AD-A270 713

AFIT/GE/ENG/93S-06

COMPARISON OF A
DISTRIBUTED KALMAN FILTER
VERSUS A CENTRALIZED KALMAN FILTER
WITH FAULT DETECTION CONSIDERATIONS

THESIS

Paul J. Lawrence Jr.
Captain, USAF
AFIT/GE/ENG/93S-06

DTIC
ELECTE
S OCT 12 1993
B D

DEPARTMENT OF THE AIR FORCE

AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GE/ENG/93S-06

COMPARISON OF A

DISTRIBUTED KALMAN FILTER

VERSUS A CENTRALIZED KALMAN FILTER

WITH FAULT DETECTION CONSIDERATIONS

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 of Electrical Engineering

Paul J. Lawrence Jr., B.S. Electrical Engineering

Captain, USAF

July, 1993

Approved for public release; distribution unlimited

Preface

The purpose of this study was to verify by comparison the contention of relative estimation comparability between the centralized filter architecture and the distributed filter architecture, specifically the federated filter design. Along with this contention, it was proposed that the federated filter offers better fault tolerance. Further, it was necessary to establish a test environment which would allow this comparison to take place, and would also allow further distributed filtering testing to occur in the future.

Extensive study of the limited amount of federated filter documentation was performed. The limitations on the availability of documentation occurred because the federated filter design is a relatively new filter concept. Comprehensive testing was accomplished using the available simulation software, DKFSIM Version 1.1, developed by Integrity Systems, Incorporated. The simulation software was also a relatively new development, resulting in a limited amount of software documentation. Consequently, a protracted learning curve was necessary to become familiar with all aspects of this project. It is hoped that the work in this area can be continued so that this new filter design might be implemented in today's aircraft and possibly future aircraft.

Several circumstances occurring during the performance of this project required that I seek the assistance of others. I am deeply grateful to my faculty advisor, Lt Col Robert Riggins, board committee member, Dr. Peter Maybeck, project consultant, Dr. Neal Carlson, and program manager, Mike Berarducci, for their unending patience and consideration in times of need. Finally, I would like to thank my wife Linda and my children for their help and understanding when I was very often consumed by this enormous task.

DTIC QUALITY INSPECTED 2

Availability Codes

Dist	Avail and/or Special

Table of Contents

List of Figures

List of Tables

Abstract

This project examined the results obtained by simulating an aircraft navigation system with a partial complement of a typical avionics sensor array. Two different techniques of estimation processes were utilized and compared: the conventional Kalman and the federated filter architectures.

Areas of interest include error state estimation accuracy and overall performance, residual behavior under normal and induced sensor failure conditions, and potential for failure detection and isolation. Several simulations were accomplished for each filter design and the results were compared in order to verify the validity of the newly developed federated filter architecture.

Comparison of the error state estimation accuracies of the two filter designs revealed excellent overall performances for both. The identification of failures showed a definite advantage in the federated filter design. Having sensor-dedicated local filters allowed for easy sensor failure identification for the federated filter, while the centralized filter design suffered from navigation solution corruption under the same circumstances.

Once established as a valuable estimation technique, the federated filter will add significantly to the viable alternatives when choosing a particular filter architecture for current avionics modifications or future avionics implementations. The federated filter may indeed prove to be the most effective filter design overall, under any circumstances.

COMPARISON OF A

DISTRIBUTED KALMAN FILTER

VERSUS A CENTRALIZED KALMAN FILTER

WITH FAULT DETECTION CONSIDERATIONS

I. *Introduction*

The centralized Kalman filter is the most common filter design implemented in the integrated navigation systems of United States Air Force aircraft today. A centralized Kalman filter receives all available measurements and combines all the information contained in those measurements to obtain an optimal navigation solution. For simple, well-modeled linear systems, the centralized Kalman filter is unquestionably the optimal estimator (19). When considering tradeoffs of data flow, algorithmic requirements, and processing speed versus optimality, fault tolerance, estimation in a multi-sensor environment is often best treated as a distributed estimation problem (19). The distributed filter architectures employ a bank of local Kalman filters dedicated to the sensors which provide measurement information to the system (19). A master filter combines the estimates from the bank of filters to obtain a typically suboptimal navigation solution. This poses less of a computational burden than a centralized filter implementation. Although these estimates are typically suboptimal, the distributed filter offers improvements over the centralized fault detection and isolation schemes. (1,2,3,19)

1-1

The federated filter is a variation of the distributed filter theory currently attracting the interest of the United States Air Force. The advantage of the federated filter architecture is obtained through the sharing of the estimation information by the local sensor-dedicated filters. The recombination of this shared information in the master filter significantly improves the quality of the error state estimates over previous distributed filter designs (1,3,19). Figure 1.1 depicts the common structure of a federated filter architecture application to a multi-sensor navigation system.
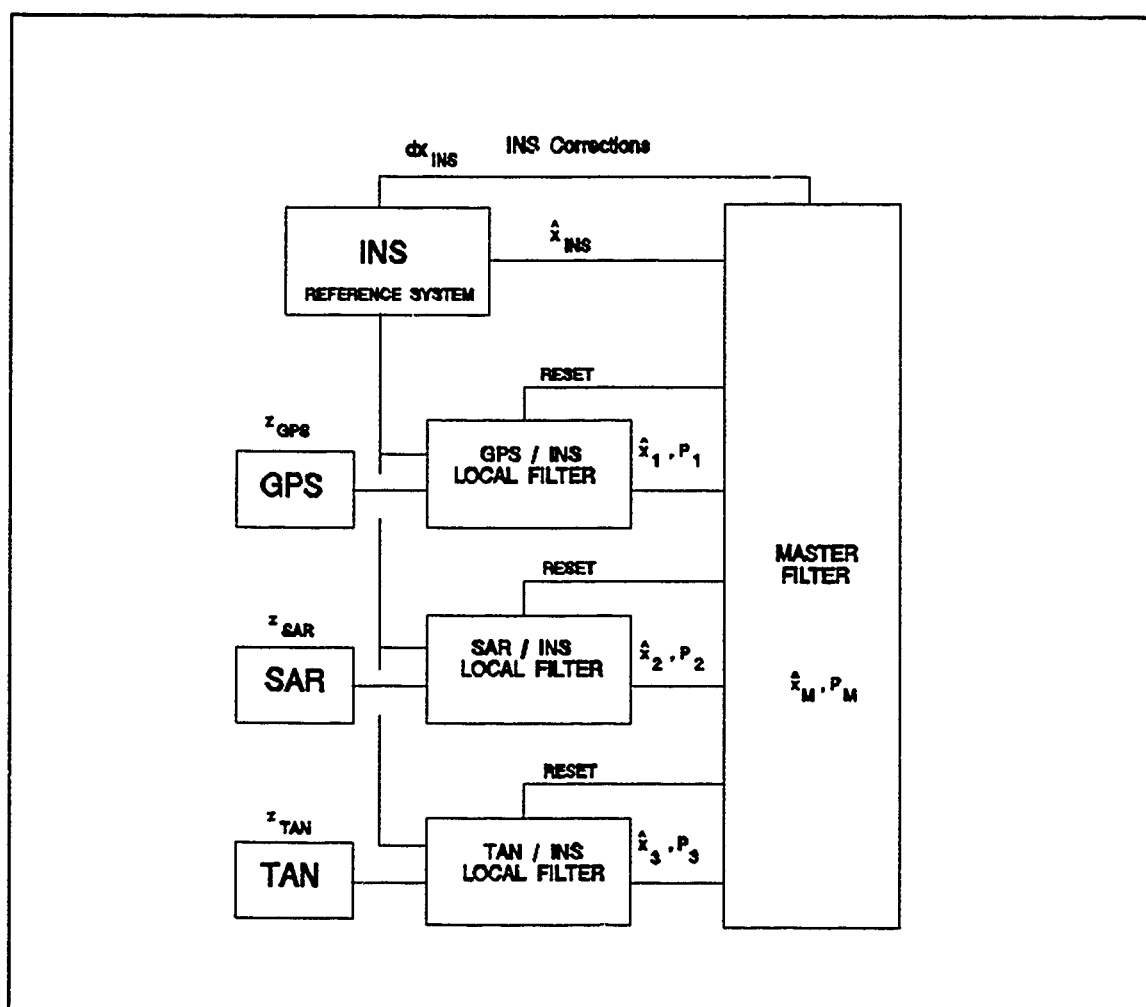
Figure 1.1 Federated Filter Application to a Multi-Sensor Navigation System

From the block diagram in Figure 1.1, the master filter provides the reset information to the sensor-dedicated local filters and the correction information to the Inertial Navigation System (INS). The local filters receive the measurements directly from the sensors, and then provide the error state information to the master filter for recombination. The INS serves as the reference system for the local filters and the master filter. While the Global Positioning System (GPS), Synthetic Aperture Radar (SAR), and Terrain Aided Navigation (TAN) systems are shown in this diagram, the federated architecture can handle larger numbers of sensor-dedicated filters than this.

1.1 Background

The Wright Laboratory, Avionics Directorate worked closely with TAU Corporation on the completion of the Common Kalman Filter development program. This program's prime objective has been to establish a basic set of estimation and fault detection, isolation and reconfiguration (FDIR) system design techniques. (19) Advantages over conventional filtering techniques were brought about by partitioning the centralized Kalman filter into several local filters feeding a single, master filter (1,3,19).

The typical distributed filter designs have been characterized as suffering from the cascading effect, which is described briefly by filter-driven-filter instabilities (1). The federated filter design was deemed the most auspicious choice for implementation of the possible distributed filter designs because it is atypical and does not suffer from those instability difficulties (19).

A means of evaluating this relatively new design was necessary to justify its use in future navigation systems. Integrity Systems Inc. was

contracted to provide simulation software for efficiently evaluating distributed Kalman filter architectures. DKFSIM, Version 1.1 (10) has been presented to Wright Laboratory, Avionics Directorate, Wright-Patterson AFB, OH as the intended simulation tool. Wright Laboratory, Navigation and Information Transmission Branch is the Office of Primary Responsibility (OPR) for the DKFSIM simulation software and the sponsor for this thesis effort. The final version is deliverable by July, 1993 and includes several changes not available in the current version. This thesis does not involve the use of the proposed updated version because of the timing of the delivery of the final version of the software simulation tool.

DKFSIM Version 1.1 is currently used by the Air Force to evaluate distributed filter architectures and compare their performance to equivalently modeled centralized filters. However, as stated earlier, the current version is incomplete as a totally comprehensive evaluation package. It is capable of providing a means of testing the estimation accuracy of the filter, but is missing the fault detection and isolation (FDI) evaluation portion, and the capability to test the federated filter design in all of its reset modes.

There are currently two of the four proposed reset modes operational in DKFSIM Version 1.1. The four reset modes encompass the information sharing possibilities. They are the full fusion-reset mode, the partial fusion-reset mode, the zero-reset mode, and the no-reset mode (1,3,7). The feedback mode selected determines the relative amount of information fed back to the local filters from the master filter. The full and partial fusion-reset modes involve feedback of all or part of the master filter fused navigation solution. The zero-reset mode involves no feedback from the master filter, but the local filters retain none of the local information. The no-reset mode

involves no feedback from the master filter, and each local filter retains its own unique portion of the total system information. (1,3,7) The two federated filter modes enabled in DKFSIM are the no-reset mode and the full fusion-reset mode. In this thesis, we focus on the no-reset mode for reasons stated below.

1.2 Problem Definition

In order to verify the quality of the federated filter estimation accuracy, it is prudent to compare the federated filter error state outputs to that of the centralized filter. Another advantage in performing this comparison is that the software simulation tool, DKFSIM Version 1.1, can be evaluated.

Error state estimation accuracy can be determined by the state history because it provides a measure of how accurate the filter's estimates are, compared to the real world model. The error state filter formulation used in DKFSIM Version 1.1 estimates the errors in the navigation and attitude information using the difference between the INS and external source information. The dynamics upon which the filter is based is the set of inertial system error propagation equations, which are relatively well developed, well behaved, low frequency, and very adequately represented as linear. Because the filter is based on low frequency dynamics, its sample rate can be much lower than that of a direct filter. For these reasons, the error state space formulation is used in essentially all terrestrial aided inertial navigation systems. (13) These error state outputs are computed under normal operating conditions for a typical flight profile.

This thesis concentrates on a federated filter in the no-reset mode (1,3,7) with three local filters operating independently on five different

1-5

sets of sensor output data. The reference system is the Inertial Navigation System (INS), and the four external sensor sources are the Global Positioning System (GPS), the Synthetic Aperture Radar (SAR), the Terrain Aided Navigation (TAN), and the Barometric Pressure Altimeter (BARO) (1,2,3,7). A centralized Kalman filter with identical sensor model representations as is used for the federated filter is selected for this comparison.

The fault detection aspects of this thesis effort are primarily centered around monitoring and comparing the residual outputs provided by each of the independent local filters and the centralized filter. The residual sequence has been shown to be a white Gaussian sequence of zero-mean with covariance being a function of the observation matrix $H(t_i)$, the error state covariance $P(t_i^-)$, and the measurement process noise $R(t_i)$:

$$E\{r(t_i)r^T(t_i)\} = H(t_i)P(t_i^-)H^T(t_i) + R(t_i) \qquad \textbf{(1.1)}$$

This is a proper representation of the residual process because the model upon which the filter is based accurately depicts the real world behavior. During operation of the filter, the actual residual sequence can be monitored and compared to this description. If the description appears to be violated consistently, then one can deduce that something has occurred to invalidate the model within the filter. Otherwise, if the violation occurs in only one component of a vector residual process, then, in some cases, it can be assumed that the measuring device generating that particular residual component is the source of the difficulty. (13)

Thus, monitoring the residuals of the centralized filter and those at the local filter level in the federated design has the potential to provide fault detection information. Further consideration should be given to master filter fusion residual monitoring in the federated filter design, which is not within the scope of this thesis.

Finally, the goals of this thesis require a comparison of the overall estimation performances between the federated filter design and the well-established centralized Kalman filter design, along with a comparison of their respective FDI capabilities. These are the questions we want to address.

1.3 Research Objectives

The driving motivation for this research is to obtain an objective comparison revealing any possible advantages of the federated architecture and the centralized architecture over one another. This thesis initially concentrates on comparing the estimation accuracy of the two designs. Accuracy is the most crucial aspect since further thesis work is justifiable if the federated filter's performance compares well to the centralized filter's performance. An error state accuracy comparison between that of the federated filter and of the centralized filter should indicate that under normal operating conditions, the centralized filter cannot be out-performed by the federated filter. However, the FDIR capabilities of the federated filter is the area where the largest disparity is expected to occur in the performances of the two designs, with substantial advantage being given to the federated filter design.

The following objectives are based on the statement of the problem and the desired results in order to validate the federated filter's performance

and to facilitate the implementation of DKFSIM Version 1.1 at AFIT for future
thesis efforts:

1. Load DKFSIM Version 1.1 onto a local computer network, meet system
requirements, and perform trial runs for debugging. Investigate DKFSIM
Version 1.1 and its associated documentation to determine the existing state
dimensions of each local filter, the order of the master filter, the sensor
truth models, the sensor filter models, the master filter algorithm, the local
filter algorithm, and the implementation of the filter computations. Ensure
that the centralized and distributed simulations are based on identical
scenarios, such as flight profile and random number seed. Clean up output
data files to facilitate the plotting program's interface compatibility.

2. Perform a centralized filter analysis based on the preceding information.
Include a ten-run Monte Carlo analysis of all available error state outputs to
facilitate the initial phase of the comparison. Obtain baseline residual
output plots which represent ordinary magnitude residuals under normal
operating conditions for the next phase of the comparison. Simulate a hard
failure and a soft failure in the sensor measurements in order to evaluate the
impact on residual output plots of the centralized filter. Perform a chi-
squared test on the residual outputs with induced failures so that a relative
comparison can be obtained regarding fault detection.

3. Perform a federated filter analysis with the exact same truch and filter
model states as was used for the centralized filter. Compare the ten-run
error state estimation performances of the two filter designs. Compare

baseline residual output plots from the centralized filter and the federated local filters. Simulate identical hard and soft failures in the sensor measurements for the federated filter to allow a direct comparison with the centralized filter. Compare the chi-squared test results from both failed-sensor simulations.

1.4 Research Approach

The largest obstacle in the approach to completing this research has been to become familiar with DKFSIM Version 1.1. There was little documentation available for instructional use. Nearly all the information associated with the simulation tool was found in the headers of each software module. Knowing what the software can accomplish, and what its limitations are, is the key to efficient analysis of the filter designs.

The remainder of this section provides an overview of the approach to completing each of the research objectives outlined in the previous section.

1. The AFIT VAX/VMS system is the current host for DKFSIM 1.1. Sufficient memory allocation was made available for several runs of data storage. Debugging was necessary because rehosting the simulation tool was required for purposes of operation at AFIT. This involved treatment of the separate software modules until all modules were operating cleanly. Comparison of the error state plots from several independent simulations, discussed in Chapter IV and presented in the Appendices, indicates no additional filter tuning was required.

The state dimensions of t' truth and filter models play a large part in the capability and validity of the two filter designs. A concise record

reflecting the state dimensions of the centralized filter, and those of the local and master filters of the federated filter design was a functional necessity. Chapter III addresses the filter error state representations for all of the filters in this comparison.

The master filter algorithm forms the basis for whether or not the distributed filter output is globally optimal (1,2,3,18,19). The centralized filter algorithm coincides with the established linearized Kalman filter theory and is therefore defined as an optimal estimator. Consequently, one can expect similar estimation performances between the two filter designs.

The simulation scenarios were considered identical because the same flight profile was used, the same random number seed was used for initializing the simulations of white noises used in the truth models, and the input data control files provided for comparable simulations. The input data file includes variable settings for the filter implementation, the output data file format, and several truth and filter model parameters. A sample input data file is provided in Appendix J. Several simulations were executed to confirm that selection of a particular random number seed reproduces exactly the same error state outputs from simulation to simulation. The output data files were compared numerically as well as visually by plot generation. The time step selection for the sensor measurement, the filter propagations, and the INS reference system were unchanged for all simulations.

The software was modified so that the output data files were simply columns of numbers. The plotting program available requires an input matrix in strictly three-column format. This limited the output data to only one error state, its standard deviation, and its time line outputs per simulation.

2. An ensemble averaging of the Monte Carlo analyses is not an available alternative for output from DKFSIM Version 1.1. Consequently, ten independent Monte Carlo runs were performed for each error state and were subsequently averaged utilizing the plotting program functions prior to output plotting. Further, DKFSIM Version 1.1 does not compute one-sigma values for every Monte Carlo run. The filter computed one-sigma values are computed for the first Monte Carlo run, and it is then assumed these values are valid for all runs.

Residual outputs are not available in the current version of DKFSIM. The software has been modified such that all residuals computed could be obtained according to which sensor provided the measurements. Three residual output data files are created for every Monte Carlo run; one for the GPS sensor, one for the SAR sensors, and one for the TAN sensor. Induction of the hard failure for both the federated and centralized filter cases consists of simulating the removal of one set of sensor outputs without notifying the central processor of the loss. Simulation of the soft failure consists of adding a ramp bias as a function of time, or a constant bias, to the measurement. The hard and soft failures induced were simulated by affecting the magnitude of the measurement directly prior to residual calculations.

3. Using the models implemented in DKFSIM Version 1.1 for the sensors and the reference system, develop the centralized and distributed truth and filter models. The five sensors are the Synthetic Aperture Radar Precision Velocity Update (SARPVU), Synthetic Aperture Radar Electro-Optical (SAREO), Global Positioning System (GPS), Terrain Aided Navigation (TAN), and Barometric Pressure Altimeter (Baro-Alt) (19). The reference system is a strapdown, medium accuracy, Inertial Navigation System (INS) (1,7,12,18,19).

Comparison of the estimation performances during normal operations depends on observations of error state outputs, state covariance behavior, and residual outputs. It is important to show that the performances of the distributed and centralized filters are nearly equivalent. The comparison of the accuracies is valid only if the sensors and the INS are identically represented in each architecture. (20) Comparison of the impact on the residual outputs for the two different filter designs should reveal the respective fault detection capabilities. A chi-squared test applied to these residuals simply facilitates conclusions based on the fault tolerances of the two filter designs.

1.5 Assumptions

All truth and reduced-order filter models included in DKFSIM Version 1.1 are assumed well researched and correct (10). Verification of the master filter data fusion algorithm is also assumed unnecessary. Because the conservation of information principal is the foundation of the federated filter design, it is assumed to be valid. However, Chapter II offers a detailed theoretical description of the federated filter and the information sharing principle.

1.6 Overview of Thesis

Chapter II provides a detailed description of the centralized, distributed, and federated Kalman filter architectures and theory. A discussion of the DKFSIM Version 1.1 software simulation tool is also provided. Additional topics of interest are the information sharing principle which is imperative to the formulation of the federated filter design, and the chi-squared

algorithm used for fault detection. Chapter III lists the truth and filter model state descriptions, and discusses the fault detection concepts used in this thesis. Chapter IV provides a discussion of the comparison of the estimation performances and fault detection capabilities observed from the centralized and federated filter analyses. Additional information regarding federated filter fault tolerance has been provided by Integrity Systems, Incorporated so that a more comprehensive assessment of that filter's capabilities can be realized. Chapter V contains the conclusions drawn from the work and theory presented in this thesis. Recommendations for further research in this area are also addressed.

II. Filtering and Fault Detection Theory

2.1 Overview

This chapter is intended to refamiliarize the informed reader of the pertinent areas of basic filtering theory and fault detection concepts associated with this thesis. Other major areas covered are the distributed Kalman filter, and the federated Kalman filter. An additional section is provided as an overview of the software simulation tool used in this thesis for filter evaluation and of the fault detection considerations made herein. The reader who is not familiar with Kalman filtering should consult Maybeck's textbooks on stochastic models, estimation, and control (13,14,15). The fault detection scheme is based on monitoring and comparing the residual outputs from the centralized filter and the DKF's local filters. Lastly, this thesis deals only with linear Kalman filter designs so that the verification effort is not unnecessarily complicated. Although the residuals of the DKF's master filter would provide much information, only the local filter residuals are considered in this thesis (4).

2.2 The Centralized Kalman Filter

The centralized Kalman filter architecture has been developed, tested, and sufficiently documented (13,14,15). Regarding this thesis effort, the centralized filter design forms the basis of comparison for the distributed

filter architecture. Therefore, a brief description of the centralized Kalman filter theory is warranted at this time as preparation for further discussion.

The standard centralized Kalman filter processes the data from different systems in one step. The update and propagation equations are generally treated in the discrete time domain for easy implementation by computer (19). The discrete-time model formulation of the Kalman filter algorithm generates the best prediction of the state at time t_i before the measurement at time t_i is processed, and then updates that estimate with the measurement available at time t_i. Figure 2.1 is a block diagram representation of this algorithm (13). To summarize the algorithm briefly, the optimal state estimate is propagated from the measurement at time t_{i-1} to time t_i by the following relations:

$$\hat{x}(t_i^-) = \Phi(t_i, t_{i-1})\hat{x}(t_{i-1}^+) + B_d(t_{i-1})u(t_{i-1}) \qquad \textbf{(2.1)}$$

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

$$+ G_d(t_{i-1})Q_d(t_{i-1})G_d^T(t_{i-1}) \qquad \textbf{(2.2)}$$

To specify a Kalman filter of this form completely, the structure $[\Phi(t_i, t_{i-1}), B_d(t_{i-1}), G_d(t_{i-1}), H(t_i)$ for all times of interest] and the uncertainties $[Q_d(t_{i-1}), R(t_i)$, and the initial state and covariance estimates] must be defined. (13) As such, for the system model, the measurement $z(t_i)$ is given by:

$$z(t_i) = H(t_i)x(t_i) + v(t_i) \qquad \textbf{(2.3)}$$
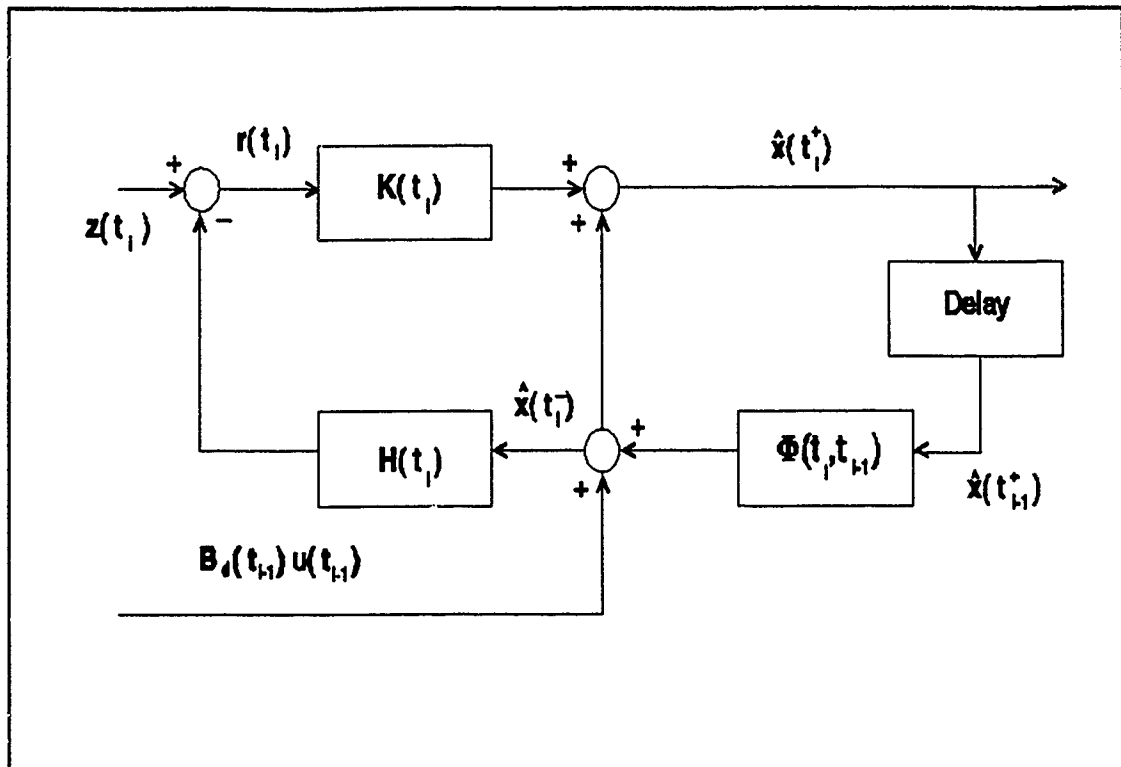
Figure 2.1 Discrete-Time Kalman Filter Block Diagram

At measurement time t_i, the measurement becomes available and includes the measurement process noise, $v(t_i)$, which is assumed independent of $x(t_i)$ and the measurement history, and is assumed Gaussian with mean zero and covariance matrix $R(t_i)$. Let the system state process of the system model satisfy the linear equation

$$\dot{x}(t) = F(t)x(t) + B(t)u(t) + G(t)w(t) \qquad (2.4)$$

where $x(t)$ would be the system state at time t, and $F(t)$ is the system dynamics matrix. Additionally, the state covariance matrix is defined by $P(t_i)$, and the discrete-time propagations are defined using the state transition matrix, $\Phi(t_i, t_{i-1})$.

The matrix, $H(t_i)$, is defined as the measurement matrix for this system. The matrix, $B_d(t_{i-1})$, is the deterministic discrete-time input matrix, with corresponding input control vector, $u(t_i)$, consisting of deterministic control input functions. The matrix, $G_d(t_{i-1})$, is the discrete-time noise input matrix associated with the discrete-time dynamic driving noise, $w_d(t_i)$, of the system with strength $Q_d(t_{i-1})$.

The estimate is then updated by defining the Kalman filter gain relationship $K(t_i)$ and using this relationship in both the mean and covariance equations:

$$K(t_i) = P(t_i^-) H^T(t_i) [H(t_i) P(t_i^-) H^T(t_i) + R(t_i)]^{-1} \qquad (2.5)$$

$$\hat{x}(t_i^+) = \hat{x}(t_i^-) + K(t_i) [z(t_i) - H(t_i) \hat{x}(t_i^-)] \qquad (2.6)$$

$$P(t_i^+) = P(t_i^-) - K(t_i) H(t_i) P(t_i^-) \qquad (2.7)$$

This system model allows generation of the best prediction of what the measurement will be before it is actually taken, $H(t_i)\hat{x}(t_i^-)$. The input to this algorithm is the realized value of the measurement at time t_i. The measurement residual is then generated as the difference between the true measurement value and the best prediction of it before it is actually taken:

$$r(t_i) = z(t_i) - H(t_i) \hat{x}(t_i^-) \qquad (2.8)$$

This term is often referred to as innovations of the algorithm (13).

For this case, the measurement noises for each of the different sensors are assumed independent in time (20). Generally, all measurements taken are sampled, transformed into the reference coordinate frame, Earth-Centered-Earth-Fixed (ECEF), and then transmitted to the central filter for processing. Due to uncertainties in the transformation process, the estimation information transmitted to the central processor contains these additional transformation errors (8,9). The processing and communication resources dictate the specific design. This is common to both the centralized and distributed architectures. However, since communication between processors is so critical, it is important to determine the relative merits of transmitting raw data between the nodes of the filter structures for comparison. (19)

The computational burden is approximately cube-proportional to the dimension of the model and the storage requirement is approximately square-proportional (13). Although diminished computational burden per filter is not the prime motivation for consideration of the distributed filter architecture, it is certainly a distinct advantage for real-time operations.

2.3 The Distributed Kalman Filter

This design has several variations, but, in general, it embodies the background for this thesis effort. The distributed filter is a two-stage data processing algorithm versus the single stage of the centralized filter. The first stage requires the local filters to arrive at independent local estimates (20).

A block diagram for a typical distributed Kalman filter architecture is shown in Figure 2.2. In this example, there are 3 sensors (GPS, SAR, TAN) and the three sensor-dedicated local filters provide estimation information to the
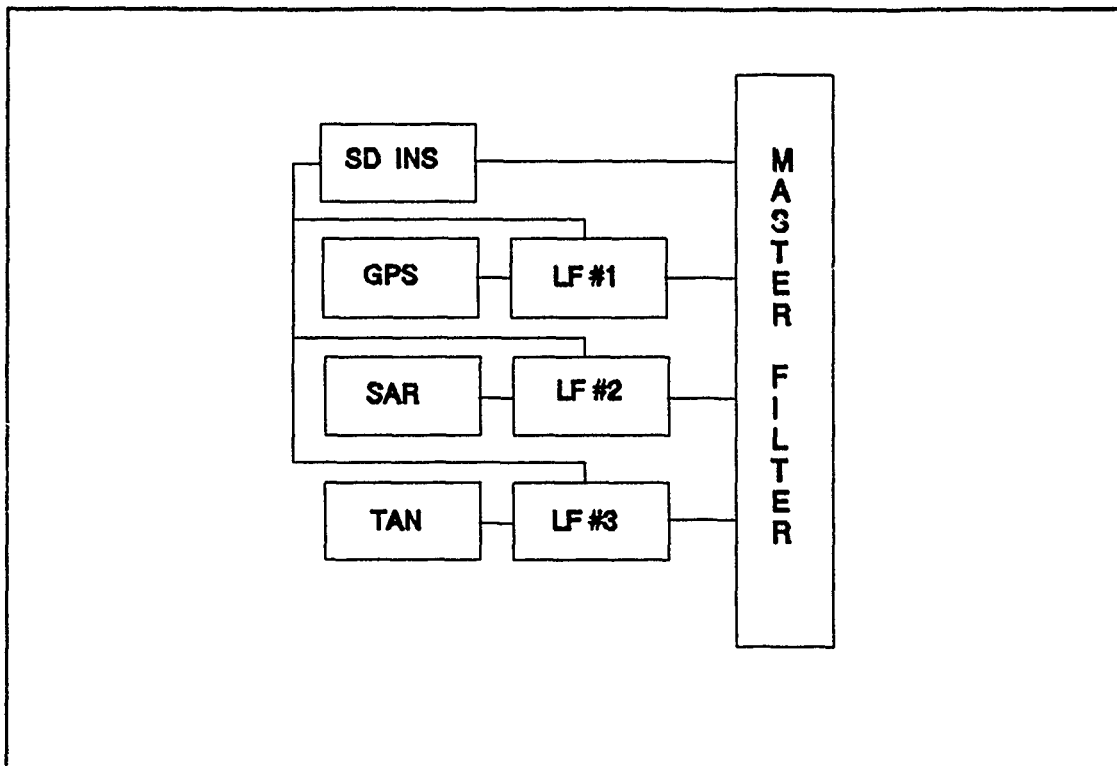
Figure 2.2 Typical Distributed Kalman Filter Architecture

master filter. The master filter then combines this information to formulate a navigation solution. The strapdown INS serves as the reference system for the master filter and all of the local filters.

The sensors are usually described in their local coordinate systems and the transformation to a single reference coordinate system always involves errors (8,9). The second stage requires the master filter to fuse the data from the local filters (20). Assuming the master filter incorporates all of the information provided by all of the sensors, this method will provide a globally optimal state estimate (19).

Comparing the computational requirements of the centralized filter to the distributed filter, it is apparent that, generically, the decentralized versions require more overall computations per cycle than their centralized

counterparts. However, it can generally be stated that the average computation per processor is reduced approximately by a factor equal to the number of local filters in the design. (17) This parallelism will give the distributed filters potential for faster estimation.

The distributed filter is also well-suited to fault detection and isolation because the local filters can enjoy complete autonomy from each other. Any failed sensor, other than the INS, can be removed from the sensor suite without affecting the results of the other local filters. (17) The INS represents a significant exception regarding failed sensor removal, especially when it is receiving feedback corrections from the navigation filter. In this case, the INS outputs are highly correlated with those of the most accurate radio navigation sensor (generally GPS) processed by the respective local filter (12). The problem of highly correlated INS outputs with those of the most accurate sensor can be avoided by having the local filters and the reference system run independently such that a gradual failure in one sensor eventually becomes visible and identifiable when compared to other local filters' navigation solutions (10). The magnitude of the divergence in the local filter solution with the failed sensor would have to exceed the respective range of accuracy for the other sensor-dedicated filters in order for the failure detection to occur under these circumstances. This is a particularly difficult problem to overcome if the failed sensor is the GPS because its accuracy is normally far better than that of the other sensors. Consequently, a gradual failure in the GPS might never be detected.

2.4 The Federated Kalman Filter Architecture

The distributed filter is the predecessor to the federated filter. Dr.

Neal Carlson developed the federated filter architecture as a variation of the distributed filter (1,2,3). The intent was to improve the accuracy of the distributed filter design without compromising its fault detection capabilities. The foundation for success of this design is the principle of conservation of information (1,2,3,7).

Without this information sharing principle, the distributed filter design is subject to poor accuracy and even divergence (1). This principle allows each local filter estimate to be treated independently by the master filter data fusion process. The master filter simply fuses the weighted state estimates and covariances together to form the globally optimal solution according to the ensuing demonstration. (1,2,3) In this case the global state vector can be partitioned into disjoint segments and each segment or subvector yields a compatible reduced-order local model. Should two local models share a common state component, the algorithms are then suboptimal. (17)

Suppose the full centralized filter solution is represented by the covariance matrix P_f and the state vector $\hat{x}_f$. Further, let the ith local filter solution be represented by P_i and $\hat{x}_i$, and the master filter solution by P_m and $\hat{x}_m$. The key to the new federated filtering method is the ability to construct individual local filter and master filter solutions such that they may be combined or recombined at any time. The ability to do so forms the essence of the information sharing principle. (1)

If the recombination of the partial solutions from the local filters was properly executed, the disjoint information would be properly added, plus the common initial conditions and process noise information would be split among the local filters such that the double-counting of this information is avoided. The information would, therefore, sum to the correct total. If the

2-8

local filter solutions are statistically independent, i.e., the optimal solution for each local partition can be determined independently at each measurement time step, then they can be optimally combined by the following additive information algorithm which yields the correct total solution (3,7):

$$P_f^{-1} = P_1^{-1} + P_2^{-1} + \ldots + P_i^{-1} + P_m^{-1} \qquad \textbf{(2.9)}$$

$$P_f^{-1} \hat{x}_f = P_1^{-1} \hat{x}_1 + P_2^{-1} \hat{x}_2 + \ldots + P_i^{-1} \hat{x}_i + P_m^{-1} \hat{x}_m \qquad \textbf{(2.10)}$$

The process noise covariance matrices Q_i for the ith local filter, and Q_m for the master filter, are additionally governed by the information sharing rules such that:

$$Q_f^{-1} = Q_1^{-1} + Q_2^{-1} + \ldots + Q_i^{-1} + Q_m^{-1} \qquad \textbf{(2.11)}$$

Because it is the kinematic information, Q_m, that is being shared, it is of the greatest importance that this information be of the highest quality. The interpretation is that the system process noise information is distributed to the local filters in given portions β_m, as fractions of the information shared. For the cases involving resets, the local filter solutions have to be reset to the combined solution before the prediction is made. (7)

In addition, for this definition, it is assumed that the local filters and the master filter are all full-sized such that the transition matrices Φ_i, Φ_m, and Φ_f are equal. Additionally, the noise distribution matrices G_i, G_m, and G_f are equal. However, in actual practice, the local filters contain only the common INS states plus their own unique sensor biases, therefore the

matrices P_i, Φ_i, Q_i, and G_i contain only the appropriate matrix partitions of
the full-sized matrices. The master filter also contains INS states and
possibly some master-filter-unique INS bias states. This application does not
include any of these unique states in the master filter formulation. (1)

2.4.1 The Federated Filter Reset Modes

The federated filter technique can be implemented in square root form to
maximize computational efficiency, numerical stability, and effective
precision. (1,2) There are currently four primary federated filter
implementations embodying different information sharing methods. The
differences in the three designs relate primarily to the feedback process, but
also include the way the master filter fuses the data. (1,3) Figure 2.3 can
represent any of the feedback modes except the no-reset mode, because the
feedback path to the local filters is severed in the no-reset mode. Again,
the local filters receive sensor measurements and reference system information
from the INS. The master filter provides the INS corrections and the reset
information to the local filters, while combining the information provided by
the filters into a globally optimal navigation solution.

The equations of implementation of the following federated filter modes
correspond to those developed in the documents found in the appendices
regarding the information sharing principle (1,2,3). The reset modes
correspond directly assuming the proper values for the information sharing
fractions for each of the local filters, β_m, while the no-reset mode has no
reset information returned from the master filter.
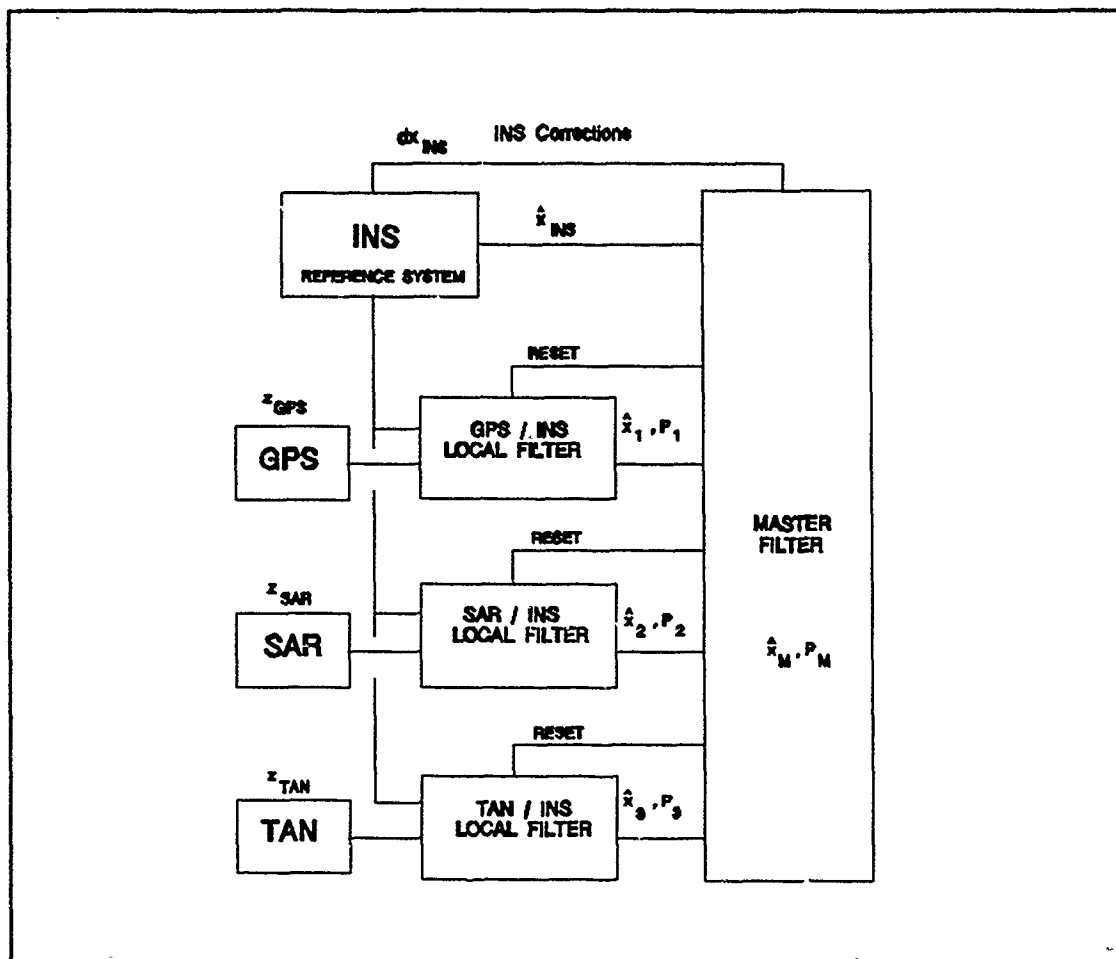
Figure 2.3 Federated Filter Design With Feedback to the Local
Filters

The following descriptions provide the details of the available
federated filter modes:

1. Zero-reset mode. The master filter retains all of the system long-term
memory and the local filters act as data compression filters with short-term
memory only. There is no feedback of the fused solutions to the local
filters. Instead, the local filter is given the command to reset to zero
information after each fusion update, resulting in an infinite covariance, or

a zero inverse covariance. The local filters can be comprised of relatively low order INS and sensor models. Higher order models can be implemented in the master filter. Furthermore, data bus loads are reduced. (1,3)

2. Partial-reset mode: The master filter and the local filters share the system long-term memory. This design involves feedback of only a portion of the full-fused solution to the local filter. The master filter would benefit from having higher order system models than the local filters, thereby, allowing for improved fault detection because the sensor data is treated independently. (1,3)

3. Full-reset mode: This mode operates much like the no-reset mode except feedback of the fused solutions to the local filters is accomplished. The long-term memory resides wholly in the local filters. (1,3)

4. No-reset mode, Figure 2.4: The master filter retains none of the fused information, while the local filters collectively retain all of the local information. This method is similar to the second, except that the master filter solution may be propagated but does not participate in the next fusion update. This no-reset design is highly fault tolerant and, therefore, provides the best overall performance for FDI because the local filters operate independently of each other. (1,3)

For each of the above reset modes, the no-reset, zero-reset, partial-reset, and full-fusion-reset modes, the time propagation and measurement update steps are essentially the same. During the propagation cycles, each of the local filters multiplies its common process noise variances by the
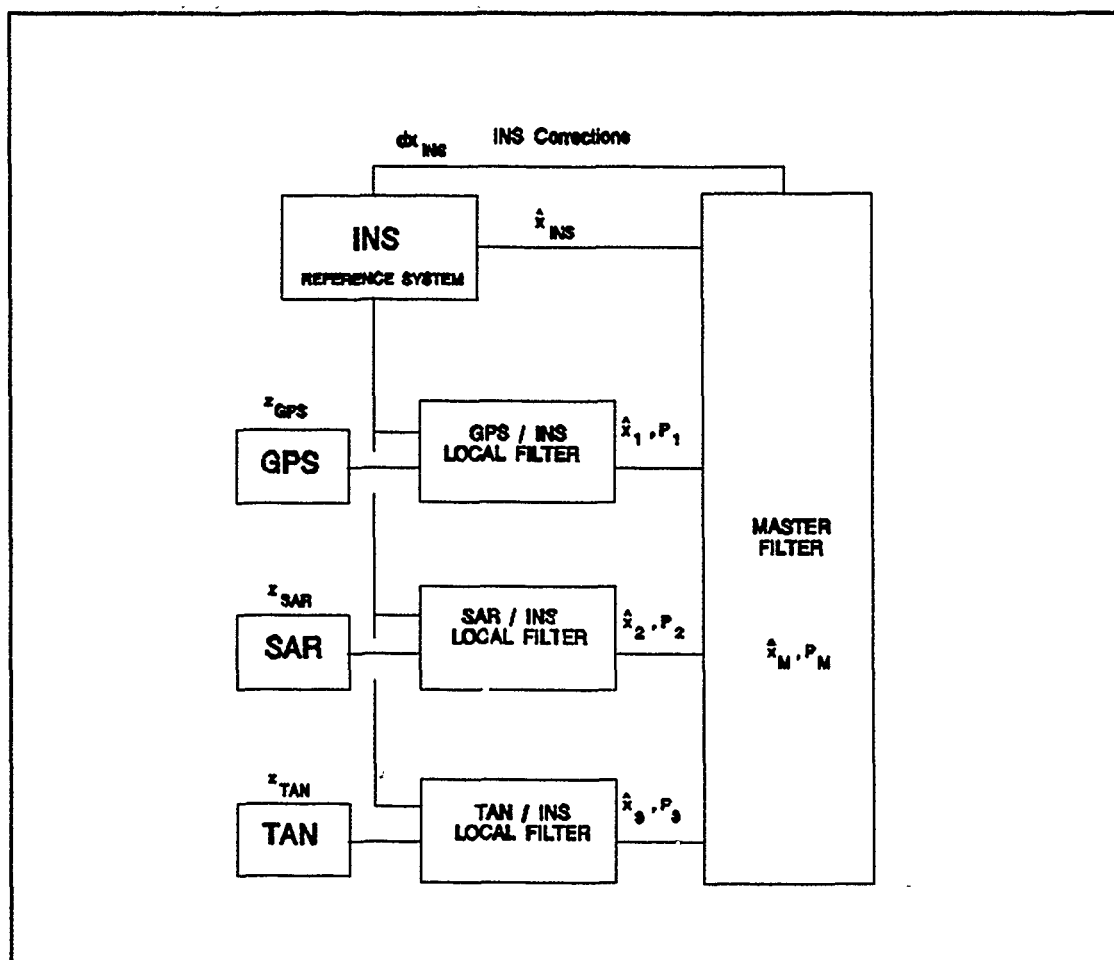
Figure 2.4 Federated Filter Design, "No-Reset" Mode

information sharing fractions in order to split the whole process noise

information up between them. During measurement update cycles, the local

filters perform normal processing of the data from their independent sensors.

2.4.2 The Information Sharing Principle

In the federated filter, the entire system is similar to the general

distributed filter architecture in that it is decentralized into a number of

subsystems. Then, an information sharing process is carried out in which the

dynamic information is distributed to the individual local filters. (7)

2-13

The basic idea of information sharing is to divide the global system information among the several component filters and then perform local processing before recombining the updated local information into a new global sum within the master filter. The precise definition of the principle is the "total system information can remain constant or decrease, but never increase, due to sharing". (3)

The primary reasons for considering the information sharing procedure are: 1) to increase the total system throughput by parallel operation of the local filters, 2) further increase system throughput by use of local filters for local data compression, and 3) improve overall system reliability by maintaining multiple component solutions usable as back-ups. (3)

In a standard Kalman filter, the optimal solution is propagated from one time to the next by adding the next propagation term to the sum, and recursively updating the solution. Measurements are incorporated into the optimal solution by adding the next measurement to the sum, and again recursively updating the solution. This sequential updating process is the basis of the discrete Kalman filter. (3)

Imagine an optimal solution $\hat{x}_i$, P_i obtained by incorporating only the measurements from an arbitrary i^{th} sensor via the sequential process described above. This solution would be optimal relative to that subset of information including the initial conditions and process noises. Because this solution ignores measurements from any other existing sensors, this solution would be considered suboptimal relative to the entire set of information. (3)

If an attempt was made to combine all of the partial solutions from any number of dedicated local filters, the disjoint information could be added from the sensors. However, this would incorrectly include the common initial

conditions and process noise information from each filter redundantly. To avoid such multiplicities of the common process information, it is necessary to divide the information among che partial solutions so they may sum to the correct overall total. (3) Again, we start with a full solution P_f, $\hat{x}_f$, and then divide that solution so that each of the local filters and the master filter all receive fractions, β_i and β_m, of the total information:

The conservation of information principle requires the fractional values sum to unity. (1) The average inverse of the multipliers is equal to 1/n, implying equal division among the local filters.

$$\beta_m + \sum_{i=1}^{n} \beta_i = 1 \qquad\qquad (2.12)$$

This equation ensures that there is no information lost during the information-sharing process. Thus, we can consider this information sharing technique valid because it conserves information and results in no increase of information relative to the original total. We can refer to this technique as optimal if it results in no loss of information, and efficient if the usable information loss is relatively small. (3,7)

The master filter computes the local filter reset solutions required when the federated filter operates in the fusion-reset modes. Therefore, the local filter reset solutions are generated in accordance with the information sharing principle, by means of the local filter information sharing fractions:

$$P_M^{-1} = \beta_1 P_M^{-1} + \ldots + \beta_n P_M^{-1} \qquad\qquad (2.13)$$

$$P_M^{-1} x_m = \beta_1 P_M^{-1} x_m + \ldots + \beta_n P_M^{-1} x_m \qquad\qquad (2.14)$$

Our basic goal is to split the global estimation process into a set of smaller local estimation processes. This splitting must be performed in such a way that simple recombination of the local solutions yields a theoretically correct and an optimal, or at least, a conservatively suboptimal result. By simple recombination we mean a weighted least squares average involving the local estimates and covariances but not involving any cross-covariances. By optimal we mean the globally optimal solution. By conservatively optimal we mean a solution which ignores some of the information available, while making optimum use of the information utilized by the filter. (3)

2.5 DKFSIM Version 1.1

The distributed Kalman filter simulation software tool, DKFSIM Version 1.1, was developed by Integrity Systems, Incorporated (10). It is designed to support performance evaluation of Distributed Kalman Filter (DKF) techniques applied to integrated, multi-sensor navigation systems (10,19).

2.5.1 General Description.
This software consists of a general purpose simulation environment for distributed Kalman filtering. Navigation sensor models are embedded so that problem-specific tailoring may be accomplished. DKFSIM is divided into a real-world simulation segment and an onboard computer simulation segment. (10)

The real-world segment generates the sensor output data that is fed to the onboard computer for use by the navigation filters. It is comprised of models describing the vehicle environment, trajectory, and navigation sensor suite. (10)

The onboard computer segment includes the DKF algorithms that process the sensor outputs to obtain estimates of the vehicle position, velocity, and attitude. It also includes the task scheduling and input/output control routines. (10)

The input data control file, indata.dat, allows for the control of the entire set of simulation control parameters, variables, and switches. An example of an indata.dat control file is provided in Appendix J. Selection of the different modes of operation are controlled by this input data file, along with selection of subsystem control parameters, such as measurement time step, and sensor specific model options.

The navigation sensors embedded within DKFSIM Version 1.1 are:

A) The Synthetic Aperture Radar Precision Velocity Update (SAR-PVU)

B) The Synthetic Aperture Radar Electro-Optical (SAR-EO)

C) The Global Positioning System (GPS)

D) The Terrain Aided Navigation (TAN)

E) The Central Air Data Computer (CADC) Barometric Pressure Altimeter (BARO-Alt).

F) The Inertial Navigation System (INS).

Each sensor has a representative truth model and filter model. The truth model generates the time-varying behavior and data outputs of the actual sensor, including errors. The filter model describes the sensor's initial state, time propagation characteristics, and measurement processes. (10,19)

Each of the external sensors may be assigned to a local Kalman filter if some requirement specifies its use for a particular scenario. Each local filter employs data from the common INS reference system. These local filters can either be operated as integrated centralized filters or as part of a DKF

architecture. The DKF structure usually has several local filters feeding a master filter so an optimal navigation solution may be obtained. Master filter parameters control the specific operations of the master filter data fusion process. This flexibility allows for an efficient means of evaluating the DKF architecture and comparing it simultaneously with an equivalent centralized architecture. (10)

2.5.2 Functional Description. Ideally, the real-world and onboard computer simulation segments run simultaneously in parallel. This is not easily done with one computer system. To obviate this problem, the two segments are interleaved to emulate real-time parallel operation between the two simulation segments. In order to establish a frame of reference for the simulation, a flight trajectory input data file provides DKFSIM Version 1.1 with a flight profile that provides aircraft translational and rotational dynamics describing the truth state. (10)

The second integral simulation segment performs the onboard navigation computer tasks. These tasks include input and output data transfers, the distributed local filter controls, and the local and master filter tasks, such as the reset, propagation, measurement and update cycles.

Input data provides for controlling parameters that specify the configuration of the simulation. There are four categories of potential variation:

A) Simulation control parameters: controls how the simulation is performed, duration, number of Monte Carlo runs, etc.

B) Output control parameters: defines the format and specifies the content of the output data files.

C) Filter configuration parameters: allows the designation of the
sensors to a specified local or master filter.

D) Sensor truth and filter model parameters: allows tuning of the
dynamic variables, and designation of the sensor measurement sample period.

Each of these sets of parameters allows for inserting desired parameter
values, switching selected modes on or off, or as applicable, changing the
operating characteristics of the system.

2.6 Chi-Squared Algorithm

The fault detection scheme used for this project is the chi-squared
algorithm, and is based on the centralized and sensor-dedicated local filter
residuals. These residuals $\gamma(t_j)$ are zero-mean and white with known residual
covariance $\Lambda(t_j)$. The chi-squared random variable $\chi(t_k)$ is given by

$$\chi(t_k) = \sum_{j=k-N+1}^{k} \gamma^T(t_j)\Lambda^{-1}(t_j)\gamma(t_j) \qquad (2.15)$$

with N being the size of the sliding window. Notice that the system dynamics
are not included in Equation (2.15) and that only one failure hypothesis is
available. A detection rule based on an established threshold ϵ would be:

$$\chi(t_k) > \epsilon \rightarrow FAILURE$$

$$\chi(t_k) \leq \epsilon \rightarrow NOFAILURE \qquad (2.16)$$

The simulated sensor failures implemented in this study were small
intentionally, yet large enough for detection by the chi-squared algorithm

with a nominal threshold established. Each set of sensor residuals, GPS, SAR, and TAN, was treated separately by the use of this algorithm, and each set of residuals was compared such that the CKF and DKF designs could reveal their respective fault detection capabilities. Chapter IV elaborates on the results obtained.

2.7 Summary

This chapter has provided a discussion of the centralized, distributed, and federated filter techniques utilized in this thesis. A brief description of the reset modes of the federated filter architecture was given, along with some discussion of the information sharing principle. A detailed description of the truth and filter models for both the centralized and distributed filters used in this thesis are presented in Chapter III.

III. Foundations For Filter Comparisons

3.1 Overview

The objective of this chapter is to provide the truth and filter model descriptions and the fault detection models used in the performance comparison of the centralized and distributed filter architectures. The truth models are composed of original error states and those states denoted as wide-band noises which are to be incorporated via state augmentation of the original system description. (11) The original error states and the wide-band noise representations are listed separately for clarity except in the filter model descriptions. The filter models are composed of all the states available for output to the output data files. A complete and thorough treatment of the true state representation has been implemented. Further consideration is given to the master filter residuals later on in Chapter IV.

3.2 The Truth Model State Description

The following tables apply for both the Centralized Kalman Filter (CKF) and the Distributed Kalman Filter (DKF) implementations. While the CKF design used all of the truth states for the INS, GPS, SAR, TAN, and BARO systems for its real-world operations, the sensor-dedicated filters in the DKF design used only the INS truth states plus the appropriate truth states for each of their respective sensors. Table 3.1 delineates the original truth states for the

INS TRUTH MODEL STATES	REPRESENTATION	COORD. FRAME
3 Position drifts	Linearized propagation driven by velocity drifts	ECEF
3 Velocity drifts	Linearized propagation driven by accel'n errors	ECEF
3 Attitude drifts	Linearized propagation driven by angular rate errors	ECEF
3 Gravity perturbations	First-order markovs (independent)	
3 Accelerometer biases	First-order markovs (independent)	Body frame
3 Accelerometer scale factor errors	Random constants (independent)	Body frame
6 Accelerometer misalignments	Random constants (independent)	Body frame
3 Gyro bias drift rates	First-order markovs (independent)	Body frame
3 Gyro scale factor errors	Random constants (independent)	Body frame
6 Gyro input-axis misalignments	Random constants (independent)	Body frame
6 Gyro accel-sensitive drift coefficients	Random constants (independent)	Body frame

42 TOTAL ORIGINAL STATES

Table 3.1   Strapdown INS Original Truth Model States

medium accuracy strapdown INS. The INS truth model states are listed along with their mathematical representation and their respective coordinate frames. There are a total of 42 original truth model states, with additional state augmentations yet to come.

