similar to Zicker's and Moose's work, and higher order models are not required at this level. Since this is a regulator problem, there is no need for a target model. Therefore, this topic need not be explored until the next section.

Because we will want to apply sampled data control inputs at discrete time intervals, the beam model should be discretized. The new beam model looks like the following:

$$\underset{\sim}{\bar{x}}(t_{i+1}) = \Phi_B(t_{i+1},t_i)\underset{\sim}{\bar{x}}(t_i) + B_d u(t_i) + G_d \underset{\sim}{w}_d(t_i) \tag{3-2}$$

where

$$\Phi_B(t_{i+1},t_i) = e^{-\Delta t/T} \tag{3-3}$$

$$G_d = I \tag{3-4}$$

$$B_d = \int_{t_i}^{t_{i+1}} \Phi_B(t_{i+1},\tau)B d\tau \tag{3-5}$$

and $\underset{\sim}{w}_d(t_i)$ is zero-mean discrete-time white Gaussian noise of strength

$$Q_d = \int_{t_i}^{t_{i+1}} \Phi_B(t_{i+1},\tau)GQG^T\Phi^T(t_{i+1},\tau)d\tau \tag{3-6}$$

For our purposes we can let B=1 in Equation (3-1). The value we are attempting to control is

$$\underset{\sim}{y}_c(t_i) = \underline{C}\ \underset{\sim}{\bar{x}}\ (t_i) \tag{3-7}$$

where C=1 and $\underset{\sim}{\bar{x}}(t_i)$ is the beam position state.

There is one other idea which must be developed before we can begin to derive the PI regulator--integral action. This particular design was chosen because of its ability to maintain a constant setpoint ($\underset{\sim}{\bar{x}}(t)$=0 in our case) with zero steady-state error even in the face of constant

unmodelled disturbance inputs. This ability is known as "type 1" control. The portion of the PI controller which gives it these qualities is the integrator. The integral term provides an integrated history of tracking errors beginning at the initial time, thereby yielding a control input even when there is no tracking error at the present time. Since we will be dealing with a digital controller, true integration is not possible and will be replaced by the pseudointegral or summation process:

$$q(t_{i+1}) = q(t_i) + [y_c(t_i) - y_d] \tag{3-8}$$

The general structure of a PI controller is shown in Figure 3-1. The most important feature of this pseudointegral term can be seen in this diagram; when y_d is not equal to $\emptyset$, even when $y_c = y_d$ so that the input to the pseudointegration is zero, nonzero inputs are still applied to the system by this channel so that in equilibrium $y_c = y_d$. This is the essence of a "type 1" system.

For the sake of PI controller derivation via LQ synthesis, we can now combine Equations (3-2) and (3-8) into the augmented state space system:

$$\begin{bmatrix} x(t_{i+1}) \\ q(t_{i+1}) \end{bmatrix} = \begin{bmatrix} \Phi_B(t_{i+1},t_i) & \emptyset \\ C & I \end{bmatrix} \begin{bmatrix} x(t_i) \\ q(t_i) \end{bmatrix} + \begin{bmatrix} B_d \\ \emptyset \end{bmatrix} u(t_i) + \begin{bmatrix} G_d \\ \emptyset \end{bmatrix} w_d(t_i) - \begin{bmatrix} \emptyset \\ 1 \end{bmatrix} y_d \tag{3-9}$$

Written another way, this equation gives

$$\bar{x}_a(t_{i+1}) + \underline{\Phi}\bar{x}_a(t_i) + \underline{B}_{da}u(t_i) + \underline{G}_{da}w_d(t_i) - \underline{D}_a y_d \tag{3-10}$$

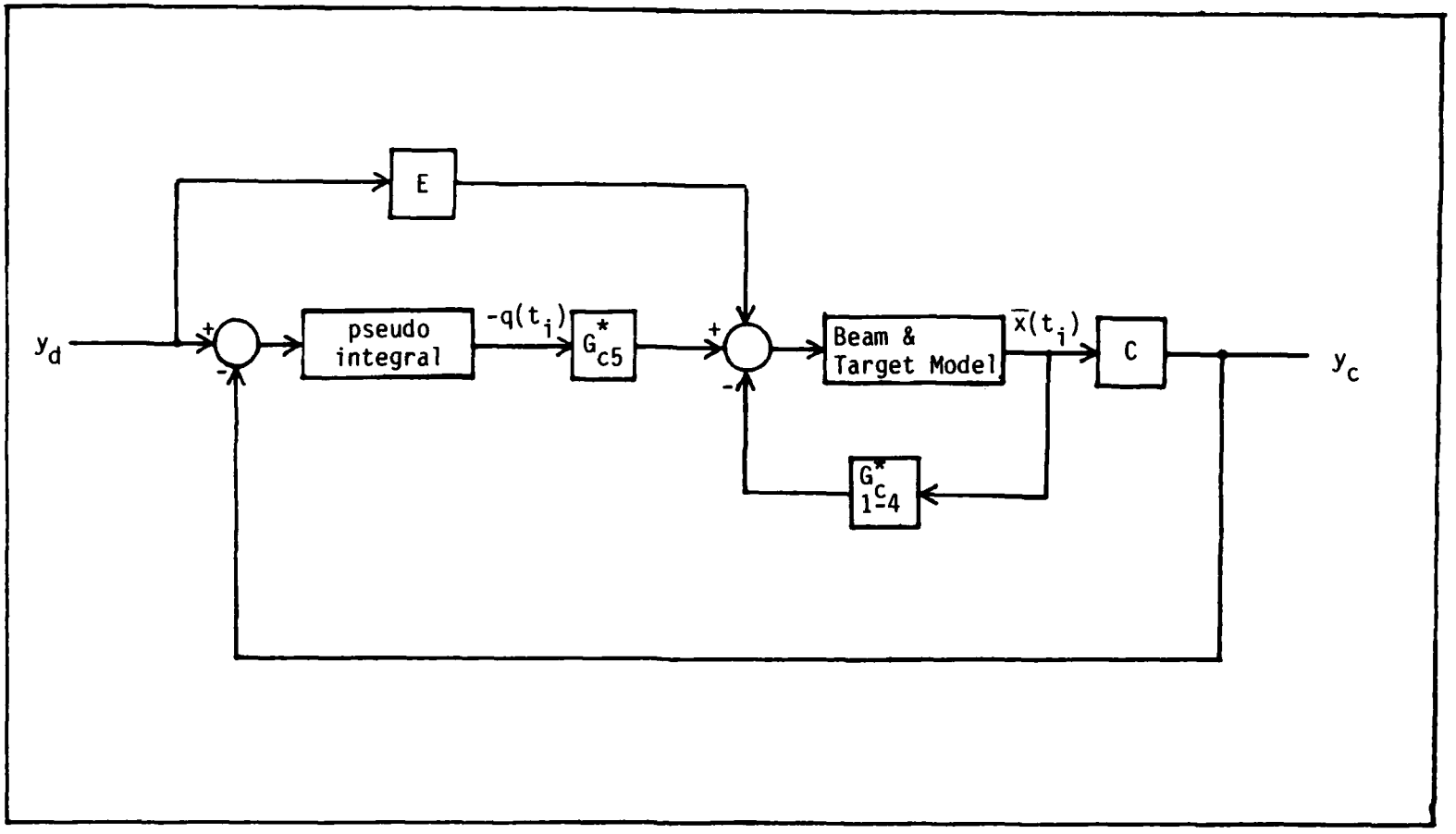


Figure 3-1. General Structure of PI Controller

With Equation (3-10) in hand, we now set about deriving the control $u(t_i)$. To accomplish this, some decision as to the optimal state trajectory must be made.

The purpose of optimal control theory is to derive an optimal (feedback) control law $u(t_i)$ which will minimize a specific performance index J. Once applied, this optimal control law produces the optimal state trajectory $x^*(t_i)$. The performance index we will employ is the quadratic cost function:

$$J = \frac{1}{2}\sum_{i=0}^{N}\left\{\begin{bmatrix} x(t_i) \\ q(t_i) \end{bmatrix}^T \begin{bmatrix} X_{11} & \emptyset \\ \emptyset & X_{22} \end{bmatrix} \begin{bmatrix} x(t_i) \\ q(t_i) \end{bmatrix} + Uu^2(t_i)\right\} + \frac{1}{2}\begin{bmatrix} x(t_{N+1}) \\ q(t_{N+1}) \end{bmatrix}^T \begin{bmatrix} X_{11f} & \emptyset \\ \emptyset & X_{22f} \end{bmatrix} \begin{bmatrix} x(t_{N+1}) \\ q(t_{N+1}) \end{bmatrix} \tag{3-11}$$

This can be written another way as

$$J = \frac{1}{2}\sum_{i=0}^{N}\{\bar{x}_a^T(t_I)\underline{X}\bar{x}_a(t_i) + Uu^2(t_i)\} + \bar{x}_a^T(t_{N+1})\underline{X}_f\bar{x}_a(t_{N+1}) \tag{3-12}$$

By minimizing J we are choosing the $u(t_i)$ which will drive $\bar{x}_a(t_i)$ toward a vector of zeros: note that any deviation of the vector $\bar{x}_a(t_i)$ from $\bar{\emptyset}$ tends to increase the cost J. However, there is a tradeoff in that there is a penalty if the control inputs $u(t_i)$ are nonzero. X_{11} and X_{22} are weighting factors on the beam state and tracking error pseudointegral respectively. U weights the amount of control energy expended, and the function outside the summation is known as the terminal cost. If U were set to $\emptyset$, the controller would try to drive the errors

to $\emptyset$ in the first sample period, thereby applying exorbitant energy to minimize the cost. If U is set too large, the optimal solution is no control at all (open loop) asymptotically as U approaches infinity. Therefore, some compromise must be reached.

Through the use of dynamic programming, we can find the optimal $u(t_i)$ to minimize the cost function in Equation (3-12) given the dynamics constraints in Equation (3-9). This controller has the general form

$$u(t_i) = -\underline{G}_c^*(t_i)\bar{x}_a(t_i) + Ey_d \tag{3-13}$$

From Maybeck (Ref. 4:Vol. 3) we find that the gains in Equation (3-13) are given by

$$\underline{G}_c^*(t_i) = [G_{c1}^*(t_i) \mid G_{c2}^*(t_i)] = [U + \underline{B}_{da}^T\underline{K}_c(t_{i+1})\underline{B}_{da}]^{-1}[\underline{B}_{da}^T\underline{K}_c(t_{i+1})\underline{\Phi}_a] \tag{3-14}$$

$$E = [G_{c1}^*(t_i) - G_{c2}^*(t_i)K_{c22}^{-1}(t_i)K_{c12}^T(t_i)]\ \Pi_{12} + \Pi_{22} \tag{3-15}$$

where $\underline{K}_c$ is derived from the backward Riccati difference equation

$$\begin{aligned} K_c(t_i) = {} & \underline{X} + \underline{\Phi}_a^T(t_{i+1},t_i)\underline{K}_c(t_{i+1})\underline{\Phi}_a(t_{i+1},t_i) \\ & - [\underline{B}_{da}^T\underline{K}_c(t_{i+1})\underline{\Phi}_a(t_{i+1},t_i)]^T\underline{G}_c^*(t_i) \end{aligned} \tag{3-16}$$

solved backward from $\underline{K}_c(t_N) = \underline{X}_f$ and

$$\underline{\Pi} = \begin{bmatrix} \Pi_{11} & \Pi_{12} \\ \Pi_{21} & \Pi_{22} \end{bmatrix} = \begin{bmatrix} [\Phi_B(t_{i+1},t_i) - I] & B_d \\ C & \emptyset \end{bmatrix}^{-1} \tag{3-17}$$

A through derivation of Equation (3-17) is given in Maybeck (Ref. 4: Vol. 3). It is founded in part on the idea that we need a nominal $u_\emptyset(t_i)$ which will produce the equilibrium solution $x(t_i)=x_\emptyset$ thus yielding $y_d = Cx_\emptyset$. Note that in Equation (3-16) we are guaranteed a solution to the Riccati equation (Ref. 4:Vol. 3) because our system is both stabilizable and detectable. For time invariant system models and constant weighting matrices, the solution often consists of a terminal transient followed by steady state behavior as shown in Figure 3-2. It was felt that these steady state values would produce desired behavior over the time period of interest. Therefore, the steady state (constant) controller gains are used for all time.

III.2 TRACKER MODEL AND KALMAN FILTER

Before beginning a development of the tracker we must first have a model for the beam's target. In the case of Zicker and Moose (Ref. 18 and 10), the target was represented by a zero-mean first order Gauss-Markov position process, as produced by the first order dynamics equation:

$$\dot{\underset{\sim}{x}}(t) = -\frac{1}{\tau}\underset{\sim}{x}(t) + G\underset{\sim}{w}(t) \qquad (3\text{-}18)$$

where $\underset{\sim}{x}(t)$ is the position state, τ is the position correlation time constant, and $\underset{\sim}{w}(t)$ is zero-mean white Gaussian noise of strength Q_T. Obviously this is a very benign target as shown in Figure 3-3. Such a simple model was chosen because Zicker and Moose were attempting first-cut feasibility studies, and there was a desire to avoid overly complex controller designs.

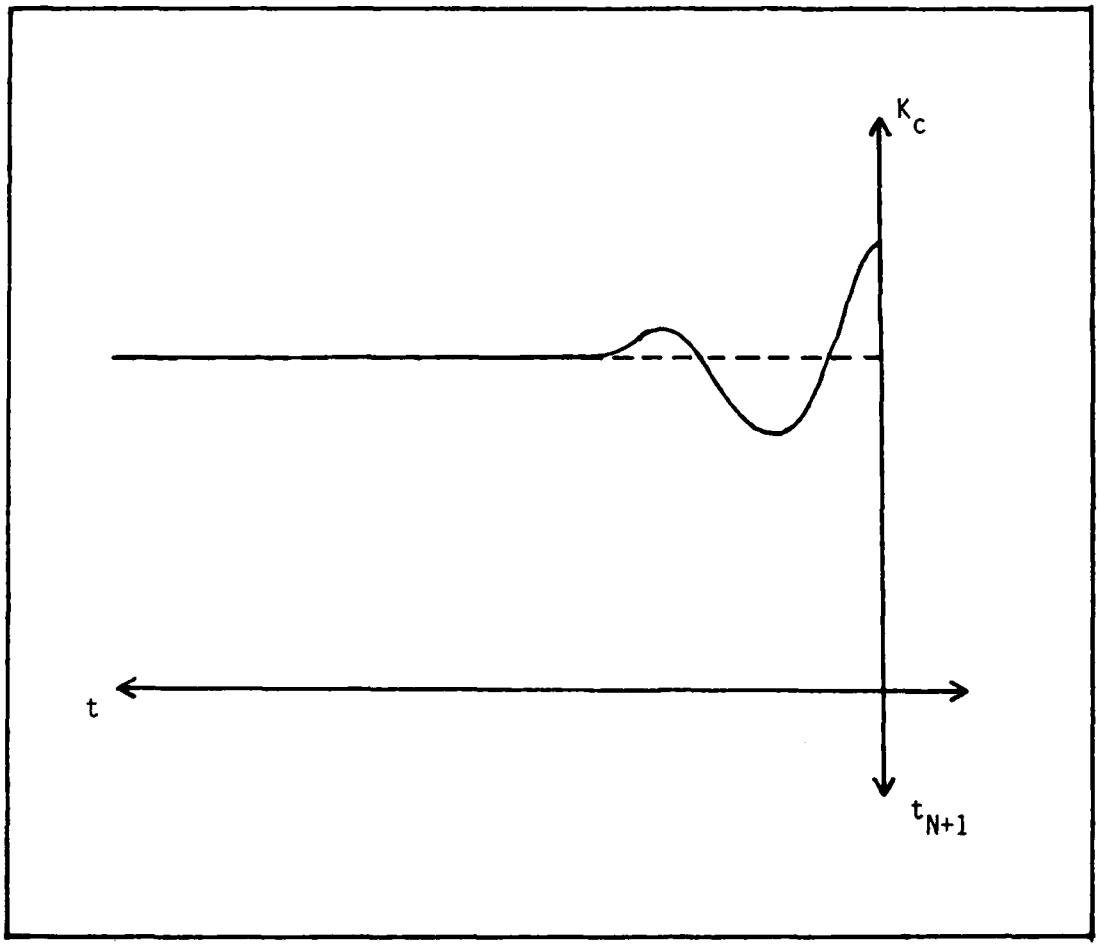


Figure 3-2. Riccati Equation Solution

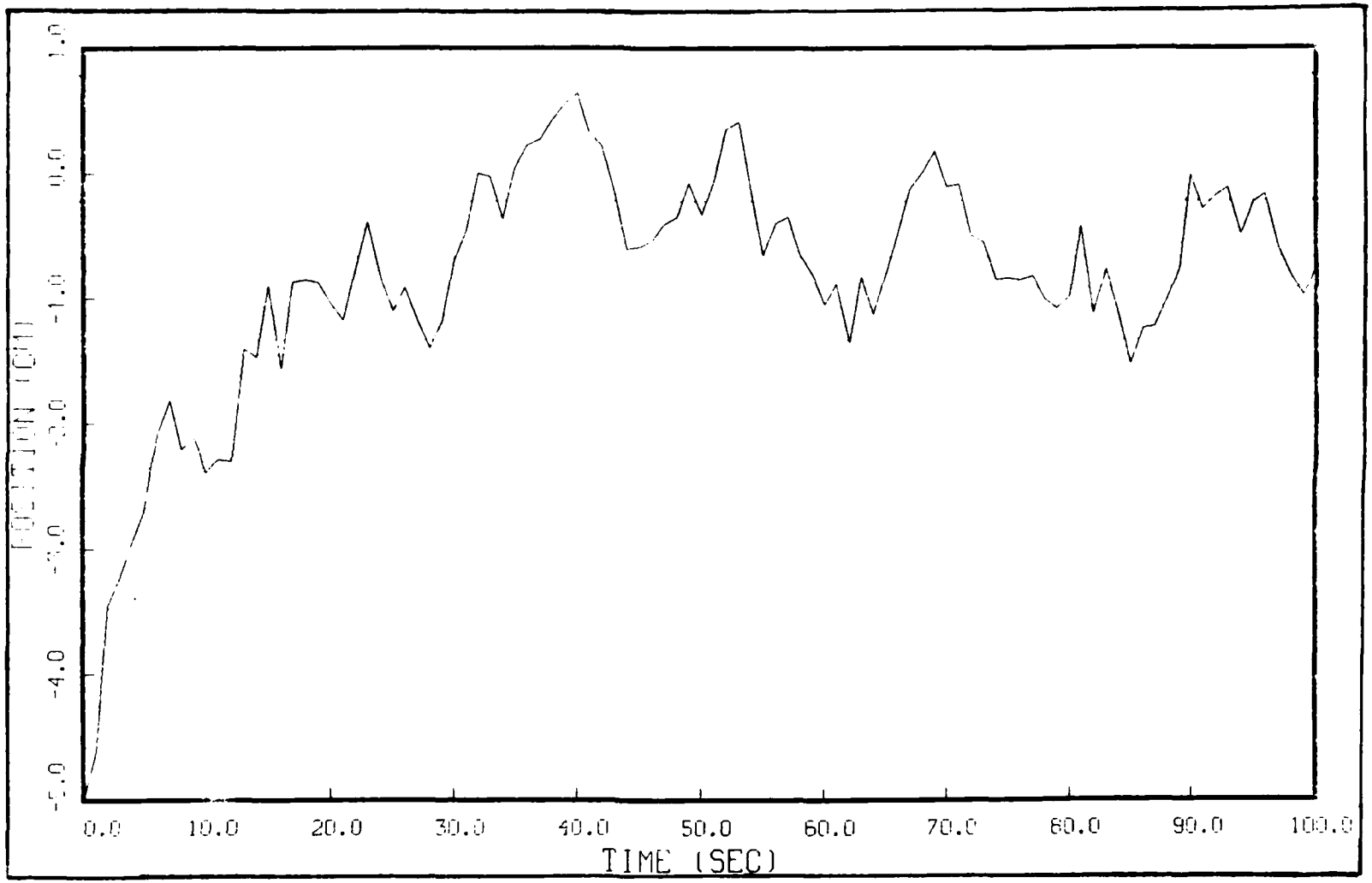


Figure 3-3. First Order Gauss-Markov Target Position Model

For this thesis the model for the target is a first order Gauss-Markov acceleration process shown in Equation (3-19):

