NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

BEARINGS ONLY TARGET TRACKING-
MANEUVERING TARGET

by

Dimitrios Kourkoulis

December 1984

Thesis Advisor: H. H. Loomis

Approved for public release; distribution is unlimited.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Bearings Only Target Tracking-Maneuvering Target		5. TYPE OF REPORT & PERIOD COVERED Masters Thesis December 1984
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Dimitrios Kourkoulis		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		12. REPORT DATE December 1984
		13. NUMBER OF PAGES 118
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Kalman Filter
Passive Tracking
Bearings-Only Tracking
Extended Kalman Filter

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

The estimation of the position and velocity of a target moving in a two-dimensional frame is studied in this paper. The estimator is a Kalman filter which processes noisy bearings of the target gathered by the tracker.

The case of maneuvering targets is examined and a solution using a variable value of the system's noise covariance matrix is studied.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Approved for public release; distribution is unlimited.

Bearings Only Target Tracking - Maneuvering Target

by

Dimitrios Kourkoulis
Lieutenant, Hellenic Navy
B.S., Hellenic Naval Academy, 1976

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN ENGINEERING SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
December 1984

ABSTRACT

The estimation of the position and velocity of a target moving in a two-dimensional frame is studied in this paper. The estimator is a Kalman filter which processes noisy bearings of the target gathered by the tracker.

The case of maneuvering targets is examined and a solution using a variable value of the system's noise covariance matrix is studied.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

The author would like to express his gratitude to his Thesis Advisor Dr. Herschel Loomis for his help and guidance during the course of this research.

Special thanks are due to his wife Helen for her patience, encouragement and support throughout this effort.

I. INTRODUCTION

The problem discussed in this paper is that of estimating the position and velocity in two dimensions of a target by means of processing passively obtained bearing measurements.

A single moving observer (tracker) monitors noisy sonar bearings from a radiating acoustic source (target). The geometric configuration is depicted in Figure 1.1.

The problem contains nonlinearities so the conventional linear analysis is not possible. Also as it will be shown in chapter IV the dynamic process remains unobservable prior to tracker maneuver. That requirement of observer maneuvering distinguishes this problem from the more usual target motion analysis (TMA) problem.

In chapter II the basic concept of the Kalman filter is described. Chapter III describes the non-linear case (Extended Kalman filter) in which category the bearings only tracking problem belongs.

In chapters IV and V the problem of bearings only tracking with nonmaneuvering and maneuvering targets is discussed. Some possible solutions from the literature are referenced, and the case of solving the problem through a specific approach, i.e by using a variable value of the system's noise covariance matrix "Q" is tested.

Chapter VI contains the results of the computer simulations on the subject and chapter VII contains conclusions and possible topics for further investigation.

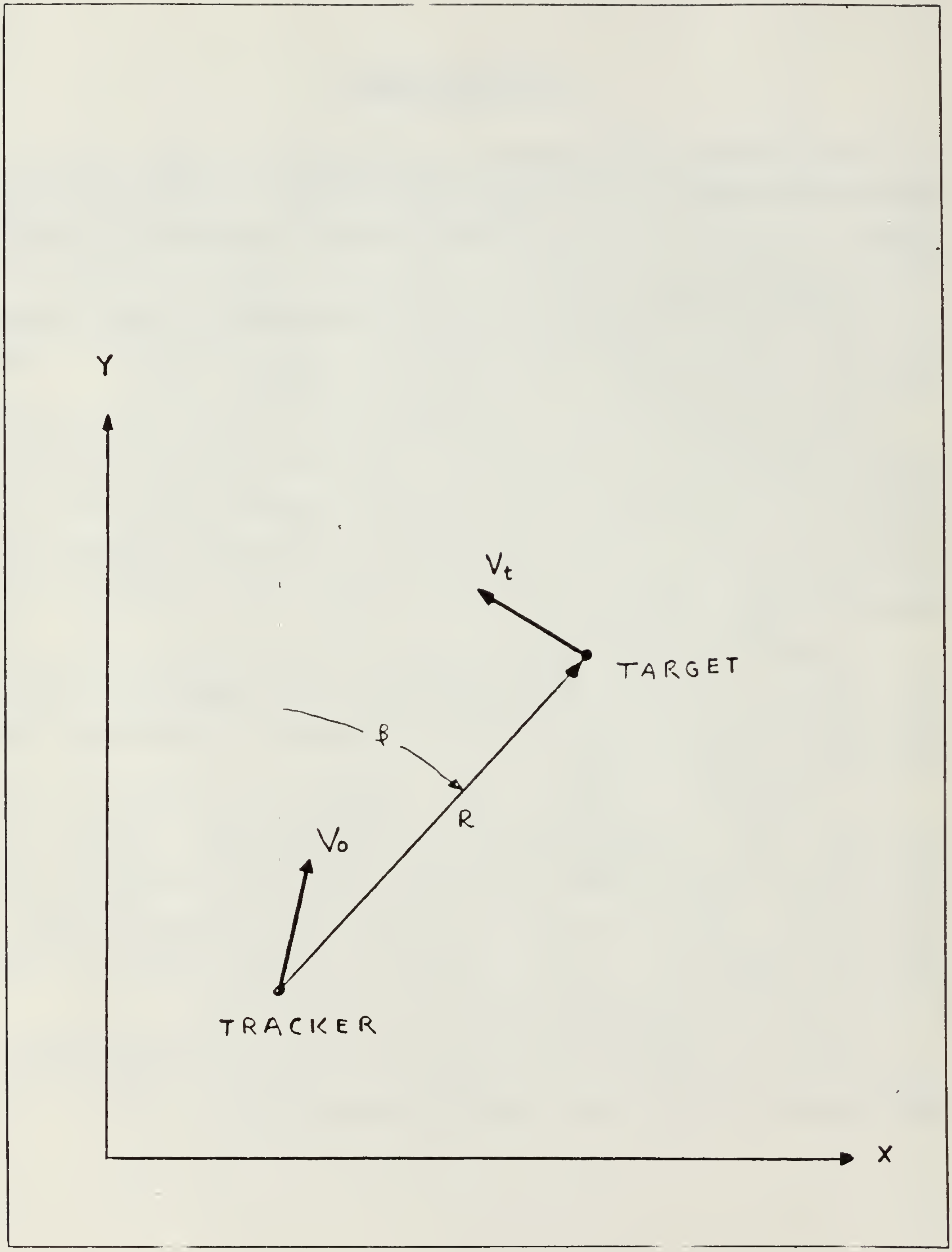

Figure 1.1 Geometrical Configuration for the B.O.T Problem.

II. KALMAN FILTERING BASICS

A. HISTORY

In 1960, R.E. Kalman provided a new way of formulating the least squares filtering problem using state-space methods [Ref. 1]. Until that time the Wiener solution of the optimal filter problem was applied , which was using the concept of the "weighting function". In effect the weighting function tells how the past values of the input should be weighted in order to determine the present value of the output, that is the optimal estimate. But the Wiener solution did not lend itself very well to the corresponding discrete-data problem nor was it easily extended to more complicated problems [Ref. 2].

The two main features of the Kalman formulation and solution of the problem are:

Vector modeling of the random processes under consideration.

Recursive processing of the noisy measurements (input data). The key element in any recursive procedure is the use of the results of the previous step to aid in obtaining the desired result for the current step. This is the main feature of the Kalman filtering and the one that clearly distinguishes it from the weighting function approach, which requires arithmetic operations on all the past data.

The Kalman filtering technique has become very popular in target tracking applications for the previous reasons plus the following:

At a given time t, the filter generates an unbiased estimate of the state vector, which means that the expected value of the estimate is the value of the state vector at time t.

The estimate is a minimum variance estimate meaning that it has the property that its error covariance is less than or equal to that of any other linear unbiased estimate.

The filter is linear and simplifies the calculations [Ref. 3].

B. THE DISCRETE KALMAN FILTER

Assume that the random process to be estimated can be modeled in the form:

$$x(k+1) = \Phi(k)x(k) + \Gamma w(k) + \Delta u(k) \tag{2.1}$$

and the observation or measurement of the process is assumed to occur at discrete points in time in accordance with the relationship:

$$z(k) = H(k)x(k) + n(k) \tag{2.2}$$

where:

$x(k) = (n\times1)$; is the process state at time $t(k)$

$\Phi(k) = (n\times n)$; is the matrix relating $x(k)$ to $x(k+1)$ in the absence of forcing function.

$w(k)$; is the random forcing input at time $t(k)$ considered to be an uncorrelated sequence with zero mean and known variance.

$\Gamma(k) = (nxp)$; is the matrix relating the random forcing inputs to the state at time t(k).[1]

u(k) ; is the deterministic forcing input at time t(k).

$\Delta(k) = (nxp)$; is the matrix relating the deterministic inputs to the state at time t(t).

z(k) = (mxl) ; is the measurement vector at time t(k)

H(k) = (mxn) ; is the matrix which gives the noiseless connection between the state vector and the measurement equation at time t(k).

n(k) = (mxl) ; is the measurement noise error which is assumed to be a white sequence with known covariance structure and uncorrelated with the w(k) sequence.

The corresponding covariance matrices are given by:[2]

$$E[w_k w_i'] = \begin{cases} Q(k) & k = i \\ 0 & k \neq i \end{cases} \tag{2.3}$$

$$E[n_k n_i'] = \begin{cases} R(k) & k = i \\ 0 & k \neq i \end{cases} \tag{2.4}$$

$$E[w_k n_i'] = 0 \quad \text{for all k and i} \tag{2.5}$$

---

[1] $p \leq n$

[2] (') denotes matrix transposition.

It is assumed that we have available an initial estimate of the process at time t(k), which is based on the knowledge about the process prior to time t(k). This prior estimate will be denoted as $\hat{x}(k/k-1)$ where the "hat" denotes estimate, and the (k/k-1) subscript means that this is our estimate prior to processing the measurement at time t(k).

With the assumption of the prior estimate $\hat{x}(k/k-1)$, we now seek to use the measurement z(k) to improve the estimate. To do that we choose a linear blending of the received noisy measurement and the prior estimate in accordance with the equation:

$$\hat{x}(k/k) = \hat{x}(k/k-1)+G(k)[z(k)-H(k)\hat{x}(k/k-1)] \qquad (2.6)$$

where $\hat{x}(k/k)$ is the update estimate, $\hat{x}(k/k-1)$ is given by:

$$\hat{x}(k/k-1) = \phi(k)\hat{x}(k-1/k-1) \qquad (2.7)$$

and the G(k) is the blending factor. G(k) is going to be determined later. At this time the problem is to find a particular value of G(k) that yields an update estimate that is optimal in some sense. The minimum Mean - Square error is the required performance criterion for that "optimization". To do that we need to define the term "error covariance matrix" P(k), associated with the update (a posteriori) estimate, which is a matrix representing the covariance of the difference between the true state vector x(k) and the estimated $\hat{x}(k)$.

$$P(k) = E\,[(x(k)-\hat{x}(k/k-1))(x(k)-\hat{x}(k/k-1))'] \qquad (2.8)$$

The optimization can be done by various mathematical ways as treated in [Ref. 4] [Ref. 2] and [Ref. 5]. The mathematical derivation which is omitted here shows that if

$$G(k)=P(k/k-1)H'(k)[H(k)P(k/k-1)H'(k)+R(k)]^{-1} \quad (2.9)$$

then this is the G(k) that minimizes the mean square estimation error, and it is called the "Kalman gain" [Ref. 2].

Next the covariance matrix associated with the optimal estimate may be computed and is given by:[3]

$$P(k/k) = [I - G(k)H(k)]P(k/k-1) \quad (2.10)$$

Now the updated estimate x(k/k) can be easily projected ahead via the transition matrix by the equation:

$$\hat{x}(k+1/k) = \phi(k)\hat{x}(k/k) \quad (2.11)$$

ignoring the contribution of w(k) because it has zero mean and also it is uncorrelated with the previous W's.

Also, the equation

$$P(k+1/k) = \phi(k)P(k/k)\phi'(k)+Q(k) \quad (2.12)$$

closes the loop and now, having the needed quantities for the next moment with the next measurement we can start again as in the previous steps.

Equations (2.6), (2,9), (2,10), (2,11), and (2,12) thus comprise the Kalman filter recursive equation set.

In Figure 2.1 the Kalman filter loop is indicated.

1. <u>A Simple Example</u>

Assume that a stationary tracker has the ability to obtain range measurements in both X and Y directions of a target moving as in Figure 2.2.

---

[3](I) is the identity matrix.

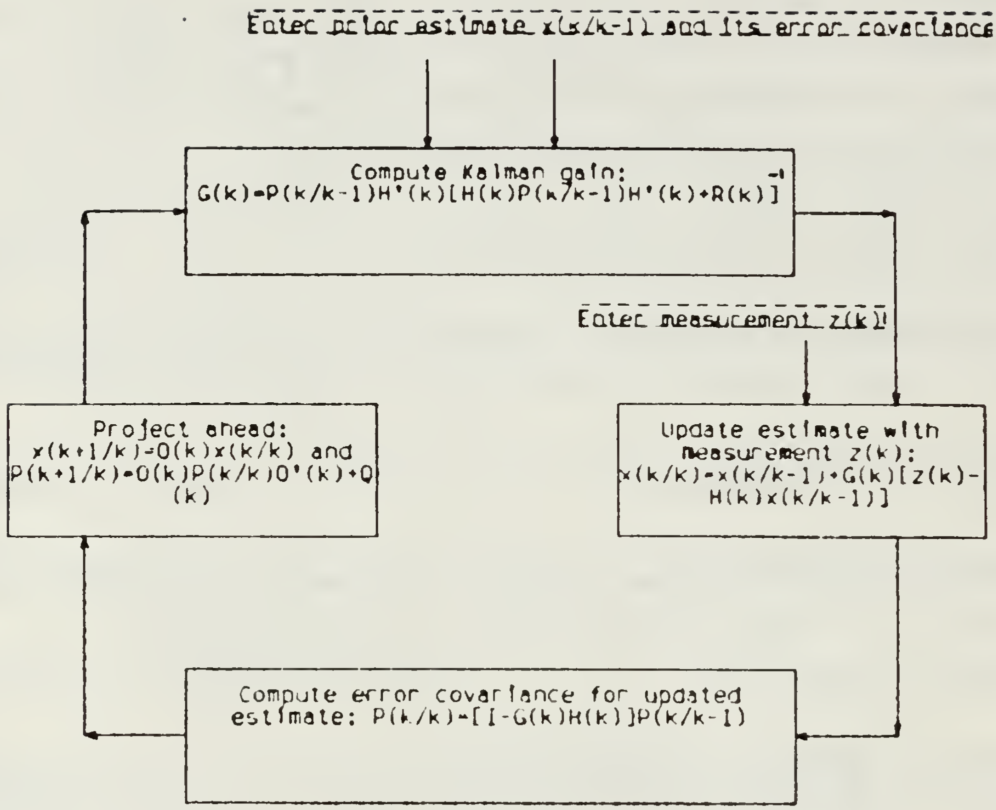

Figure 2.1 The Kalman Filter Loop.

Let the target be moving with a tangential velocity of 1,660 m/min so that it covers the arc of 90° in 10 minutes. The tracker makes its measurements every 1 min. It is desired to estimate the state vector of the target defined as X, V_X, Y, and V_Y, i.e., range and velocity in X and Y directions. Given are: an initial estimate $\hat{x}(k/k-1)$ and its error covariance matrix $P(k/k-1)$. Let them be:

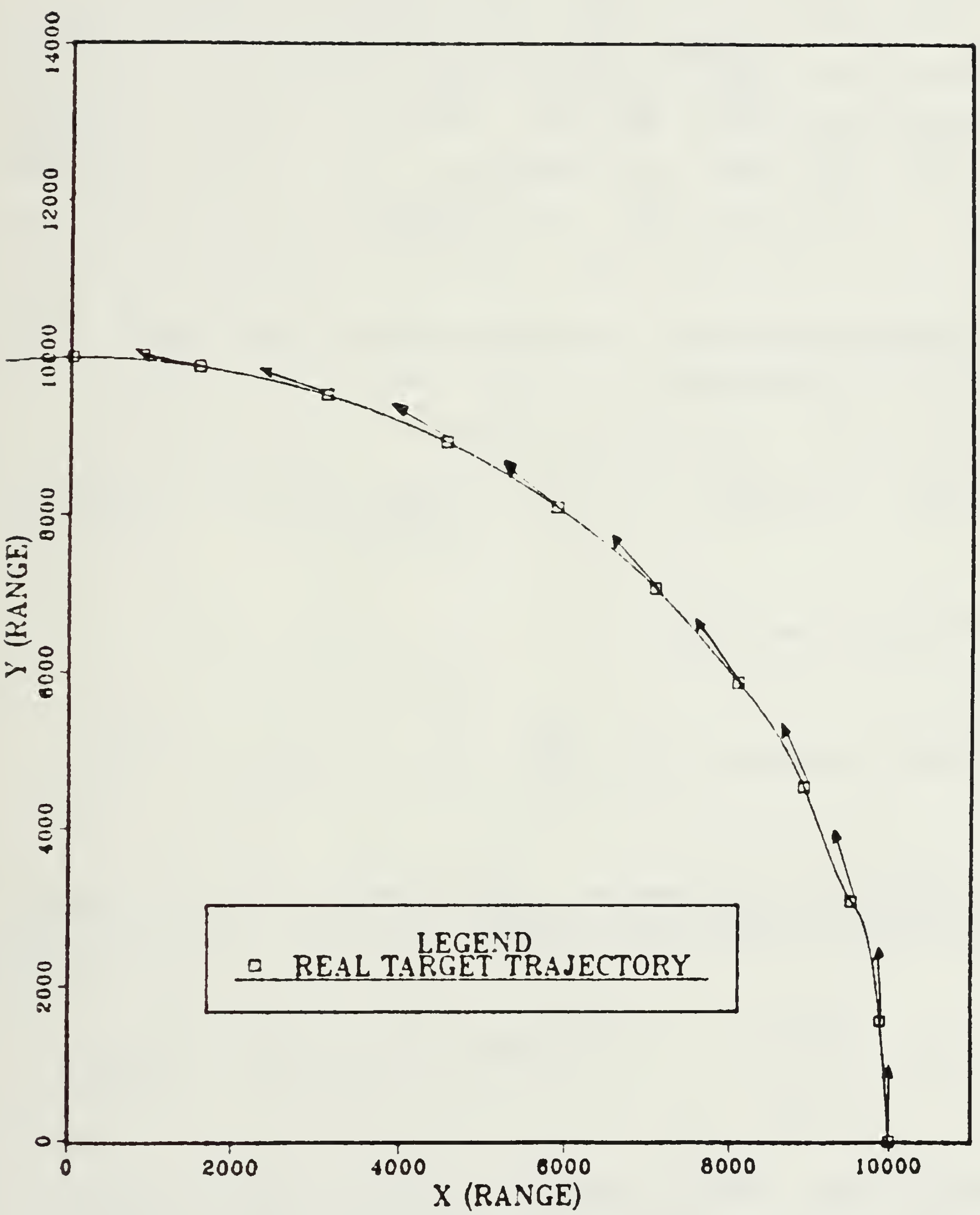

Figure 2.2 Tracker-Target Configuration in Example.

$$\hat{x}(k/k-1) = \begin{bmatrix} 10000 \\ 0 \\ 0 \\ 1600 \end{bmatrix} \quad (2.13)$$

and

$$P(k/k-1) = \begin{bmatrix} 1000 & 0 & 0 & 0 \\ 0 & 1000 & 0 & 0 \\ 0 & 0 & 1000 & 0 \\ 0 & 0 & 0 & 1000 \end{bmatrix} \quad (2.14)$$

Then we can calculate the Kalman filter gain G(k) as in equation (2.9) where:

$$H(k) = \text{constant} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (2.15)$$

and R(k) has the value:

$$R(k) = \text{constant} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (2.16)$$

Next, given the measurement, the updated estimate is calculated using equation (2.6) where:

$$z(k) = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \quad (2.17)$$

We can see here that the updated estimate $\hat{x}(k/k)$ depends on the previous $\hat{x}(k-1/k-1)$ propagated for the instant (k) i.e.,

$\hat{x}(k/k-1)$, and another portion equal to $G(k)[z(k)-H(k)x(k)]$. That second portion depends on the $G(k)$ and on how much the estimated and the received measurements differ.

The updated error covariance matrix $P(k/k)$ is then computed using equation (2.10). The updated error covariance $P(k/k)$ is going to be less than the previous $P(k/k-1)$ since the filter processed an observation and thus the uncertainty about the estimate is less. The term $[I-G(k)H(k)]$ is always less than unity if $G(k)$ is not zero. That means that if we used the last observation (i.e., $G(k)$ not zero) then the term in the brackets is less than unity and $P(k)$ becomes less than $P(k/k-1)$.

Now having $\hat{x}(k/k)$ and $P(k/k)$ we must propagate them for the next instant when the next measurement will be taken in order to be able to compare it with the real one through the new measurement. So we project ahead our estimate by the equation (2.11) where:

$$\phi(k) = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.18}$$

and the error covariance matrix by equation (2.12) with $Q(k)$ such that:

$$Q(k) = \Gamma(k)E[w(k)w'(k)]\Gamma'(k) \tag{2.19}$$

where:

$$\Gamma'(k) = \begin{bmatrix} \frac{T^2}{2} & T & 0 & 0 \end{bmatrix} \tag{2.20}$$

and w(k) is the random forcing input at time t(k) which is to be formulated as a white noise with known variance. Finally Q(k) is given by:

$$Q(k) = \begin{bmatrix} \frac{T^4}{4} & \frac{T^3}{2} & 0 & 0 \\ \frac{T^3}{2} & T^2 & 0 & 0 \\ 0 & 0 & \frac{T^4}{4} & \frac{T^3}{2} \\ 0 & 0 & \frac{T^3}{2} & T^2 \end{bmatrix} \cdot w(k) \tag{2.21}$$

and it counts for the uncertainty introduced by the fact that we do not know if the target during the next coming time interval will maneuver or not. A big value of w(k) means that the target is very likely during the propagation interval to maneuver. In this case the Kalman gain will also be large and the filter will weight the observation more than the propagated state. On the opposite case if w(k) is zero the filter assumes that the target did not maneuver during that interval so it weights more the last estimate than the measurement. In the above case it is also assumed that the target acceleration in X direction is uncorrelated to the acceleration in Y direction for simplicity.

Having the propagated values of $\hat{x}(k+1/k)$ and P(k+1/k) we can start over again from the initial step.

The above algorithm was simulated in the computer. The interesting result obtained is that for the case that the target maneuvers the choice of w(k) is very important. If it is small or zero the filter does not include any extra uncertainty due to possible target maneuver. So at any moment it gives more weight to the last propagated estimation and less to the received measurement. Thus the tracking accuracy is not good compared with that in which it includes uncertainty as can be seen in the results shown in Figures 2.3, 2.4, 2.5, 2.6, and 2.7.

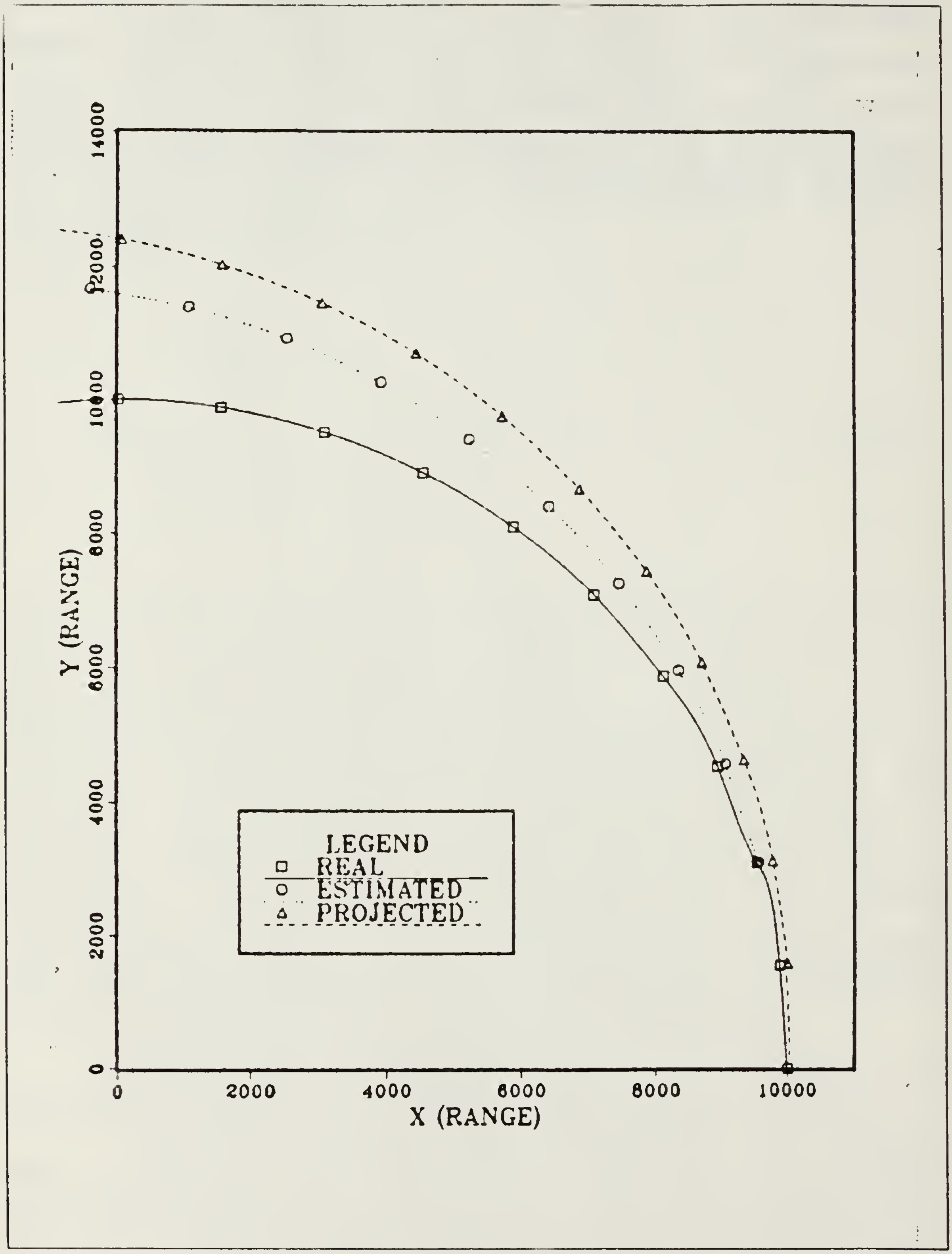

Figure 2.3 Filter Behavior for w=0.0.

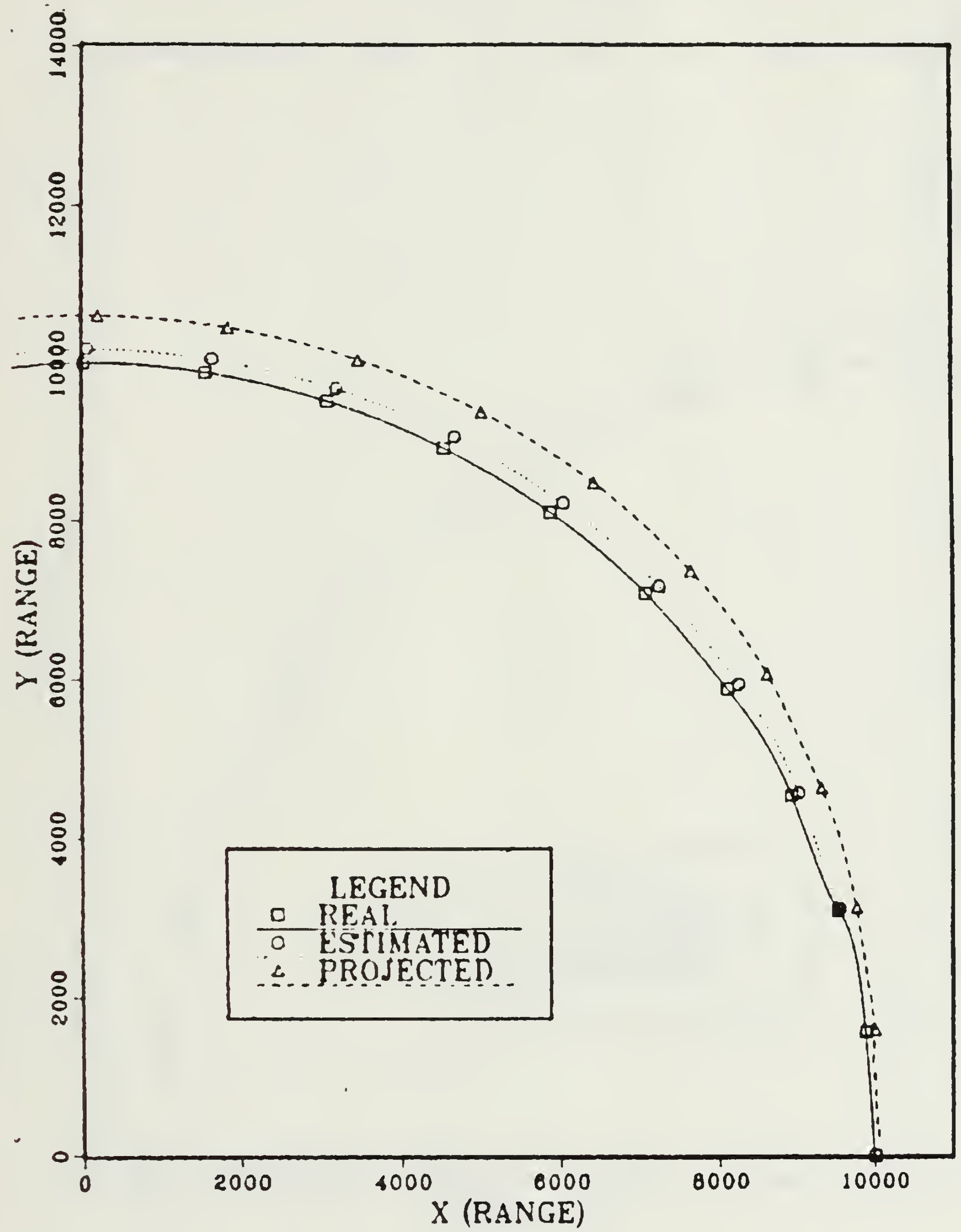

Figure 2.4 Filter Behavior for w=0.5.

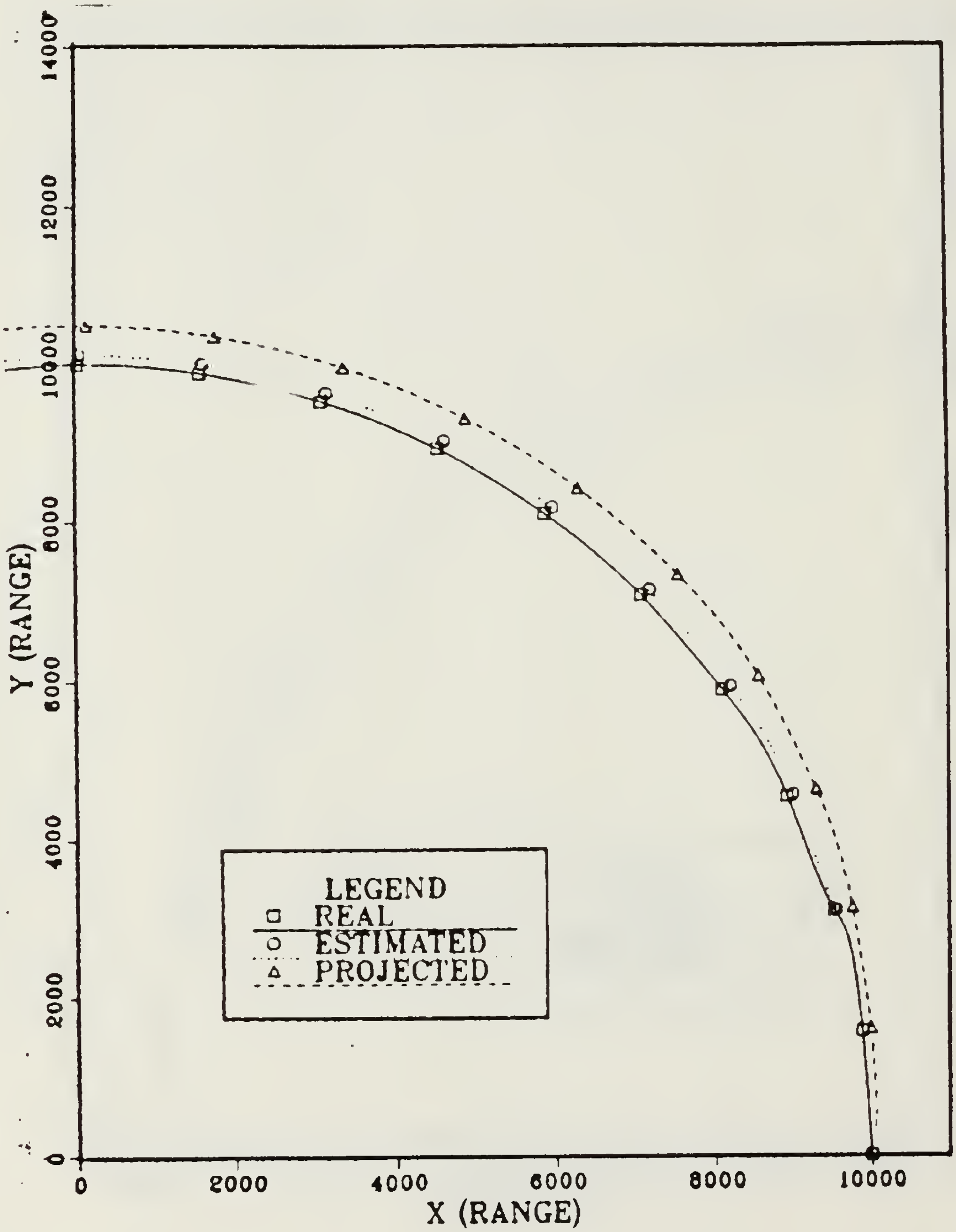

Figure 2.5 Filter Behavior for w=1.0.