The GPS truth model assumes four-satellite operation. Several states are pertinent only to their respective satellite. Table 3.2 delineates the 29 original GPS truth model states, while additional state augmentations are provided later. All of the original GPS truth model states are listed with their respective mathematical representations and the applications of each

GPS TRUTH MODEL STATES	REPRESENTATION	APPLICATION
Four Satellite Operation		
1 User clock phase drift	Integral of user clock frequency drift	All channels
4 Satellite range errors due to satellite position & clock phase drifts	Second-order markov errors (independent)	Each channel
4 Ionospheric range errors after L_1/L_2 correction	Second-order markov errors (independent)	Each channel
4 Tropospheric range errors after compensation	Second-order markov with $1/\sin(e_k)$ scale factor	Each channel
1 User clock frequency bias drift	First-order markov	All channels
3 User clock frequency accel-sensitivity drift coefficients	Random constants (independent)	All channels
4 Satellite range-rate errors due to satellite velocity and clock frequency drifts	Derivatives of satellite range errors (independent)	Each channel
4 Ionospheric range-rate errors after L_1/L_2 correction	Derivatives of ionospheric range errors (independent)	Each channel
4 Tropospheric range-rate error after compensation	Derivatives of tropospheric range error with $1/\sin(e_k)$ scale factor	Each channel

29 TOTAL ORIGINAL STATES

Table 3.2   GPS Original Truth Model States

with regard to GPS channel utilizations. In this table, $1/\sin e_k$ represents

the deterministic path-length factor for tropospheric errors.

Simulated outages of normal satellite operations are not modeled such

that the number of states in the truth model is reduced accordingly. Three-

satellite, or less, simulated GPS operation is not an available alternative

for a GPS scenario. The study of GPS failure modes would include these

scenarios plus others, such as poor Geometrical Dilution of Precision (GDOP).

This is not viewed as a drawback in the implementation of the DKFSIM software. At this time, the software is not intended to serve as a tool for any number of unspecified sensor failures. It is intended to concentrate in the area of fault detection, but requires significant source code changes prior to accomplishing this task beyond the simplest of failure mode simulations.

The total SAR system utilizes two subsystems for normal operations. The SARPVU and SAREO original truth model states are delineated in Table 3.3, while further state augmentation is provided later. Each of the 20 original

SAR TRUTH MODEL STATES	REPRESENTATION	APPLICATION
SARPVU		
3 Velocity measurement bias errors	First-order markovs (independent)	All components
3 Velocity scale factor errors	First-order markovs (independent)	All components
6 Mounting misalignment errors	Random constants (independent)	All components
SAREO		
3 Landmark position bias components	First-order markovs (independent)	Each landmark
1 Range bias error	First-order markov	All range measurements
1 Range scale factor error	Random constant	All range measurements
1 Range-rate bias error	First-order markov	All rng-rate measurements
1 Elevation bias error	First-order markov	All elev. measurements
1 Azimuth bias error	First-order markov	All azimuth measurements

20 TOTAL ORIGINAL STATES

Table 3.3 SARPVU and SAREO Original Truth Model States

states are listed along with their mathematical representation, and their respective applications. The SARPVU errors included in the truth model consist of instrument, measurement bias, and scale factor errors. The SAREO errors included in the truth model consist of instrument, measurement bias, scale factor, and landmark location errors.

While the SARPVU and SAREO subsystems may operate independently in the DKF filter implementation of the SAR dedicated local filter, these two systems complement each other such that simulations using independent operation of the two systems is not recommended. The SAREO capability is useful because it bounds the growth of the INS position errors when GPS is not available, while the SARPVU capability provides more accurate velocity estimates. (11)

Table 3.4 delineates the original truth model states for the TAN system, while further state augmentation is provided later in this section. There are a total of 33 original TAN truth model states. Each of the states are listed with their mathematical representations and their respective parameters.

In practice, each TAN fix reduces the vehicle position uncertainties in two directions normal to the local elevation contours. No information is gained parallel to the contours. Given several fixes with a variety of contour directions, and an INS to associate the fixes relative to one another, the positional uncertainty can be reduced in all three spatial directions.

The BARO-altimeter models are intended to represent the Central Air Data Computer (CADC) performance typical of current aircraft systems. Table 3.5 delineates the 4 original truth model states for the BARO-altimeter, along with its corresponding mathematical representation.

TAN TRUTH MODEL STATES	REPRESENTATION	PARAMETERS
1 Radar altimeter bias error	First-order markov	
1 Radar altimeter scale factor error	First-order markov	
1 Terrain map elevation variation	Second-order markov	
1 Terrain map sector bias (vertical)	First-order markov	
2 Terrain map sector bias (horizontal)	First-order markov	
1 True slope along-track	Second-order markov, spatially correlated; variable roughness	
1 True slope across-track	Second-order markov, spatially correlated; variable roughness	
25 Random point elevation errors	Random constants	5 pt square grid

33 TOTAL ORIGINAL STATES

Table 3.4   TAN Original Truth Model States

BARO TRUTH MODEL STATES	REPRESENTATION
1 Pressure altitude bias error	First-order markov
1 BARO-altitude scale factor error due to nonstandard temperature	First-order markov
1 Static pressure coefficient error	Random constant
1 BARO-altimeter time delay	Random constant

4 TOTAL ORIGINAL STATES

Table 3.5   BARO-Altimeter Original Truth Model States

STATE AUGMENTATION	REPRESENTATION	APPLICATION

INS Truth Model

3 Accelerometer wide-band noises	Wide-band noises on velocity drifts with constant power (independent)	Body frame
3 Gyro wide-band noises	Wide-band noises on attitude drifts with constant power (independent)	Body frame

GPS Truth Model

4 Receiver channel random phase noises	Purely random range errors (independent)	Each channel
4 Receiver channel random frequency noises	Purely random range-rate errors (independent)	Each channel

SAR Truth Model

3 Velocity random measurement noises	Purely random (independent)	All components
1 Range random measurement noise	Purely random	All range msmts
1 Range-rate random measurement noise	Purely random	All range-rate msmts
1 Elevation random measurement noise	Purely random	All elev. msmts
1 Azimuth random measurement noise	Purely random	All azim. msmts

TAN Truth Model

1 Radar altimeter random measurement noise	Purely random	

BARO Truth Model

1 BARO-altimeter random measurement noise	Purely random	

23 TOTAL ADDITIONAL STATES

Table 3.6   System State Augmentation

Thus far we have shown listings totalling 128 original truth states for all of the pertinent systems. Table 3.6 describes the 23 total additional states required for augmenting the original state truth models. Each of the systems utilized in the centralized and distributed filter implementations contributes to this state augmentation listing.

The INS truth model is augmented by adding 3 accelerometer and 3 gyro wide-band noise representations. The GPS truth model is augmented by adding one channel random phase noise and one channel random frequency noise per satellite. The SAR truth model is augmented by adding one random measurement noise for each of the seven SAR measurements provided. The TAN and BARO truth models are also augmented by adding a random measurement noise for each of their respective measurements.

The final form of the CKF truth model results in combining the original and the augmenting states for a total of 151 truth model states. The final form of the DKF truth models depends on the filter designation. Local filter #1 utilizes only the INS, GPS, and BARO system truth model states for a total of 90 truth states. Local filter #2 utilizes only the INS, SAR, and BARO system truth model states for a total of 80 truth states. Local filter #3 utilizes only the INS, TAN, and BARO system truth model states for a total of 87 truth states. The master filter in the no-reset mode utilizes only the INS and BARO system truth model states for a total of 53 truth states. Table 3.7 delineates these state totals for clarity.

FILTER DESIGNATION	STATE DESCRIPTION	STATE TOTAL
Centralized Filter	Includes all systems	
	Original state formulation	128
	Augmenting states	_23_
	Total state formulation	151
Local Filter #1	GPS dedicated with INS	
	Original state formulation	75
	Augmenting states	15
	Total state formulation	90
Local Filter #2	SAR dedicated with INS	
	Original state formulation	66
	Augmenting states	14
	Total state formulation	80
Local Filter #3	TAN dedicated with INS	
	Original state formulation	79
	Augmenting states	_8_
	Total state formulation	87
Master Filter	Reference systems only	
	Original state formulation	46
	Augmenting states	_7_
	Total state formulation	53

Table 3.7 Specific Filter Truth Model State Totals

3.3 The Filter Model State Description

The filter models provided for the CKF and DKF designs are reduced-order implementations of the truth models for each system as appropriate. For the sake of expediency in achieving operational status for software simulations, all of the previously implemented filter order reductions are assumed correct.

The CKF and DKF filter models include states from the INS and all of the sensors except the BARO-altimeter. These BARO states were not utilized in the filter model representation, for simplicity, even though the truth model includes the BARO truth states. When BARO damping for the vertical channel is off in the filter model representation, the INS vertical channel must be

stabilized by corrective feedback from the filter. (10) Dedication of a fourth local filter for the BARO-altimeter alone was not an available alternative for testing. Also, inclusion of the BARO-altimeter sensor in one of the other sensor-dedicated local filters might obscure that sensor's behavioral characteristics.

There are three different INS filter model selections available. The BASIC filter model consists of the first 10 states listed in the INS filter model description. The GBIAS includes the first 10 states plus the additional 3 gyro bias drift rate error states. The AGBIAS was chosen for completeness in the filter model representation. It includes the 13 INS states in the GBIAS filter model, plus the 3 additional accelerometer bias error states, for a total of 16 filter states.

Table 3.8, located on the next page, describes the filter state model representations for each of the systems utilized. The AGBIAS model for the INS includes 16 total filter model states. The GPS model chosen includes 5 states. The SAR systems include 3 states for the SARPVU system and 4 states for the SAREO system. Lastly, the TAN model consists of only one state.

There are two different GPS filter model selections available. The minimum filter model representation requires only user clock phase drift and the user clock frequency bias drift. Again, for completeness the enhanced filter model representation of 5 states was chosen. This consists of the user clock frequency bias drift, and four states which represent the four satellite range errors, plus clock phase drift components for each of the four channels.

FILTER MODEL STATE	REPRESENTATION	PARAMETERS
INS States - AGBIAS Model		
3 Position drifts	Linearized propagation driven by velocity drifts	ECEF
3 Velocity drifts	Linearized propagation driven by accel'n errors	ECEF
3 Attitude drifts	Linearized propagation driven by angular rate errors	ECEF
1 Vertical acceleration error	First-order markov (independent)	
3 Accelerometer biases	First-order markovs (independent)	Body frame
3 Gyro bias drift rates	First-order markovs (independent)	Body frame
16 TOTAL INS FILTER STATES		
GPS States - 5 State Model		
4 Satellite range errors due to satellite position & clock phase drifts	Second-order markovs (independent)	Each channel
1 User clock frequency bias drift	First-order markov	All channels
5 TOTAL GPS FILTER STATES		
SARPVU States		
1 X-axis mounting misalignment error	Random constant (independent)	
1 Y-axis velocity scale factor error	First-order markov (independent)	
1 Z-axis mounting misalignment error	Random constant (independent)	
3 TOTAL SARPVU FILTER STATES		
SAREO States		
1 Range bias error	First-order markov	All msmts
1 Range-rate bias error	First-order markov	All msmts
1 Elevation bias error	First-order markov	All msmts
1 Azimuth bias error	First-order markov	All msmts
4 TOTAL SAREO FILTER STATES		
7 TOTAL SAR FILTER STATES		
TAN States		
1 Radar altimeter bias error	First-order markov	
1 TOTAL TAN FILTER STATE		

Table 3.8   System Filter Model State Representations

SPECIFIC FILTER	SYSTEM ASSIGNMENT	STATES
Centralized filter	INS	16
	GPS	5
	SAR	7
	TAN	1
CKF TOTAL		29
Local Filter #1	INS	16
	GPS	5
LF1 TOTAL		21
Local Filter #2	INS	16
	SAR	7
LF2 TOTAL		23
Local Filter #3	INS	16
	TAN	1
LF3 TOTAL		17
Master Filter	INS	16
MF TOTAL		16

Table 3.9 Specific Filter Listings With System Model State Assignments

Table 3.9 depicts the application of each system filter state model as required for each of the filter implementations, i.e., centralized, local and master filter utilizations. The DKF design applies each system filter state model according to its specified dedicated local filter. The master filter of the DKF design encompasses only those states of the INS filter model, while the CKF design encompasses all of the filter states.

This completes the discussion of the truth and filter models used for the performance comparison. These representations fulfill all of the minimum requirements necessary for this comparison.

SENSOR	RESIDUAL	DIMENSIONS
GPS - 4 Satellite Operation		
2 Satellite #1	Pseudorange	ft
	Pseudorange-rate	ft/sec
2 Satellite #2	Pseudorange	ft
	Pseudorange-rate	ft/sec
2 Satellite #3	Pseudorange	ft
	Pseudorange-rate	ft/sec
2 Satellite #4	Pseudorange	ft
	Pseudorange-rate	ft/sec
8 TOTAL GPS RESIDUALS		
SAR		
3 SARPVU	X-Velocity	ft/sec
	Y-Velocity	ft/sec
	Z-Velocity	ft/sec
4 SAREO	Range	ft
	Range-rate	ft/sec
	Elevation	ft
	Azimuth	ft
7 TOTAL SAR RESIDUALS		
TAN		
1 TAN	Ground clearance	ft
1 TOTAL TAN RESIDUAL		
16 TOTAL SENSOR RESIDUALS		

Table 3.10 Residual Listing by Sensor for Both CKF and DKF Designs

3.4 Fault Detection Considerations

Residual monitoring is the foundation of the fault detection scheme utilized for this comparison. Table 3.10 delineates the residuals used according to the measurements available from the specific sensors. The CKF design resulted in processing all sensor measurements through one Kalman filter. The DKF design resulted in Local Filter #1 processing only the GPS residuals, Local Filter #2 processing only the SAR residuals, Local Filter #3 processing only the TAN residual, and the Master Filter having access to all the residual outputs from the local filters. Because there are 16 like

residuals for each design, we are able to observe directly the fault detection capabilities of the two filters.

The principal effort of residual monitoring for the DKF design was conducted at the local filter level. Local filter residual monitoring was the auspicious choice for fault detection because the required source code changes were minimized, and the local filter level monitoring allowed for efficient extraction of the centralized filter residuals and sensor-dedicated filter residuals simultaneously. Further, this facilitated direct comparison of the most desirable fault tolerance aspects, detection and isolation of sensor failures. Additional information regarding residual monitoring at the master filter level involves detection of INS and other sensor failures, and will be discussed further in Chapters IV and V.

3.5 Summary

The foremost topics of this chapter allow for an understanding of the basis of the comparison between the centralized and distributed architectures. The truth and filter model states define the implementations of the designs. Residual monitoring provides a means for conclusions regarding fault detection. Chapters IV and V will expand upon the information provided here and will describe the steps taken and the results obtained by using these architectures as defined.

IV. Results and Observations

4.1 Overview

The objective of this chapter is to reconstruct the method and process of obtaining the results of this thesis, to provide the information obtained from the simulations and subsequently analyze those results. An opening discussion is presented on the discoveries made while using DKFSIM Version 1.1. The subsequent discussions address the comparison between the centralized and federated filters, the error state and residual outputs, and then the fault detection considerations. Finally, some observations and commentary in the documentation provided by Integrity Systems, Incorporated is augmented to the conclusions of this thesis.

4.2 DKFSIM Version 1.1

DKFSIM Version 1.1 was utilized as the software simulation tool for this thesis effort. This particular version of the software was quite capable of providing Monte Carlo simulations of the federated and centralized filter performances for study. However, there are several aspects of this software which required attention, and, in fact, complicated the research process. Because the final version of DKFSIM was not available at the time of performing the research, the software orientation involved a great deal of investigation into the source code itself. Further, the required software

changes made to facilitate the achievement of the thesis goals were made under unfavo able conditions due to the inherent complexity of the software.

4.2.1 Software Orientation

Initially, the largest obstacles were the incompatibilities with the tools available at AFIT. The conversion from one fortran format to another was necessary due to a change of the computer host. Additionally, the plotting tools available were incompatible with the format of the output data files. Once a basic implementation of the software was compiled and linked, the trial and error verification phase began. Immediately, the data files had to be restricted to just three column format in order to load them into the plotting software. This restricted the data output files to including only one state, its corresponding one-sigma values, and the time line associated with the pertinent time steps of the simulation. This was no fault of the version of DKFSIM available, but had an impact on the efficiency of the performance of the full-state simulations.

To further ensure the duplication of simulation characteristics between the federated and centralized filters, it was necessary to use the exact same flight profile for each error state output for both filters and, likewise, all of the residual outputs for both filters. A typical flight profile was used for the exercise and includes ingress to a low level terrain following and terrain avoidance segment, a SAR target acquisition pop-up maneuver, a target bombing run, and egress from the low level portion (19).

As a result of breaking up the simulation into state-by-state testing, it was absolutely essential to verify that the random number sequences from one simulation to the next were identical. The exact same random number

sequences were used for all of the simulations, except for the ten-run Monte

Carlo analyses for the error state outputs, and to verify the contention of

repeatability. A multiple-run analysis was accomplished to provide data

representing the ten-run performance of an arbitrary filter configuration.

This exact same scenario was run again, and the data points were compared

directly. The comparison revealed identical results.

Another ten-run simulation was performed using a different random number

seed to verify the differences of the random number seeds. The comparison of

these data files to the previous ones revealed that the simulations were not

identical. Therefore, it was acceptable to obtain the entire set of error

state output data files for one simulated scenario one state at a time.

Further, this permitted a reasonable federated and centralized filter compari-

son with respect to the random number sequences.

A comprehensive study of the input control data file, indata.dat, a

sample of which is found in Appendix J, was the next logical step in the

research process. Variables such as IRUN1, IRUNL, TSTART, and TFINAL were

identified as critical system control switches, with IRUN1 designating the

first run in the simulation, IRUNL designating the last run, TSTART equalling

the simulation start time, and TFINAL equalling the simulation completion

time. The desired simulation data required a ten-run Monte Carlo analysis,

accomplished with a flight duration from zero to 7200 seconds.

The input control data file is labeled indata.dat. This file provides

the opportunity to vary certain parameters and variables, along with the

simulation control switches. Several simulations were accomplished just to

verify the application of the control switches in indata.dat. It was

discovered that the error-state one-sigma values were computed for the first

Monte Carlo run of the simulation only. It was attempted to set the UPSTAT switch to true in order to verify the impact on the one-sigma values. The proposed function of UPSTAT is when the Monte Carlo statistics of one run are to be added to those from a previous series of runs (10). There was no apparent impact to any of the output data files, in any manner, by selecting the variable as true. It was set to false for the remainder of the simulations.

The variables OUTASC and OUTSIG were used to designate the output data file format to ASCII form, and to include the one-sigma values in the output data file. OUTERR set to true indicates that the error state outputs were provided in these data files. The last step in output file designation, aside from selecting the desired state per file, was to select the time step increments for the filter computations and sensor measurements. All other aspects of the indata.dat file correspond to the filter definition and the system parameter definitions.

The first attempts at verifying the filter mode options yielded only two available modes; the full-fusion-reset and no-reset modes (see Chapter II). The full-reset mode was used initially in getting the software operating smoothly. A visual comparison revealed that the centralized and federated filter outputs were virtually indistinguishable. All the information in the master filter is shared with the sensor-dedicated local filters to the point where the sensor with the greatest accuracy dominated all local filter outputs essentially like a centralized filter. This mode was not used any further than taking this cursory look at the outputs for operational verification.

The real intent of using the federated filter design for this thesis is to show the benefit from its fault detection and identification capabilities

of the local and master filters. Consequently, the only mode used in subsequent federated filter implementations was the no-reset mode, because this mode has been shown to be the most fault tolerant. No information is fed back to the sensor-dedicated local filters so that there is no corruption of the local filter navigation solution when a failure occurs in any of the other sensors. Although some compromise is expected in the estimation accuracy when using the no-reset mode instead of the other modes available, the no-reset mode is the most efficient federated filter mode with respect to fault detection and isolation.

DKFSIM Version 1.1 allows the single-sensor-dedicated local filter #1 operations to convert easily to a centralized filter by simply designating all of the sensors to provide measurements. Thus, local filter #1 was the designated filter location for the centralized filter implementation and had all of the sensors selected for those simulations. For the federated filter implementations, local filter #1 was dedicated to the GPS sensor, local filter #2 was dedicated to the SAR sensor array, and local filter #3 was dedicated to the TAN sensor.

Residual tolerance values are selectable for filter residual rejection thresholds from within the indata.dat file. These tolerance values are set so the filter will compare the magnitude of the newly available residual against that of the desired acceptable magnitude. If the residual falls outside of this range of tolerance, the filter will not use this most recent measurement for an update cycle. The filter will propagate the estimate again without an update for this particular cycle. For effective filter fault tolerance, these tolerance values are set low, about 3 to 5, representing multiples of the one-sigma values, so that the filter does not incorporate the bad information.

The time segment designations in indata.dat allow for simulating long-term measurement rejection by the filter by increasing the time step between measurements for each sensor. The number of segments which may be designated is up to the user. However, the simulations revealed that there were maximum and minimum limitations on the size of the time step for the SAR and TAN sensors. The simulation would fail and halt when the step sizes were too large or too small for the SAR or TAN. The GPS sensor appeared to be unlimited in its step size. Independent testing was accomplished to verify this contention.

The input control file, indata.dat, also allows for selection of the INU model; either the BASIC, GBIAS, ABIAS, or AGBIAS (see Chapter III). The BASIC model includes only the first ten INS error states. The GBIAS model has the basic ten INS error states plus 3 gyro bias states. Likewise, the ABIAS has the basic ten INS error states plus 3 accelerometer bias states. Finally, the AGBIAS includes the basic ten INS error states plus the 3 gyro bias and 3 accelerometer bias states, totalling 16 error states. This is the desired INU model for this thesis, because it offers more information to compare between the two filter error state outputs.

The alignment mode is also selectable from the indata.dat file. The ALNMOD variable can either be set to SIMALN, representing a normal ground alignment of the INU, or to AIRALN, representing an in-flight alignment. The AIRALN mode was not used in any of the simulations performed.

The verification of the states pertaining to the state output select switches in indata.dat was the next step in becoming familiar with the software. While there exists documentation which lists the states of the truth and filter models, there is not an actual listing of the states as they

are assigned for output by the control file. It was necessary to verify the expected state designation to each of these switches by investigating the source code of DKFSIM Version 1.1 and by observing the behavior of the output variables. It was concluded that the state assignments in indata.dat were as expected according to the state listings found in the documentation (11).

Finally, familiarity of the software turned out to be the most critical aspect of this thesis. It was certainly the largest obstacle to the beginning of the software simulations. Eventually, the simulations were accomplished and the results obtained represented the desired scenarios. Then, it was possible to continue on with the next phase of the research.

4.2.2 DKFSIM Version 1.1 Source Code Changes

Several source code changes were required because this version of DKFSIM required a clean-up of the existing error state output data files in order to plot the error state data for the local and master filters. Plus, DKFSIM does not provide the filter residual output data necessary for fault detection. Once the residual data was obtained, the simulated failures had to be accomplished by affecting the measurements directly.

The output data files for the error states consisted of four ASCII-character columns, as shown in Table 4.1. The first column described the function accomplished for a particular increment of time. Several functions could be shown for each time step depending on the filter control definitions. The column was titled TASKID, which identified the task being accomplished. The possibilities for this column included initialization (LFINIT, MFINIT), measurement receipt and update (LFMEAS), filter reset mode operations (LFRSET, MFRSET), filter propagation (LFPROP,MFPROP), and end run (ENDRUN). Almost all

of the same TASKID possibilities exist for the master filter output files as the local filter output files, simply having different designations. The write statements were altered to replace the TASKID characters with blank spaces such that the top row of information could be deleted, then the three column matrix could be given a name and executed into the plotting software.

The remaining columns represent the time step designation, the error states, and their associated one-sigma values. The error state columns are shown first and then their one-sigma values are the columns to the right. The numbers at the top of the columns indicate the state assignments for that column. The table lists data which shows a run initialization at the 1600.00 second time step, and the end of the run occurring at the 2000.00 second time step. This was done for no specific purpose other than to shorten the run for illustrative purposes.

TASKID	TIME	1	1
LFINIT	1600.00	-19.8715	173.205
LFRSET	1600.00	-19.8715	173.205
LFPROP	1605.00	-19.8340	173.418
LFMEAS	1605.00	5.06294	18.5948
LFRSET	1605.00	4.50609	31.8418
LFPROP	1610.00	4.64870	31.8517
LFMEAS	1610.00	5.92014	20.8378
LFRSET	1610.00	5.29310	26.4917
LFPROP	1615.00	5.23778	26.5019
LFMEAS	1615.00	-10.5534	21.4807
LFRSET	1615.00	-1.56338	24.4573
"	"	"	"
ENDRUN	2000.00	-6.78328	29.6733

Table 4.1 Example of an Error State Output Data File.

All of the following software changes pertain to the task of obtaining the basic residual output data files, and then modifying the residual computations to simulate hard and soft failures in the sensor measurements. The initial residual output files were the most difficult changes to execute because of the location of the "pick-off" is essential to know.

Local filter residual monitoring was determined as the most auspicious choice for the fault detection scheme. The purpose of this choice was twofold. Monitoring the residuals from the dedicated local filters allowed for greater certainty in the problem of fault identification. Knowing which sensor is exhibiting the failure provides a distinct advantage in fault detection because no further failed-system identification is required. Thus, the source code associated with the residual computations of the local filters and its output routines was the starting point for changes to the software.

Secondarily, local filter #1 was not only dedicated to the GPS sensor for the federated filter architecture, but it was also the filter used for the centralized filter implementations in the simulations. This required the residual output from that local filter.

When DKFSIM performs its checks on residual values versus the set tolerances, and with the tolerances set as low as five times the one-sigma bounds, the rate of rejection of the residuals was expected to be very high under even the smallest induced failed conditions. Therefore, the normal-valued residuals were simply picked off after the residual calculation and the measurement update cycle, while the rejected residuals were picked off prior to calling the source code routine which compares the tolerances with the magnitudes and then rejects if bad. This allows continuity in the plotting of the residual data points. Without maintaining a record of the rejected

residuals at the time of their occurrence relative to normal-valued residuals, the fault detection algorithm has no failure information to work with, thus, no detection. However, it is quite undesirable for the filter to be forced to use this bad measurement information. Having set the tolerance levels so low allows the filter to continue to reject the bad measurements while, at the same time, the entire residual sequences are saved for evaluation.

The input control data file, indata.dat, has the capability to simulate a hard sensor failure by increasing the time step increment to a value which represents the length of time during which measurements from that sensor are unavailable. This is sufficient for evaluating the error state performance with hard failure conditions for a particular sensor, but fault detection is not available under these circumstances. This simply represents residual rejection without ever calculating a residual under the failed scenario.

There are a total of three residual output data files. This was done for simplicity. The GPS, SAR, and TAN residuals were maintained in their respective groups so they might be treated independently, especially in light of the fact that they had different time step increments. An appropriate condition statement was used to test for the proper sensor prior to calling the associated write statement. The GPS provides eight measurements every five seconds. The four satellite operation requires a pseudorange and a pseudorange-rate residual for each satellite. The SARPVU and SAREO systems provide seven measurements every 100 seconds. The TAN sensor provides one measurement every ten seconds. These time step values were chosen based on the observed filter performance. Several time step selections were made to maximize the number of time steps per simulation without receiving simulation interrupts or fatal errors.

Initially, some difficulty arose during the centralized filter testing where the residual output data files were providing twice as many residual outputs as expected. It was determined that each filter calculates its own residuals when a sensor is assigned to that local filter for measurement inputs. The other local filter sensor assignments did not have to be deselected, but, as one might expect, the other local filter sensors produced similar residual calculations. As a result of this discovery, there was then the requirement not only to test for the right sensor, but also to test for the right local filter depending on the desired filter operation, federated or centralized, before the residual was chosen for output to the data file.

The residuals obtained under normal operating conditions are found in Appendix C for the centralized filter and Appendix D for the federated filter. These plots are referred to as baseline plots and were necessary to establish the proper starting point for the order of magnitude change in the measurement values. As it turned out, the nominal residual values were the same for the centralized and federated filters. This was useful because the same values could be used equating the failures in the federated and centralized filters.

The next step for obtaining fault detection evaluations, was to implement the hard failures by directly affecting the measurements prior to the calculation of the residuals. Just prior to the write statement for data output, the hard failure result was achieved by setting the measurement, ZM, equal to zero in the following residual calculation, where DZ is the residual,

$$DZ = ZM - ZF \tag{1}$$

ZM is the measurement, and ZF is the filter computed expectation of the measurement's value. Thus, with ZM equal to zero, then the measurement is

representing a hard failure for the sensor. It was determined that the value of ZM should equal only the measurement noise values with no measurement information. The shear magnitude of the hard failure eliminated the reason for this proposal because of its dominance in the residual calculations. In other words, the level of measurement noise would be utterly undetectable under any hard failure circumstances. So, it was not pursued any further.

Soft failures were not induced in all the measurements for all the sensors. It was decided that a constant bias on the second GPS residual, the pseudorange-rate residual for satellite #1, and on the TAN residual would represent a soft constant bias failure quite well for this study. This proved to be is a feasibility study, and does not intend to look at all possible failure scenarios. The constant biases were turned on abruptly and turned off abruptly for expediency. A further condition statement was used to test for the proper residual before the biases were initiated. The effective time of GPS biases is from 1000 seconds to 3500 seconds. This was deemed a sufficient duration for failure and detection. The effective time of the TAN bias is from 4500 seconds to 6500 seconds so that there was sufficient time for filter recovery before inducing the TAN failure. The constant bias added to the TAN residual was unchanged for all failure subsequent simulations in this study.

Another important soft failure was a ramp bias added to the same GPS residual. The ramp bias scenario can be representative of gradual failures in the sensors. These gradual failures can be the most difficult to detect because the filter tends to track the ramping affect of the bias without any notice of a failure. The peak magnitude of the additive ramp bias was chosen to be twice that of the constant bias so as to maintain relativeness. The ramp bias was also started at 1000 seconds and gradually grew to its peak

4-12

value at 3500 seconds. It was then abruptly reset to a zero bias value so that the large transient effects would be detectable. This coincides with a possible scenario of having a satellite with a soft failure contributing to the navigation solution, and then when that satellite passes out of range, an unfailed satellite would immediately take its place providing accurate information. Thus, an abrupt change in residual calculations would occur. Additional discussion regarding fault detection is provided later on in this chapter.

Finally, it was discovered that the current residual values are impacted by the previous residuals when the filter performs the associated update cycle for that previous residual. This is because the software implementation uses scalar measurement update cycles so that the filter handles one measurement at a time. It would be most efficient to handle the residuals in vector form so that the filter computed expectation of the new measurement is not affected by bad information provided by a residual during that measurement and update cycle. There was nothing to be done to the source code regarding this aspect of the residual characteristics. Implementing a change to offset this procedure would require enormous changes in the source code. This is not the purpose of this thesis effort. However, it will remain a consideration upon further discussion.

4.3 *The Centralized Versus Federated Filter Comparisons*

The error state behavior for the centralized and federated filters can be visually inspected by viewing the plots located in Appendices A and B. Appendix A contains the 29 error state plots for the centralized filter design and Appendix B contains the 29 error state plots for the federated filter

design. Please, make note of the scales used for plotting when comparing these state plots. The plotting software was somewhat inconsistent when assigning the vertical magnitude scales when generating these plots. Some contribution to this complication was made by an initial large transient in the error estimations. This would account for a relatively small plot of the data versus what might seem to be a more appropriate scale. Further, the error states are shown with their respective one-sigma bound instead of two- or three-sigma bounds because the plotting resolution is much better without unnecessarily increasing the vertical scaling.

A general comparison can be made with respect to filter performance by comparing the like error state plots for each filter implementation. For both the centralized and federated filters, the first 16 states correspond to the INU filter states and can be compared directly regarding relative magnitudes of error and their associated one-sigma bounds. The last 13 states correspond to the GPS, SAR, and TAN filter error states. The centralized filter provides these state outputs based on single filter operations. The federated filter design has the GPS states output from local filter #1, the SAR states output from local filter #2, and the TAN state output from local filter #3. While a direct comparison can be made of these last 13 sensor-specific states, it is important to remember that the two filter designs have created these error state histories differently.

The X and Y components of position drift, Figures A.1, B.1, A.2, and B.2, for each filter, compare favorably with the exception of the degree of variance divergence in the two filters. The two filters have excellent estimate histories until about 4000 seconds when the error appears to increase resulting in a corresponding increase in the one-sigma bounds. Although the

one-sigma bounds apparently increase more for the federated filter, the vertical scales on these plots seem to over-emphasize this phenomena. It is assumed that this divergence is attributed to the high-dynamics portion of the flight profile beginning at about 4000 seconds into the mission.

The Z component of the position drift, Figures A.3 and B.3, for each filter, appears to exceed the one-sigma bounds. This is not a point of difficulty because the errors do not appear to be large enough to exceed a three-sigma bound value, three times the plot of the one-sigma bound. It should be noted that this state was estimated poorly when expectations for excellent performance was so high for all states. The vertical channels are hardest to estimate.

All three components of the velocity drift, shown in Figures A.4, B.4, A.5, B.5, A.6, and B.6, for each filter, are difficult to view because of the initial large transients in the states. This shrinks the plots down to a relatively small vertical range. It is, however, evident that all three states remain within their one-sigma bounds as desired and the estimates appear accurate.

The three components of the attitude drift, Figures A.7, B.7, A.8, B.8, A.9, and B.9, for both filters, indicate excellent performance. The federated filter performance appears slightly better than that of the centralized filter, but not by much. Perhaps, an analysis which involves more than ten Monte Carlo runs would not provide the same results. The one-sigma bounds seem to follow the behavior of the ensemble average more closely than the previous states. Also, the ensemble average has more variations than what was expected. This is not a drawback, however, because all of the errors are within the one-sigma bounds.

The vertical acceleration error, Figures A.10 and B.10, for both filters, indicates excellent performance. The greatest error is obtained when the aircraft experiences its greatest altitude changes during the flight. The one-sigma bounds for the centralized filter appears to be somewhat larger than that of the federated filter, although the error state performance appears to be quite similar with regard to the magnitude of the errors in the estimates.

The body frame accelerometer biases, Figures A.11, B.11, A.12, B.12, A.13, and B.13, for each filter, clearly indicate where the high-dynamic portions of the flight profile are encountered. Again, the one-sigma bounds for the centralized filter seems to be somewhat larger than that of the federated filter while the estimation accuracies are similar.

The body frame gyro bias drift rates, Figures A.14, B.14, A.15, B.15, A.16, and B.16, for each filter, also clearly indicate where the high-dynamic portions of the flight profile are encountered. These error values are seen to be very small when conditions of high-dynamics are present, and especially when they are not. It is apparent that the errors and one-sigma bounds for the majority of the flight profile are relatively insignificant compared to those of the high-dynamics portions.

The four satellite range errors, Figures A.17, B.17, A.18, B.18, A.19, B.19, A.20, and B.20, for each filter, are estimated quite well. The growth of the one-sigma bound on the federated filter states would seem to indicate a lack of confidence in the measurements, but the performance of the estimates compares favorably with that of the centralized filter. These error states and one-sigma values provided by the federated filter design are extracted directly from the GPS dedicated local filter. The master filter does not have these states available for output. Consequently, it is considered that this

condition is due to the local filter's efforts to provide the master filter with t much of the measurement information as possible. The centralized filter's errors are relatively similar to that of the federated filter's, however, the centralized filter's one-sigma bounds behave much more appropriately. As expected, they maintain relatively constant magnitudes under normal operating conditions.

The GPS user clock frequency bias drift, Figures A.21 and B.21, for each filter, behaves almost identically. The differences between the two plots are virtually indistinguishable.

All SAR plots, Figures A.22 thru A.28 and B.22 thru B.28, with respect to each filter, compare favorably. The jerky appearance of the plots is due to the 100 second time step increment in the measurement availability. The one-sigma bounds behave more erratically for the federated filter than for the centralized.

Finally, the TAN radar altimeter bias error, Figures A.29 and B.29, compare exceptionally well. There appears to be very little difference in the error and one-sigma values for both filters. Regarding the magnitude of the errors throughout the mission, it would seem that the closer the aircraft is to the ground, the better the error estimation performance becomes.

The objective of this section was to show the comparability of the federated filter estimation accuracy versus that of the centralized filter. The results are quite favorable in this respect. Therefore, there is no significant loss of estimation accuracy when selecting the federated filter over the centralized filter. When selecting between the two filter implementations, the contention of accuracy deficiencies of the federated filter should not employ a great deal of influence on that decision.

4.4 *Fault Detection Considerations*

As was stated earlier in this chapter, the baseline residual plots provided a measure as to the amount of the bias that should be added to each affected residual. Because each residual has its own range of magnitudes, the application of any one magnitude of bias failure could not easily be applied to all sensor residuals at once. The objective for this thesis is to show how a possible failure scenario might impact certain specific residuals, and how those failures could be detected and perhaps dealt with appropriately.

The hard failure scenario residual plots are not provided in the Appendix because the residuals responded so dramatically to the simulation of a zero-valued measurement. There was no advantage in pursuing this avenue because the application of the fault detection algorithm would provide no additional information. The failure was so blatantly obvious that even if the residual tolerance levels were set to 50 times the one-sigma bounds, the filter would still reject these residuals. The only possible advantage to this scenario would have been to observe the error state performance under these failed conditions based on filter rejection of the bad residual values. The prime motivation of this research effort was to show the filter behaviors when performing fault detection of soft failures which distinguish the two designs from each other.

4.4 *and TAN Constant Bias Simulated Failures*

The first soft failures implemented were constant bias values added to the GPS pseudorange-rate residual of satellite #1, and to the TAN ground clearance residual, during different phases of the flight profile. An Appendix E

displays the residual plots from the centralized filter, while Appendix F displays those of the local filters from the federated filter design.

For the sake of brevity, only those scenarios which provide an efficient display of the intent of the research are presented. Several simulations were run as trial and error testing to ensure the proper magnitudes for affecting the measurement in the failure scenarios. However, it seems worth mentioning that constant bias values were also added to the pseudorange residual of satellite #1. It was determined, after several iterations, that the appropriate bias value would be about 100 ft for minimum fault detection. The SAR residuals were left unaffected for the entire research effort because the GPS is the most accurate sensor and the TAN was the easiest to affect.

The GPS pseudorange-rate residual of satellite #1 was initially affected from 1000 seconds to 3500 seconds by adding a constant value of 1.00 ft/sec to the incoming measurement. This resulted in a much greater than desired reaction from the filter. The next value attempted was 0.25 ft/sec and this value was found to be nearly undetectable. The original intent was to implement a constant bias value which could be detected, yet would still allow a close comparative analysis to the normal-valued residuals. The value chosen for the magnitude of the bias was 0.5 ft/sec over the entire duration of the GPS failure.

The TAN ground clearance residual was initially affected from 4500 to 6500 seconds by adding a constant bias value of 1000 ft to the incoming measurement. Again, the filter reaction was far beyond the desired response, so smaller values were chosen until the proper value was found, 150 ft. The TAN failure was left unchanged for all failed-condition simulations for

convenience which allows two different sensor failure evaluations to take place simultaneously.

The residual behavior of the pseudorange of satellite #1 for both filters, Figures E.1 and F.1, reflected the greatest impact. Both sets of residuals showed a bias of about 20 feet. This occurs because, when viewing the pseudorange-rate residual of satellite #1, the filter has an initial transient due to the constant bias addition at 1000 seconds, and then settles back into normal operations accepting the biased residual as accurate. Consequently, the pseudorange calculations are impacted by a 0.5 ft/sec error and the filter is continuously mistaken when calculating the pseudorange. Another transient is experienced when the constant bias is removed, and the filter again settles into normal operations accepting these values as accurate. This shows another benefit of using the smallest detectable bias for each of the residuals. The filter does not reject these biased residuals, but instead uses them to update the navigation solution. All GPS residuals for both filters behaved nearly identically. Lastly, even though the TAN residual failed from 4500 to 6500 seconds, the dominance of the GPS accuracy did not allow this failure to impact the GPS residuals even in the centralized filter design.

The subsequent GPS pseudorange-rate residual calculations for both filter designs are affected by the induced failure in the first satellite pseudorange-rate measurement. The bias is evident in the pseudorange residual for satellite #1 but is not carried to the other satellites' pseudorange residuals because this is an error produced by the calculations involving the pseudorange-rate measurements. The other pseudorange-rate residuals are affected because the filter expectation of the residuals is calculated after

the most recent update cycle from the one-by-one scalar measurement processing

algorithm used in the software, as was discussed previously. The plots in

both appendices clearly show reactionary characteristics because the filter

expects a smaller pseudorange-rate measurement than what was actually

received, thereby forcing the filter to expect a larger pseudorange-rate

measurement from satellite #2 than what was actually received, and so forth,

for all four satellites. However, this effect is diminished as more and more

measurements are received, which is indicated by the smaller transients in the

pseudorange-rate for satellite #4. The one-sigma bounds for the transient

portions of the GPS pseudorange-rate residuals are seen to increase

appropriately. The one-sigma bounds for the offset portion in the pseudorange

residuals of satellite #1 do not increase appropriately because the filter is

unaware of the failure and continues on without compensation.

The real benefit of this comparison is seen when comparing the SAR

residuals of the two filter designs, Figures E.9 thru E.16 and F.9 thru F.15.

The federated filter SAR residuals, located in Appendix F, are unaffected by

the GPS or TAN failures as expected because the SAR dedicated local filter

does not have access to the bad measurement information from the GPS or TAN.

Intentionally, there was no failure induced in the SAR system. The SAR

residuals are the tools with which an efficient evaluation of fault tolerance

can be made. The centralized filter SAR residuals, located in Appendix E, are

adversely affected just before and at the time the simulated GPS failure is

finally removed. Fault indications in the SAR residuals, specifically E.13,

E.14, and E.15, occur before the GPS failure is removed. This might indicate

the disagreement in the centralized filter between the SAR and GPS systems.

There is no actual failure in the SAR system, yet fault detection occurs falsely and perhaps forces the sensor to be taken out of the navigation solution. Therefore, isolating the failures in the centralized filter would be much more difficult.

The federated filter TAN ground clearance residual in Appendix F, Figure F.16, shows the same type of residual behavior characteristic of a constant bias failure, as was shown in the GPS pseudorange-rate residuals. The residual spikes at 4500 and 6500 seconds indicate the TAN failure and reset. For the federated filter, the GPS failure is not affecting the TAN dedicated filter residuals because there is no information fed back to this local filter by the master filter and the local filter is therefore not receiving any of the bad measurement information. This occurs as expected and by design.

The centralized filter TAN ground clearance residual in Appendix E, Figure E.16, shows the effects of the GPS failure and as a secondary consideration, it also shows the relative effects of the SAR residual impact due to the GPS failure. The filter "sees" a SAR failure when actually the system is operating normally. There is still a transient at 4500 and 6500 seconds, but the filter reacts much differently to this information. Additionally, it would appear that the filter believes the TAN has failed during most of the entire mission, also possibly incorrectly convincing an FDI algorithm to remove it from the navigation solution.

4.4.2 GPS Ramp Bias and TAN Constant Bias Simulated Failures

The GPS ramp bias and TAN constant bias failures were implemented in the same fashion as the previous constant bias failures. The gradual failure scenario is the most important failure simulation because it is the most

difficult to detect. The objective is to show how each filter design reacts to these gradual failures and what difficulties might be experienced by each.

As stated earlier, the TAN constant bias failure was left unchanged as a contrasting comparison with the ramp bias failure. The GPS ramp bias affects the same pseudorange-rate residual for satellite #1 during the exact same portion and duration of the mission. This, too, allows for a contrasting comparison with the constant bias failure. Appendix G displays the residual plots for the centralized filter design, while Appendix H displays those of the federated filter design.

As was seen for the constant bias failure, the pseudorange residual for satellite #1 for both filters, Figures G.1 and H.1, has been affected by the ramping failure. The residual values ramp downward from zero-mean in an apparent display of the filters' erroneous expectations due to the calculations from the pseudorange-rate measurements. Again, no other pseudorange residuals from the other satellites for either filter are affected.

For this scenario, the pseudorange-rate residuals, Figures G.2 and H.2, have no initial transient, but do have a final transient indicating that the ramp failure has been removed. Both filter designs allow the failures to ramp and then track them with no detection of the failure at all, except in the pseudorange residual. The filter assumes normal operating conditions during this failure and then settles back into normal operations again after the transient. Therefore, a transient in the behavior of a residual sequence does not necessarily indicate a failure. It may, in fact, represent the removal of a failure. This poses a problem for failure identification. Also, as observed previously, the TAN failure has no affect on the dominant GPS residuals in either filter design implementation.

The residuals for the remaining satellites, Figures G.3 thru G.8 and H.3 thru H.8, behave as expected from the observations of the previous constant bias simulations. Only the pseudorange-rate residuals are affected.

A similar reaction from the SAR residuals, Figures G.9 thru G.15 and H.9 thru H.15, was observed. In the centralized filter design, Appendix G, the residuals are adversely affected. The federated design, Appendix H, shows no impact at all, as expected.

Likewise, the TAN residual reacts as expected for the federated design, Figure H.16, showing no impact from the GPS failure, and only its transient behavior for its own constant bias failure. The TAN residual for the centralized filter, Figure G.16, is affected adversely by the GPS failure and mistaken SAR system failure, while showing the transients due to its own constant bias failure.

The two filter designs suffered from failure detection problems under these simulated ramp bias failure circumstances. While the federated filter did not show any isolation problems with regard to which sensor was experiencing the failure, it did not appear to provide any advantage over the centralized filter in detecting the failure.

It should be noted that the first part of fault tolerance involves detection, which neither filter was able to do until the SAR and TAN systems disagreed with the GPS. The two filter designs treat this information differently. The centralized filter would appear to vote the SAR and TAN sensor measurements out of the navigation solution, while the federated filter would allow the SAR and TAN dedicated local filters to vote the GPS local filter out of the master filter navigation solution. However, this would not

4-24

occur until the errors in the GPS local filter solution become large enough for the SAR and TAN filters to detect.

Because the GPS sensor is the most dominant in accuracy, it would be difficult to detect an impact in the centralized filter's GPS residuals due to any other, less-accurate, sensor having failed. This was shown by the failure of the TAN sensor in the latter parts of the simulations. However, the federated filter, by design, does not allow corruption of the other local filter navigation solutions when a particular sensor is failed, regardless of the sensor's accuracy.

4.4.3 Fault Detection of Simulated Failures

The purpose of obtaining the residual output data files for this thesis is to apply an algorithm, such as the chi-squared algorithm, with the intention of detecting a failure in a subsystem of a complex navigation suite. The detection algorithm is applied to the residuals for each sensor from both of the filter designs. The centralized filter provides all of the residuals necessary, while the federated filter residuals are obtained from the three dedicated filters within its architecture. Appendix I displays the plots created from the application of the chi-squared algorithm on these sets of residuals. There are six plots, three for each filter, and there is one plot for each sensor used, GPS, SAR, and TAN.

As expected, because the GPS residual behaviors, Figures I.1 and I.4, were so similar for the two filter designs, the two chi-squared plots indicate almost identical detection information. The window of calculation for these plots was chosen to be 25 sample periods. This choice provided sufficient evidence of the failure simulated in the GPS sensor. While this failure does

not appear to be obviously detectable from these plots, the failure was intended to represent a minimum bias value which could be possibly detected. If the threshold for detection were to be set very low, it is conceivable that the fault alarm would be set off at about the 3000 second mark. The plot peaks at 3500 seconds for this failure and then resumes apparent normal operations.

The SAR residual behavior for the federated filter, Figure I.5, indicates normal operations. There is no failure, so the threshold would be set far above these peaks. The SAR residual behavior for the centralized filter, Figure I.2, clearly implies failed conditions beyond the 3000 second mark. The occurrence of the fault alarm is inevitable even though there was no SAR system failure. This shows that isolating the failure in the centralized filter would be complicated.

The window for calculations in the detection algorithm for these sets of SAR residuals was chosen to be 5 sample periods. It was necessary to choose this number because of the small number of measurements available to the filter. One set of residuals was available only every 100 seconds. This put a constraint on the algorithm's performance. Although it was small, suffi-cient response was obtained with this value for the window. A larger window was attempted, but the chi-squared plot did not provide enough information regarding the relative time of failure. It simply ramped upward, peaked at about 3500 seconds, and then ramped downward to zero again at 7200 seconds. This was deemed insufficient for detection considerations.

The TAN residual behavior for the federated filter, Figure I.6, barely indicates the failed condition of the TAN sensor at 4500 seconds. The transient upon the removal of the failure is more evident than the initial

transient. Again, this failure exhibits the minimum constant bias detectable by this algorithm.

The TAN residual behavior for the centralized filter, Figure I.3, allows for very easy detection at the 2000 second mark, but the true TAN failure does not occur until 4500 seconds. The failure at 4500 seconds is easily detectable, but the magnitude of the failure indication is far less than for the period where the TAN residual is affected by the GPS failure. The potential for failure misidentification is indicated by this behavior.

The window for calculations in the detection algorithm for the TAN residuals was chosen to be 15 sample periods. This proved to be adequate for both filter designs because the measurements were available every 10 seconds, and the plot characteristics were quite responsive to the failure indications.

The window size for any of the applications of the chi-squared detection algorithm was a variable. Although several values could have been used for GPS and TAN residuals, the window size was left relatively small intentionally to maintain an equitable view when considering the constrained calculations of the SAR detection scheme.

This detection algorithm does not allow the individual residual failure indications to be identified. All of the residuals for one sensor are handled by the algorithm simultaneously, yielding only an indication of a whole sensor failure. More information might be obtainable if the residuals were treated independently for failure identification. This could possibly provide more options for reconfiguration of the failed system.

4.5 *Additional Fault Tolerance Topics*

As a result of the literature review and personal consultations with Dr.

Neal Carlson (1,2,3,4), additional comments should be made regarding the FDI capability of the federated filter design. The objective of this section is to reveal the additional possibilities available to the federated filter design with regard to fault tolerance and detection. Specifically, residual monitoring of the local filters is not the only method for fault detection in the federated filter design. The federated filter supports sensor fault detection at both the local and master filter levels, to include the INS, via residual monitoring.

The individual local filters can accumulate sufficient sensor information such that a sensor soft failure has more time to reach a detectable magnitude before it meets the residual tolerance testing in the master filter. Each local filter maintains its corresponding sensor bias states. Once a sensor fault has occurred, an uncorrupted navigation solution will still exist in the other local filters. Thus a clean navigation solution will be available immediately.

The master filter also provides some additional capability to detect and isolate INS failures. The propagated master filter solution allows an INS fault to be more visible than in the local filter solutions, when the master filter uses no external sensor data between fusion updates. This is especially so when the master filter contains an enhanced INS truth and filter model in comparison to the local filter models. Up to this point, the INS failures have not been considered for detection, but do pose an interesting problem. Gradual INS failures can be the most difficult failures to detect.

The no-reset mode of the federated filter design provides an important advantage regarding fault tolerance. Once a fault has been detected from a sensor-dedicated filter, a new master filter solution can be generated

immediately from the remaining local filter solutions. In contrast to this, unless detected, the centralized filter will distribute a failure into all of the navigation state and sensor bias estimates. If the centralized filter incorporates the bad information into its solution, it becomes irreversibly corrupted. Consequently, the only safe means of recovery is to reinitialize the centralized filter solution.

4.5 Summary

The topics covered in this chapter encompass the essential aspects of the research accomplished for this thesis. The error state estimation accuracies of the centralized and federated filter designs were witnessed and indicated excellent results for both filters. The residual monitoring from both of the filter designs revealed some of the fault tolerance and fault identification capabilities. One of the most difficult faults to detect was investigated, the ramp bias, and revealed some of the difficulties associated with gradual sensor failures. Chapter V provides the conclusions and recommendations drawn from these results.

V. Conclusions and Recommendations

5.1 Overview

The comments of this chapter are provided as an objective opinion based on the results and observations, along with the theoretical considerations, obtained from the research effort put forth. Discussions of the main purposes of this thesis follow, including the comparison of the error state performances, the fault detection considerations, and the software tool itself, DKFSIM Version 1.1. Recommendations for further study are also provided as additional insight into the potential for future usage.

5.2 Filter Error States

The error-state estimation performance comparison of the centralized and federated filter architectures establishes a reasonable basis for considering the federated filter design as a viable alternative to centralized filtering for future filtering implementations. The two filter designs performed with very nearly the same accuracy when estimating the 29 error states using the ten-run Monte Carlo analysis. The critical aspects of this comparison involved a visual and numerical analysis of the filters' performances in estimating the INS error states.

Although it was not necessary to compare the sensor bias states, one-for-one, the comparison done in this thesis depicted the two filter error state estimation accuracies also as nearly equal. The sensor bias states are not considered the prime motivation for this effort. However, with the exception of some of the one-sigma bounds of the federated filter being larger than those of the centralized filter, the overall comparison of the error-state estimation performances showed the federated filter's performance resulted in no significant loss of accuracy compared to the centralized filter's performance. This eliminates the contention that the federated filter should be ruled out of the possibilities for future implementations in aircraft avionics systems because of the disparity in the accuracies of the two filters. This result is based on the use of the GPS, SAR, and TAN sensors with the medium-accuracy strapdown INS, under normal conditions.

Also, the filter designs under scrutiny performed excellently considering the variations in the flight dynamics of the aircraft for the flight profile used. It is easily seen that the two filter designs compare quite well in all aspects during all phases of the simulations.

Additional testing under simulated sensor and INS failure conditions would allow the extended comparison of the two filters' performances. A comparison of the error-state estimation accuracies would prove interesting when several different failure conditions could be simulated during the most adverse flight conditions in this flight profile. Filter stability factors could prove to be an additional deciding factor under certain high-dynamic, failed-sensor circumstances. Improved modelling of the subsystems would facilitate enhanced simulations involving higher flight dynamics. It appears that the error-state performance suffers during these high-dynamic maneuvers.

The best way to seek out the solutions to these problems is to attempt to simulate them more efficiently and then to allow for proper compensation in the systems.

5.3 *Fault Detection Considerations*

The conclusions formed regarding fault detection and fault tolerance make up the bulk of the objective of this thesis because the two filter designs were deemed comparable in their estimation accuracies. Although the software did not lend itself easily to the task of fault detection, it proved to be adequate for this purpose.

Residual monitoring of the local filters was decidedly the most efficient implementation of fault detection of all the options available. This task was necessary, not only to monitor the sensor-dedicated local filter residuals, but it also was necessary because local filter #1 implements the centralized filter operations. The centralized filter residuals would have to have been monitored under any circumstances. Thus, the objective of detecting a sensor failure for a centralized or federated filter was accomplished while minimizing the source code changes.

The performances of the two filter designs under failed-sensor conditions revealed their respective fault tolerance capabilities. The obvious choice for fault tolerance would have to be the federated filter design using the no-reset mode because the sensor-dedicated filters' measurements and navigation solutions are kept independent. Other federated filter modes do not offer the same tolerance, and yet they also do not offer sufficient justifica-tion to select them over the no-reset mode with regard to estimation accuracy.

The centralized filter suffers from the lack of accurate sensor fault identification when a failure occurs. It is noted that the exception is the hard failure which exists and is detected immediately, and the filter rejects that bad information immediately. If the detection is not immediate and the failure information is incorporated into the navigation solution, missed detections and false alarms appear inevitable under almost any failed conditions. Thus, a perfectly good sensor may be removed from the navigation solution. This is based upon the level of confidence the filter places on the particular sensor's accuracy. The GPS is so dominant that the filter would have more confidence in its measurement information than that of the SAR or TAN under these circumstances. This is the reason for the GPS residuals not reflecting any impact from the TAN failures.

The federated filter allows for a more efficient voting scheme because the residuals from the local filters are not affected by the other sensor's bad information. This is limited to the level of accuracy of each of the respective sensors. It would take a great deal of error on the part of the GPS before the TAN and SAR could vote that filter out of the master filter navigation solution.

While the master filter will enhance the fault detection of the INS, it is still considered efficient to monitor the residuals of the local filters for sensor failures because of the advantages of the separation of the navigation solutions in those filters. Any further fault detection enhancements to DKFSIM should be seriously considered as viable alternatives to the existing fault detection schemes implemented as part of this thesis.

Additionally, failure modelling would allow more complicated failure modes in the sensor operations. Perhaps, for the GPS system, allowing for three or fewer satellite operations, or even satellite selection based on optimum Geometric Dilution of Precision (GDOP), would prove fruitful for further study.

Further, INS failure modes could easily be implemented in source code. Gradual failures of the INS appear to be a topic of great concern. The failure simulations in the SAR and TAN systems can be relatively de-emphasized in both filter implementations because the GPS obviously dominates the estimation processes due to its greater accuracy. The SAR and TAN systems proved most useful when considering the impact of GPS failures on other systems and the aspects of filtering. It appears to be worth investigating the SAR residuals' behavior for the centralized filter prior to the large reactions while being affected by the GPS failed conditions. The plots were so insignificantly small prior to these harsh responses that the residuals' characteristic behaviors were indistinguishable.

Finally, the central conclusion here is that the detection scheme used does not allow for the determination of the specific residuals affected by any sensor failure, unless, like in the TAN sensor, there is only one residual to affect. The federated filter allows better sensor-fault isolation than the centralized filter. Consequently, the centralized filter suffers from false alarms in the fault detection scheme, while the federated filter does not. This proves to be a critical deficiency of the centralized filter's ability for system reconfiguration under sensor-failed conditions.

5.4 DKFSIM Version 1.1

This version of DKFSIM has a great deal of capability, but with respect to fault tolerant testing, this tool was deficient as received and therefore modified. The ability to simulate the federated filter was limited to two modes, however, the most important mode of operation was available, the no-reset mode. In addition, the centralized filter implementation was available to provide the essentials for the comparison of the two designs.