$$\begin{bmatrix} \dot{\underset{\sim}{x}}_p(t) \\ \dot{\underset{\sim}{x}}_v(t) \\ \dot{\underset{\sim}{x}}_A(t) \end{bmatrix} = \begin{bmatrix} \emptyset & 1 & \emptyset \\ \emptyset & \emptyset & 1 \\ \emptyset & \emptyset & -\frac{1}{\tau} \end{bmatrix} \begin{bmatrix} \underset{\sim}{x}_p(t) \\ \underset{\sim}{x}_v(t) \\ \underset{\sim}{x}_A(t) \end{bmatrix} + \begin{bmatrix} \emptyset \\ \emptyset \\ G \end{bmatrix} w_T(t) \qquad (3\text{-}19)$$

where $\underset{\sim}{x}_p(t)$ is the target position state, $\underset{\sim}{x}_v(t)$ is the target velocity state, $\underset{\sim}{x}_A(t)$ is the acceleration, τ is the acceleration correlation time constant, and $\underset{\sim}{w}_T(t)$ is zero mean white Gaussian noise of strength Q_T associated with the target model (and thus the subscript T). The new model is a much more realistic target and has been used successfully for many applications. A typical position state trajectory is shown in Figure 3-4. As opposed to Figure 3-3, the target is seen to have a divergent nature (in this instance it goes to $-\infty$), thereby making a more difficult tracking problem. Note that there are now poles at the origin of the s-plane. Since there are no control inputs that directly affect the target and this system is inherently astable, we see that the system is no longer stabilizable. Without this condition we are no longer guaranteed a solution of the steady state Riccati equation used for generating controller gains. This problem will be addressed further in the next section of this chapter.

Equation (3-19) has the special form of a linear time-invariant system driven by white Gaussian noise. Further, assume that there will be discrete-time measurements of the target position state as shown by the equation:

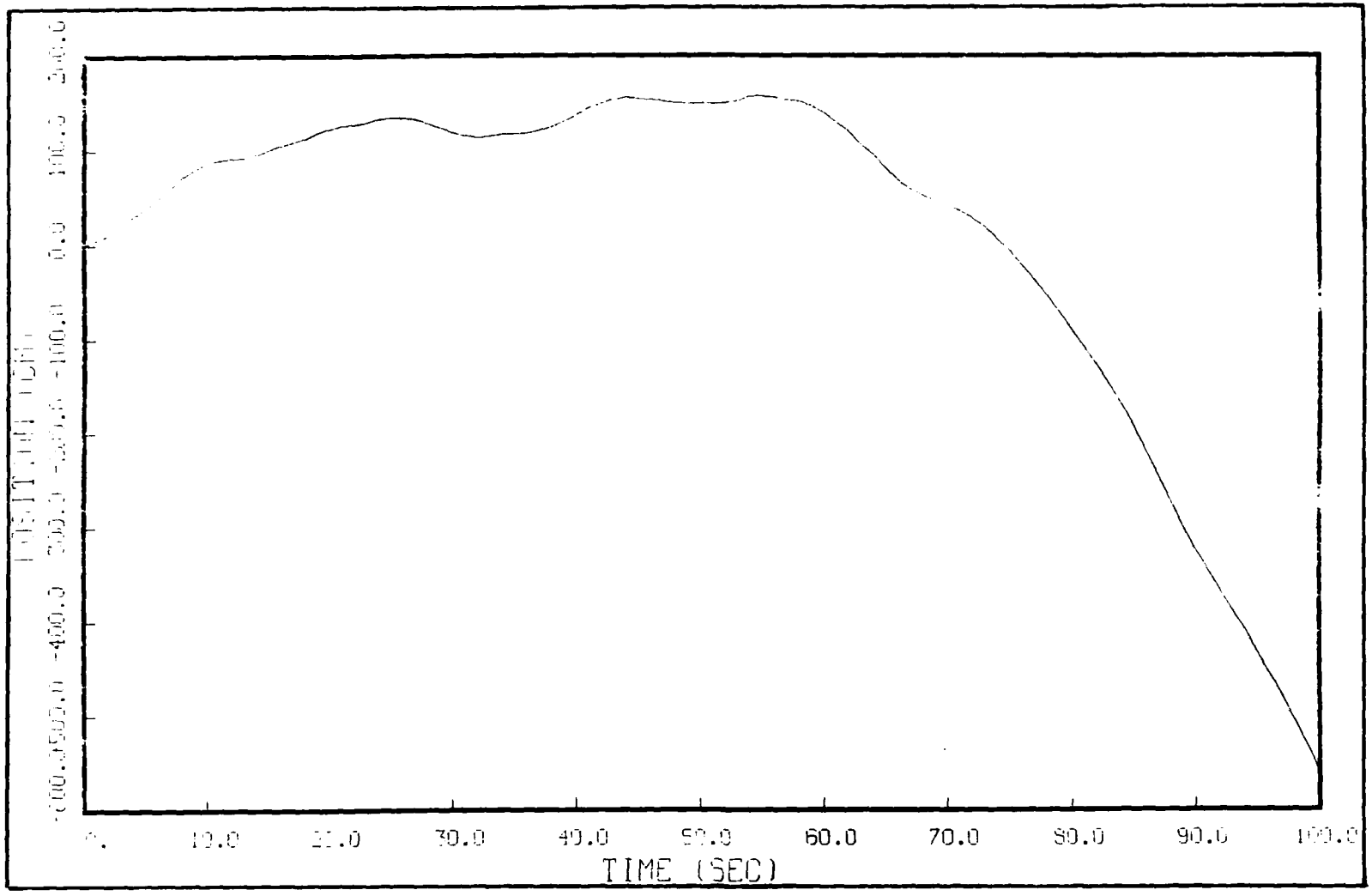


Figure 3-4. First Order Gauss-Markov Target Acceleration Model

$$z(t_i) = [1 \quad \emptyset \quad \emptyset] \begin{bmatrix} \underset{\sim}{x}_p(t_i) \\ \underset{\sim}{x}_v(t_i) \\ \underset{\sim}{x}_A(t_i) \end{bmatrix} + v(t_i) \tag{3-20}$$

where $v(t_i)$ is discrete-time white Gaussian noise of variance R and is assumed to be uncorrelated with the dynamics driving noise $\underset{\sim}{w}_T(t)$. We now have enough information to derive a state estimator using standard Kalman filter techniques. The state estimates and self-computed state error covariances will propagate between measurement times as:

$$\hat{\bar{x}}(t_i^-) = \underline{\Phi}_T(t_i,t_{i-1})\hat{\bar{x}}(t_{i-1}^+)$$

$$\underline{P}(t_i^-) = \underline{\Phi}_T(t_i,t_{i-1})\underline{P}(t_{i-1}^+)\underline{\Phi}_T^T(t_i,t_{i-1})$$

$$+ \int_{t_{i-1}}^{t_i} \underline{\Phi}_T(t_i,\tau)\underline{G}_T(\tau)\underline{Q}_T(\tau)\underline{G}_T^T(\tau)\underline{\Phi}_T^T(t_i,\tau)d\tau \tag{3-21}$$

where the subscript T denotes "target". At measurement times the estimates and covariances are updated as:

$$\hat{\bar{x}}(t_i^+) = \hat{\bar{x}}(t_i^-) + \underline{K}(t_i)[z(t_i) - \underline{H}_T(t_i)\hat{\bar{x}}(t_i^-)]$$

$$\underline{P}(t_i^+) = \underline{P}(t_i^-) - \underline{K}(t_i)\underline{H}_T(t_i)\underline{P}(t_i^-)$$

$$\underline{K}(t_i) = \underline{P}(t_i^-)\underline{H}_T^T(t_i)[\underline{H}_T(t_i)\underline{P}(t_i^-)\underline{H}_T^T(t_i) + R_T(t_i)]^{-1} \tag{3-22}$$

The initial conditions are given as:

$$\hat{\bar{x}}(t_0) = \bar{x}_0$$

$$\underline{P}(t_0) = \underline{P}_0$$

Note that as measurement noise variance $R(t_i)$ is increased, the filter places more emphasis on the filter's own dynamics model. If the dynamics driving noise strength Q is relatively high, the filter places relatively heavy emphasis on the incoming measurements $z(t_i)$. These trends are seen by examining the Kalman filter gain $(K(t_i))$ in Equation (3-22).

III.3 PI TRACKER

Now that there are equations for the beam and target, we can form a new augmented state space which contains both sets of variables:

$$\begin{bmatrix} \dot{\underset{\sim}{x}}(t) \\ \dot{\underset{\sim}{x}}_p(t) \\ \dot{\underset{\sim}{x}}_v(t) \\ \dot{\underset{\sim}{x}}_A(t) \end{bmatrix} = \begin{bmatrix} -\frac{1}{T} & \emptyset & \emptyset & \emptyset \\ \emptyset & -\varepsilon_1 & 1 & \emptyset \\ \emptyset & \emptyset & -\varepsilon_2 & 1 \\ \emptyset & \emptyset & \emptyset & -\frac{1}{\tau} \end{bmatrix} \begin{bmatrix} \underset{\sim}{x}(t) \\ \underset{\sim}{x}_p(t) \\ \underset{\sim}{x}_v(t) \\ \underset{\sim}{x}_A(t) \end{bmatrix} + \begin{bmatrix} 1 \\ \emptyset \\ \emptyset \\ \emptyset \end{bmatrix} u(t) + \begin{bmatrix} \underset{\sim}{w}(t) \\ \emptyset \\ \emptyset \\ \underset{\sim}{w}_T(t) \end{bmatrix} \tag{3-23}$$

where $\underset{\sim}{w}(t)$ is zero-mean white Gaussian noise of strength Q, and $\underset{\sim}{w}_T(t)$ is zero-mean white Gaussian noise of strength Q_T. The inclusion of the terms ε_1 and ε_2 is necessary in order to force Equation (3-23) to be stabilizable and detectable. Otherwise, the system is astable due to poles at the origin of the s-plane. Values of $\varepsilon_1=\varepsilon_2=.0001$ were chosen to ensure at least an order of magnitude difference between the poles forced away from the origin and the time constant terms $\frac{1}{T}$ and $\frac{1}{\tau}$. Note, this approach applies to the controller gain development via Riccati equation solution exclusively; all filter and dynamics equations assume pure integrations of acceleration and velocity states.

We now set about discretizing the system in Equation (3-23) and including the pseudointegral terms:

$$\begin{bmatrix} \underset{\sim}{x}(t_{i+1}) \\ \underset{\sim}{x}_p(t_{i+1}) \\ \underset{\sim}{x}_v(t_{i+1}) \\ \underset{\sim}{x}_A(t_{i+1}) \\ \underset{\sim}{q}(t_{i+1}) \end{bmatrix} = \left[\begin{array}{c|ccc|c} \Phi_B & \emptyset & \emptyset & \emptyset & \emptyset \\ \hline \emptyset & & & & \emptyset \\ \emptyset & & \underline{\Phi}_T & & \emptyset \\ \emptyset & & & & \emptyset \\ \hline 1 & -1 & \emptyset & \emptyset & 1 \end{array}\right] \begin{bmatrix} \underset{\sim}{x}(t_i) \\ \underset{\sim}{x}_p(t_i) \\ \underset{\sim}{x}_v(t_i) \\ \underset{\sim}{x}_A(t_i) \\ \underset{\sim}{q}(t_i) \end{bmatrix} + \begin{bmatrix} B_d \\ \emptyset \\ \emptyset \\ \emptyset \\ \emptyset \end{bmatrix} u(t_i) + \begin{bmatrix} \underset{\sim}{w}_d(t_i) \\ \emptyset \\ \emptyset \\ \underset{\sim}{w}_{dT}(t_i) \\ \emptyset \end{bmatrix} \tag{3-24a}$$

where $\underset{\sim}{w}_d(t_i)$ is zero mean discrete-time white Gaussian noise of strength Q_d and $\underset{\sim}{w}_{dT}(t_i)$ is zero mean discrete-time white Gaussian noise of strength Q_{dT}. The term $u(t_i)$ is the control input used to achieve the desired system characteristics. This equation can be rewritten as

$$\overline{\underset{\sim}{x}}_A(t_{i+1}) = \underline{\Phi}_A(t_{i+1},t_i)\overline{\underset{\sim}{x}}_A(t_i) + \underline{B}_{dA}\, u(t_i) + \overline{\underset{\sim}{w}}_{dA}(t_i) \tag{3-24b}$$

In the case of a tracker we are trying to drive the quantity

$$\underset{\sim}{e}(t_i) = \underset{\sim}{x}(t_i) - \underset{\sim}{x}_p(t_i) \tag{3-25}$$

toward $\emptyset$ so a quadratic penalty is placed on this error, as:

$$\tfrac{1}{2}X_{11}e(t_i)^2 = \tfrac{1}{2}[x(t_i)\ x_p(t_i)]\begin{bmatrix} X_{11} & -X_{11} \\ -X_{11} & X_{11} \end{bmatrix} \begin{bmatrix} x(t_i) \\ x_p(t_i) \end{bmatrix} \tag{3-26a}$$

Therefore, the performance index used to derive our new controller must include this error term. The new cost function upon which the

deterministic optimal LQ controller is based will be

$$J = \frac{1}{2}\sum_{i=1}^{N}\left\{\bar{x}_A^T(t_i)\begin{bmatrix} X_{11} & -X_{11} & \emptyset & \emptyset & \emptyset \\ -X_{11} & X_{11} & \emptyset & \emptyset & \emptyset \\ \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & \emptyset & \emptyset & \emptyset & X_{55} \end{bmatrix}\bar{x}_A(t_i) + Uu^2(t_i)\right\}$$

$$+ \frac{1}{2}\bar{x}_A^T(t_{N+1})\begin{bmatrix} X_{11f} & -X_{11f} & \emptyset & \emptyset & \emptyset \\ -X_{11f} & X_{11f} & \emptyset & \emptyset & \emptyset \\ \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & \emptyset & \emptyset & \emptyset & \emptyset \\ \emptyset & \emptyset & \emptyset & \emptyset & X_{55f} \end{bmatrix}\bar{x}_A(t_{N+1}) \qquad (3\text{-}26b)$$

Note that since $x_v(t_i)$ and $x_A(t_i)$ cannot be affected by control, they are not weighted into the cost function. A discussion of how X_{11} and X_{55} affect the resulting controller performance is given in Chapter 5. This equation can also be written as

$$J = \frac{1}{2}\sum_{i=0}^{N}\{\bar{x}_A^T(t_i)\,\underline{X}_A\,\bar{x}_A(t_i) + Uu^2(t_i)\} + \frac{1}{2}\bar{x}_A^T(t_{N+1})\,\underline{X}_{Af}\,\bar{x}_A(t_{N+1}) \qquad (3\text{-}27)$$

For steady state constant-gain designs, we let N go to ∞ and ignore the terminal transient in the resulting Riccati difference equation used for controller gain computations.