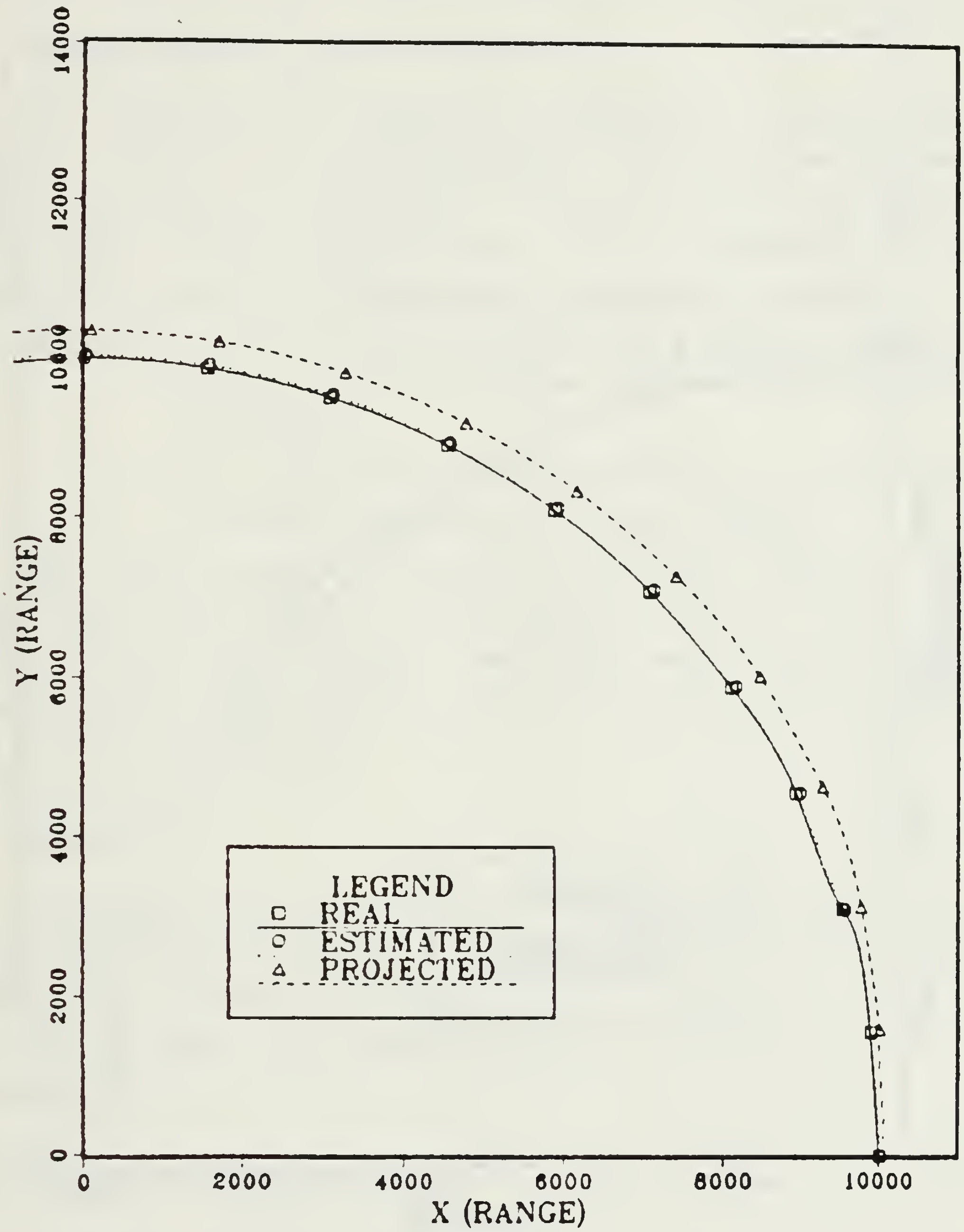

Figure 2.6 Filter Behavior for w=3.0.

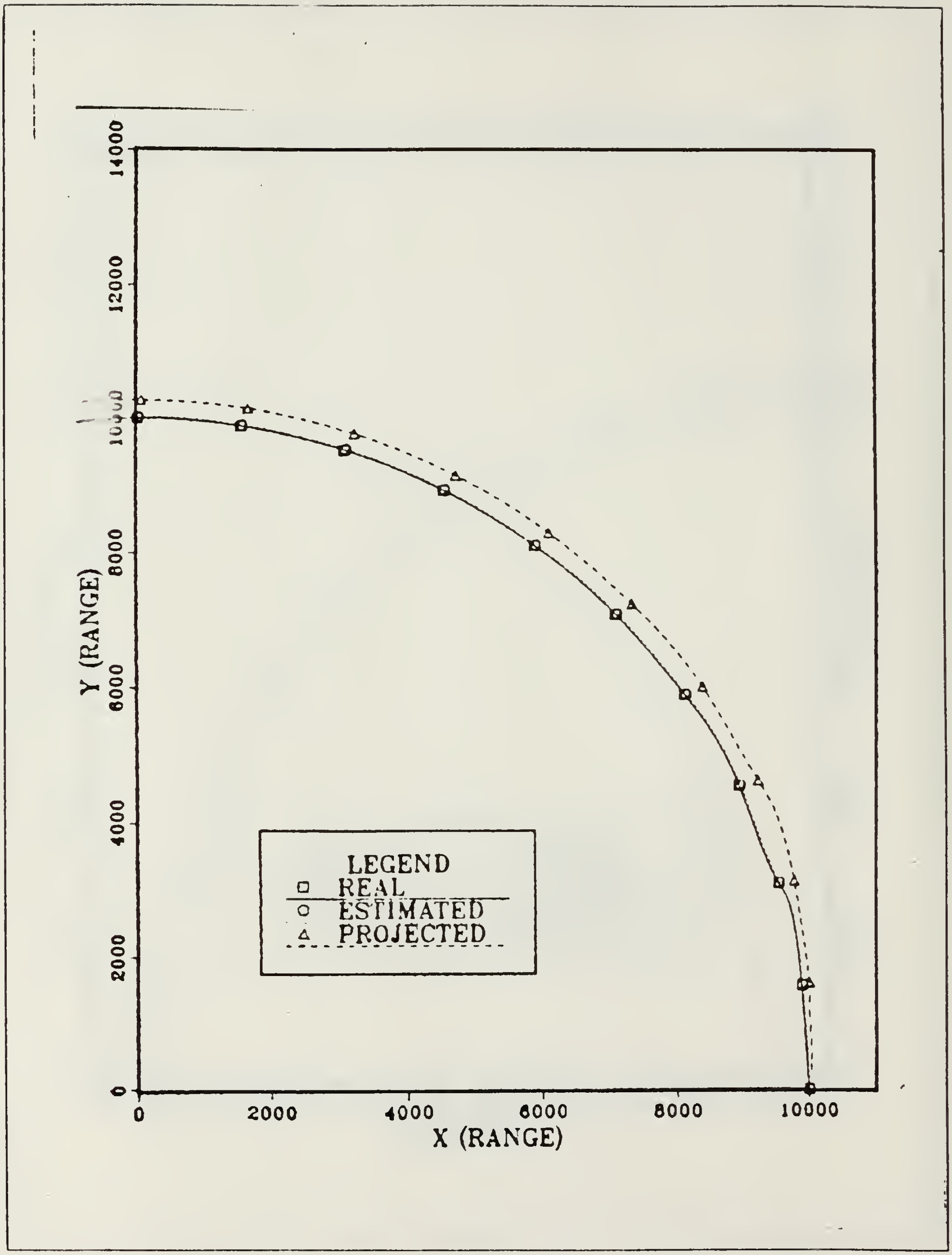

Figure 2.7 Filter Behavior for w=10.0.

III. NONLINEAR ESTIMATION

A. INTRODUCTION

The majority of physical phenomena are nonlinear in nature. So as a result ,usually ,the state and/or measurement equations are nonlinear. Since the basic Kalman filter theory deals with linear cases, it is necessary to find a "method" to use it in nonlinear estimation problems.

There are two ways of solving that problem: [Ref. 4].

1. By deriving an optimal filter for the nonlinear problem or
2. By linearizing (approximating) the problem and applying the linear filter theory.

The first method is hard to follow and will involve complicated mathematical computations. On the other hand the second method is easier and the more usual. For the reasons above the second method will be followed in this and the following chapters.

B. ANALYSIS

In the following analysis it is assumed that both the state and the measurement equations are nonlinear although this is not always the case.

Assume that the random process to be estimated can be modeled by:

$$x(k+1) = a[x(k),u(k),k] + w(k) \tag{3.1}$$

with the measurement equation:

$$z(k) = c[x(k)] + n(k) \tag{3.2}$$

It is necessary to have available a nominal trajectory $x\ (k)$, $k=0,1,2,\ldots.$ about which the linearization will be performed. The vector function $a[x(k),u(k),k]$ is expanded in Taylor series about the nominal trajectory $x\ (k)$. Then the linearized state equations can be written:

$$x(k+1)=a[x^{(o)}(k),u(k),k]+\left.\frac{\partial a}{\partial x}\right|_{[x\ (k),u(k),k]}[x(k)-x\ (k)]+w(k) \tag{3.3}$$

If $A(k)$ is defined to be the first partial derivative of the nonlinear function $a[x\ (k),u(k),k]$, with respect to the state vector $x(k)$, i.e.,

$$A(k) \triangleq \left.\frac{\partial \alpha}{\partial x}\right|_{[x^{(o)}(k),u(k),k]} \tag{3.4}$$

Then, the ijth entry of matrix A is given by:

$$(A)_{ij} \triangleq \left.\frac{\partial \alpha_i}{\partial x_j}\right|_{[x^{(o)}(k),u(k),k]} \tag{3.5}$$

Also ,the vector function $a[x^{(o)}(k),u(k),k]$ is a known function of k. Thus the linearized state equations can be written as:

$$\begin{aligned} x(k+1)=A(k)x(k)+a[x^{(o)}(k),u(k),k] \\ -A(k)x^{(o)}(k)+w(k) \end{aligned} \tag{3.6}$$

The accuracy of this approximation depends on how close the nominal trajectory to the actual one was selected.

Let us now consider the measurement equation. We have:

$$z(k) = c[x(k)]+n(k) \tag{3.7}$$

Again we can expand the nonlinear vector function c about the nominal trajectory x (k) with the result:

$$z(k)=c[x^{(o)}(k)]+\left.\frac{\partial c}{\partial x}\right|_{x^{(o)}(k)}[x(k)-x^{(o)}(k)]+n(k) \qquad (3.8)$$

Defining

$$H(k) \triangleq \left.\frac{\partial c}{\partial x}\right|_{x^{(o)}(k)} \qquad (3.9)$$

we can write:

$$z(k)=H(k)x(k)+c[x^{(o)}(k)]-H(k)x^{(o)}(k)+n(k) \qquad (3.10)$$

Again as in the linearized state equation, the two terms in the middle of the equation (3.10) are known quantities and they can be handled as if they were a time varying but known measurement bias. For simplification if we will define[4]

$$u'(k) = a[x^{(o)}(k),u(k),k] - A(k)\ x^{(o)}(k) \qquad (3.11)$$

and

$$z'(k) = z(k)-c[x^{(o)}(k)]+H(k)x^{(o)}(k) = z(k)-\beta(k) \qquad (3.12)$$

where:

[4] (') in this case means "prime"

$$\beta(k) = c[x^{(0)}(k) - H(k)x^{(0)}(k)] \tag{3.13}$$

we can rewrite equations (3.6) and (3.7) as:

$$x(k+1) = A(k)x(k) + u(k) + w(k) \tag{3.14}$$

and

$$z(k) = H(k)x(k) + \beta(k) + n(k) \tag{3.15}$$

Then starting with these linearized equations, the appropriate Kalman filter equations are:

the gain equation:

$$G(k) = P(k/k-1)H'(k)[H(k)P(k/k-1)H'(k)+R(k)]^{-1} \tag{3.16}$$

the covariance of estimation error equations:

$$P(k/K-1) = A(k-1)P(k-1/k-1)A'(k-1)+Q(k-1) \tag{3.17}$$

$$P(k/k) = [I-G(k)H(k)]\, P(k/k-1) \tag{3.18}$$

the filter update equation:

$$\hat{x}(k/k) = \hat{x}(k/k-1)+G(k)[z(k)-c(\hat{x}(k/k-1))] \tag{3.19}$$

and the prediction equation:

$$\hat{x}(k+1/k) = a[\hat{x}(k/k),u(k),k] \tag{3.20}$$

Notice that in equations (3.19) and (3.20) the nonlinear state and measurement relationships are used. An alternative is to use the linearized relationships in which case we have:

$$\hat{x}(k/k) = \hat{x}(k/k-1)+G(k)[z(k)-H(k)\hat{x}(k/k-1)] \tag{3.21}$$

and

$$\hat{x}(k+1/k) = A(k)\hat{x}(k/k) + u(k) \tag{3.22}$$

One question to be answered now is how to determine the "nominal" trajectory used before. One way is to use an approximate trajectory that is known in advance. This trajectory may be available from known data, or may have been computed by solving the state equations:

$$\hat{x}^{(o)}(k+1) = a[x^{(o)}(k),u(k),k] \tag{3.23}$$

with the initial condition $x^{(o)}(0) = E[x(0)]$. Unfortunately, if the uncertainty in x(0) is large the solution of equation (3.23) may be "too far" from x(k), the linerization error too big and the whole method inaccurate.

C. THE EXTENDED KALMAN FILTER

Another possibility is to use the estimates produced by the filter as the nominal trajectory about which the linearization is performed. The estimator equations are again given by equations (3.21) and (3.22). The matrices A(k) and H(k) must be used to generate G(k) so that it is available to process z(k) when it is available. Thus the best information we have when H(k) must be evaluated is $\hat{x}(k/k-1)$; when A(k) is to be evaluated, however, $\hat{x}(k/k)$ is available. Hence:

$$A(k) \triangleq \left. \frac{\partial a}{\partial x} \right|_{[\hat{x}(k|k), u(k), k]} \tag{3.24}$$

and

$$H(k) \triangleq \left. \frac{\partial c}{\partial x} \right|_{[\hat{x}(k|k-1)]} \tag{3.25}$$

The H(k) and A(k) matrices must be computed on-line and not in advance since they depend on the last estimate.

IV. BEARINGS ONLY TARGET MOTION ANALYSIS - NONMANEUVERING TARGET

A. PROBLEM DEFINITION

The problem considered here is that of estimating the position and velocity of a target, in two dimensions, by processing passively obtained bearing measurements.

The main application area is the Antisubmarine Warfare area where either a surface ship tries to locate a submarine through its cavitation noise or sonar transmissions, or vice versa.

In the following discussion we will consider a moving observer (own ship) that monitors noisy sonar bearings to an acoustic source (target), and processes these measurements to obtain estimates of target position and velocity.The geometric configuration is shown in Figure 1.1.

B. FORMULATIONS OF THE PROBLEM

As it was mentioned earlier the problem contains nonlinearities, and the linear Kalman filter is not applicable. Depending on the selection of the working coordinate system the nonlinear term may be either the state equation or the measurement equation. Even models with mixed elements from different coordinate systems have been used. Following are the most commonly used formulations of the problem:

1. Modified Polar Coordinates

In the modified polar (MP) coordinates the state vector is comprised of the following components:

- Bearing
- Bearing rate
- Range rate divided by range

. The reciprocal of range.

In this case the measurement equation is linear and the state equation nonlinear. The nonlinearities exhibited by the state equations are considerably more complicated than those exhibited by a formulation where the measurement equation is the nonlinear. Consequently the computational load for this formulation is increased. Details about the modified polar coordinates formulation can be found in [Ref. 6].

2. Mixed Coordinates

In this case as in the previous one the measurement equation is the linear one and the state equation non linear. The state vector consists of:

. Bearing
. Range
. Velocity component in x-direction
. Velocity component in y-direction

Again in this formulation there is the same complexity in linearizing the state equation as well as computational load. Analysis of the mixed-coordinate formulation can be found in [Ref. 7].

Pseudo-Linear Formulation

This formulation involves replacing of the measured bearings with pseudo-linear measurement residuals, to decouple the covariance computations from the estimated solution. The attractive feature of this method is that it permits a solution to the problem via linear estimation techniques. This formulation is similar to the Cartesian formulation which will be discussed in the next subsection. How does it differ from the Cartesian formulation can be found in appendix D. More details on this formulation can be found in [Ref. 8].

4. Cartesian Coordinates Formulation

This is the traditional way of formulating the problem. The state vector consists of:

1. Range in x-direction
2. Velocity component in x-direction
3. Range in y-direction
4. Velocity component in y-direction.

The state equation is linear and the measurement equation is now the nonlinear part. However the exhibited nonlinearity is easily circumvented without complicated or lengthy computations as it will be shown in the next section.

Finally the cartesian coordinate formulation will be adapted in the following discusion mainly because of its simplicity.

C. DESCRIPTION OF THE FILTER IN CARTESIAN COORDINATES

1. Derivation of the State Equations

If we will consider the geometric configuration of Figure 1 and with the restriction of target and tracker being in the same horizontal plane, the Cartesian formulation state vector may contain relative ranges and relative velocities in X and Y directions. The state vector that will be followed in this analysis is:

$$\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix} = \begin{bmatrix} x(t) \\ v_x(t) \\ y(t) \\ v_y(t) \end{bmatrix} \tag{4.1}$$

with:

$$
\begin{array}{ll}
x(t) & x_t(t)-x_o(t) \\
v_x(t) & v_{xt}(t)-v_{xo}(t) \\
y(t) & y_t(t)-y_o(t) \\
v_y(t) & v_{yt}(t)-v_{yo}(t)
\end{array}
\qquad (4.2)
$$

where $x_t(t), y_t(t), v_{xt}(t), v_{yt}(t)$ are the target absolute components of position and velocity in X and Y directions, and $x_o(t), y_o(t), v_{xo}(t)$, and $v_{yo}(t)$ are the tracker absolute components of position and velocity. The linear differential equations of motion of the model are given by:

$$
\dot{\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{bmatrix}} = \begin{bmatrix} x_2(t) \\ a_x(t) \\ x_4(t) \\ a_y(t) \end{bmatrix} \qquad (4.3)
$$

with:

$$
\begin{bmatrix} a_x(t) \\ a_y(t) \end{bmatrix} = \begin{bmatrix} a_{xt}(t)-a_{xo}(t) \\ a_{yt}(t)-a_{yo}(t) \end{bmatrix} \qquad (4.4)
$$

where $a_x(t)$ and $a_y(t)$ are the relative accelerations in both directions, and $a_{xt}(t), a_{yt}(t), a_{xo}(t)$ and $a_{yo}(t)$ are the corresponding absolute accelerations of target and tracker in both directions correspondingly.

The solutions of the differential equations above in matrix notation give:

$$
x(t) = A(t,t0)x(t0) + u(t,t0) \qquad (4.5)
$$

with:

$$A_{(t,t_0)} = \begin{bmatrix} 1 & (t-t0) & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & (t-t0) \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.6}$$

and:

$$u_{(t,t_0)} = \begin{bmatrix} u_1\ (t,t0) \\ u_2\ (t,t0) \\ u_3\ (t,t0) \\ u_4\ (t,t0) \end{bmatrix} = \begin{bmatrix} \int_{t_0}^{t} (t-\lambda) a_x(\lambda) d\lambda \\ \int_{t_0}^{t} a_x(\lambda) d\lambda \\ \int_{t_0}^{t} (t-\lambda) a_y(\lambda) d\lambda \\ \int_{t_0}^{t} a_y(\lambda) d\lambda \end{bmatrix} \tag{4.7}$$

and (t0) denotes any arbitrary fixed value of time.

Although (4.5) is valid for unconstrained vehicle motion, solution requirements necessitate that the bearings-only target motion analysis be formulated under the restrictive assumption of constant target velocity. [Ref. 9]. In this case $a_{xt}(t)$ and $a_{yt}(t)$ become zero and u(t,t0) reduces to a deterministic input vector which depends only upon the tracker's acceleration (maneuvers). So

$$u(t,t0) = -u_0(t,t0) \tag{4.8}$$

where:

$$u_0(t,t_0) = \begin{bmatrix} u_{01}(t,t0) \\ u_{02}(t,t0) \\ u_{03}(t,t0) \\ u_{04}(t,t0) \end{bmatrix} = \begin{bmatrix} \int_{t0}^{t}(t-\lambda)a_{x0}(\lambda)d\lambda \\ \int_{t0}^{t}a_{x0}(\lambda)d\lambda \\ \int_{t0}^{t}(t-\lambda)a_{y0}(\lambda)d\lambda \\ \int_{t0}^{t}a_{y0}(\lambda)d\lambda \end{bmatrix} \quad (4.9)$$

2. <u>Derivation of the Measurement Equation</u>

The measurement process is described by a scalar time varying equation of the form:

$$\beta(t) = h[x(t)] + n(t) \quad (4.10)$$

where

$$h[x(t)] = \arctan x_3(t)/x_1(t) \quad (4.11)$$

and $\beta(t)$ represents the measured target bearing corrupted by additive measurement noise $n(t)$. It is assumed that $n(t)$ is a white noise with zero mean and known variance σ^2, i.e.,

$$E[n(t)] = 0 \quad (4.12)$$

and

$$E[n(t)n(t+\lambda)] = \begin{cases} \sigma^2 & \lambda = 0 \\ 0 & \lambda \neq 0 \end{cases} \quad (4.13)$$

D. THE DISCRETE TIME MODEL

The previously defined model in discrete form is described by:

$$\hat{x}(k+1) = A(k)\hat{x}(k) - u(k) \tag{4.14}$$

and

$$\beta(k) = h[x(k)] + n(k) \tag{4.15}$$

where:

$\hat{x}(k)$ is the (4x1) state vector consisted from relative range and velocity of the target in X and Y directions.

$A(k)$ is the (4x4) state transition matrix which is constant and given by:

$$A(k) = \begin{bmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.16}$$

$u(k)$ is the (4x1) vector of deterministic inputs due to tracker's movement and given by Equation (4.9).

$\beta(k)$ is the scalar noisy bearing measurement taken at time $t(k)$.

$n(k)$ is the scalar additive measurement noise at time $t(k)$.

Equation (4.14) assumes that the target moves with zero acceleration, (non maneuvering). Also it is assumed that the additive measurement noise $n(k)$ has zero mean and a known variance $\sigma^2(k)$. Finally an initial estimate of the state vector and its error covariance matrix is presumed to be given.

The extended Kalman filter technique is applied to the problem and yields:

$\hat{x}(0/0)$ is the initial estimate of the state vector which is considered to be given.

$P(0/0)$ is the initial estimate error covariance matrix which is also considered to be given.

$$\hat{x}(k/k-1) = A(k)\hat{x}(k-1/k-1)-w(k) \tag{4.17}$$

is the projection ahead of the estimated state vector.

$$P(k/k-1) = A(k)P(k-1/k-1)A'(k) + Q(k) \tag{4.18}$$

is the projection of the error covariance matrix and $Q(k)$ is the maneuver excitation covariance matrix (assumed zero if the target does not maneuver).

$$H(k) = \left.\frac{\partial h}{\partial x}\right|_{x=\hat{x}(k/k-1)} \tag{4.19}$$

is a (1x4) matrix given by:

$$H(k)= \left.[x_3/(x_1^2+x_3^2),0,-x_1/(x_1^2+x_3^2),0]\right|_{x=\hat{x}(k/k-1)} \tag{4.20}$$

$$G(k)=P(k/k-1)H'(k)[H(k)P(k/k-1)H'(k)+\sigma^2(k)]^{-1} \tag{4.21}$$

is the gain equation

$$\hat{x}(k/k) = \hat{x}(k/k-1)+G(k)[\beta(k)-h\hat{x}(k/k-1)] \tag{4.22}$$

is the update equation and

$$P(k/k) = [I-G(k)H(k)]P(k/k-1) \tag{4.23}$$

is the error covariance update equation.

The above algorithm was formulated in computer simulation program as in Appendix B and tested for the situations shown in Figure 4.1 and Figure 4.2.

In the first case (Figure 4.1),the target was moving from east to west on a constant course and speed of 20 m/sec, while the tracker was maneuvering following a sinusodial track with main course from east to west also, and a velocity of 10 m/sec in x-direction. The target had a relative velocity of 10 m/sec with respect the tracker in the X-axis and 0 m/sec in the Y-axis.

In the second case (Figure 4.2), the target was moving as the first case but the tracker was following a circular path of radius 2000 m with a turning rate of 2°/sec.

The measurement error was taken as zero mean and 0.1 covariance and the measurement interval 1 sec. The behavior of the filter is displayed in the following Figures and is considered to be satisfactory.

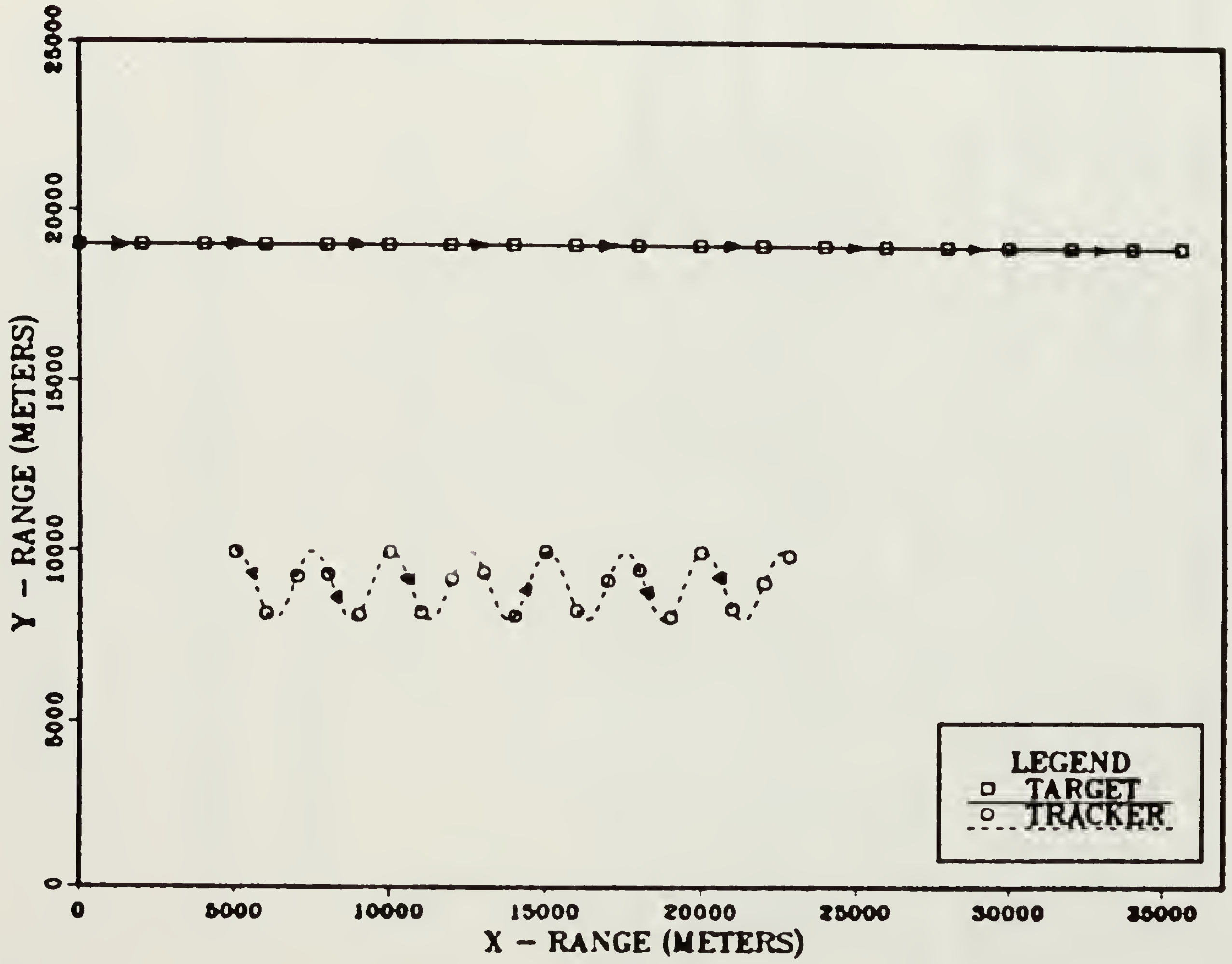

Figure 4.1 Nonmaneuvering Target. Case 1.

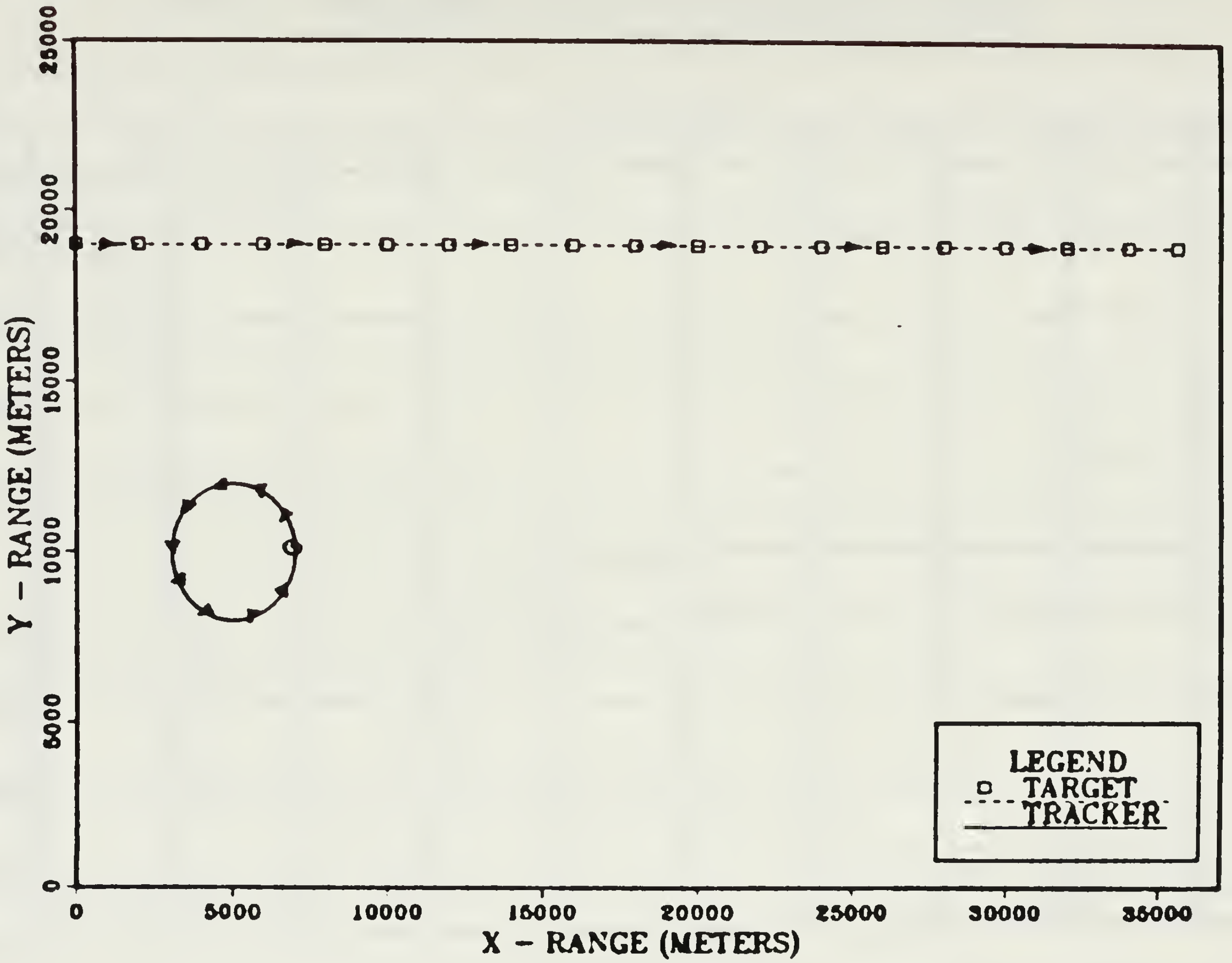

Figure 4.2 Nonmaneuvering Target. Case 2.