Improvement to the software could be wide-ranging, covering many aspects. Perhaps, one of the most important aspects of fault detection is the application of the detection algorithm. If the software had this algorithm included such that it could provide detection information to the output data files, then failure analysis, and possibly system reconfiguration, could be performed on line while the simulation is being performed. This might also include some Multiple Model Adaptive Estimation (MMAE) considerations. Thus, enhancing the possibilities for system reconfiguration testing under failed subsystem conditions.

As a further suggestion for software enhancements, it would certainly be beneficial to augment the data output capabilities to include an ensemble average of a ten-run, or more, Monte Carlo analysis. By utilizing an ensemble average for the comparisons, the error-state behavior is more efficiently characterized, thus yielding a more valid comparison. Lastly, it would seem important to include all of the ten-run statistics when calculating the one-sigma values for each error state.

5.5 Summary

The potential for follow-on study is great in this area. The federated filter is an excellent implementation of filtering theory. Comparisons of the federated design to the centralized design will likely be carried on into the twenty-first century. Hopefully, any advancements in filtering theory which enable engineers to simplify the task of obtaining a navigation solution under normal or failed operating conditions will be utilized to their full advantage. This study definitely shows that the federated filter is worthy of serious consideration under any circumstances.

Bibliography

1. Carlson, Neal. "Federated Filter for Fault-Tolerant Integrated Navigation Systems". *IEEE PLANS, Position Location And Navigation Symposium, 1988*: 1-10 (Dec 1988).

2. Carlson, Neal. "Federated Square Root Filter for Decentralized Parallel Processes". *IEEE Transactions on Aerospace and Electronic Systems, 26*: No. 3, 517-525 (Apr 1989).

3. Carlson, Neal. "Information-Sharing Approach to Federated Kalman Filtering". *NAECON, 1988, National Aerospace and Electronics Conference*: 1-10 (May 1988).

4. Carlson, Neal, President. Telephone interviews. Integrity Systems, Inc., Winchester, MA, May 1993.

5. Da, Ren. "The Error Covariance Analysis Algorithms for Suboptimal Decentralized Kalman Filters". *The American Institute of Aeronautics and Astronautics (AIAA) Guidance, Navigation and Control Conference, Aug 1991*: 1983-1988 (Aug 1991).

6. Felter, Stephen. "A Covariance Analysis Technique for the Federated Kalman Filter". *NAECON 1992, National Aerospace and Electronics Conference*: 399-406 (May 1992).

7. Gao, Y. and others. "Experience With the Application of Federated Filter Design to Kinematic GPS Positioning". *IEEE PLANS, Position Location and Navigation Symposium, 1992*: 314-320 (Mar 1992).

8. Hong, Lang. "Adaptive Distributed Filtering in Multicoordinate Systems". *IEEE Transactions on Aerospace and Electronic Systems, 27*: No. 4, 715-724, (Jul 1991).

9. Hong, Lang. "Centralized and Distributed Kalman Filtering in Multi-Coordinate Systems with Uncertainties". *NAECON 1990, National Aerospace and Electronics Conference*: 389-394 (May 1990).

10. Integrity Systems, Inc. *DKFSIM 1.1 User's Manual*. Contract No. F33615-87-C-1520, Document No. TM-89-005. Winchester, MA, Dec 1989.

11. Integrity Systems, Inc. *Technicl Operating Report #3: Distributed Kalman Filter Mission Performance Simulation Plan*. Contract No. F33615-87-C-1520, Document No. TM-89-001-2. Winchester, MA, May 1989.

12. Kerr, Thomas. "Decentralized Filtering and Redundancy Management for Multisensor Navigation". *IEEE Transactions on Aerospace and Electronic Systems, 23*: No. 1, 83-119 (Jan 1987).

13. Maybeck, Peter S. *Stochastic Models, Estimation, and Control, Volume 1*. San Diego, CA: Academic Press, 1979.

14. Maybeck, Peter S. *Stochastic Models, Estimation, and Control, Volume 2*. San Diego, CA: Academic Press, 1982.

15. Maybeck, Peter S. *Stochastic Models, Estimation, and Control, Volume 3*. San Diego, CA: Academic Press, 1982.

16. Mercier, Daniel E. *An Extended Kalman Filter For Use In a Shared Aperture Medium Range Tracker*. MS Thesis AFIT/GA/EE/78D-3, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH, January 1979. (AD-A064191).

17. Roy, S., and others. "Square Root Parallel Kalman Filtering Using Reduced-Order Local Filters". *IEEE Transactions on Aerospace and Electronic Systems, 27*: No. 2, 276-287 (Aug 1990).

18. Smith, M.S. and G.S. Ladde. "Processing of Prefiltered GPS Data", *IEEE Transactions on Aerospace and Electronic Systems, 25*: No. 5, 711-727 (Sep 1989).

19. TAU Corporation. *Common Kalman Filter Development*. Contract No. F33615-86-C-1047, Document No. WRDC-TR-89-1152. Los Gatos, CA, Feb 1990.

20. Wei, M. and K.P. Schwarz. "Testing a Decentralized Filter for GPS/INS Integration", *IEEE PLANS, Position Location and Navigation Symposium, 1990*: 429-435 (Mar 1990)

Appendix A: Centralized Kalman Filter Error State Plots

GENERAL INFORMATION

Each of the error state plots contained in this appendix are obtained from Monte Carlo simulations using DKFSIM Version 1.1. All error state outputs are recorded for the entire 7200 second duration of the simulation. The information provided represents error state outputs at the time of the measurements, after the update cycle, and after the propagation cycle, plotted versus time in increments of five seconds for the centralized Kalman filter algorithm.

Ten simulations were performed for each state. Therefore, the plots provided represent the ensemble average of the error state output data plus the upper and lower one-sigma bounds computed for that state. The one-sigma bound was computed only for the first Monte Carlo simulation.

There are 29 plots, one for each centralized filter error state. Each plot has the state designation along with its corresponding error definition. A listing of the titles of each these figures is found in the List of Figures at the beginning of this document.

KEY TERMS

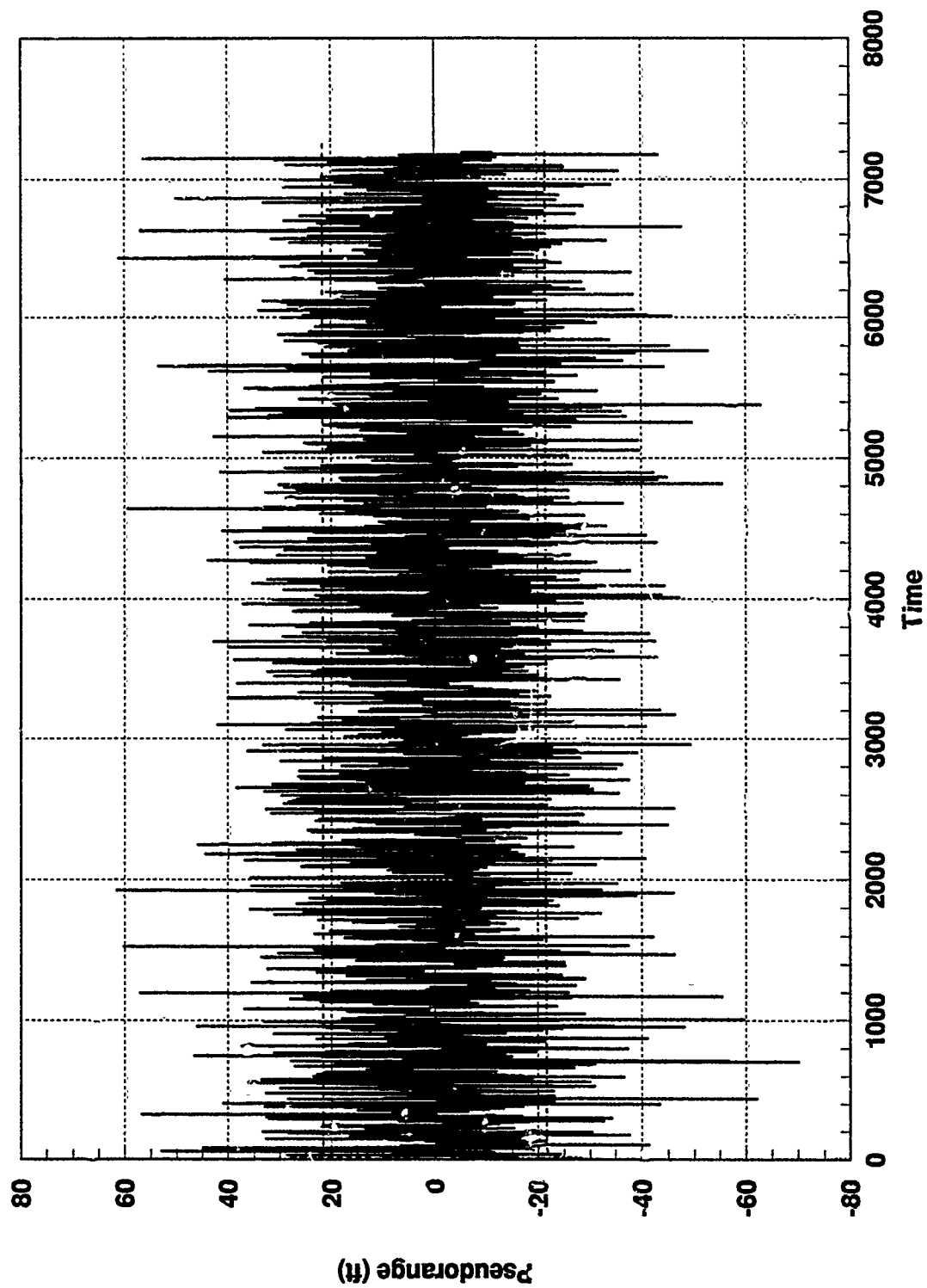1. ECEF: Earth-Centered-Earth-Fixed coordinate frame.
2. Yaw, Pitch, and Roll Axes: Body Frame with respect to the aircraft.
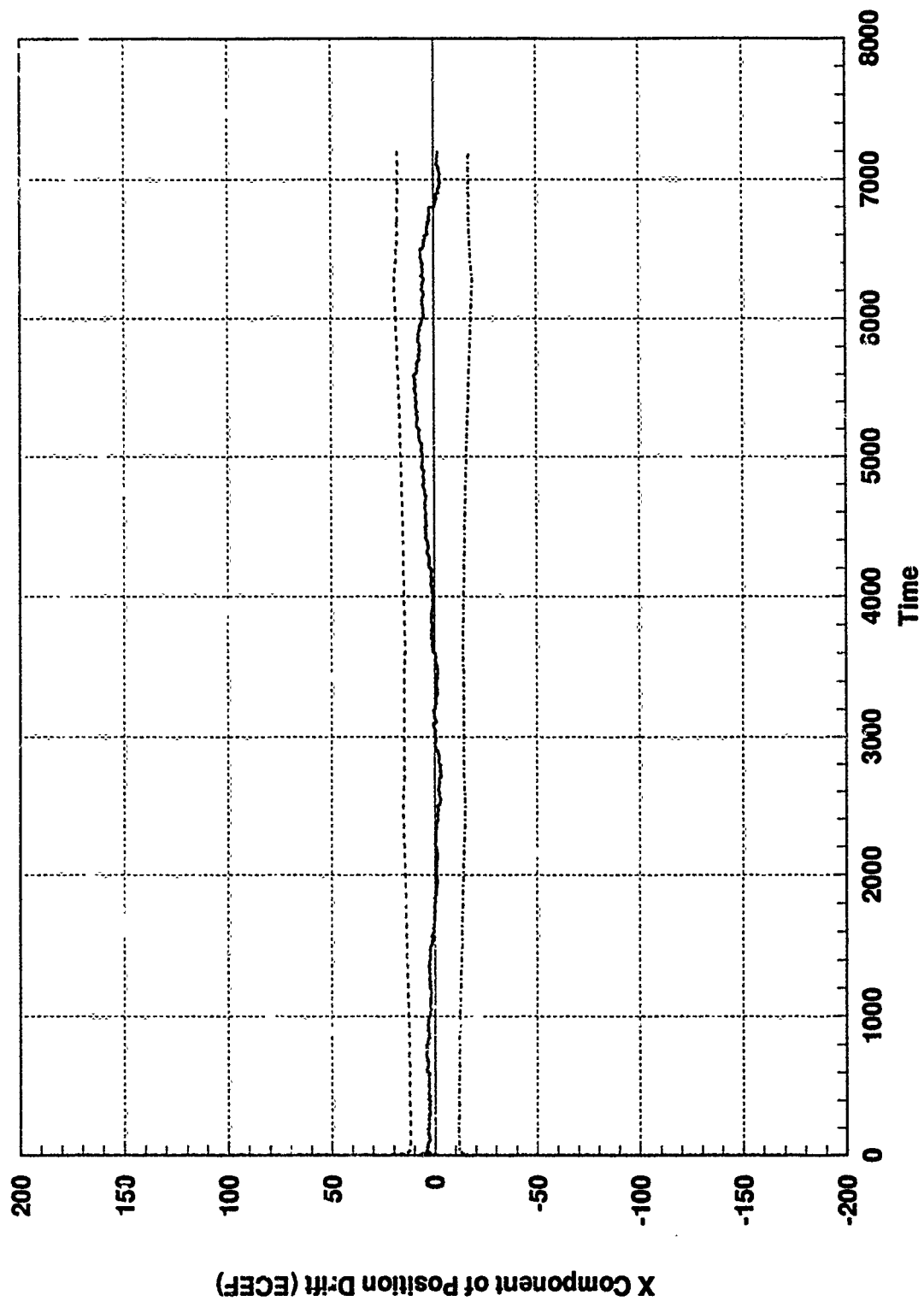
Figure A.1 Centralized Kalman Filter Design, Filter State 1
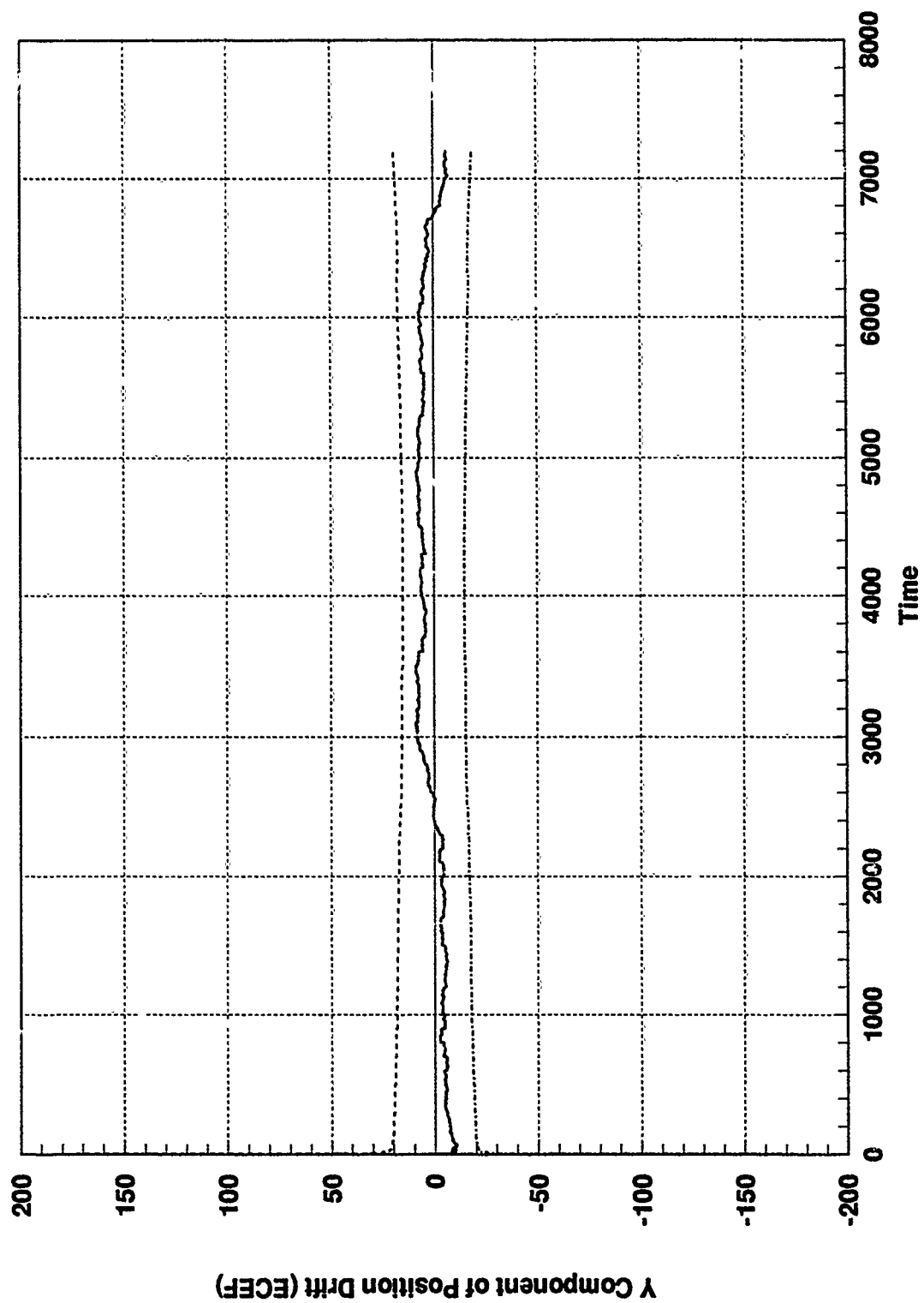
Figure A.2 Centralized Kalman Filter Design, Filter State 2
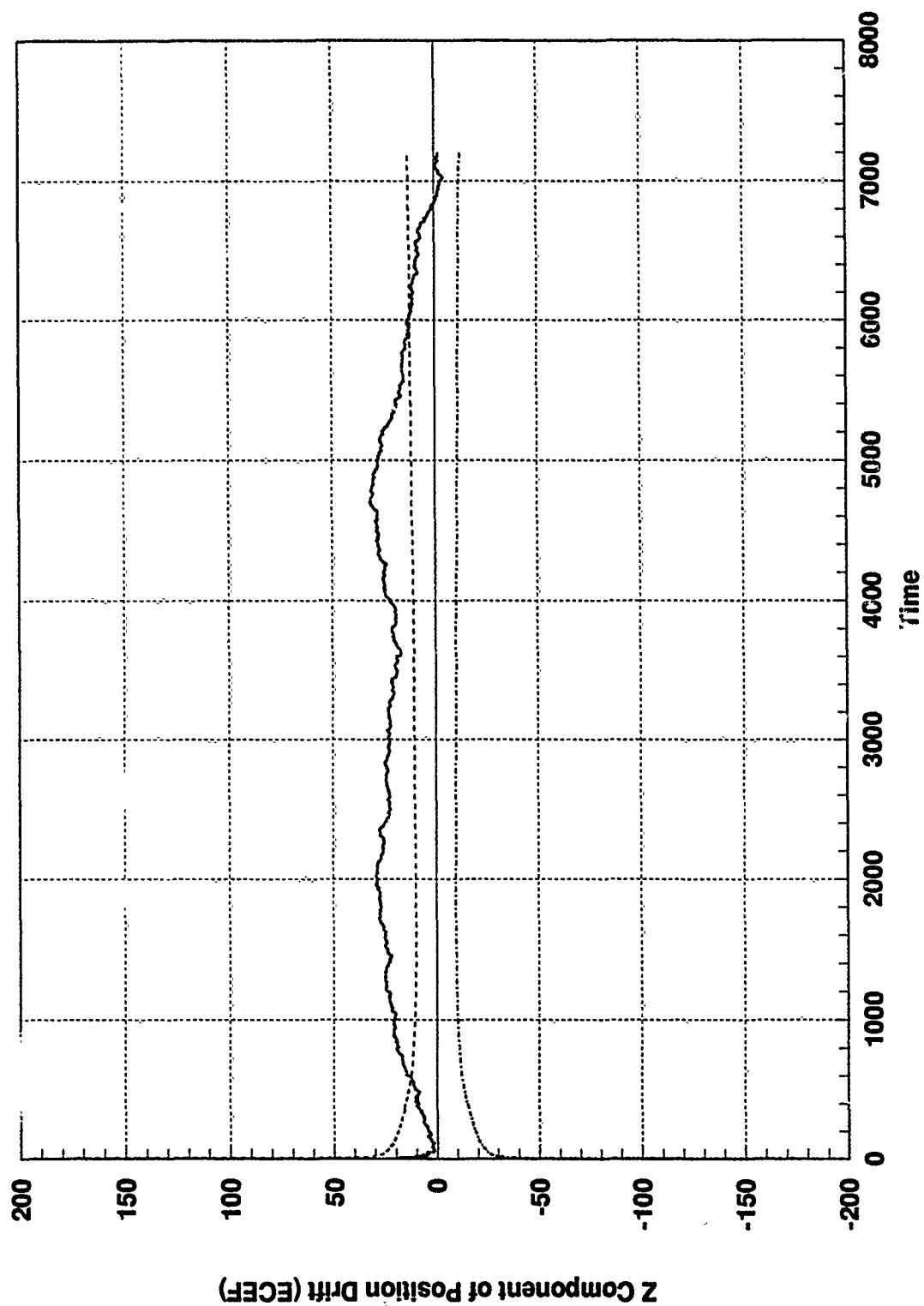
Figure A.3 Centralized Kalman Filter Design, Filter State 3
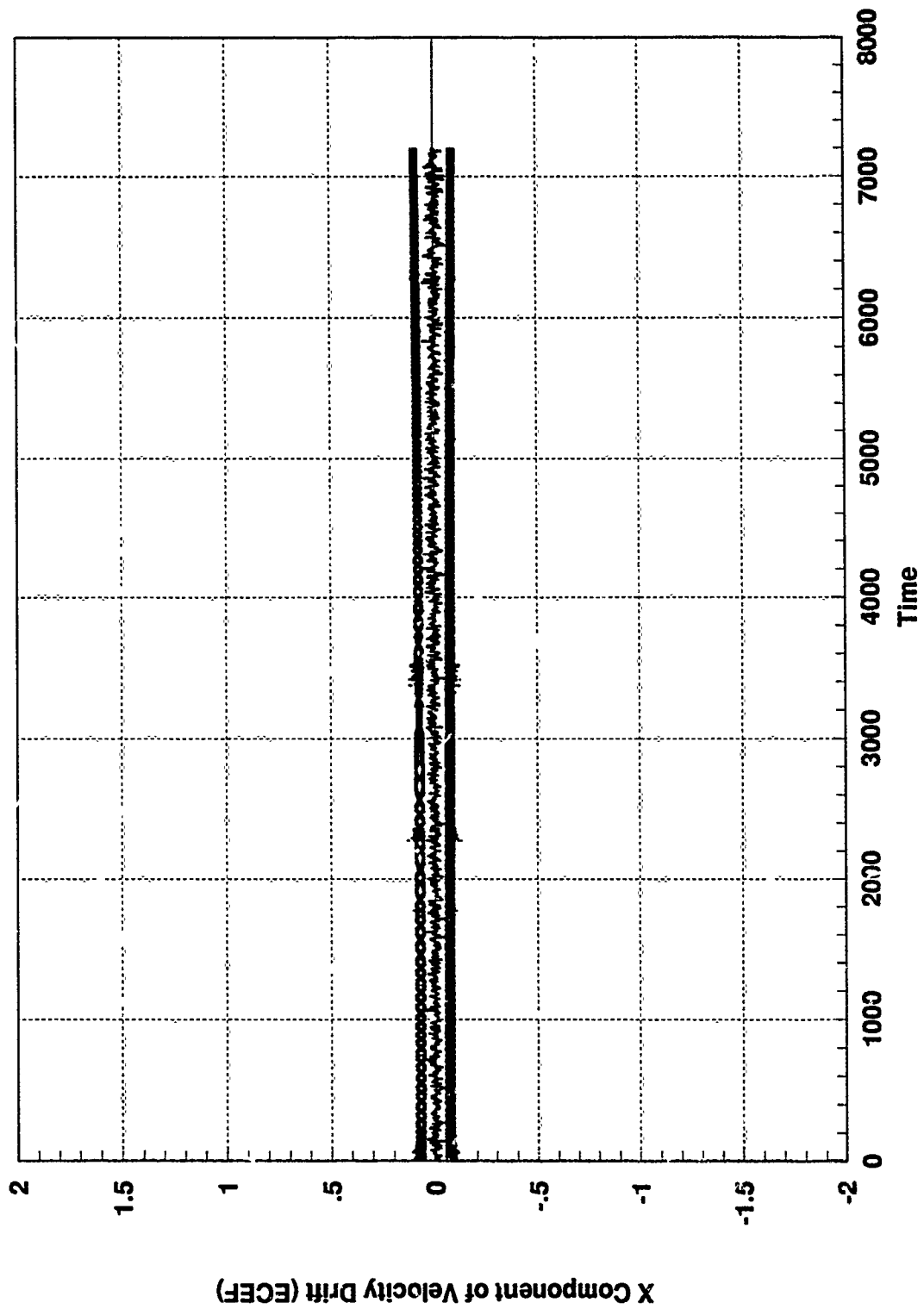
Figure A.4 Centralized Kalman Filter Design, Filter State 4
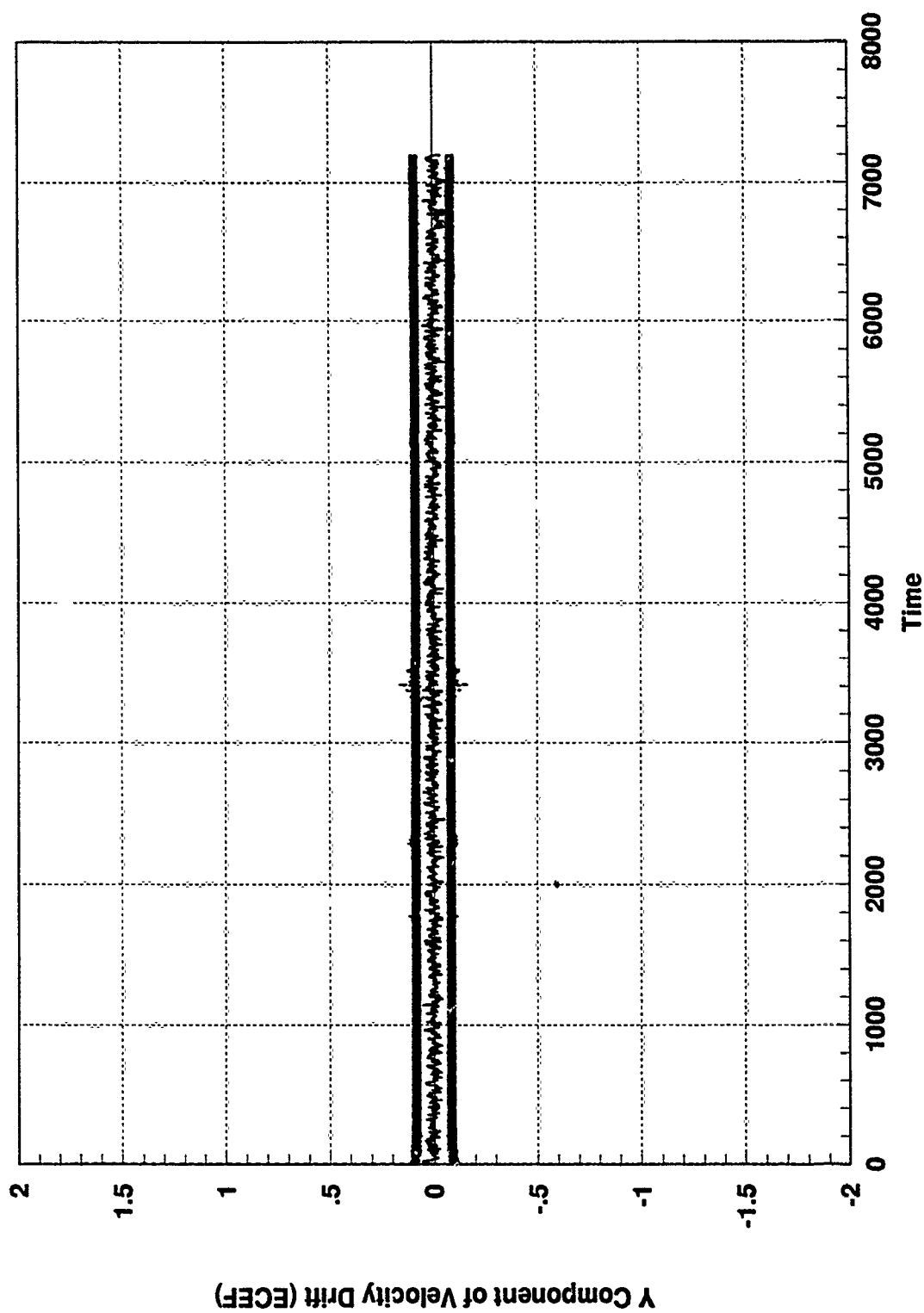
Figure A.5 Centralized Kalman Filter Design, Filter State 5
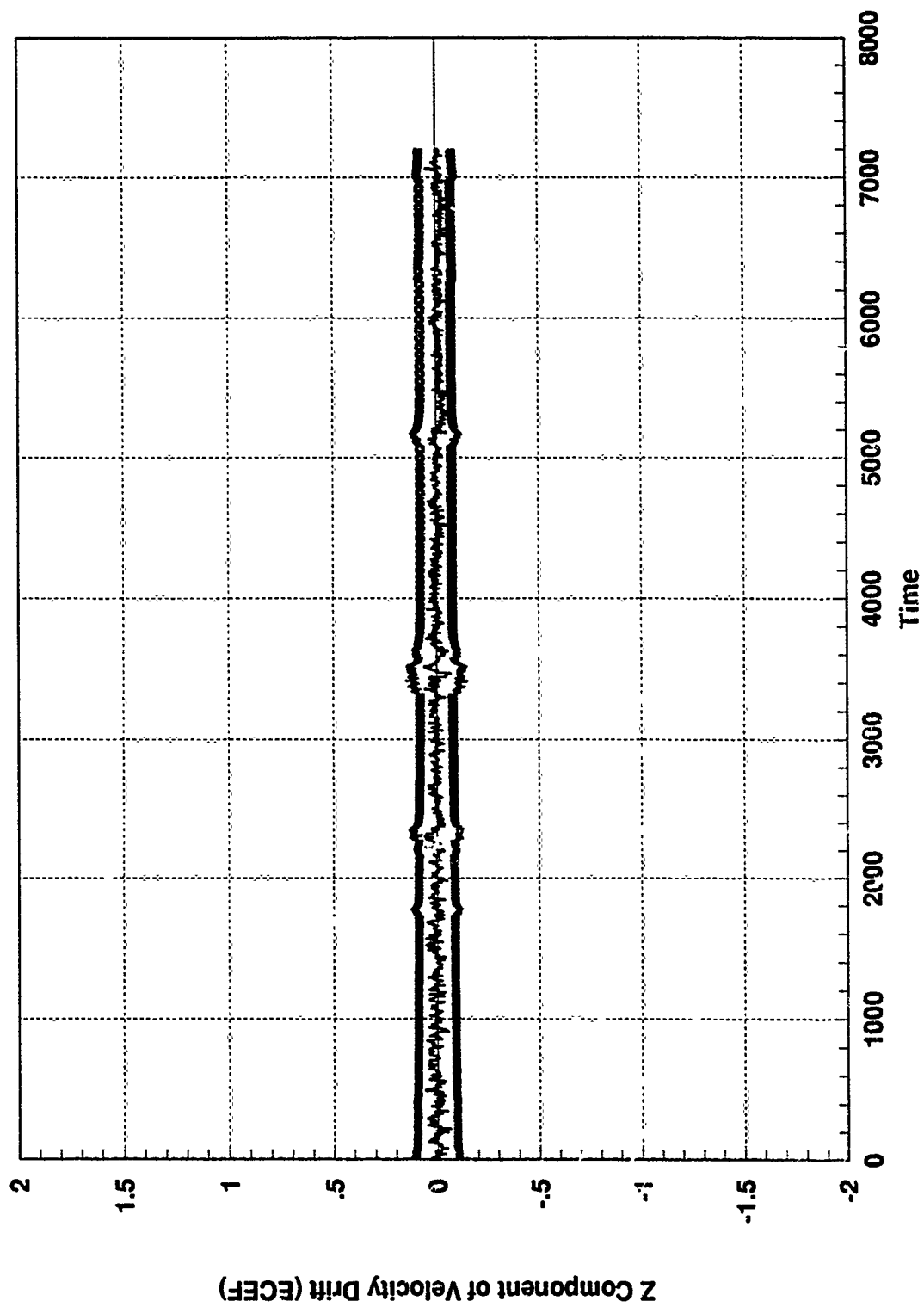
Figure A.6 Centralized Kalman Filter Design, Filter State 6
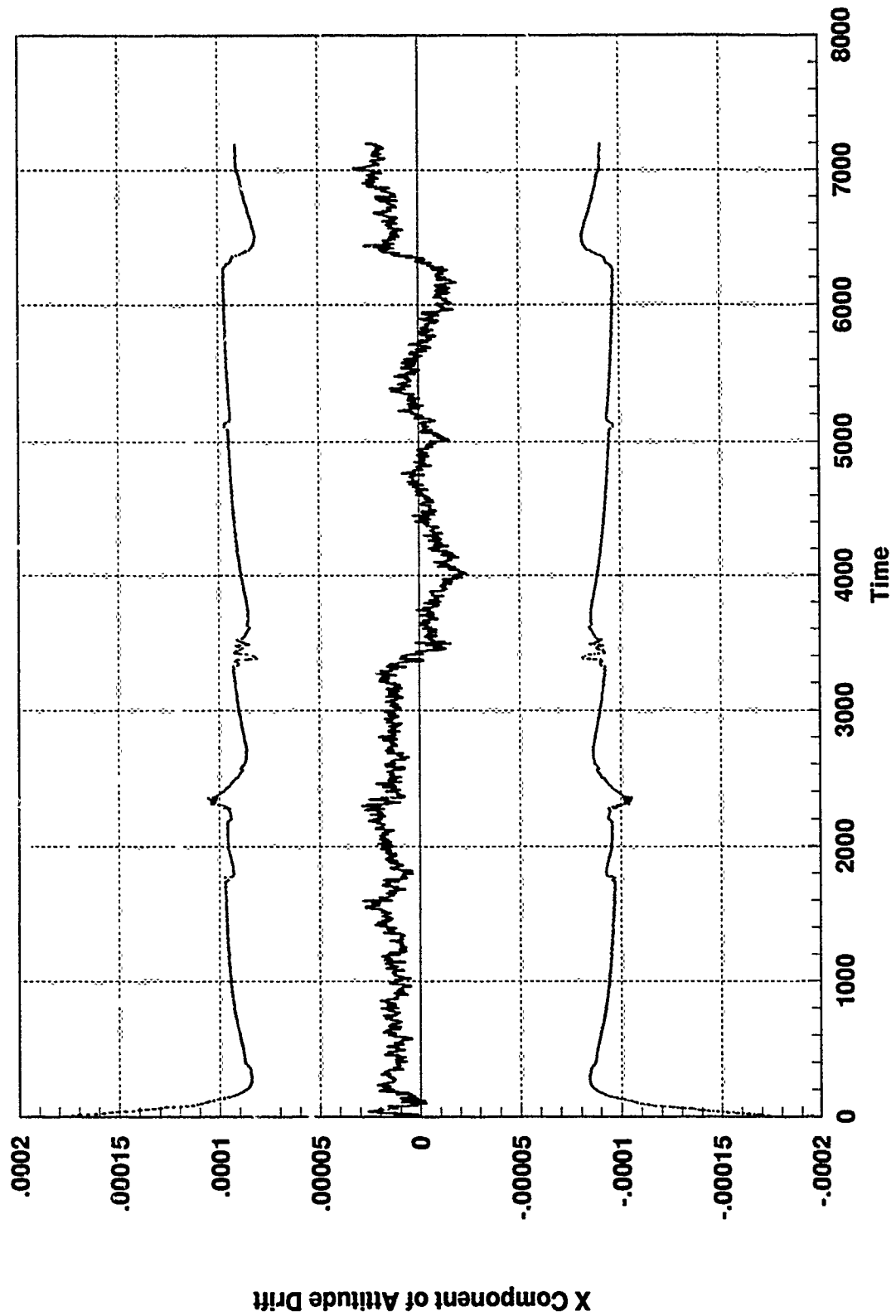
Figure A.7 Centralized Kalman Filter Design, Filter State 7

X Component of Attitude Drift
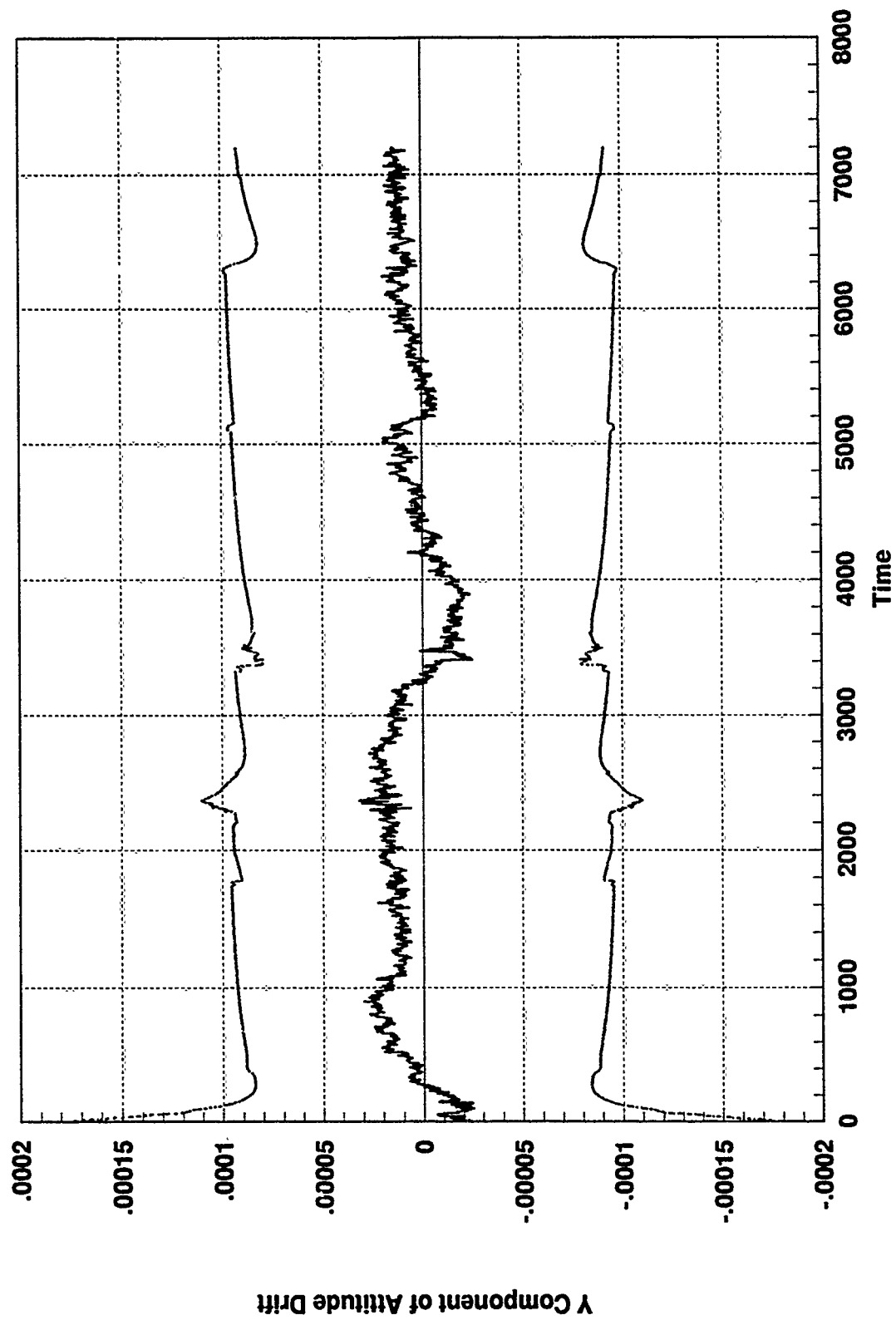
Figure A.8 Centralized Kalman Filter Design, Filter State 8
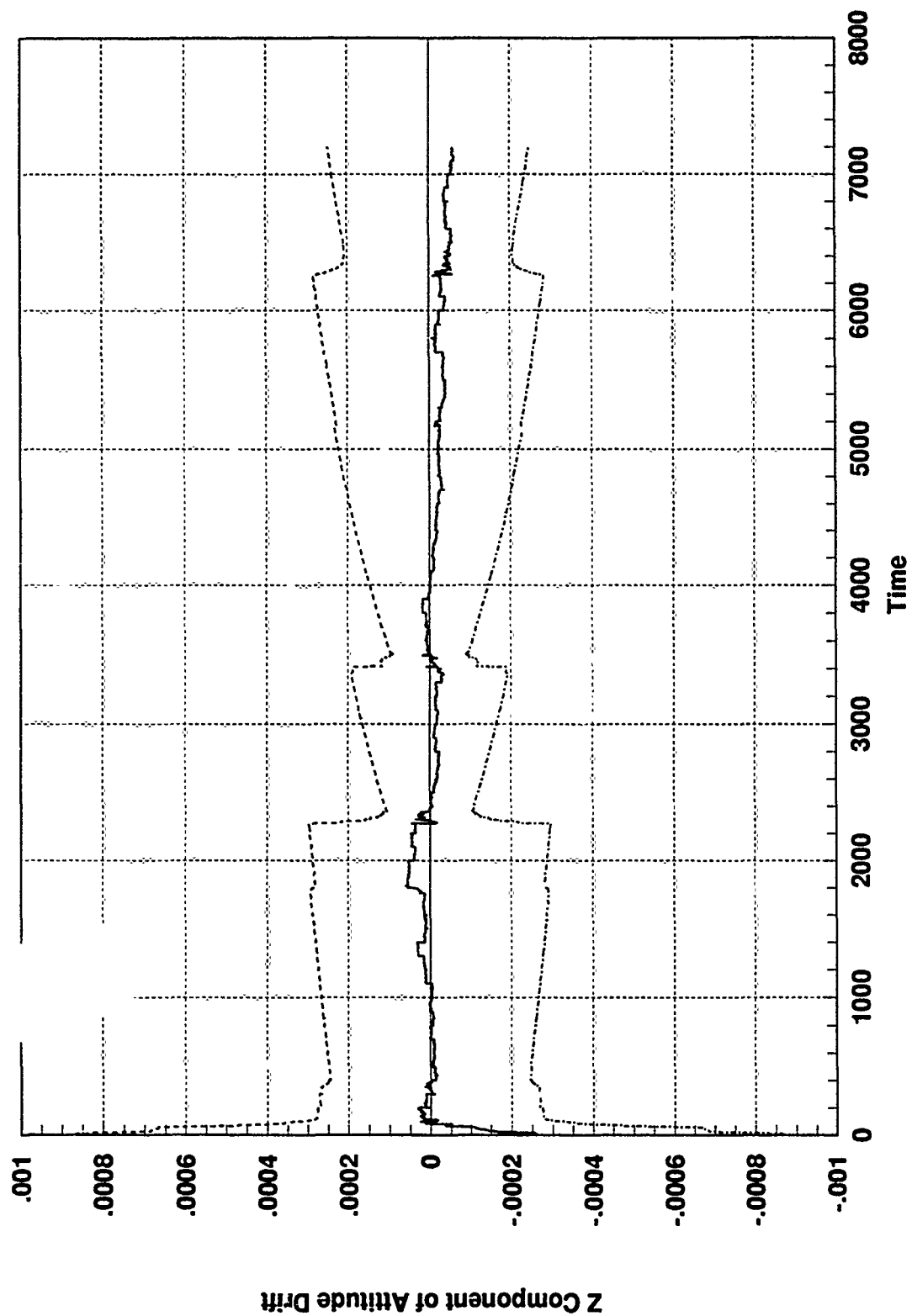
Figure A.9 Centralized Kalman Filter Design, Filter State 9

A-10
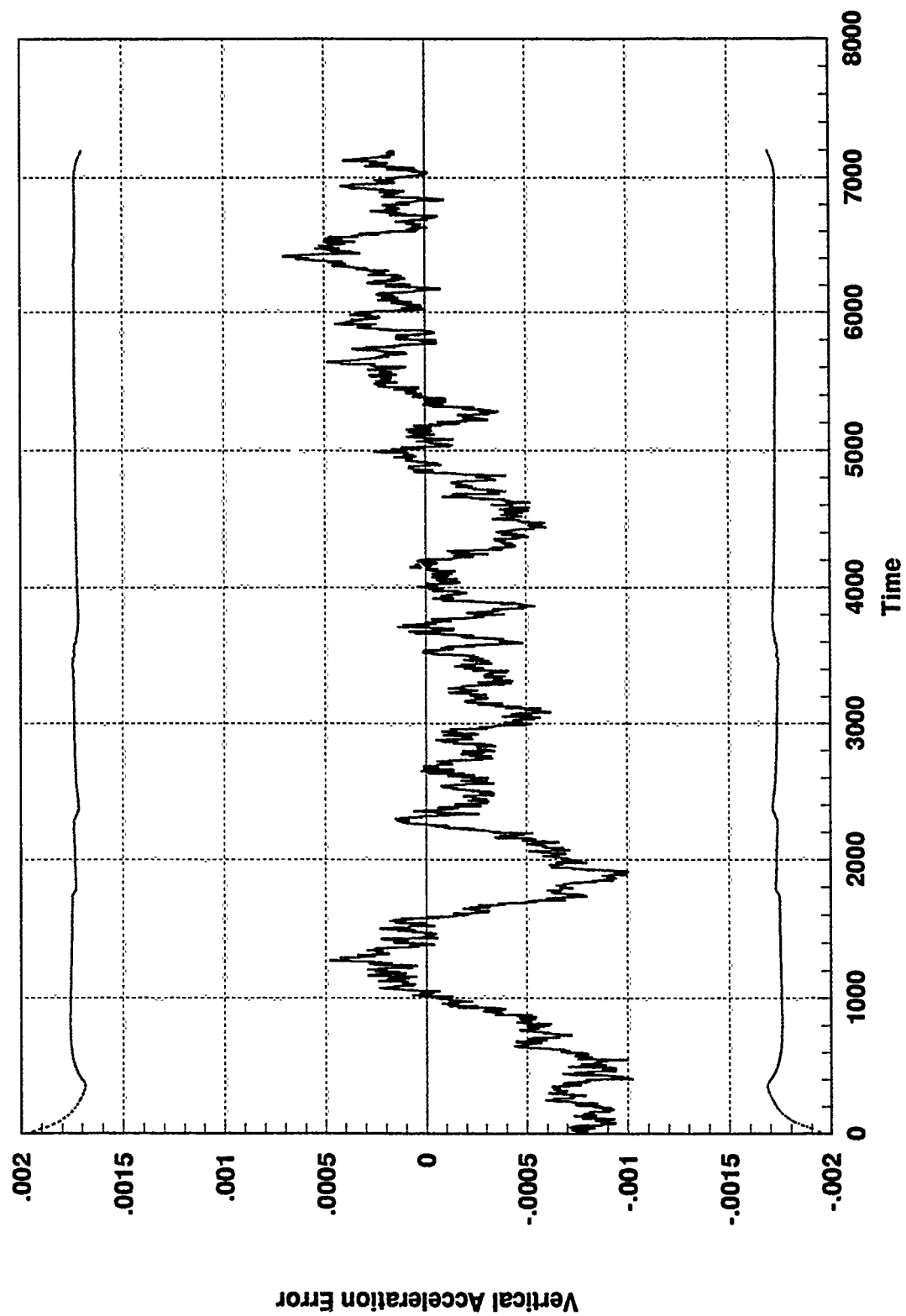
Figure A.10 Centralized Kalman Filter Design, Filter State 10

A-11
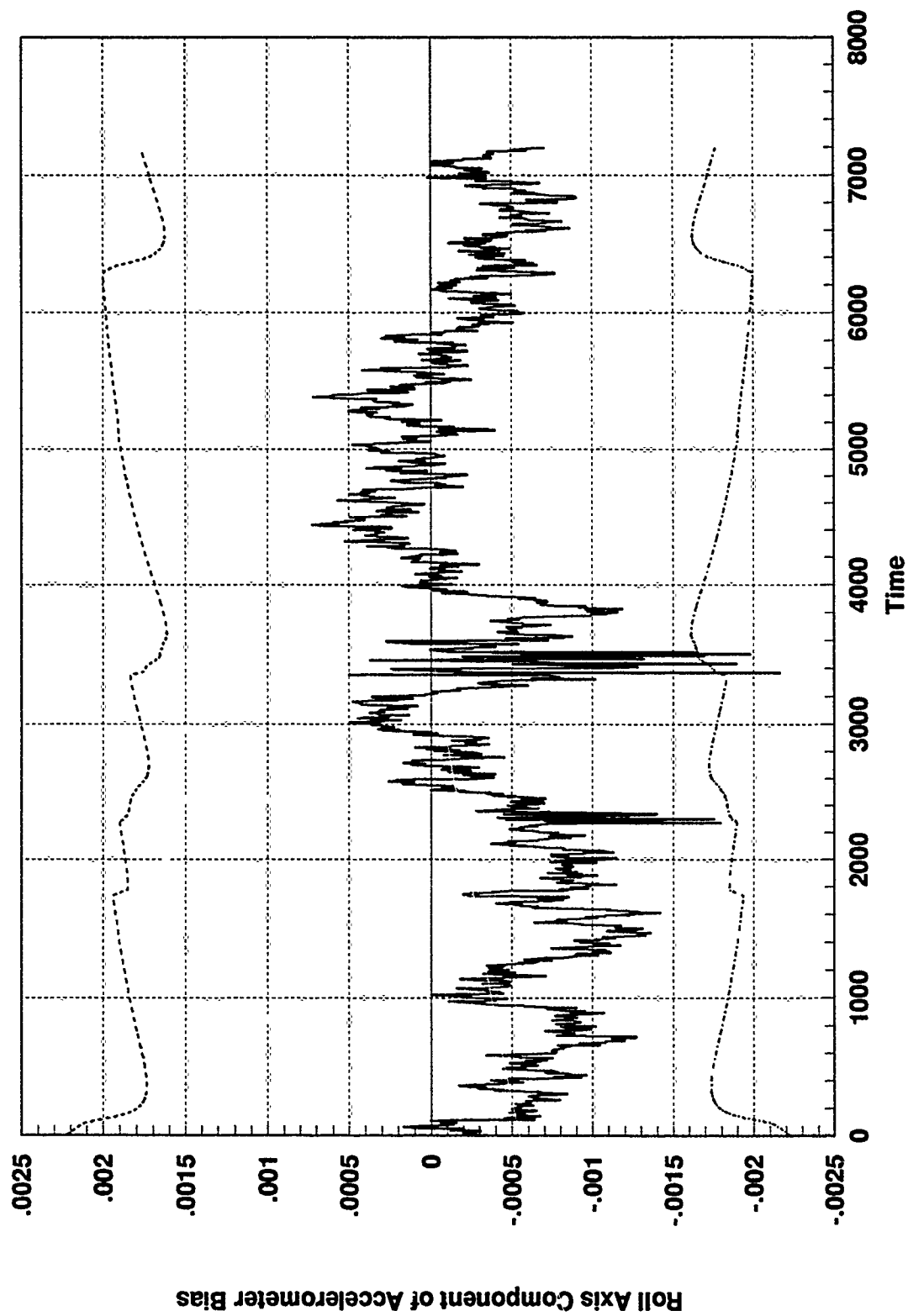
Figure A.11 Centralized Kalman Filter Design, Filter State 11
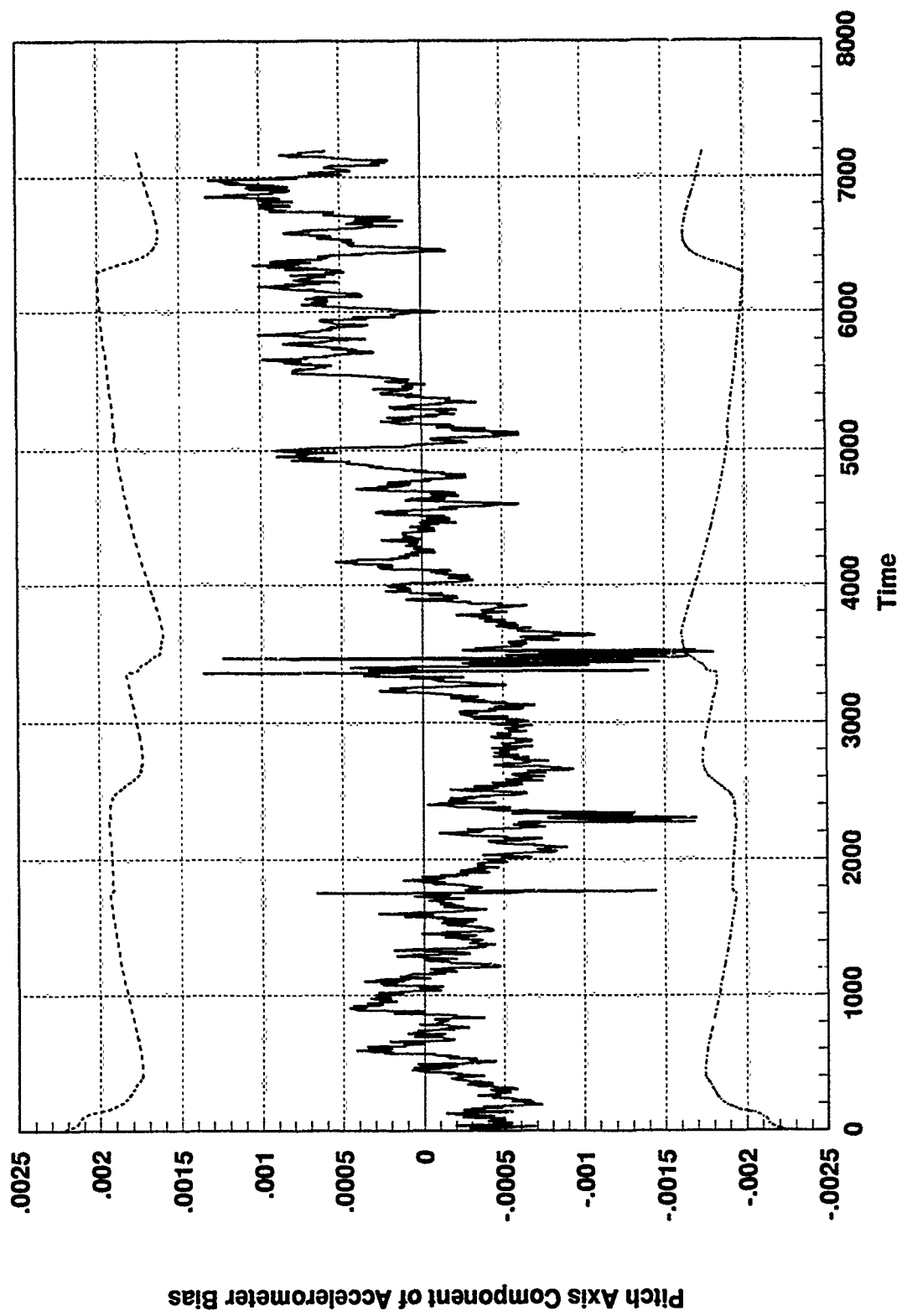
Figure A.12 Centralized Kalman Filter Design, Filter State 12
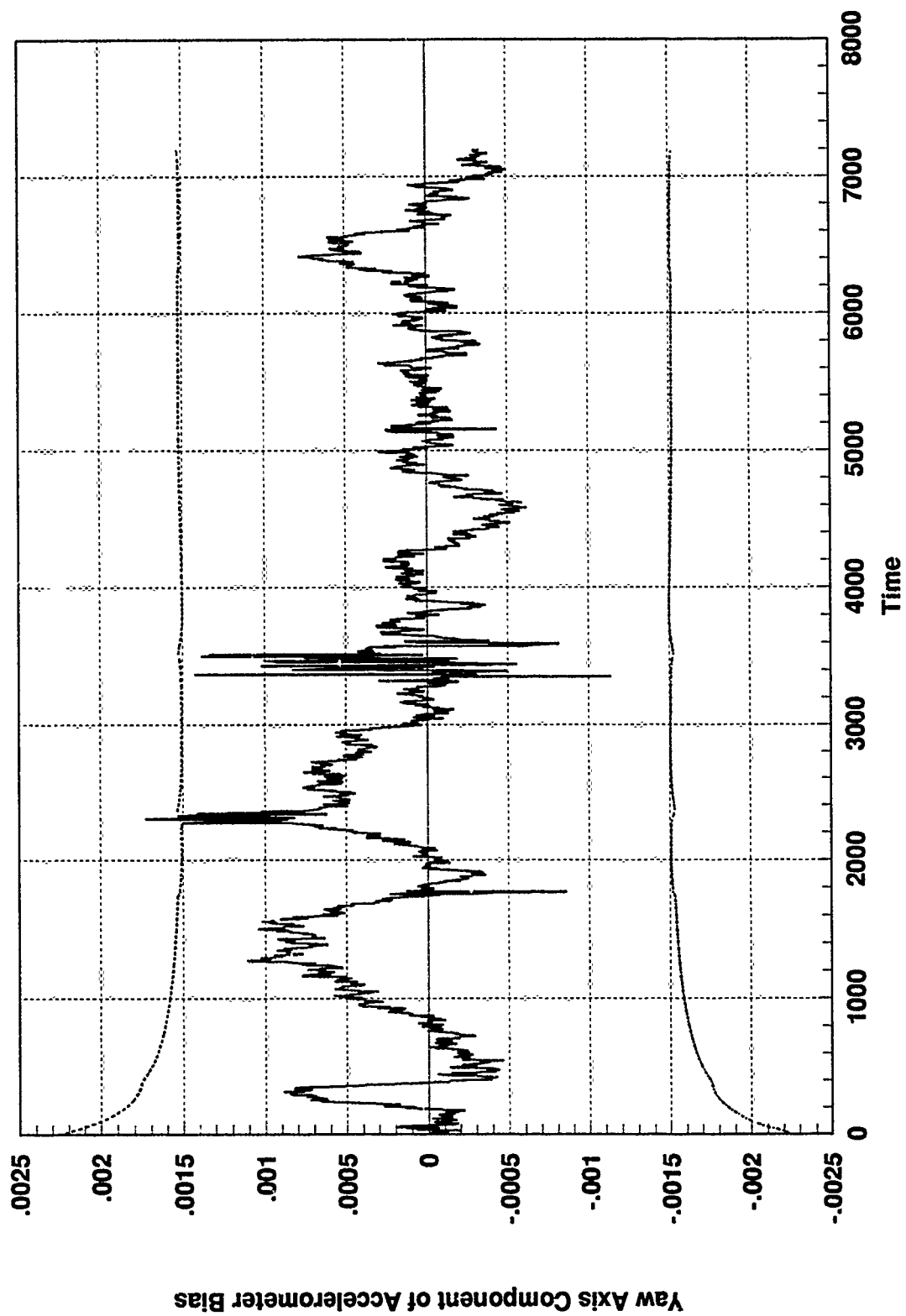
Figure A.13 Centralized Kalman Filter Design, Filter State 13
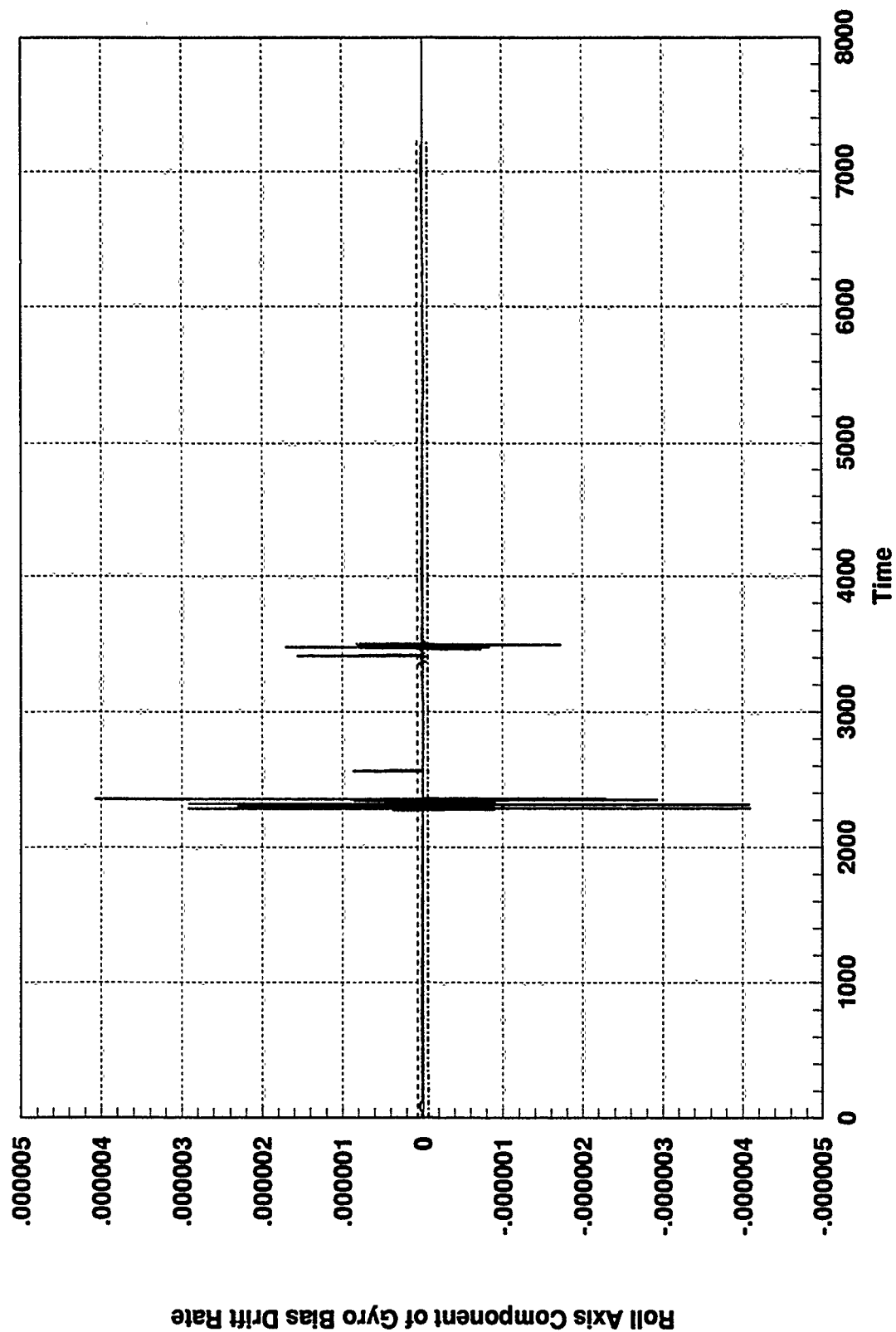
Figure A.14 Centralized Kalman Filter Design, Filter State 14