By the use of dynamic programming we find that the controller that minimizes a cost of the form of Equation (3-27) is given as:

$$u(t_i) = -G_{c1}^{*}\bar{\underset{\sim}{x}}(t_i) - G_{c2}^{*}x_p(t_i) - G_{c3}^{*}x_v(t_i) - G_{c4}^{*}x_A(t_i) - G_{c5}^{*}q(t_i) + Ey_d \tag{3-28}$$

as is consistent with the structure of Equation (3-13). The term y_d is a nonzero setpoint input. If the quantity in Equation (3-25) is to be zero, then y_d is $\emptyset$. If a particular separation is desired between the beam centroid and target (as to locate on a point offset from the centroid by a known amount), y_d is set accordingly. The controller gains of Equation (3-28) are given by

$$\underline{G}_{cA}^{*} = [G_{c1}^{*} \mid G_{c2}^{*} \mid G_{c3}^{*} \mid G_{c4}^{*} \mid G_{c5}^{*}] = [U + \underline{B}_{dA}^{T}\,\underline{K}_{cA}\,\underline{B}_{dA}]^{-1}[\underline{B}_{dA}\,\underline{K}_{cA}\,\underline{\Phi}_A] \tag{3-29}$$

where B_{dA} and $\underline{\Phi}_A$ are the same as in Euqations (3-24a/b) and $\underline{K}_{cA}$ is given by the algebraic Riccati equation (the backward Ricatti difference equation in steady state condition; analogous to Equation (3-16) but with successive values of $\underline{K}_{cA}$ equated to each other):

$$\underline{K}_{cA} = \underline{X}_A + \underline{\Phi}_A^{T}\,\underline{K}_{cA}\,\underline{\Phi}_A - [\underline{B}_{dA}\,\underline{K}_{cA}\,\underline{\Phi}_A]^{T}\,\underline{G}^{*}_{cA} \tag{3-30}$$

Finally, the controller term applied to the nonzero setpoint input in Equation (3-28) is given as

$$E = \left\{ [G_{c1}^{*} \mid G_{c2}^{*} \mid G_{c3}^{*} \mid G_{c4}^{*}] - G_{c5}^{*}\,K_{cA55}^{-1} \begin{vmatrix} K_{cA15} \\ K_{cA25} \\ K_{cA35} \\ K_{cA45} \end{vmatrix}^{T} \right\} \begin{vmatrix} \Pi_{15} \\ \Pi_{25} \\ \Pi_{35} \\ \Pi_{45} \end{vmatrix} + \Pi_{55} \tag{3-31}$$

where

$$\underline{\Pi} = \begin{bmatrix} \Pi_{11} & \cdots & \Pi_{15} \\ \vdots & & \vdots \\ \Pi_{51} & \cdots & \Pi_{55} \end{bmatrix} = \left[\begin{array}{c|ccc|c} (\Phi_B - I) & \emptyset & \emptyset & \emptyset & B_d \\ \hline \emptyset & & & & \emptyset \\ \emptyset & & \underline{\Phi}_T - \underline{I} & & \emptyset \\ \emptyset & & & & \emptyset \\ \hline 1 & -1 & \emptyset & \emptyset & \emptyset \end{array}\right]^{-1} \tag{3-32}$$

The term Φ_B is the state transition matrix of the beam and $\underline{\Phi}_T$ is the state transition matrix for the target model which included imperfect integrations as in Equation (3-23).

III.4 SUMMARY

This chapter has been a presentation of the PI controller to be used in this thesis. First, PI regulator equations were developed in a manner similar to that of Moose (Ref. 10). Second, the target model was presented along with the Kalman filter equations used to estimate the target's acceleration, velocity, and position states. Finally, the PI tracker was developed. When deriving the tracker controller gains, it became necessary to force the target model integrator poles inside the left half plane (or the unit circle if we are in the z-domain). This step was necessary to guarantee steady-state gains from the matrix Riccati equations. Once these gains are derived they are placed in the PI structure as displayed in Figure 3-1. Since access to the true states of the system is not realistic, the Kalman and Meer filters are embedded as shown in Figure 3-5.

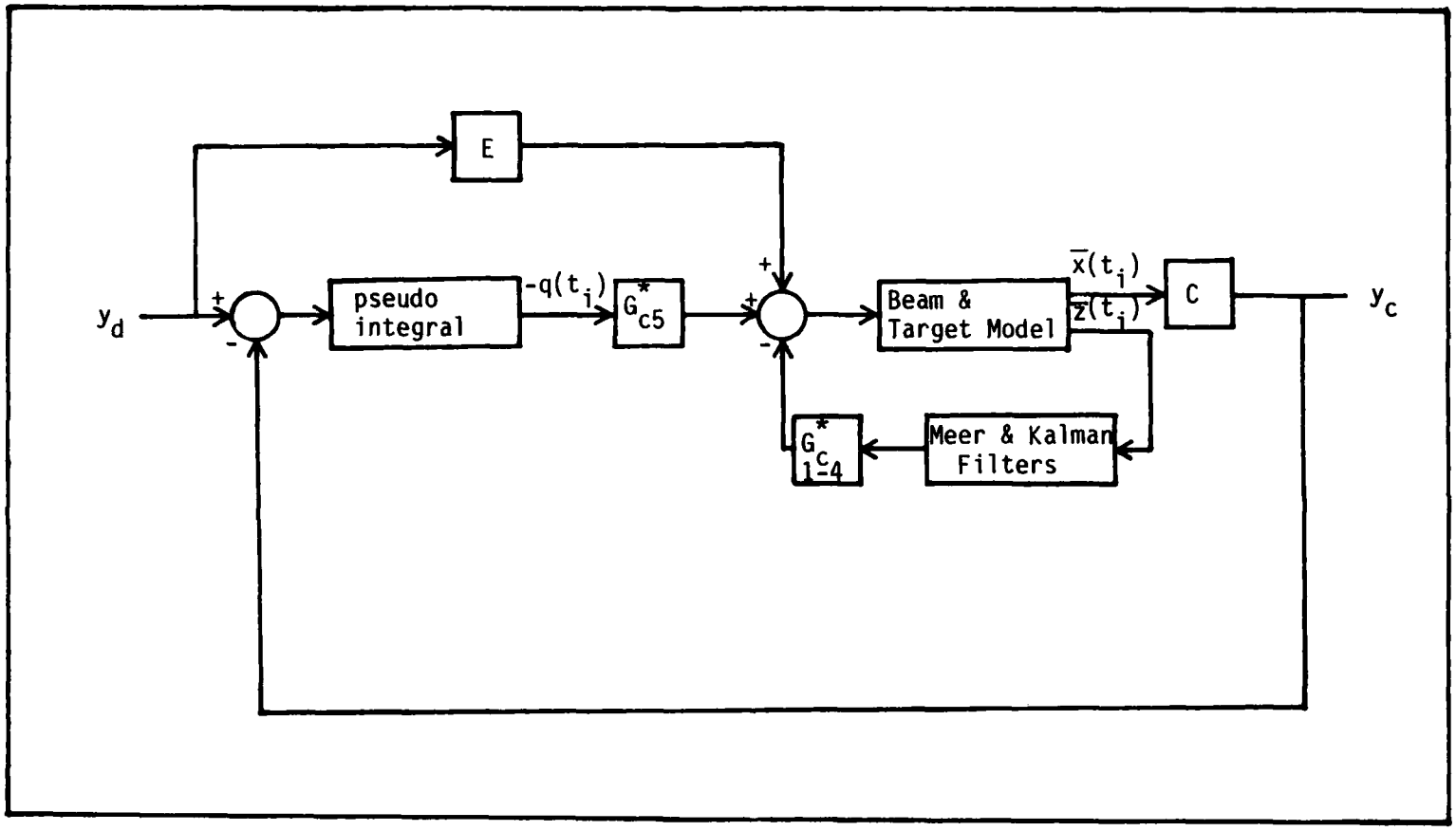


Figure 3-5. Filter/Controller Structure

IV. ANALYSIS TOOLS AND METHODS

Now that the particle beam PI controller has been designed as well as the filters for estimating the states of the beam and target, attention can now be focused on the analysis tools and methods which are used to evaluate the performance capabilities of those algorithms in Chapter V. As noted at the end of Chapter I, the filter/controller structure is tested using Monte Carlo simulations because covariance analysis techniques are impossible for the particle beam problem. The Monte Carlo runs produce the sample statistics required to analyze the effectiveness of the system.

This chapter is divided into four sections. The first section gives a general description of the analysis tools used in this thesis as well as a brief history of their development. The second section describes the method in which these tools are used to form the performance parameters of the system (RMS errors). These performance parameters indicate how well the filter/controller combination is behaving under various conditions. The third section describes the sensitivity analysis which was performed on the corrected form of the PI controller originally conceived in Moose's thesis (Ref. 10). The remaining section addresses the robustness study which is carried out using the PI tracker described in Chapter III. This controller incorporates a first order Gauss-Markov acceleration model for target dynamics.

IV.1 ANALYSIS TOOLS

The analysis tool for the problem has been modified several times before reaching its final form. At the heart of the simulation is a

software package by the name of SOFE: Simulation for Optimal Filter Evaluation (Ref. 11). This program was written by Stanton H. Musick of the Air Force Avionics Laboratory to analyze the performance of Kalman filters. SOFE allows the designer to embed a truth model (for accurate real world simulation) as well as a typically simpler design model (upon which to base a filter) into the software. Propagation of the state estimates and filter-computed error covariances is achieved by integrating these differential equations which describe the quantities from one sample time to the next. Updates are performed at prespecified sample times. These differential equations have the general form

$$\dot{\underset{\sim}{\bar{x}}}(t) = \bar{f}[\underset{\sim}{\bar{x}}(t),\underset{\sim}{\bar{u}}(t),t] + \underline{G}(t)\underset{\sim}{\bar{w}}(t) \tag{4-1}$$

where $\underset{\sim}{\bar{w}}(t)$ is zero-mean white Gaussian noise of strength $\underline{Q}$. SOFE will solve Equation (4-1) by first solving the deterministic differential equation

$$\dot{\bar{x}}(t) = \bar{f}[\bar{x}(t),\bar{u}(t),t] \tag{4-2}$$

between sample periods and incorporating the stochastic term as

$$\bar{x}_t = \bar{x}_t + \overline{GAUSS}\ (\emptyset,\ SNDEV_Q) \tag{4-3}$$

where $\bar{x}_t$ is the solution of Equation (4-2) after each sample period, and the expression $\overline{GAUSS}(\cdot,\cdot)$ is a vector random number generator term, where the terms within the parentheses indicate zero mean and standard deviation respectively. The $SNDEV_Q$ term is the standard deviation of $\underline{Q}_d$ (i.e., a matrix square root of $\underline{Q}_d$) where $\underline{Q}_d$ is the second moment of the equivalent discrete time noise to represent the effect of $\underline{G}(t)\bar{w}(t)$

over one sample period. The general measurement equation is given as:

$$\underset{\sim}{\overline{z}}(t_i) = \overline{h}_t[\underset{\sim}{\overline{x}}(t_i),t_i] + \underset{\sim}{\overline{V}}_t(t_i) \tag{4-4}$$

The equation used by the truth model to supply noisy measurements is

$$z(t_i) = \overline{GAUSS}\ [\overline{h}_t(\overline{x}_t(t_i),t_i),\ SNDEV_R] \tag{4-5}$$

The term $\overline{h}_t[x_t(t_i),t_i]$ is the true measurement that would be seen if the measurements were not noise corrupted. $SNDEV_R$ is the standard deviation of $\underline{R}_t$ where $\underline{R}_t$ is the discrete time variance of the measurement noise process $\underset{\sim}{\overline{V}}_t$. Once these measurements are made, they can be fed to the Kalman filter which estimates the states based upon its own dynamics model and internally computed error covariances as well as the measurement residuals. Several sample runs can be generated (vs. time) and the corresponding states and covariances are stored in an external file.

In order to process this data from SOFE, the postprocessor SOFEPL was written (Ref. 12). SOFEPL takes the data from several SOFE runs (such as state values and error covariances) and forms the statistics desired by the user. These statistics include such values as means and standard deviations of states and errors over time. With SOFE and the postprocessor SOFEPL, we have effective tools for evaluating the true performance of a Kalman filter, assuming truth model adequacy as described at the end of Chapter I. However, when dealing with the particle beam problem, a Meer filter rather than a standard Kalman filter must be used to estimate the position state of the beam.

When Meer set out to use this software package for the particle beam application, certain modifications were required to be made. As noted in Chapter II, propagation and update equations for the Meer

and Kalman filters are exactly the same. The differences arise at the measurement times. SOFE expects measurements to occur at prespecified discrete-time intervals, whereas in the particle beam problem, measurements are only available when photo-electron events occur. The events behave as a Poisson space-time point process. Even though the beam exhibits a spatially Gaussian behavior across the detector array, the event times are represented by a Poisson distribution. Meer wrote software which generated event times in a Poisson manner, while simultaneously distributing the events in a Gaussian manner across the detector. He also included a subroutine for filter "pruning" using the Best Half Method as discussed in the fifth section of Chapter II.

After Meer's work, Zicker (Ref. 18) had the task of incorporating a proportional gain controller into the system in order to regulate the beam to a desired point as well as subsequently to track an intended target. Zicker's primary contributions to the software were the addition of the Merge Method of filter "pruning" (as discussed in the fifth section of Chapter 2) and subroutines for calculating the controller gains and feedback control inputs. Zicker's main problem involved the calculation of control inputs. These inputs were added to the system at pre-specified intervals; however, since the measurements of the events are randomly separated in time, one cannot anticipate having an updated state estimate $\hat{\bar{x}}(t_i^+)$ at each control application sample time, to form $u(t_i) = -\underline{G}_c^*(t_i)\hat{\bar{x}}(t_i^+)$ as in the standard LQG control problem. To address this problem, Zicker propagated the Meer beam estimate to the control sample times so that the estimate at the control time would be appropriate (Ref. 18).

The next modification to the simulation software was made when Moose (Ref. 10) added a PI controller to regulate the beam and track an intended target modelled as a first order Gauss-Markov position process. The control gain subroutine and feedback control subroutine needed rederivation; however, no other major changes were incorporated. It was discovered later that the cost function used to calculate the controller gains was in error, so the validity of the results came into question.

The final modification made to the simulation software comes in this thesis, in which a first order Gauss-Markov acceleration process is used to describe the intended target more accurately than is possible with the simplistic target model used for the initial feasibility studies. To accomplish this end, the controller gain and feedback subroutines are changed as well as the order of the truth adn filter models. With the simulation tools finally in hand, we can set about describing the analysis method.

IV.2 ANALYSIS METHOD

The purpose of this section is to describe the performance parameter used in this thesis--root mean square (RMS) error. The previous section describes the tools available to compute the RMS error. This section begins by describing how the data is collected by SOFE, as well as which data is retained and which is not. Next, we look at what RMS errors are and why they are used. Finally, we look at how the software is adjusted to accommodate RMS errors.

Each Monte Carlo simulation involves runs which begin at zero seconds and conclude at 100 seconds. Several runs must be carried out due to the statistical nature of our performance analysis. Zicker

showed that 200 runs was enough to generate truly representative statistics for our problem (Ref. 18). The more runs we have, the more closely the computed sample statistics will represent the true ensemble average statistics. Once the data is collected in SOFE we turn to SOFEPL to calculate the time histories of RMS errors and then also the time-averaged value of these RMS errors.

The error for the tracker problem is the difference between the target and beam position states:

$$\underset{\sim}{e}(t_i) = \underset{\sim}{x}(t_i) - \underset{\sim}{x}_p(t_i) \tag{4-6}$$

where $\underset{\sim}{x}(t_i)$ and $\underset{\sim}{x}_p(t_i)$ are defined in Equations (3-1) and (3-19), respectively. SOFEPL can take the errors at each time during a run and average them over all the runs (sample ensemble average). Therefore, there is now a mean value and standard deviation for each error of interest, for each time t_i.

At this point SOFEPL can generate the RMS errors for each time t_i by using the equation

$$RMS_e(t_i) = \sqrt{m_e^2(t_i) + \sigma_e^2(t_i)} \tag{4-7}$$

The terms $m_e(t_i)$ and $\sigma_e(t_i)$ are the mean error and standard deviation of errors for each time t_i. Now that we have a history of RMS errors versus time, SOFEPL computes the time average. Since the earlier times involve a great deal of transients, we will only look at times where $t_i \geq 50$ seconds. During these times the filter/controller has had time to move toward its steady state solution. We now have one RMS error value representing 200 runs where the data was averaged only over the

time interval corresponding to $t_i \geq 50$ seconds. If time averages of actual errors had been used to evaluate the performance instead of time averages of RMS errors, a positive error in one term in the averaging sum might cancel a negative error in the next, therefore giving lower number averages of the errors. Thus, time-averaged RMS errors are an appropriate measure of actual performance.

SOFEPL is designed for use with errors between the truth and filter model, i.e., to evaluate filter estimation performance instead of control system effectiveness. However, the error we use in Equation (4-6) is between the true beam and target position. Fortunately, SOFEPL is easily modified to accommodate our needs. By the insertion of two simple subroutines, the postprocessor is capable of computing the RMS errors involved in Equations (4-6) and (4-7). The first subroutine is used to store values needed for the calculations. The second subroutine is used to calculate the means and standard deviations (Ref. 12). With this capability in hand, we are now ready to look at exactly what types of studies are to be carried out in this thesis.

IV.3 SENSITIVITY ANALYSIS

The first task to be carried out in this thesis is to perform a sensitivity analysis on the corrected PI controller originally designed by Moose (Ref. 10). As noted before, the controller, Kalman filter, and truth model all assume a single-state Gauss-Markov position model. To run a sensitivity analysis, we begin by choosing the weighting matrices of our cost function based upon the controller's performance at a nominal parameter setting. Once the weighting matrices are chosen, we set about varying one parameter at a time (in the truth and filter model) while

leaving the other parameters at their nominal values. In this manner, the controller effectiveness for several "real-world" environments can be studied. Specifically, the filter model is told what the environment looks like, and controller effectiveness can be studied, parameterized by important descriptors of real world characteristics.