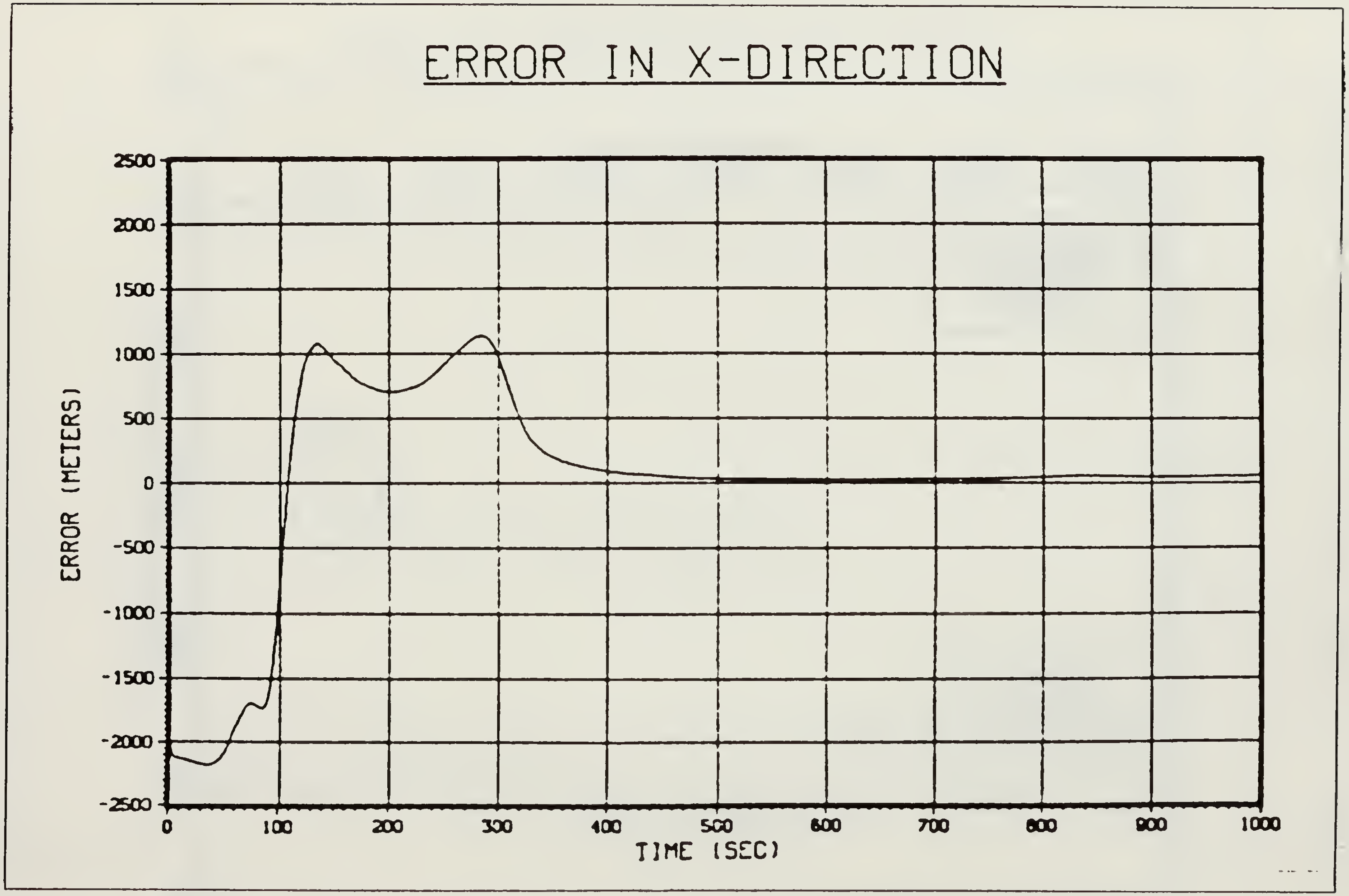

Figure 4.3 Range Error in X-Direction Case 1.

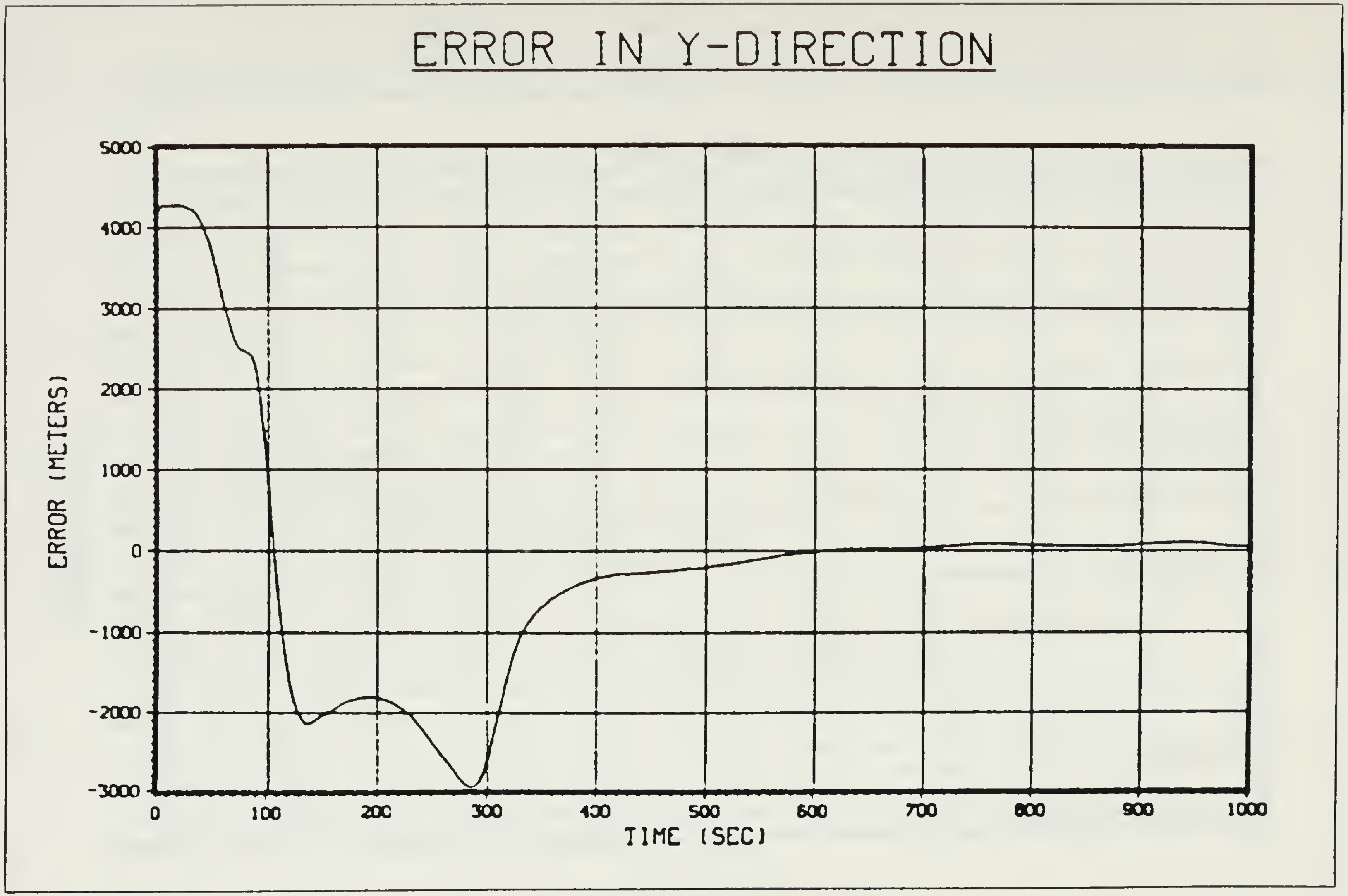

Figure 4.4 Range Error in Y-Direction Case 1.

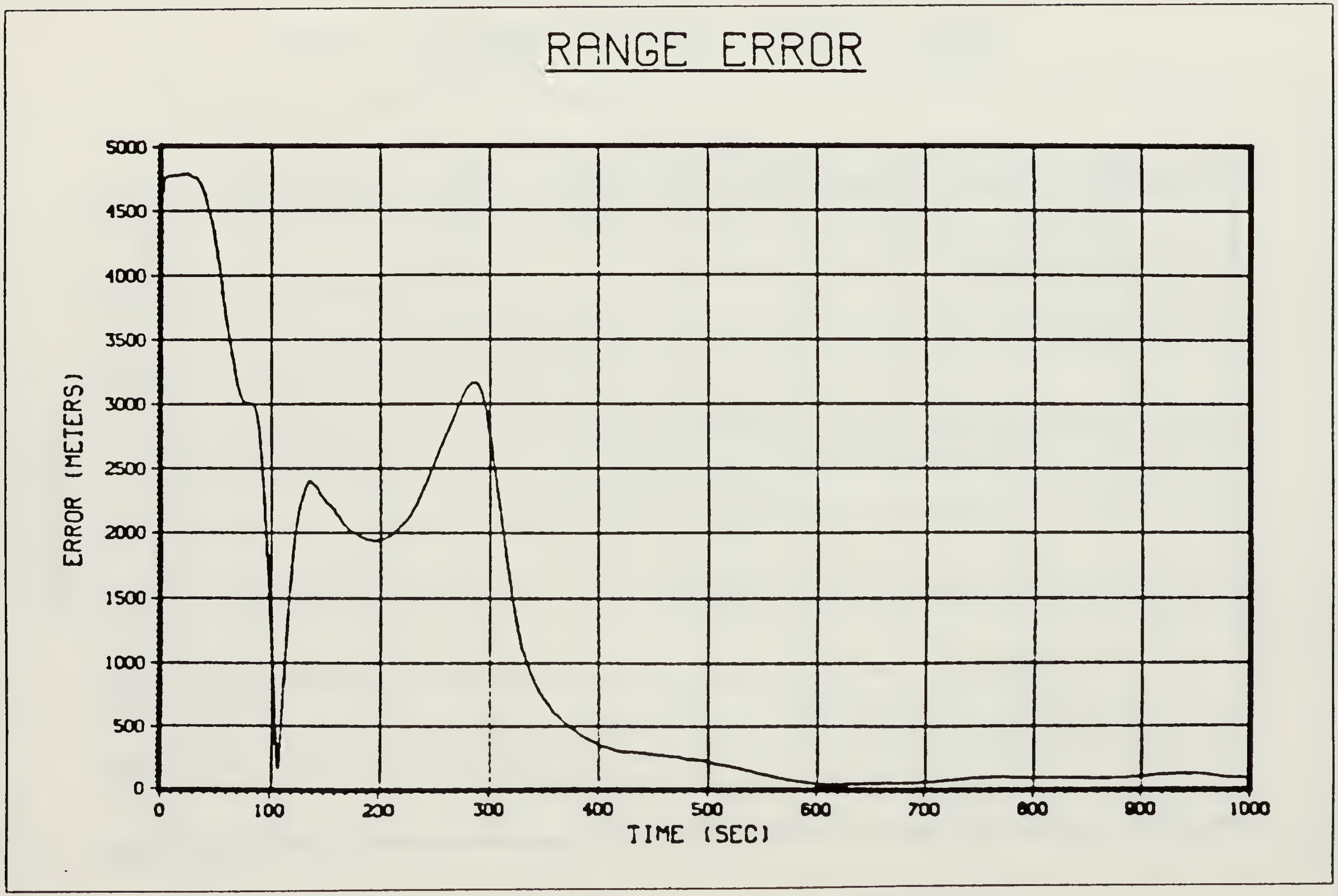

Figure 4.5   Range Error Case 1.

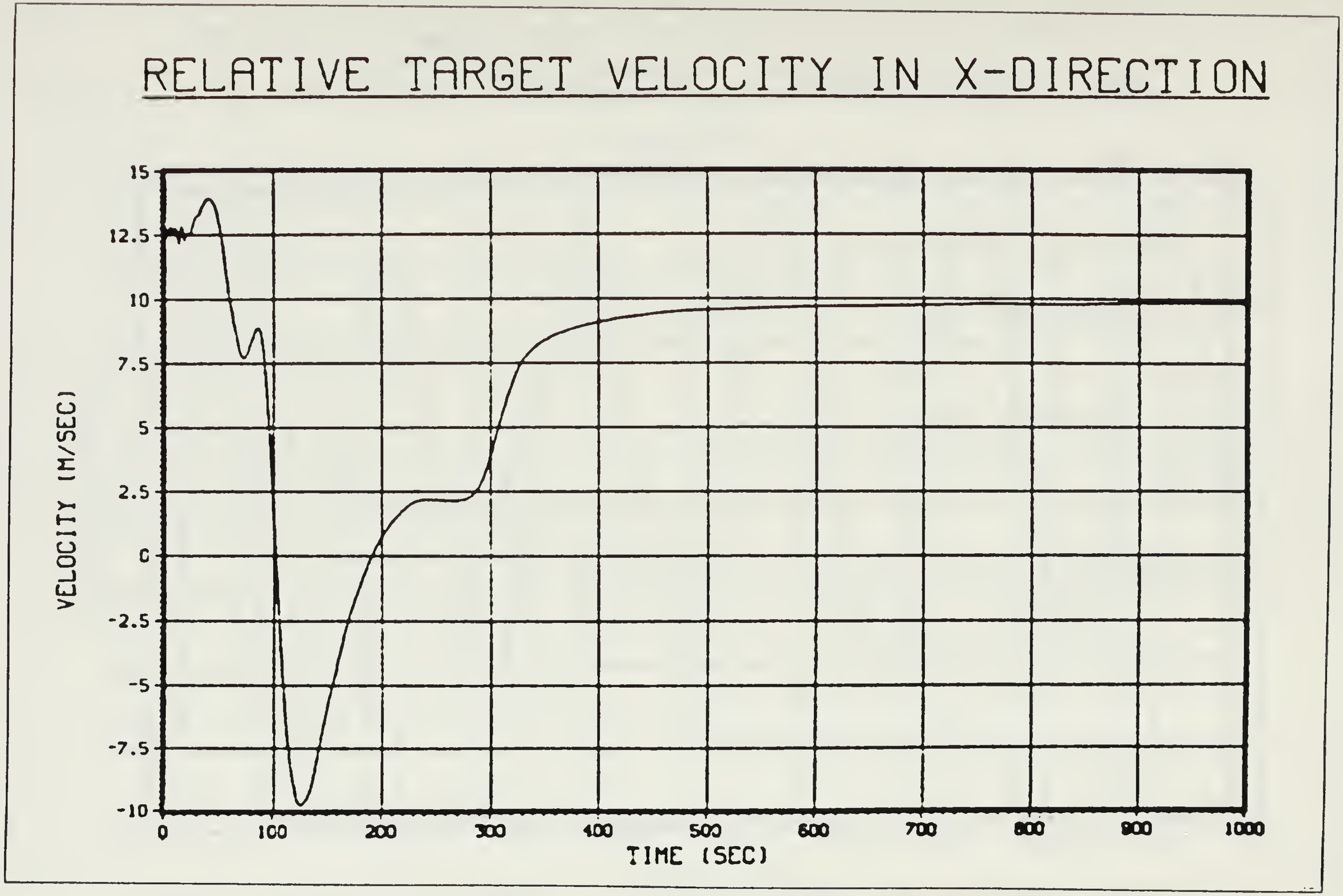

Figure 4.6 Relative Target Velocity in X-Direction Case 1.

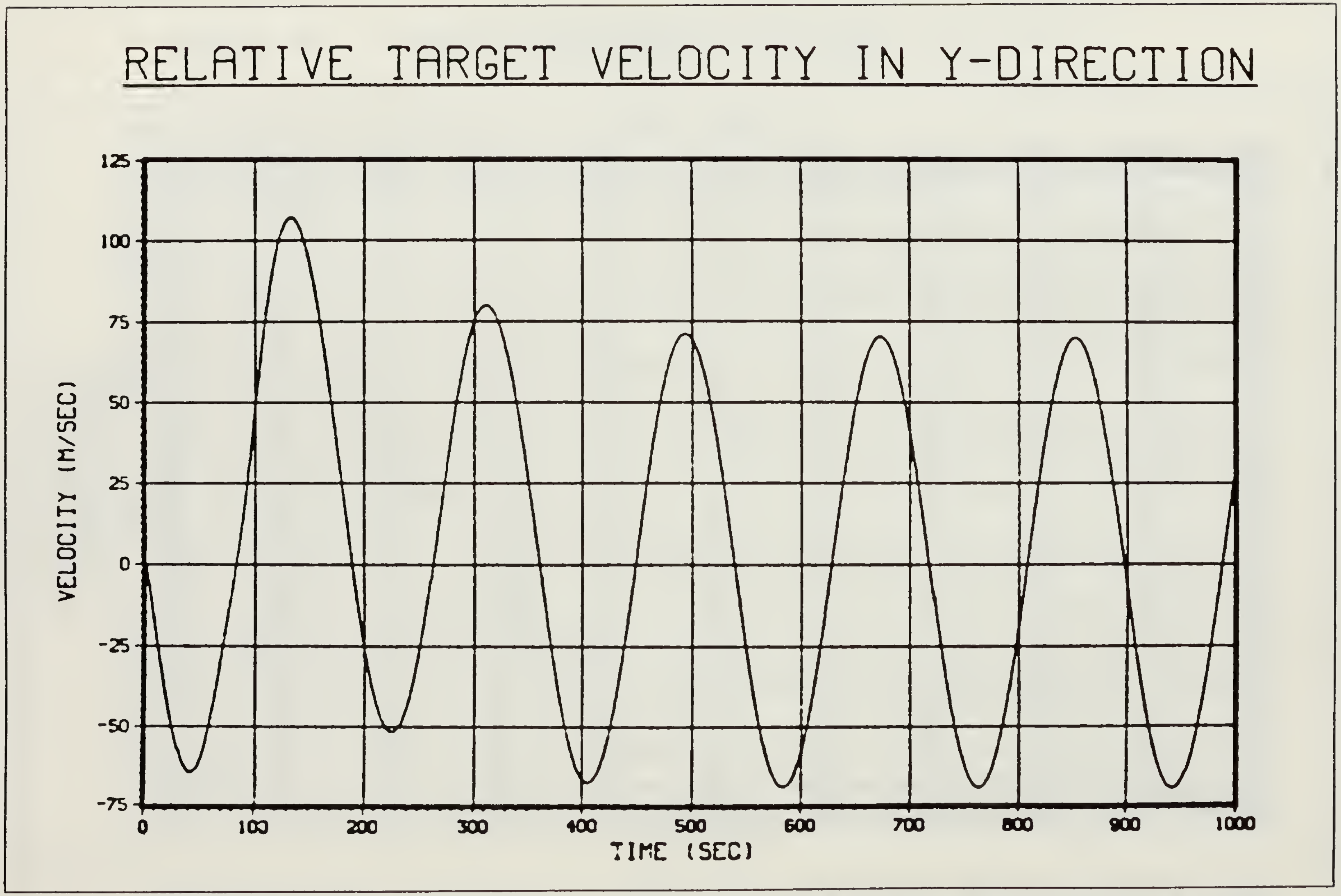

Figure 4.7 Relative Target Velocity in Y-Direction Case 1.

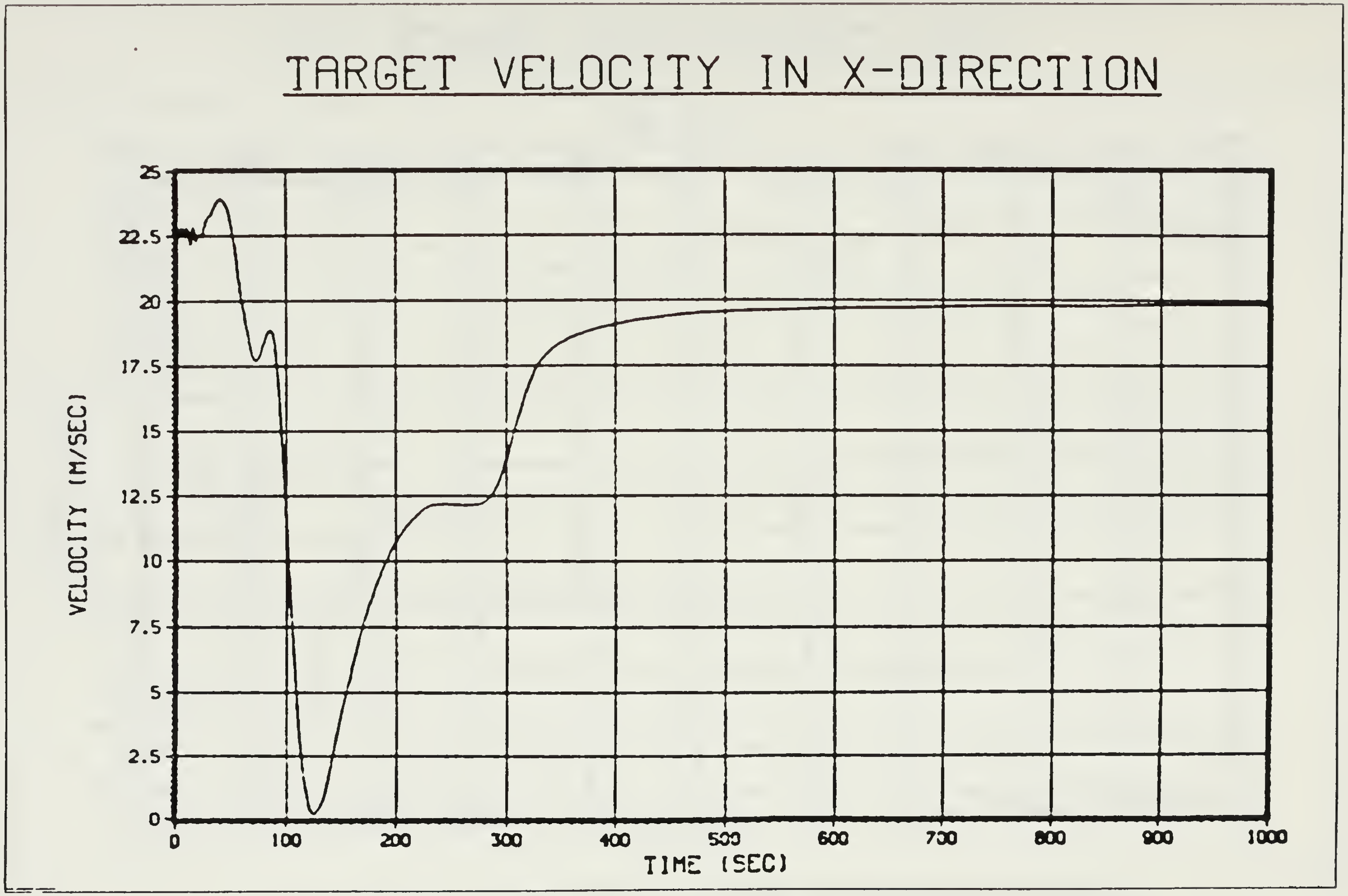

Figure 4.8 Target Velocity in X-Direction Case 1.

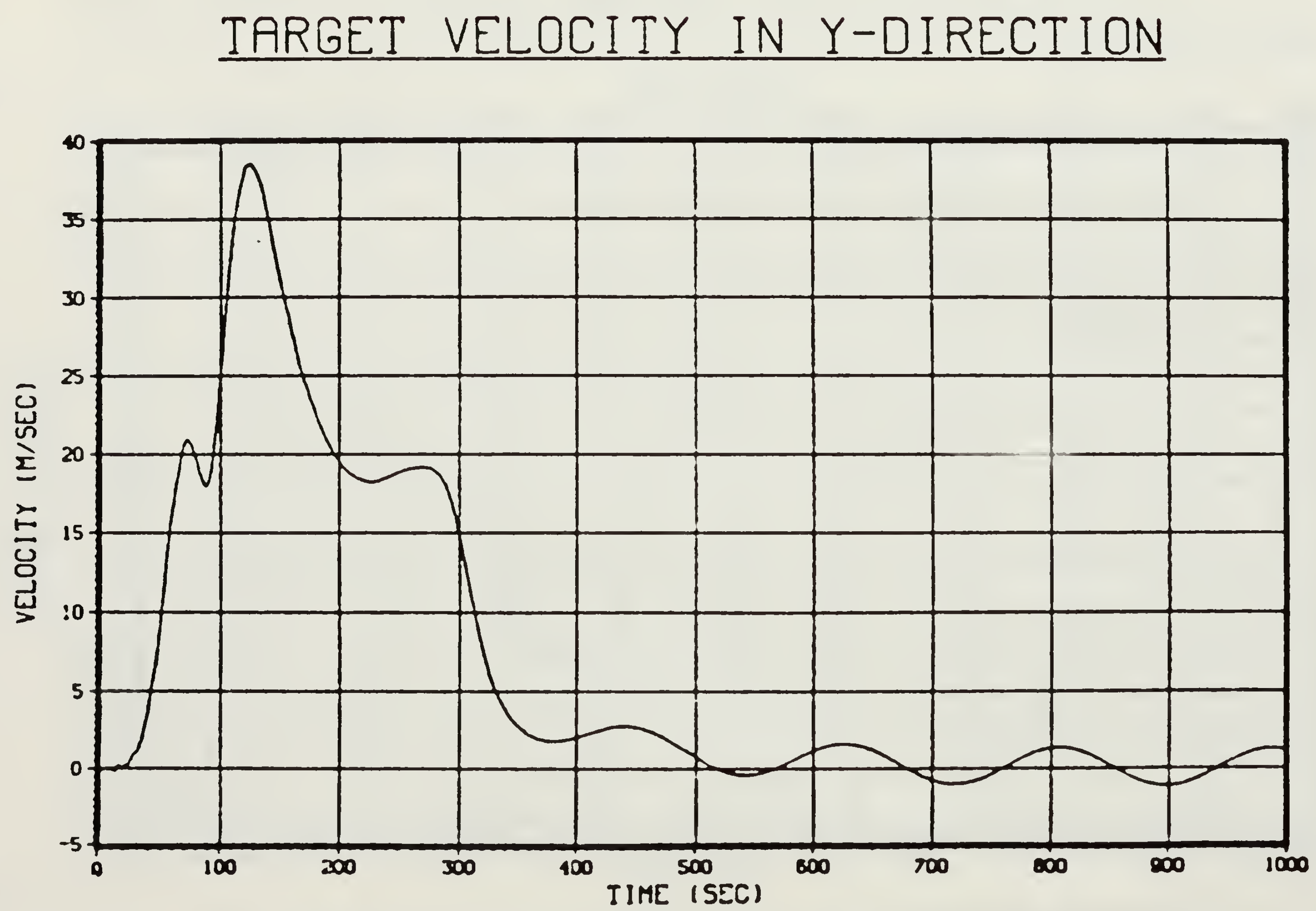

Figure 4.9 Target Velocity in Y-Direction Case 1.

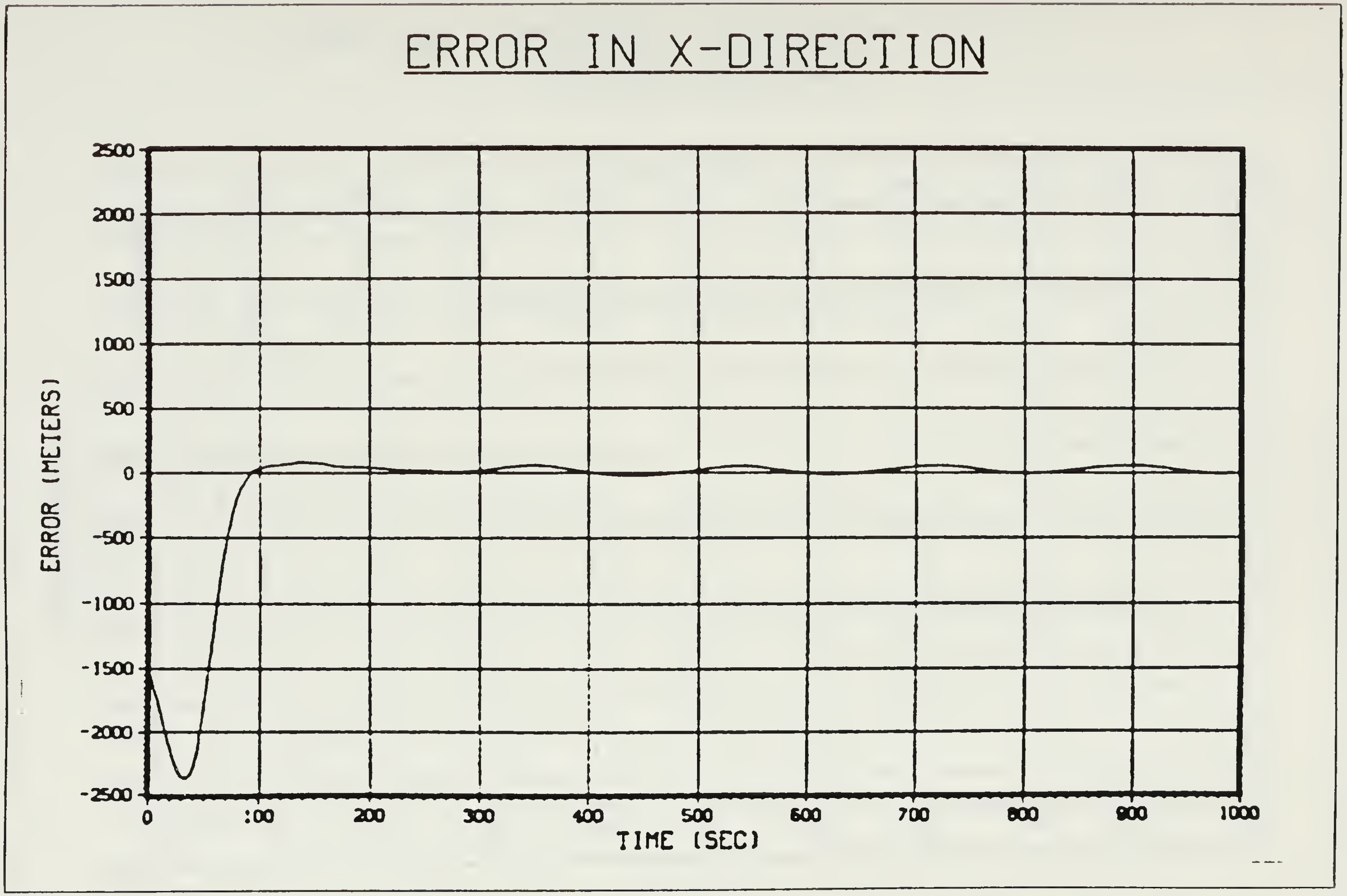

Figure 4.10 Range Error in X-Direction Case 2.