Figure A.15 Centralized Kalman Filter Design, Filter State 15
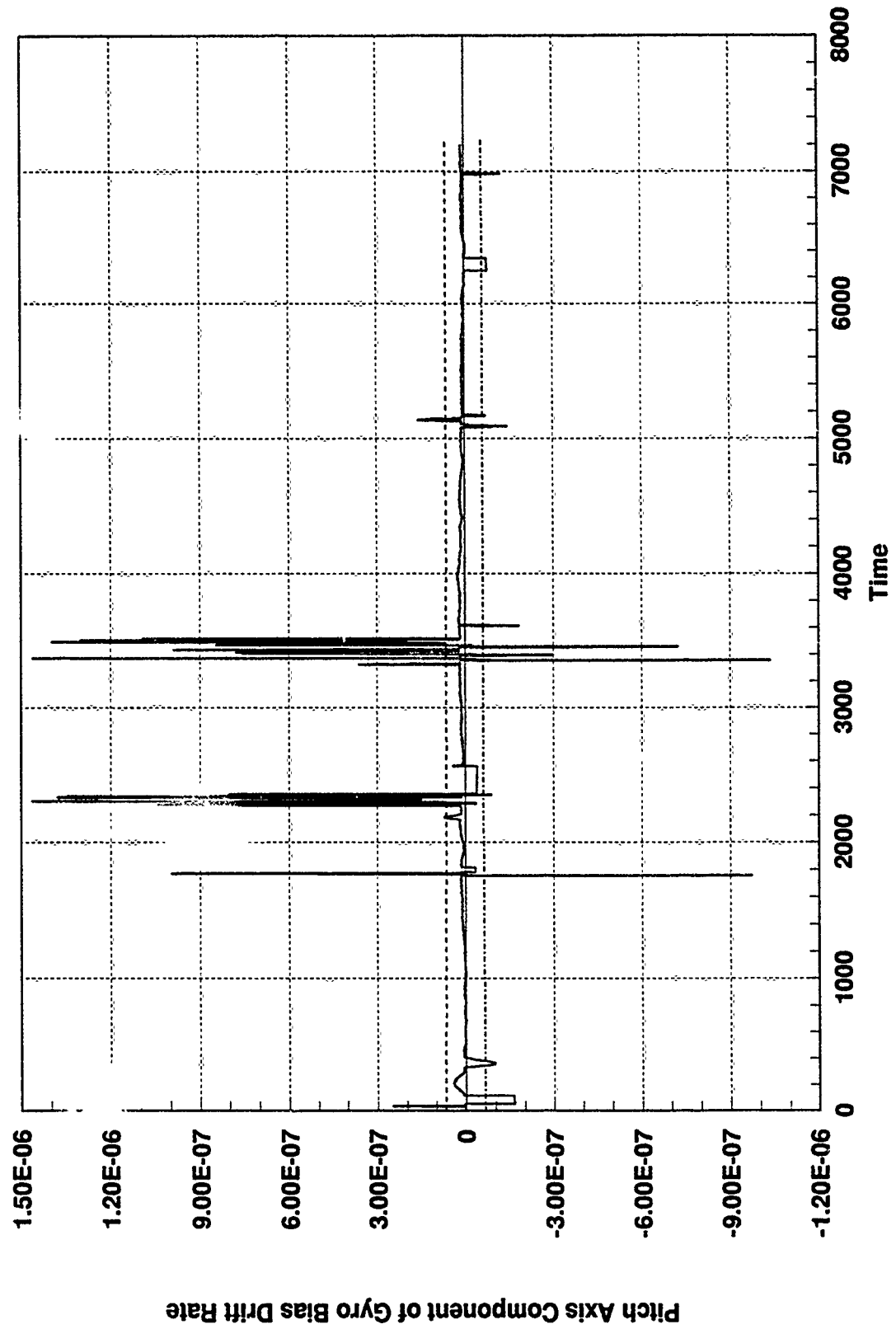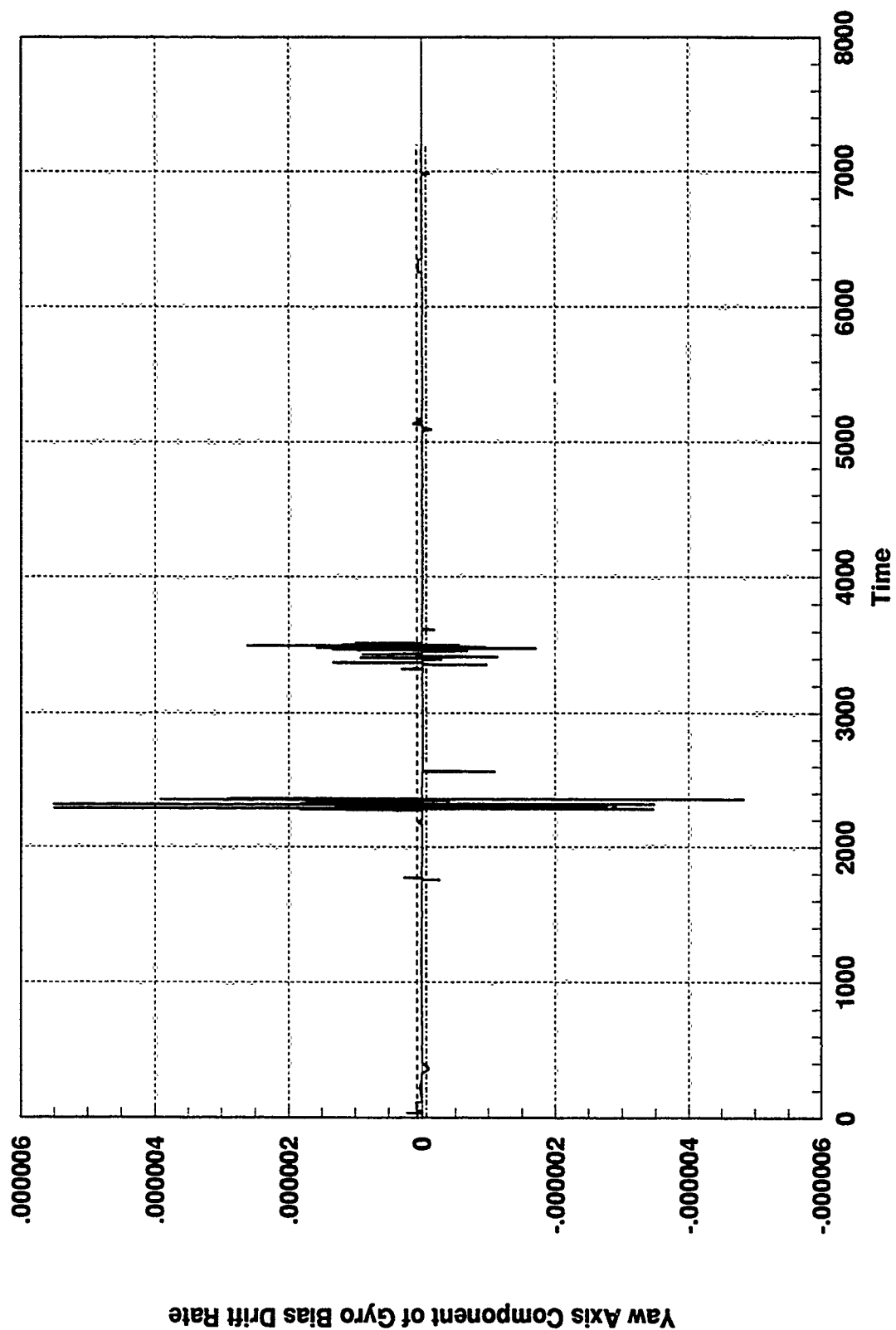
Figure A.16 Centralized Kalman Filter Design, Filter State 16

A-17
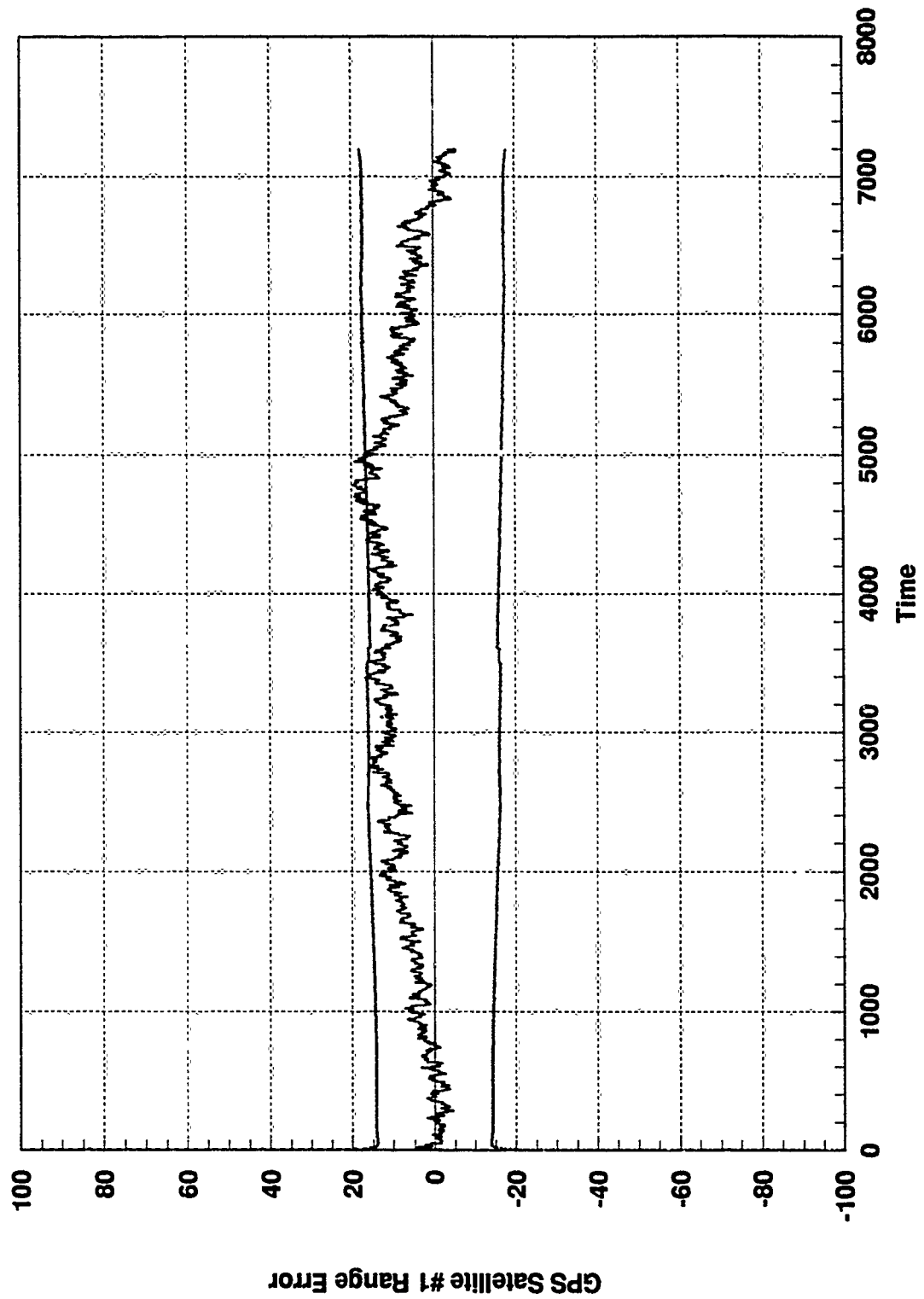
Figure A.17 Centralized Kalman Filter Design, Filter State 17

A-18
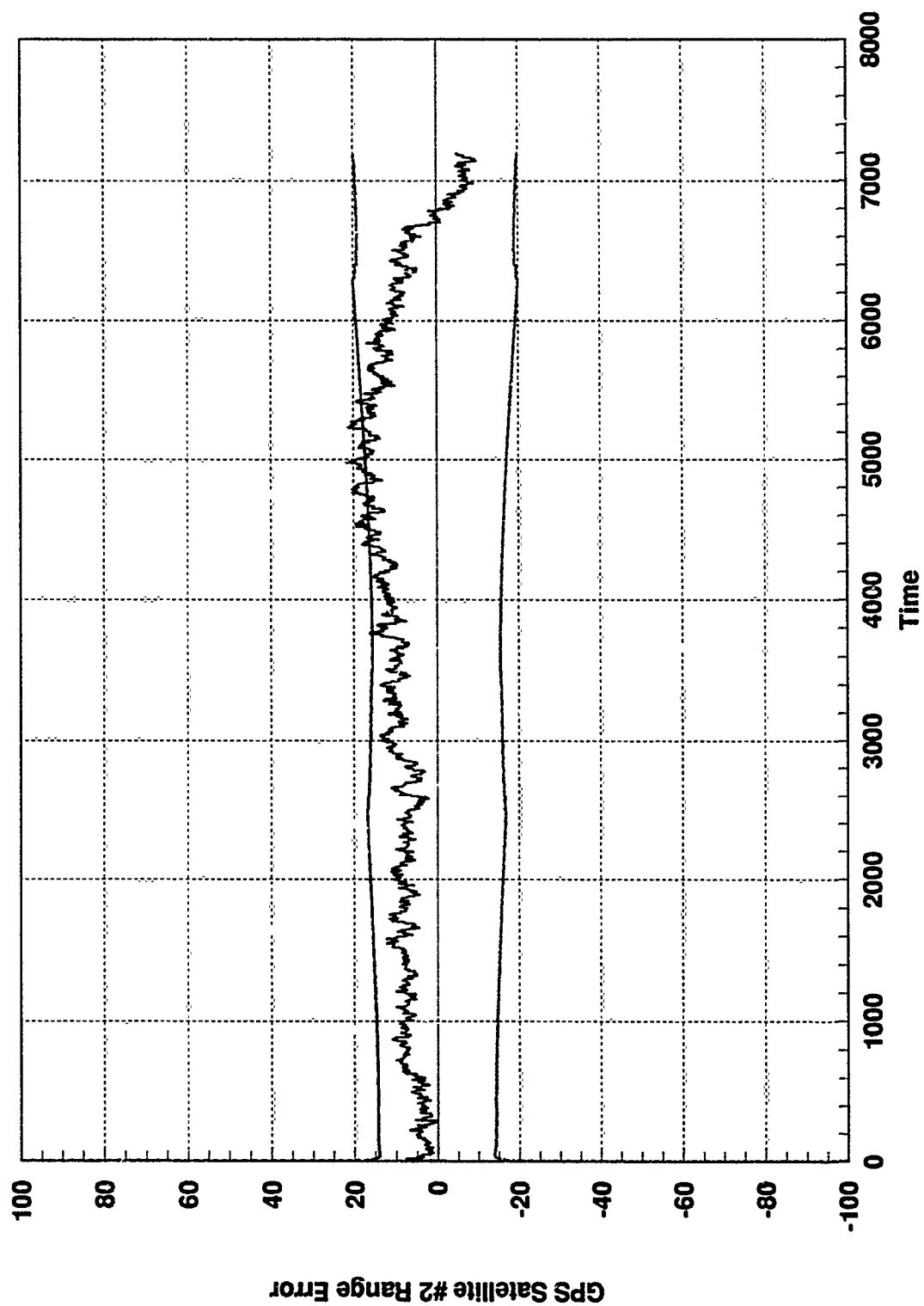
Figure A.18 Centralized Kalman Filter Design, Filter State 18
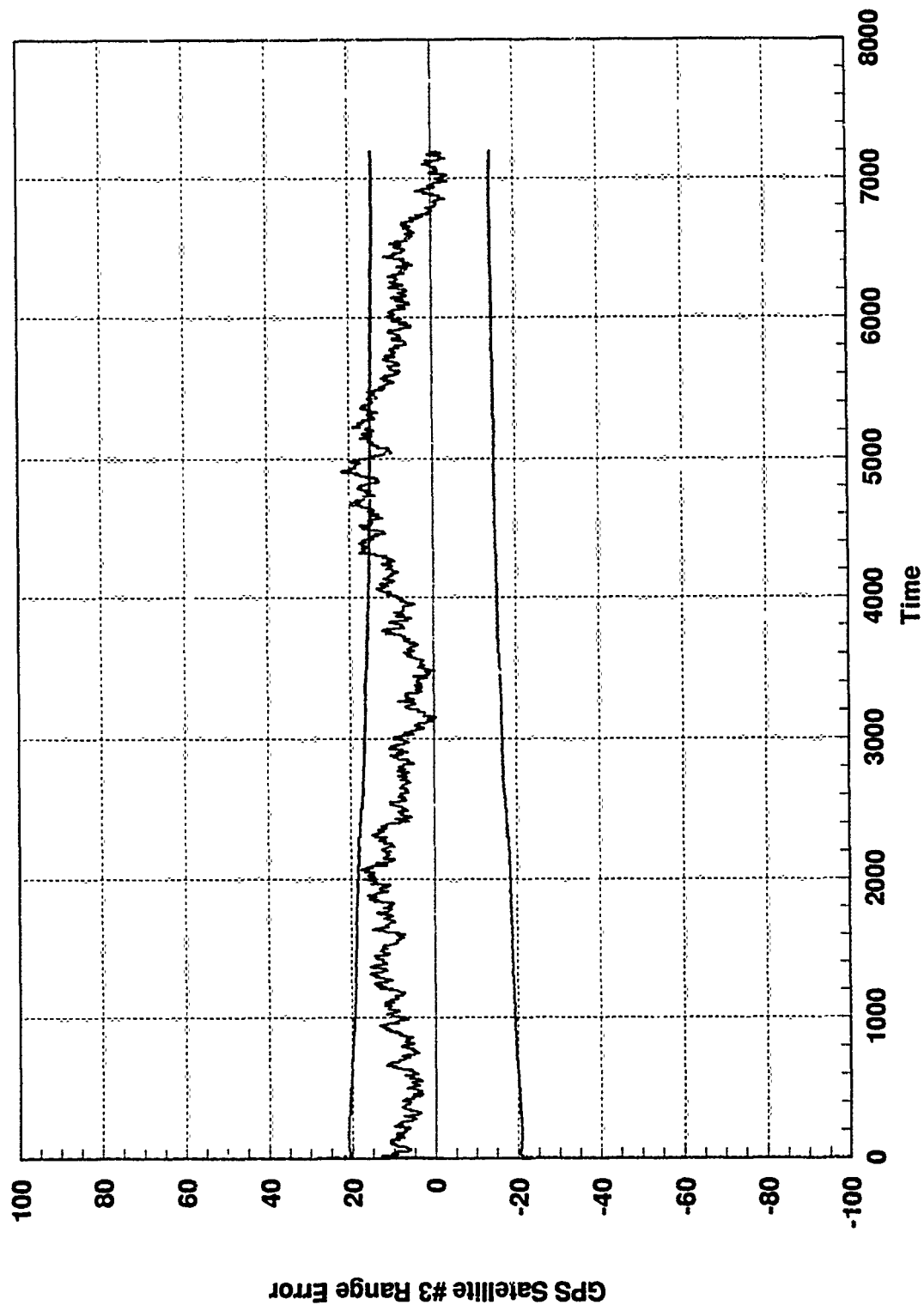
Figure A.19 Centralized Kalman Filter Design, Filter State 19
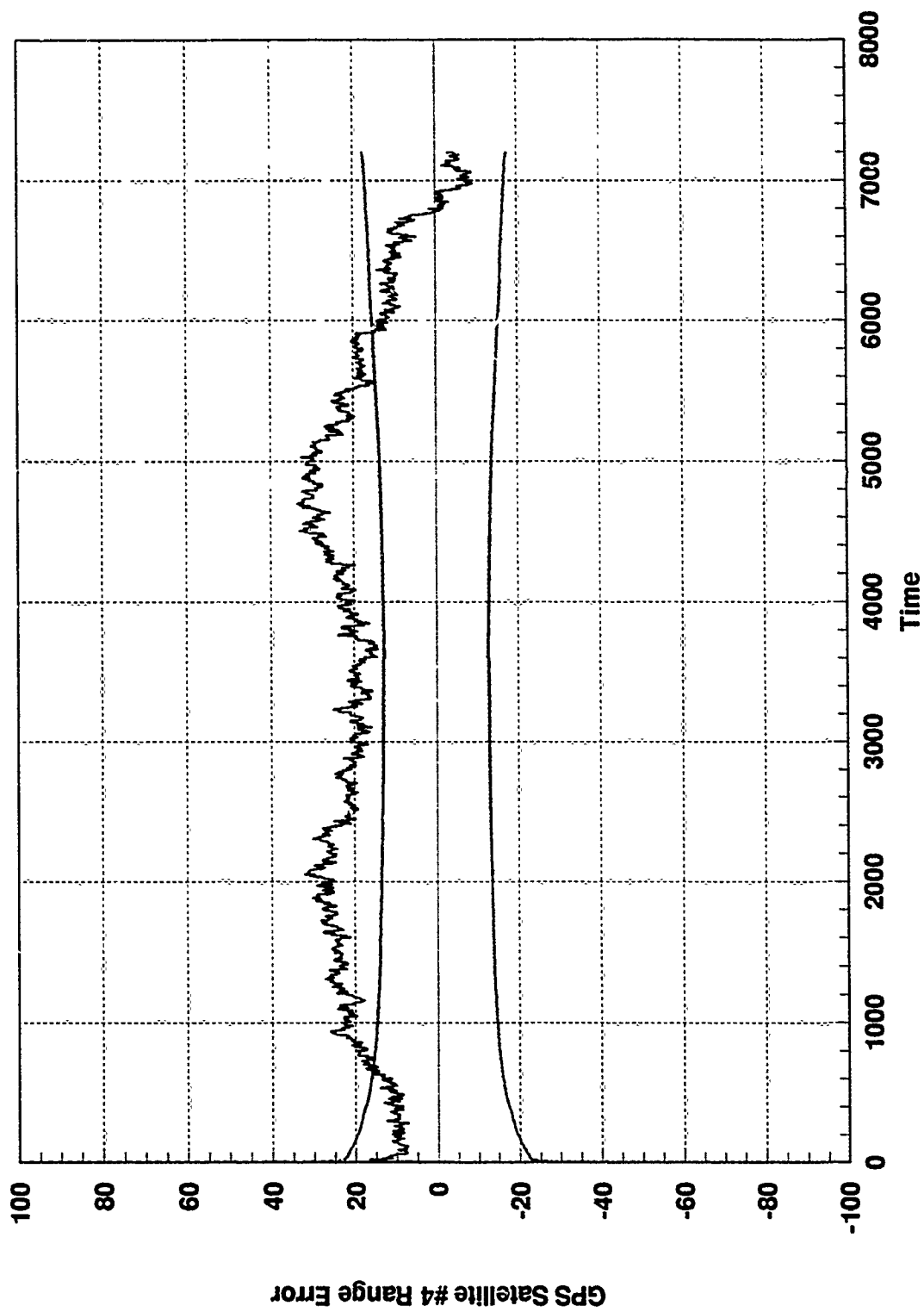
Figure A.20 Centralized Kalman Filter Design, Filter State 20

GPS Satellite #4 Range Error

Time
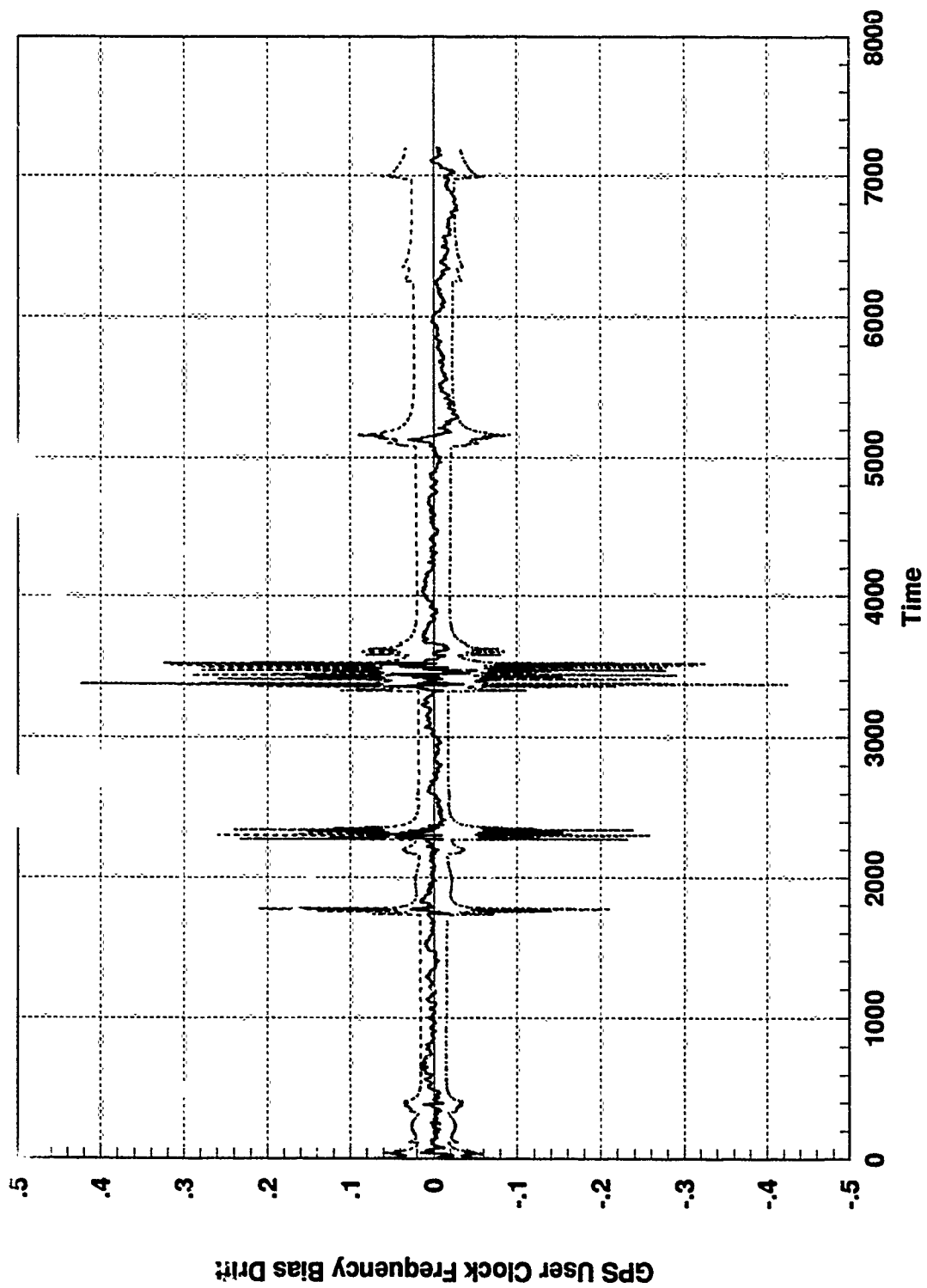
Figure A.21 Centralized Kalman Filter Design, Filter State 21

Y-axis: GPS User Clock Frequency Bias Drift

X-axis: Time
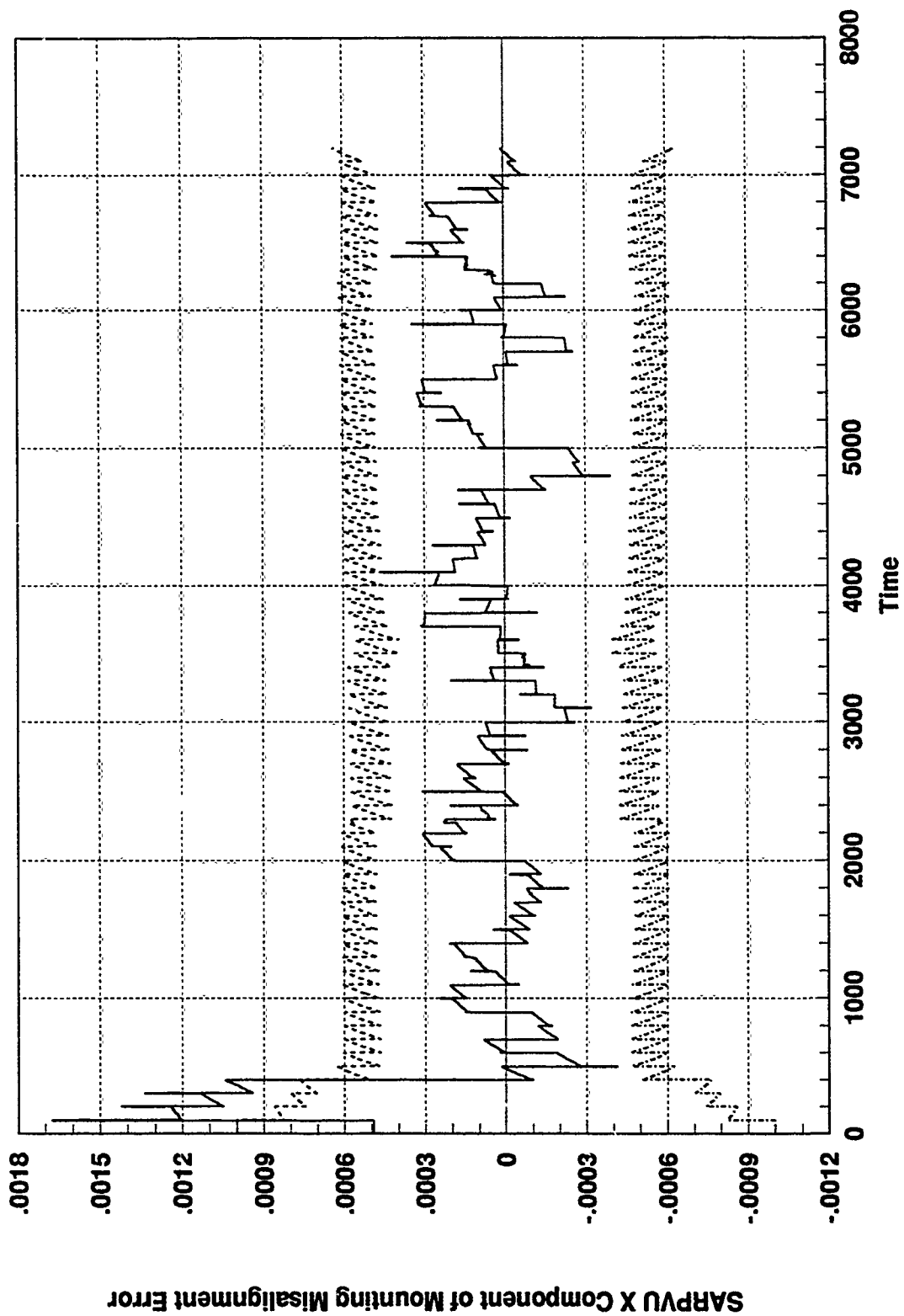
Figure A.22 Centralized Kalman Filter Design, Filter State 22

SARPVU X Component of Mounting Misalignment Error
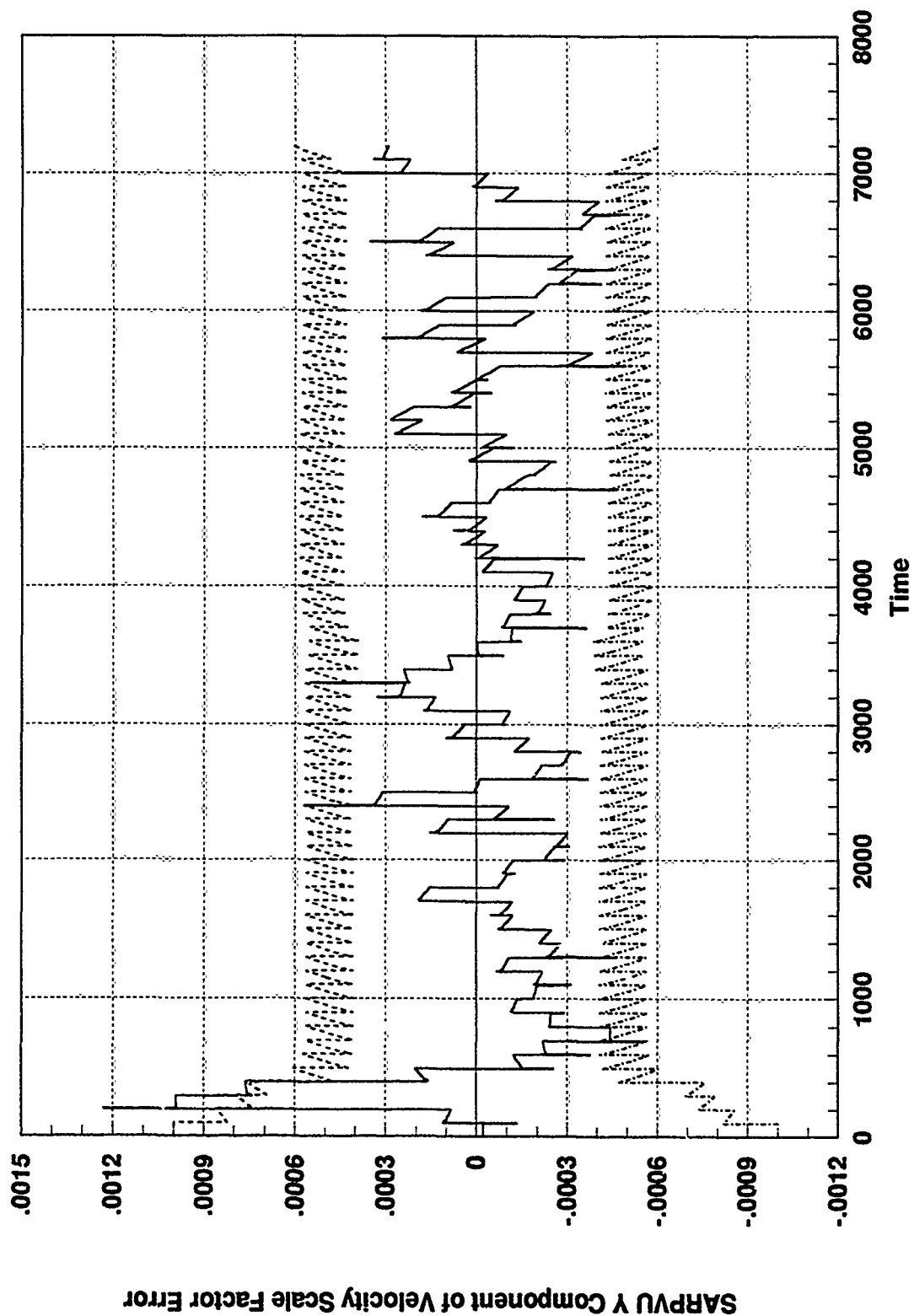
Figure A.23 Centralized Kalman Filter Design, Filter State 23

SARPVU Y Component of Velocity Scale Factor Error
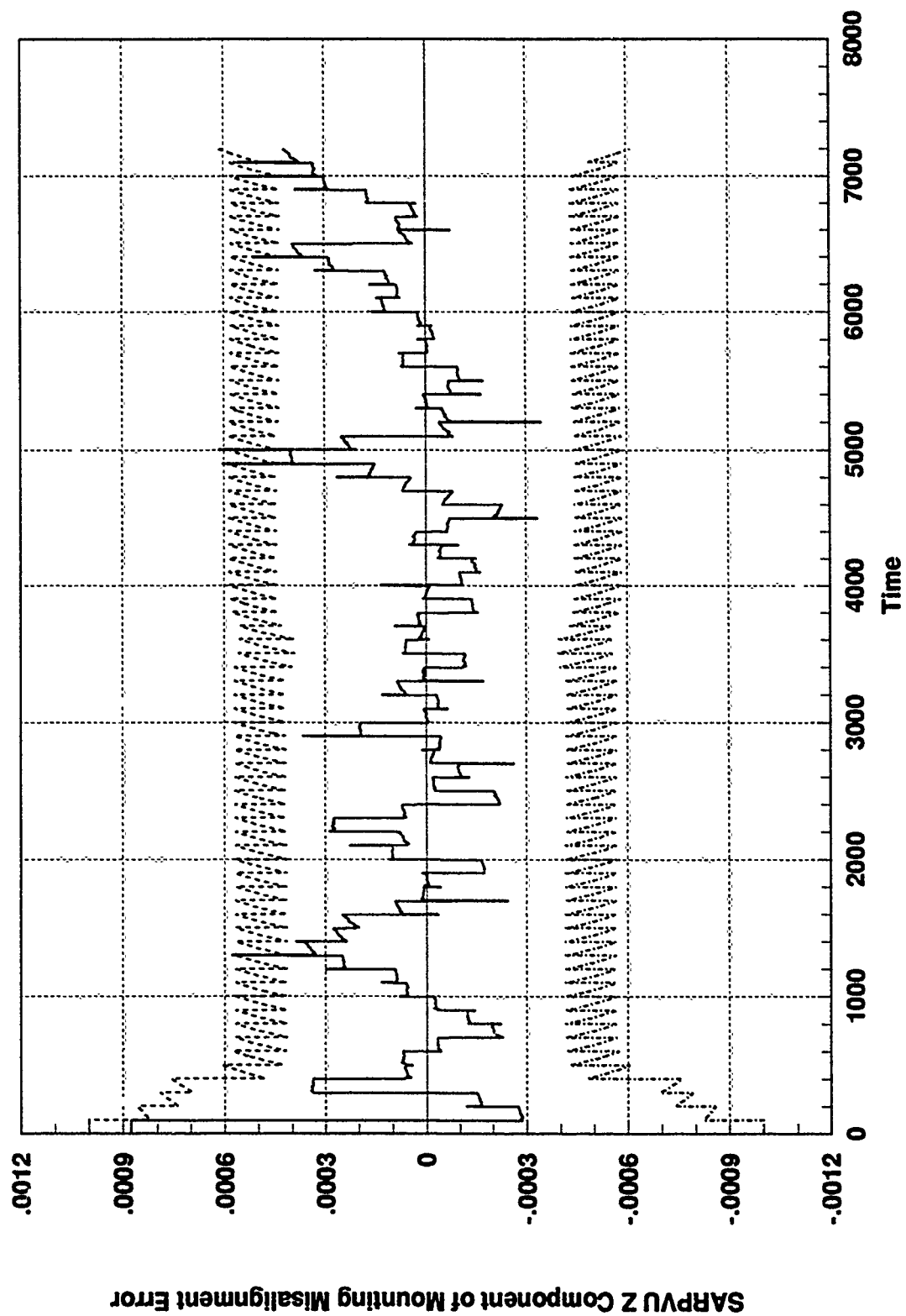
Figure A.24 Centralized Kalman Filter Design, Filter State 24
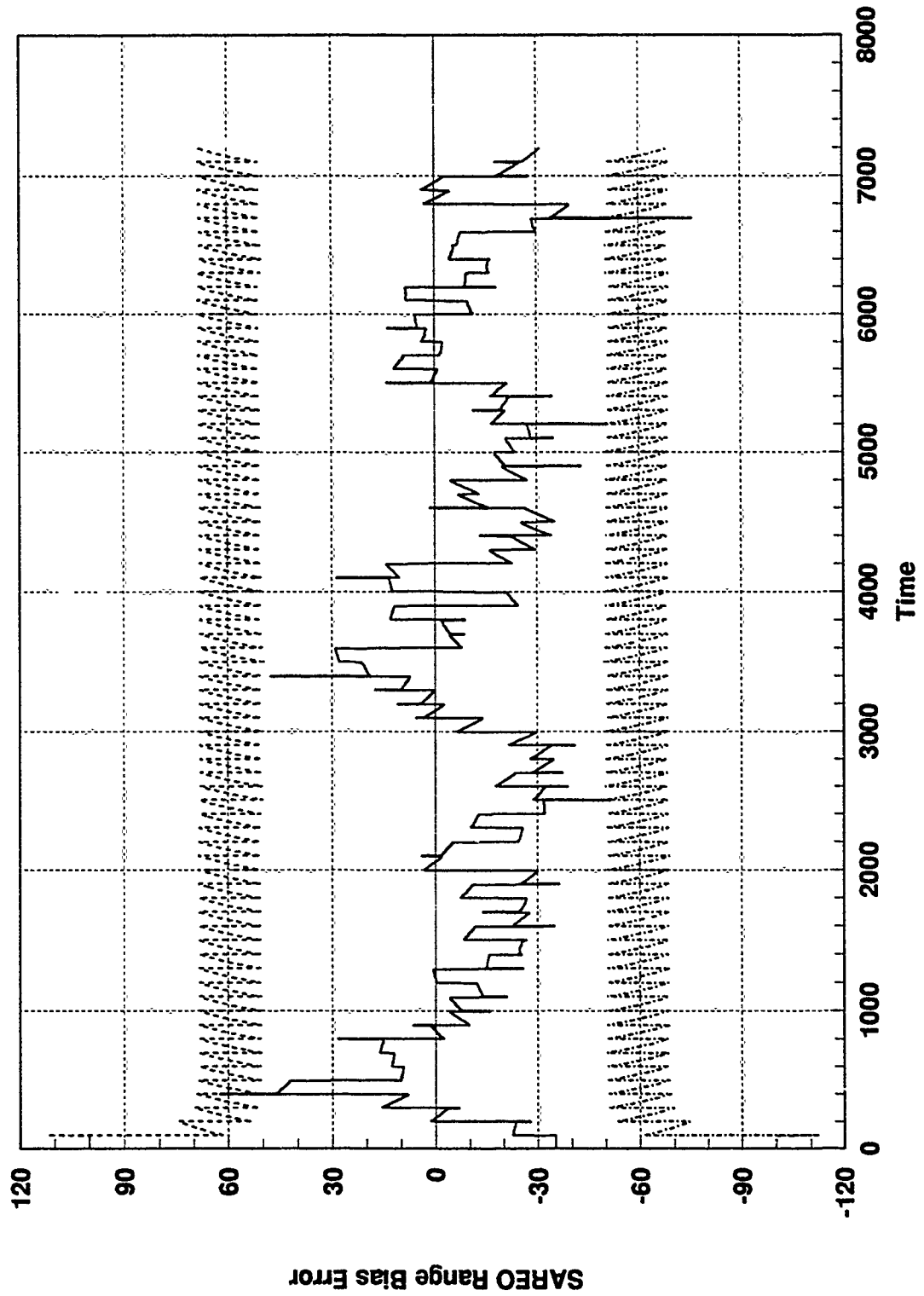
Figure A.25 Centralized Kalman Filter Design, Filter State 25
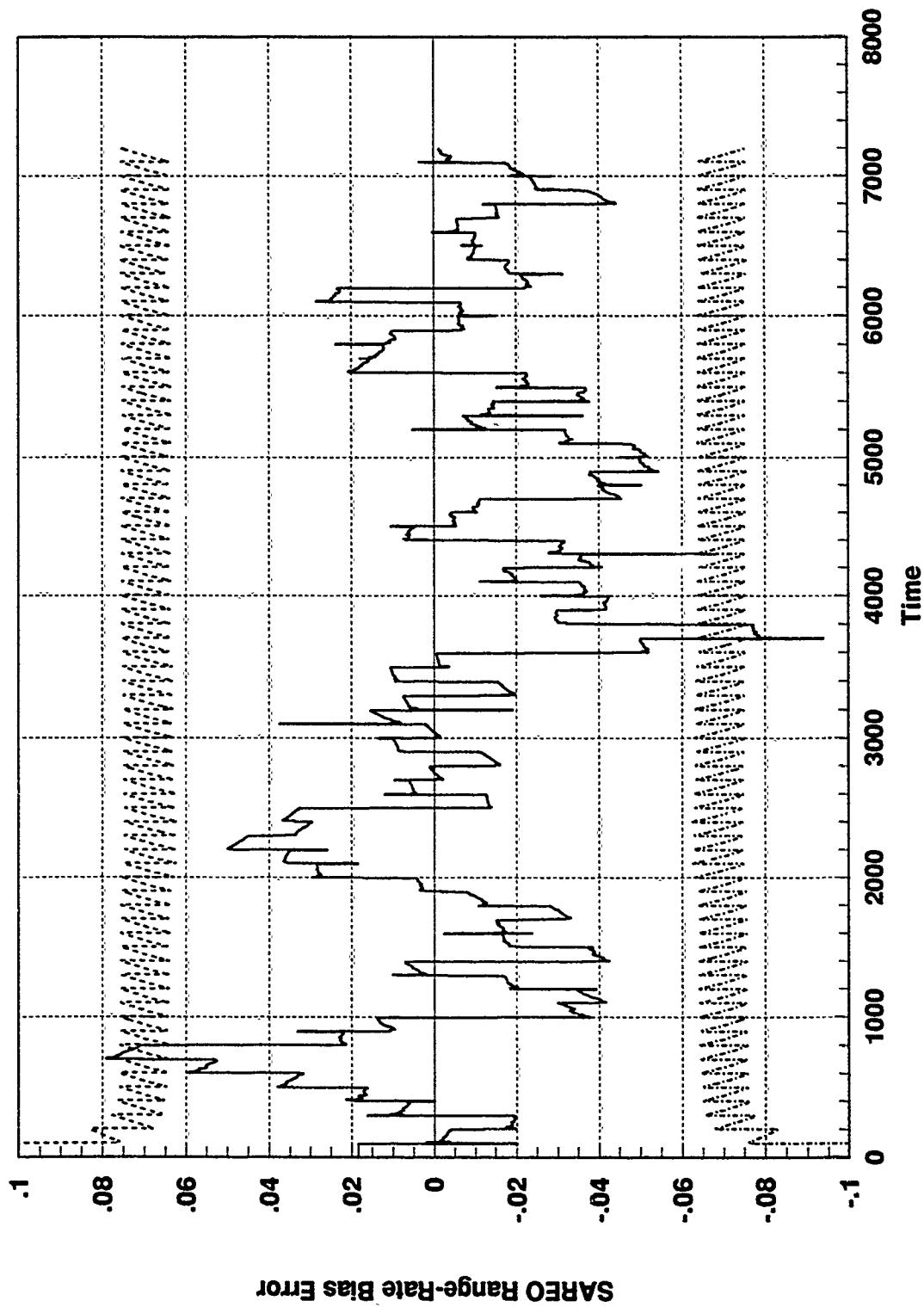
Figure A.26 Centralized Kalman Filter Design, Filter State 26
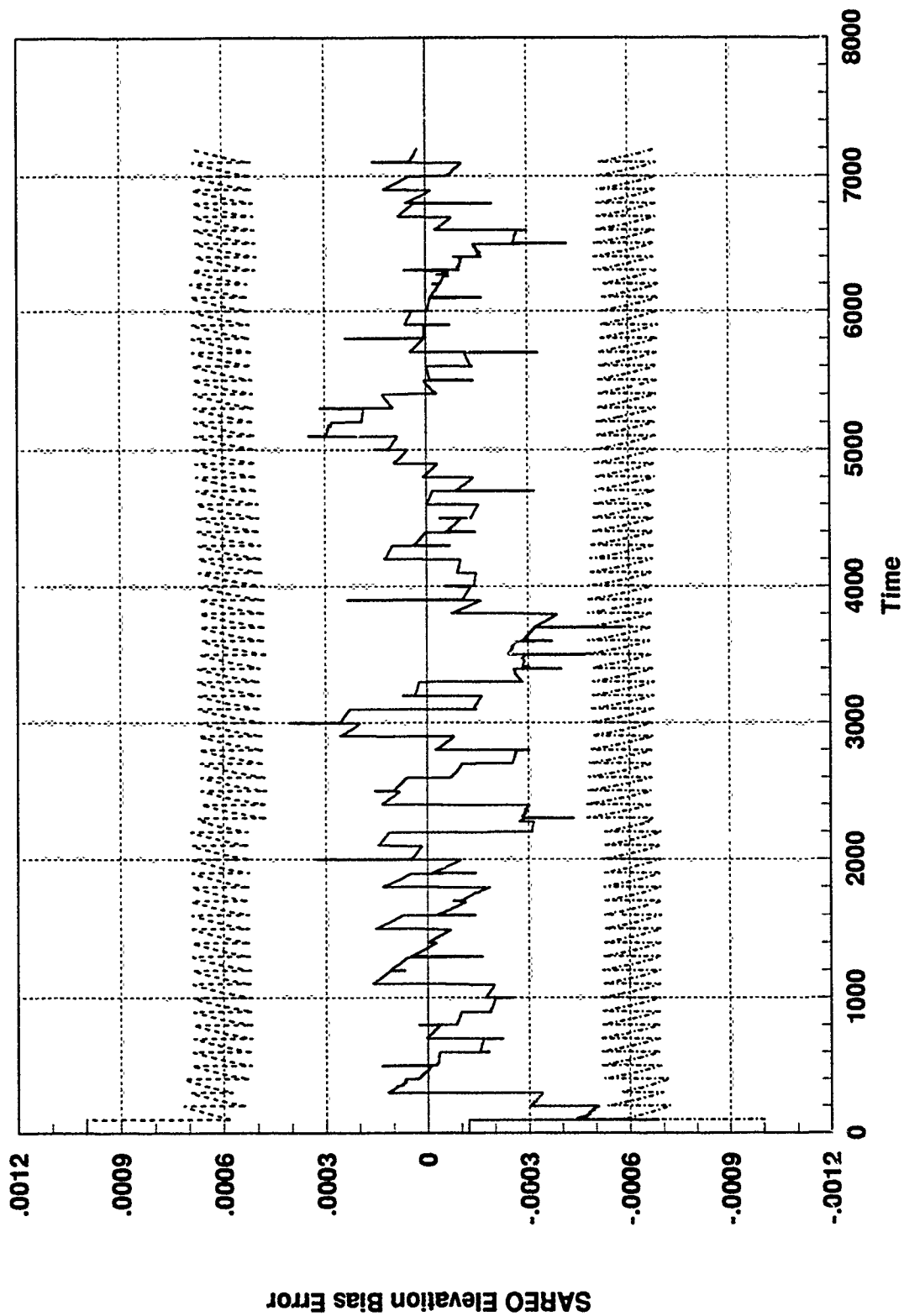
Figure A.27 Centralized Kalman Filter Design, Filter State 27
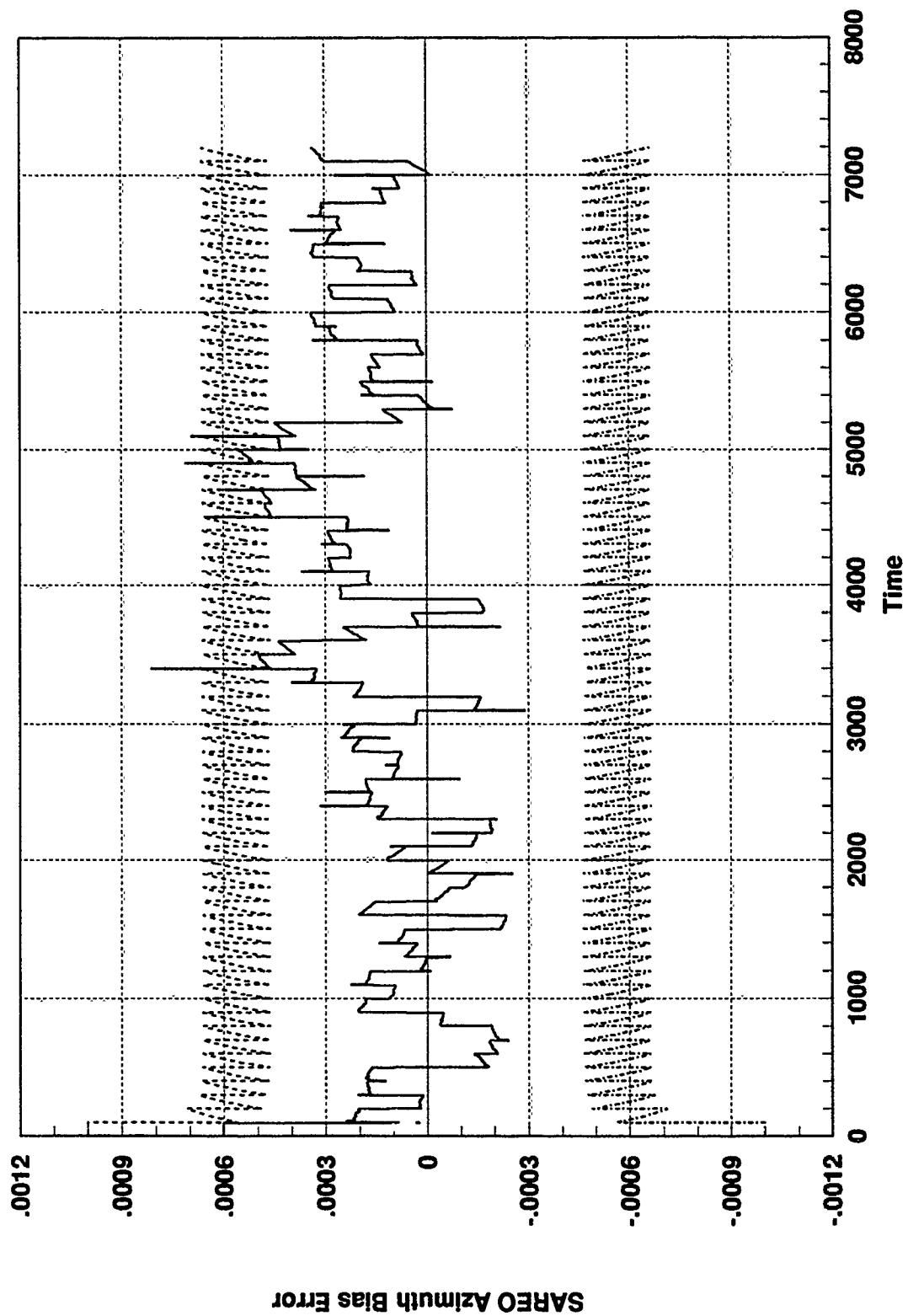
Figure A.28 Centralized Kalman Filter Design, Filter State 28

A-29
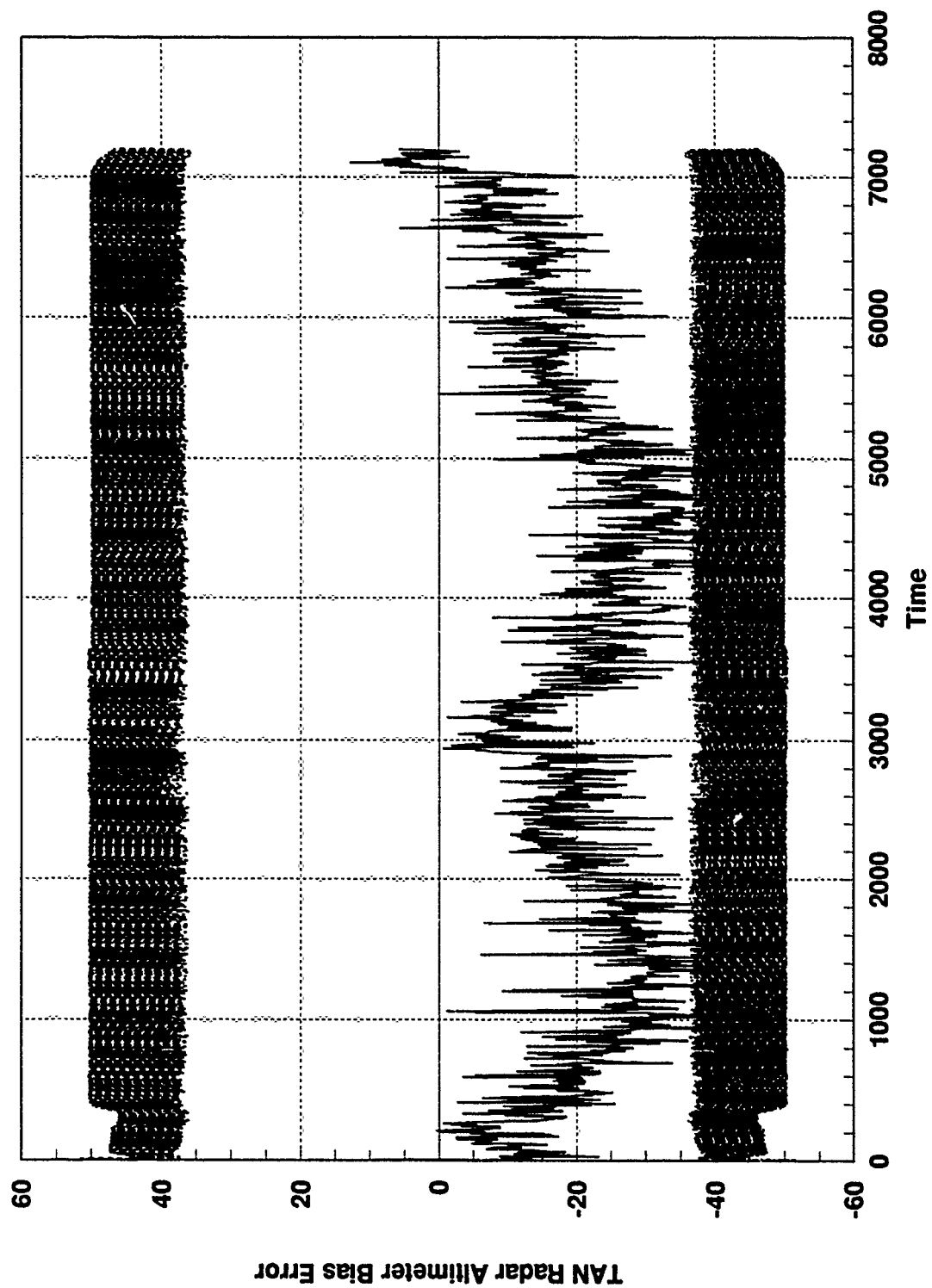
Figure A.29 Centralized Kalman Filter Design, Filter State 29

TAN Radar Altimeter Bias Error

Appendix B: Federated Filter Error State Plots

GENERAL INFORMATION

Each of the error state plots contained in this appendix are obtained from Monte Carlo simulations using DKFSIM Version 1.1. All error state outputs are recorded for the entire 7200 second duration of the simulation. The information provided represents error state outputs at the time of the measurements, after the update cycle, and after the propagation cycle, plotted versus time in increments of five seconds for the federated filter algorithm.