The sensitivity analysis completed in this thesis involved only the parameters associated with the target model because Moose had already accomplished a correct analysis for the regulator parameters (Ref. 10). The parameters associated with the beam are as follows: standard deviation of process noise, ($\sqrt{Q}$), beam dispersion (R), time constant (T), filter depth (D), the expected number of signal events (count), and the detector length (L). These parameters were discussed previously in the first section of Chapter II. The standard deviation of process noise indicates our confidence in the model, and a value of 0.2 was chosen, in correspondence with the nominal value previously identified. Beam dispersion indicates the spread of signal events across the detector. As a nominal value, R=0.5 was chosen. The time constant of the beam was 20 seconds, the depth was made 3, and the length of our detector is 10 cm. The expected number of signal events was a value of 100 during the 100-second interval of time for each run. In addition to these, it was desired to have a parameter which indicates the relative frequency of occurrence of signal and noise events. The signal to noise ratio is given as

$$SNR = \frac{\Lambda\ (2\pi R)^{\frac{1}{2}}}{\lambda_n L} \tag{4-8}$$

where Λ and R are from the signal rate parameter given in Equation (2-1), L is the detector length and λ_n is the noise rate parameter (constant).

The parameters associated with the target are as follows: acceleration time constant τ, measurement noise variance R, and dynamic driving noise strength Q_T. Each is defined in the third section of Chapter 3. These are the parameters to be varied in the sensitivity analysis. The dynamic driving noise strength indicates our confidence in the target dynamics model and has a nominal value of $Q_T = 0.1$. Measurement noise variance R indicates that our sensors are noise-corrupted; the nominal value is R = 0.5. The target time constant indicates the maneuverability of the target and will be set to 10 seconds. Because the fastest transient at nominal conditions has a time constant of 10 seconds, the sampling period of the Kalman filter and feedback control period will be set to 1 second. Shannon's sampling theorem dictates that there should be at least a factor of two difference between sample rate and the highest signal frequency content of interest; however, the engineering rule of thumb is a factor of ten.

IV.4 ROBUSTNESS ANALYSIS

The final topic to be covered is the robustness analysis of this thesis. When a filter and controller are designed, they assume that the system model adequately describes what actually happens in the real world. By varying the truth model without telling the filter/controller, one can determine how well the design stands up against a changing or off-design real world application. During this thesis effort, two different types of robustness studies were carried out. First, the corrected PI controller of Moose was tested by inputting constant and sinusoidal

disturbances into truth model beam equation without telling the Meer filter (or, equivalently, into the target position simulation without telling the Kalman filter). Second, the PI controller which incorporates a first order Gauss-Markov acceleration model for the target was analyzed by varying target parameters in the truth model without telling the Kalman filter. In order to recover some of the robustness qualities inherent in full state feedback, Loop Transmission Recovery (LTR) techniques (Refs. 6 and 16) are applied. Both of the analysis techniques will be described in this section.

The primary purpose behind the PI controller structure is to allow tracking with zero-mean steady state errors even in the face of unmodelled constant disturbances. The reasoning is discussed in the first section of Chapter III. There are basically two types of disturbances added to the truth model of the beam: a constant disturbance and a sinusoidal disturbance. These were incorporated into the truth model as follows:

$$x(t_{i+1}) = \Phi(t_{i+1},t_i)x(t_i) + B_d u(t_i) + d(t_i) \qquad (4\text{-}9)$$

where

$$d(t_i) = d_0 \qquad \text{(constant)} \qquad (4\text{-}10a)$$

$$d(t_i) = d_0 \sin \omega_0(t) \qquad \text{(sinusoid)} \qquad (4\text{-}10b)$$

The constant disturbance is included in order to evaluate the tracker's performance even in the face of unmodelled constant disturbances. The sinusoidal errors are included to see how the system responds to dynamic disturbances such as target jinking.

The second robustness study in this thesis is carried out after the new target acceleration model is included in the system. However, the study is dramatically different from that in the previous section on sensitivity analyses. Here we are changing target parameters in the truth model without telling the Kalman filter. In other words, the target may be vastly different from that which is assumed in the design. If we were to pull out the Meer and Kalman filters and say that the controller has complete access to the true states of the system, the system would be inherently robust. However, this quality is lost when the true states are replaced by their best estimates as supplied by the filters. To recover this robustness quality, Doyle and Stein (Ref. 16) propose the addition of white Gaussian noise at the control input points of the system. This Loop Transmission Recovery (LTR) technique is actually derived for continuous-measurement LQG problems; however, we apply it in an ad hoc manner to this problem involving space-time point process measurements. Whereas before the strength of the white Gaussian noise for the beam was Q, it is now

$$Q' = Q + q\underline{B}\ V\ \underline{B}^T \tag{4-11}$$

where V is the noise strength of the white Gaussian noise inserted at the control input points; here we use V = 1. $\underline{B}$ is the input matrix, and q is scalar. According to Doyle and Stein, as q goes to infinity, the full state feedback robustness properties are recovered, while the performance around the nominal conditions is degraded, because of the resultant mistuning of the filter for design conditions. To see how this technique is used on higher ordered systems, see Maybeck, Miller, and

Howey's paper (Ref. 6) on the robustness enhancement of their flight controller design as well as the original LTR references (Ref. 16). In general terms, the inclusion of the added noise of Equation (4-11) is telling the filter to place more weighting on the measurements and less on its internal model. Note, since the only input to the system is through the scalar beam dynamics equation, it appears as though we are simply adding more process noise to the system. However, this is rarely the case with an LQG/LTR design, and then the specific structure of adding noise at the control input points (i.e., changing $\underline{Q}$ by a scalar times $\underline{B}V\underline{B}^T$ rather than just arbitrarily increasing $\underline{Q}$) becomes important to robustness recovery.

IV.5 SUMMARY

The purpose of this chapter was to describe some of the tools and analysis techniques to be used in this thesis. Monte Carlo runs are performed on a version of software known as SOFE while a program named SOFEPL is used to compute the RMS errors. Two different types of analysis were performed during this work. First, a sensitivity analysis was run on the corrected PI controller originally designed by Moose. Second, robustness studies were performed on the corrected PI controller as well as the controller incorporating a target acceleration model. Unmodelled disturbances were placed in the former while truth model parameter variations were performed during the latter. LTR techniques were applied in an effort to recover much of the full state feedback robustness. The results of these studies are shown in the following chapter.

V. RESULTS

Results of the analysis techniques presented in Chapter IV will now be examined. First, results using the controller based on a first-order Gauss-Markov position model are studied. This is the corrected version of Moose's controller (Ref. 10). Included in this presentation will be a sensitivity analysis as well as a robustness study where disturbances are added to the truth model without telling the filter. Second, results using the controller based on a first-order Gauss-Markov acceleration model will be investigated. This robustness study involves changes in certain parameters within the truth model without telling the filter. Loop Transmission Recovery (LTR) techniques are used to recover some of the robustness characteristics which are lost when true states are replaced by their best estimates in the feedback loop.

At the beginning of each of the two major sections, the choice of proper weighting matrices is discussed. There are major insights to be gained by varying the weighting elements and seeing how the tracker reacts. To view this, plots are given to show target and beam position versus time. These are the only plots in this chapter where the actual run is shown. In all the other figures, RMS errors are displayed versus the particular parameter being varied. It is important to realize that each point on each graph represents an ensemble average of 200 runs, and then a subsequent temporal averaging over the time period of interest (50 seconds $\leq t \leq$ 100 seconds).

V.1 FIRST ORDER GAUSS-MARKOV POSITION MODEL

V.1.1 CHOICE OF WEIGHTING MATRICES. To examine the choice of weighting factors, we must first write out the cost function for a PI Tracker which incorporates the first order position model for the target.

$$J = \frac{1}{2}\sum_{i=1}^{N}\left\{\begin{bmatrix} x(t_i) \\ x_e(t_i) \\ q(t_i) \end{bmatrix}^T \begin{bmatrix} X_{11} & -X_{11} & \emptyset \\ -X_{11} & X_{11} & \emptyset \\ \emptyset & \emptyset & X_{33} \end{bmatrix} \begin{bmatrix} x(t_i) \\ x_p(t_i) \\ q(t_i) \end{bmatrix} + Uu^2(t_i)\right\}$$

$$+ \frac{1}{2}\begin{bmatrix} x(t_{N+1}) \\ x_p(t_{N+1}) \\ q(t_{N+1}) \end{bmatrix}^T \begin{bmatrix} X_{11f} & -X_{11f} & \emptyset \\ -X_{11f} & X_{11f} & \emptyset \\ \emptyset & \emptyset & X_{33f} \end{bmatrix} \begin{bmatrix} x(t_{N+1}) \\ x_p(t_{N+1}) \\ q(t_{N+1}) \end{bmatrix} \qquad (5\text{-}1)$$

Note that in the constant controller gain case, there are an infinite number of terms in the summation and the terminal cost is omitted. The X_{11} terms weight the position error between the beam and target at time t_i while the X_{33} is used to weight the integration value (i.e., the history of errors). If X_{33} is larger, this is saying that the controller should place greater emphasis on minimizing the integral of past errors. Therefore, the best response to a large positive error at time t_i is often a large negative error at time t_{i+1}. This causes a very oscillatory response, as shown in Figure 5-1 (note the large initial overshoot). The oscillation provides speed to the response; however, the ringing errors are unacceptable. To taper the response while maintaining its under-damped (oscillatory) nature, X_{11} is increased. The result is shown in Figure 5-2. The initial overshoot has been decreased considerably.

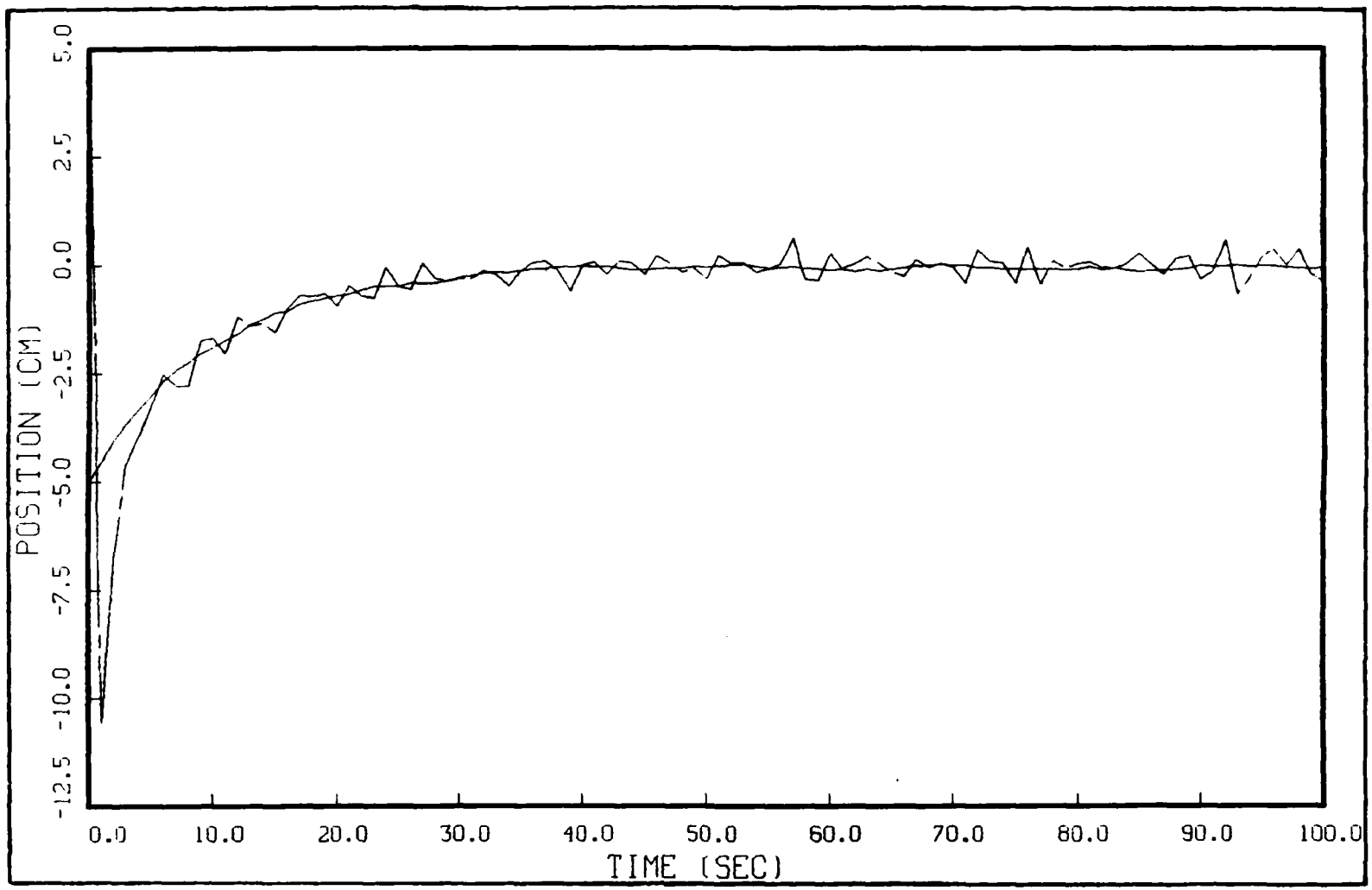


Figure 5-1. Controller Response with Heavy Weighting on Integral Term

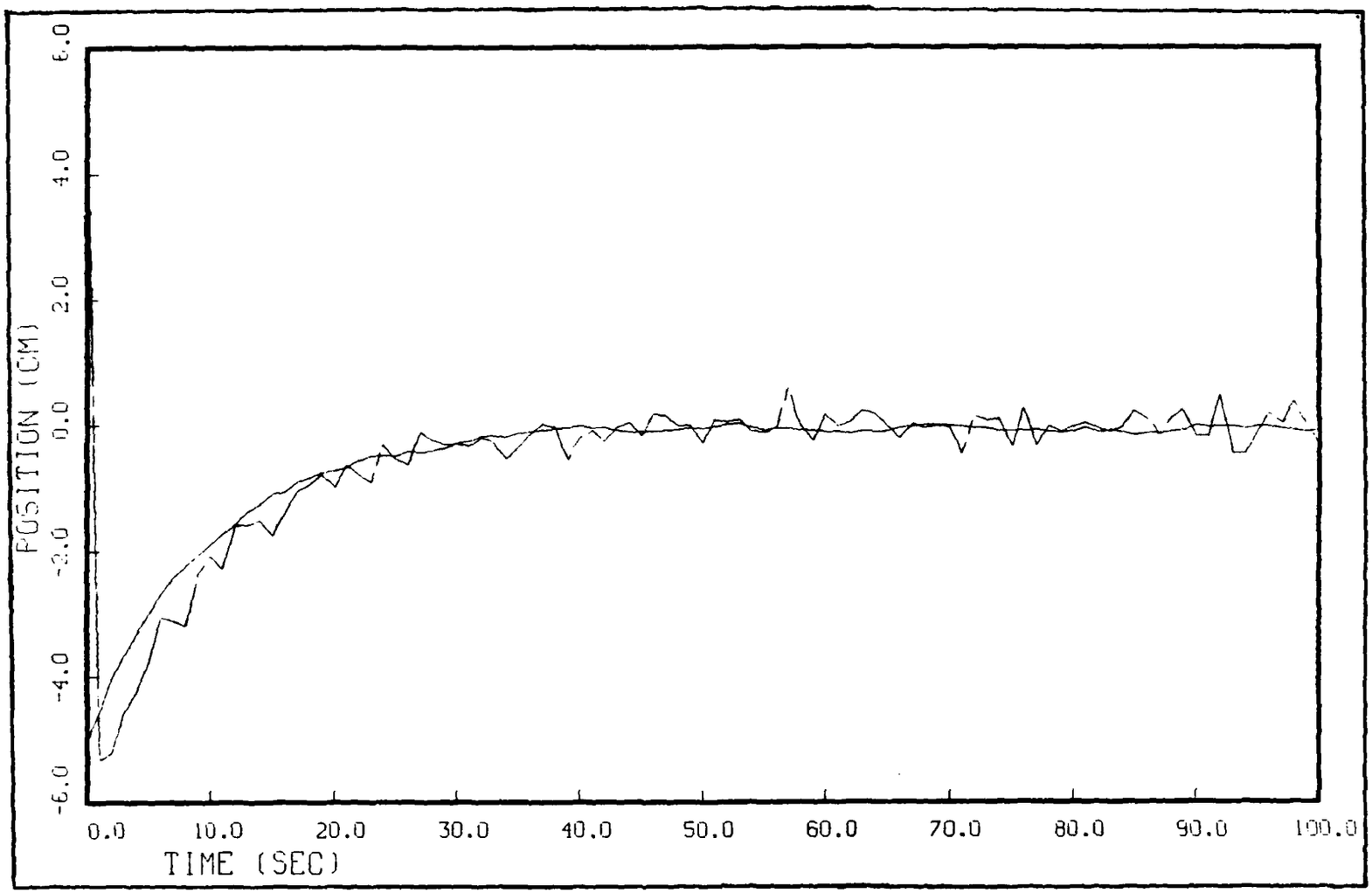


Figure 5-2. Controller Response with Proper Weighting of Integral and Error Terms

Ideally the X_{11} term should be on the order of at least a factor of 10 above X_{33} to provide the desired nature of the response. The actual values chosen are $X_{11} = 100$ and $X_{33} = 1$.

V.1.2 PARAMETER SENSITIVITY. A parameter sensitivity analysis is carried out on the PI controller described above. Only the tracker parameters are varied because the parameter sensitivity of the regulator was carried out correctly in Moose's thesis (Ref. 10). The parameters to be changed are Q_T, R, and τ of the target truth and filter models. These parameters are explained in Equations (3-19) and (3-20). They are varied one at a time while the others are held at their nominal values. In this manner, we see how the system performance is affected by the parameters of the system.