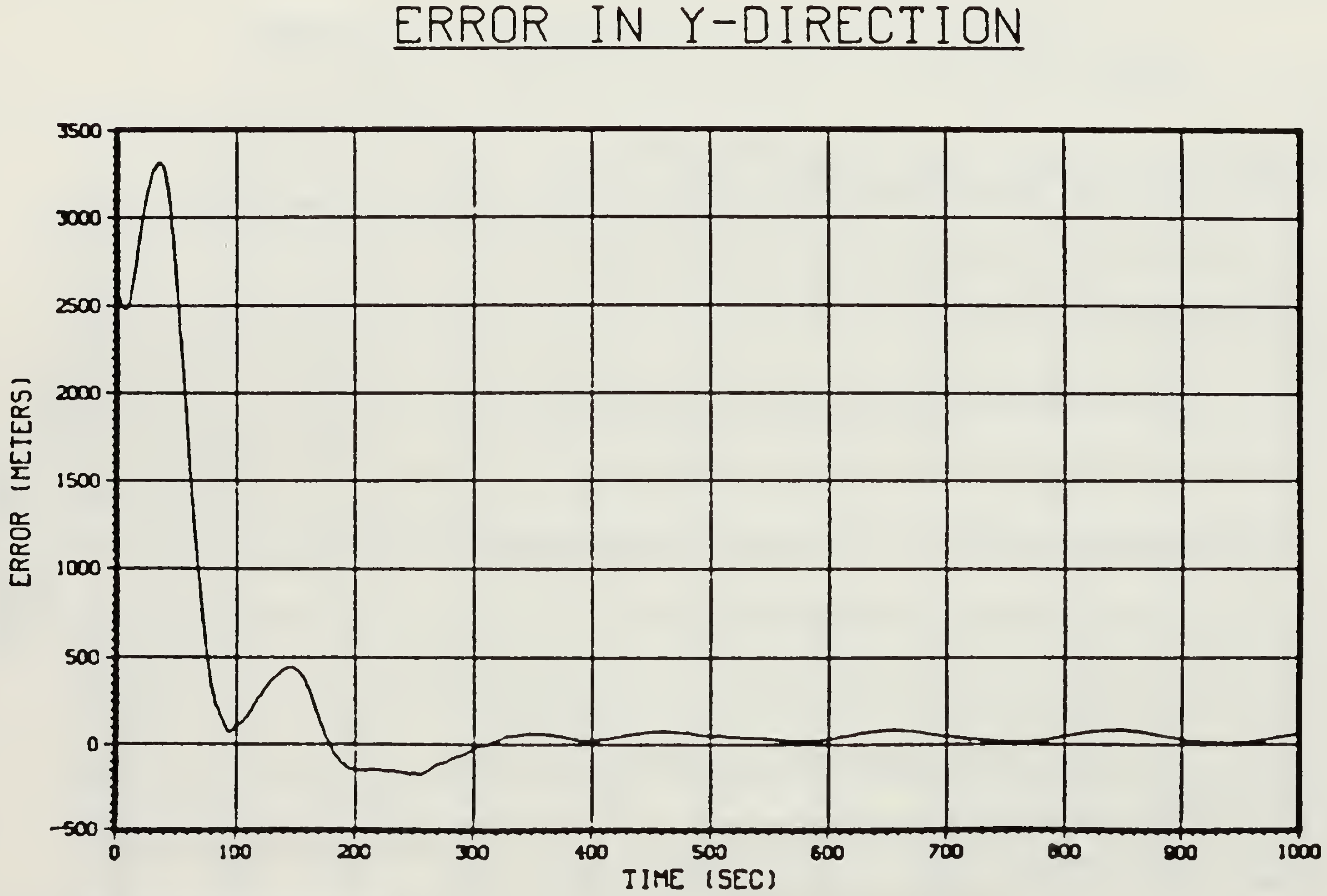

Figure 4.11 Range Error in Y-Direction Case 2.

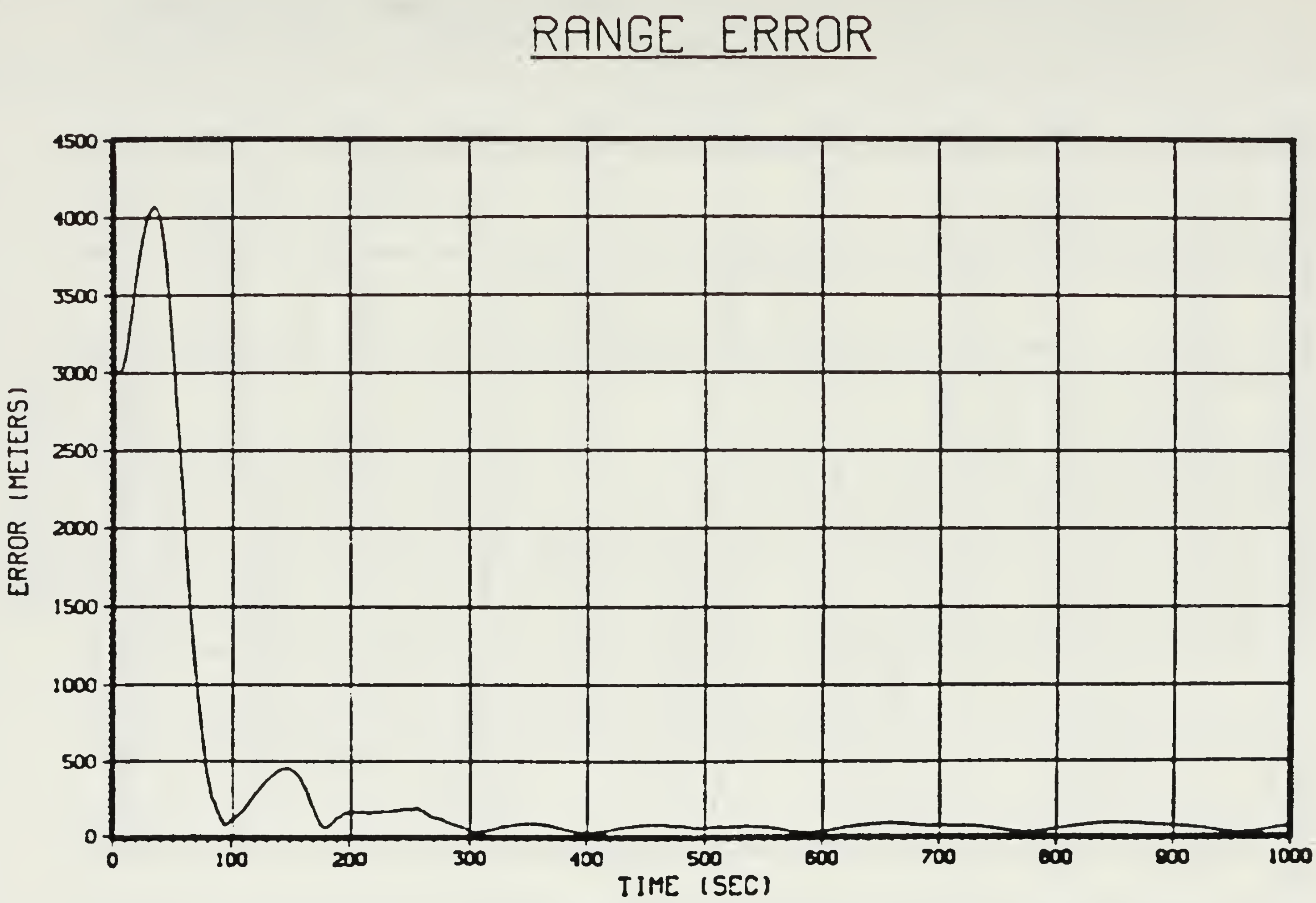

Figure 4.12 Range Error Case 2.

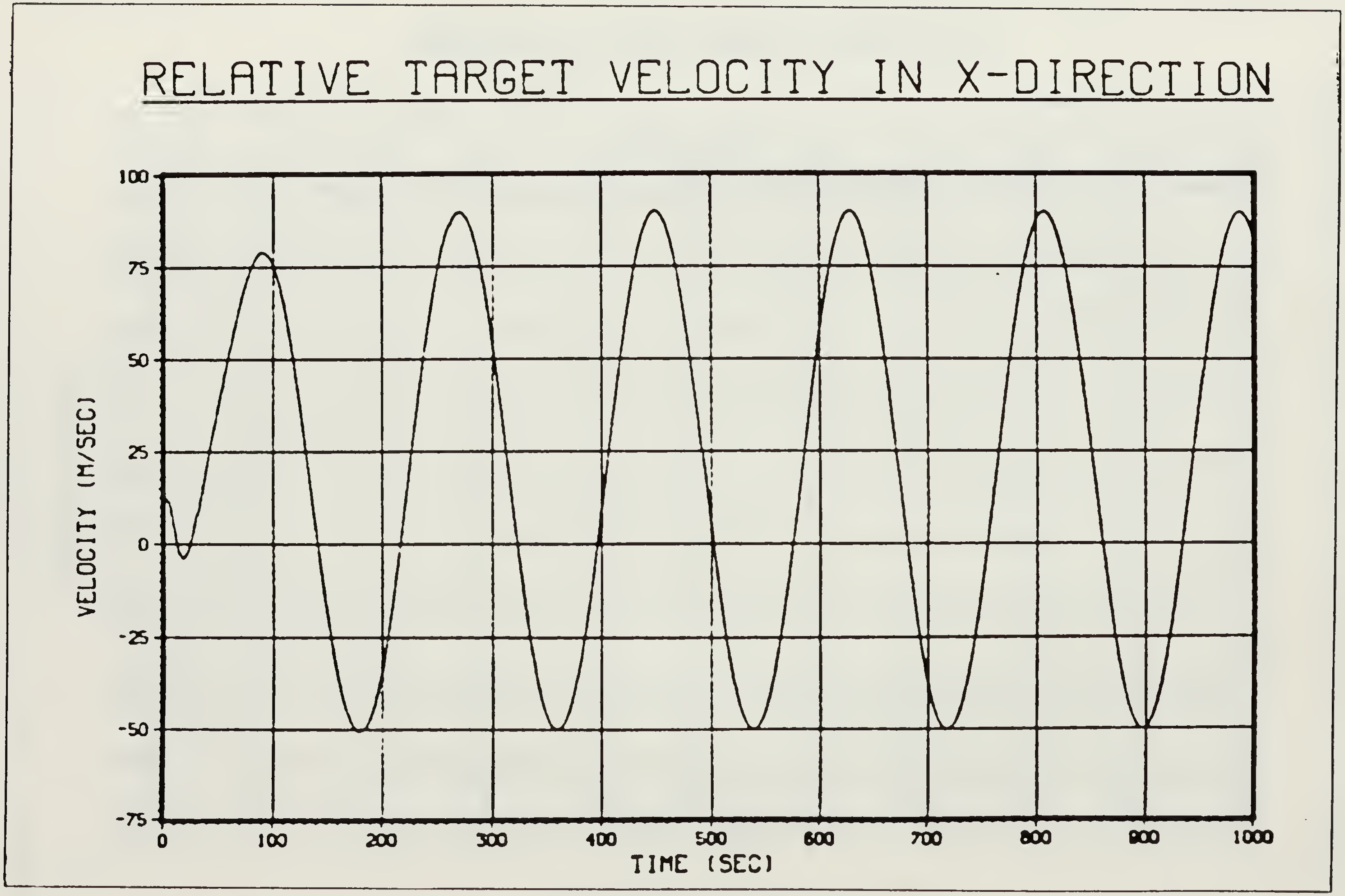

Figure 4.13 Relative Target Velocity in X-Direction Case 2.

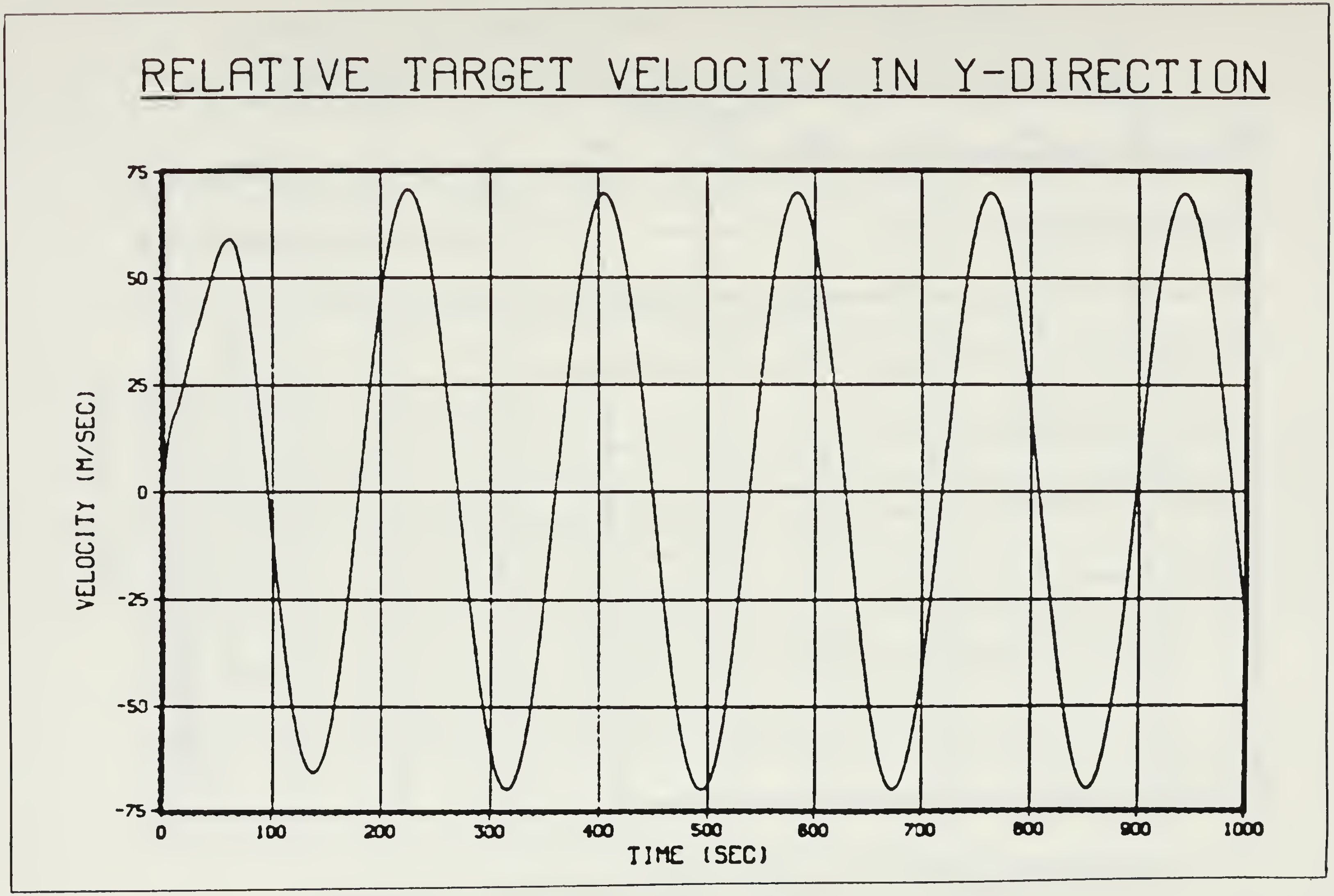

Figure 4.14 Relative Target Velocity in Y-Direction Case 2.

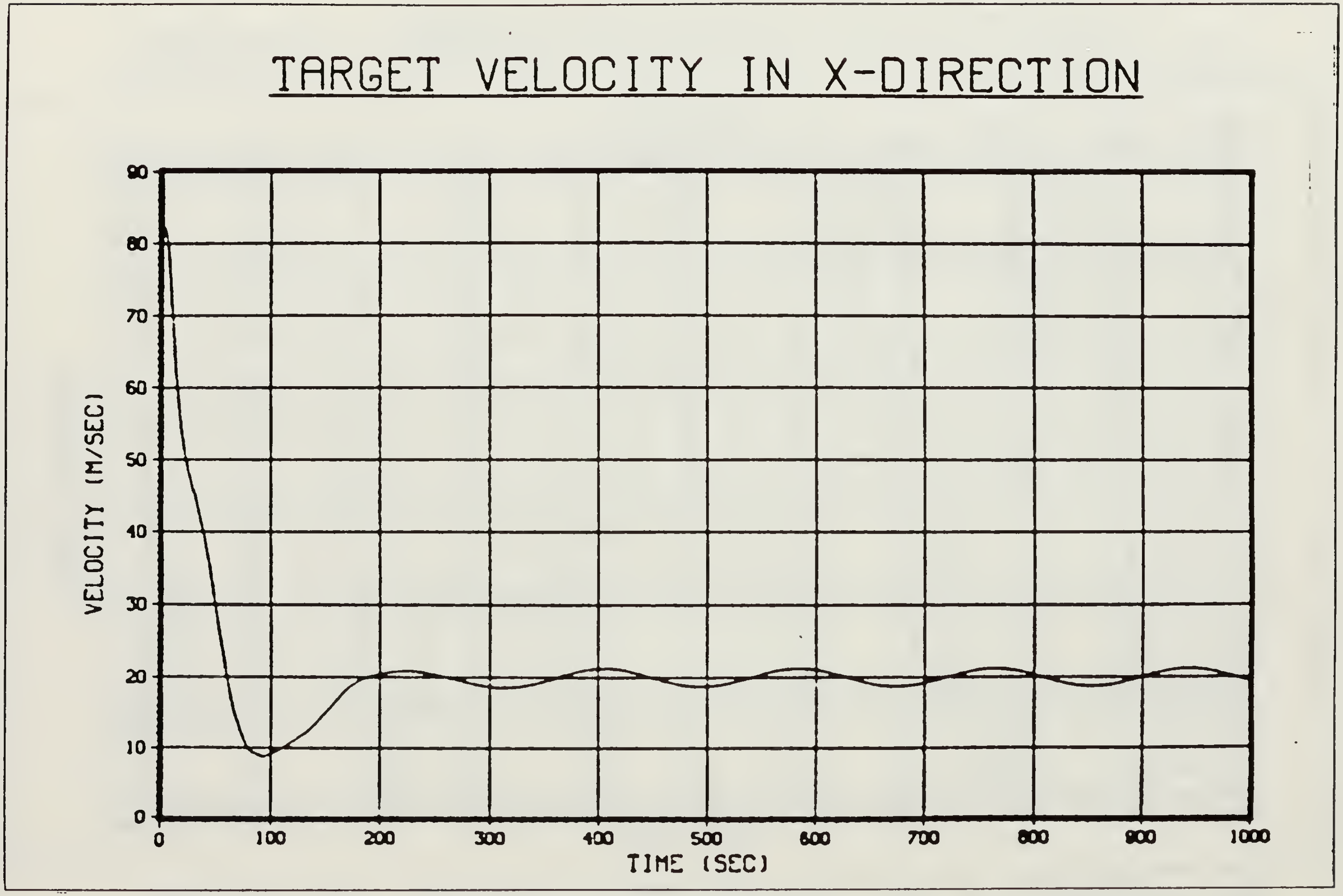

Figure 4.15 Target Velocity in X-Direction Case 2.

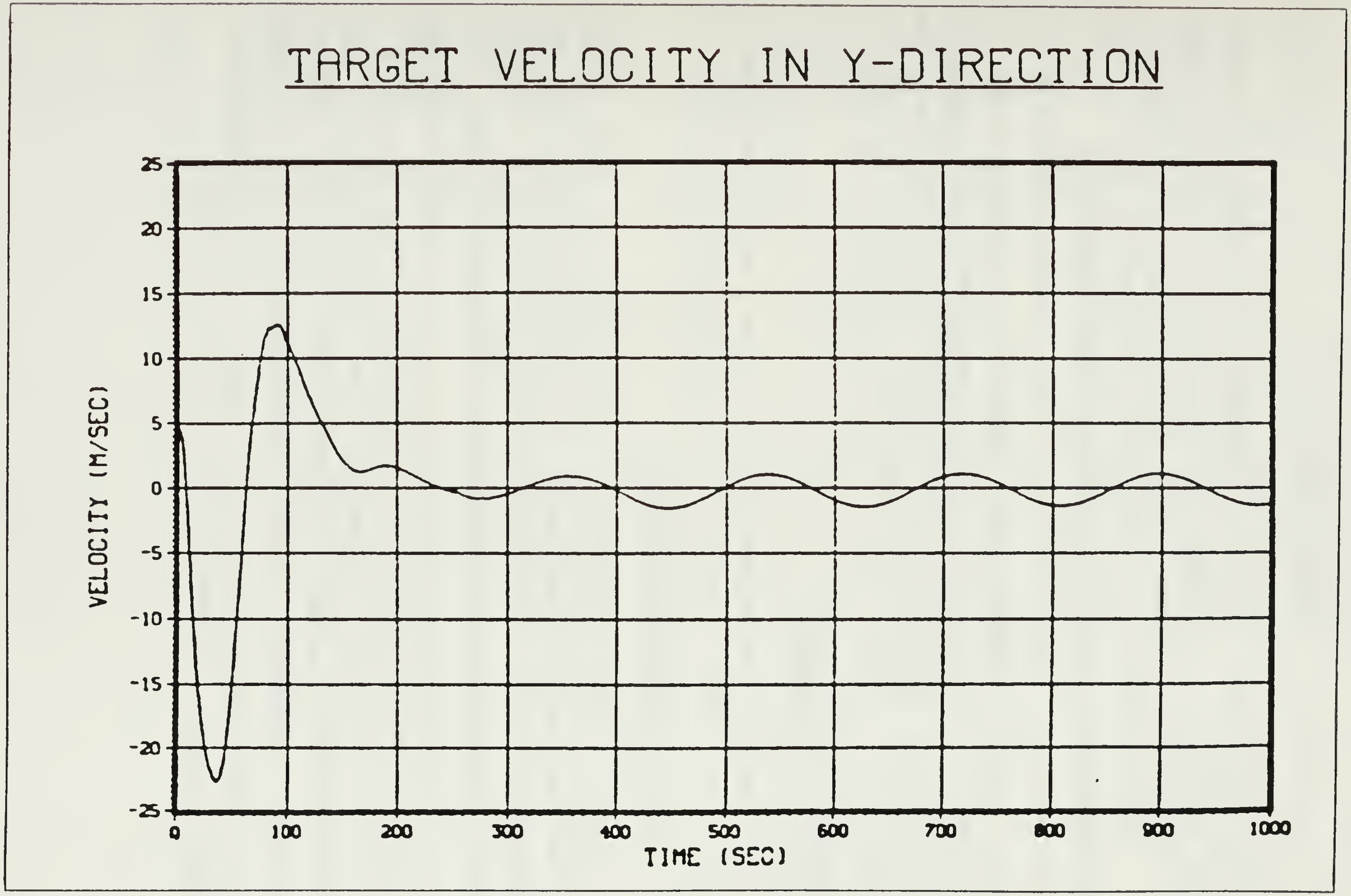

Figure 4.16 Target Velocity in Y-Direction Case 2.

V. MANEUVERING TARGET

Up to this time we made the assumption that the target does not maneuver. However in real world applications this is not the usual case and hence the assumption is unreasonable. Specifically, in the main application area of the bearings-only tracking, i.e., in the A.S.W scene, it is expected that the target will not keep constant velocity but instead it will command some kind of zig-sag during the normal open sea transit and strong maneuvering or evasion after detection of a potential threat. It is evident thus that there is a need to accommodate the maneuvering case.

A. POSSIBLE APPROACHES

There are various approaches relative to the problem in general. Some found in the literature are following:

1. Variable Dimension Filter

In this case, the filter operates in its normal mode in the absence of any maneuvers. A detection scheme is used to determine that a maneuver is indeed occuring. Once a maneuver is detected, a different state model is used. The extent of the maneuver as detected is then used to yield an estimate for the extra state components. The tracking is then continued with the augmented state model until it will be reverted to the normal model by another decision. The two models are a constant velocity and a constant acceleration model for the maneuvering case. Details on the analysis of that method can be found in [Ref. 10].

2. Expanded Number of States

In this case the model includes the acceleration component in it. This method has the disadvantage that if the target does not have acceleration, using a third order model increases the estimation errors for both position and velocity [Ref. 10]. Also,the computational load increases drastically by augmenting the model by one term.

3. Modeling Target Acceleration as Random Process of Known Form

This method is based on the fact that the target acceleration and thus the target maneuver, is correlated in time; i.e., if the target is accelerating at time t, it is likely to be accelerating at time t+&tau for sufficiently small τ. A typical representative model of the correlation function r( ) associated with the target acceleration is given by:

$$r(\tau) = E\,[a(t)a(t+\tau)] = \sigma^2 e^{-a|\tau|}, a \geq 0 \qquad (5.1)$$

where (σ^2) is the variance of the target acceleration and (a) is the reciprocal of the maneuver time constant. The maneuver excitation covariance matrix Q(k) then depends on the correlation function $r(\tau)$,which also depends on the type of the target.

The above formulation includes the acceleration term in the state vector. So the performance of the filter is degraded by the computational overhead. The quality of the estimate is also degraded when the target is moving with constant velocity.

Analysis of the above method can be found at [Ref. 11].

4. Use Variable Maneuver Excitation Error Covariance Matrix

The filter is modeled as a second order and it does not include acceleration term in it. The idea is [Ref. 12]. to use a set of different values for the forcing input covariance Q(k). The filter monitors the innovation error in the equation:

$$\hat{x}(k/k) = \hat{x}(k/k-1) + G(k)[\beta(k) - h(\hat{x}(k/k-1))] \qquad (5.2)$$

i.e., the term $[\beta(k)-h(\hat{x}(k/k-1))]$ in every iteration. If that error becomes larger than a predetermined threshold, that means that the received bearing measurement does not agree with that the filter generated and was supposed to receive. Correspondingly, the estimated vector does not agree with the actual. So the filter assumes that the target made a maneuver. Depending on the size of the innovation error, a value for the excitation covariance matrix Q(k) is applied to the error covariance propagation equation:

$$P(k/k-1) = A(k)P(k/k-1)A'(k) + Q(k) \qquad (5.3)$$

The effect of the above is to increase the uncertainty of the filter which consequently causes an increase of the gain G(k). The bigger the G(k) the more the filter "believes" the measurements rather than the previous estimates.

So the filter is "partially" reinitialized. By partially is meant that the new initial estimates of the state vector and specifically the range terms are very close to the real ones estimated just before the maneuver. Thus the filter has good conditions to start over and estimate the new state after the maneuver.

B. BEARINGS-ONLY TRACKING WITH MANEUVERING TARGET.

From the previously mentioned methods of dealing with maneuvering targets, we are going to develop the last one i.e., that of using a variable maneuver excitation error covariance matrix Q(k).

This method uses a four-state model, so it is faster than the others using a six-state models and is the simplest of all. Actually only a few extra lines of program are added to that of a nonmaneuvering target.

1. Determination of the Q(k) Matrix

If we will suppose that the target made a maneuver, (acceleration (a)) during the state propagation time from (k) to (k+1), in one direction say X, then the error introduced to our propagated estimate in the range term will be $(1/2)aT^2$ and the error introduced in the velocity term will be aT. Combining that fact in both direction and with the assumption that an acceleration in X is uncorrelated to an acceleration in Y the resulting Q(k) is given by:

$$Q(k) = \Gamma(k)E[w(k)w'(k)]\Gamma'(k) \qquad (5.4)$$

where:

$$\Gamma(k) = \begin{bmatrix} \frac{T^2}{2} \\ T \\ 0 \\ 0 \end{bmatrix} \qquad (5.5)$$

2. Simulation Results for Cases 3 and 4 with $w=1, \sigma^2=0.1$

The above method was modeled and simulated in the computer. Two geometric configurations of target and tracker as shown in Figures (5.1) and (5.2) were tested.

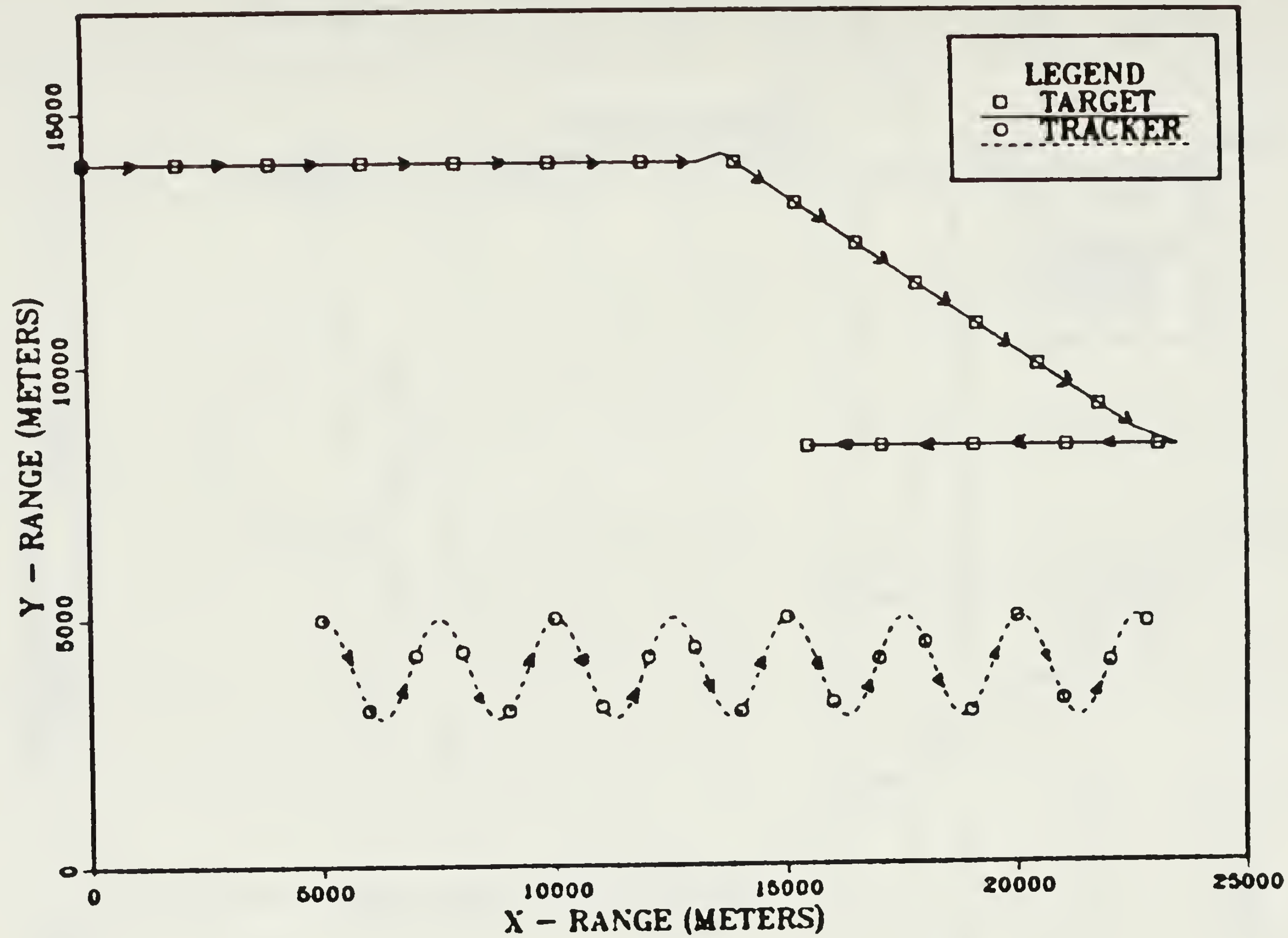

Figure 5.1   Maneuvering Target. Case 3.

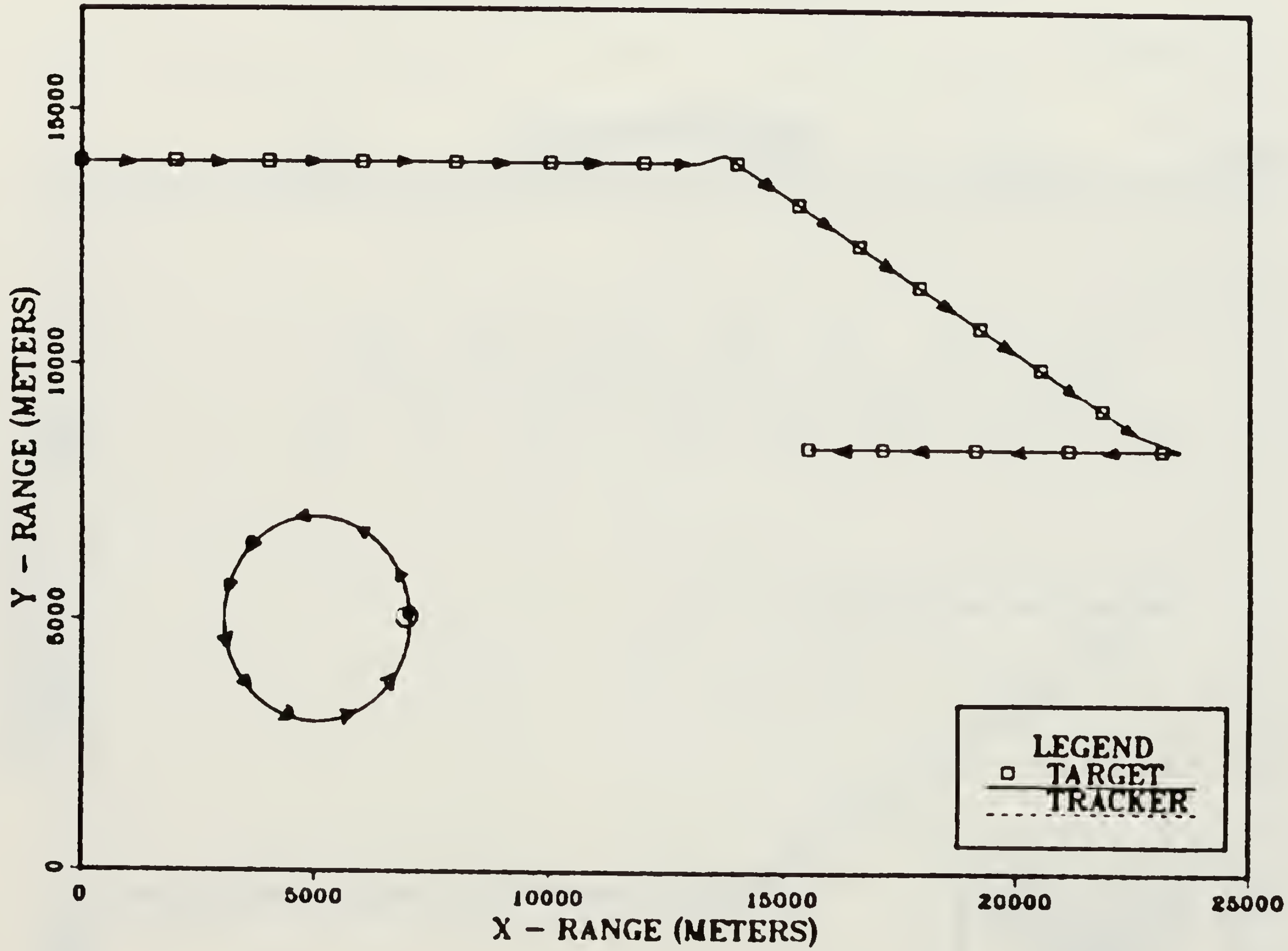

Figure 5.2 Maneuvering Target. Case 4.