Ten simulations were performed for each state. Therefore, the plots provided represent the ensemble average of the error state output data plus the upper and lower one-sigma bounds computed for that state. The one-sigma bound was computed only for the first Monte Carlo simulation.

There are 29 plots, 16 for each master filter error state, plus the error states associated with the sensor dedicated local filter. Each plot has the state and filter designation along with its corresponding error definition. A listing of the titles of each these figures is found in the List of Figures at the beginning of this document.

KEY TERMS

1. ECEF: Earth-Centered-Earth-Fixed coordinate frame.
2. Yaw, Pitch, and Roll Axes: Body Frame with respect to the aircraft.
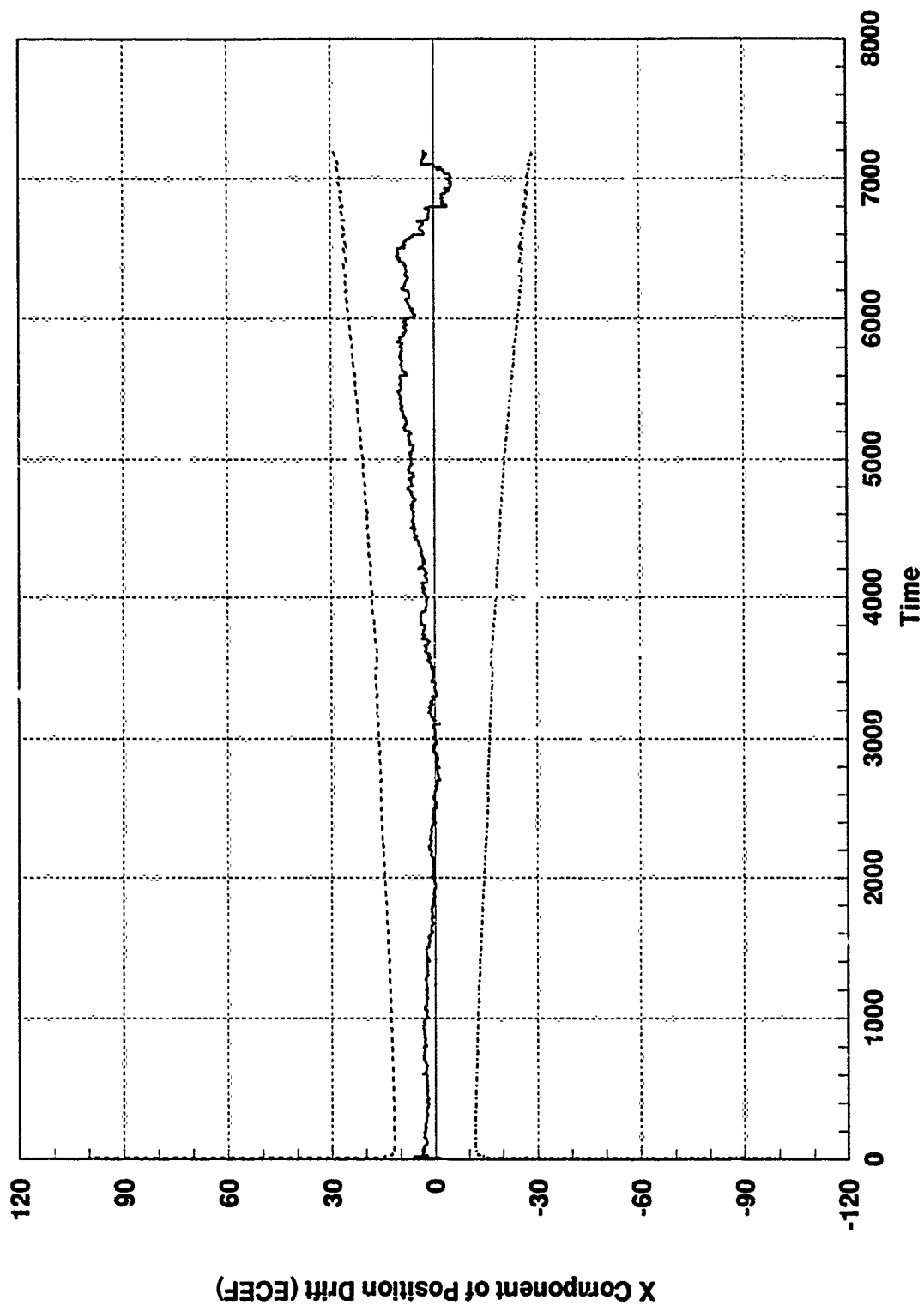
Figure B.1 Federated Filter Design, Master Filter State 1
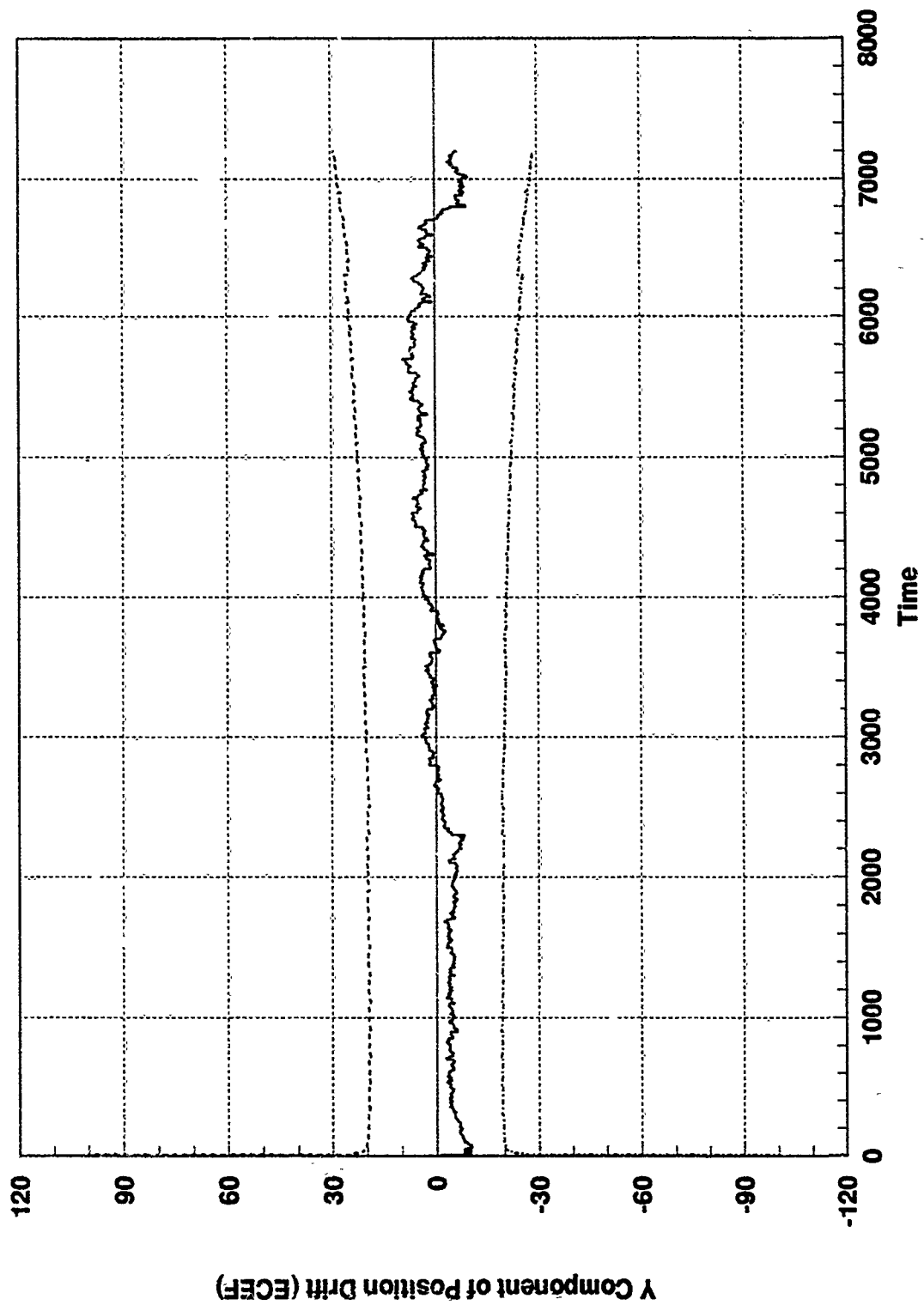
Figure B.2 Federated Filter Design, Master Filter State 2
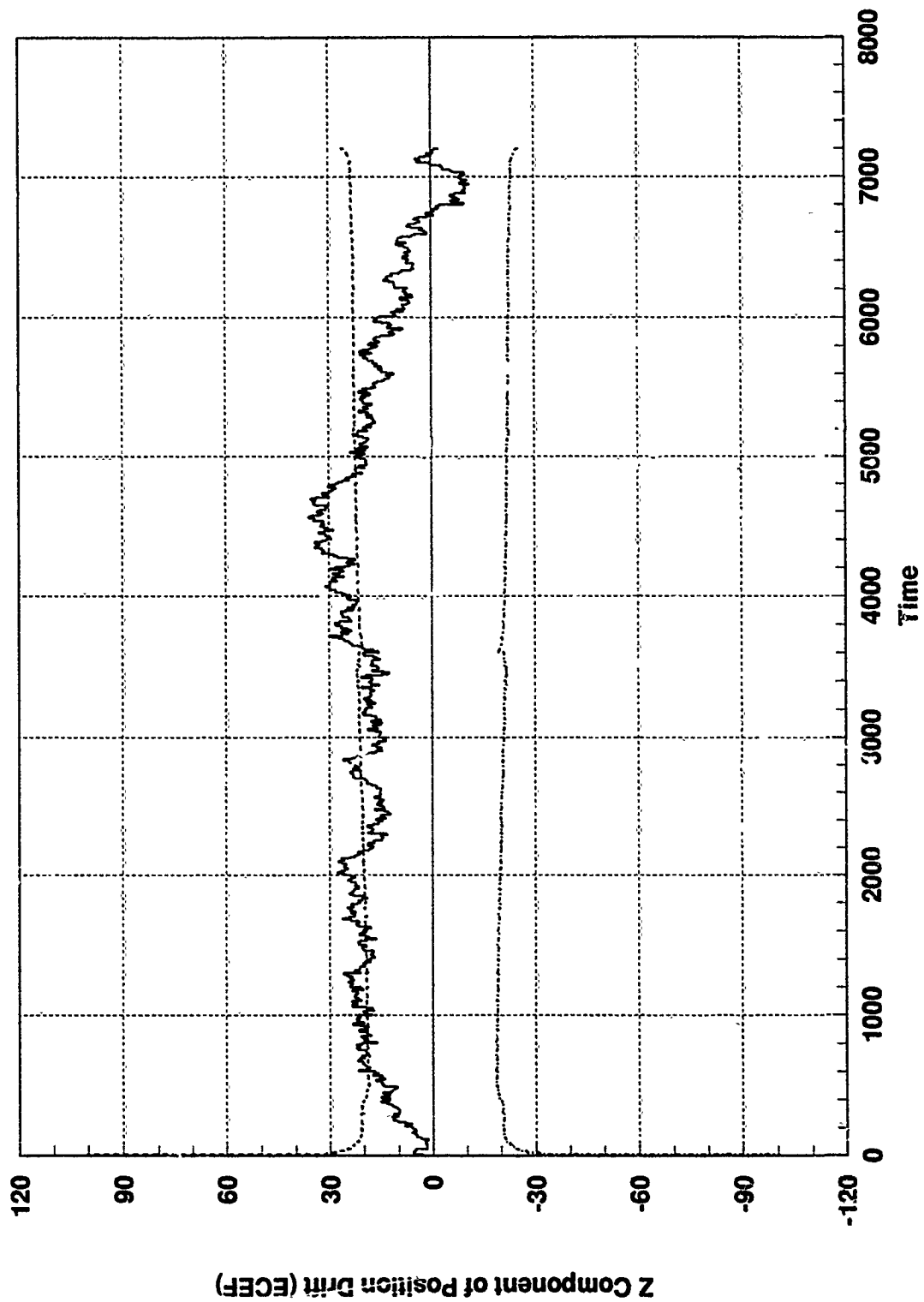
Figure B.3 Federated Filter Design, Master Filter State 3
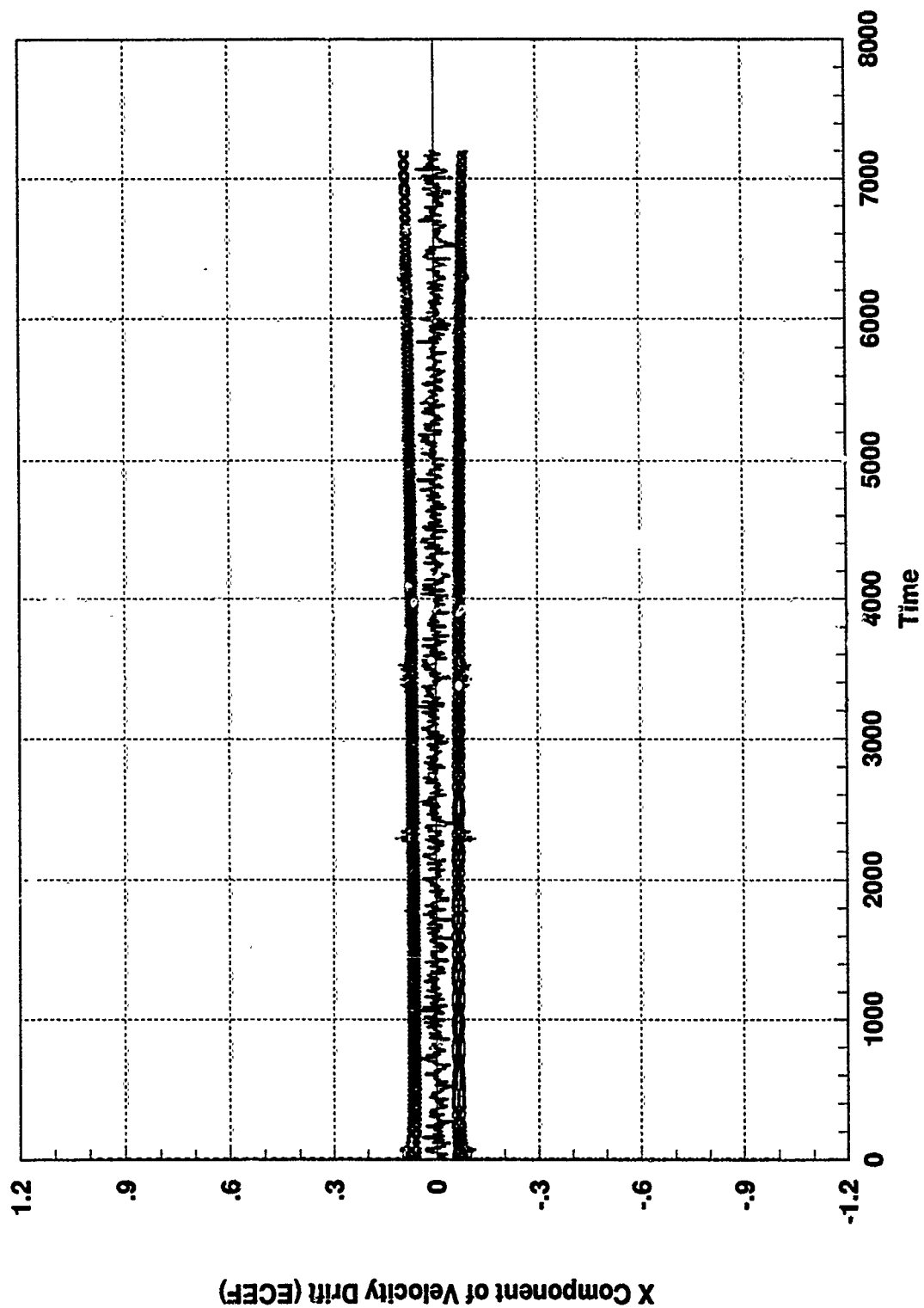
Figure B.4 Federated Filter Design, Master Filter State 4
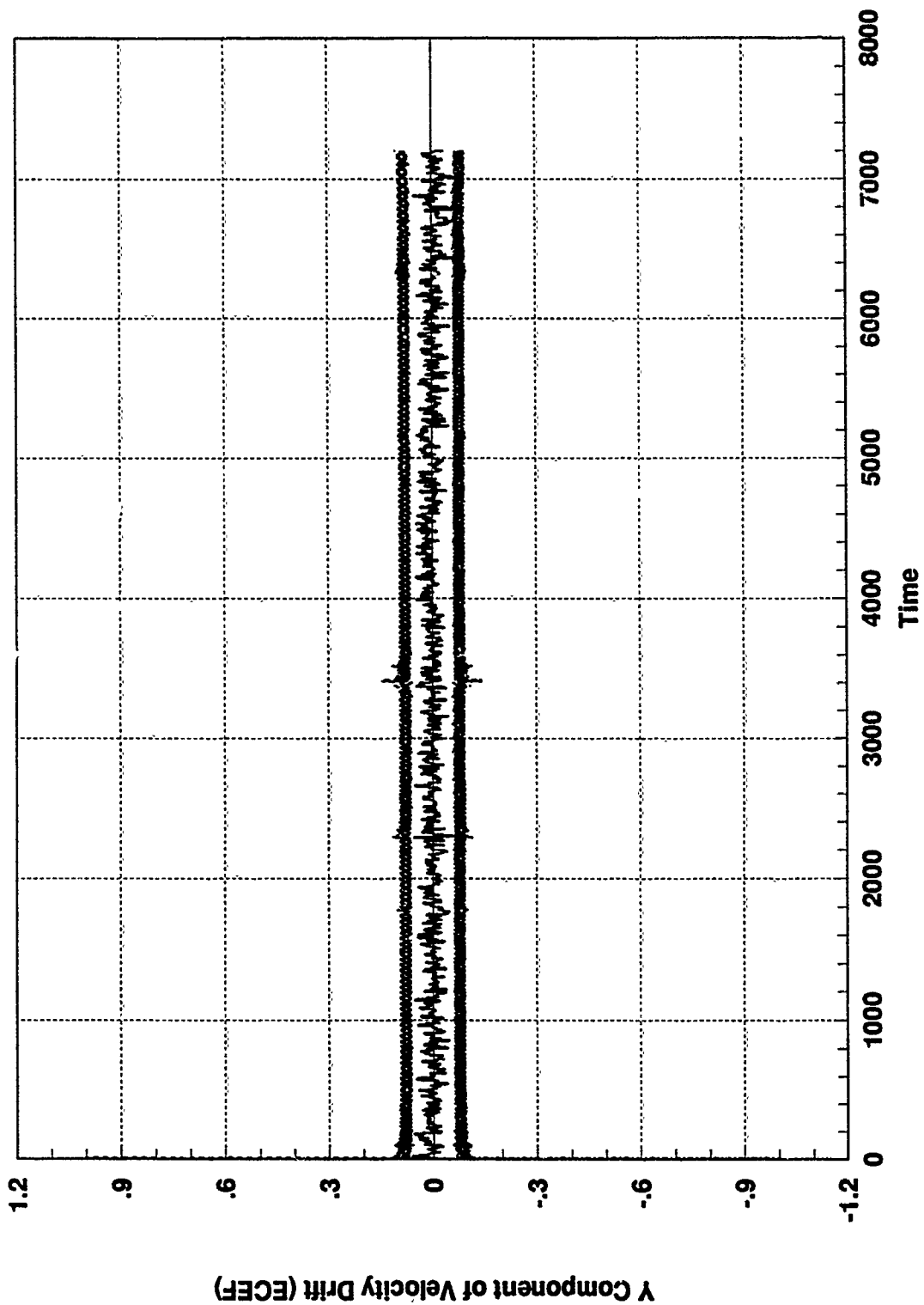
Figure B.5 Federated Filter Design, Master Filter State 5
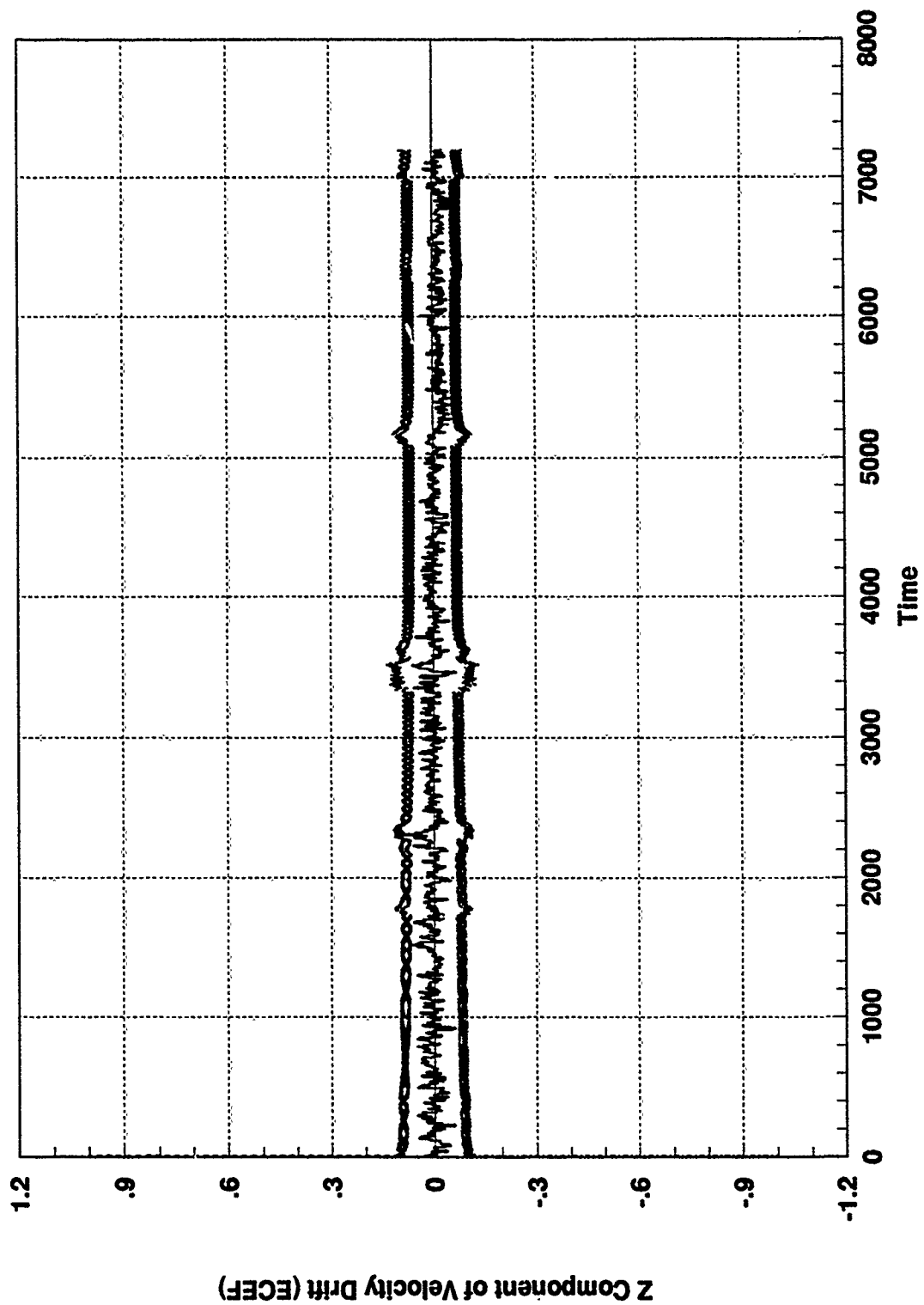
Figure B.6 Federated Filter Design, Master Filter State 6
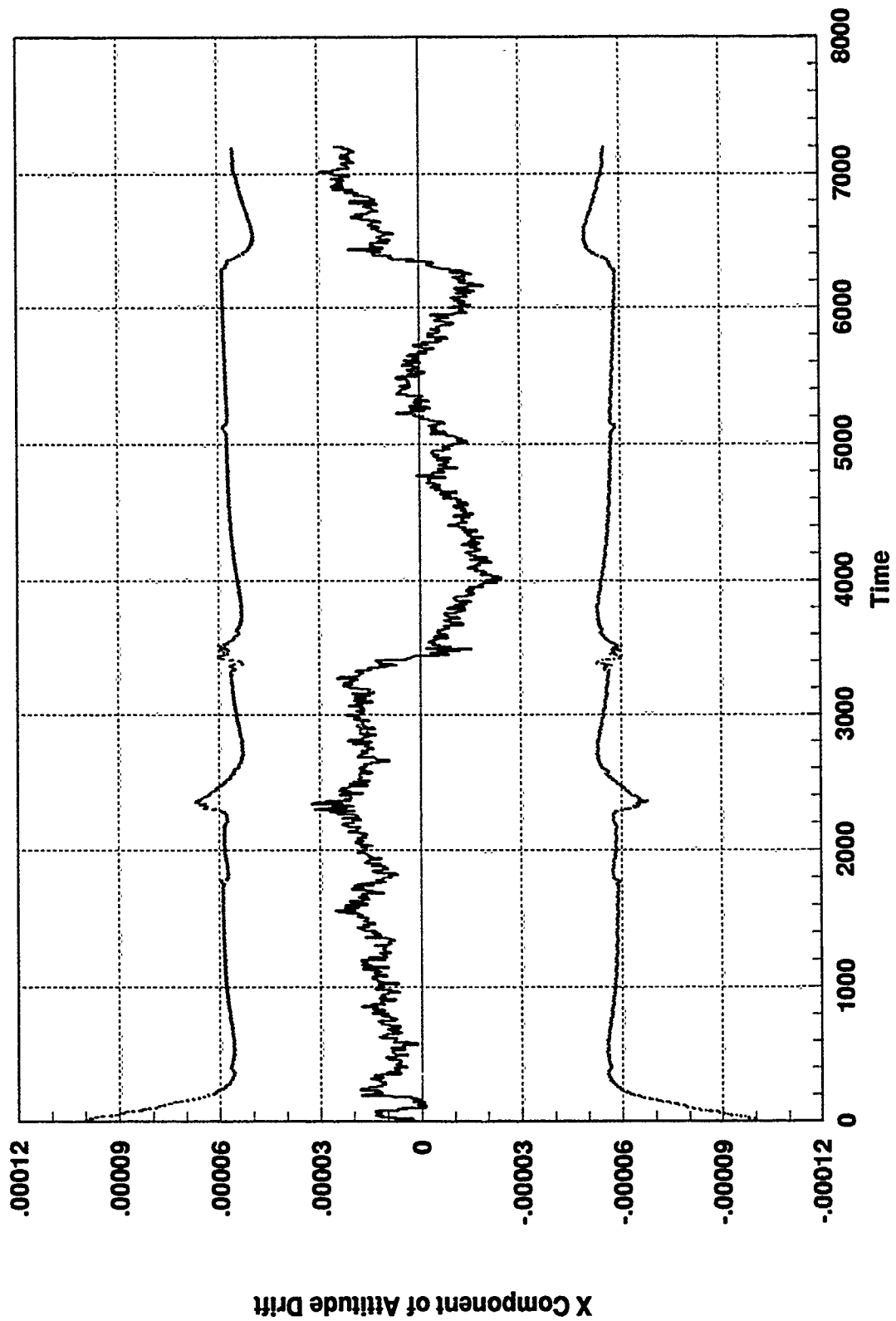
Figure B.7 Federated Filter Design, Master Filter State 7

X Component of Attitude Drift
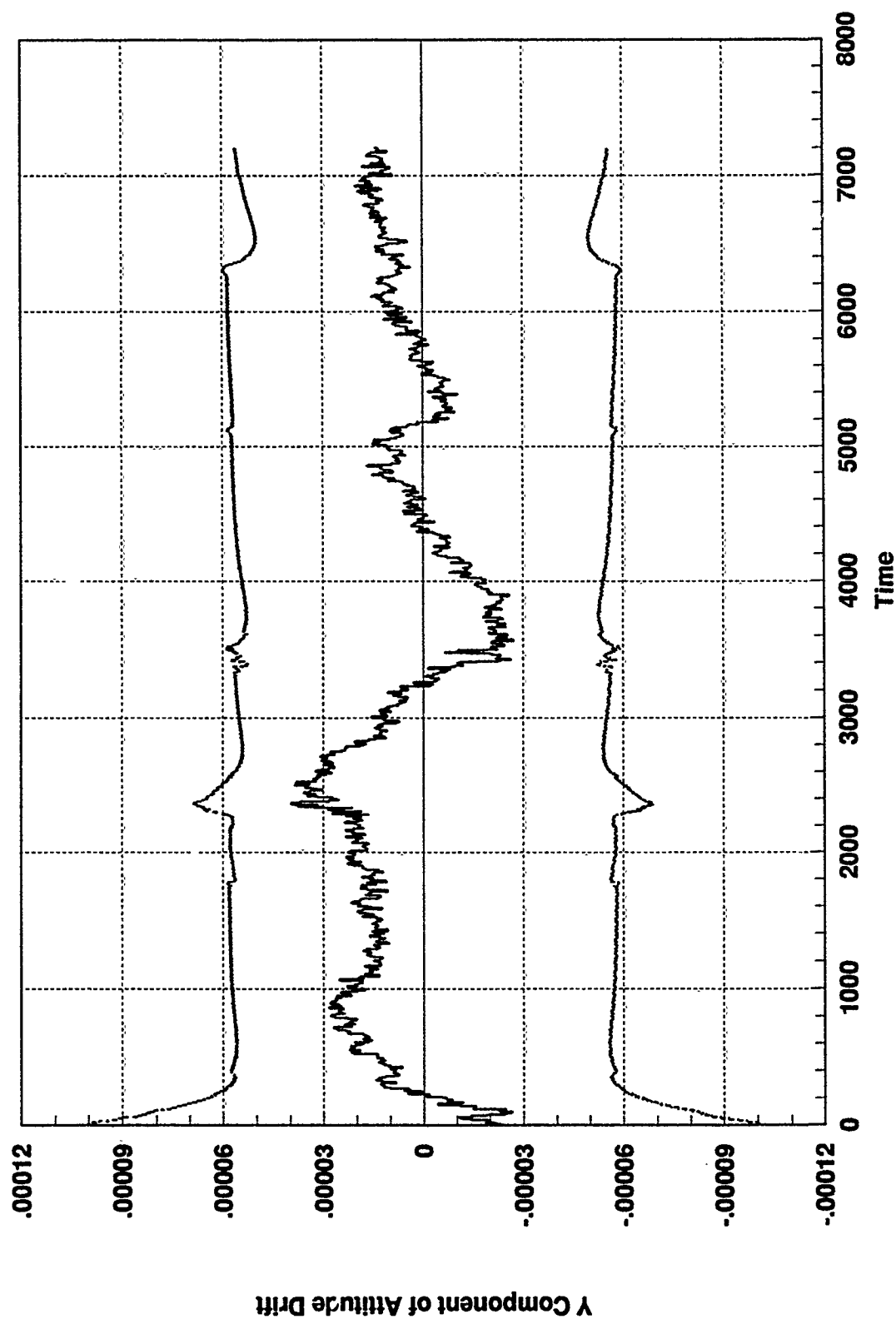
Figure B.8 Federated Filter Design, Master Filter State 8

Y Component of Attitude Drift
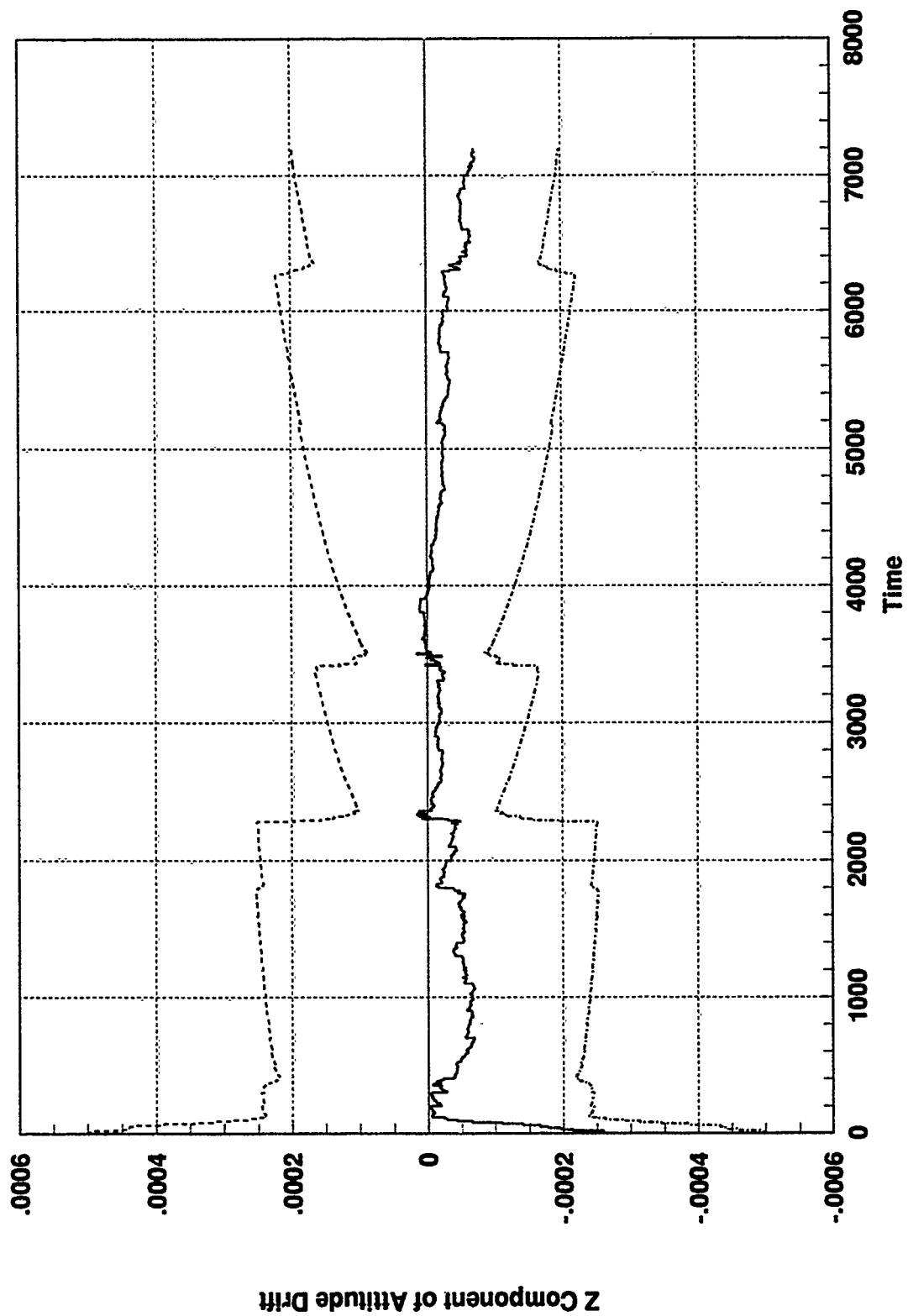
Figure B.9 Federated Filter Design, Master Filter State 9

Z Component of Attitude Drift
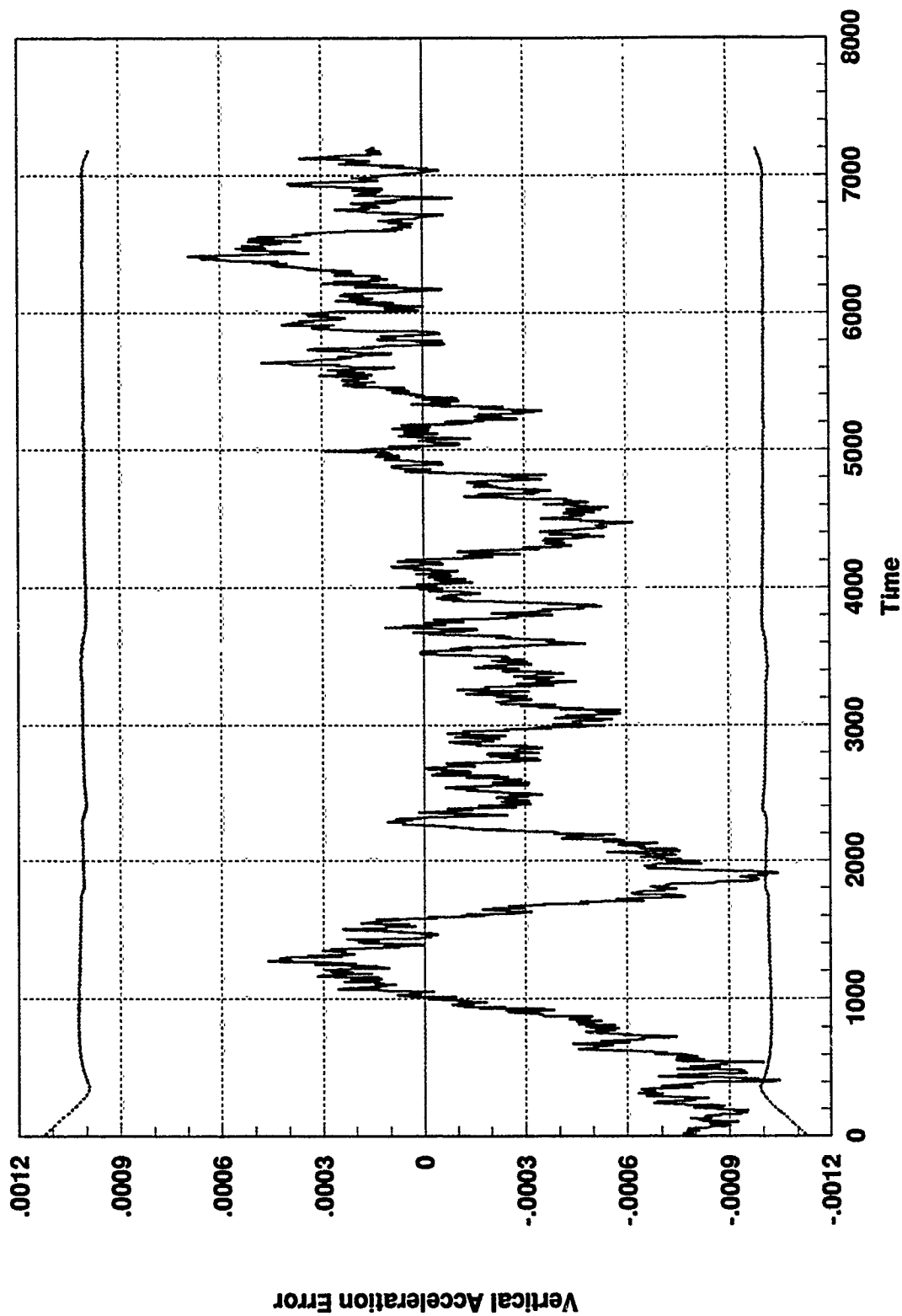
Figure B.10 Federated Filter Design, Master Filter State 10
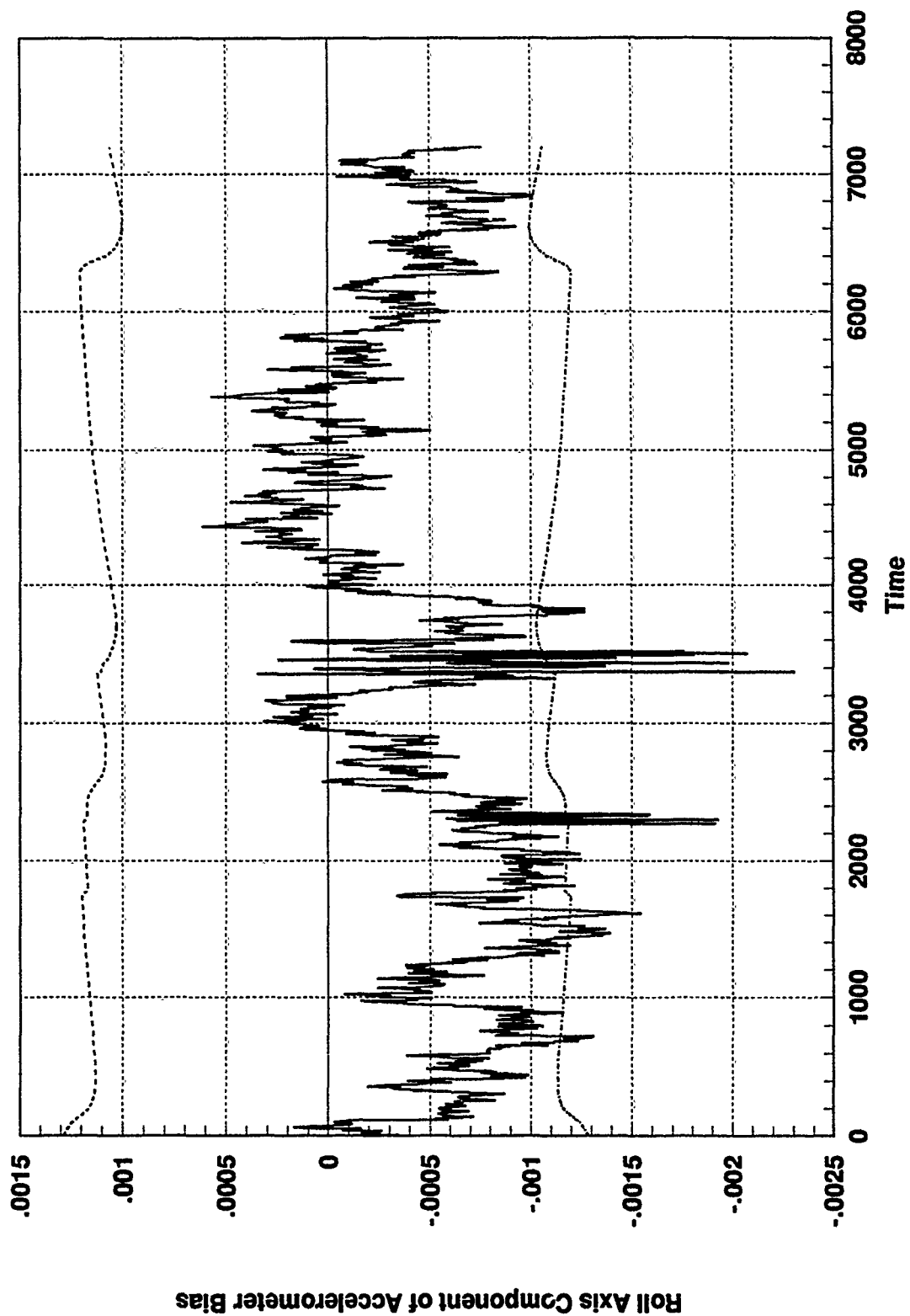
Figure B.11 Federated Filter Design, Master Filter State 11
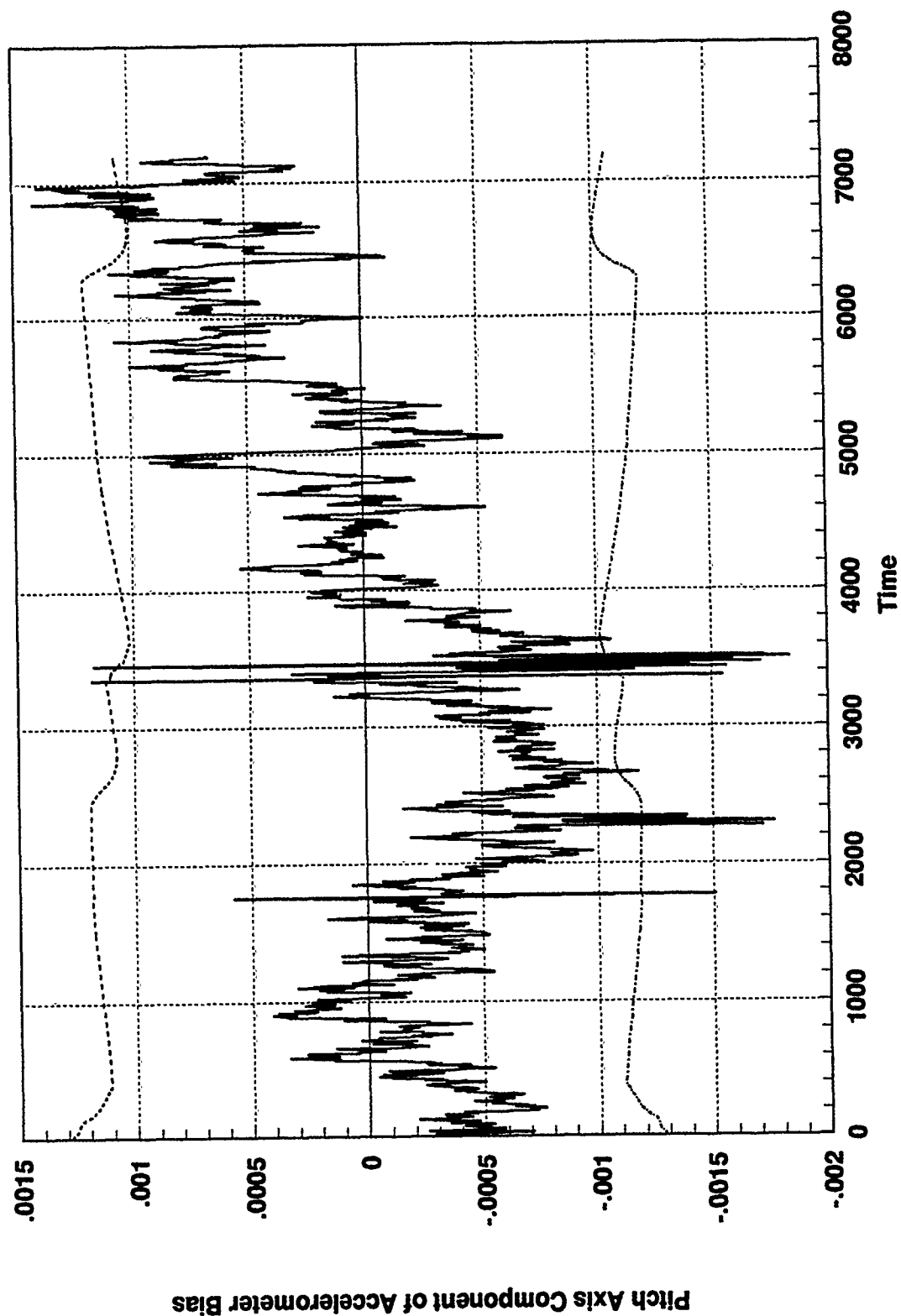
Figure B.12 Federated Filter Design, Master Filter State 12

Pitch Axis Component of Accelerometer Bias
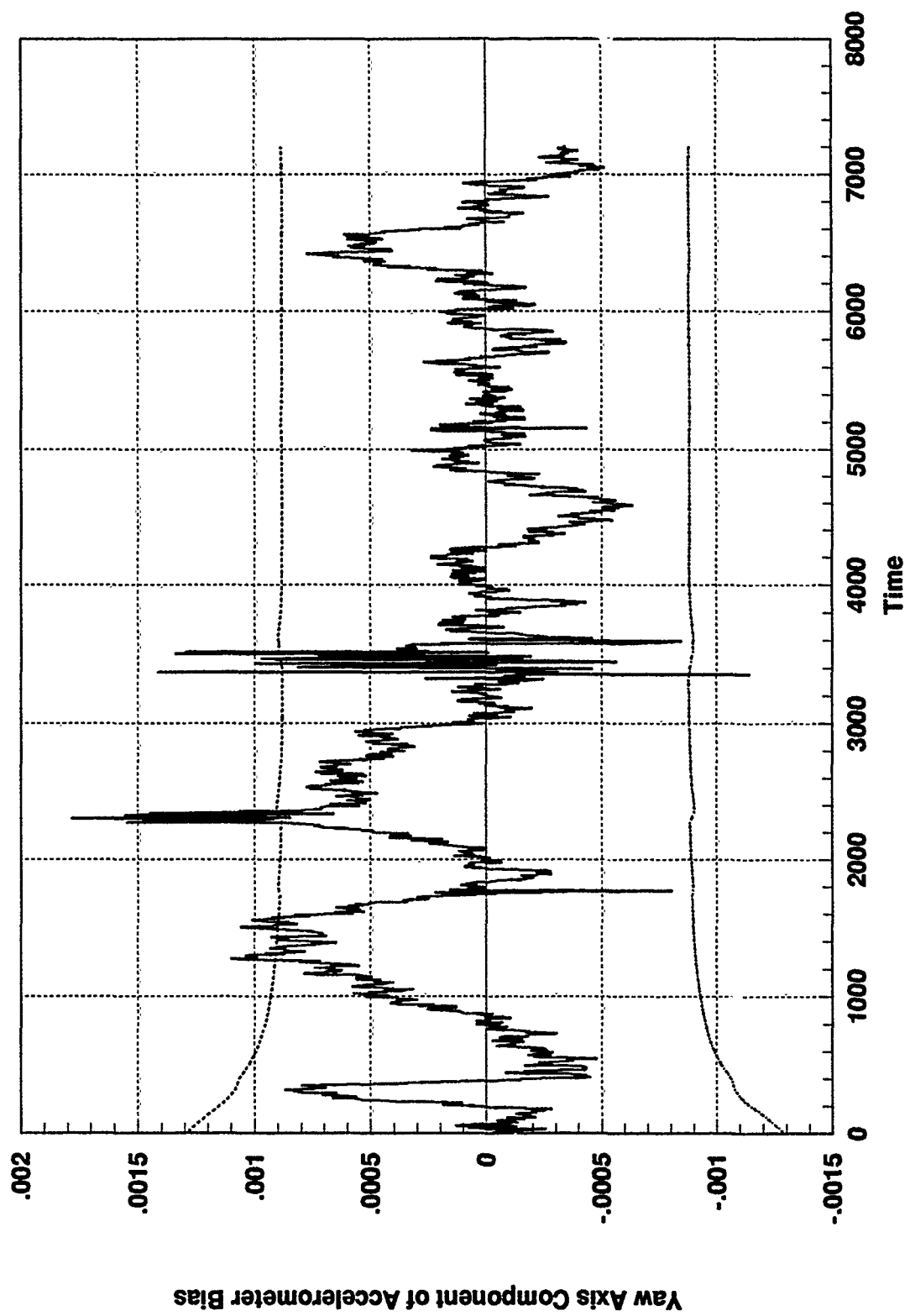
Figure B.13 Federated Filter Design, Master Filter State 13