SENSITIVITY TO Q_T. The parameter Q_T is an indication of how confident we are of the target model being correct. As Q_T is increased, the target's position becomes more dynamic due to the fact that the strength of the white noise driving term is increasing. Therefore, we expect the tracker RMS tracking error to increase as Q_T is increased. This trend can be seen in Figure 5-3. As the driving noise is increased from .01 to 1 we see that the beam is having more trouble tracking the increasingly dynamic target. In the figure, "state feedback" refers to the fact that the controller for those simulations is artificially assumed to have complete access to the true states of the system, whereas "filter feedback" means that these true states have been replaced by their best estimates as provided by the Meer and Kalman filters. This notation will be used throughout the analysis. The RMS errors are considerably higher

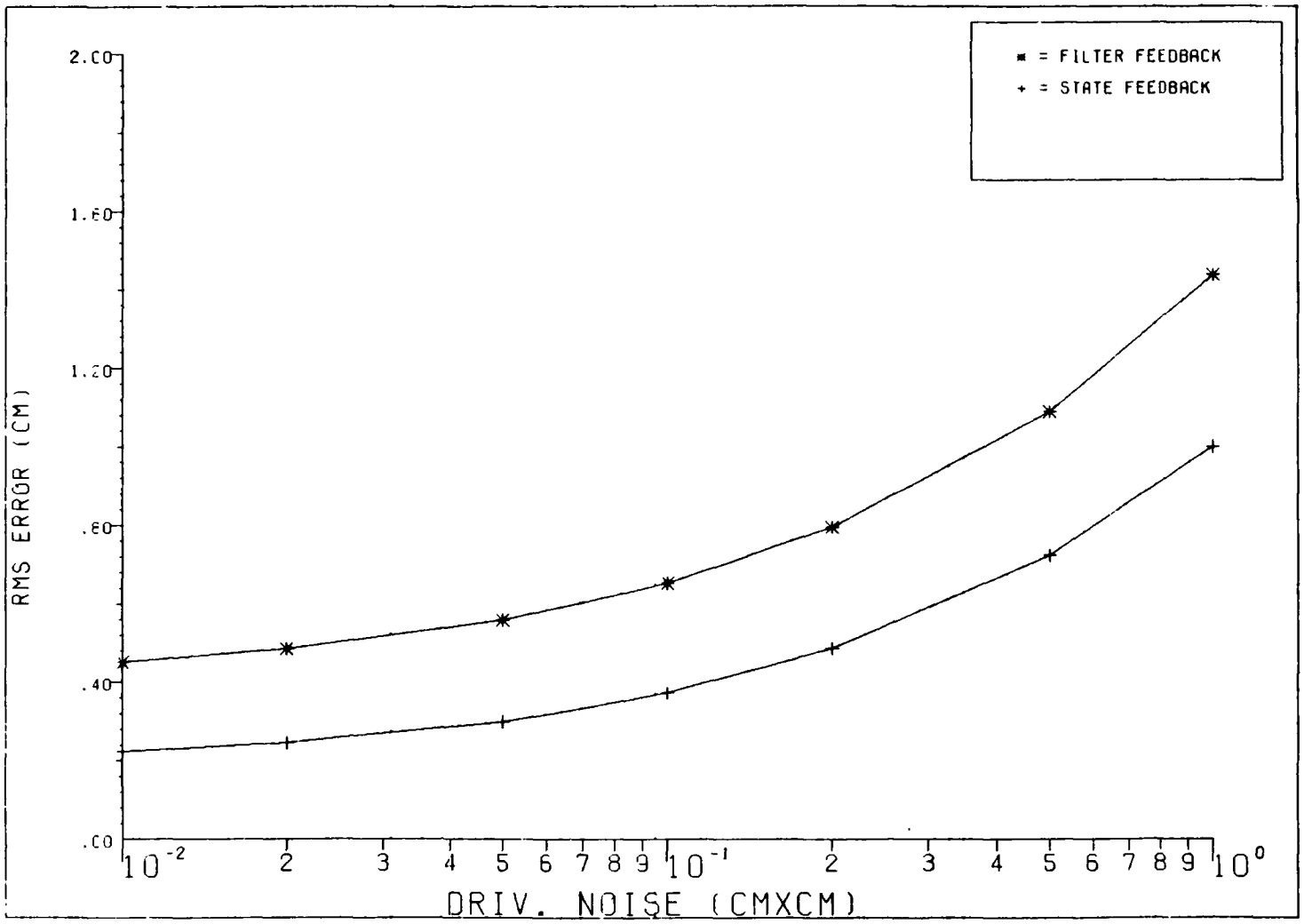


Figure 5-3. Sensitivity Analysis: Tracker RMS Errors vs Target Model Driving Noise

for filter feedback because performance is always degraded when state estimates are used rather than the true values.

SENSITIVITY TO R. The parameter R is an indication of the level of noise corruption in the target measurement model. As R increases, our confidence in the accuracy of measurements decreases. If the true states are fed back to the controller, variations in R have no effect because the Kalman filter is out of the control loop. However, if filtered states are provided to the controller, we expect an increasing R to yield increasing RMS errors for the tracker. By turning to Figure 5-4, we see that this is indeed the case. The state feedback controller is unaffected by changes in R when larger values of R are used.

SENSITIVITY TO τ. The parameter τ is an indication of how fast the target's velocity can be changed. If τ is increased, there is more persistent motion once it is established. If τ is decreased, velocity can change rapidly. To see how the parameter τ affects RMS errors, turn to the RMS position equation for the target:

$$(\text{RMS position})^2 = \frac{\tau Q_T}{2} \tag{5-2}$$

Again we see that RMS error is directly linked to the time constant τ. By examining Figure 5-5, we see that this logic is correct, and once again state feedback outperforms filter feedback.

In this diagram, τ is varied from 5 to 50 seconds. At $\tau = 5$, it is necessary to reduce the measurement sampling period and control input intervals to 0.5 seconds in order to have the sampling rate a factor of 10 larger than the fastest transient in the system. All other points

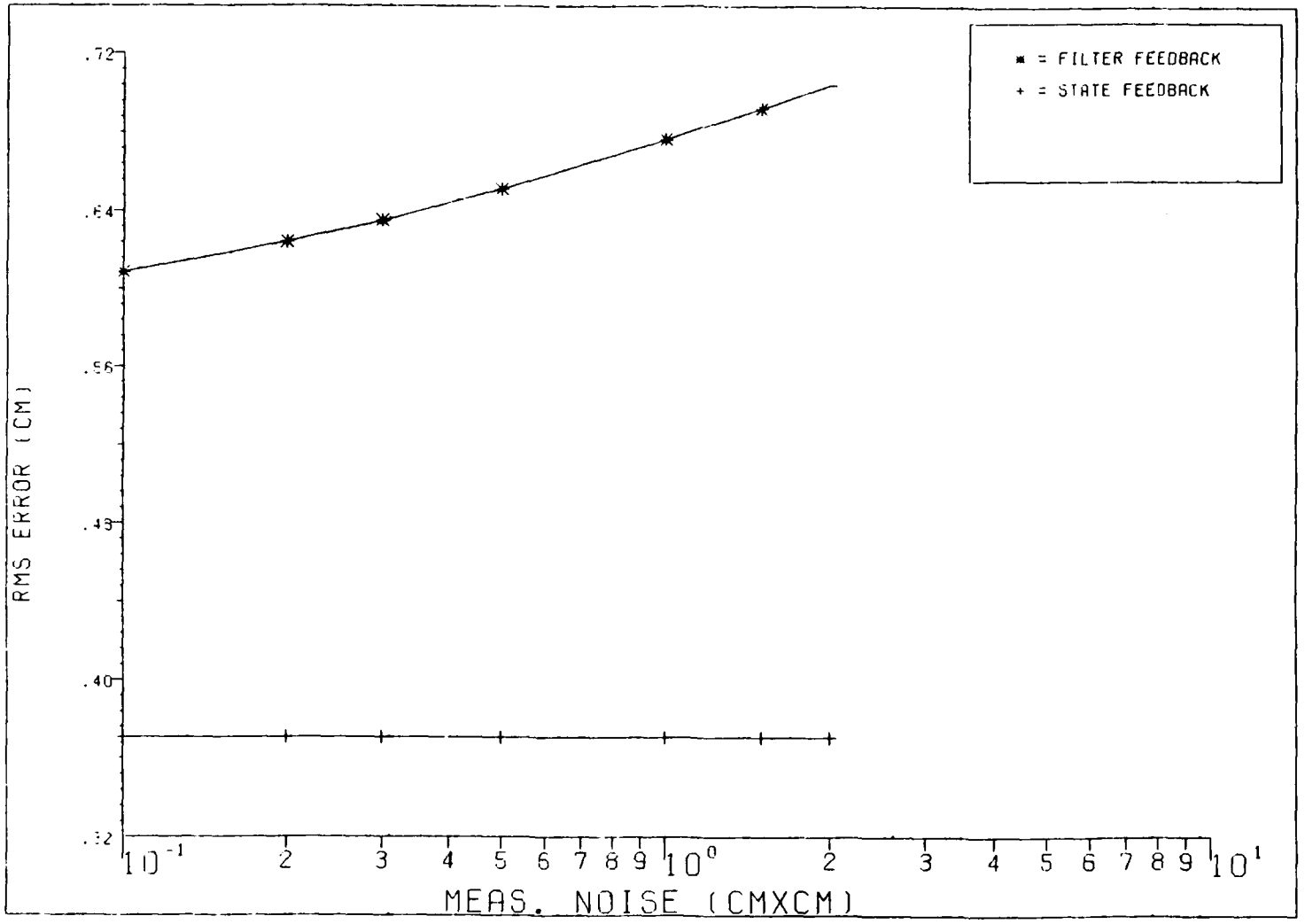


Figure 5-4. Sensitivity Analysis: Tracker RMS Errors vs Target Measurement Noise

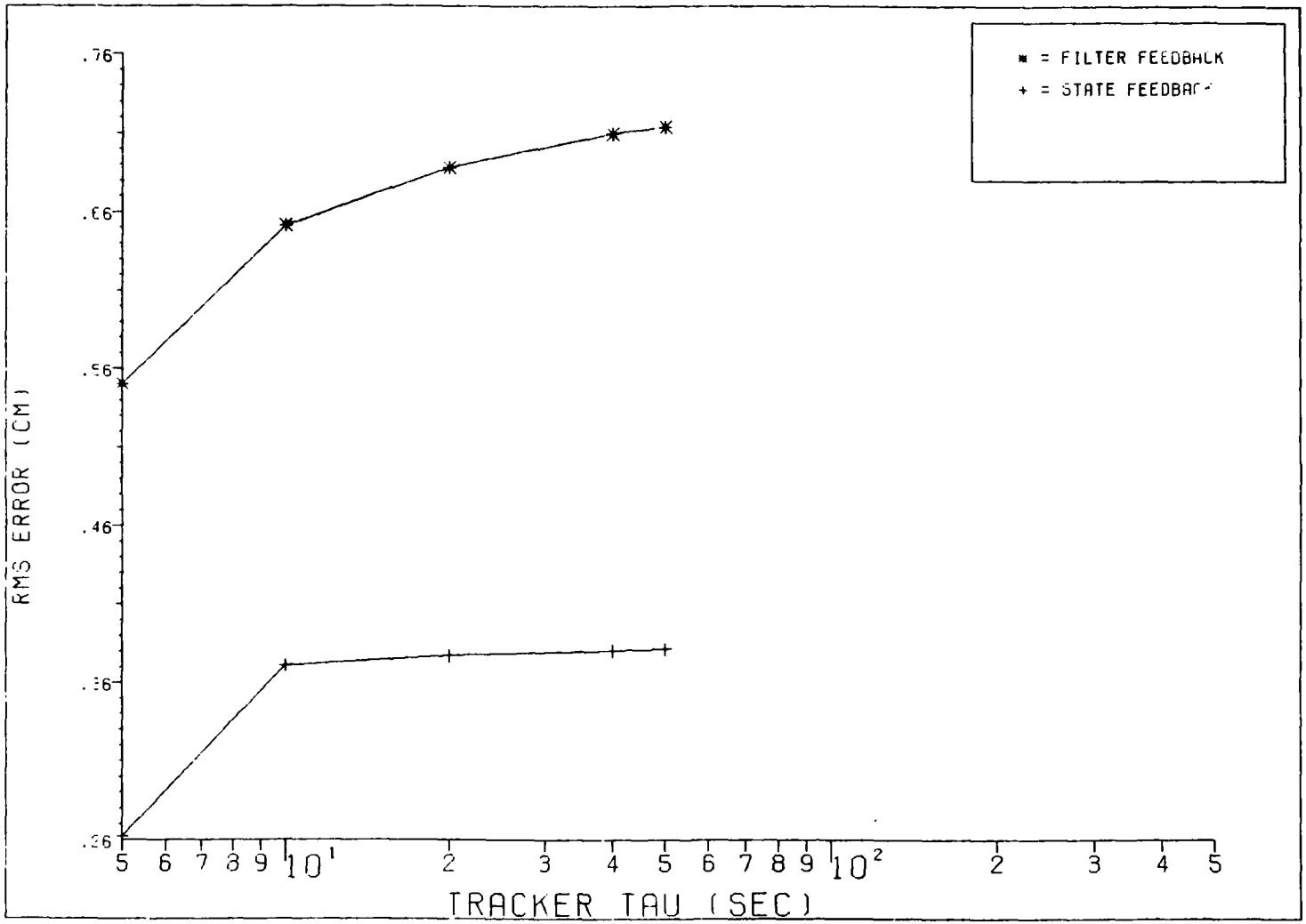


Figure 5-5. Sensitivity Analysis: Tracker RMS Errors vs Target Position Time Constant

used sampling rates of 1 second. This is the cause of the dramatic change in the figure below $\tau = 10$.

V.1.3 ROBUSTNESS TO DISTURBANCE INPUTS. An important question to ask when designing the filter/controller is how the system will hold up against variations in the truth model which are not known to the filter. One such variation is the addition of disturbances into the truth model beam equation. Part of the justification for going to the PI controller configuration is so that unmodelled constant disturbances to the truth model as well as the addition of sinusoidal disturbances will be addressed in this subsection.

CONSTANT DISTURBANCES. When constant disturbances are added to the system, we should see the controller at least attempting to maintain track with zero steady state error. This is due to the "type 1" characteristics brought about by integral action (see Chapter III). The PI controller is run with several values for d_0 and the results are shown in Figure 5-6. By observing the lower plot in the figure we note that as d_0 is varied between 0.0 and 4.0, the tracker full-state feedback RMS error increases by only 2.4 cm. Therefore, we see that the constant disturbances are not having a severe impact on the RMS error. The result is very different with the filters in the loop. With this constant disturbance affecting the true states of the beam, the filter begins putting out erroneous state estimates as well as erroneous control inputs based on their estimates. Therefore, the filter feedback performance is degraded much more than what we see for state feedback.

SINUSOIDAL DISTURBANCES. Another interesting case to consider, from the standpoint of the designer, is the addition of dynamic inputs

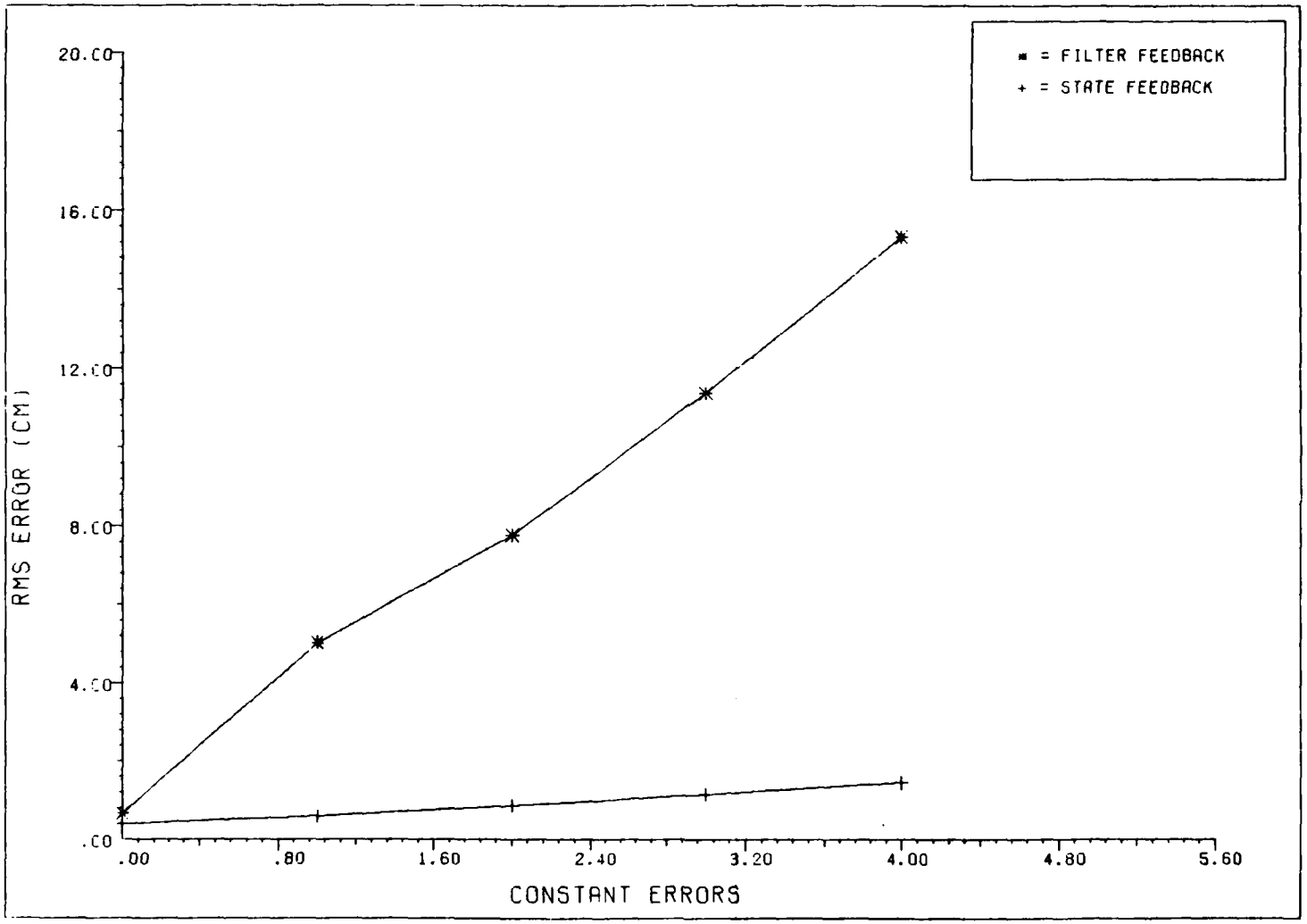


Figure 5-6. Robustness Analysis: Tracker RMS Error vs Constant Unmodelled Disturbance Magnitude

into the truth model system. These unmodelled inputs are assumed to take the form of Equation (5-3):

$$d(t) = d_0 \sin \omega_0 t \tag{5-3}$$

where d_0 is the maximum amplitude of the disturbance and ω_0 is the frequency in rad/sec. This might represent unmodelled target jinking for instance. No longer do we have insights to say beforehand what the results should look like. In Figure 5-7 we see the effects of letting d_0 be 1 cm and varying ω_0 from 0 rad/sec to 5.0 rad/sec. At the lower frequencies, the RMS errors are not increased dramatically for the full state feedback case. However, when filter feedback is used, performance is degraded severely (at low ω_0). The reason is that the persistence of a slowly varying nonzero disturbance is inducing poor state estimates in the filter. Without accurate state estimates, we cannot hope to track the target. It is suspected that LTR tuning might improve performance at the lower frequencies; however, there was not enough time to do this here (see Chapter VI). As frequencies increase, so does the performance of the controller. This is due to the fact that at higher frequencies, the disturbances are not as persistent and the filter can achieve better estimates. The state feedback case always outperforms the filter feedback case by a small amount over this region. During Moose's thesis, filter feedback actually outperformed the state feedback case; the reasoning was a poorly tuned controller. Obviously, this error has been corrected. At the upper end of the frequency scale we note that the RMS errors begin an upswing again, for both the full-state feedback and filter-in-the-loop cases. This may be because the fastest transient in