In the case 3 the target was following a steady course from east to west and a constant speed of 20 m/sec. At the 700th second it changed course to the right and speed components to 12 m/sec in X and -10 m/sec in Y direction. After another 700 seconds it changed course again that time from west to east and resumed a speed of 20 m/sec. The tracker was moving as in case 1 of the nonmaneuvering target. The intervals displayed in the Figures (5.1) and (5.2) correspond to time intervals of 100 seconds.

In the case 4 the target was following the same track as in case 3 but that time the tracker was maneuvering as in case 2.

In the following simulations the measurement error was supposed to have zero mean and 0.1 variance. The measurement interval was again taken as 1 sec which is also considered as reasonable for a real application.The (W) was taken equal to 1.0.

The simulation program (Appendix C) for the above conditions gave the results shown in the following Figures 5.3 to 5.16. In both cases the filter detected the maneuver and very rapidly after approximately 300 seconds estimated the new target parameters. The fluctuations of the errors due to the target maneuver are smaller than those during the first initialization of the problem. This can be explained by the fact that after the target maneuver detection the filter had an "accurate" reinitialization state from the previous tracking.

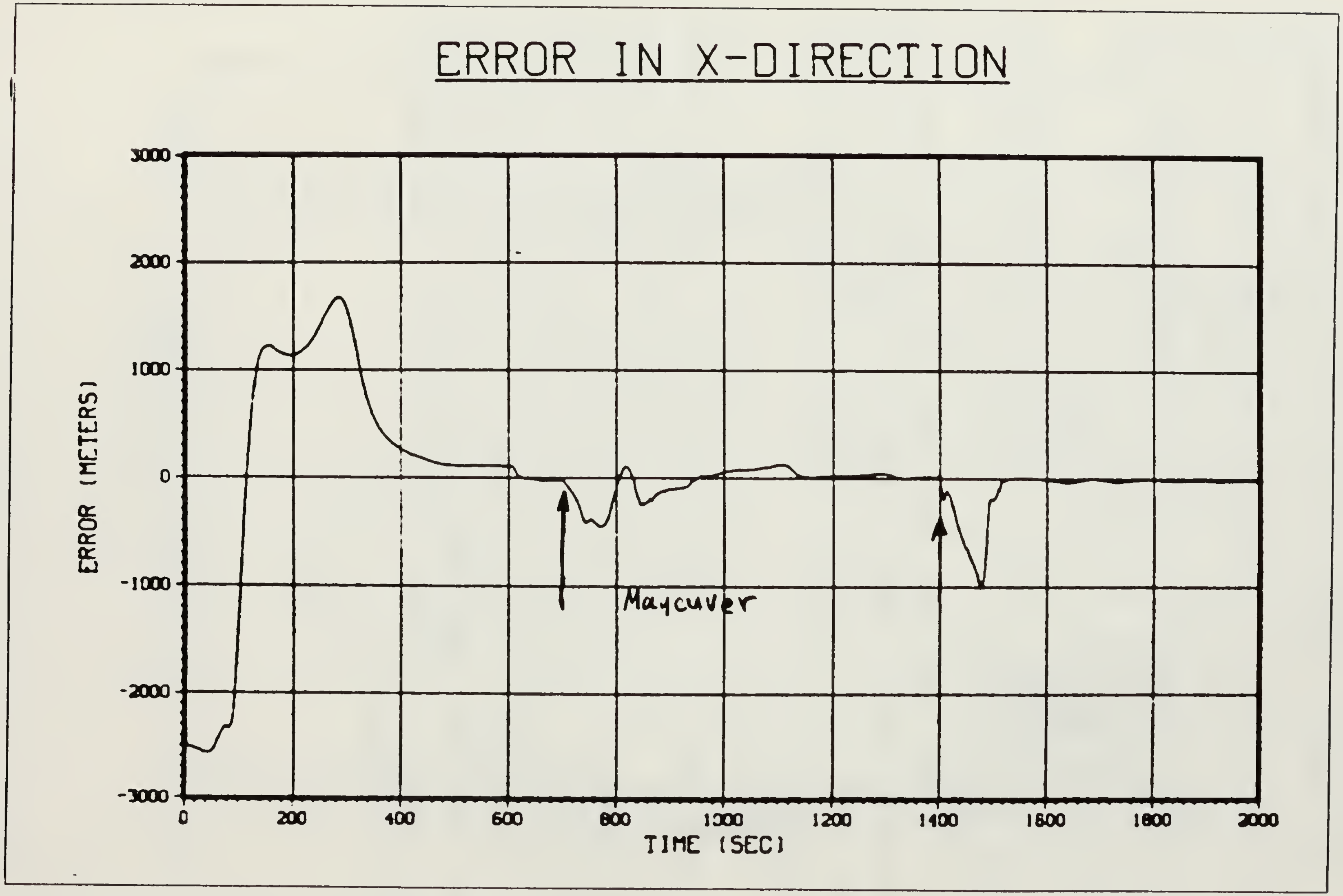

Figure 5.3 Range Error in X-Direction Case 3.

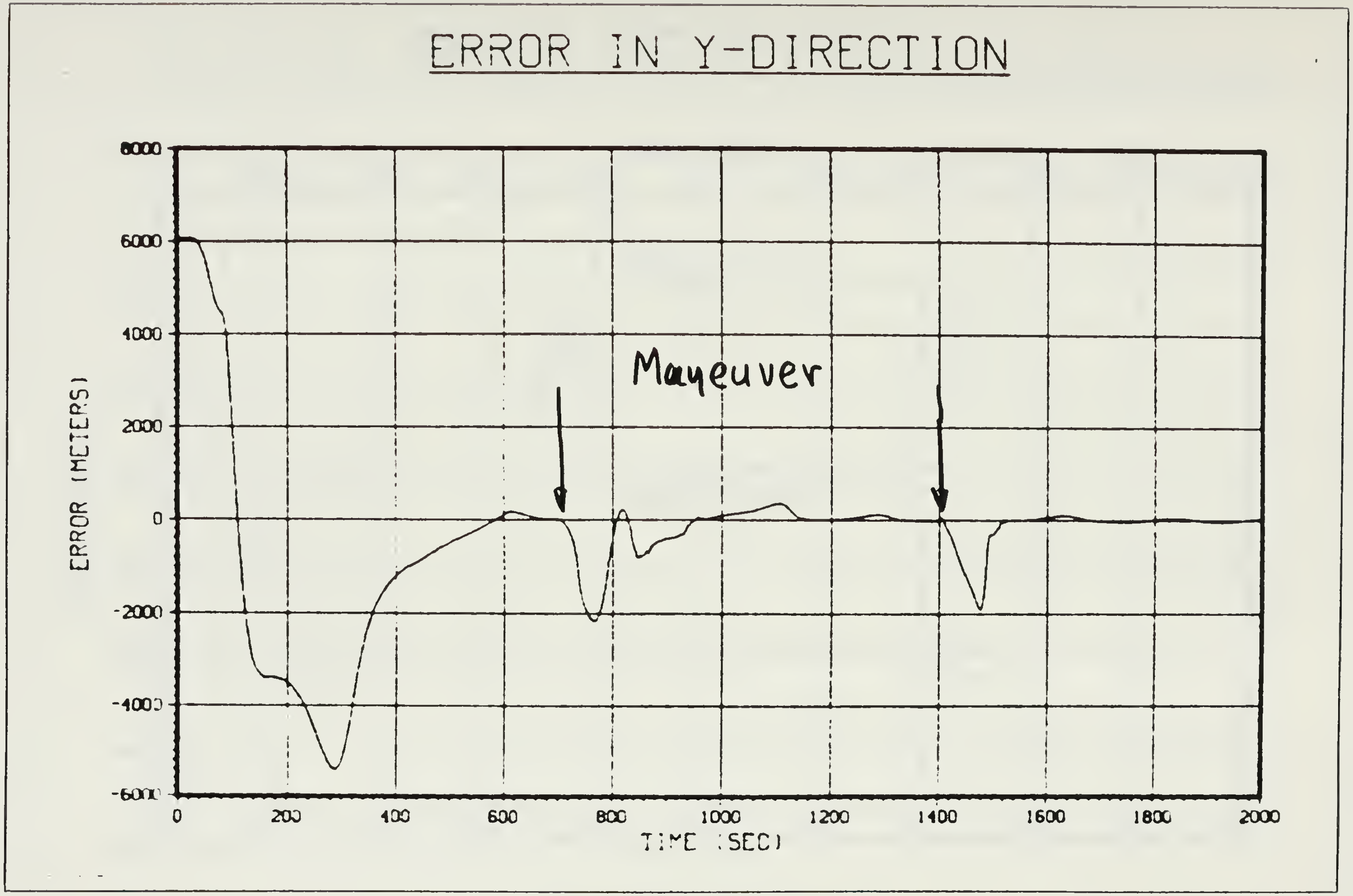

Figure 5.4 Range Error in Y-Direction Case 3.

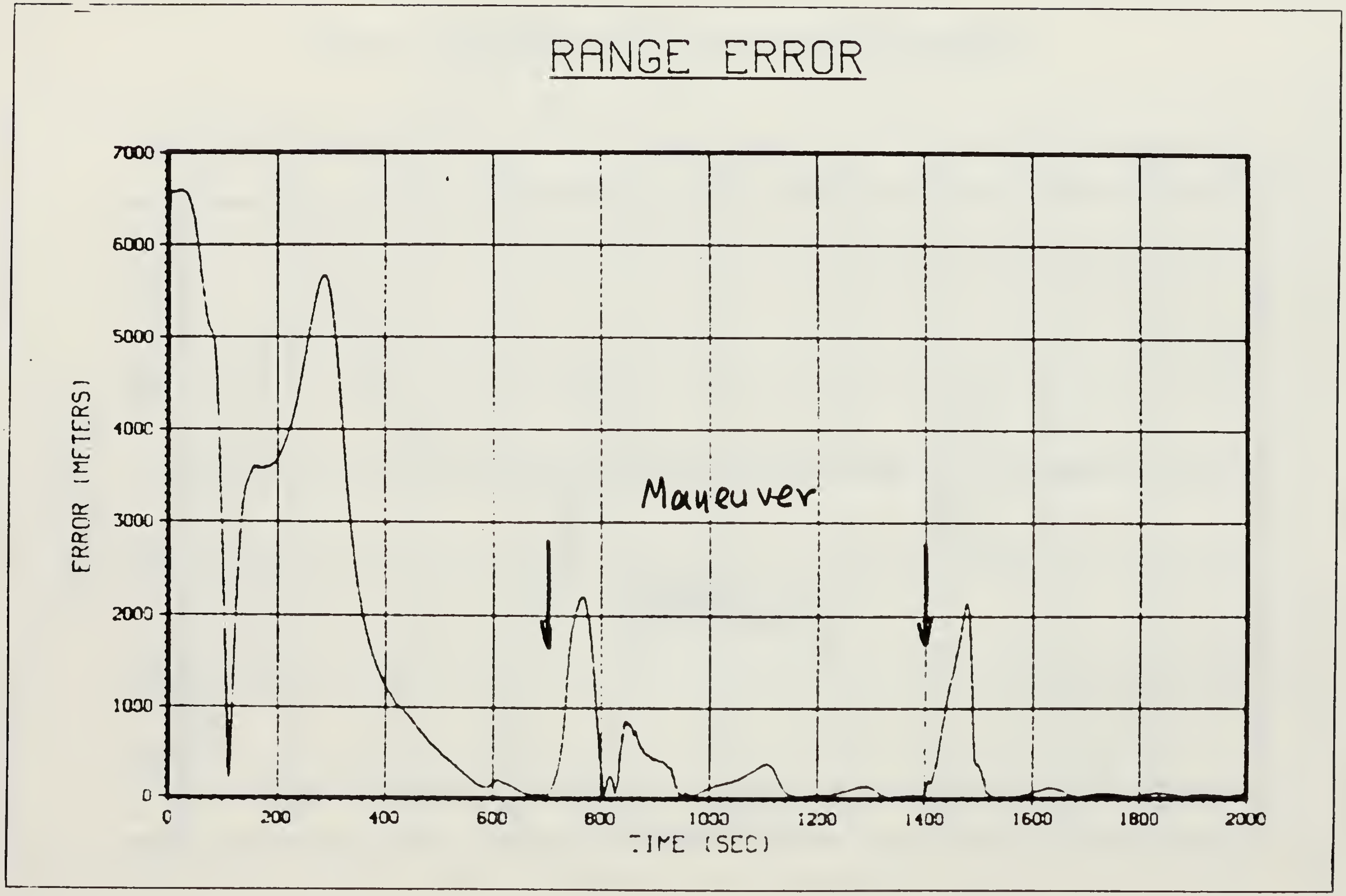

Figure 5.5 Range Error Case 3.

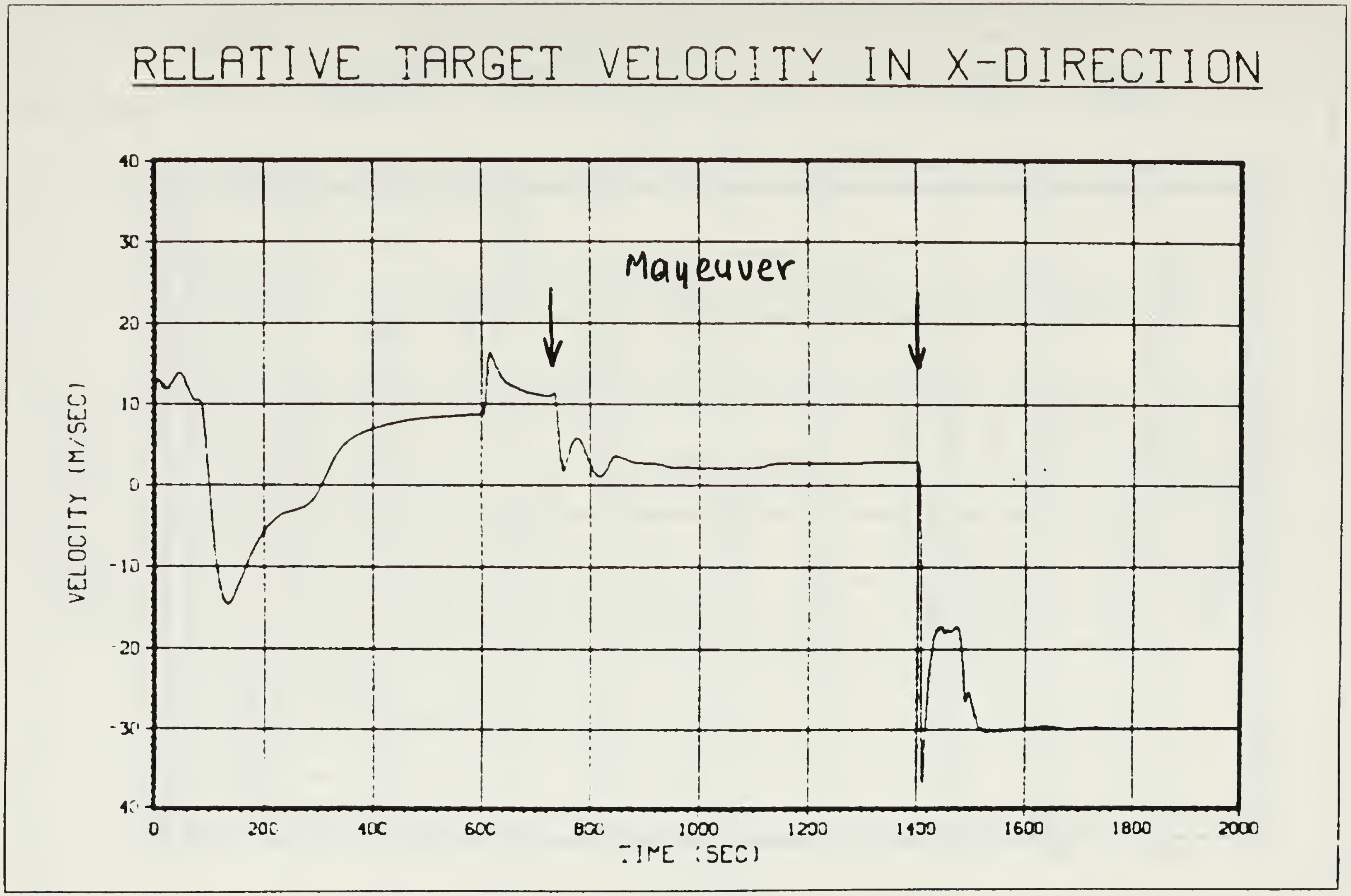

Figure 5.6 Relative Target Velocity in X-Direction Case 3.

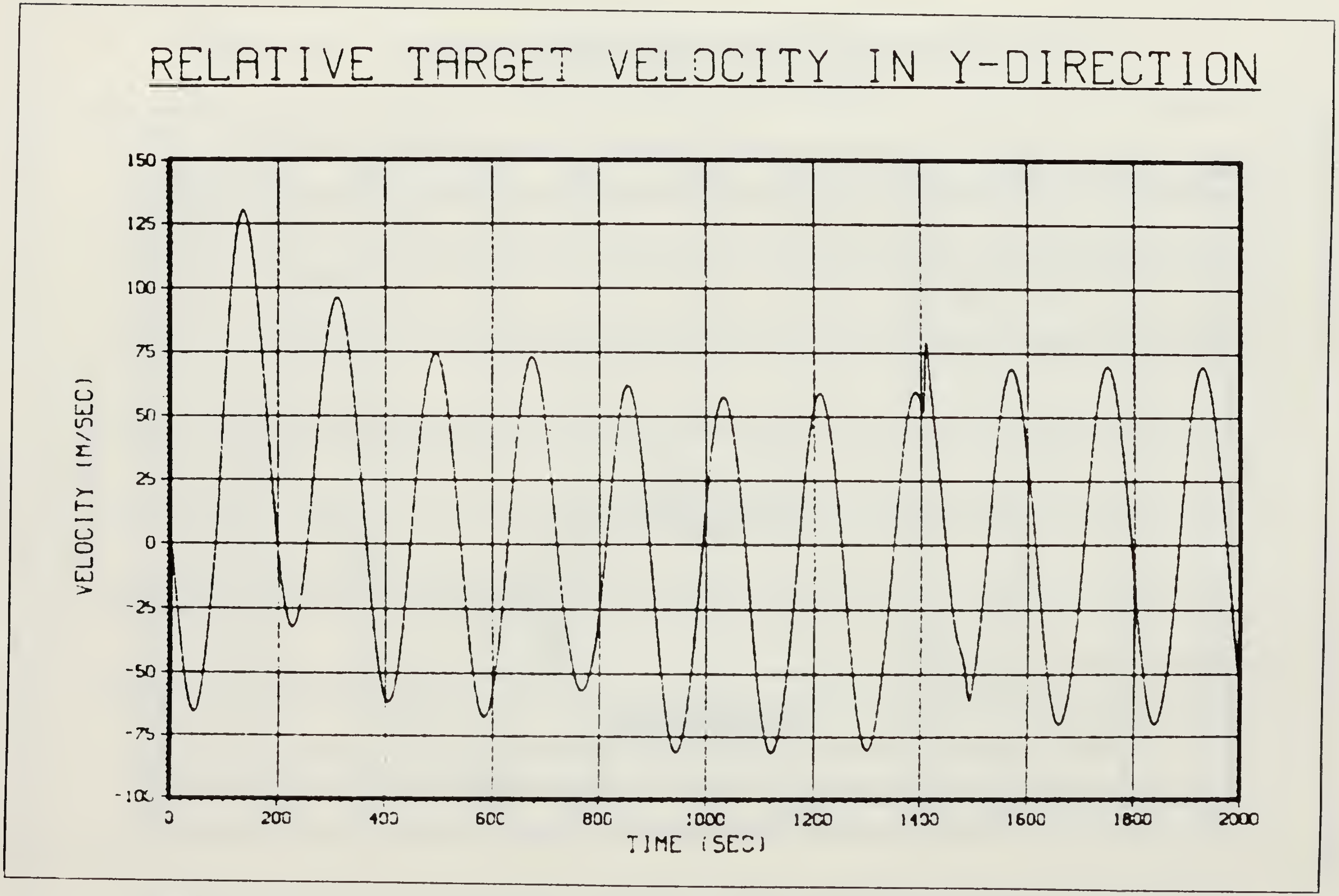

Figure 5.7 Relative Target Velocity in Y-Direction Case 3.

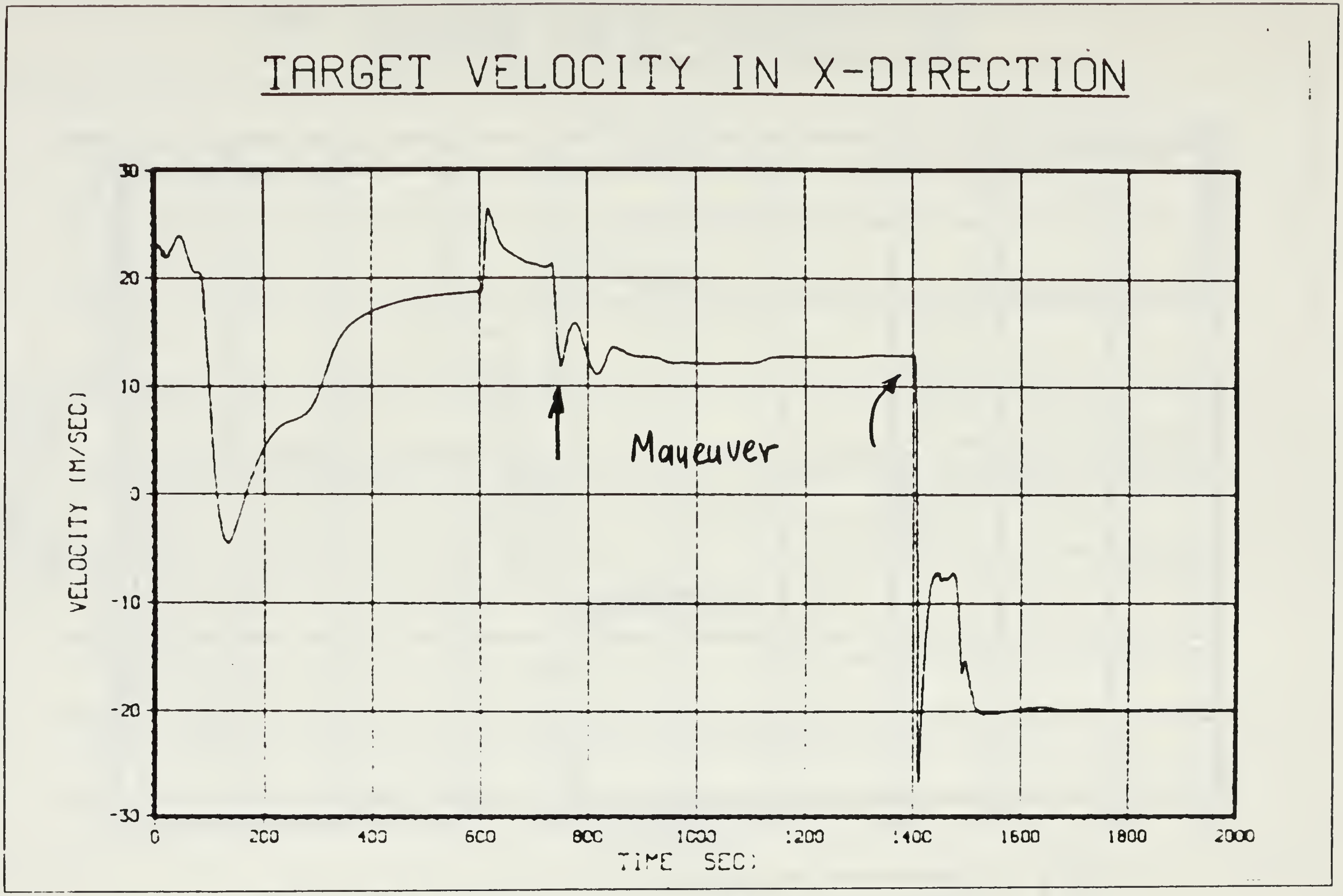

Figure 5.8 Target Velocity in X-Direction Case 3.

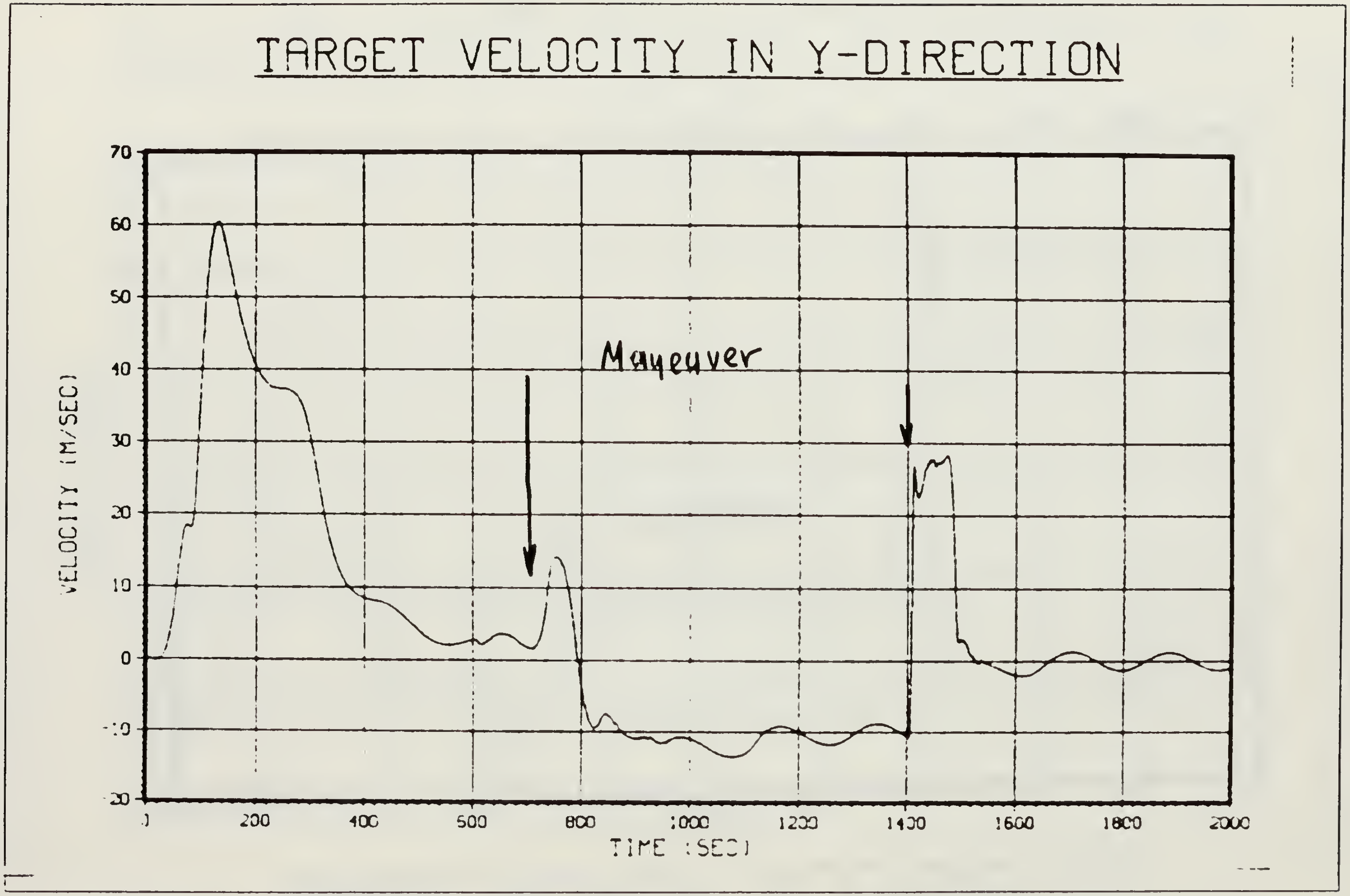

Figure 5.9 Target Velocity in Y-Direction Case 3.

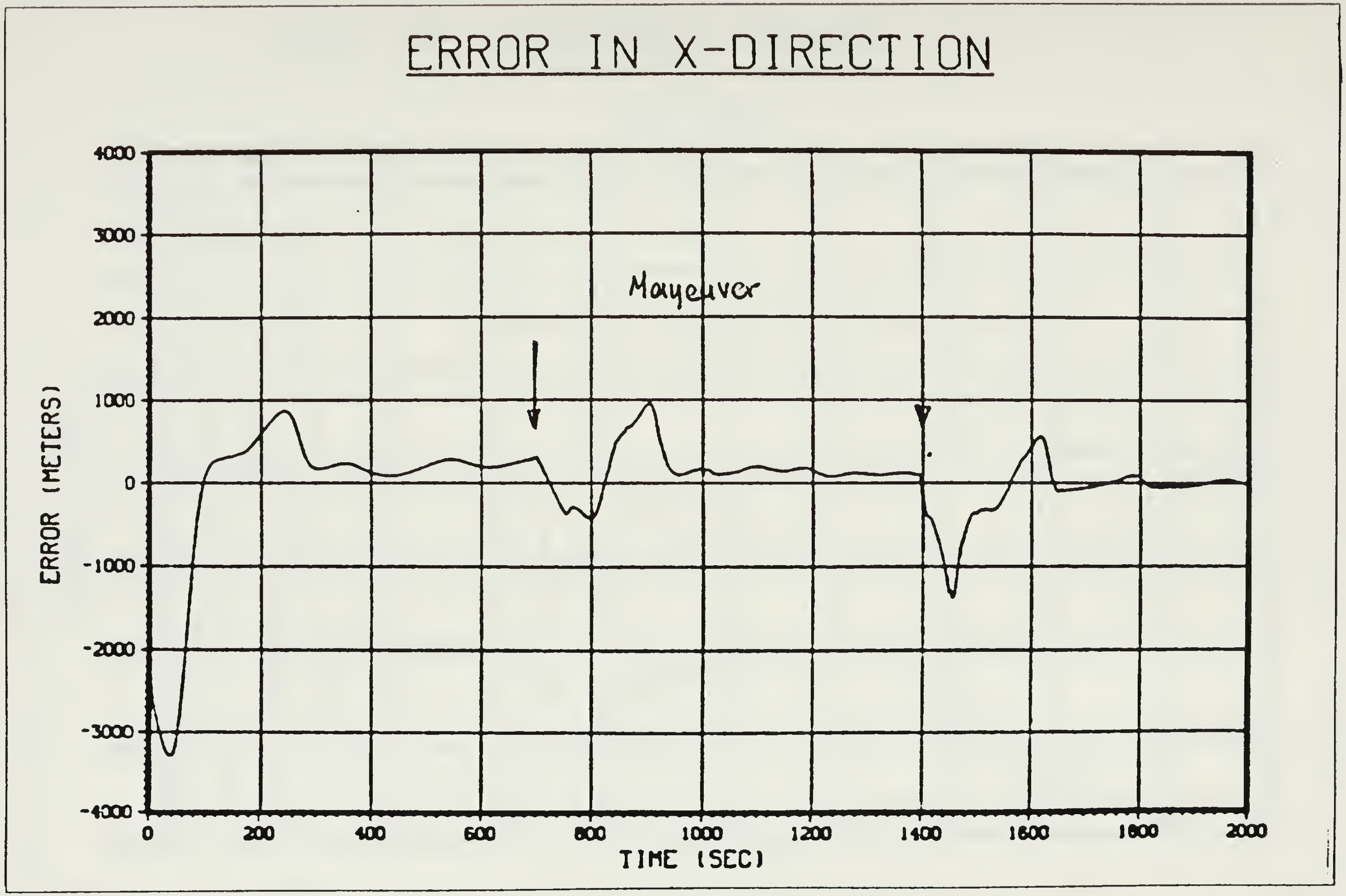

Figure 5.10 Range Error in X-Direction Case 4.