Yaw Axis Component of Accelerometer Bias
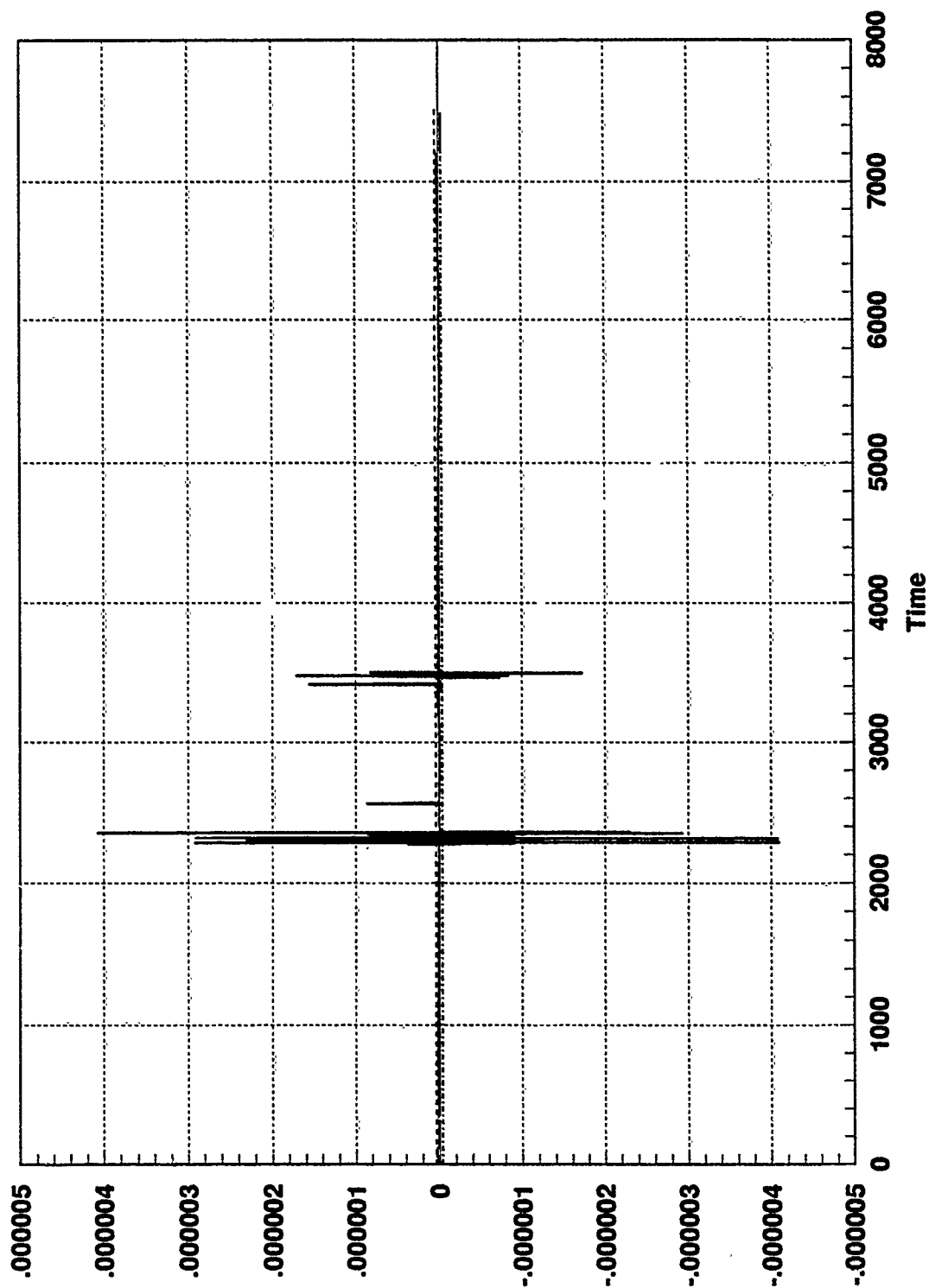
Figure B.14 Federated Filter Design, Master Filter State 14

Roll Axis Component of Gyro Bias Drift Rate
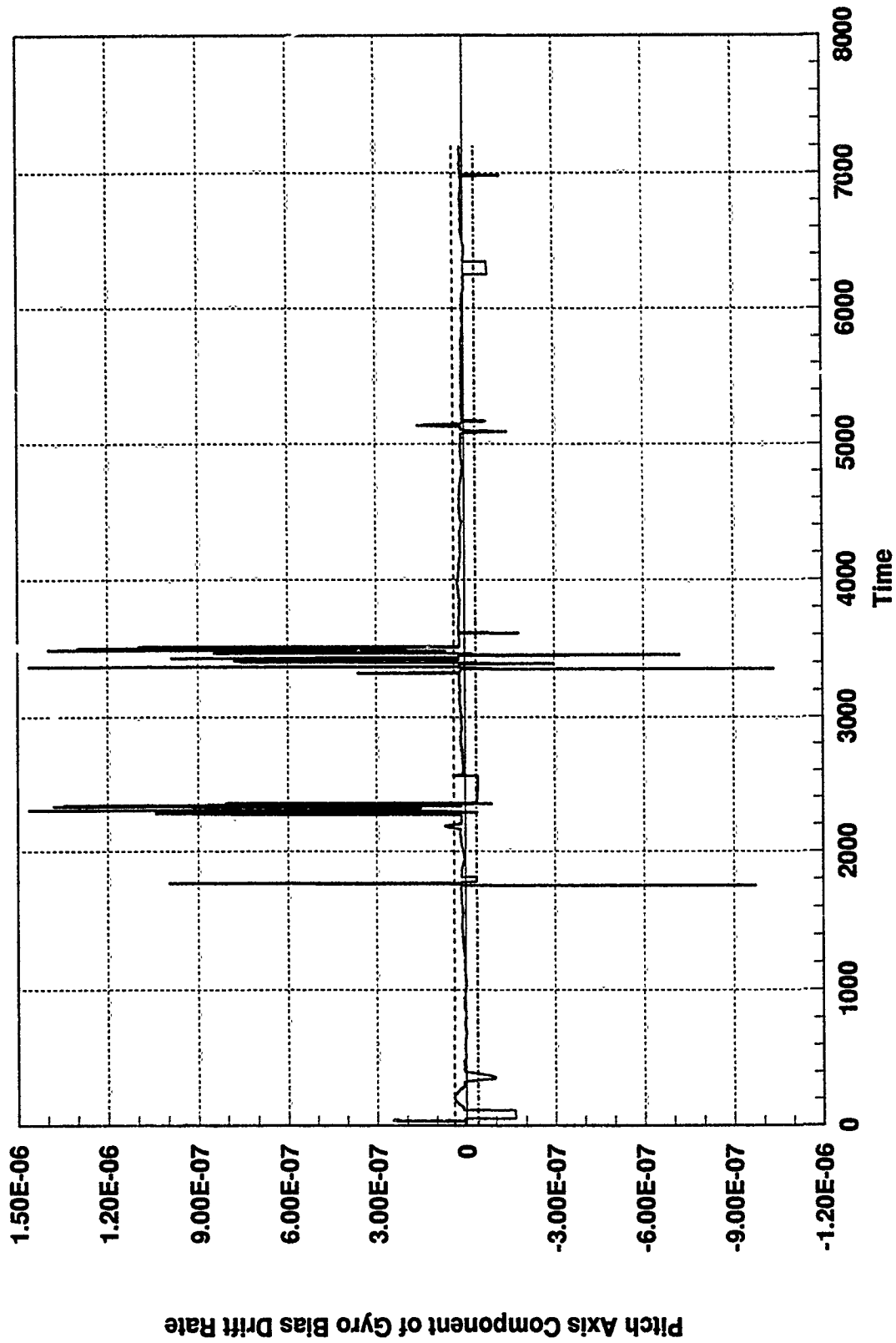
Figure B.15 Federated Filter Design, Master Filter State 15

Pitch Axis Component of Gyro Bias Drift Rate
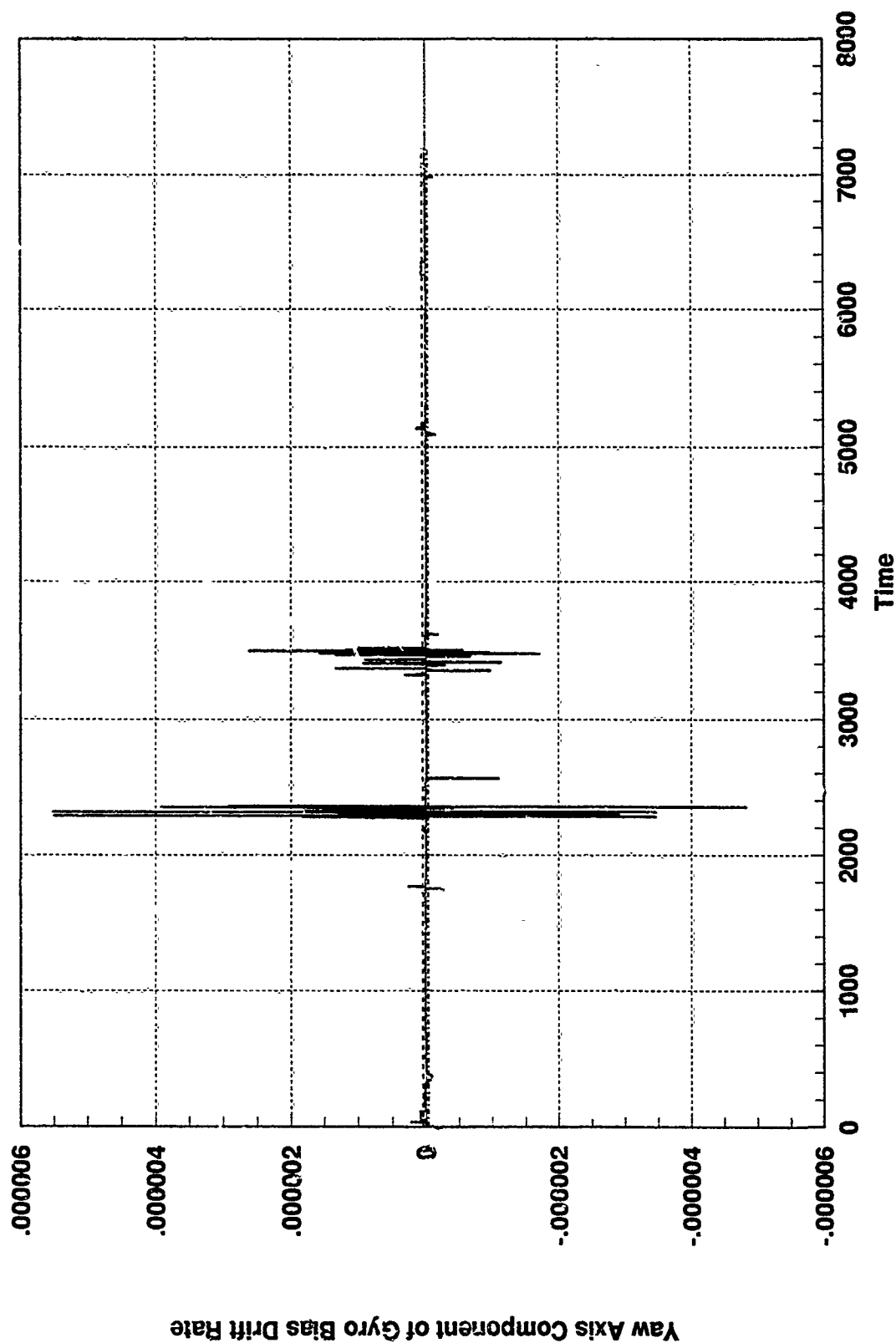
Figure B.16 Federated Filter Design, Master Filter State 16
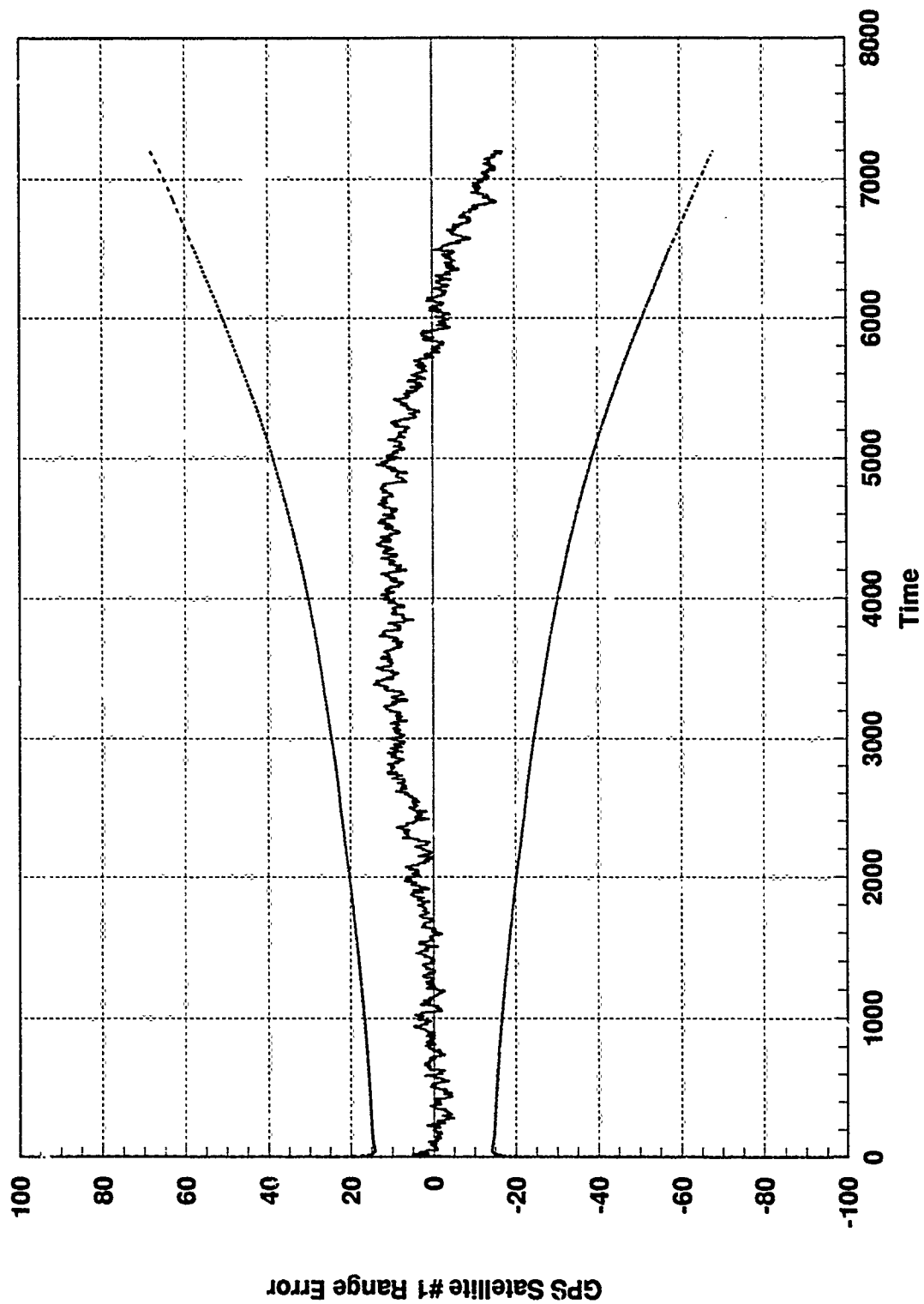
Figure B.17 Federated Filter Design, Local Filter #1 State 17

GPS Satellite #1 Range Error

Time
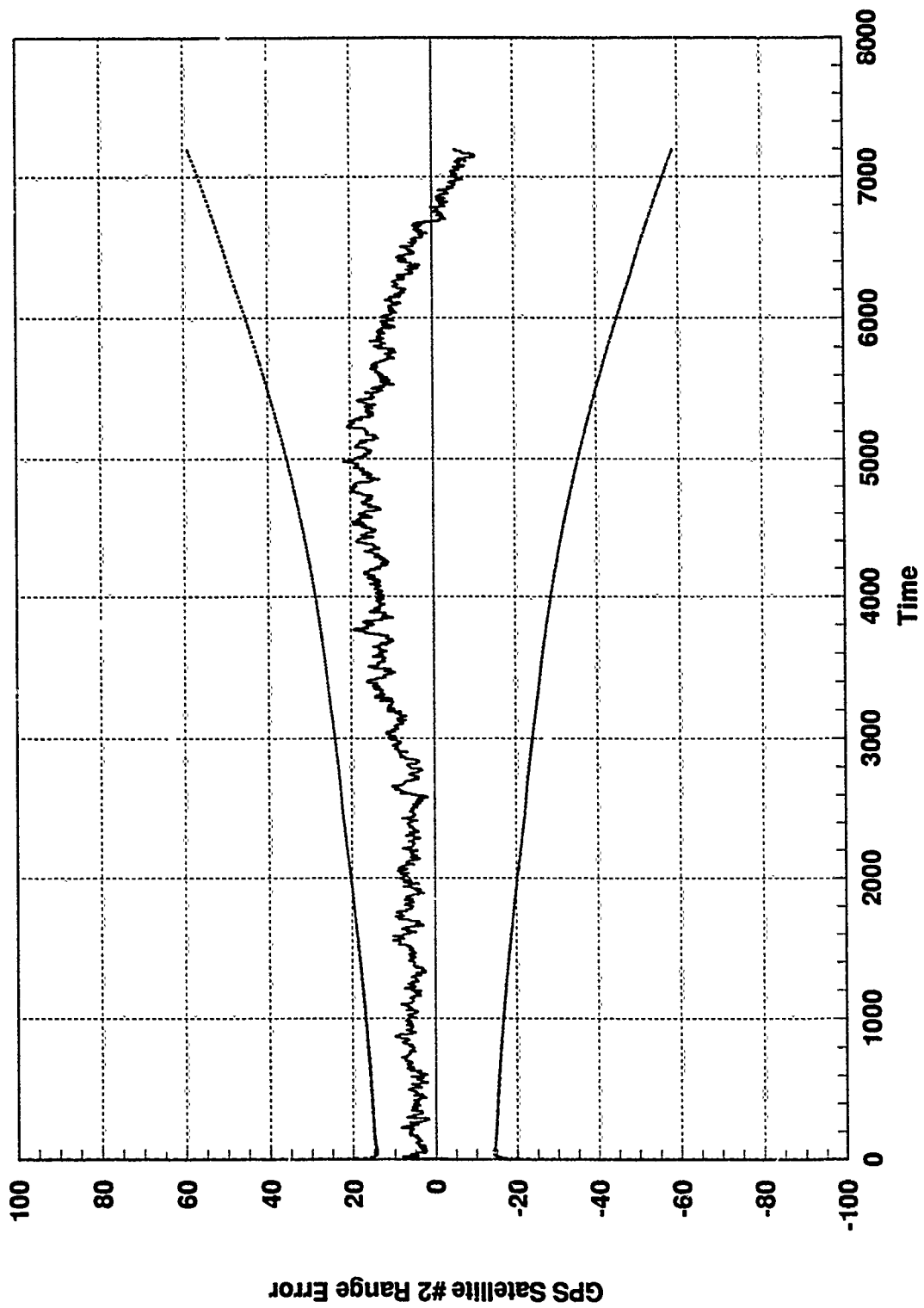
Figure B.18 Federated Filter Design, Local Filter #1 State 18
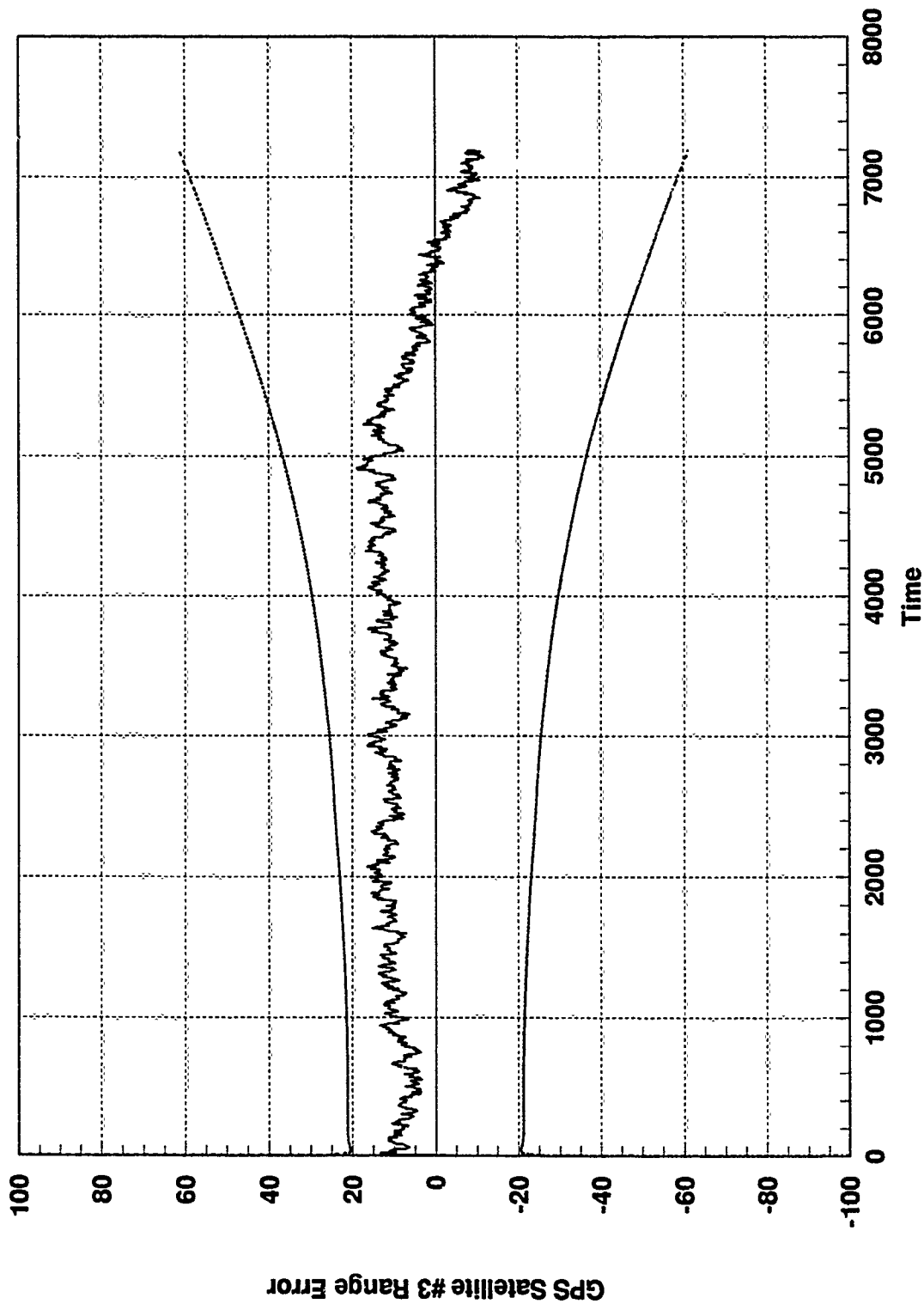
Figure B.19 Federated Filter Design, Local Filter #1 State 19
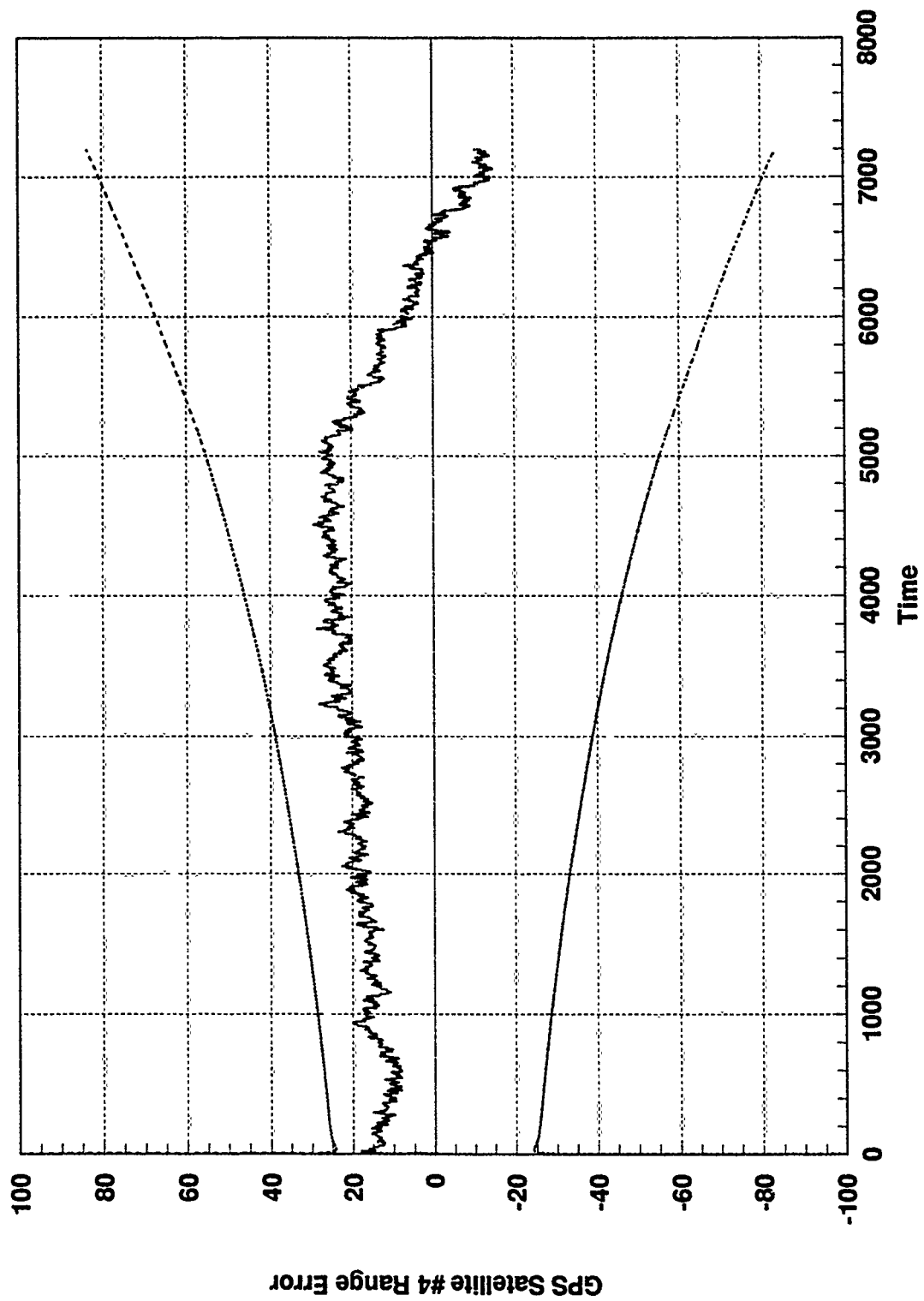
Figure B.20 Federated Filter Design, Local Filter #1 State 20
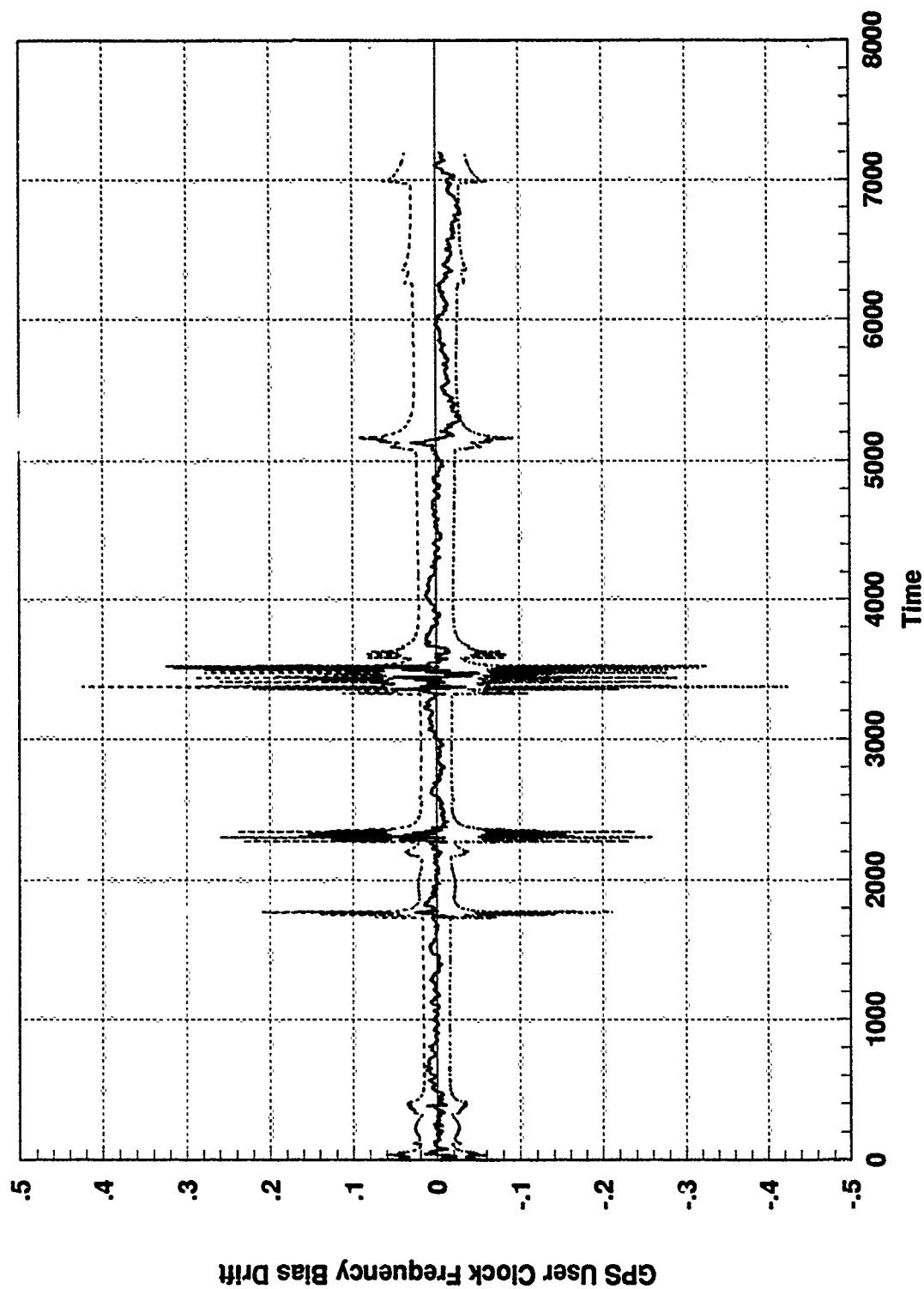
Figure B.21 Federated Filter Design, Local Filter #1 State 21
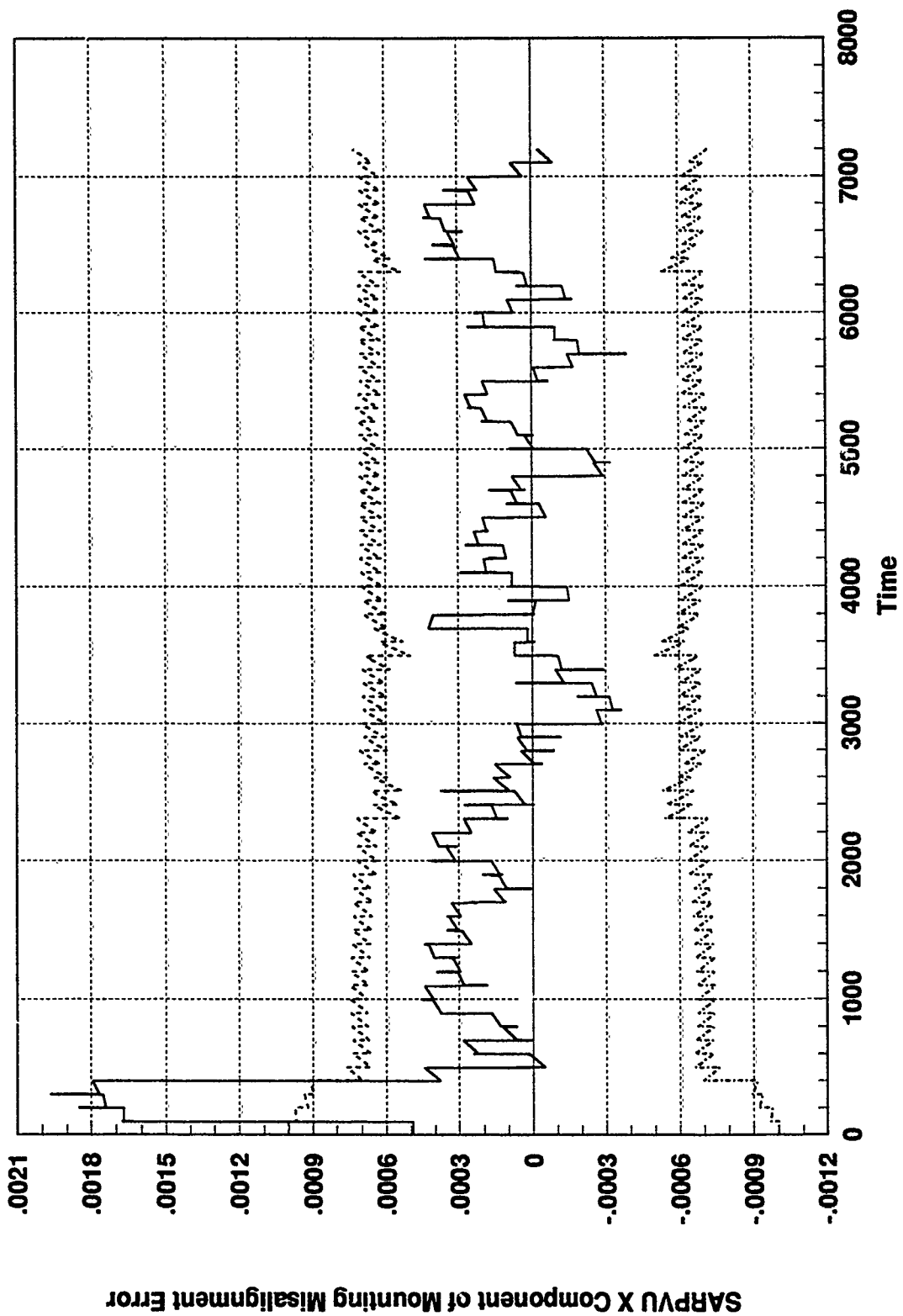
Figure B.22 Federated Filter Design, Local Filter #2 State 17
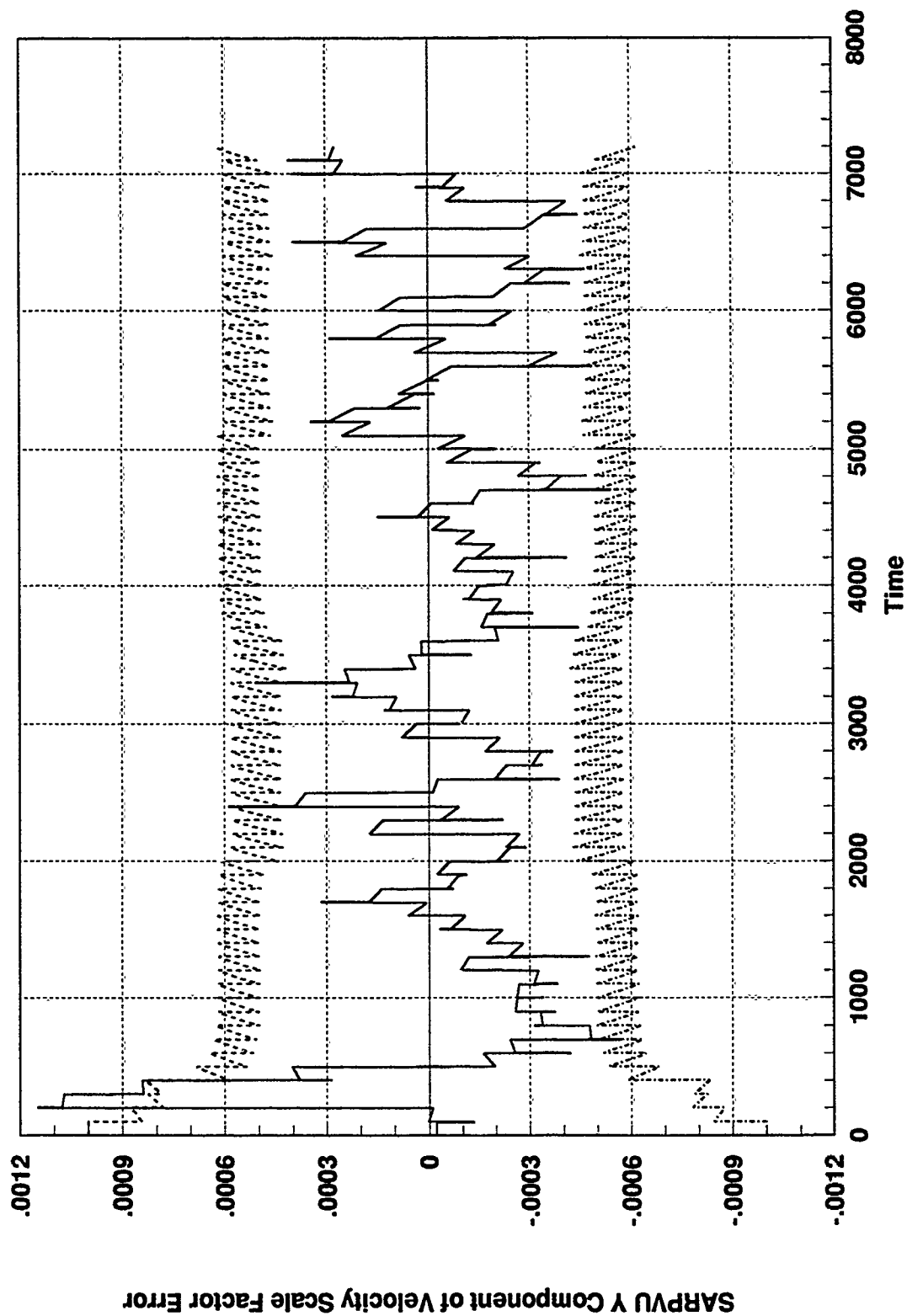
Figure B.23 Federated Filter Design, Local Filter #2 State 18
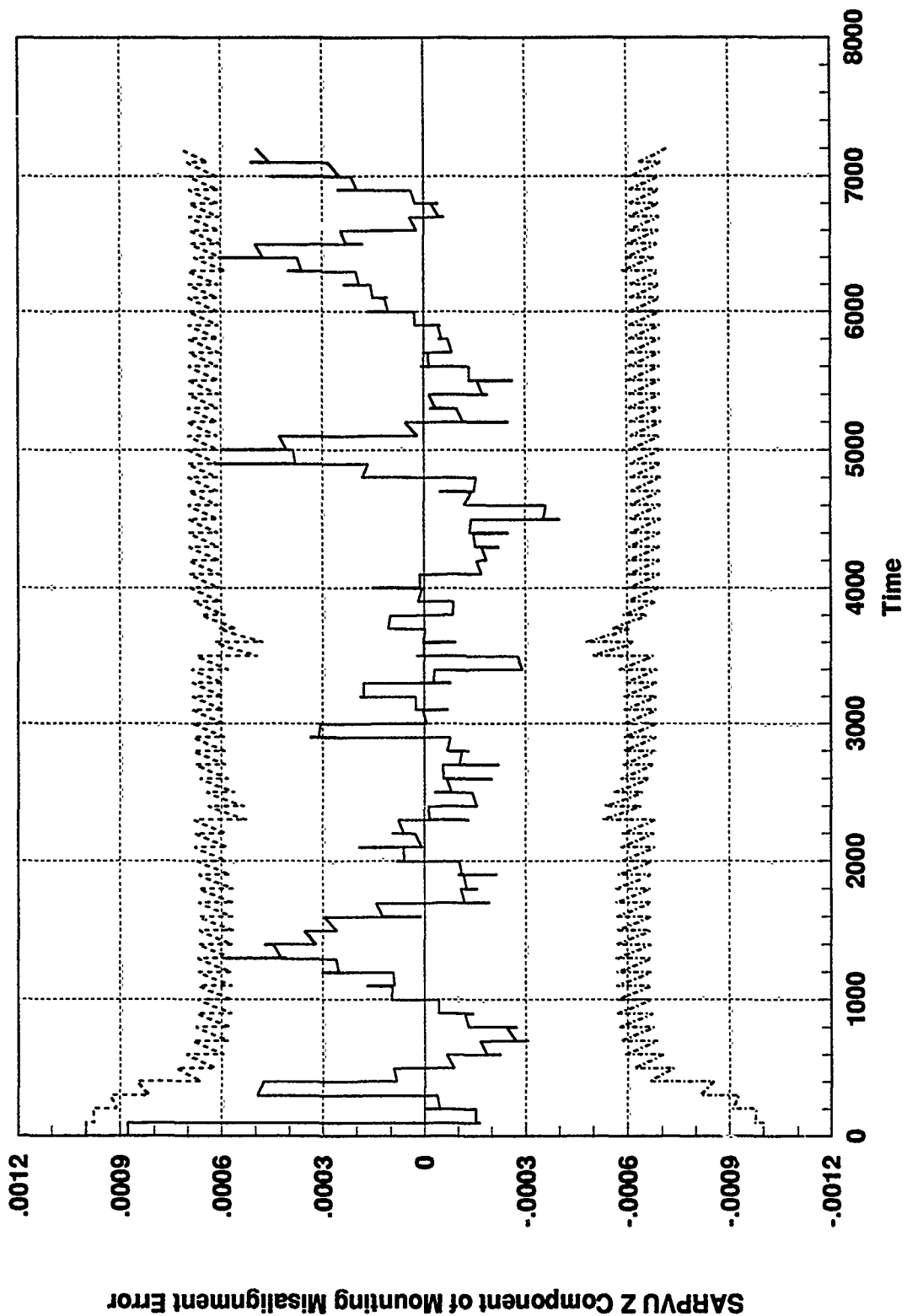
Figure B.24 Federated Filter Design, Local Filter #2 State 19

SARPVU Z Component of Mounting Misalignment Error
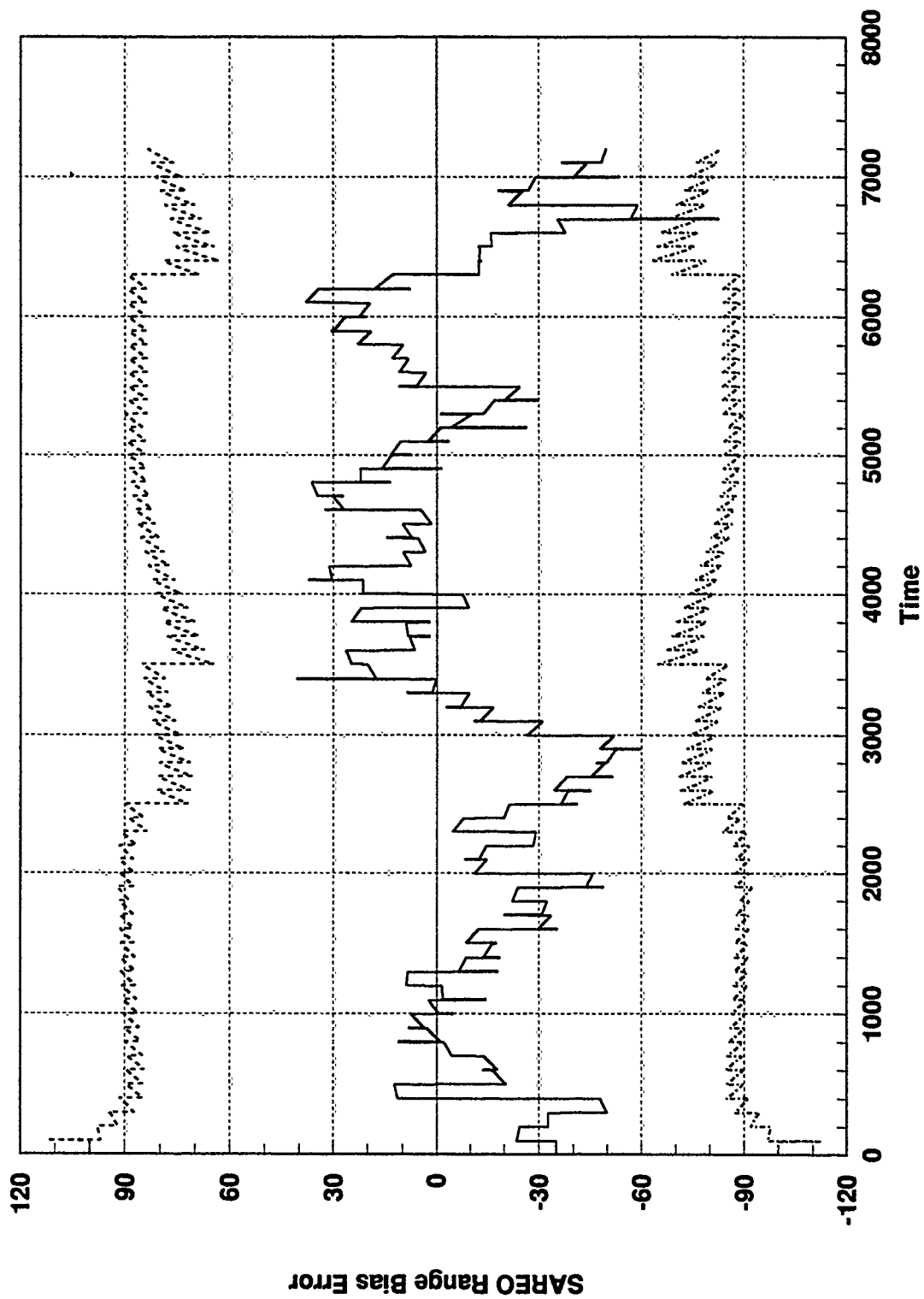
Figure B.25 Federated Filter Design, Local Filter #2 State 20
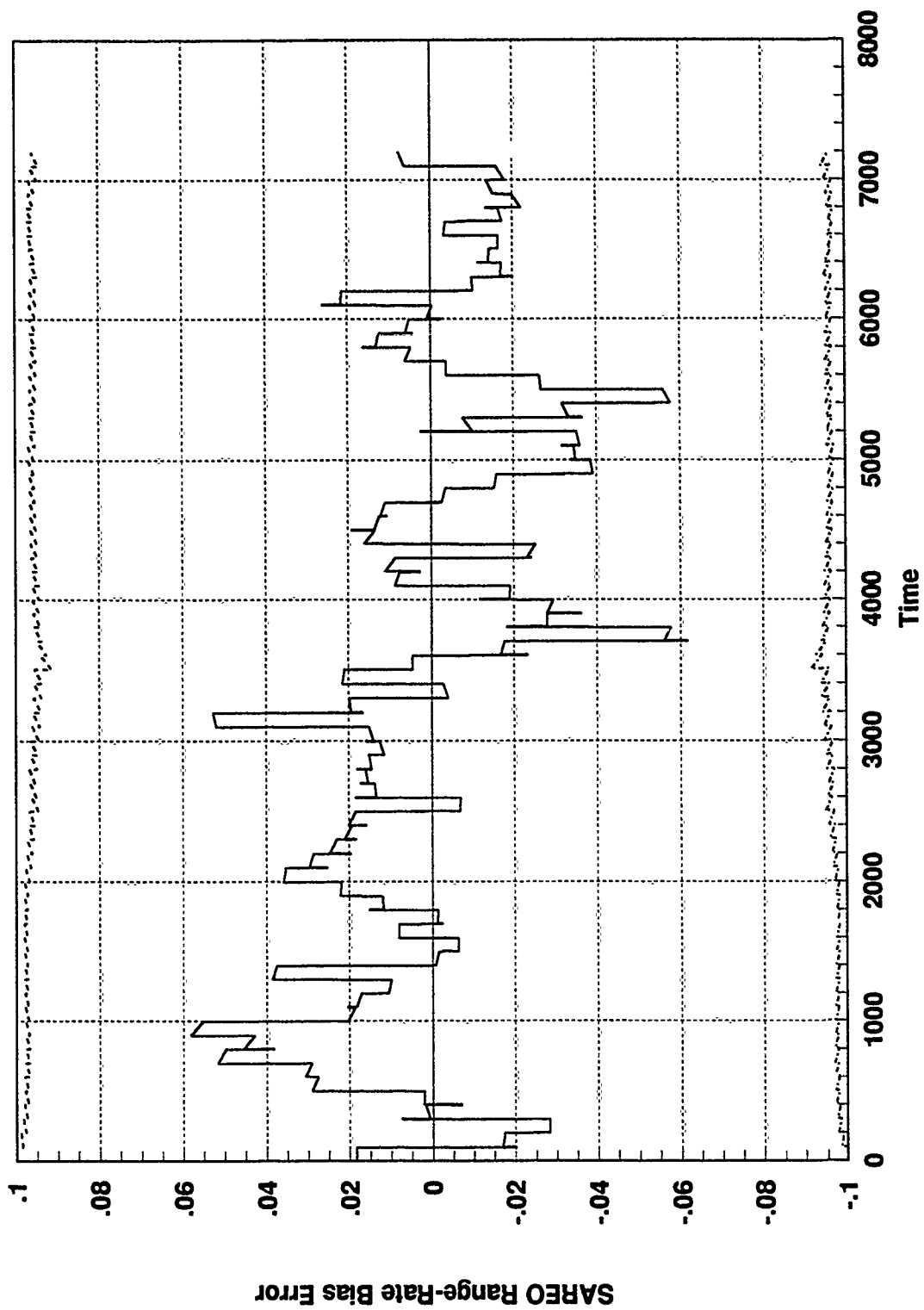
Figure B.26 Federated Filter Design, Local Filter #2 State 21
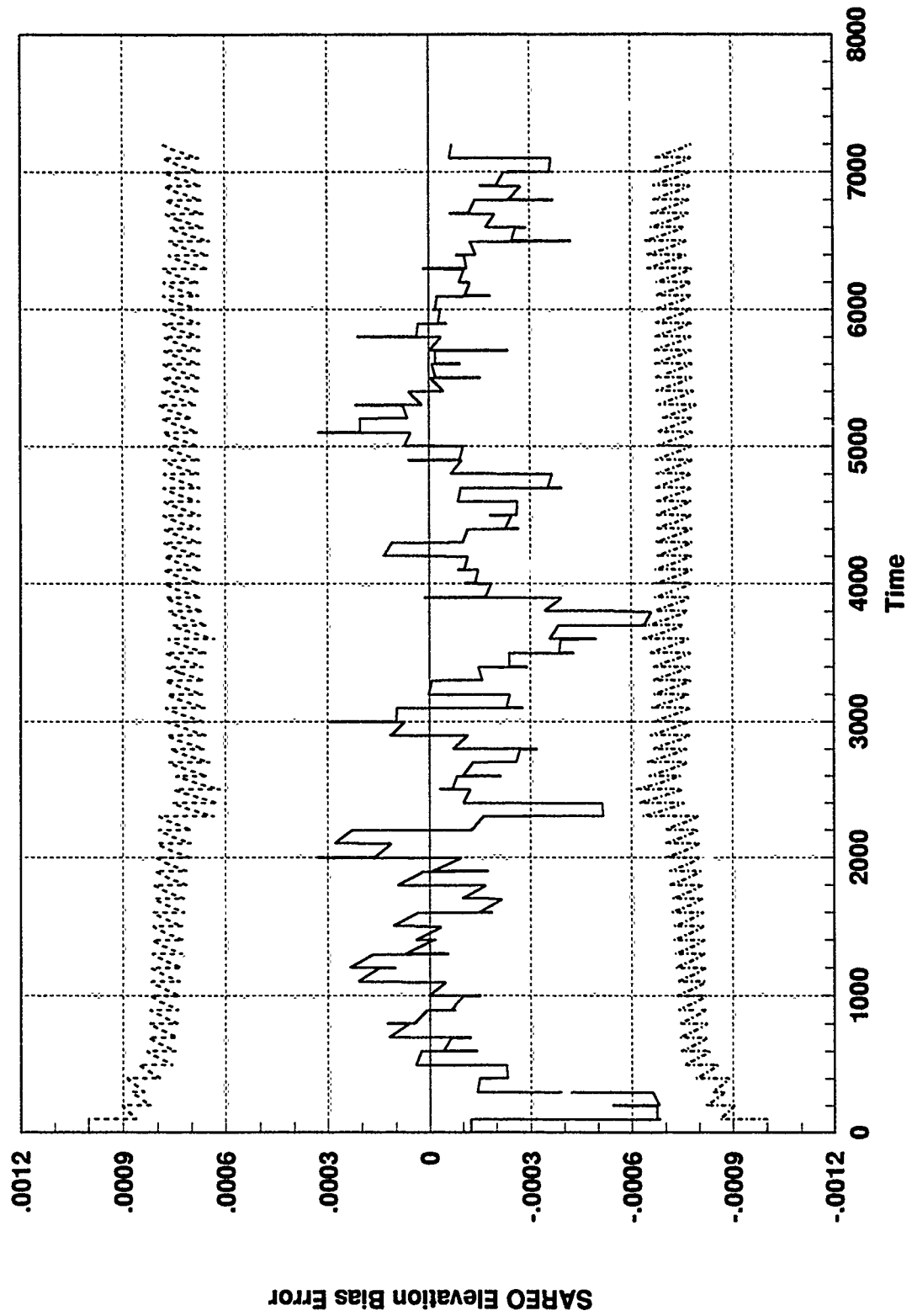
Figure B.27 Federated Filter Design, Local Filter #2 State 22
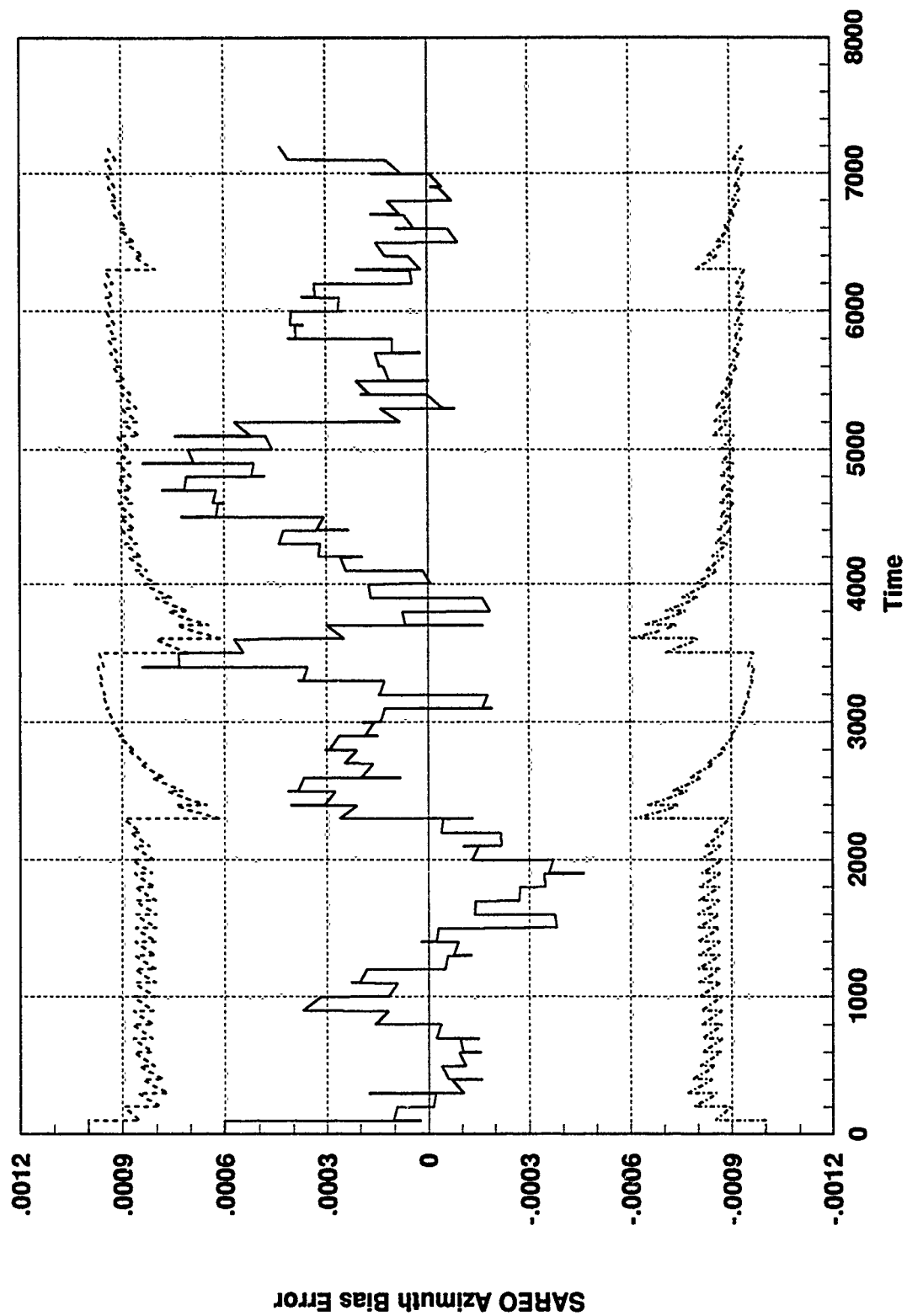
Figure B.28 Federated Filter Design, Local Filter #2 State 23
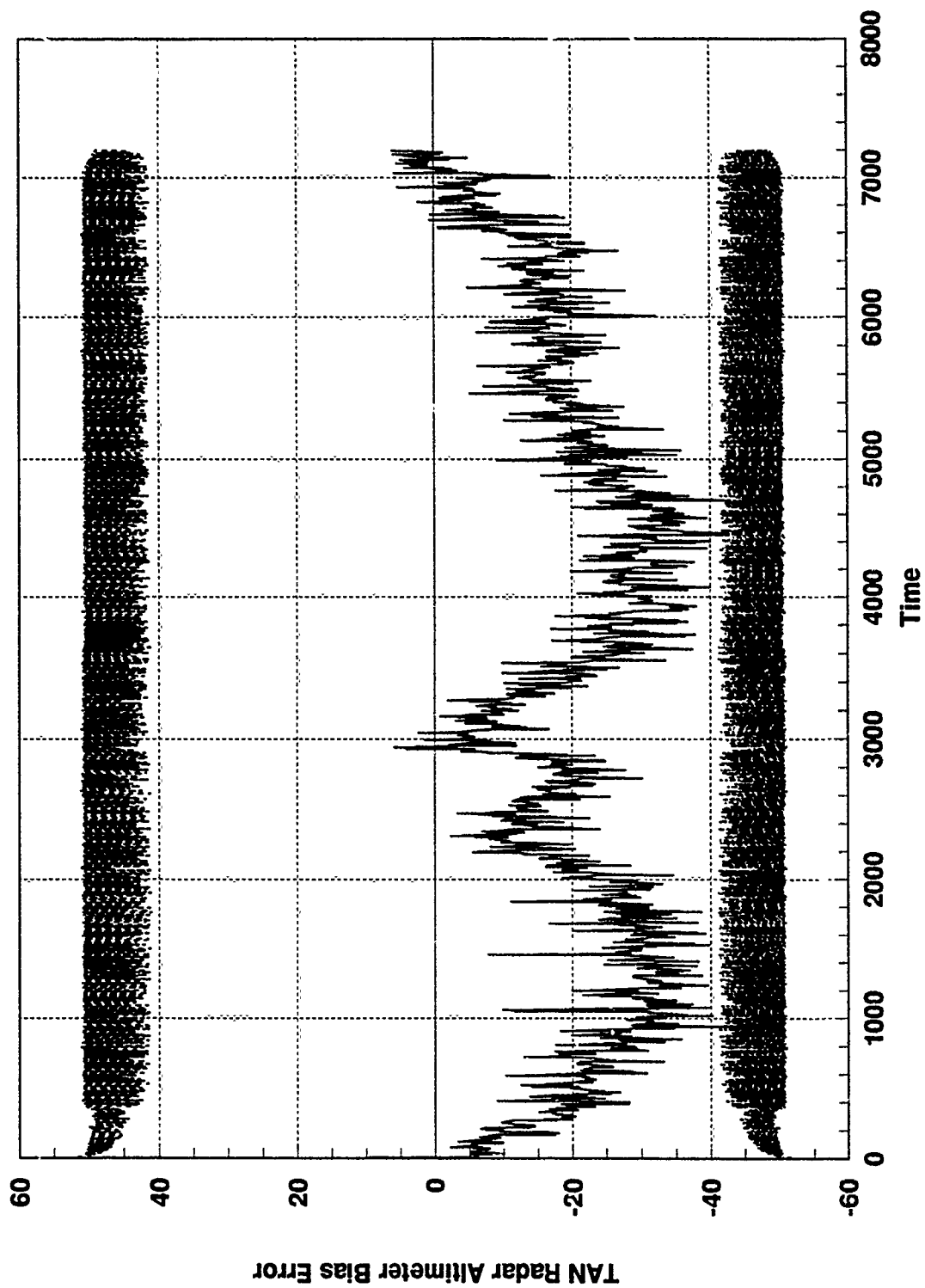
Figure B.29 Federated Filter Design, Local Filter #3 State 17

Appendix C: Centralized Filter Baseline Residual Plots

GENERAL INFORMATION

The series of plots in this appendix allows for a baseline comparison for the fault detection considerations of this thesis. Establishing the specific expectations for the centralized filter residual behavior during normal operating conditions with no induced failures is the intent.