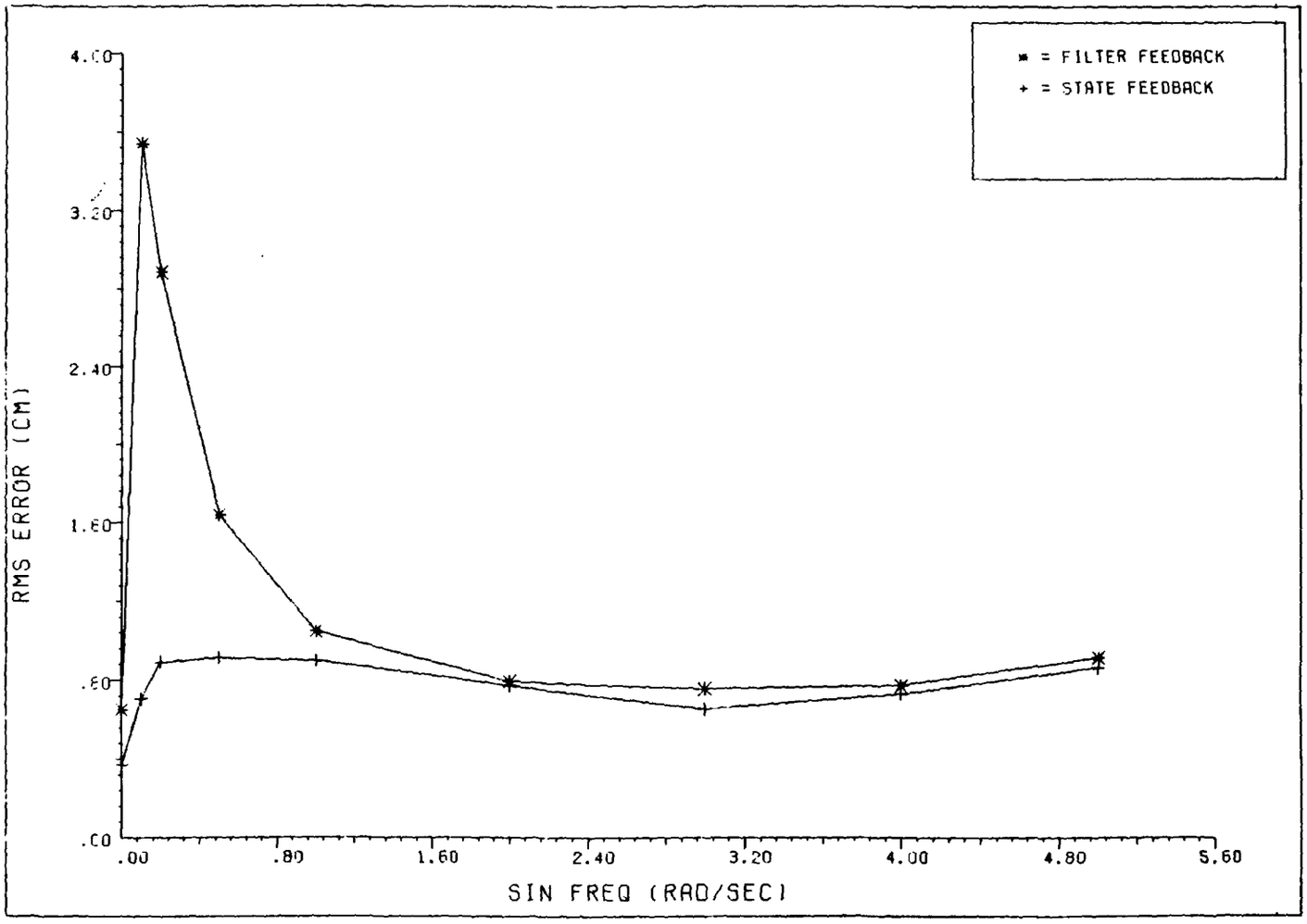


Figure 5-7. Robustness Analysis: Tracker RMS Error vs Frequency of Unmodelled Sinusoidal Disturbances

our system is increasing, but the sampling periods have remained at 1 second. Therefore, aliasing effects are beginning.

In Figure 5-8 we see the effects of maintaining ω_0 at 1 rad/sec and varying d_0 from 0.0 to 5.00 cm. As we expect, the RMS errors tend to increase as the magnitude of the disturbance input increases. During a small region the filter seems to be outperforming the state feedback case. This trend is small and short-lived, so that no conclusive statement may be made. In any event, the difference between state feedback and filter feedback cases is not significant.

V.1.4 GENERAL COMMENTS. The purpose of using the first order Gauss-Markov position model for the target dynamics is to go back and check if Moose's errors in his cost function (see Chapter 1) severely impacted his PI controller performance. In all cases there was little difference between this controller and his. Any improvements might be attributed to the proper choice of weighting matrices. Therefore, it must be concluded that the inclusion of the off-diagonal terms in Equation (5-1) does not impact this design significantly.

V.2 FIRST ORDER GAUSS-MARKOV ACCELERATION MODEL

V.2.1 CHOICE OF WEIGHTING MATRICES. To begin making proper choices for cost function weighting matrices, we start by observing the new cost function, which is displayed in Equation (3-26). Once again we have two weighting terms X_{11} and X_{55} which weight the position error and pseudointegral state, respectively. The method of choosing these two is similar to that of the previous section and will not be repeated here. For the new controller, X_{55} (the pseudointegral weighting) affects the speed and oscillatory nature of the response while the X_{11} term

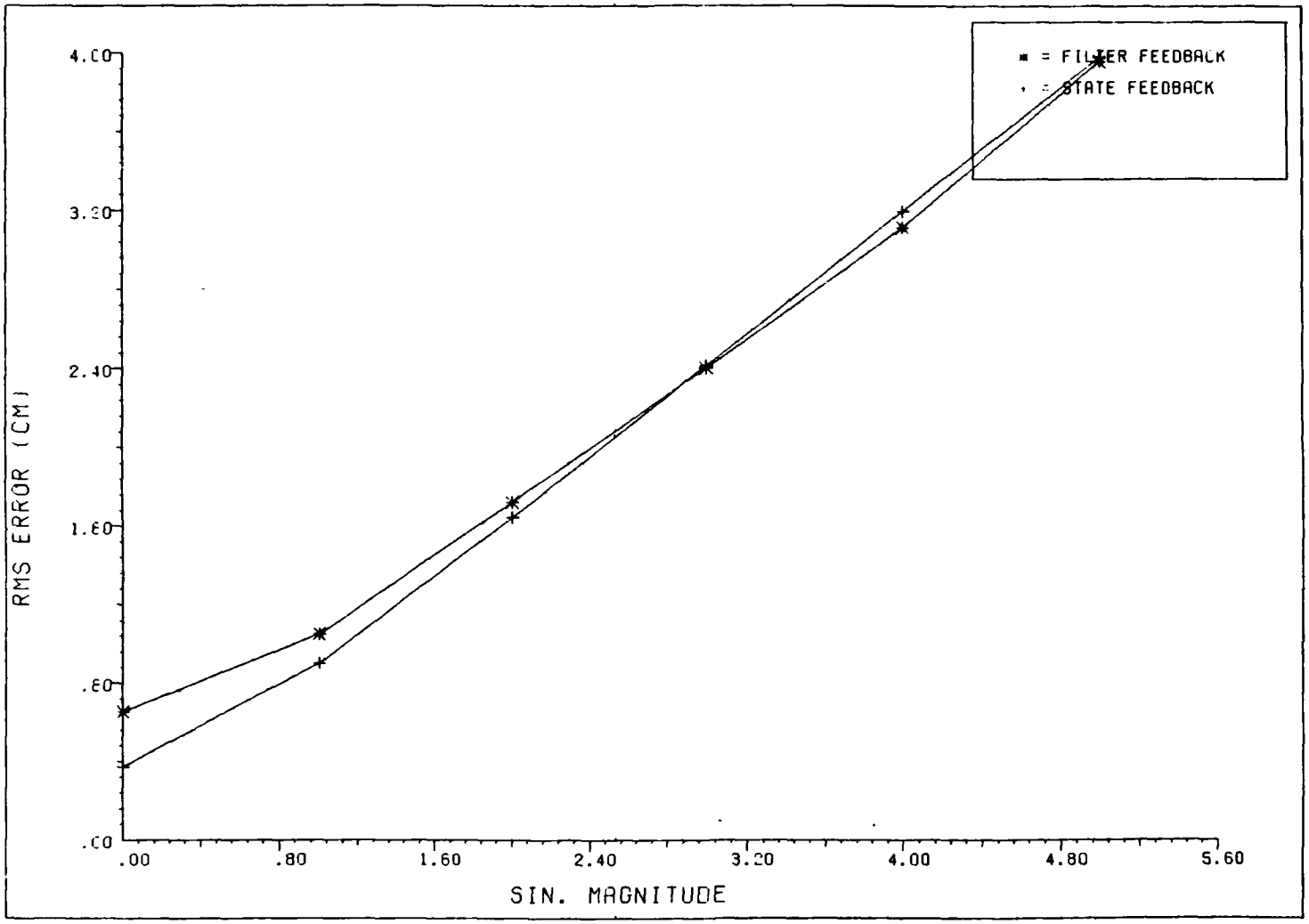


Figure 5-8. Robustness Analysis: Tracker RMS Error vs Magnitude of Unmodelled Sinusoidal Disturbances

(error weighting) provides increased damping and tighter control of error amplitudes. It is a much more difficult job to choose these terms as opposed to the previous problem because only a few choices provide a stable response. The final choice of $X_{11} = X_{55} = 100$ gives the best results. If X_{55} goes below 100, the beam cannot react fast enough to the accelerating target. If X_{11} is increased dramatically, instabilities are noted.

At one point, addition of weights on the target velocity and acceleration states was attempted, with surprising results. By weighting these states over which we have no control, in an ad hoc manner, the beam was able to track the target with reasonable RMS errors. By incorporating target velocity and acceleration terms into the cost weightings, it seems possible that the controller will take into account the anticipated future location of the target.

V.2.2 ROBUSTNESS TO PARAMETER VARIATIONS. Once the new PI controller is formulated, simulations on SOFE and SOFEPL are desired. In the interest of time, a parameter sensitivity analysis for the new model is not conducted and attention is instead focused on the robustness analysis. It is desired to see if the PI controller can maintain its performance characteristics even when the true target is different from that which is modelled in the filter. To accomplish this end, the parameters τ, Q_T, and R of the target as seen in Equations (3-19) and (3-20) are varied within the truth model without telling the filter. In order to attempt recovery of full state feedback robustness qualities, the LTR technique is employed as discussed in Chapter IV. In all cases, the performance bound of full-state feedback is shown to see the impact

of the filters being in the control loop. Full state feedback indicates the best performance possible of the PI controller.

VARIATIONS IN Q_T. In this study the target model noise strength Q_T is varied from .01 to 100 while the filter value remains at 0.1. The strength of the beam model noise embedded in the truth model is kept constant at 0.04. We begin the analysis by setting the Meer filter noise strength to 0.04 and varying the target Q_T. As can be seen in Figure 5-9, the performance severely degrades as the result of increased target process noise. This arises because, as this noise strength is added, the target is becoming more dynamic.

In order to recover some robustness, white noise is injected at the control input of the beam model, upon which the Meer filter is based. The noise inputs of the Meer filter system now appear as:

$$Q' = Q + qBVB^T \tag{5-4}$$

Q is 0.04, B is 1, V is the strength of the white Gaussian noise inserted at the control input points (we use V = I), and q is the scalar. For this simple problem, it appears as though we are merely increasing the Q of the beam. Q is raised to a value of 400 with little effect. A table of RMS errors both with and without LTR tuning is shown in Table 5-1. The new points are also displayed in Figure 5-9. We see that there is only a slight improvement in the higher RMS region while performance about the nominal is degraded. This is the correct trend; however, when the noise of the filter is increased by 4 orders of magnitude, we would expect more significant changes.

TABLE 5-1

Average RMS Tracking Errors with Variations in Q_T of Target

	State Feedback	Filters in the Loop Beam Driving Noise Strength	
		0.04	400
Q = 0.01	3.13787	8.06478	8.32884
0.05	6.01577	9.64186	9.84316
0.10	8.32847	11.3221	11.4819
0.20	11.6612	14.1039	14.2145
0.50	18.3455	20.2948	20.3399
1.00	25.9156	27.7070	27.7110
5.00	57.9502	60.1843	60.0917
10.00	81.9751	84.8745	84.6816

TABLE 5-2

Average RMS Tracking Errors with Variations in R of Target

	State Feedback	Filters in the Loop Beam Driving Noise Strength	
		0.04	400
R = 0.1	8.32847	9.30874	9.53589
0.2	"	9.85120	10.0568
0.3	"	10.3649	10.5506
0.5	"	11.3221	11.4819
1.0	"	13.4188	13.5369
2.0	"	16.8461	16.9173
5.0	"	24.3955	24.4087
10.0	"	33.3840	33.3540
30.0	"	56.5089	56.4130

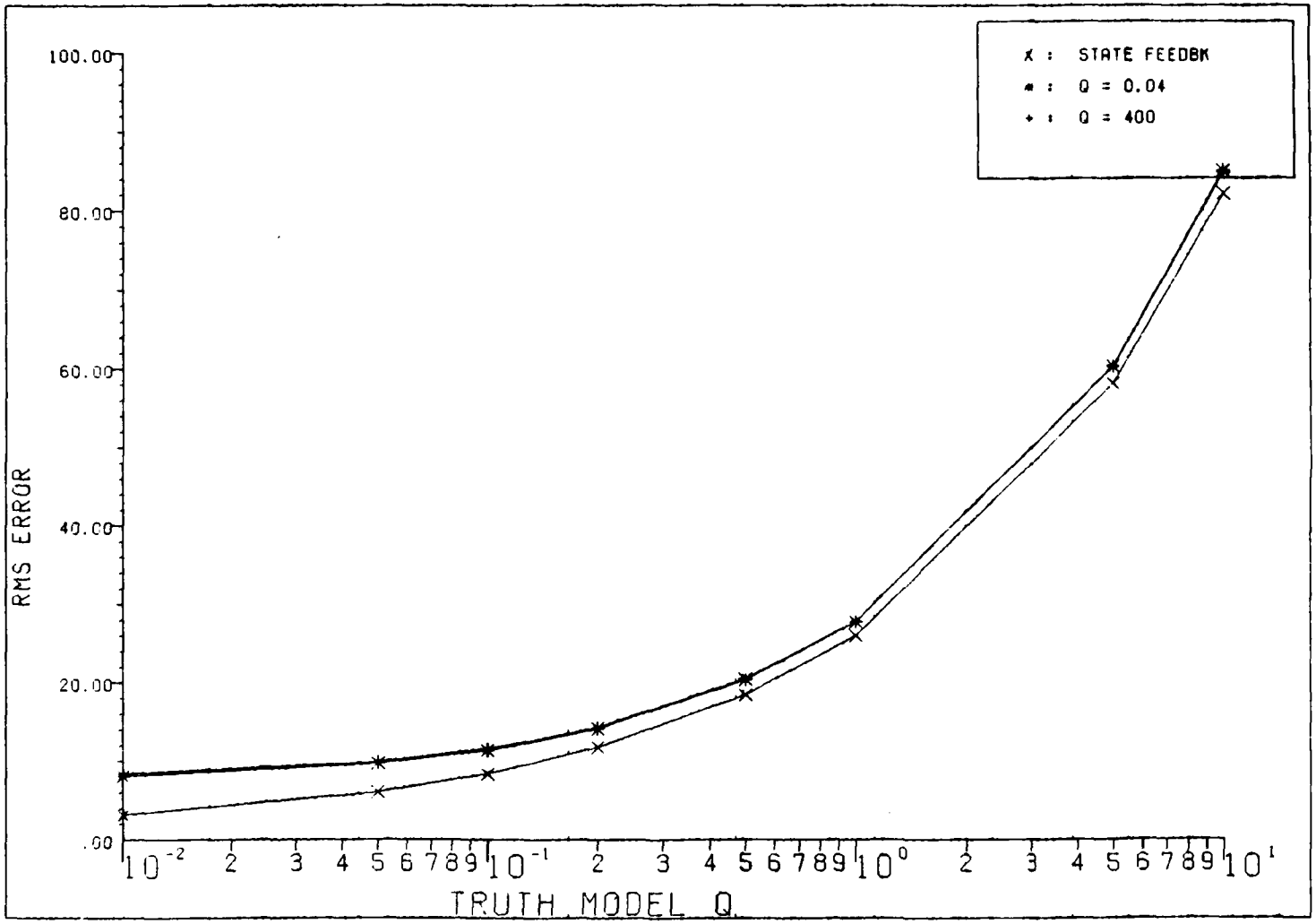


Figure 5-9. Robustness Analysis: Tracker RMS Error vs True Target Model Driving Noise

One explanation for the lack of robustness enhancement by LTR tuning is that the Meer filter is already confident of the detector's ability to identify events (the beam dispersion is 0.5 cm on a detector of length 10 cm). Therefore, when the extra noises are added, the Meer filter is unable to apply much more weighting to incoming measurements. Another way to explain the lack of effectiveness is to compare the RMS errors for true state feedback with those of filtered feedback. Remembering that true state feedback is the best that can be achieved, we see that the filtered case is not that more than the full state case to begin with, particularly for high values of Q_T. Consequently, LTR tuning cannot impact the performance significantly.