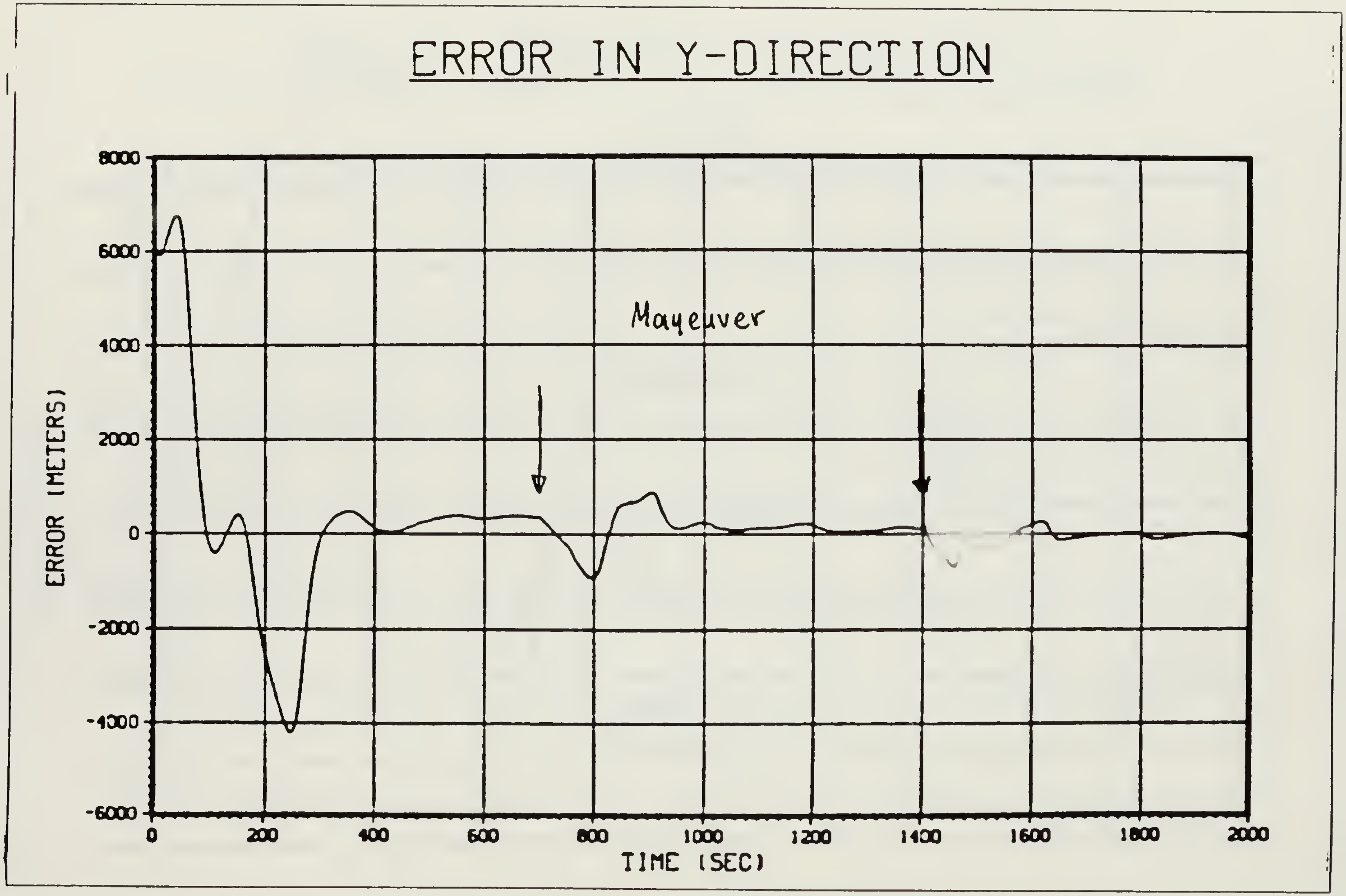

Figure 5.11   Range Error in Y-Direction Case 4.

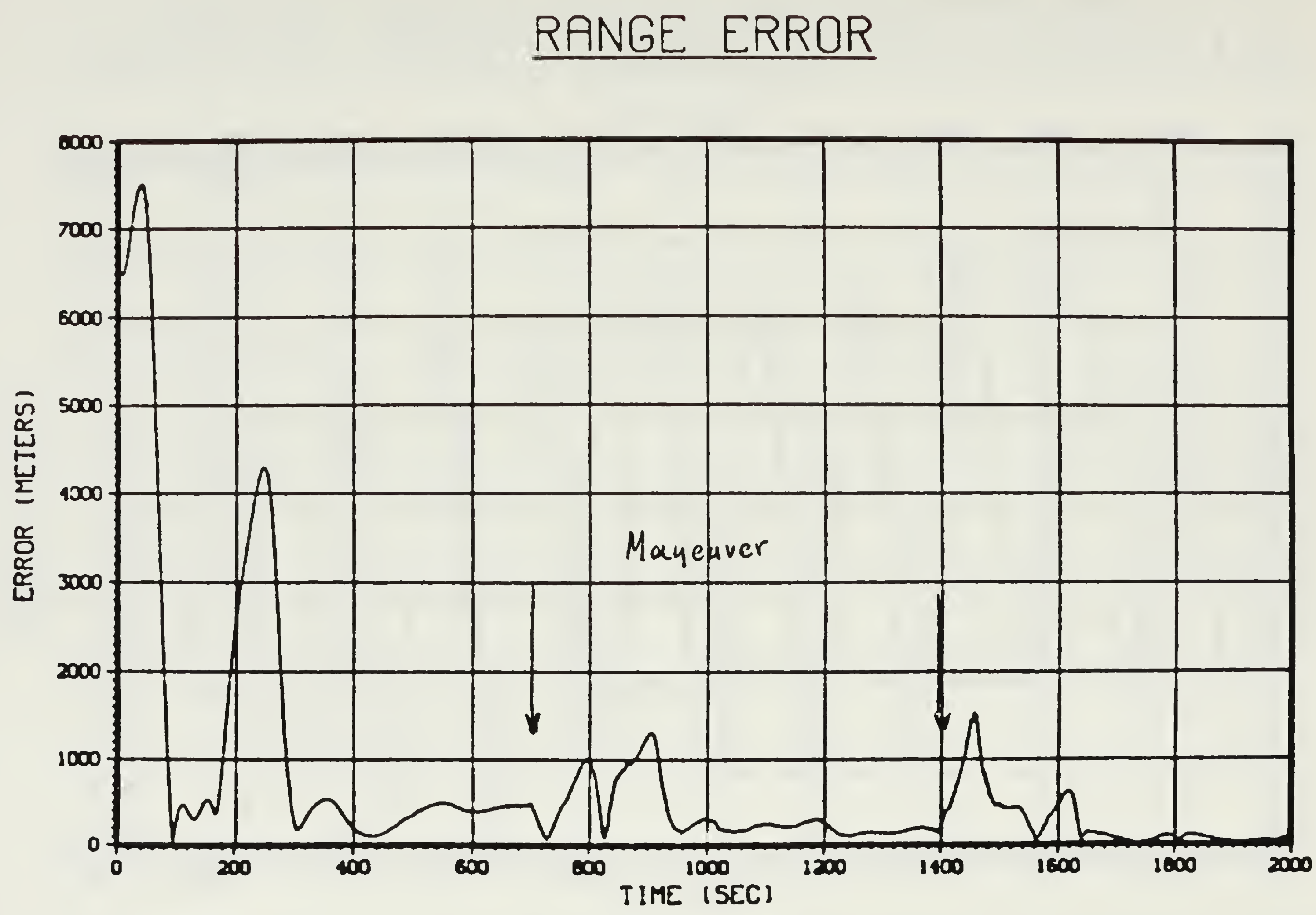

Figure 5.12 Range Error Case 4.

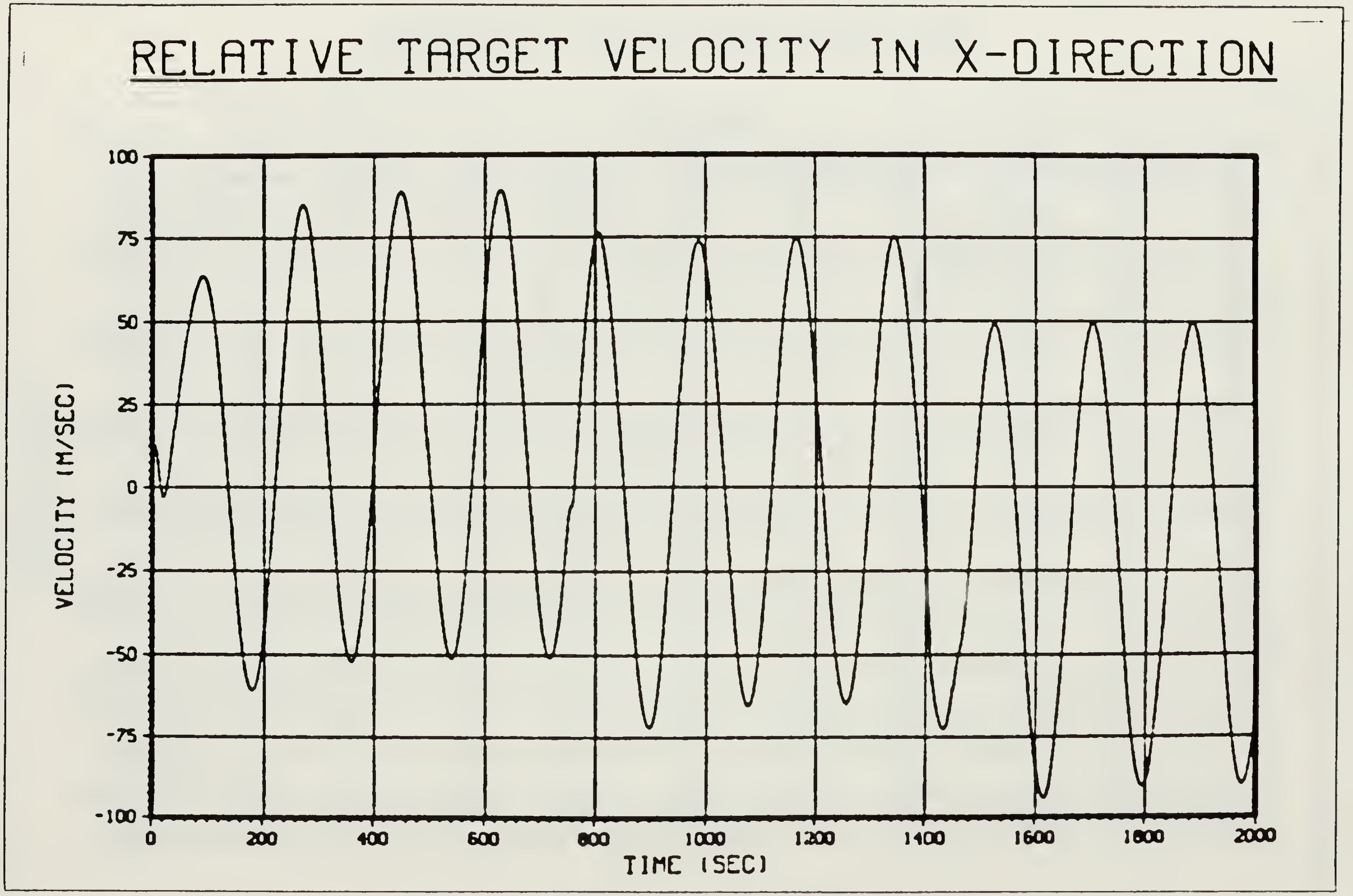

Figure 5.13 Relative Target Velocity in X-Direction Case 4.

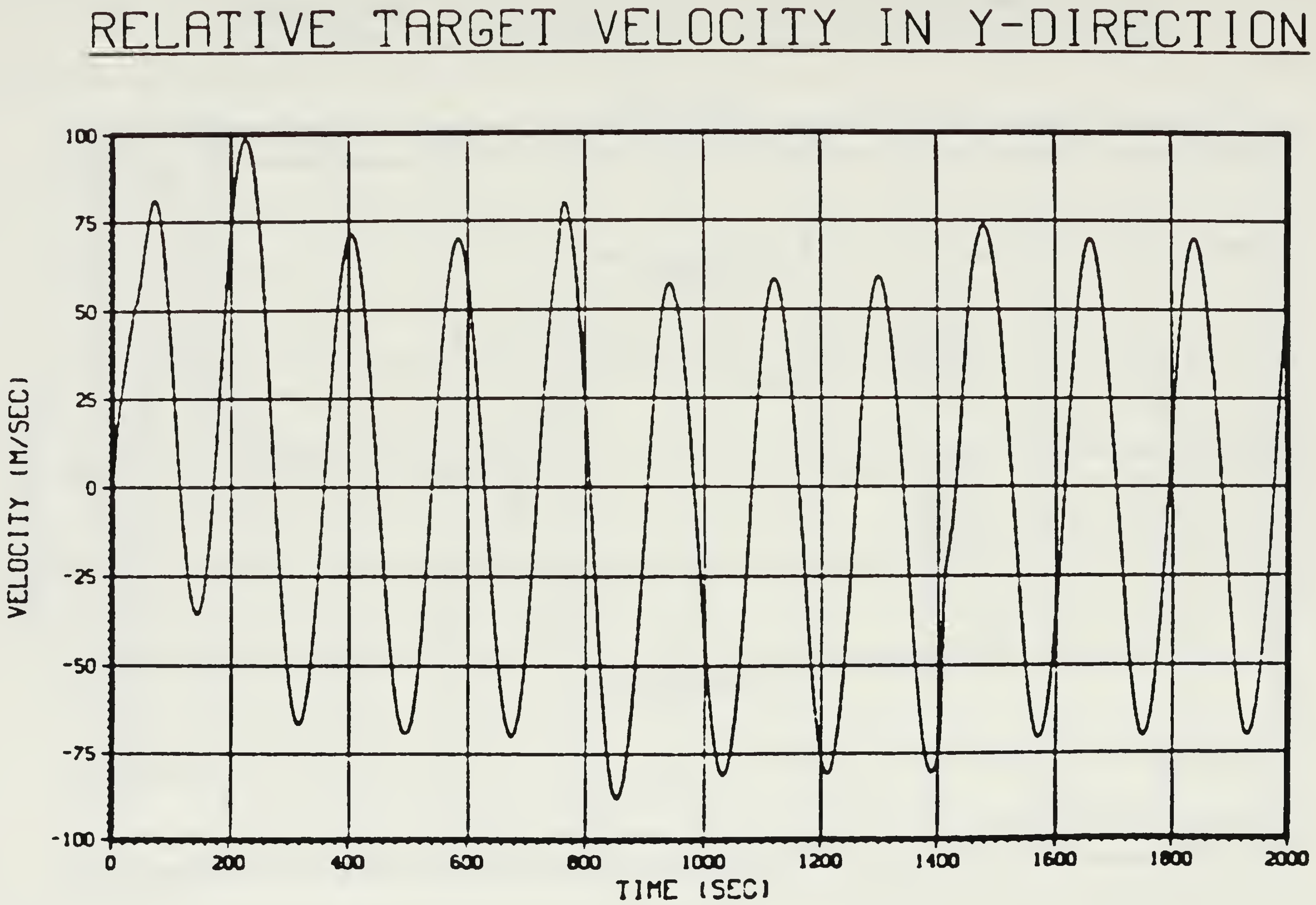

Figure 5.14 Relative Target Velocity in Y-Direction Case 4.

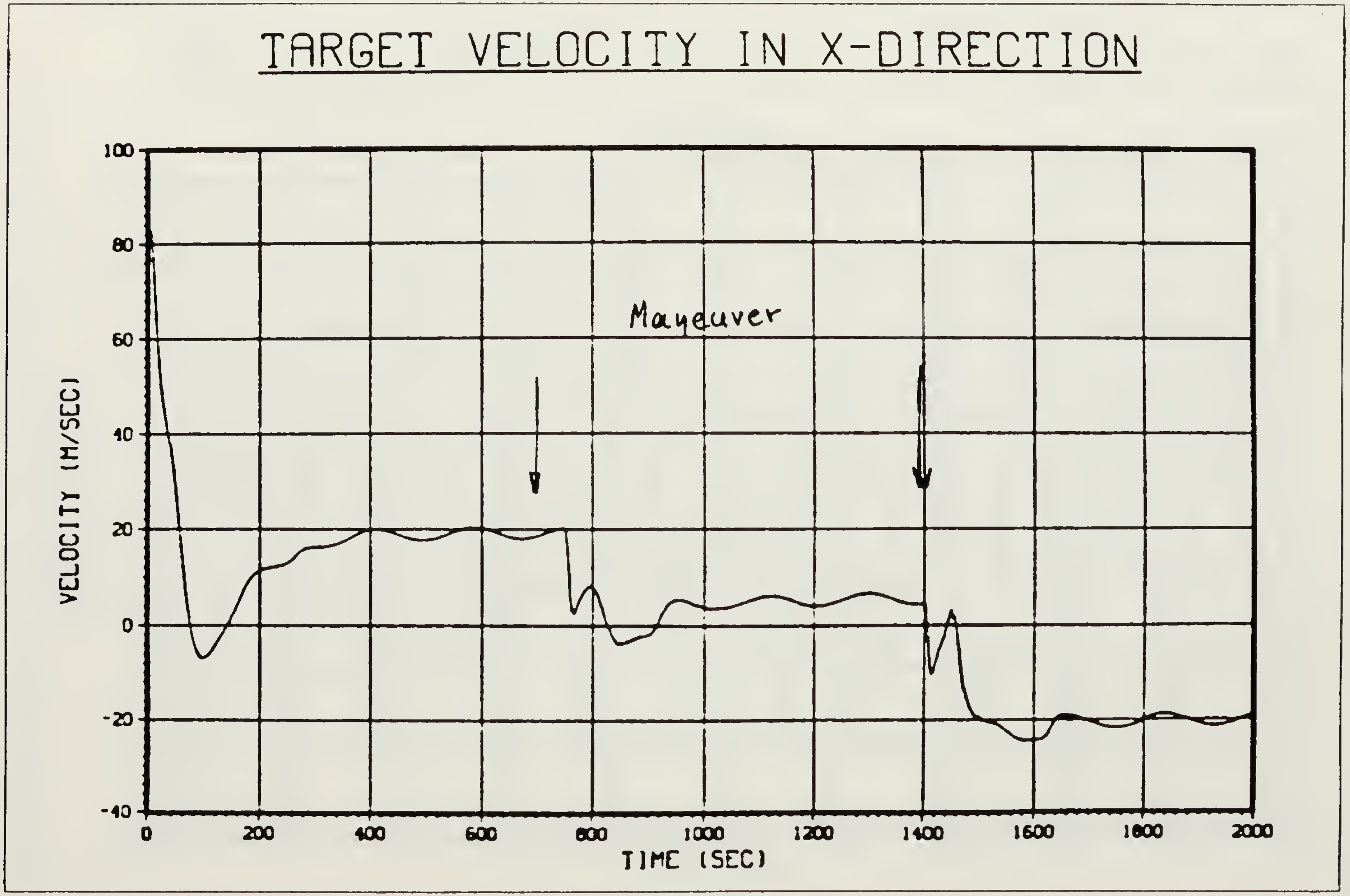

Figure 5.15 Target Velocity in X-Direction Case 4.

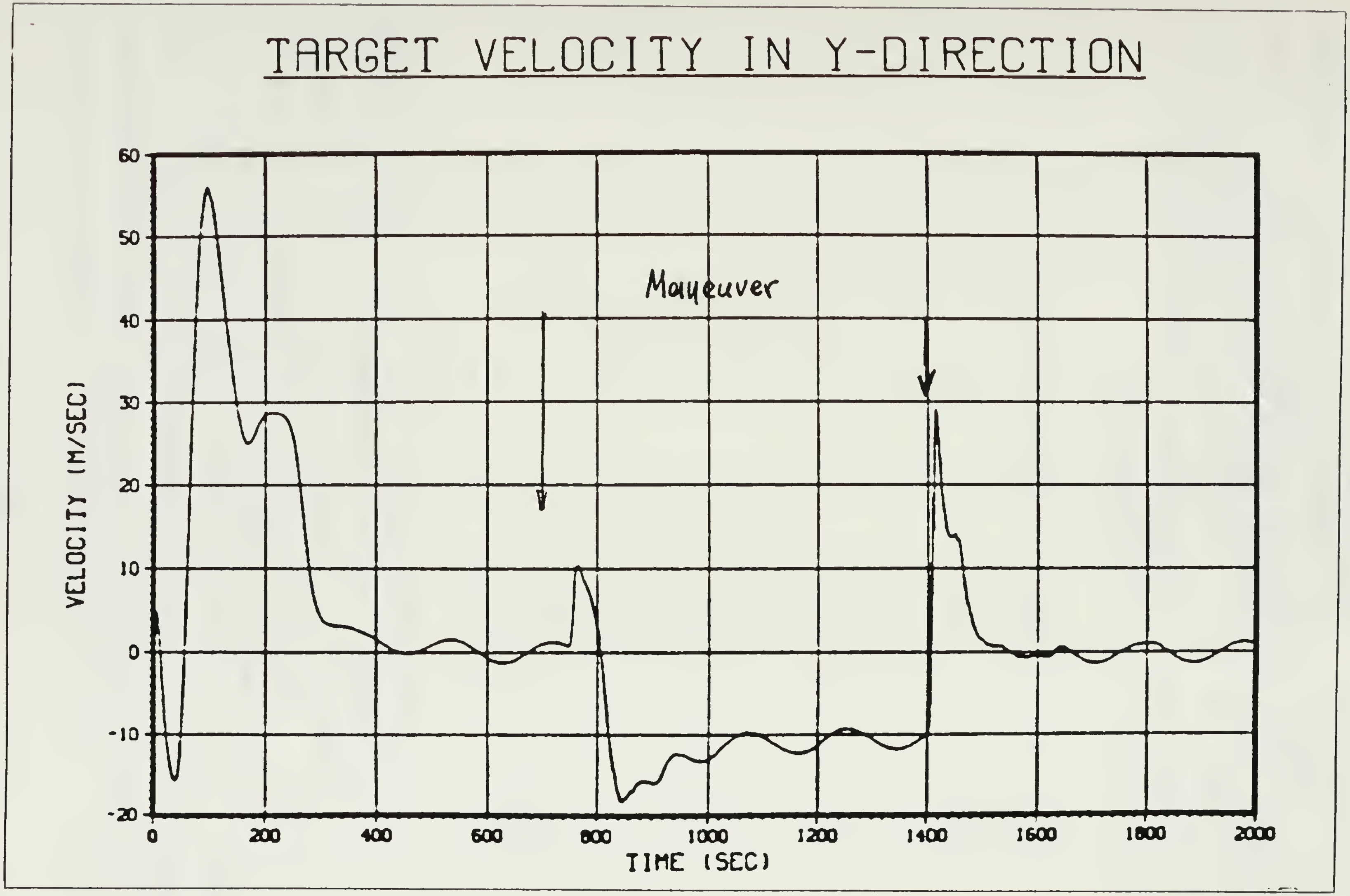

Figure 5.16 Target Velocity in Y-Direction Case 4.

3 Filter Behavior Under Different Values of w and σ^2

In order to investigate the behavior of the filter for more extreme conditions, simulations where conducted with various values of measurement error variance (σ^2) and various values of (w). The following combinations where tested for the case 3 configuration and the range error was obtained in each of the combinations.

w	σ^2
0.1	0.5
0.1	2.0
0.1	4.0
1.0	0.5
1.0	2.0
1.0	4.0
3.0	0.5
3.0	2.0
3.0	4.0
10.0	0.5
10.0	2.0
10.0	4.0

In the following Figures 5.17 to 5.28 the filter behavior is displayed. It is characteristic that the filter tracking accuracy and quality is related to both values of (w) and (σ^2). For the specific configuration it came out that if the (σ^2) was more than 0.5 then the filter was very sensitive to the value of (w). The best results were obtained with the smallest tested value of w=0.1. This should be expected because in the case that the measurement noise is too big and we additionally introduce uncertainty due to the target maneuver, then the filter assumes a lot of uncertainty and at any time behaves erratically following the noisy received measurements. It seems that for each kind of target and environmental condition (i.e. measurement noise variance) there will be an optimal (w) to account for the target maneuvers.

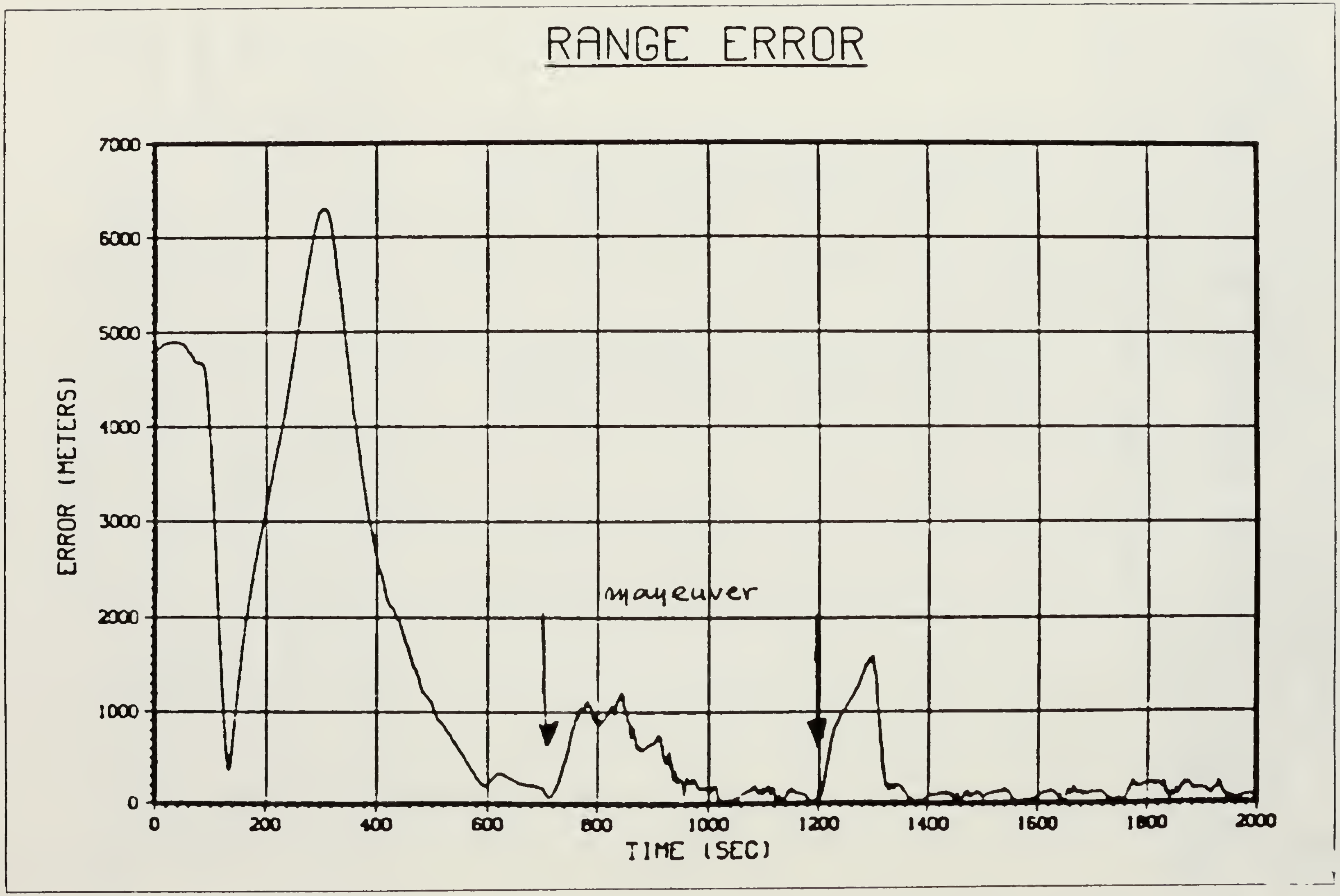

Figure 5.17 Range Error Case 3, w=0.1, σ^2=0.5.

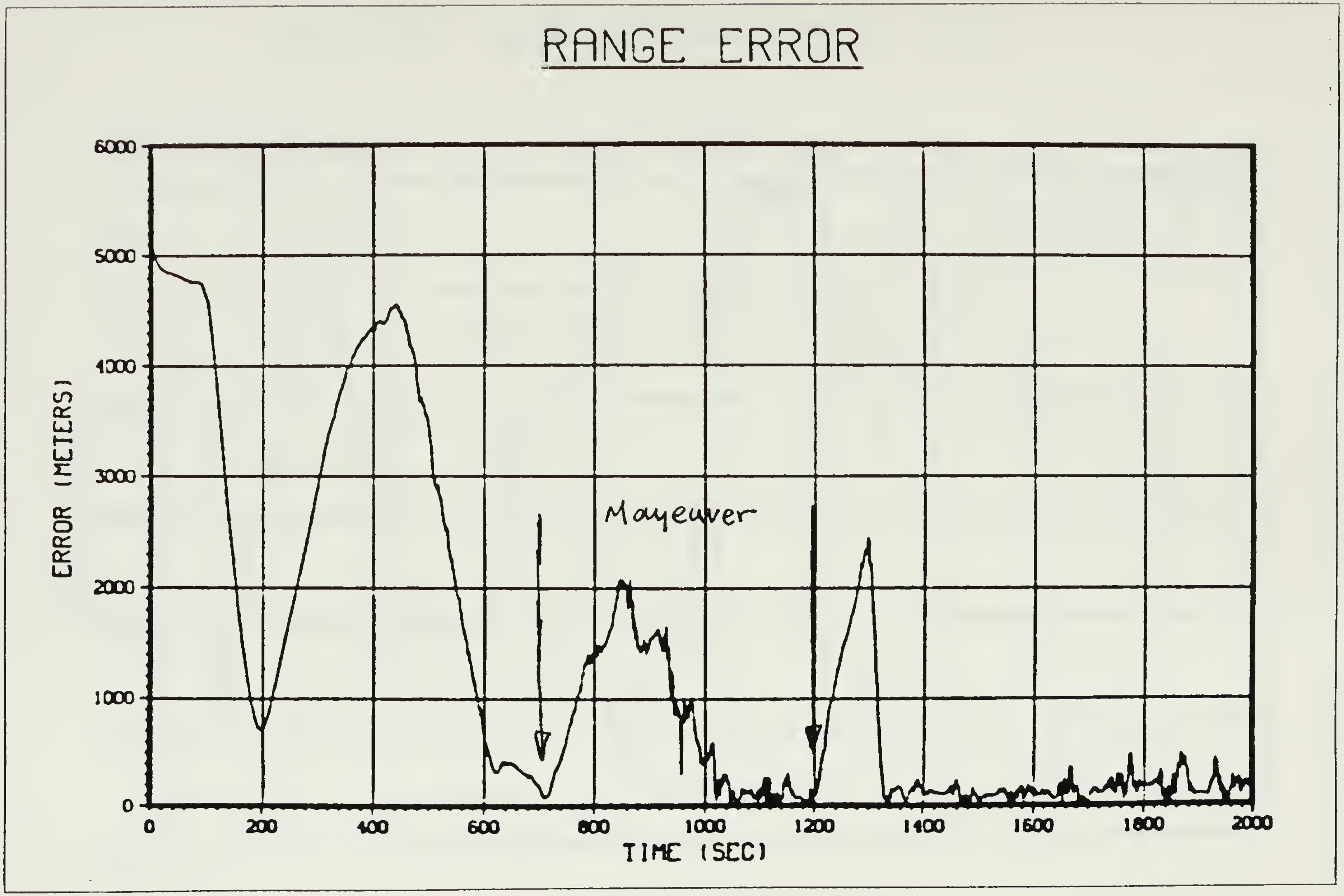

Figure 5.18 Range Error Case 3, w=0.1, σ^2=2.0.