Each of the residual plots contained in this appendix are obtained from Monte Carlo simulations using DKFSIM Version 1.1. All residual outputs are recorded for the entire 7200 second duration of the simulation. The residual values were extracted from two different locations in the software operations. Residuals which fall within the designated tolerance for that particular sensor are output to the data file normally. Residuals which are rejected by the filter due to excessive magnitude were picked off prior to rejection. This allows for continuity in the residual data point plotting regardless of whether or not the filter uses the measurement for an update cycle. This is a critical aspect of the filter residual monitoring scheme for fault detection. The residual outputs and the upper and lower one-sigma bounds were computed for one Monte Carlo simulation.

There are 16 plots, 8 for the GPS, 7 for the SAR, and one for the TAN sensor measurements. Each plot has the appropriate residual description. A listing of the titles of each these figures is found in the List of Figures at the beginning of this document.

Figure C.1 GPS Sat 1 Residual and One-Sigma Bound, Normal Conditions
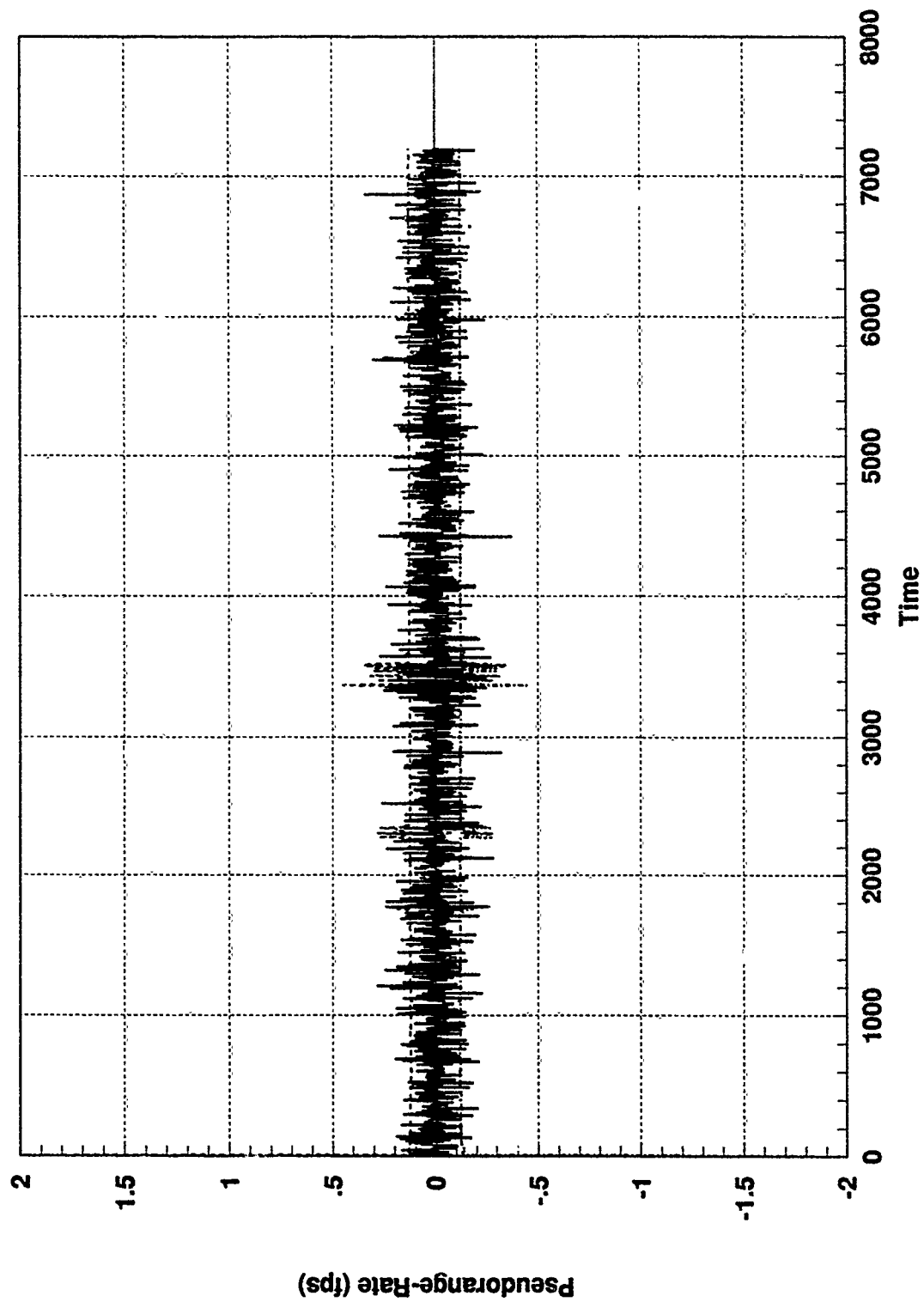
Figure C.2 GPS Sat 1 Residual and One-Sigma Bound, Normal Conditions
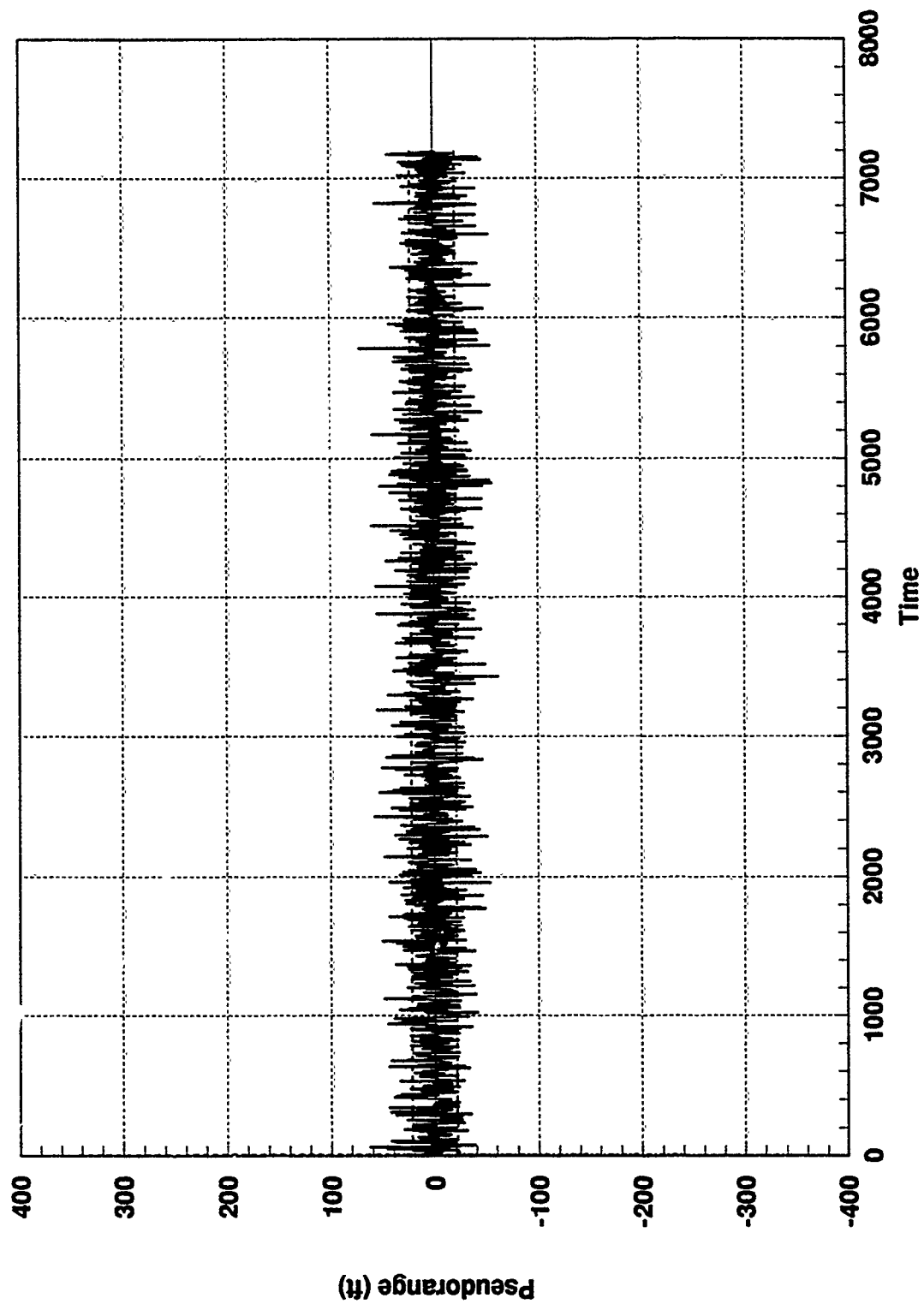
Figure C.3 GPS Sat 2 Residual and One-Sigma Bound, Normal Conditions
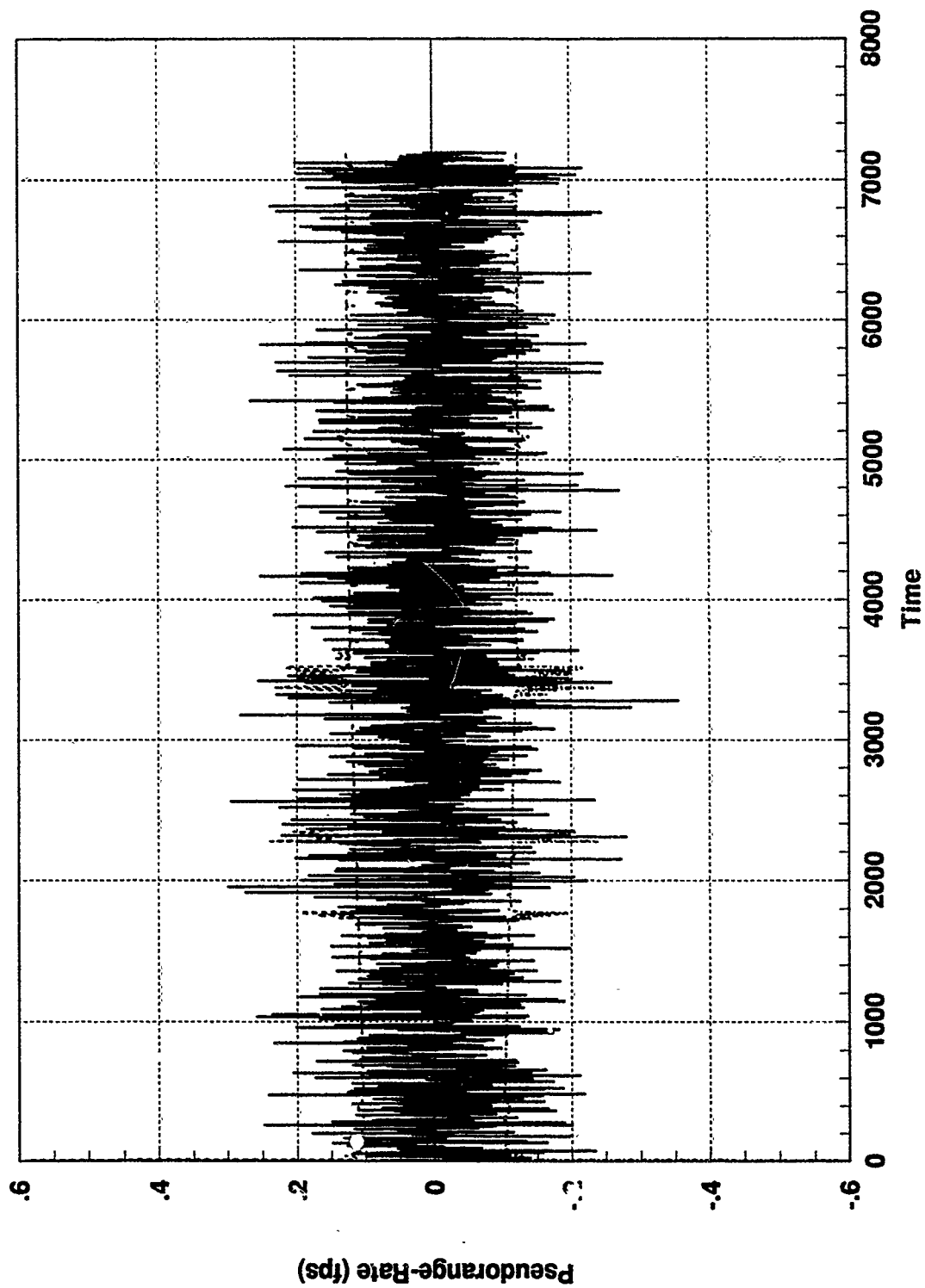
Figure C.4 GPS Sat 2 Residual and One-Sigma Bound, Normal Conditions
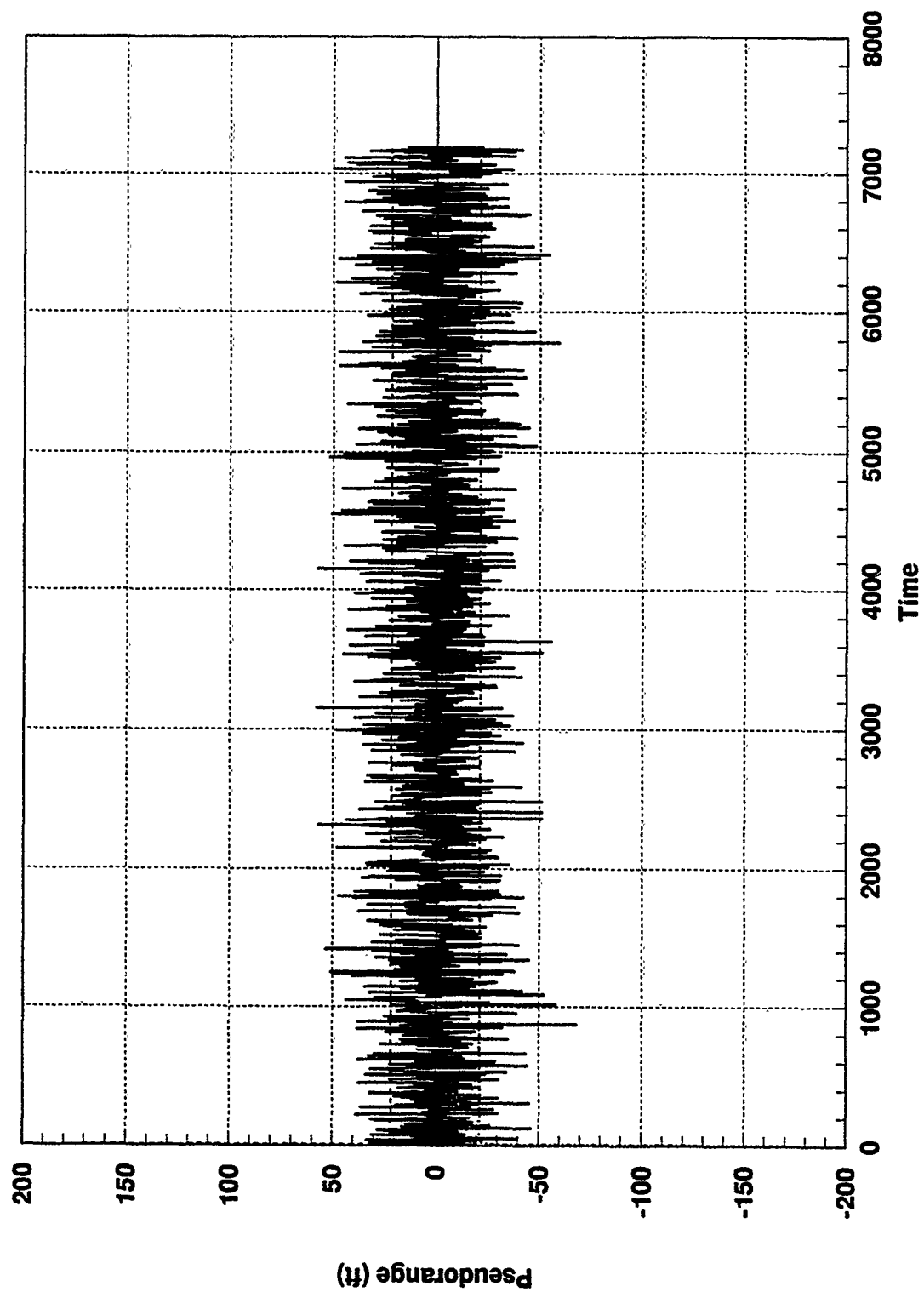
Figure C.5 GPS Sat 3 Residual and One-Sigma Bound, Normal Conditions
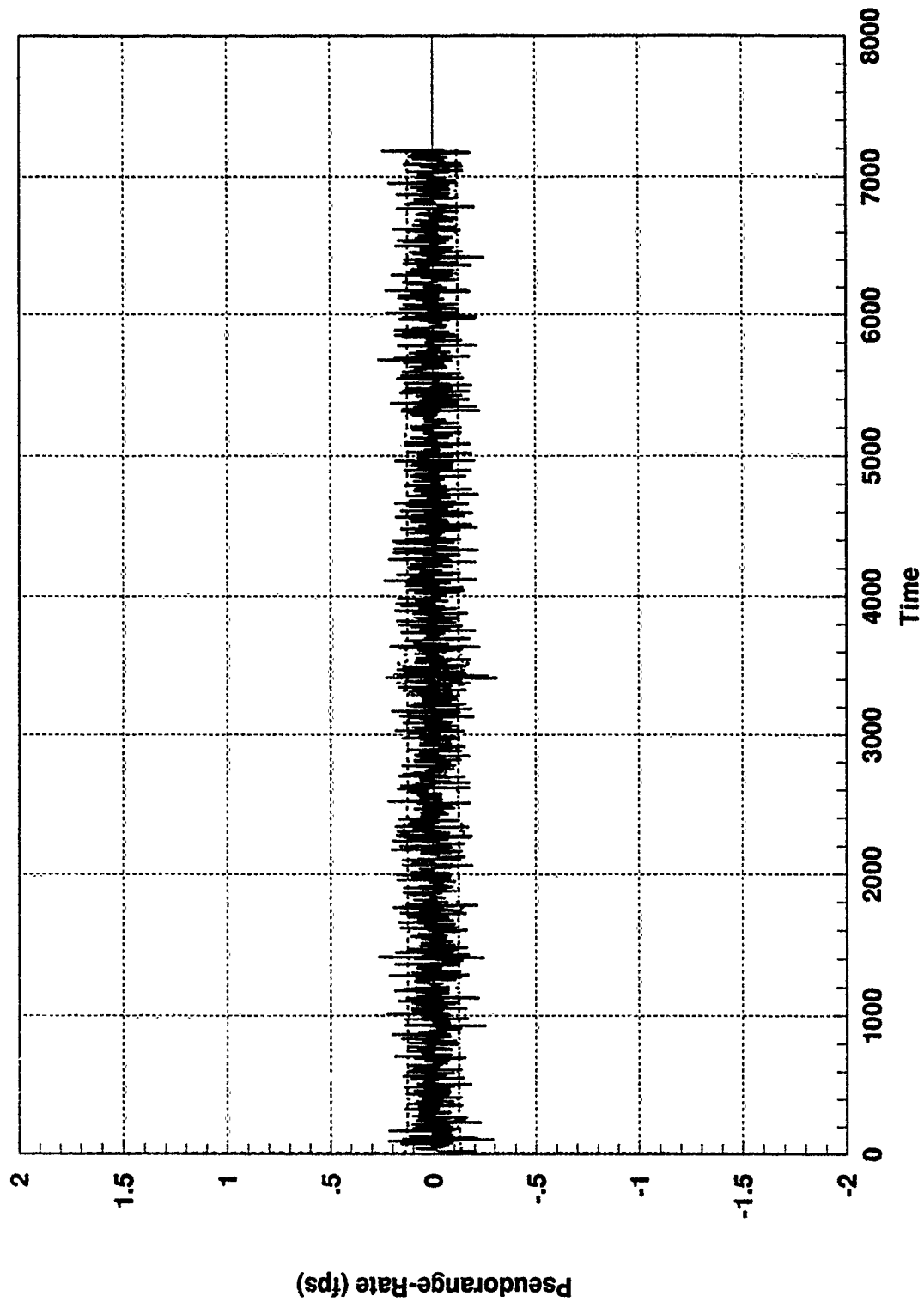
Figure C.6 GPS Sat 3 Residual and One-Sigma Bound, Normal Conditions
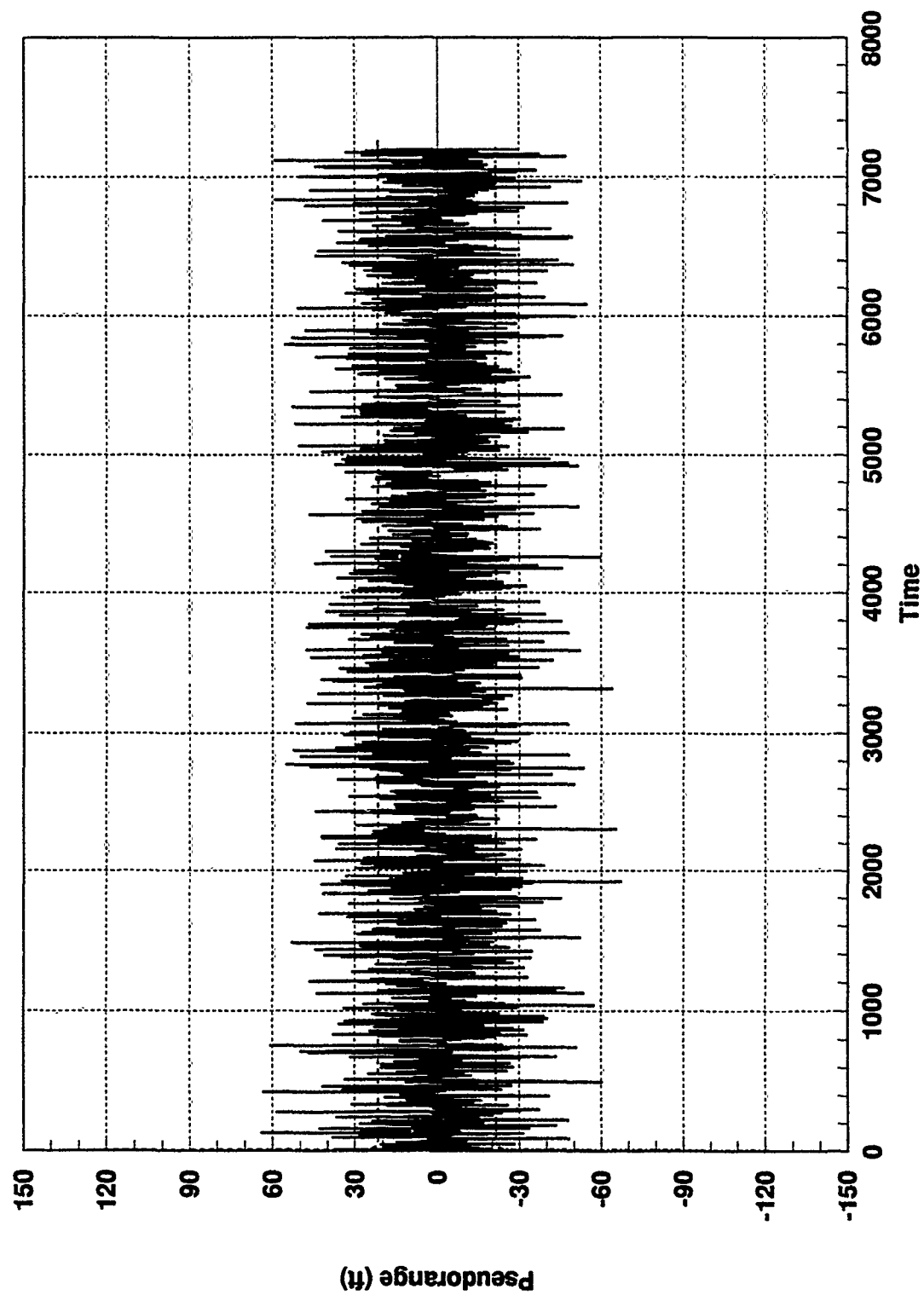
Figure C.7 GPS Sat 4 Residual and One-Sigma Bound, Normal Conditions
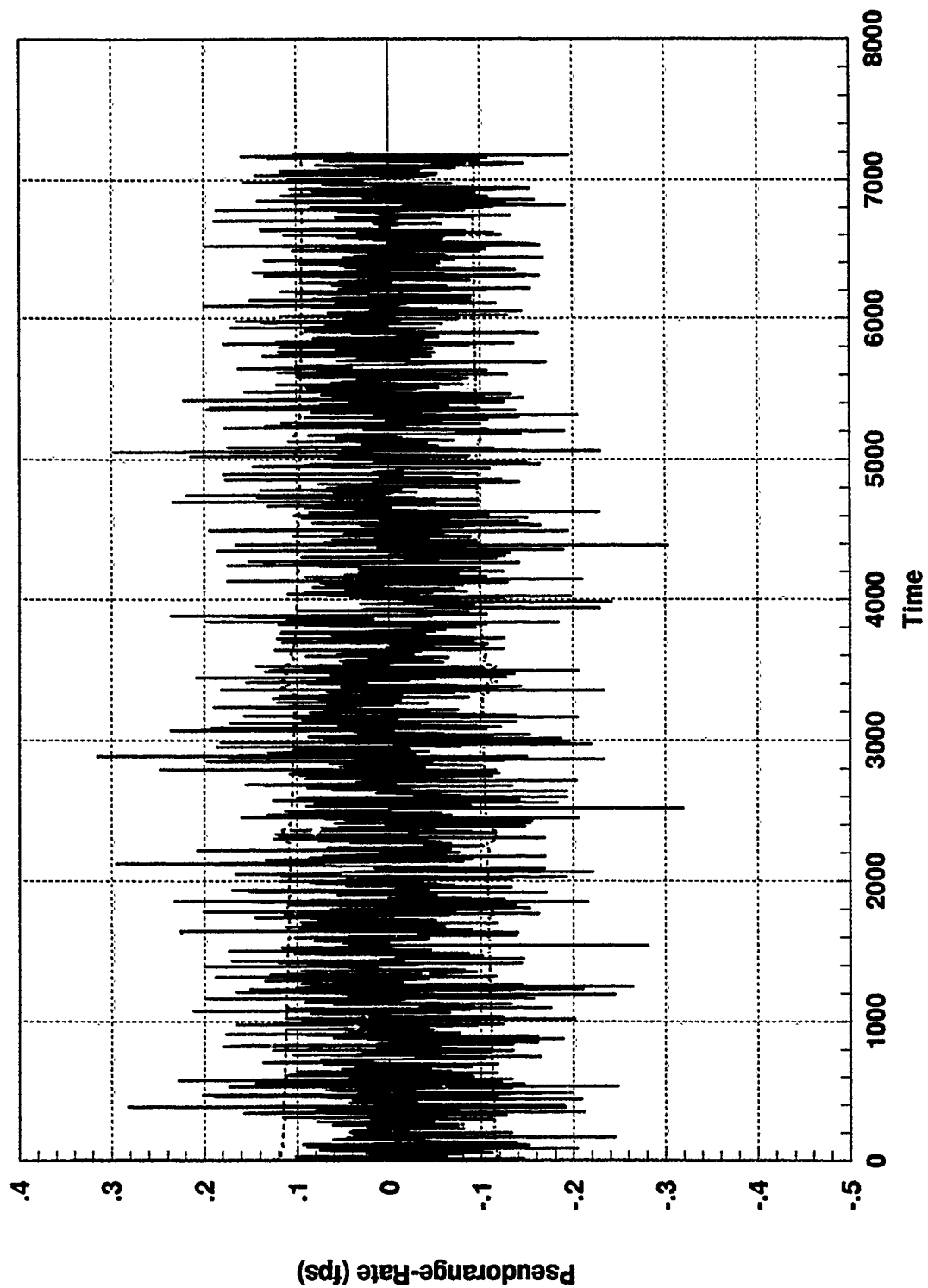
Figure C.8 GPS Sat 4 Residual and One-Sigma Bound, Normal Conditions
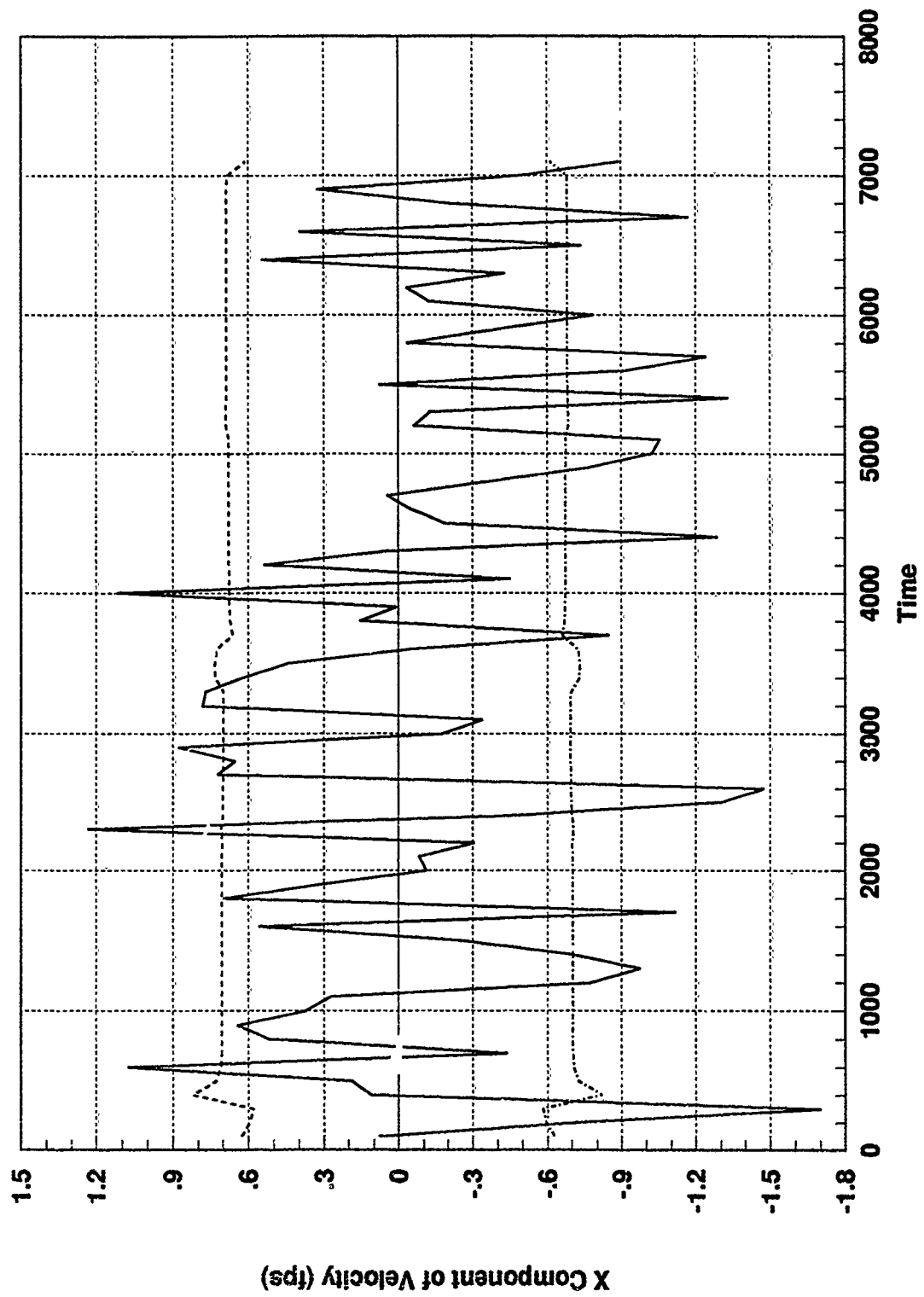
Figure C.9 SARPVU Residual, One-Sigma Bound, Normal Conditions
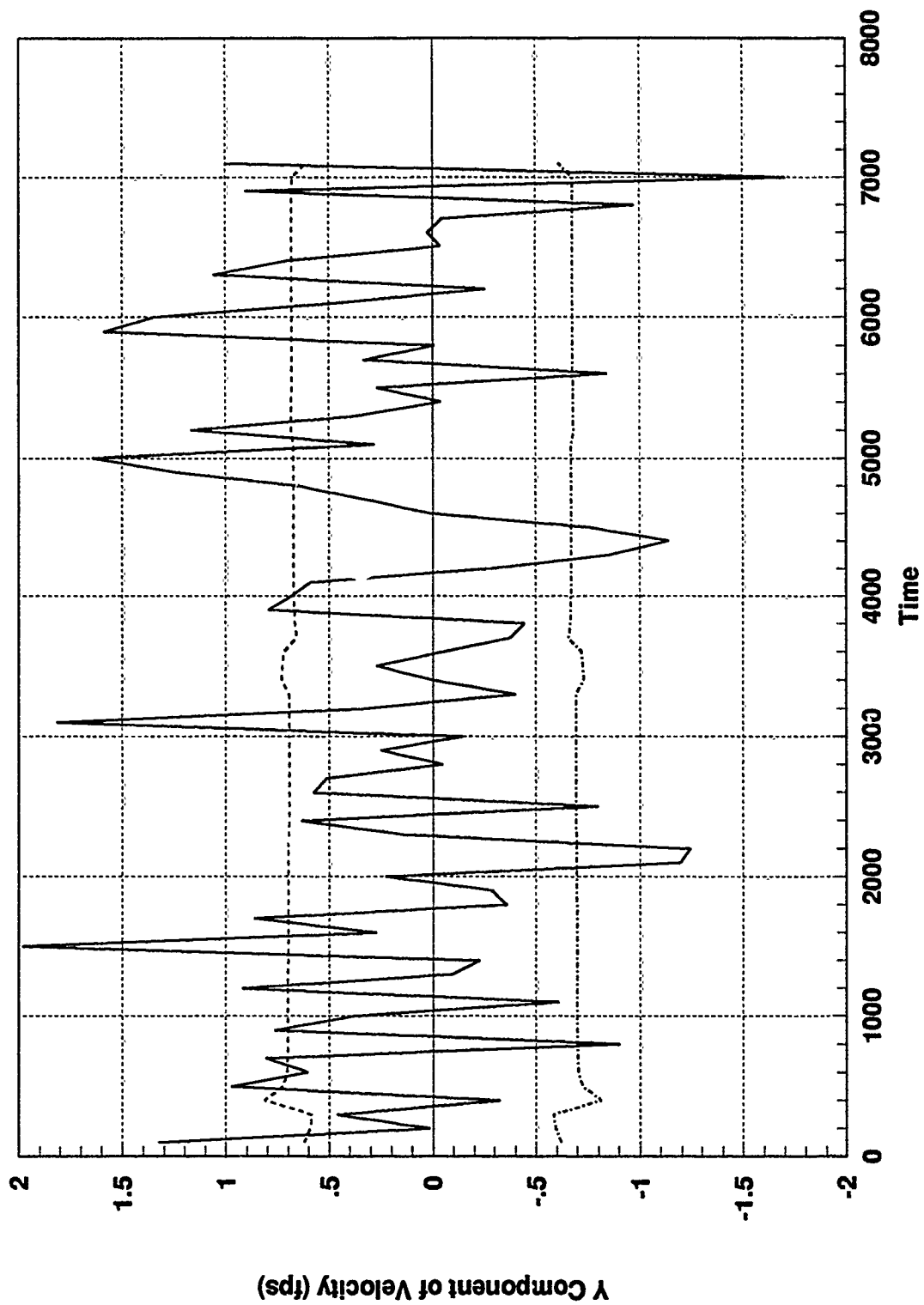
Figure C.10 SARPVU Residual, One-Sigma Bound, Normal Conditions
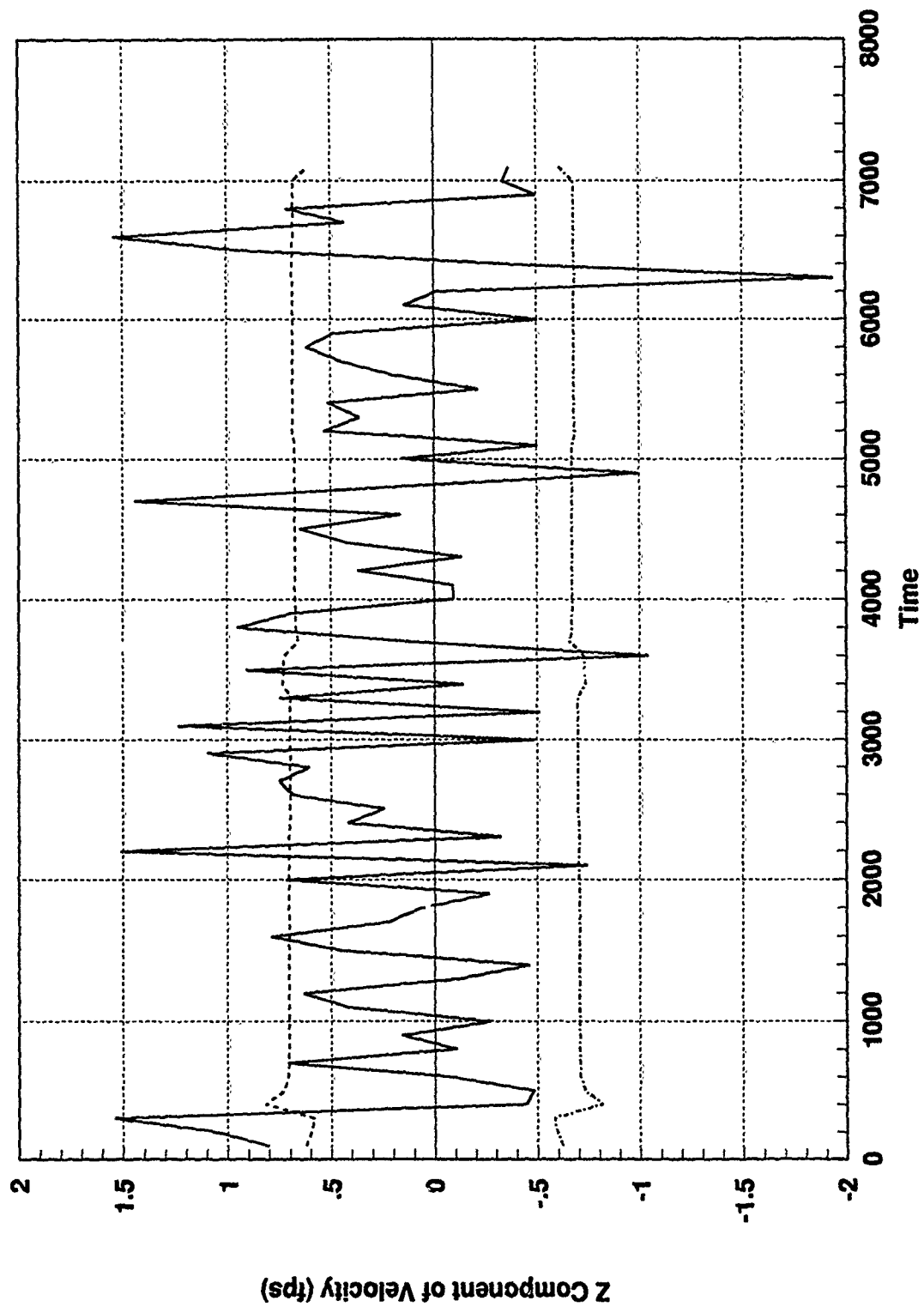
Figure C.11 SARPVU Residual, One-Sigma Bound, Normal Conditions
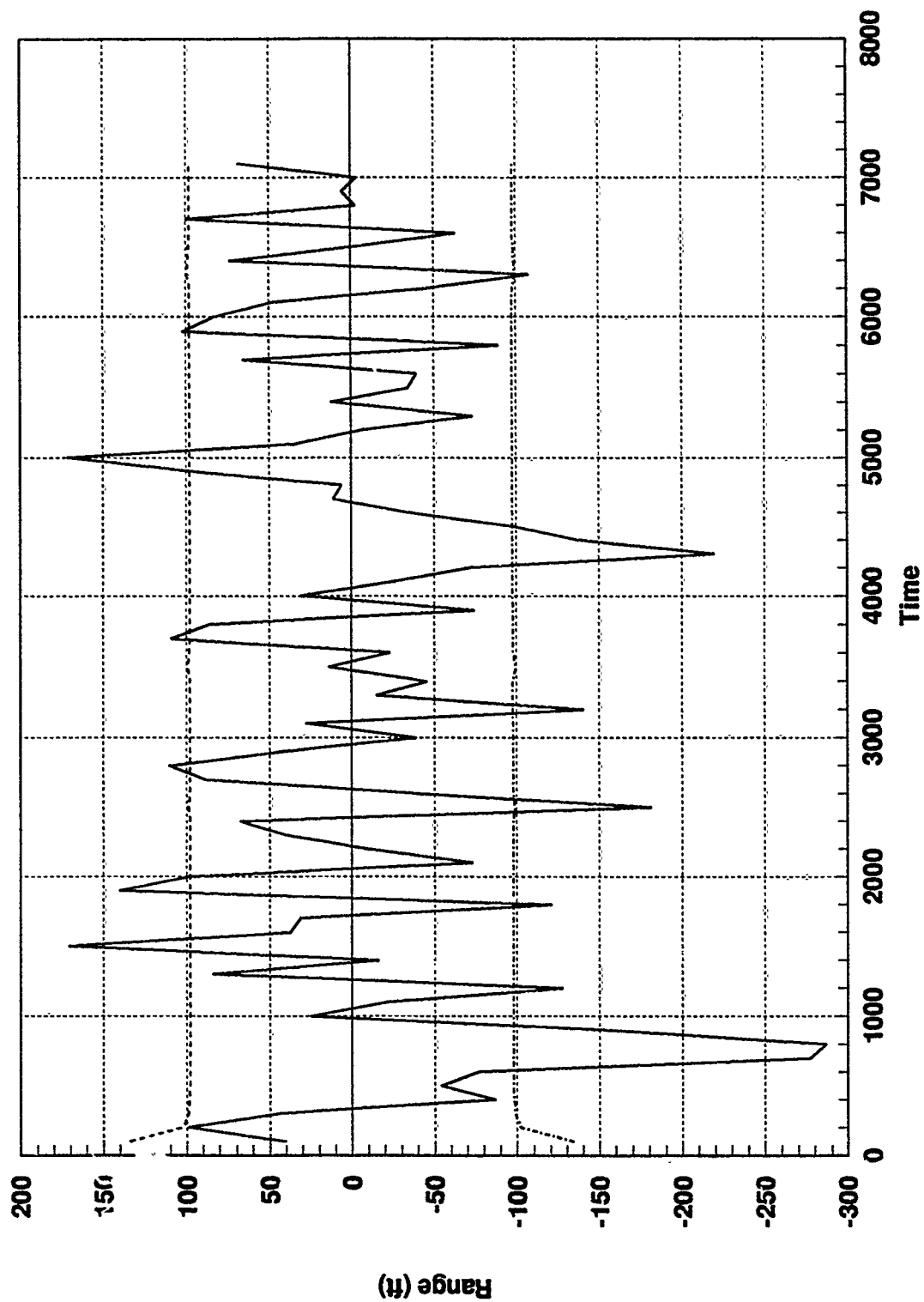
Figure C.12 SAREO Residual, One-Sigma Bound, Normal Conditions
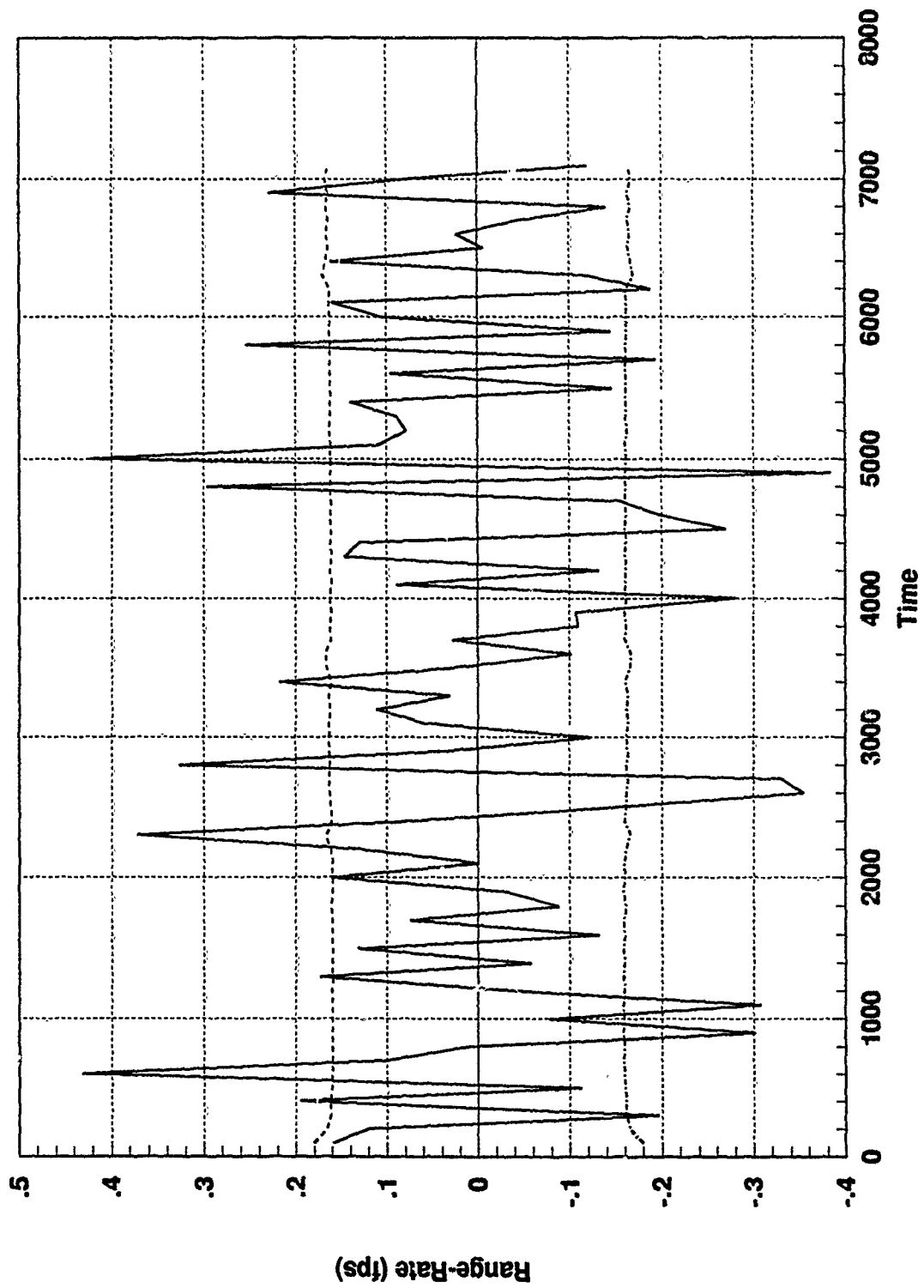
Figure C.13 SAREO Residual, One-Sigma Bound, Normal Conditions

C-14
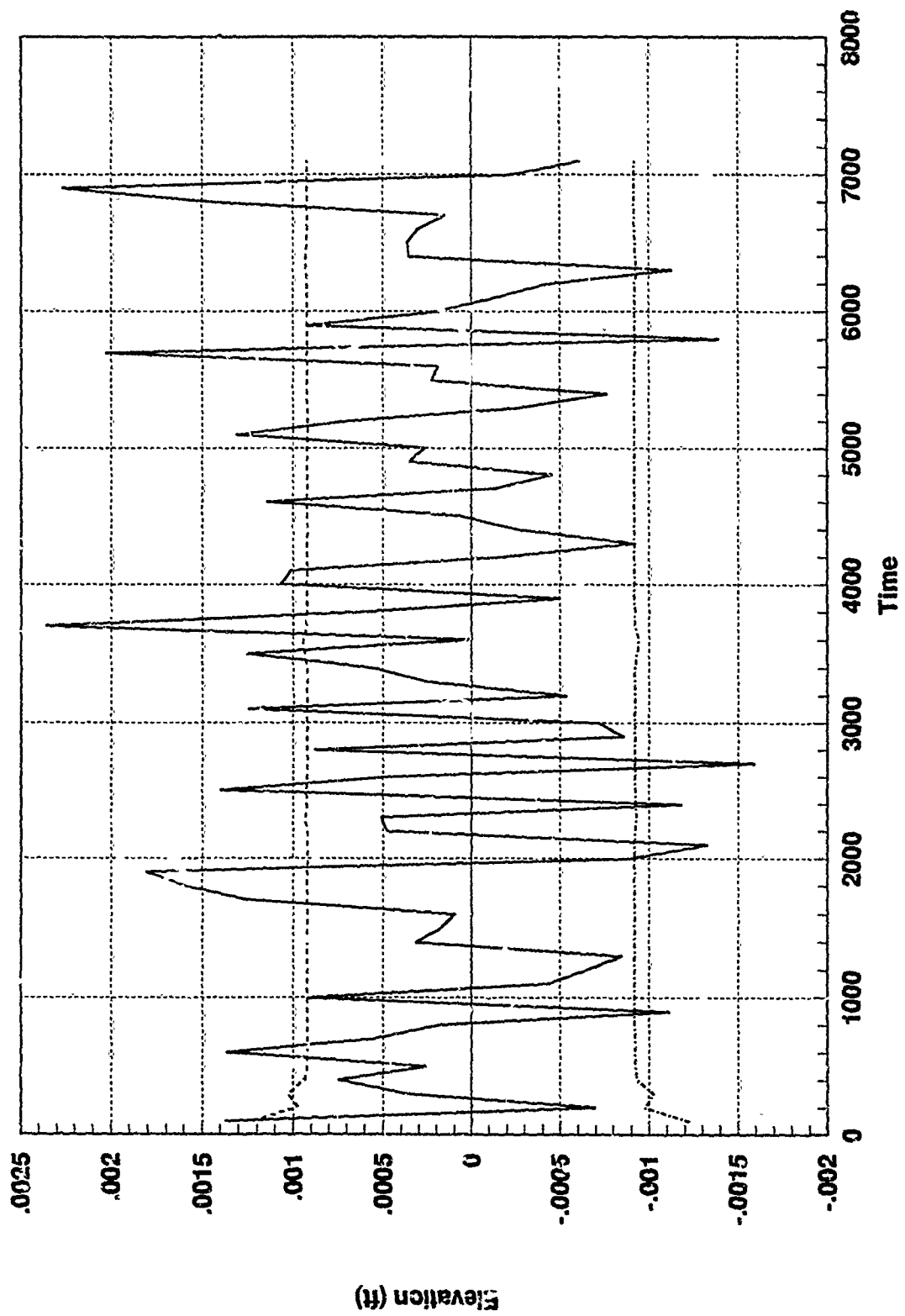
Figure C.14 SAREO Residual, One-Sigma Bound, Normal Conditions
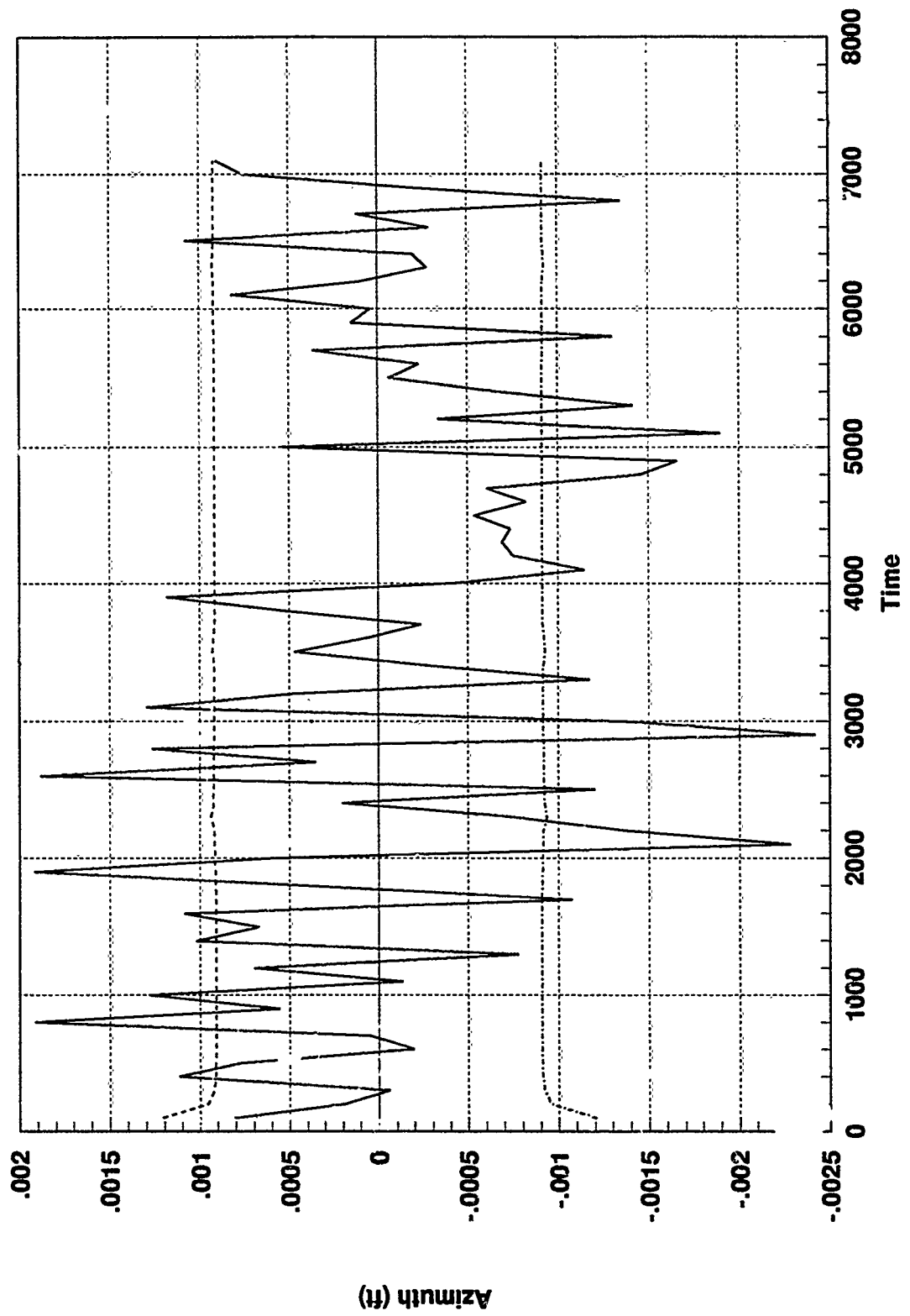
Figure C.15 SAREO Residual, One-Sigma Bound, Normal Conditions
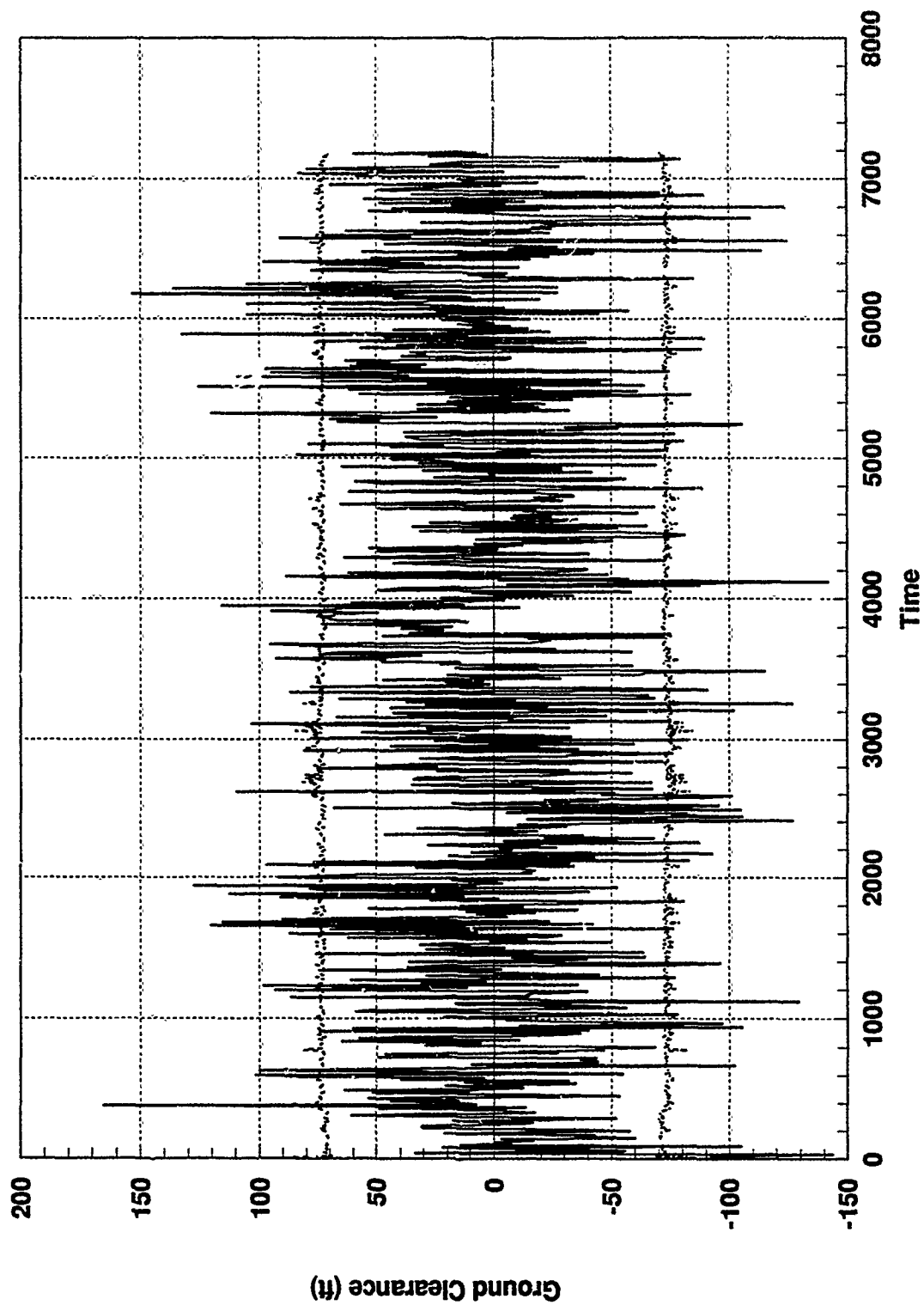
Figure C.16 Centralized Filter TAN Residual, One-Sigma Bound, Normal Conditions

Appendix D: Federated Filter Baseline Residual Plots

GENERAL INFORMATION

The series of plots in this appendix allows for a baseline comparison for the fault detection considerations of this thesis. Establishing the specific expectations for the federated filter residual behavior during normal operating conditions with no induced failures is the intent.