VARIATIONS IN R. In the study R is varied from 0.1 to 100 while the filter's nominal remains at 0.5. We expect the RMS errors to increase as our true measurements become more and more noisy while the filter R remains the same. Figure 5-10 shows this trend. Again runs are completed for Meer filter noise strengths of 0.04 and 400. The results are shown in Table 5-2. There is not a significant improvement at the higher RMS range, but the values around the nominal are being degraded. This is the proper trend, but the ineffectiveness of the LTR technique is again expected due to the high confidence already placed in the beam detector. The extreme degradation in performance when filters are in the loop suggests that further work is needed. Either better choices of R for the filter or adaptive estimation of R might be attempted (see Chapter VI). The errors are constant for the state feedback case because the Kalman filter has been removed from the control loop.

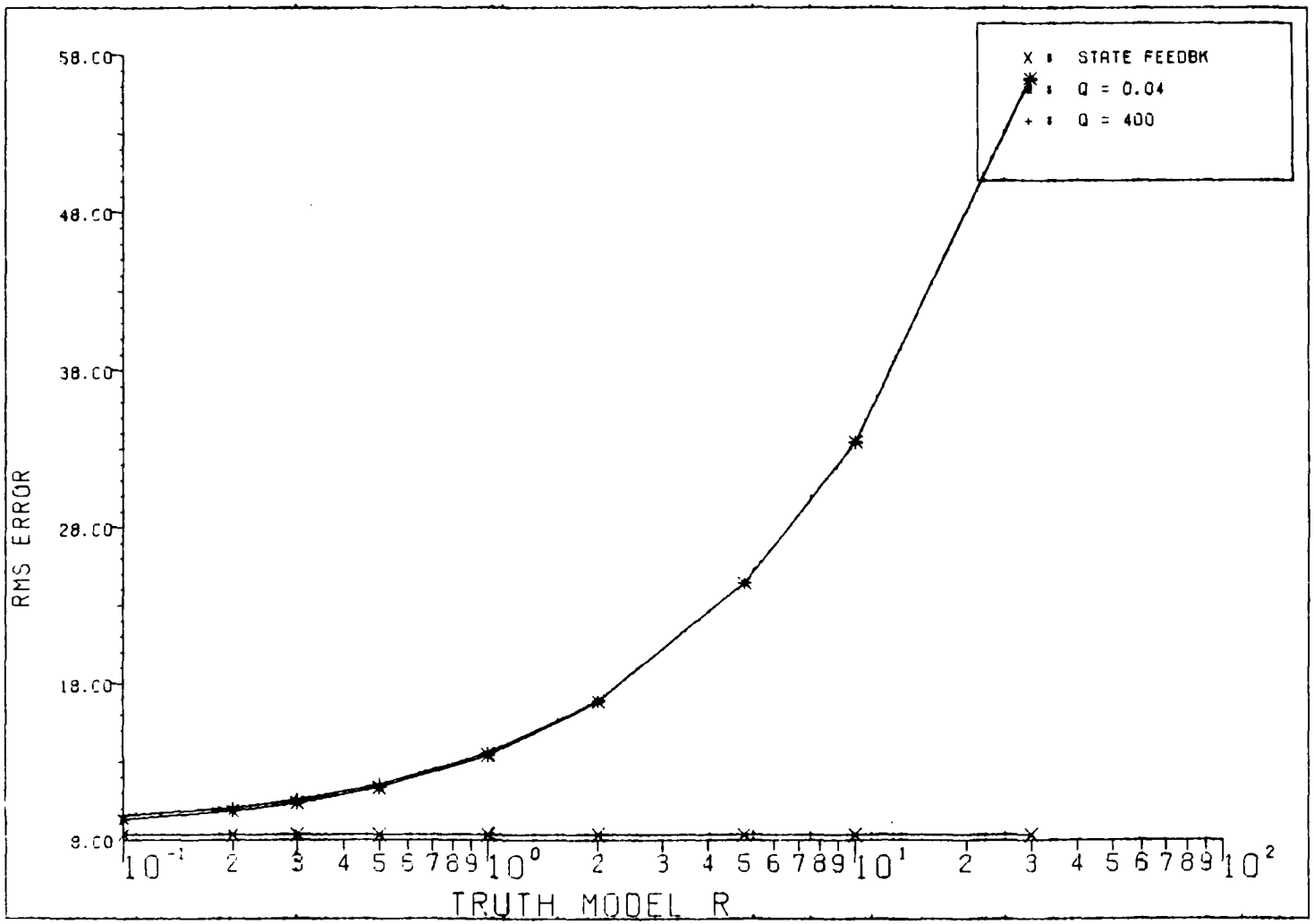


Figure 5-10. Robustness Analysis: Tracker RMS Error vs True Target Model Measurement Noise

VARIATIONS IN τ. There are three separate approaches to the problem of varying τ in the target truth model. The first approach simply involves varying τ between 1 and 1000 and observing the effects on RMS errors, for the two cases of assumed noise strength in the beam dynamics model of the Meer filter (0.04 and 400). As shown in Figure 5-11, the RMS errors increase as τ increases. Remembering the equation for RMS acceleration,

$$(\text{RMS Acceleration})^2 = \frac{Q_T \tau}{2} \tag{5-5}$$

we see that if τ is increased, the RMS errors should indeed increase. Specific values of RMS error are displayed in Table 5-3. There is a basic conceptual problem with this approach; if the target is maneuvering more dramatically (lower τ), we expect the beam to have a more difficult problem in trying to track. This resulted in two other approaches to the means of conducting robustness studies.

Initially, when τ is increased and the target process noise remains constant, we are inherently increasing the RMS acceleration of the target and creating a more difficult tracking problem. On this second cut at the problem, we vary Q_T and τ simultaneously in order to maintain RMS acceleration at a constant value. The results are displayed in Figure 5-12 and Table 5-4. We see that for τ values greater than the nominal, the trend is that slower targets are easier to track. However, we can still see the trend of the previous analysis to the left of the nominal. Also note, the state feedback case shows that there is a great margin for improvement. Therefore, one final attempt was made to view the desired trends.

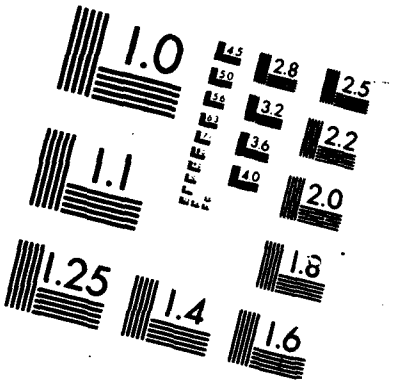
AD-A163 945 PARTICLE BEAM TRACKER FOR AN ACCELERATING TARGET(U) AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOOL OF ENGINEERING L C JAMERSON DEC 85 AFIT/GE/ENG/85D-22 2/2

UNCLASSIFIED F/G 20/7 NL

END

FILMED

DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

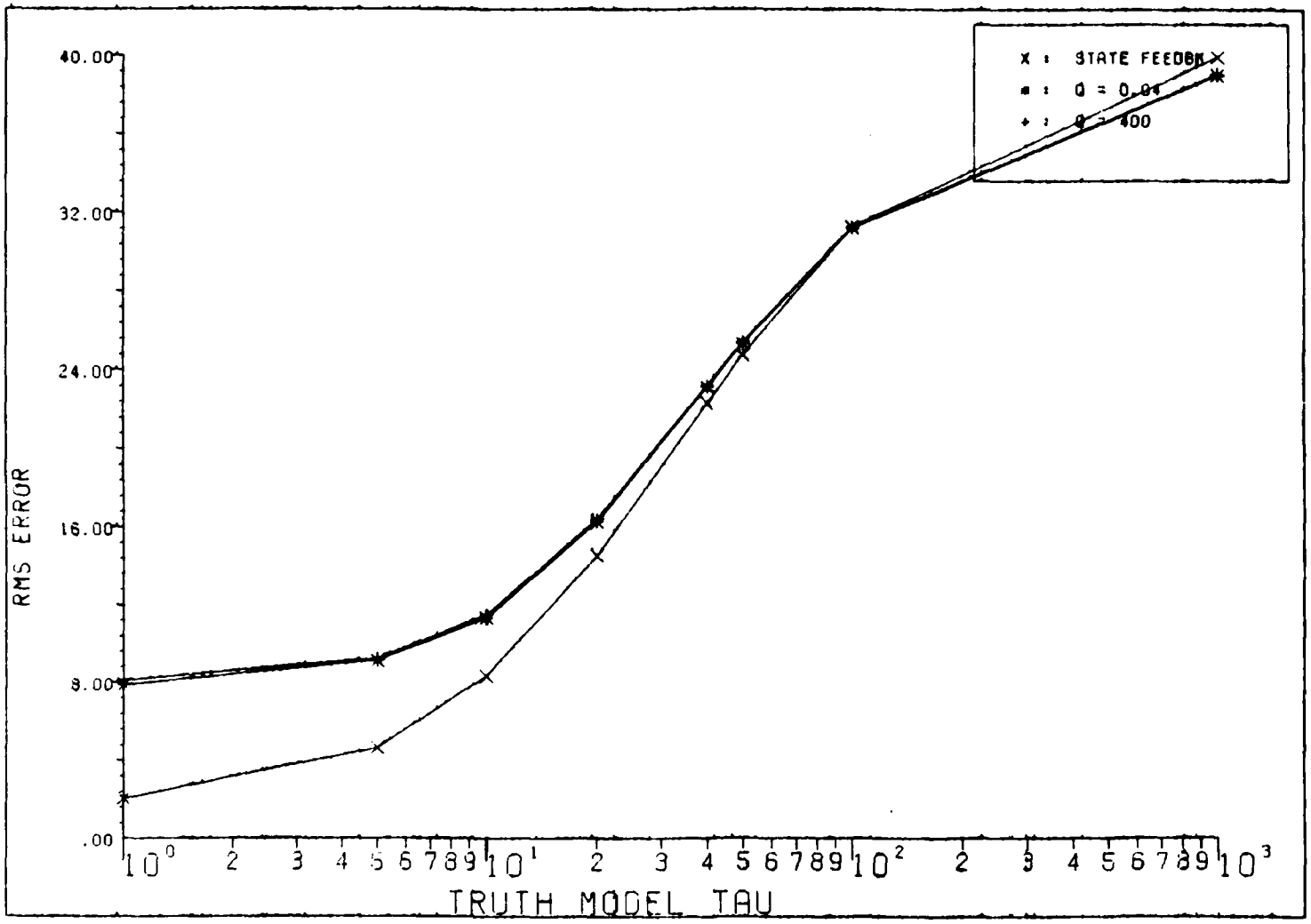


Figure 5-11. Robustness Analysis: Tracker RMS Error vs True Target Time Constant (First)

TABLE 5-3

Average RMS Tracking Errors with Variations in τ of Target (#1)

	State Feedback	Filters in the Loop Beam Driving Noise Strength	
		0.04	400
τ = 1	2.03647	7.86385	8.11772
5	4.62579	9.09537	9.18442
10	8.32847	11.3221	11.4819
20	15.5018	16.2714	16.3765
40	22.2687	23.1029	23.1753
50	24.7370	25.3176	25.3832
100	31.2880	31.2187	31.2707
1000	39.9876	38.9876	39.0280

TABLE 5-4

Average RMS Tracking Errors with Variations in τ of Target (#2)

	State Feedback	Filters in the Loop Beam Driving Noise Strength	
		0.04	400
τ = 1	3.35321	9.52125	9.56537
5	6.29858	10.1702	10.3286
10	8.32847	11.3221	11.4819
20	10.3189	12.6983	12.8485
40	11.2219	13.3181	13.4690
50	11.1577	13.2409	13.3942
100	10.0216	12.2943	12.4663
1000	4.39752	8.57866	8.83193

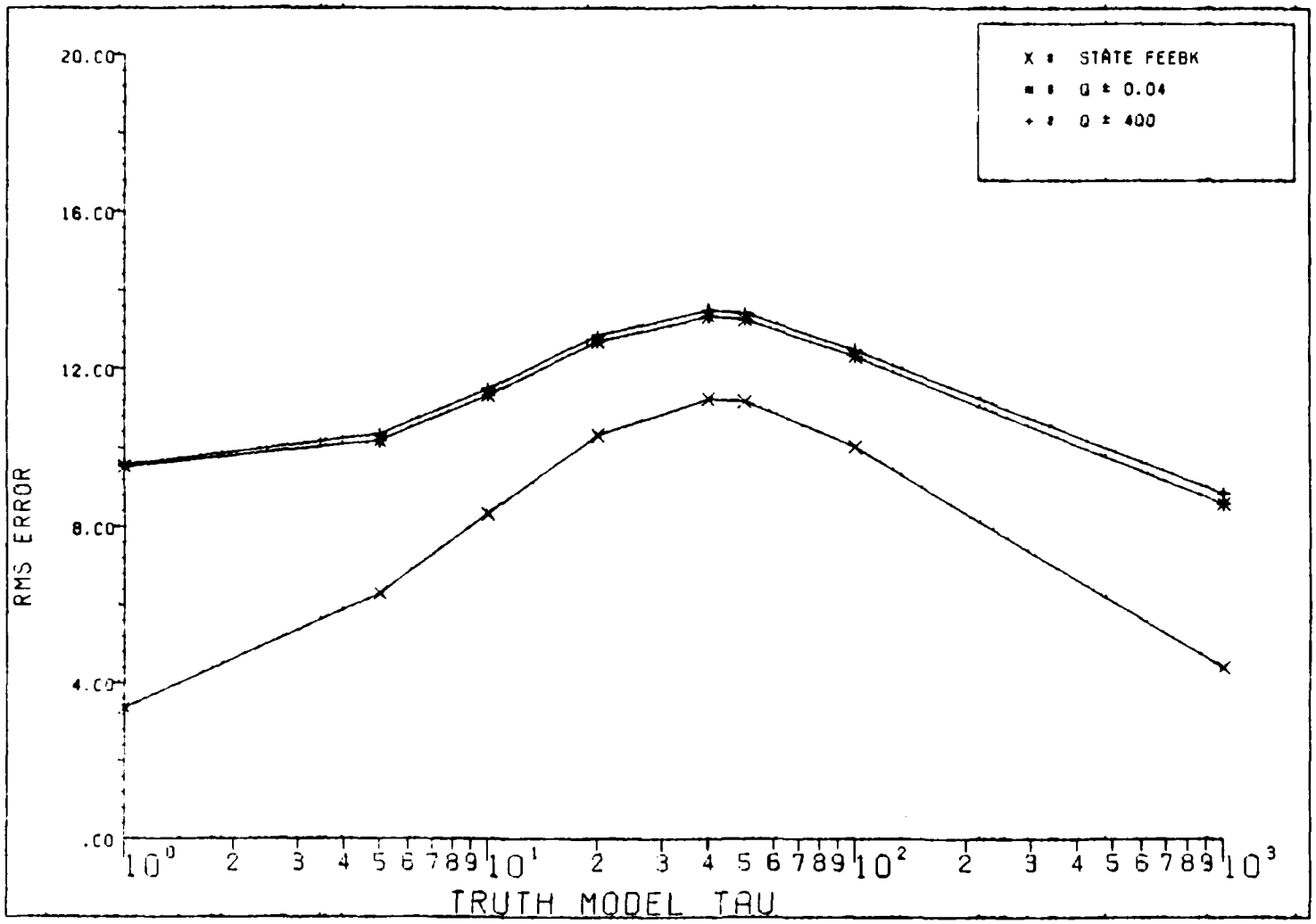


Figure 5-12. Robustness Analysis: Tracker RMS Error vs True Target Time Constant (Second)

TABLE 5-5

Average RMS Tracking Errors with Variations in τ of Target (#3)

	State Feedback	Filters in the Loop Beam Driving Noise Strength	
		0.04	400
τ = 1	9.15374	18.7839	18.6426
5	8.74176	12.2810	12.3202
10	8.32847	11.3221	11.4819
50	5.24540	9.04158	9.28021
100	3.62381	8.22565	8.49233
500	2.00789	7.68029	7.97129
1000	1.89804	7.65600	7.94813

RMS acceleration is an indication of the total area beneath the power spectral density (PSD) curve as shown in Figure 5-13. The maximum value of this curve is $Q_T\tau^2$ (at $\tau = 0$). During this final attempt at the problem, Q_T and τ are varied while the PSD value (or height) is kept constant as bandwidth is changing. The results are displayed in Figure 5-14 and Table 5-5. In these we see that indeed a faster target is more difficult to track and a slower one is easier. Also, the full state feedback case has considerably lower RMS errors than the filtered case.