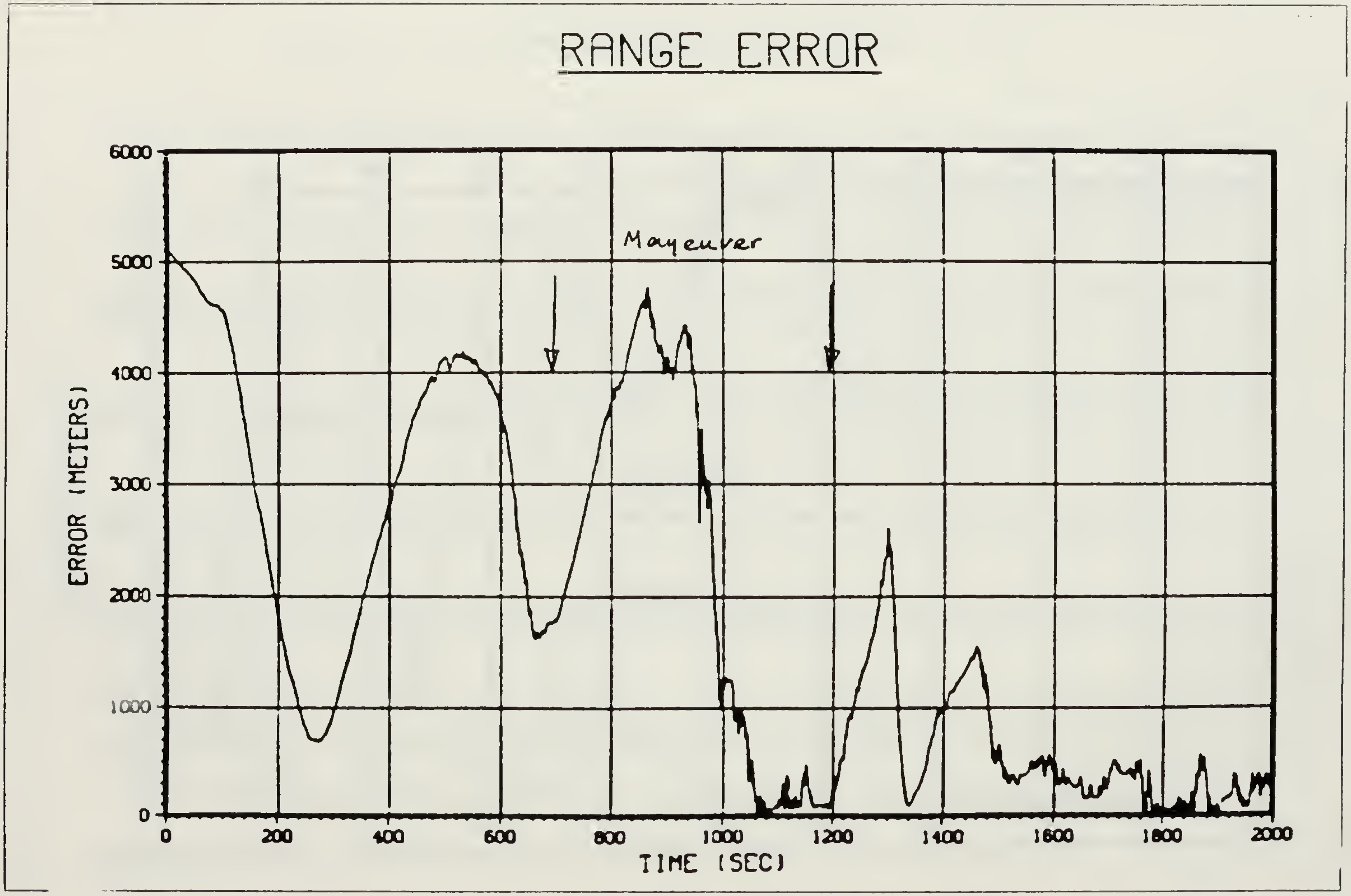

Figure 5.19   Range Error Case 3, w=0.1, σ^2=4.0.

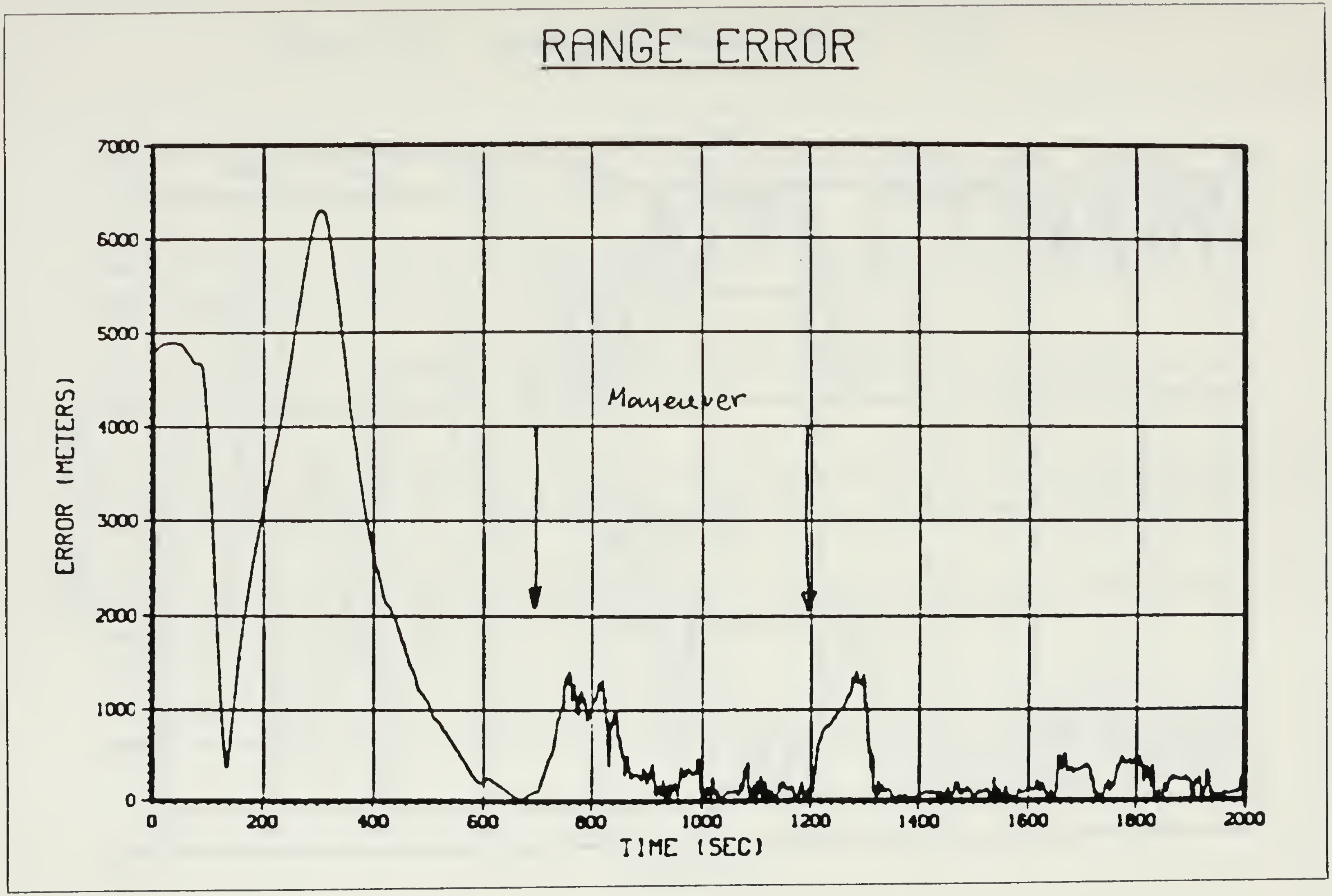

Figure 5.20 Range Error Case 3, w=1.0, $\sigma^2=0.5$.

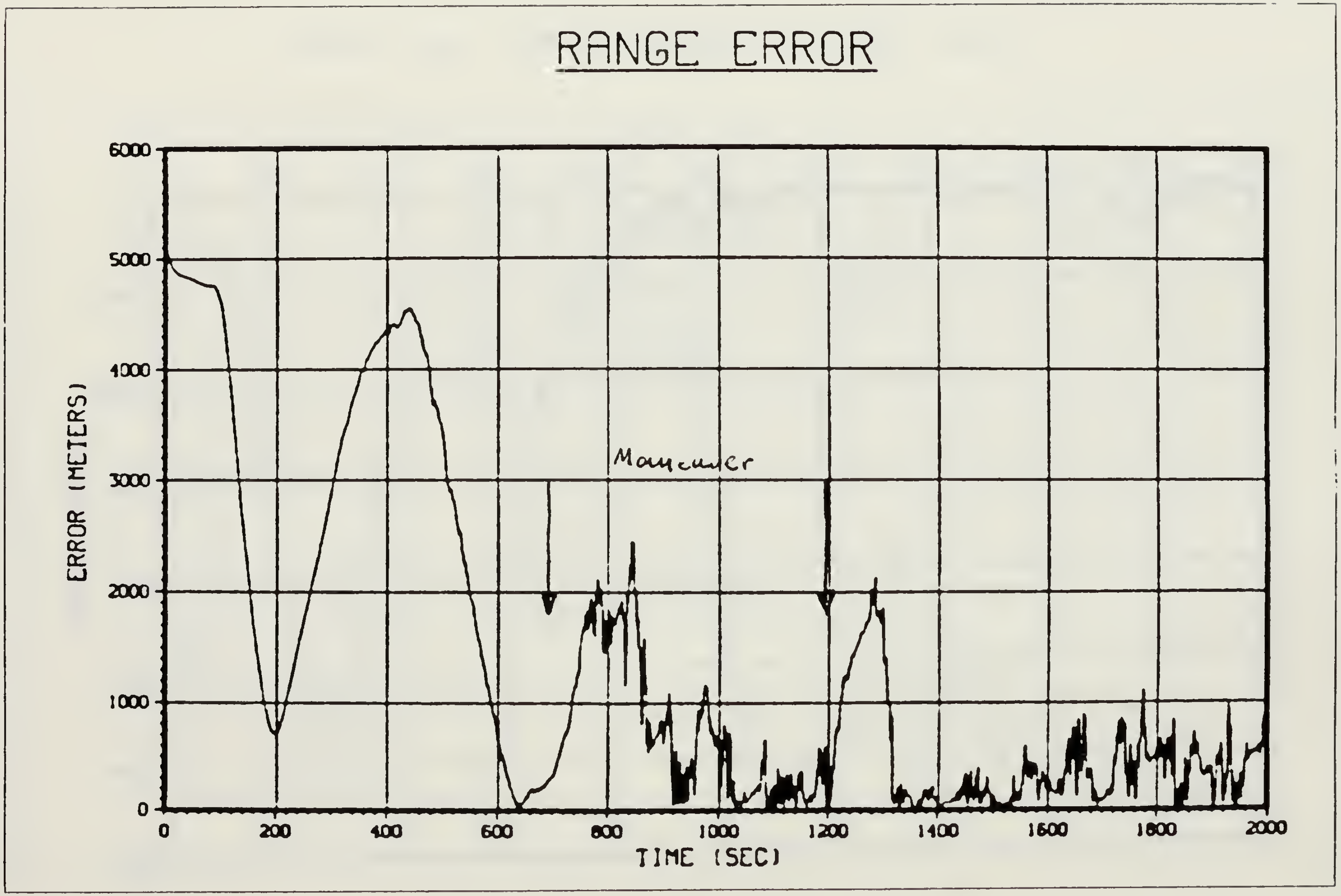

Figure 5.21 Range Error Case 3, w=1.0, σ^2=2.0.

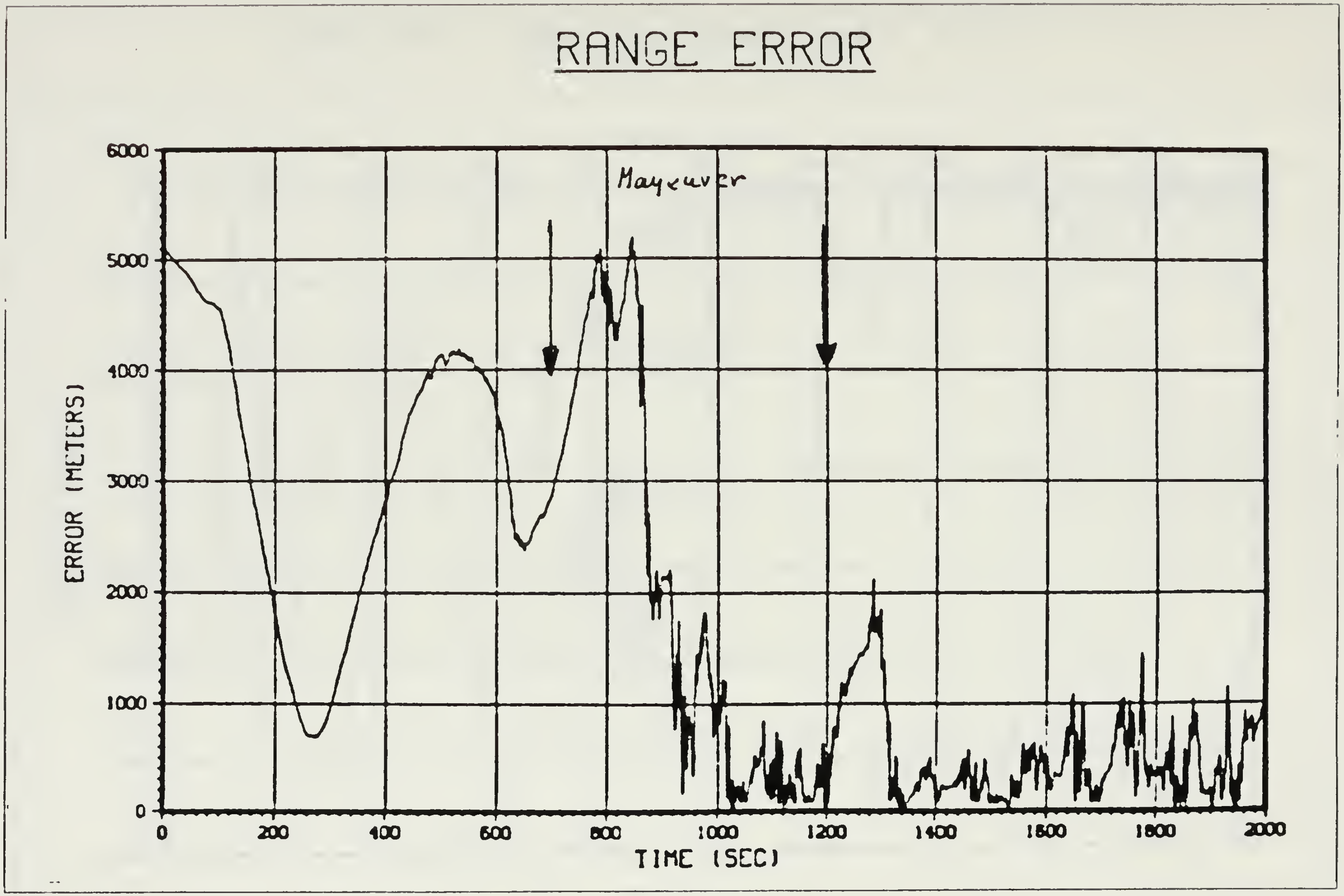

Figure 5.22   Range Error Case 3, w=1.0, σ^2=4.0.

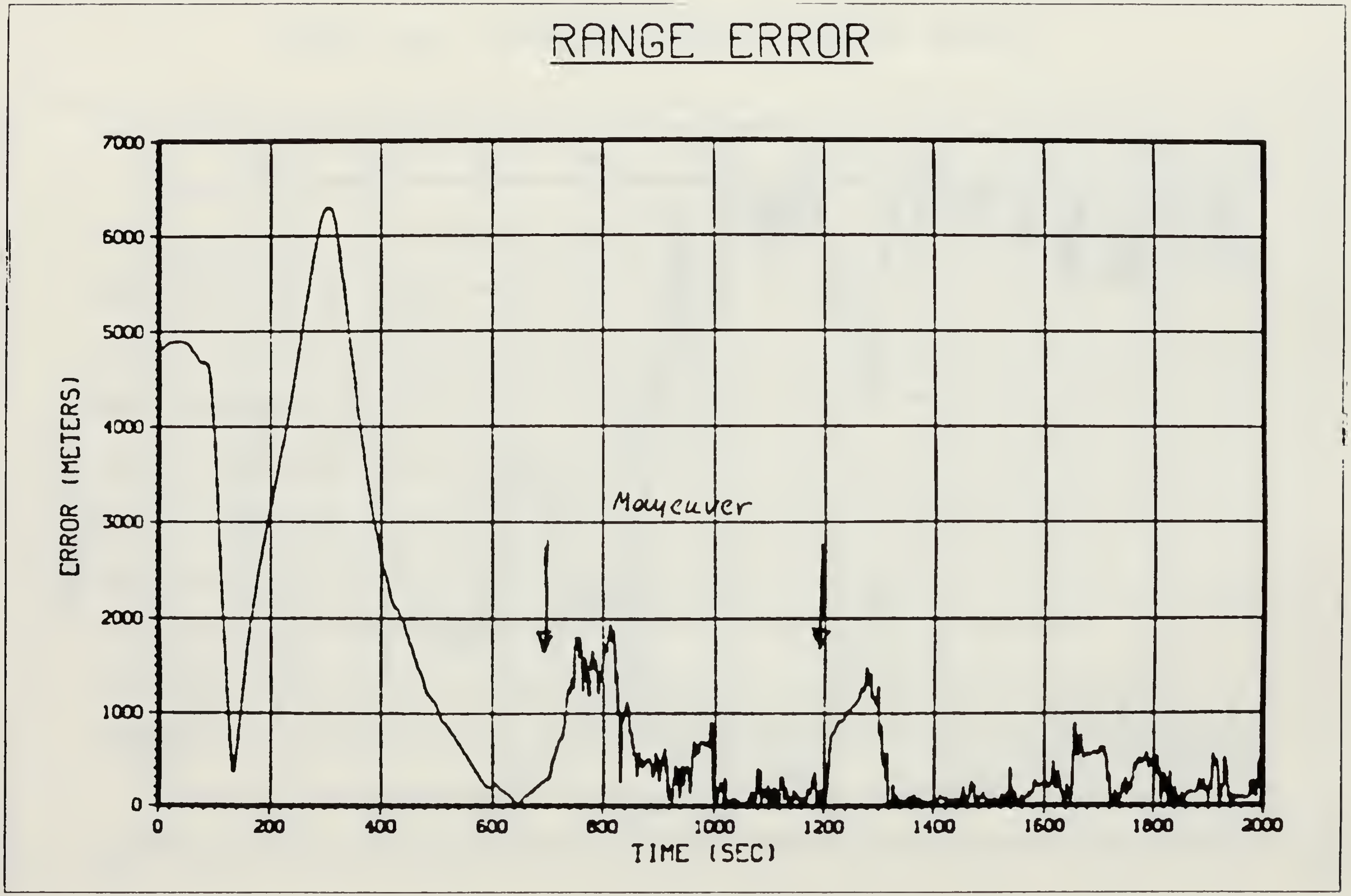

Figure 5.23 Range Error Case 3, w=3.0, σ^2=0.5.

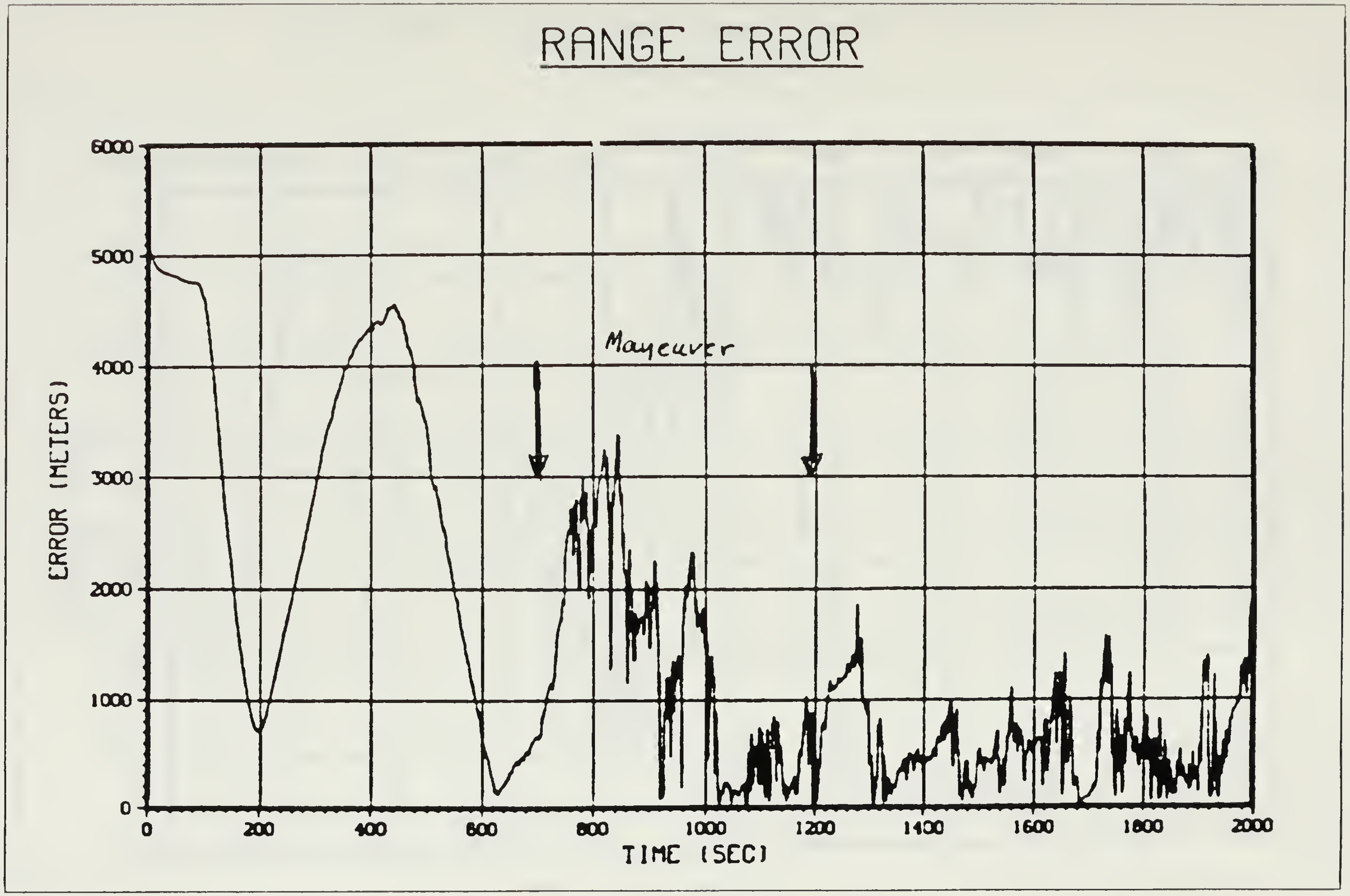

Figure 5.24   Range Error Case 3, $w=3.0$, $\sigma^2=2.0$.

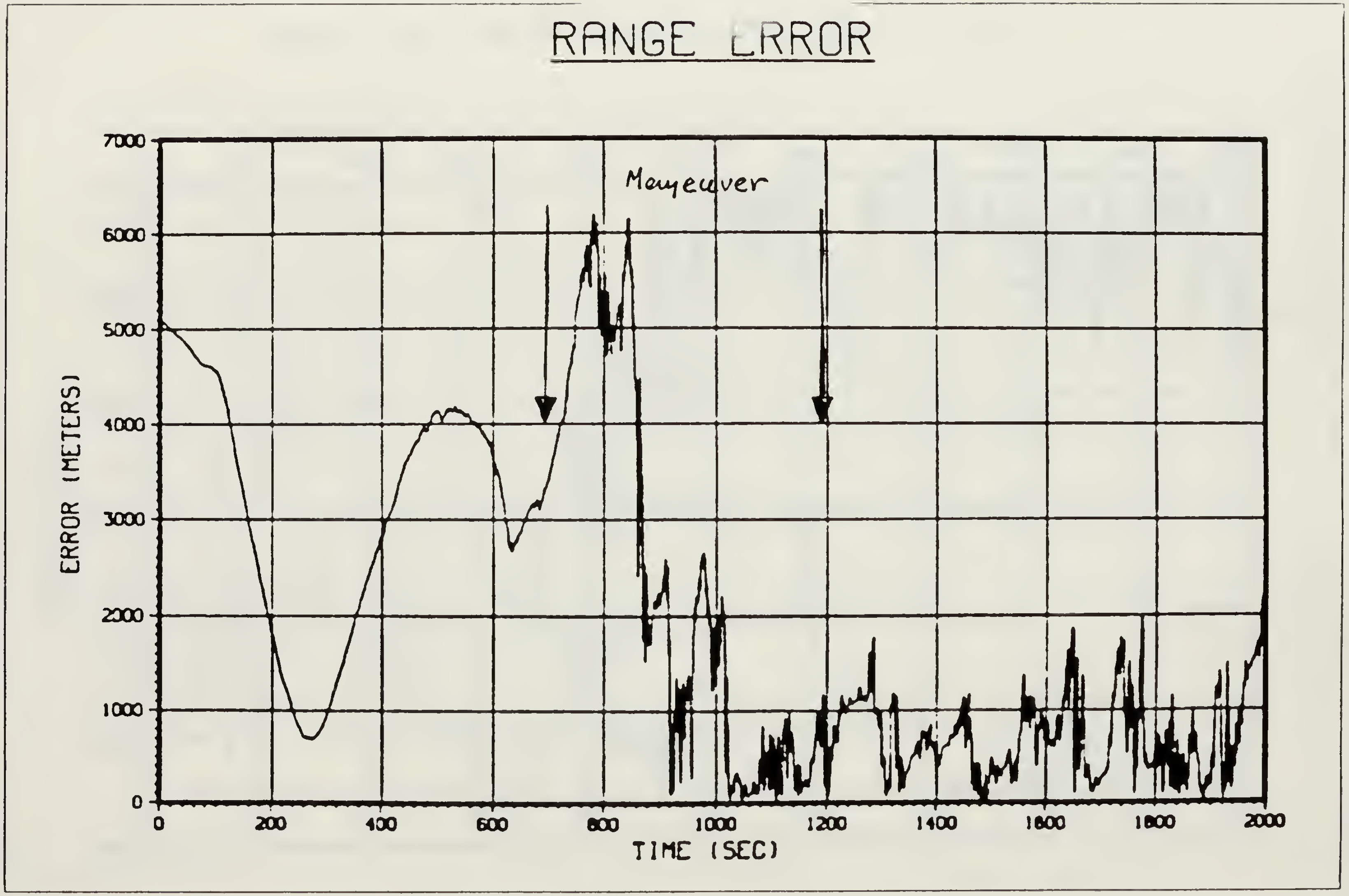

Figure 5.25   Range Error Case 3, w=3.0, σ^2=4.0.

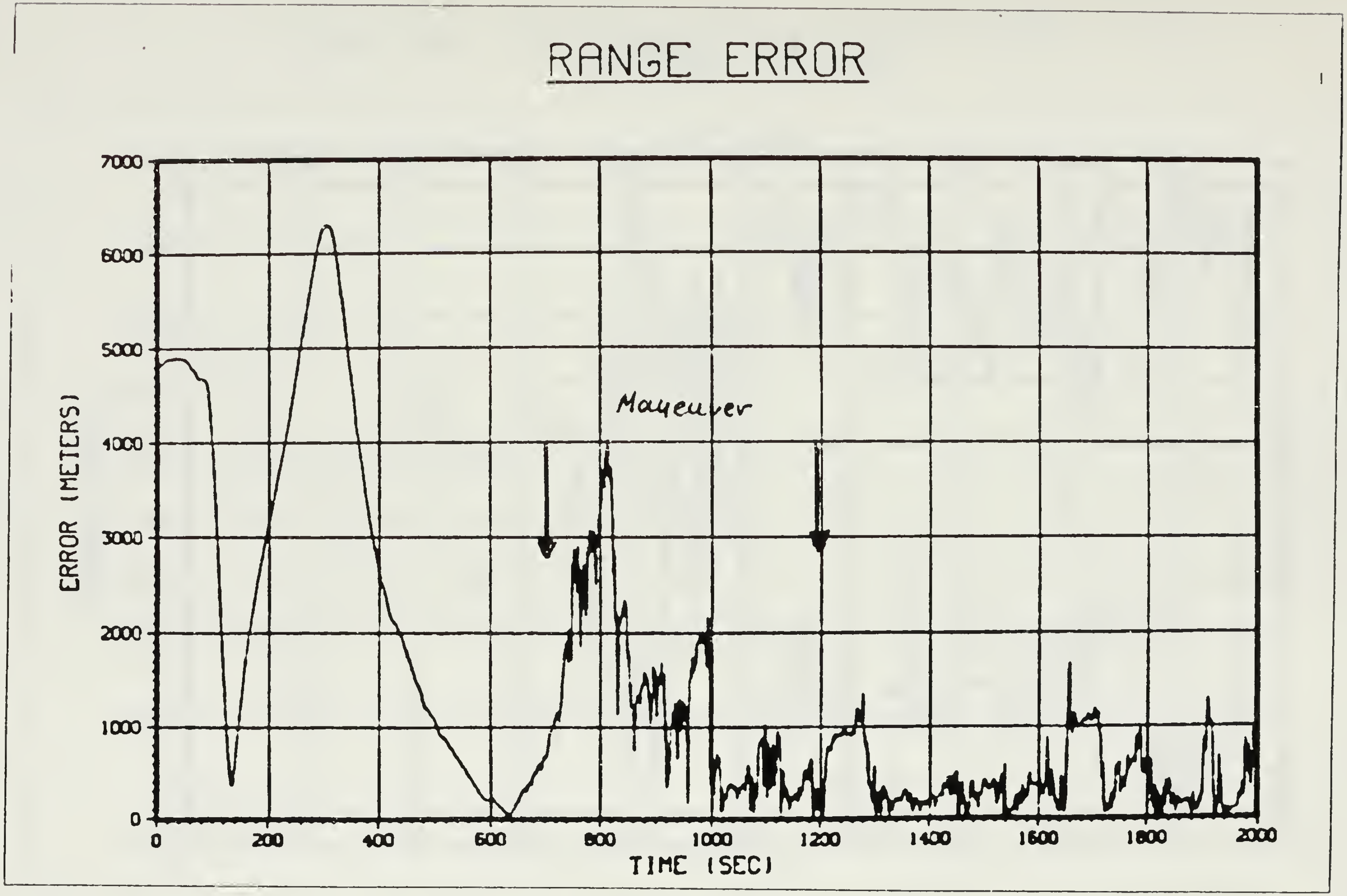

Figure 5.26 Range Error Case 3, w=10.0, σ^2=0.5.

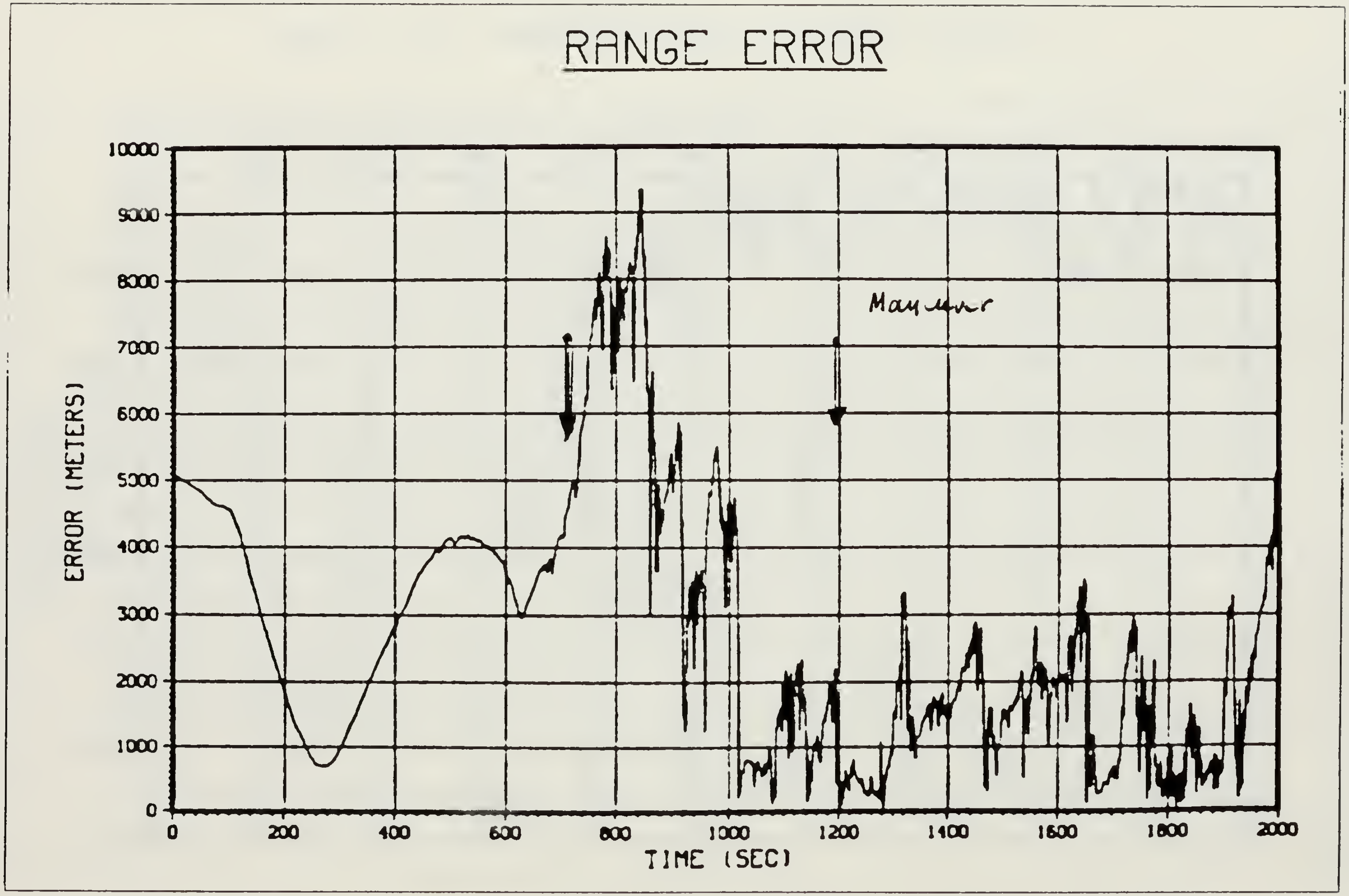

Figure 5.27 Range Error Case 3, w=10.0, σ^2=2.0.