V.3 SUMMARY

In this chapter we have looked at the results of the analysis techniques described in Chapter IV. First, a sensitivity analysis and a robustness study of disturbance inputs is used to test the corrected PI controller of Moose. The parameter sensitivity studies involve varying Q_T, R, and τ of the target in both the truth and filter models. The robustness study is performed by adding unmodelled constant and sinusoidal disturbances into the truth model without telling the filter. In both cases, improvements to Moose's work are minimal. Therefore, the incorrect off-diagonal terms in the weighting matrix of that previous work do not have a great impact on the system performance. Once a first order Gauss-Markov acceleration model is used as the target, a robustness study is carried out to determine whether changing target parameters in the truth model will impact performance severely. They do; however, LTR techniques are not capable of restoring the robustness to the quality of full state feedback.

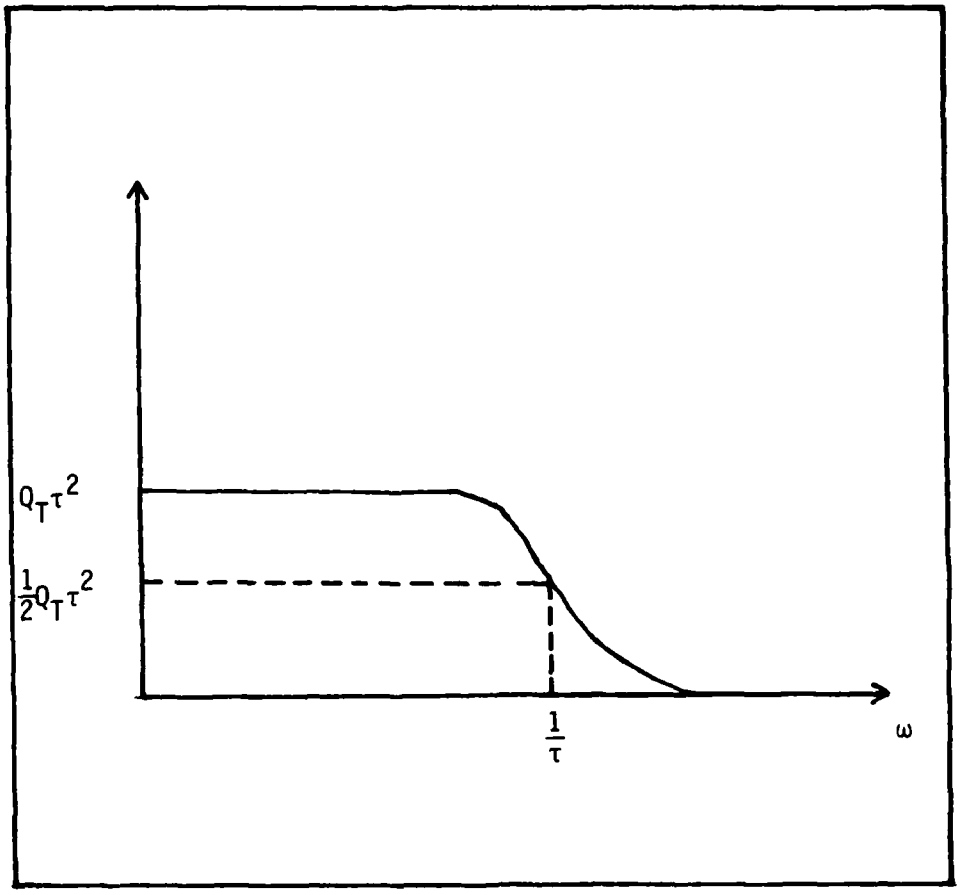


Figure 5-13. PSD Curve and RMS Acceleration

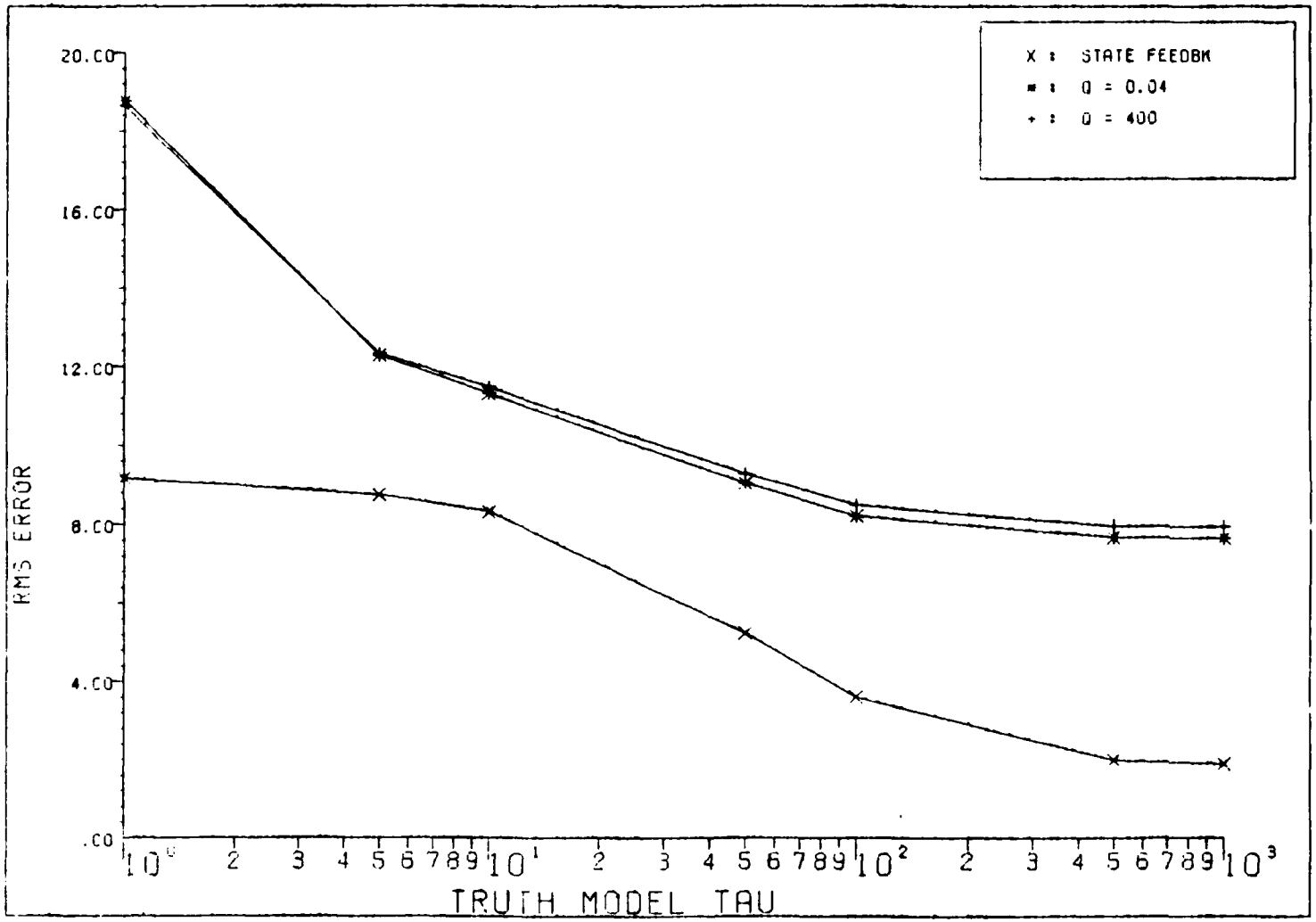


Figure 5-14. Robustness Analysis: Tracker RMS Error vs True Target Time Constant (Third)

VI. CONCLUSIONS AND RECOMMENDATIONS

VI.1 CONCLUSIONS

The purpose of this thesis was to continue working the particle beam problem which was begun by Meer, Zicker, and Moose (Refs. 9, 18, and 10). The first step was to correct Moose's software for the PI controller and perform the same analyses on the tracker as he did in order to verify and improve his results. First, a sensitivity analysis was performed on the tracker by varying the parameters associated with the target in both the truth and filter models. Second, robustness of Moose's corrected PI controller was studied by inputting unmodelled constant and sinusoidal disturbances into the truth model.

The results obtained by performing the above sensitivity and robustness analyses were not very much different from the values obtained by Moose. By properly adjusting the weighting matrix of the cost function, RMS errors were improved by approximately 0.1-0.2 cm--not vastly significant. The conclusion here is that the corrected off-diagonal terms in the weighting matrix of the cost function provide some impact, but not a great deal.

Once finished with Moose's corrected software, attention was turned to the incorporation of a first order Gauss-Markov target acceleration model into the software. As far as the Kalman filter was concerned, there were no problems associated with the additional integration terms provided by this model. However, for the PI controller, we were required to force these poles away from the origin in order to guarantee stabilizability which, along with detectability, gives us the sufficient

conditions for steady state Riccati equation solutions.

With this new filter/controller in hand, a robustness study was performed by varying target parameters in the real world (truth model) without telling the filter. As a result, performance was severely degraded. In order to robustify the system, LTR techniques were used, but they were ineffective. The reason is that the Meer filter already places heavy emphasis on the measured events, so added white noise at the control inputs of the dynamics model has little effect.

VI.2 RECOMMENDATIONS FOR FURTHER WORK

There is still a great deal of work which needs to be accomplished on the particle beam pointing and tracking problem. Now that a realistic accelerating target model has been incorporated into the PI controller structure, several other studies might be carried out.

First, a full scale sensitivity analysis for the tracker might be carried out. It was not possible to accomplish such a study in this thesis due to time constraints. This would provide insight as to how effective the PI controller is when placed in different environments, where the filter is given complete information about the environment of interest.

Second, robustness could be studied by the incorporation of unmodelled beam actuator dynamics in the truth model. Quite often it is found in controller designs that the addition of the time delays caused by higher order actuator dynamics can impact the performance and robustness of the system greatly.

Third, unmodelled sinusoidal disturbances could be incorporated into the problem. A performance analysis could be completed similar to

that involving the first order Gauss-Markov position model. It is suspected that LTR tuning could have a significant impact on performance for the lower frequency sinusoids.

Fourth, non-LTR tuning could be carried out for the robustness analysis described in the previous section. The Q_T, R, and τ terms could be adjusted within the filter in order to allow a larger range of variations (of these same parameters) in the truth model.

Fifth and by far the most important, an adaptive estimator for the states and parameters of the system (and thus also an adaptive controller) must be designed for the particle beam problem if it is to become more robust to changing parameters in the real world. This should be the primary focus of the next thesis dealing with the particle beam control problem.

References

1. Hecht, J. Beam Weapons: The Next Arms Race, New York: Plenum Press, 1984.

2. Kirk, D. E., Optimal Control Theory, New Jersey: Prentice Hall, 1970.

3. Kwakernaak, H. and R. Sivan, Linear Optimal Control Systems, New York: Wiley-Interscience, 1972.

4. Maybeck, P. S., Stochastic Models, Estimation, and Control, New York: Academic Press, 1979 (Vol. I), 1982 (Vols. II and III).

5. Maybeck, S. and D. E. Meer, "Multiple Model Adaptive Estimation for Space-Time Point Process Observation," Proceedings of the IEEE National Aerospace and Electronics Conference, Las Vegas, NV, Dec 1984.

6. Maybeck, P. S., W. Miller, and J. Howey, "Robustness Enhancement for LQG Digital Flight Controller Design," Proceedings of the IEEE National Aerospace and Electronics Conference, Dayton, OH, May 1984.

7. Maybeck, P. S., S. R. Robinson, and J. Santiago, "Tracking a Swarm of Fireflies in the Presence of Stationary Stragglers," Proceedings of the 1979 International Symposium on Information Theory, Grignano, Italy, June 1979.

8. Maybeck, P. S. and L. Zicker, "MMAE - Based Control with Space Time Point Process Observations," IEEE Transaction on Aerospace and Electronic Systems, Vol. AES-21, No. 3, May 1985, pp. 292-300.

9. Meer, D. E., Multiple Model Adaptive Estimation for Space-Time Point Process Observations, PhD Dissertation, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, Ohio, September, 1982.

10. Moose, W. J., A Proportional-Plus-Integral Controller for a Particle Beam Weapon, MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, Ohio, December 1984.

11. Musick, S. H., SOFE: A Generalized Digital Simulation for Optimal Filter Evaluation User's Manual, AFWAL Technical Report 80-1108, Wright-Patterson AFB, Ohio: Air Force Wright Aeronautical Laboratories, 1980.

12. Musick, S. H., R. E. Feldman, and J. G. Jensen, SOFEPL: A Plotting Postprocessor for 'SOFE' User's Manual, AFWAL Technical Report 80-1108, Wright-Patterson AFB, Ohio: Air Force Wright Aeronautical Laboratories, 1981.

13. Papoulis, A., Probability, Random Variables, and Stochastic Processes. New York: McGraw-Hill, 1965.

14. Santiago, J. M., Jr., Fundamental Limitations of Optical Trackers, MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, Ohio, December 1978.

15. Snyder, D. C. and P. M. Fishman, "How to Track a Swarm of Fireflies by Observing Their Flashes," IEEE Transactions on Information Theory: 692-695 (November 1975).

16. Stein, G., "LQG-Based Multivariable Design: Frequency Domain Interpretation," Systems and Research Center, Honeywell Inc., Minneapolis, Minnesota.

17. Weiss, J. L., T. N. Upadhyay, and R. Tenney, "Finite Computable Filters for Linear Systems Subject to Time Varying Model Uncertainty," paper presented at IEEE National Aerospace and Electronics Conference May 1983 of results obtained in a Master's thesis by Jerold L. Weiss, "A Comparison of Finite Filtering Methods for Status Directed Processes," Massachusetts Institute of Technology, Cambridge, Massachusetts, June 1983.

18. Zicker, W. L., Pointing and Tracking of Particle Beams, MS Thesis, School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, Ohio, December 1983.

VITA

Lawrence C. Jamerson was born on 20 December 1962 in Memphis, Tennessee. After attending First Colonial High School in Virginia Beach, Virginia, he was awarded a four-year AFROTC Scholarship to attend the University (of Virginia). While at Virginia, he participated in the ROTC program and completed the academic requirements for a Bachelor of Science in Electrical Engineering. Upon graduation and commissioning, he was assigned to the Air Force Institute of Technology at Wright-Patterson Air Force Base. While there, he completed the Advanced, Stochastic, and Flight Control sequences and was consequently assigned to the Avionics Laboratory of the Air Force Wright Aeronautical Labs (Wright-Patterson AFB).

Permanent address: 1807 Sheringham East
Virginia Beach, Virginia
23454

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

AD-A163 945

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GE/ENG/85D-22	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION / 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (City, State and ZIP Code) Air Force Institute of Technology Wright-Patterson AFB, Ohio 45433	7b. ADDRESS (City, State and ZIP Code)
8a. NAME OF FUNDING/SPONSORING ORGANIZATION AF Weapons Lab / 8b. OFFICE SYMBOL (If applicable) NTYP	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State and ZIP Code) Kirtland AFB, NM	10. SOURCE OF FUNDING NOS. PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT NO.
11. TITLE (Include Security Classification) See Box 19	
12. PERSONAL AUTHOR(S) Lawrence C. Jamerson, B.S., 2d Lt, USAF	
13a. TYPE OF REPORT MS Thesis / 13b. TIME COVERED FROM _____ TO _____	14. DATE OF REPORT (Yr., Mo., Day) 1985 December / 15. PAGE COUNT 110
16. SUPPLEMENTARY NOTATION	

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB. GR.	Point Process, Space-time Point Process, Multiple Model Adaptive Estimator, Adaptive Estimation, Optimal Filtering, Optimal Control Theory, LQG Controller, (see reverse)
09	04		
12	01		

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

Title: PARTICLE BEAM TRACKER FOR AN ACCELERATING TARGET

Thesis Chairman: Peter S. Maybeck, PhD

Approved for public release IAW AFR 190-1.
LYNN E. WOLAVER 16 JAN 86
Dean for Research and Professional Development
Air Force Institute of Technology (AFSC)
Wright-Patterson AFB OH 45433

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL Peter S. Maybeck	22b. TELEPHONE NUMBER (Include Area Code) (513) 255-3576 / 22c. OFFICE SYMBOL AFIT/ENG

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE.

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

Block 18: PI Controller Design, Tracker, Assumed Certainty Equivalence, Snyder-Fishman Filter, Gauss-Markov Acceleration Model, LTR Tuning.

(Proportional plus INTEGRAL)

The purpose is to use a PI controller to point the centroid of a particle beam at an intended target. Multiple Model Adaptive Estimator is used to estimate the centroid of a one-dimensional Gaussian shaped source of photo-electron events. "Merge Method" of filter pruning is used to limit the size of this filter. A standard Kalman filter is used to estimate the position of the target where the target is initially modelled as a first-order Gauss-Markov position process and later as a first-order Gauss-Markov acceleration process. A PI controller is designed using LQG methods, and true states are replaced by their best estimates by invoking the principle of assumed certainty equivalence. With a target position model within the software, a parameter sensitivity analysis is performed as well as a robustness study where unmodelled constant and sinusoidal disturbances are added to the system. With a target acceleration model within the software, a robustness study is carried out where target parameters are varied within the truth model without telling the filter. In an attempt to recover full state feedback robustness qualities, Loop Transmission Recovery tuning is attempted. (Thesis)

END

FILMED

3-86

DTIC