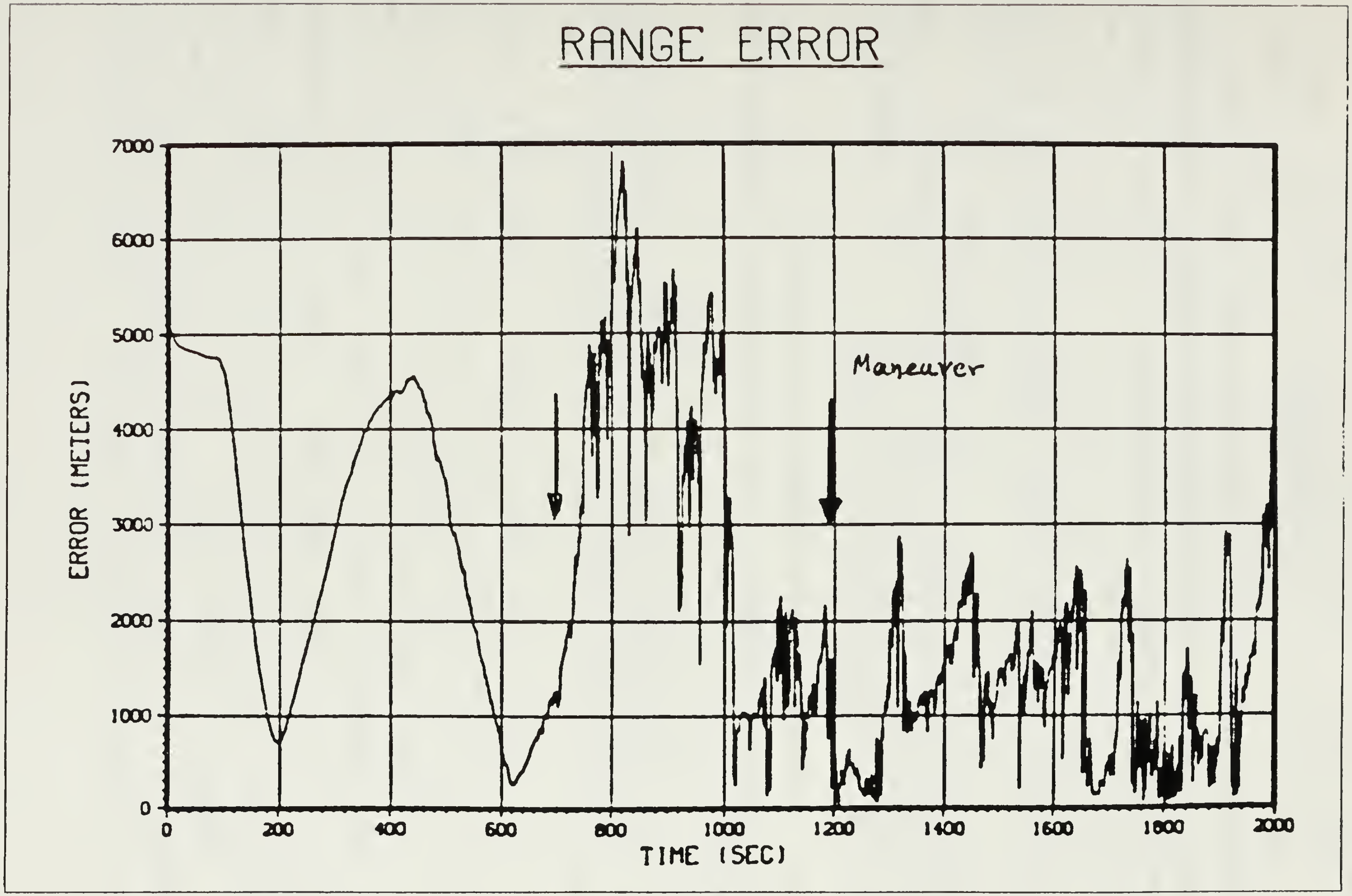

Figure 5.28   Range Error Case 3, w=10.0, σ^2=4.0.

VI. CONCLUSIONS

The proposed way of solving the problem of tracking a maneuvering target using noisy bearings-only measurements was tested and it exhibited satisfactory behavior. The main characteristics of it are:

1. The filter responds satisfactorily in the case that the target maneuvers. The filter appears to be very sensitive to the value of σ^2. The results were satisfactory up to the value of $\sigma^2=0.5$. After that the behavior of the filter depends very much on the value of w. The smaller the value of w the better the filter tracks.

2. The design is simple and almost no extra computational power is needed beyond that of a nonmaneuvering target filter.

3. The estimation is accurate for a nonmaneuvering target as well, and it does not pay the overhead of reduced accuracy as the other methods do in the nonmaneuvering case.

4. Some other target-tracker configurations were tested which are not referenced in the previous chapters. In some of them the filter exhibited disability to track the target. In those cases the characteristic event was that the target was moving in such a way that even the tracker's maneuvers did not cause significant changes in the measured bearings. So the tracker maneuvers are very important in the bearings-only tracking problem. They must be such that will cause changing bearing rates. Of course the tracker's maneuvers are restricted by various factors as speed capability, tactical situation, intentions (evade or attack), etc.

Possible subjects for further investigation:

1. Analytically how does the filter behave in a variety of tracker-target, w - σ^2 configurations? For example, for a given value of (σ^2), what is the optimal (w)?

2. In the simulations the tracker was supposed to move with a continuously changing course which is not the real case. Also the tracker was supposed to assume huge amounts of acceleration during its maneuvers, i.e. it was supposed to change course and speed in one second which also is not realistic. How does this assumption differ from the real case?

3. Investigate the tracker motion under realistic constraints with the requirement of obtaining tactical advantage and simultaneously providing needed bearing rate to accurately solve the tracking problem.

4. Investigate the effect of assuming realistic constraints on target motion.

In this Thesis we dealt with the problem of maneuvering target passive tracking using a simple method. The first results are satisfactory, however the method needs further detailed investigation for even better performance.

APPENDIX A

SIMPLE EXAMPLE SIMULATION PROGRAM

```
      REAL*4 P(4,4),H(2,4),HT(4,2),F(4,4),FT(4,4),
     *S7(4,4),
     *X(4,1),T,TT,Z(2,1),XPR(4,1),PPR(4,4),Q(4,4),DET,
     *G(4,2),
     *hX(2,1),ZHX(2,1),S1(4,2),S2(2,2),S6(2,2),GH(4,4),
     *FX(4,1),R1,R2,XI(4,1),R(2,2),W,FPUP(4,4),FPUPFT(4,4),
     *A1,A2,A3,R3
      INTEGER N,M,KK,I,J,K,L,NR,S
C
      N=4
      W=0.5
      M=1
      S=2
      NR=15
      DO 1 I=1,N
      DO 1 J=1,N
1     F(I,J)=0.
      F(1,1)=1.
      F(1,2)=1.
      F(2,2)=1.
      F(3,3)=1.
      F(3,4)=1.
      F(4,4)=1.
      DO 2 I=1,N
      DO 2 J=1,N
2     FT(I,J)=F(J,I)
      DO 3 I=1,N
      DO 3 J=1,N
3     S7(I,J)=0.
      DO 4 I=1,N
```

```
      DO 4 J=1,N
4     S7(I,I)=1.
      X(1,1)=10000.
      X(2,1)=0.
      X(3,1)=0.
      X(4,1)=1600.
      R(1,1)=1.
      R(1,2)=0.
      R(2,1)=0.
      R(2,2)=1.
      DO 5 I=1,N
      DO 5 J=1,N
5     P(I,J)=0.
      P(1,1)=1000.
      P(2,2)=1000.
      P(3,3)=1000.
      P(4,4)=1000.
      DO 6 I=1,S
      DO 6 J=1,N
6     H(I,J)=0.
      H(1,1)=1.
      H(2,3)=1.
      DO 7 I=1,S
      DO 7 J=1,N
7     HT(J,I)=H(I,J)
      DO 71 I=1,N
      DO 71 J=1,N
71    Q(I,J)=0.
      Q(1,1)=.25*W
      Q(3,3)=.25*W
      Q(1,2)=.5*W
      Q(3,4)=.5*W
      Q(2,1)=.5*W
      Q(4,3)=.5*W
      Q(2,2)=W
```

```
      Q(4,4)=W
      DO 999 KK=1,NR
      T=FLOAT(KK)
      L=KK-1
      TT=FLOAT(L)
      Z(1,1)=10000.*COS(.157*TT)
      Z(2,1)=10000.*SIN(.157*TT)
      CALL MM(P,HT,S1,N,N,S)
      CALL MM(H,S1,S2,S,N,S)
      DO 8 I=1,S
      DO 8 J=1,S
8     S2(I,J)=S2(I,J)+R(I,J)
      DET=S2(1,1)*S2(2,2)-S2(1,2)*S2(2,1)
      S6(1,1)=S2(2,2)/DET
      S6(1,2)=-S2(1,2)/DET
      S6(2,1)=-S2(2,1)/DET
      S6(2,2)=S2(1,1)/DET
      CALL MM(S1,S6,G,N,S,S)
      CALL MM(H,X,HX,S,N,M)
      DO 9 I=1,S
      DO 9 J=1,M
9     ZHX(I,J)=Z(I,J)-HX(I,J)
      CALL MM(G,ZHX,XI,N,S,M)
      DO 10 I=1,N
      DO 10 J=1,M
10    X(I,J)=X(I,J)+XI(I,J)
      CALL MM(G,H,GH,N,S,N)
      DO 11 I=1,N
      DO 11 J=1,N
11    IGH(I,J)=S7(I,J)-GH(I,J)
      CALL MM(IGH,P,PUP,N,N,N)
      CALL MM(F,X,XPR,N,N,M)
      CALL MM(F,PUP,FPUP,N,N,N)
      CALL MM(FPUP,FT,FPUPFT,N,N,N)
      DO 13 I=1,N
```

```
      DO 13 J=1,N
13    PPR(I,J)=FPUPFT(I,J)+Q(I,J)
      WRITE(6,106)Z(1,1),Z(2,1),X(1,1),X(3,1),XPR(1,1),
     *XPR(3,1)
      X(1,1)=XPR(1,1)
      X(2,1)=XPR(2,1)
      X(3,1)=XPR(3,1)
      X(4,1)=XPR(4,1)
      DO 14 I=1,N
      DO 14 J=1,N
14    P(I,J)=PPR(I,J)
106   FORMAT(6(F8.1))
999   CONTINUE
      STOP
      END
      SUBROUTINE MM(A,B,C,N1,N2,N3)
      REAL*4 A(N1,N2),B(N2,N3),C(N1,N3)
      DO 100 I=1,N1
      DO 100 K=1,N3
      C(I,K)=0.
      DO 100 J=1,N2
100   C(I,K)=C(I,K)+A(I,J)*B(J,K)
      RETURN
      END
```

APPENDIX B

B.O.T NONMANEUVERING TARGET SIMULATION PROGRAM.

```
      REAL*8 P(4,4),H(1,4),HT(4,1),Q(4,4),AX,
     *XA(1,1),YA(1,1),
     *RI(1,1),VV(4000,4),RR(4000),S7(4,4),
     *GRT(4,4),VXA(1,1),VYA(1,1)
     *,V(1,4000),Z(1,1),Y(1,1),S1(4,1),
     *S2(1,1),XI(4,1),TT,X1,X3,E1,UU,
     *HH(1,1),HI(4,1),X(4,1),G(4,1),
     *S6(1,1),GY(4,1),HX(1,1),TY,W,
     *GHX(4,1),F(4,4),FT(4,4),XPR(4,1),
     *PPR(4,4),FP(4,4),GT(1,4),R(1,1),
     *GH(4,4),IGH(4,4),IGHT(4,4),PUP(4,4),
     *IGHP(4,4),IGHPT(4,4),GR(4,1)
      INTEGER N,M,NN,NR,L,KK,I,J,K,NS
C    *
      N=4
      M=1
      NN=1
      W=3.D0
      NR=2000
      NS=4000
      SB=.1D0/57.295779D0
      SX=50. D0
      SY=50. D0
      SVX=1. D0
      SVY=1. D0
      DS=211133.D0
      DS1=333333.D0
      HHH=0.D0
C
      DO 1 J=1,N
```

```
      DO 1 I=1,N
      Q(I,J)=0.D0
1     F(I,J)=0.D0
      F(1,1)=1.D0
      F(1,2)=1.D0
      F(2,2)=1.D0
      F(3,3)=1.D0
      F(3,4)=1.D0
      F(4,4)=1.D0
      DO 2 I=1,N
      DO 2 J=1,N
2     FT(I,J)=F(J,I)
C
C     GENERATION AND STORAGE OFF I.C.NOISE
      DO 10 I=1,N
      CALL GGNML (DS,NS,RR)
      DO 10 J=1,NN
10    VV(J,I)=RR(J)
C
C     MAKE MATRIX S7=IDENTITY.
      DO 11 I=1,N
      DO 11 J=1,N
11    S7(I,J)=0.D0
      DO 12 I=1,N
12    S7(I,I)=1.D0
C
C  START THE FIRST RUN,INITIAL STATE VALUE.
      DO 99 JJ=1,NN
      X(1,1)=-4000. D0
      X(2,1)=+12.D0
      X(3,1)=5000.D0
      X(4,1)=2.D0
C
C DEFINE R AND RI MATRICES.
      R(1,1)=SB**2
```

```
      RI(1,1)=1.D0/R(1,1)
C INITIALIZE P MATRIX
      DO 13 I=1,N
      DO 13 J=1,N
13    P(I,J)=0.D0
      P(1,1)=SX**2
      P(2,2)=SVX**2
      P(3,3)=SY**2
      P(4,4)=SVY**2
C     GENERATION OFF MEASUREMENT NOISE - STORAGE.
      DO 14 I=1,M
      CALL GGNML(DS1,NS,RR)
      DO 14 J=1,NR
14    V(I,J)=RR(J)
C
C  TIME EVOLUTION
C
      DO 999 KK=1,NR
      T=DFLOAT(KK)
      L=KK-1
      TT=DFLOAT(L)
C  GENERATION OFF MEASUREMENT DATA
      X1=-5000.D0+10.D0*TT
      X3=8000.D0+2000.D0*DCOS(0.035 D0*TT)
      UU=X1/X3
      Z(1,1)=DATAN2(X1,X3)
C     ADD NOISE
      Y(1,1)=Z(1,1)+SB*V(1,KK)
C PROJECTION OF X: XPR = X(K+1/K) = F * X(K/K)+ D*U
      CALL MM(F,X,XPR,N,N,M)
C
      AX=0.D0
      VY=-70.D0*DSIN(0.035D0*(TT+0.D0))
      AY=-2.45 D0*DCOS(0.035D0*TT)
```

```
      D(1,1)=0.D0
      D(2,1)=0.D0
      D(3,1)=AY/2.0 D0
      D(4,1)=AY
C
      DO 77 I=1,N
      DO 77 J=1,M
77    XPR(I,J)=XPR(I,J)+D(I,J)
C
C PROJECTION OF P : PPR = P(K+1/K) = F * P(K/K) * FT + Q
      CALL MM(F,P,FP,N,N,N)
      CALL MM(FP,FT,PPR,N,N,N)
      DO 78 I=1,N
      DO 78 J=1,N
78    PPR(I,J)=PPR(I,J)+Q(I,J)
      X(1,1)=XPR(1,1)
      X(2,1)=XPR(2,1)
      X(3,1)=XPR(3,1)
      X(4,1)=XPR(4,1)
C
      DO 68 I=1,N
      DO 68 J=1,N
68    P(I,J)=PPR(I,J)
C   H- MATRIX
      U=X(1,1)**2+X(3,1)**2
      H(1,1)=X(3,1)/U
      H(1,2)=0.D0
      H(1,3)=-X(1,1)/U
      H(1,4)=0.D0
C H- TRANSPOSE MATRIX
      HT(1,1)=H(1,1)
      HT(2,1)=H(1,2)
      HT(3,1)=H(1,3)
      HT(4,1)=H(1,4)
C MEASUREMENT UPDATING
```

```
C   COMPUTATION OF GAIN MATRIX G=P*HT/(H*P*HT+R)
       CALL MM(P,HT,S1,N,N,M)
       CALL MM(H,S1,S2,M,N,M)
       DO 23 I=1,M
       DO 23 J=1,M
23     S2(I,J)=S2(I,J)+R(I,J)
       S6(1,1)=1.D0/S2(1,1)
       CALL MM(S1,S6,G,N,M,M)
C
       GT(1,1)=G(1,1)
       GT(1,2)=G(2,1)
       GT(1,3)=G(3,1)
       GT(1,4)=G(4,1)
C ERROR COVARIANCE MATRIX UPDATE:
C P(K+1/K+1) = {I-G*H}*P(K+1/K)
       CALL MM(G,H,GH,N,M,N)
       DO 73 I=1,N
       DO 73 J=1,N
73     IGH(I,J)=S7(I,J)-GH(I,J)
       CALL MM(IGH,P,IGHP,N,N,N)
       DO 75 I=1,N
       DO 75 J=1,N
75     PUP(I,J)=IGHP(I,J)
       DO 76 I=1,N
       DO 76 J=1,N
76     P(I,J)=PUP(I,J)
C STATE UPDATE AT MEASUREMENT
C      X(+)=X(-)+G*(Y-H(X(-))) !! BUT FOR E.K.F.
       CALL MM(H,X,HX,M,N,M)
       HH(1,1)=Y(1,1)-DATAN2(X(1,1),X(3,1))
       CALL MM(G,HH,XI,N,M,M)
       DO 80 I=1,N
       DO 80 J=1,M
80     X(I,J)=X(I,J)+XI(I,J)
       E1=DSQRT((X1-X(1,1))**2+(X3-X(3,1))**2)
```

```
      E2=(X1-X(1,1))
      E3=(X3-X(3,1))
      TY=X(4,1)-VY-AY
      WRITE (6,107)T,E2,E3,E1,X(2,1),TY
107   FORMAT(6(3X(F14.4)))
999   CONTINUE
99    CONTINUE
      STOP
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE MM(A,B,C,N1,N2,N3)
      REAL*8 A(N1,N2),B(N2,N3),C(N1,N3)
      DO 100 I=1,N1
      DO 100 K=1,N3
      C(I,K)=0.D0
      DO 100 J=1,N2
100   C(I,K)=C(I,K)+A(I,J)*B(J,K)
      RETURN
```

APPENDIX C

B.O.T MANEUVERING TARGET SIMULATION PROGRAM.

```
C
 REAL*8 P(4,4),H(1,4),HT(4,1),Q(4,4),AX,AY,
*D(4,1),XA(1,1),YA(1,1),RI(1,1),VV(4000,4),
*RR(4000),S7(4,4),GRT(4,4),VXA(1,1),VYA(1,1),
*V(1,4000),Z(1,1),Y(1,1),S1(4,1),S2(1,1),
*XI(4,1),TT,X1,X3,E1,UU,HH(1,1),HI(4,1),X(4,1),
*G(4,1),S6(1,1),GY(4,1),HX(1,1),TY,HHH,W,
*GHX(4,1),F(4,4),FT(4,4),XPR(4,1),PPR(4,4),
*FP(4,4),GT(1,4),R(1,1),GH(4,4),IGH(4,4),
*IGHT(4,4),PUP(4,4),IGHP(4,4),IGHPT(4,4),GR(4,1)
  INTEGER N,M,NN,NR,L,KK,I,J,K,NS,HHHH,HHHHH
C    *
      N=4
      M=1
      NN=1
CCCCCCCCCCCCCCCCCC
      W=1.D0
CCCCCCCCCCCCCCCCCC
      NR=2000
      NS=4000
CCCCCCCCCCCCCCCCCC
      SB=.1D0/57.295779D0
CCCCCCCCCCCCCCCCCC
      SX=50. D0
      SY=50. D0
      SVX=1. D0
      SVY=1. D0
      DS=211133.D0
      DS1=333333.D0
      HHH=0.D0
```

```
C
      DO 1 J=1,N
      DO 1 I=1,N
      Q(I,J)=0.D0
1     F(I,J)=0.D0
      F(1,1)=1.D0
      F(1,2)=1.D0
      F(2,2)=1.D0
      F(3,3)=1.D0
      F(3,4)=1.D0
      F(4,4)=1.D0
      DO 2 I=1,N
      DO 2 J=1,N
2     FT(I,J)=F(J,I)
C
C     GENERATION AND STORAGE OFF I.C.NOISE
      DO 10 I=1,N
      CALL GGNML (DS,NS,RR)
      DO 10 J=1,NN
10    VV(J,I)=RR(J)
C     MAKE MATRIX S7=IDENTITY.
      DO 11 I=1,N
      DO 11 J=1,N
11    S7(I,J)=0.D0
      DO 12 I=1,N
12    S7(I,I)=1.D0
C  START THE FIRST RUN,INITIAL STATE VALUE.
      DO 99 JJ=1,NN
      X(1,1)=-4000. D0
      X(2,1)=+12.D0
      X(3,1)=5000.D0
      X(4,1)=2.D0
C DEFINE R AND RI MATRICES.
      R(1,1)=SB**2
      RI(1,1)=1.D0/R(1,1)
```

```
C INITIALIZE P MATRIX
      DO 13 I=1,N
      DO 13 J=1,N
13    P(I,J)=0.D0
      P(1,1)=SX**2
      P(2,2)=SVX**2
      P(3,3)=SY**2
      P(4,4)=SVY**2
C     GENERATION OFF MEASURMENT NOISE - STORAGE.
      DO 14 I=1,M
      CALL GGNML(DS1,NS,RR)
      DO 14 J=1,NR
14    V(I,J)=RR(J)
C  TIME EVOLUTION
      DO 999 KK=1,NR
      T=DFLOAT(KK)
      L=KK-1
      TT=DFLOAT(L)
C  GENERATION OFF MEASURMENT DATA
      X1=-5000.D0+10.D0*TT
      X3=8000.D0+2000.D0*DCOS(0.035 D0*TT)+2000.D0
      IF (KK.LT.700) GO TO 33
      X1=-100.D0+3.D0*TT
      X3=8000.D0+2000.D0*DCOS(0.035 D0*TT)-10.D0*TT+9000.D0
      IF (KK.LT.1400.AND.KK.GE.700) GO TO 33
      X1=46100.D0-30.D0*TT-6500.D0+6500.D0
      X3=8000.D0+2000.D0*DCOS(0.035 D0*TT)+7000.D0-12000.D0
33    CONTINUE
      Z(1,1)=DATAN2(X1,X3)
C     ADD NOISE
      Y(1,1)=Z(1,1)+SB*V(1,KK)
C PROJECTION OF X: XPR = X(K+1/K) = F * X(K/K)+ D*U
      CALL MM(F,X,XPR,N,N,M)
      AX=0.D0
      VY=-70.D0*DSIN(0.035D0*(TT+0.D0))
```

```
      AY=-2.45 D0*DCOS(0.035D0*TT)
      D(1,1)=0.D0
      D(2,1)=0.D0
      D(3,1)=AY/2.0 D0
      D(4,1)=AY
C
      DO 77 I=1,N
      DO 77 J=1,M
77    XPR(I,J)=XPR(I,J)+D(I,J)
C PROJECTION OF P : PPR = P(K+1/K) = F * P(K/K) * FT + Q
      CALL MM(F,P,FP,N,N,N)
      CALL MM(FP,FT,PPR,N,N,N)
      IF (KK.LT.600.) GO TO 86
      IF (KK.GT.600.AND.HHHHH.LT.1.) GO TO 66
      IF (KK.GT.600.AND.HHHHH.GT.1.) GO TO 67
66    DO 79 I=1,N
      DO 79 J=1,N
79    Q(I,J)=0.D0
      GO TO 86
67    Q(1,1)=.25 D0*W
      Q(1,2)=.5 D0*W
      Q(2,1)=.5 D0*W
      Q(2,2)=W
      Q(3,3)=.25 D0*W
      Q(3,4)=.5 D0*W
      Q(4,3)=.5 D0*W
      Q(4,4)=W
86    CONTINUE
      DO 78 I=1,N
      DO 78 J=1,N
78    PPR(I,J)=PPR(I,J)+Q(I,J)
      X(1,1)=XPR(1,1)
      X(2,1)=XPR(2,1)
      X(3,1)=XPR(3,1)
      X(4,1)=XPR(4,1)
```

```
C
      DO 68 I=1,N
      DO 68 J=1,N
68    P(I,J)=PPR(I,J)
c  H- MATRIX
      U=X(1,1)**2+X(3,1)**2
      H(1,1)=X(3,1)/U
      H(1,2)=0.D0
      H(1,3)=-X(1,1)/U
      H(1,4)=0.D0
C H- TRANSPOSE MATRIX
      HT(1,1)=H(1,1)
      HT(2,1)=H(1,2)
      HT(3,1)=H(1,3)
      HT(4,1)=H(1,4)
C UPDATING AT MEASUREMENT
C   COMPUTATION OF GAIN MATRIX G=P*HT/(H*P*HT+R)
      CALL MM(P,HT,S1,N,N,M)
      CALL MM(H,S1,S2,M,N,M)
      DO 23 I=1,M
      DO 23 J=1,M
23    S2(I,J)=S2(I,J)+R(I,J)
      S6(1,1)=1.D0/S2(1,1)
      CALL MM(S1,S6,G,N,M,M)
      GT(1,1)=G(1,1)
      GT(1,2)=G(2,1)
      GT(1,3)=G(3,1)
      GT(1,4)=G(4,1)
C    ERROR   COVARIANCE    MATRIX   UPDATE:P(K+1/K+1)      =
{I-G*H}*P(K+1/K) {      }
      CALL MM(G,H,GH,N,M,N)
      DO 73 I=1,N
      DO 73 J=1,N
73    IGH(I,J)=S7(I,J)-GH(I,J)
      CALL MM(IGH,P,IGHP,N,N,N)
```

```
      DO 75 I=1,N
      DO 75 J=1,N
75    PUP(I,J)=IGHP(I,J)
      DO 76 I=1,N
      DO 76 J=1,N
76    P(I,J)=PUP(I,J)
C STATE UPDATE AT MEASURMENT
C     X(+)=X(-)+G*(Y-H(X(-))) !! BUT FOR E.K.F.
      CALL MM(H,X,HX,M,N,M)
      HH(1,1)=Y(1,1)-DATAN2(X(1,1),X(3,1))
      HHH=HH(1,1)*150.DO
      HHHH=SNGL(HHH)
      HHHHH=IABS(HHHH)
      CALL MM(G,HH,XI,N,M,M)
      DO 80 I=1,N
      DO 80 J=1,M
80    X(I,J)=X(I,J)+XI(I,J)
      E1=DSQRT((X1-X(1,1))**2+(X3-X(3,1))**2)
      E2=(X1-X(1,1))
      E3=(X3-X(3,1))
      TY=X(4,1)-VY
      TX=X(2,1)+10.DO
C     WRITE (6,107)T,E2,E3,E1,X(2,1),X(4,1)
C     WRITE (6,107)T,X1,X3
      WRITE (6,107)T,TX,TY,HH(1,1)
107   FORMAT(6(3X(F14.4)))
999   CONTINUE
99    CONTINUE
      STOP
      END
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      SUBROUTINE MM(A,B,C,N1,N2,N3)
      REAL*8 A(N1,N2),B(N2,N3),C(N1,N3)
      DO 100 I=1,N1
      DO 100 K=1,N3
```

```
      C(I,K)=0.D0
      DO 100 J=1,12
100   C(I,K)=C(I,K)+A(I,J)*B(J,K)
      RETURN
      END
      END
```

APPENDIX D

PSEUDO-LINEAR FORMULATION

If we will start we the Cartesian formulation equations:

$$\beta(k) = h[x(k)] + n(k) \qquad (D.1)$$

$$h[x(k)] = \arctan[x_1(k)/x_3(k)] \qquad (D.2)$$

$$E\,[n(k)] = 0 \qquad (D.3)$$

$$E\,[n(i)n(j)] = \begin{cases} \sigma^2(k) & i = j \\ 0 & i \neq j \end{cases} \qquad (D.4)$$

then after algebraical manipulations yield:

$$0 = \hat{H}(k)x(k) + R(k)n(k) \qquad (D.5)$$

where:

$$\hat{H}(k) = [\cos\beta(k),\ -\sin\beta(k),\ 0,\ 0] \qquad (D.6)$$

$$R(k) = \sqrt{x_1{}^2(k) + x_3{}^2(k)} \qquad (D.7)$$

The nonlinearity has been embeded in the measurement noise. If

$\varepsilon(k)=R(k)n(k)=$effective measurement noise at time kT (D.8) it can been shown [Ref. 13]. that $\varepsilon(k)$ has the following statistics:

$$E\left[\varepsilon(k)\right] = 0 \tag{D.9}$$

$$E\left[\varepsilon^2(k)\right] = R^2(k)\sigma^2(k) \tag{D.10}$$

Finally the pseudo-linear model is analogous to that of Cartesian formulation model with the following modifications:

1. replacing $H(k)$ with that given by equation (C.6)
2. replacing $\sigma(k)$ with $R(k/k-1)\sigma(k)$
3. replacing $\beta(k)-h[x(k/k-1)]$ with $-\hat{H}(k)x(k/k-1)$ '

Detailed analysis on the subject can be found in [Ref. 8].

LIST OF REFERENCES

1. Kalman,R.E, "A New Approach to Linear Filtering and Prediction Problems". Trans. of the ASME - J. of Basic Engr. 35-45 (March 1960).

2. Brown,R.G. Introduction to Random Signal Analysis and Kalman Filtering. John Wiley & Sons,inc. 1983.

3. Gelb,A., Applied Optimal Estimation. The Analytic Sciences Corporation, 1974.

4. Kirk D.E.,"Optimal Estimation: An Introduction to the Theory and Applications", Mimeographed Notes for Course EE4413, Naval Postgraduate School, Monterey, CA. Spring 1984.

5. Jazwinski,A.H., Stochastic Processes and Filtering Theory. Academic Press, 1970.

6. Aidala,V.J.,and Hammel,S.E. "Utilization of Modified Polar Coordinates for Bearings-Only Tracking". IEEE Trans. Automat. Contr. vol.AC-28,No.3,March 1983.

7. Hwang,C.S. and Mohler,R.R., "Nonlinear Observability and Mixed-Coordinate Bearings-Only Signal Processing". Department of Electrical and Computer Engineering Oregon State University Corvallis, OR, to be published.

8. Aidala,V.J., "Behavior of the Kalman Filter Applied to Bearings-Only Target Motion Analysis". Naval Underwater Systems Center. Newport,Rhode Island.

9. Murphy,D.J., Noisy Bearings-Only Target Motion Analysis. Ph.D dissertation Dep.Elec.Eng., Northeastern Univ.Boston, MA, 1970.

10. Bar-Shalom, Y. and Bitminal, K., "Variable Dimension Filter for Maneuvering Target Tracking," IEEE T-AES., September 1982.

11. Singer,R.A., "Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets," IEEE T-AES., July 1970.

12. Martzoucos, V., Digital Computer Applications of Kalman Filter in Target Tracking. Master's Thesis; Naval Postgraduate School, Monterey Ca, June 1984.

13. NUSC Technical Report 4934, Behavior of the Kalman Filter Applied to Bearings-Only Motion Analysis, November 1976.

BIBLIOGRAPHY

Brown, R.G., Introduction to Random Signal Analysis and Kalman Filtering. John Wiley & Sons,Inc. 1983.

Gelb,A., Applied Optimal Estimation, The M.I.T. Press, July 1982.

Jazwinski, A. H., Stochastic Processes and Filtering Theory. Academic Press 1970.

Kirk, D.E, "Optimal Estimation: An Introduction to the Theory and Applications."Notes for course EE4413, Naval Postgraduate School, Monterey, Ca. Spring 1984. N.P.S.,Spring 1984.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Library,code 0142 Naval Postgraduate School Monterey, California 93943	2
2.	Department Chairman, Code 62 Department of Electrical and Computer Engineering Naval Postgraduate School Monterey,California 93943	1
3.	Professor H. Loomis,Code 62Lm Department of Electrical Engineering Naval Postgraduate School Monterey, California 93943	4
4.	Professor H. Titus, Code 62Ts Department of Electrical Engineering Naval Postgraduate School Monterey, California 93943	1
5.	Hellenic Navy General Staff Education Department Stratopedon Papagou Holargos Athens, GREECE	3
6.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
7.	LT. D. Kourkoulis, H.N. Argyrocastrou 84 Papagou Athens, GREECE.	5

2[illegible]2602

Thesis
K834623 Kourkoulis
c.1 Bearings only target tracking - maneuvering target.