Each of the residual plots contained in this appendix are obtained from Monte Carlo simulations using DKFSIM Version 1.1. All residual outputs are recorded for the entire 7200 second duration of the simulation. The residual values were extracted from two different locations in the software operations. Residuals which fall within the designated tolerance for that particular sensor are output to the data file normally. Residuals which are rejected by the filter due to excessive magnitude were picked off prior to rejection. This allows for continuity in the residual data point plotting regardless of whether or not the filter uses the measurement for an update cycle. This is a critical aspect of the filter residual monitoring scheme for fault detection. The residual outputs and the upper and lower one-sigma bounds were computed for one Monte Carlo simulation.

There are 16 plots, 8 for the GPS, 7 for the SAR, and one for the TAN sensor measurements. Each plot has the appropriate residual description. A listing of the titles of each these figures is found in the List of Figures at the beginning of this document.
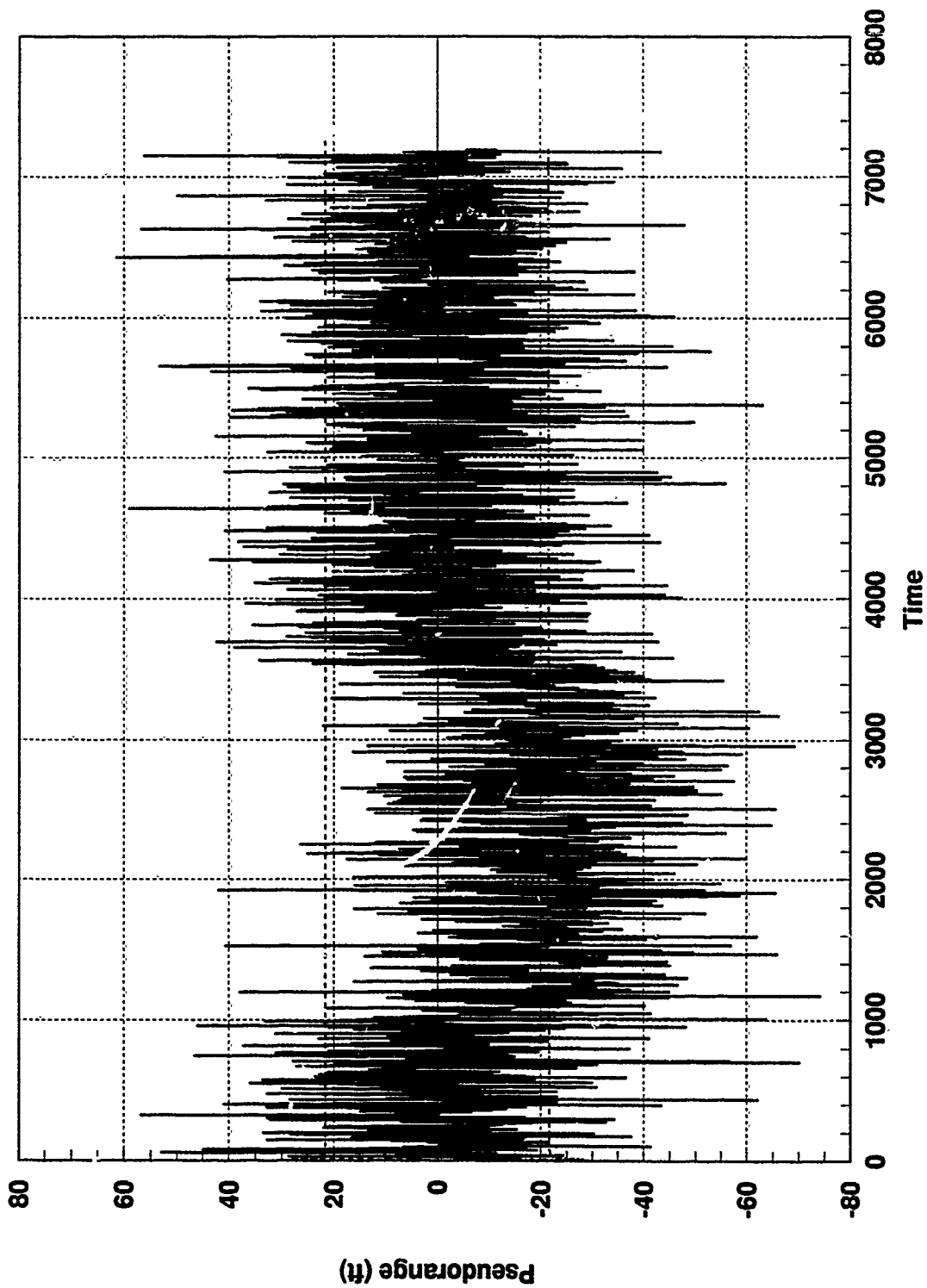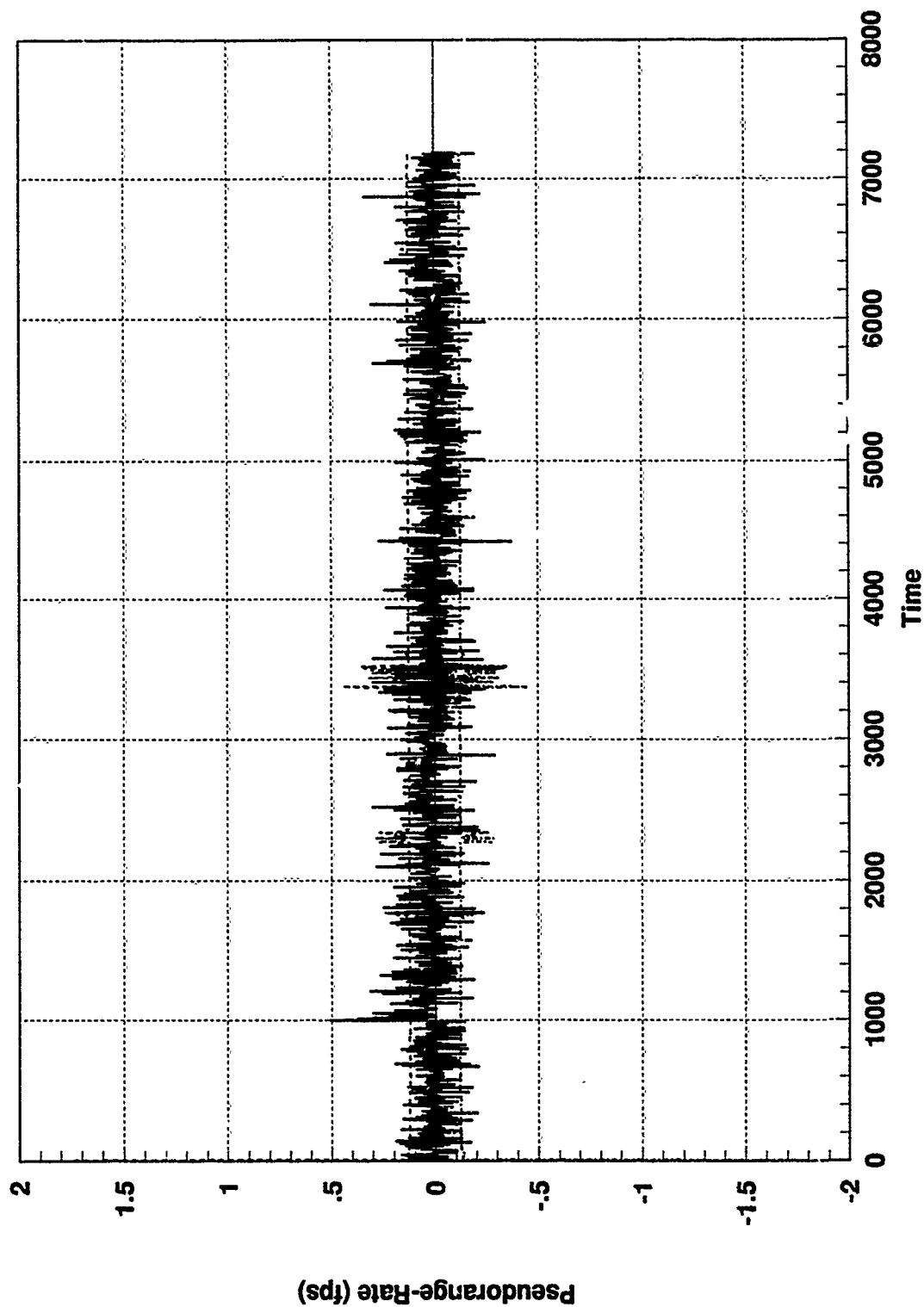
Figure D.1 GPS Sat 1 Residual and One-Sigma Bound, Normal Conditions
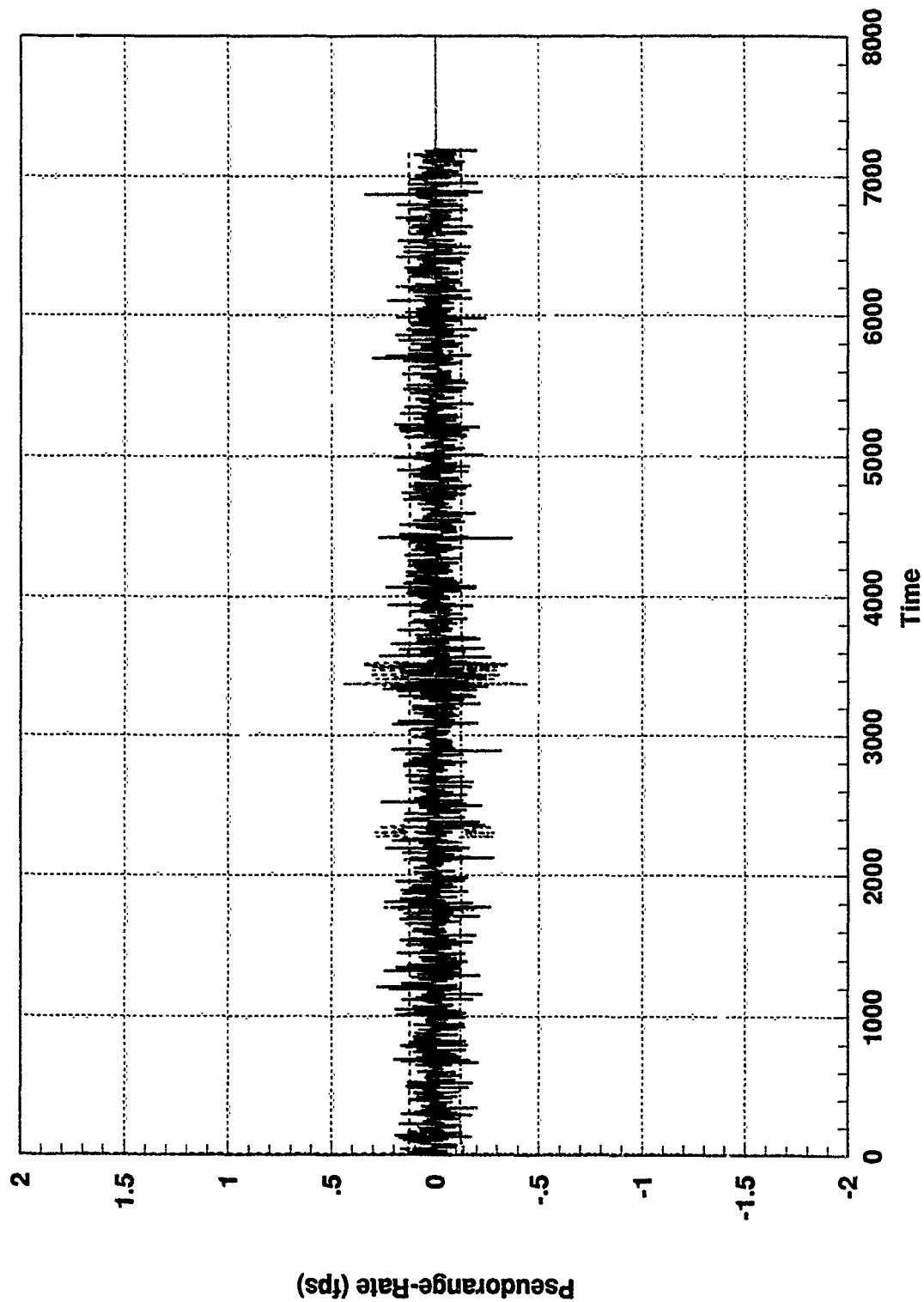
Figure D.2 GPS Sat 1 Residual and One-Sigma Bound, Normal Conditions
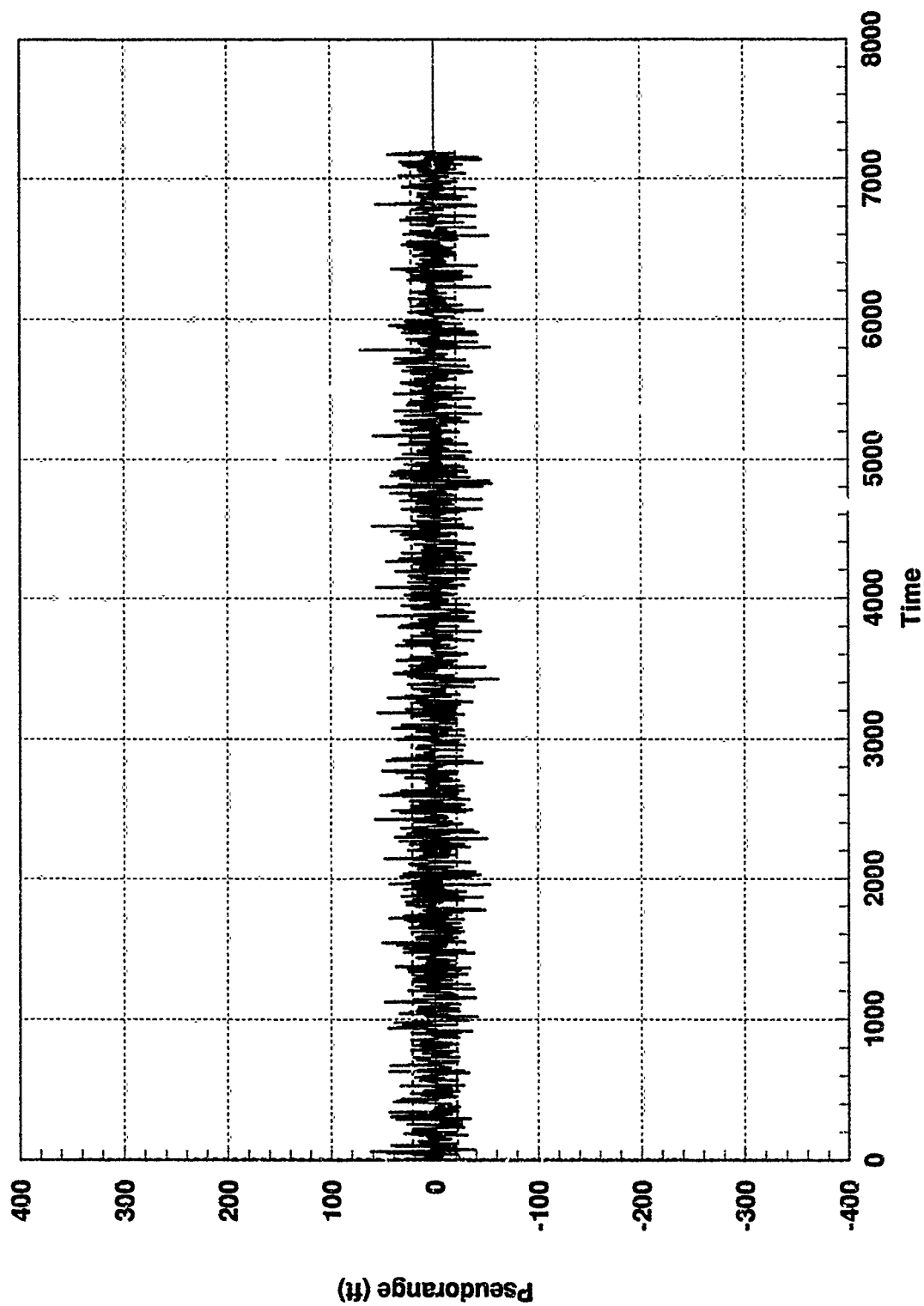
Figure D.3 GPS Sat 2 Residual and One-Sigma Bound, Normal Conditions

D-4
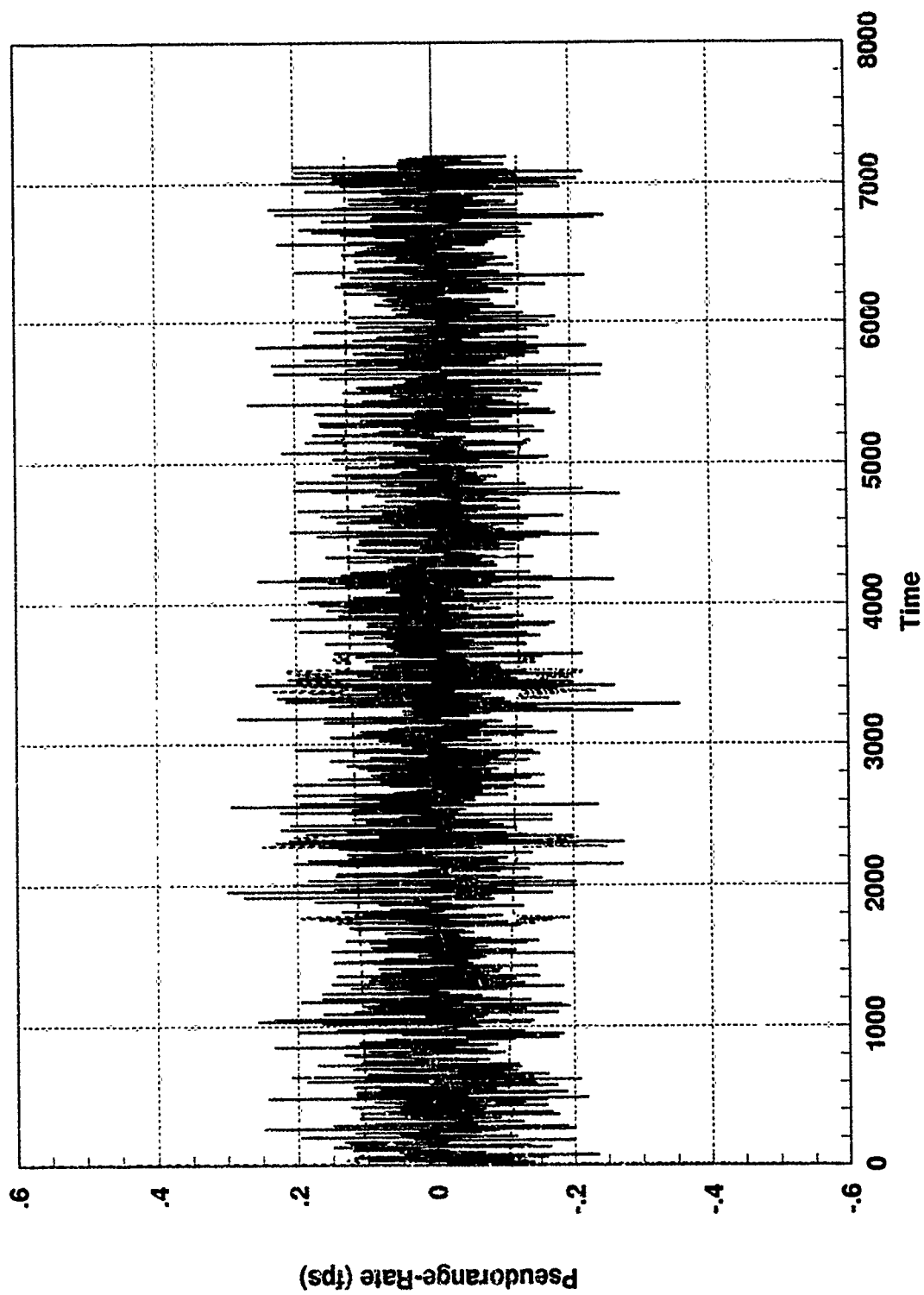
Figure D.4 GPS Sat 2 Residual and One-Sigma Bound, Normal Conditions

D-5
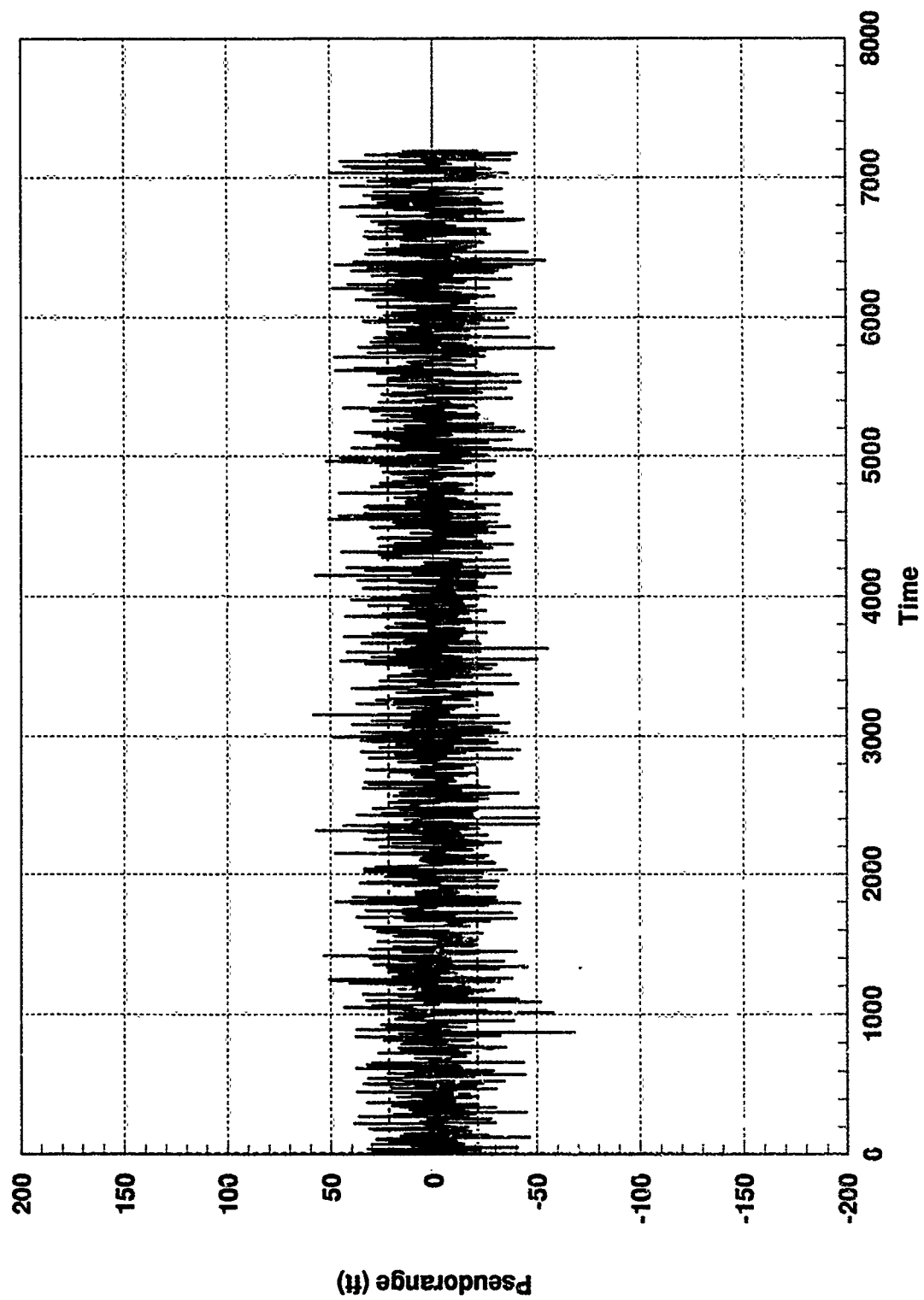
Figure D.5 GPS Sat 3 Residual and One-Sigma Bound, Normal Conditions
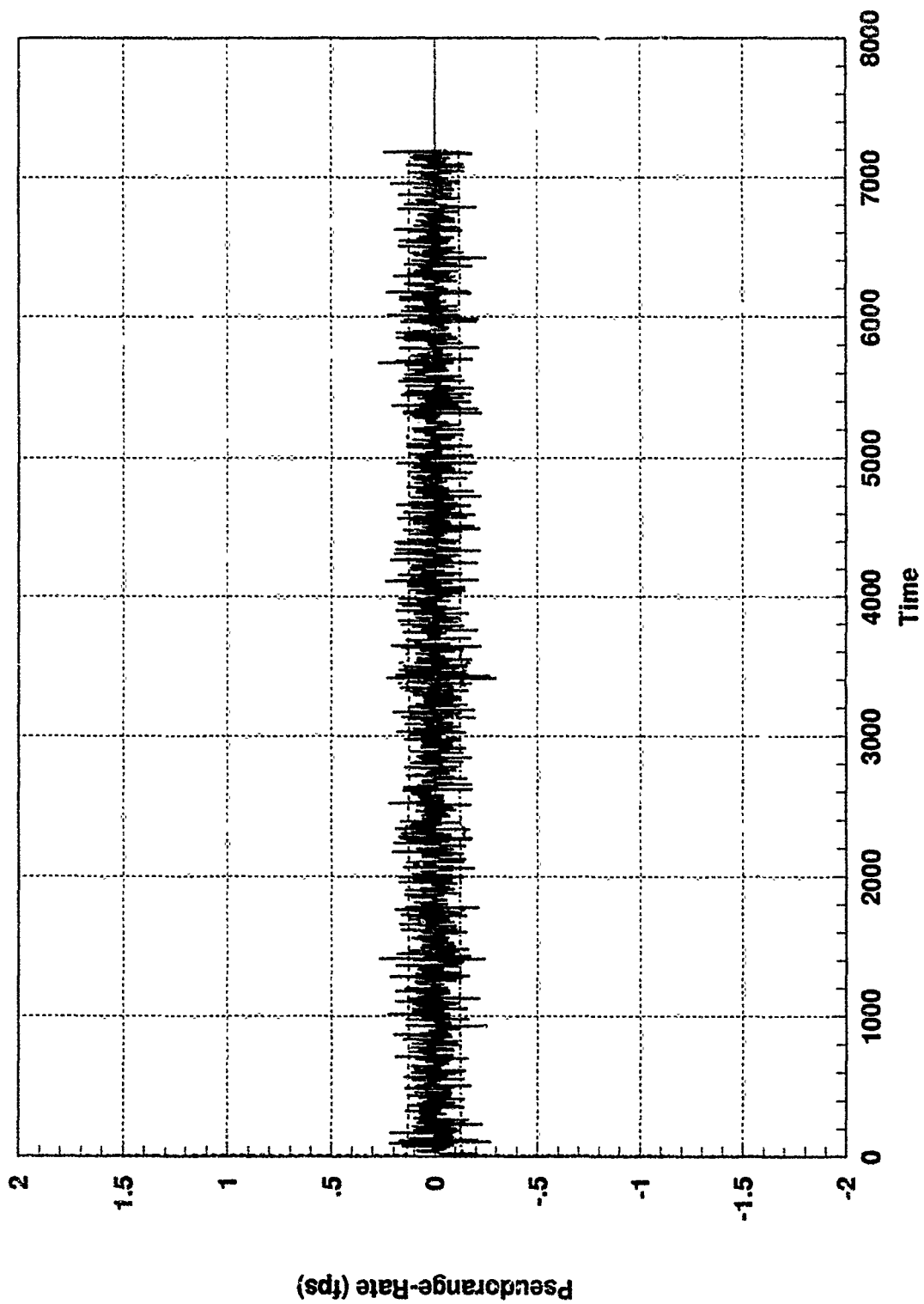
Figure D.6 GPS Sat 3 Residual and One-Sigma Bound, Normal Conditions

D-7
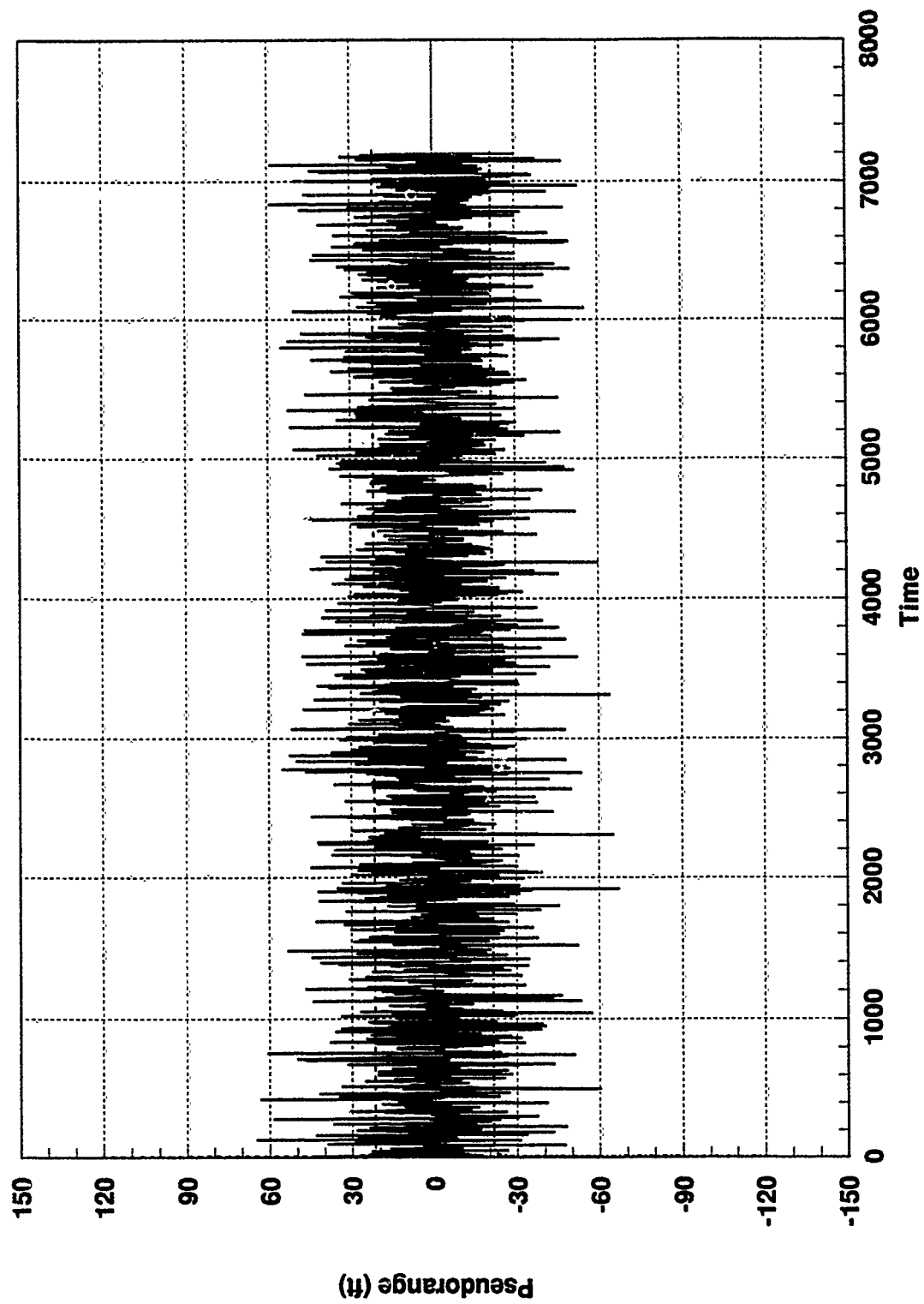
Figure D.7 GPS Sat 4 Residual and One-Sigma Bound, Normal Conditions
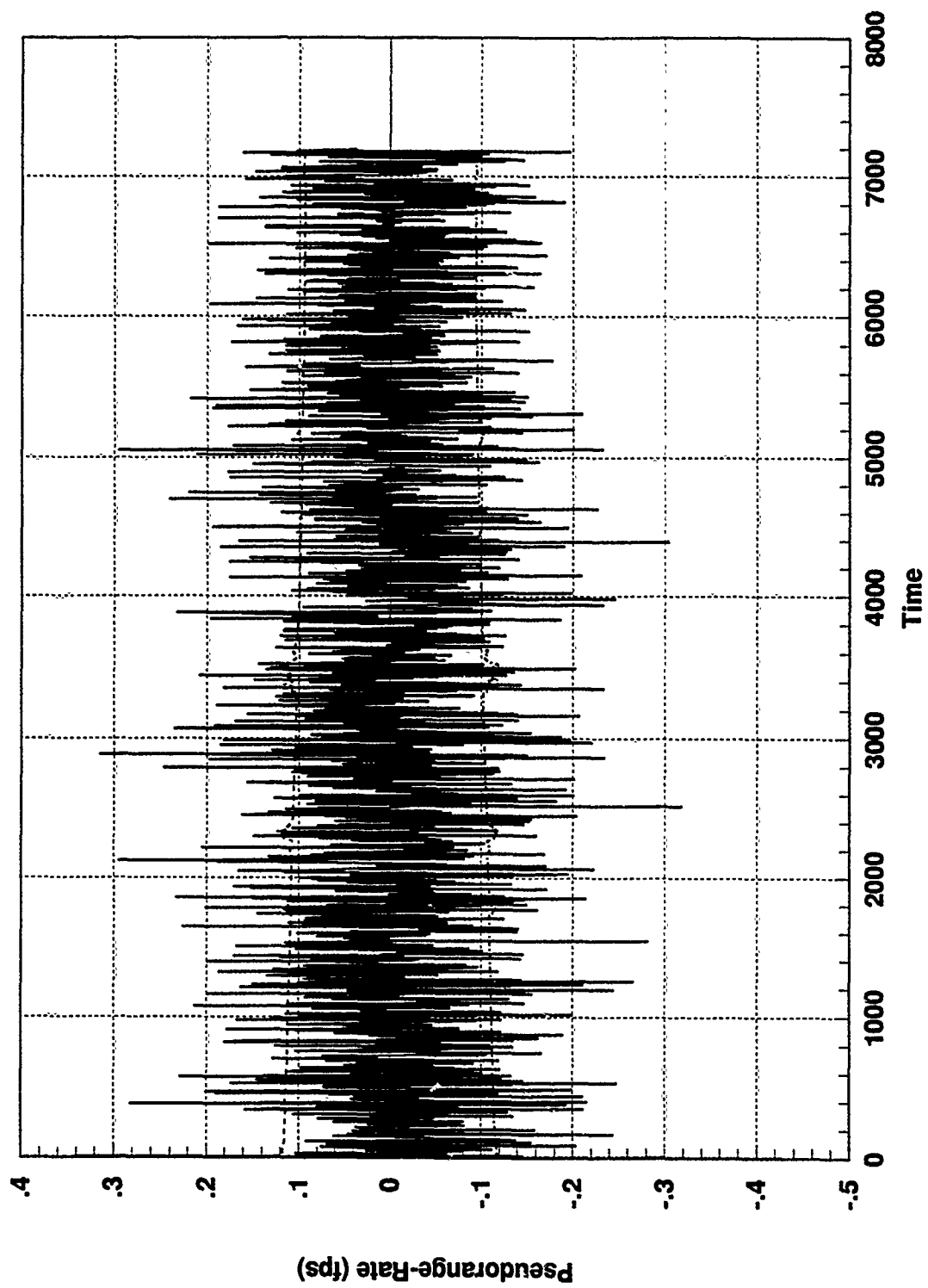
Figure D.8 GPS Sat 4 Residual and One-Sigma Bound, Normal Conditions
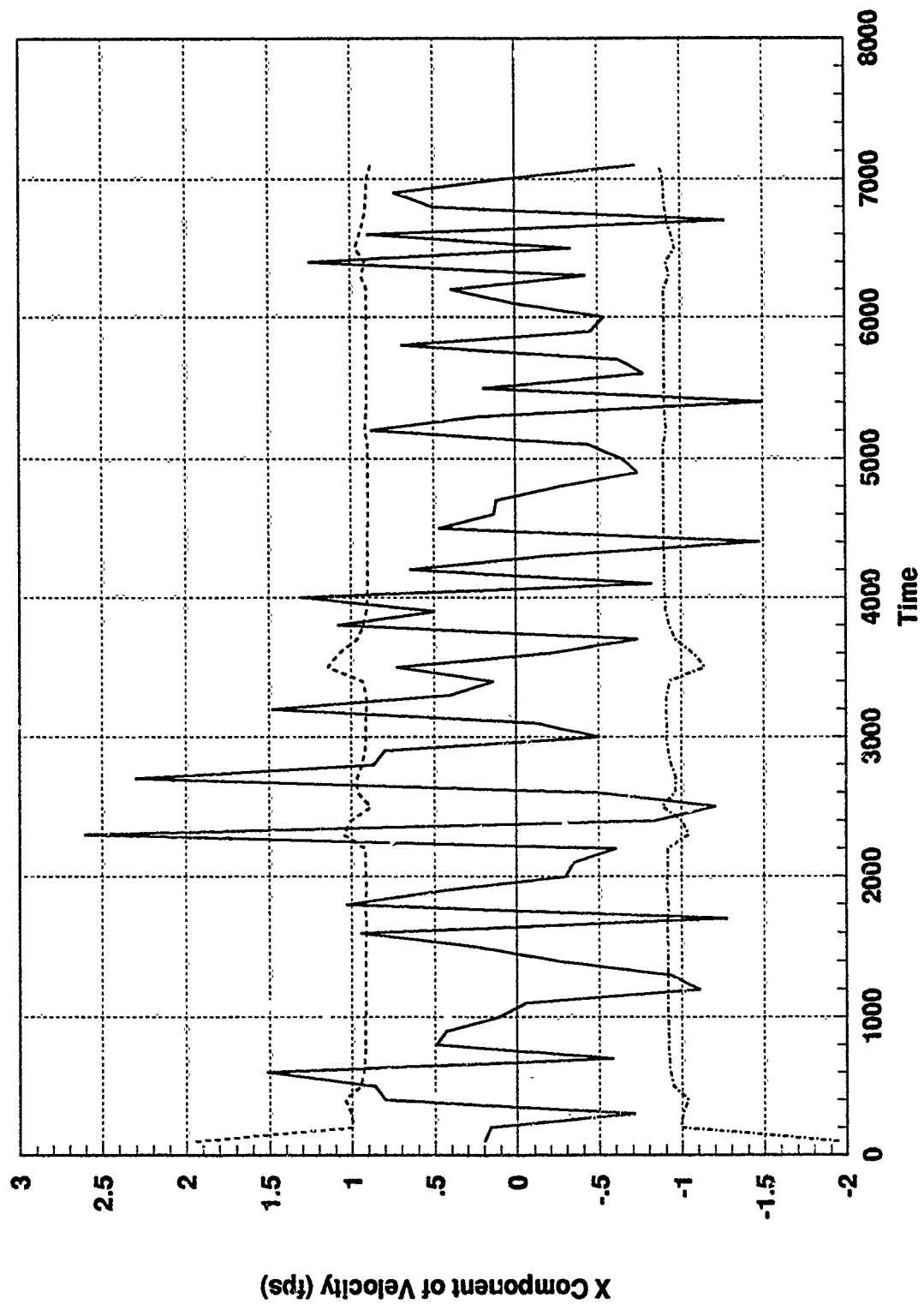
Figure D.9 SARPVU Residual, One-Sigma Bound, Normal Conditions
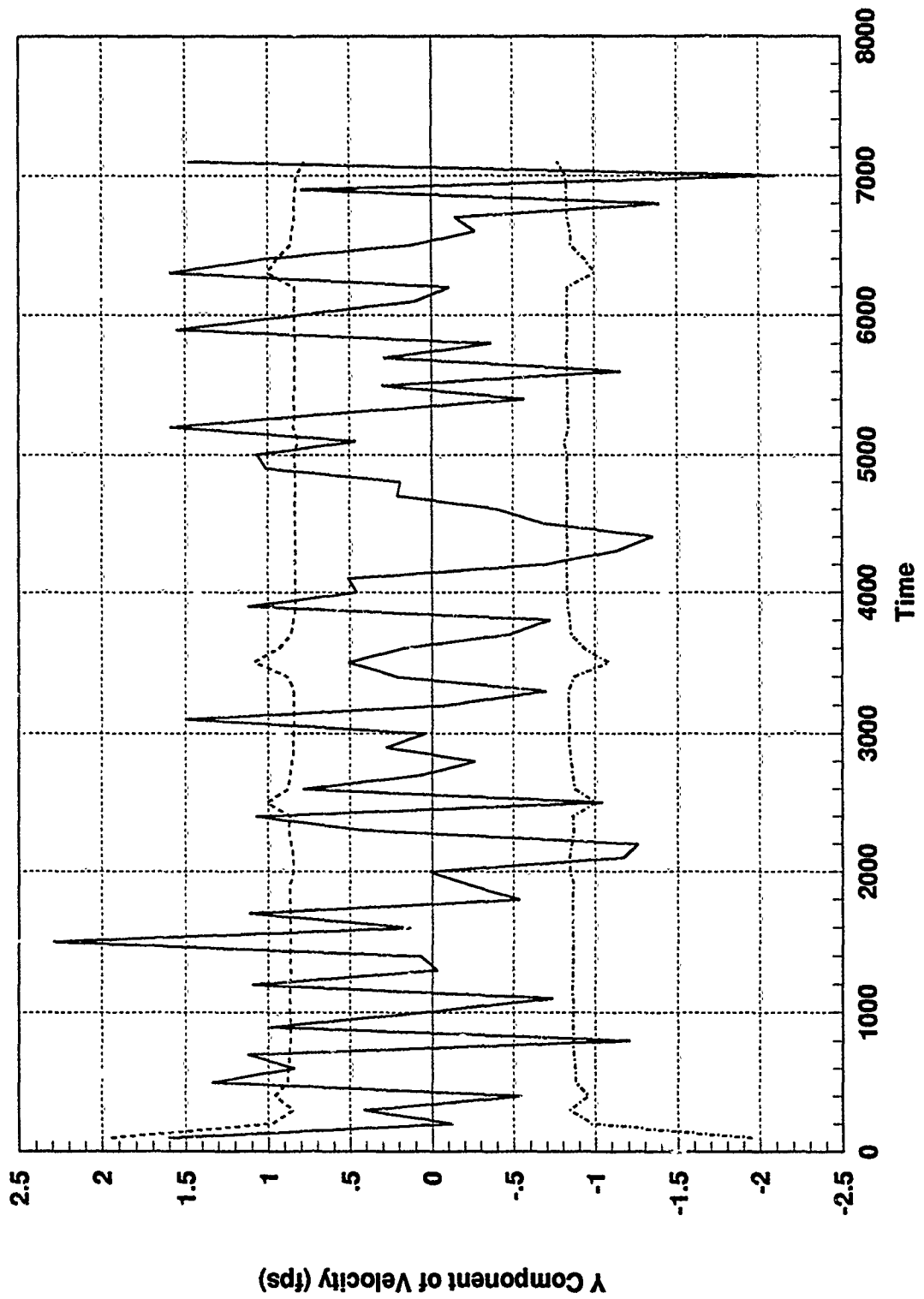
Figure D.10 SARPVU Residual, One-Sigma Bound, Normal Conditions
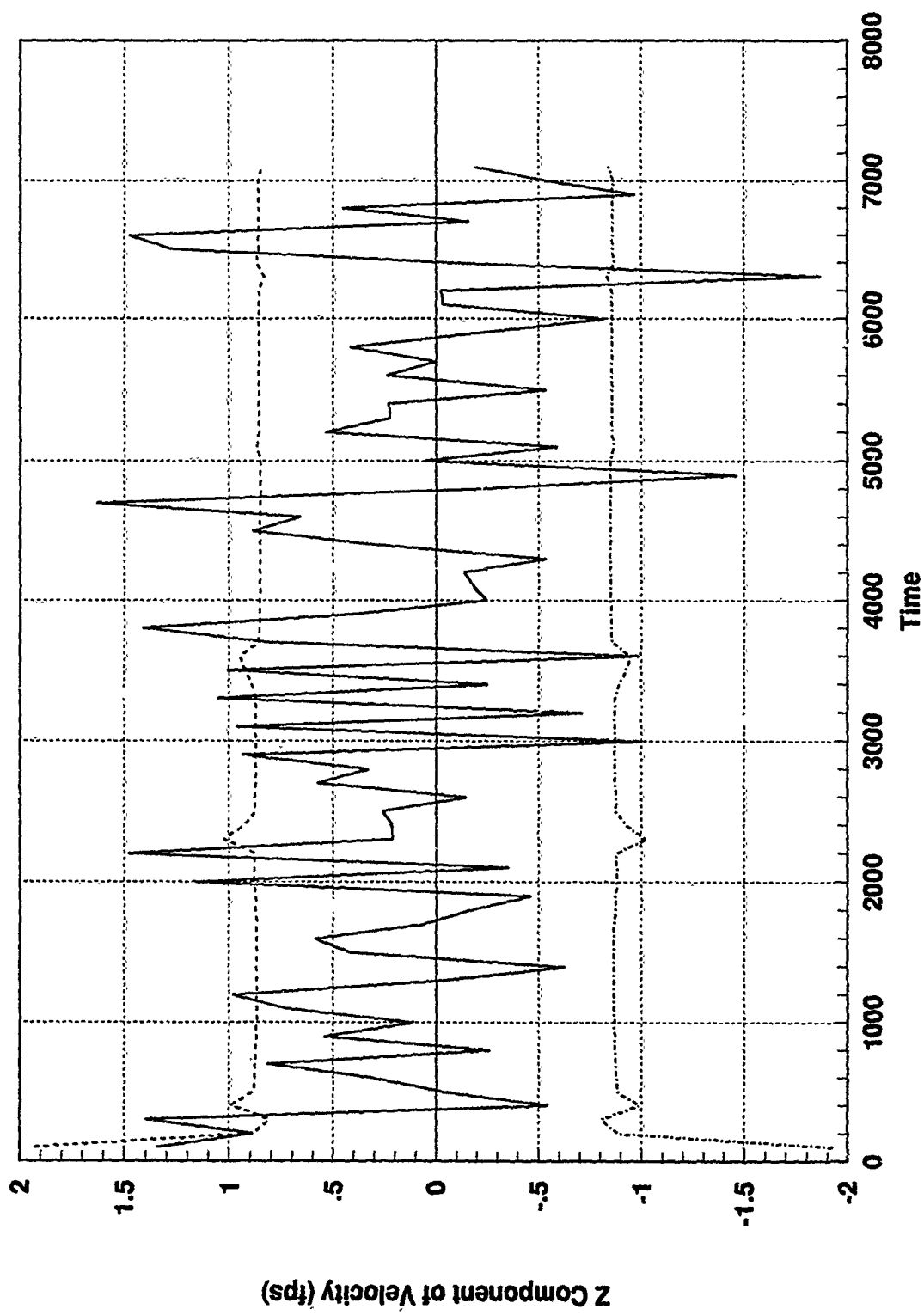
Figure D.11 SARPVU Residual, One-Sigma Bound, Normal Conditions
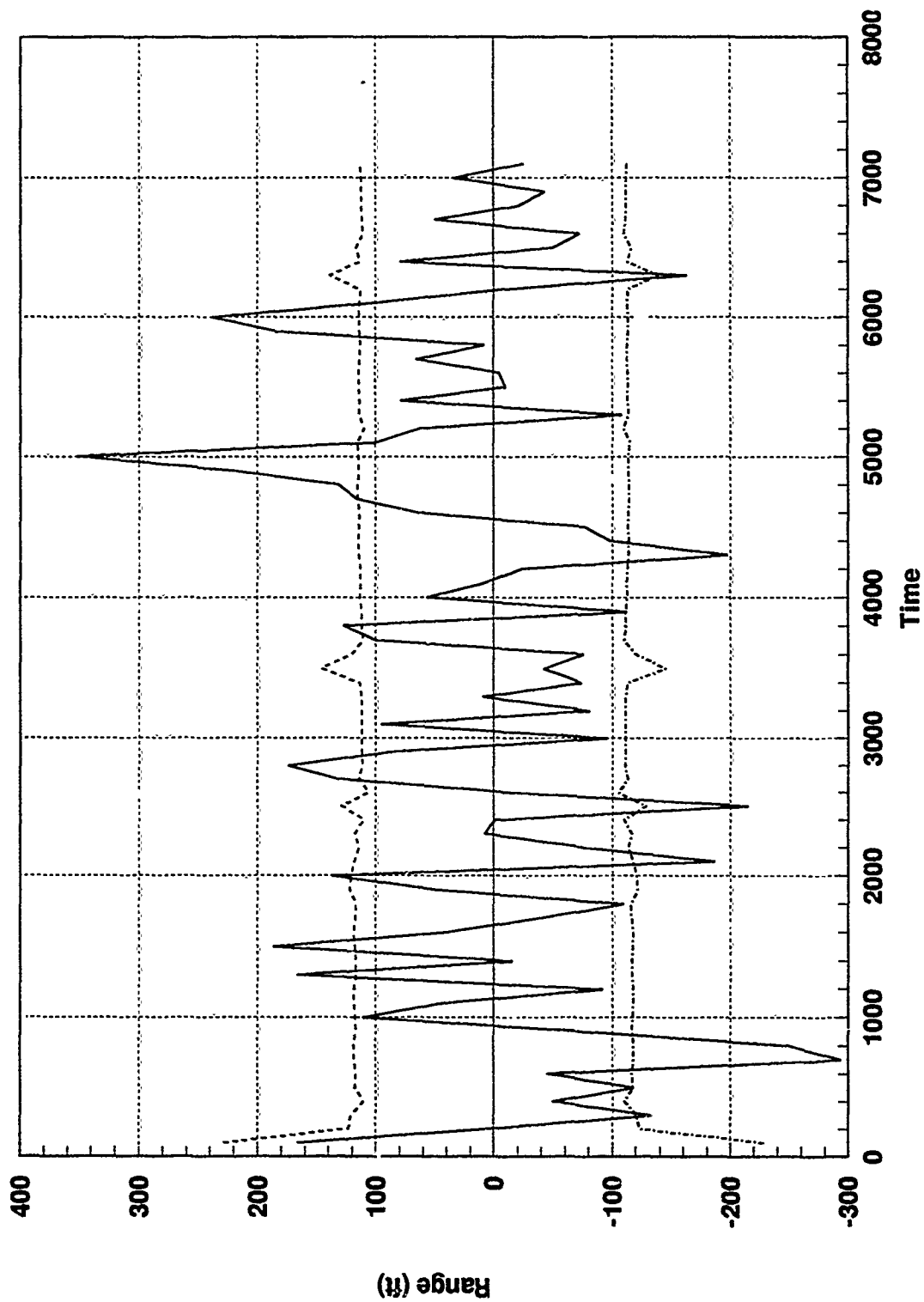
Figure D.12 SAREO Residual, One-Sigma Bound, Normal Conditions
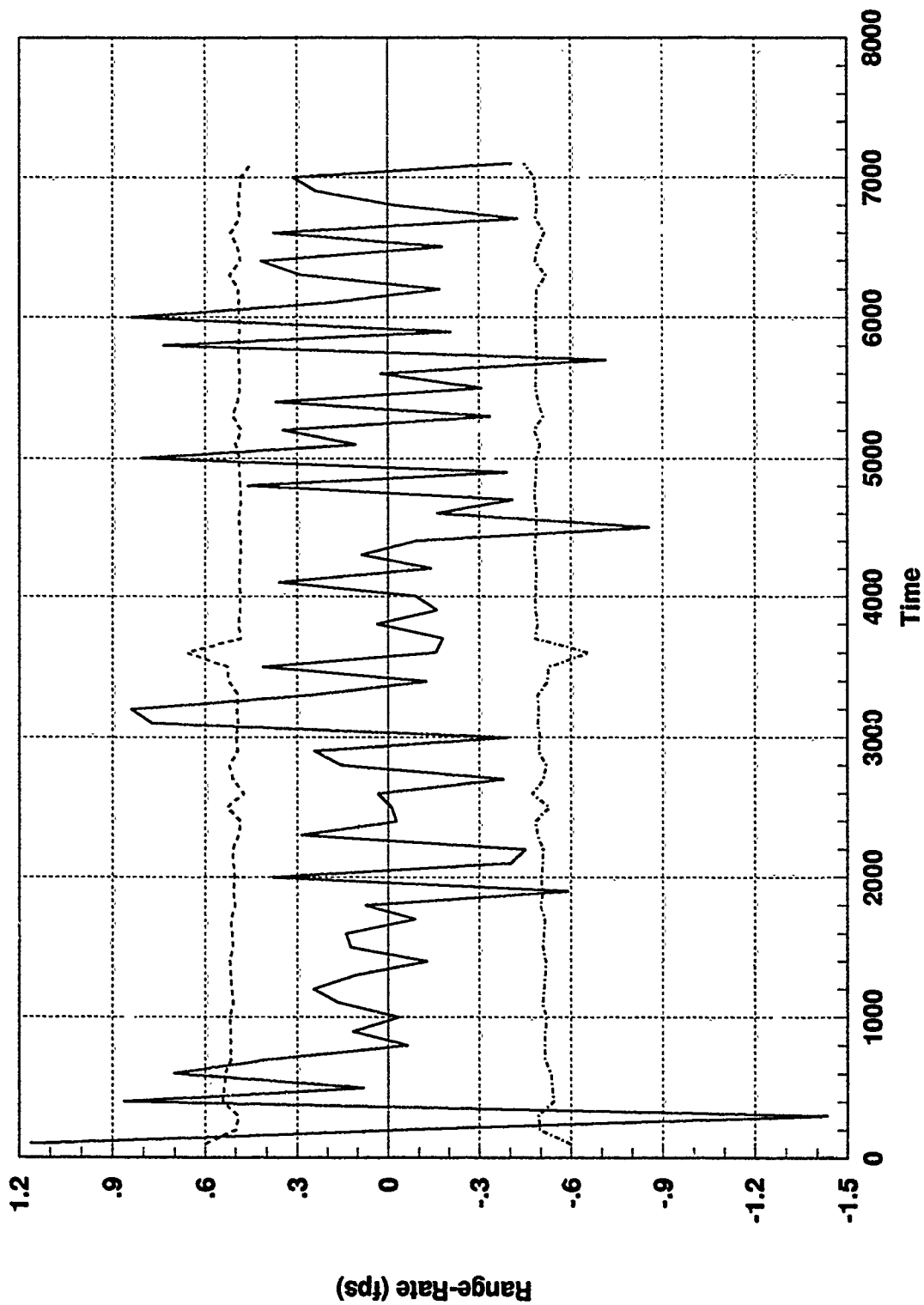
Figure D.13 SAREO Residual, One-Sigma Bound, Normal Conditions
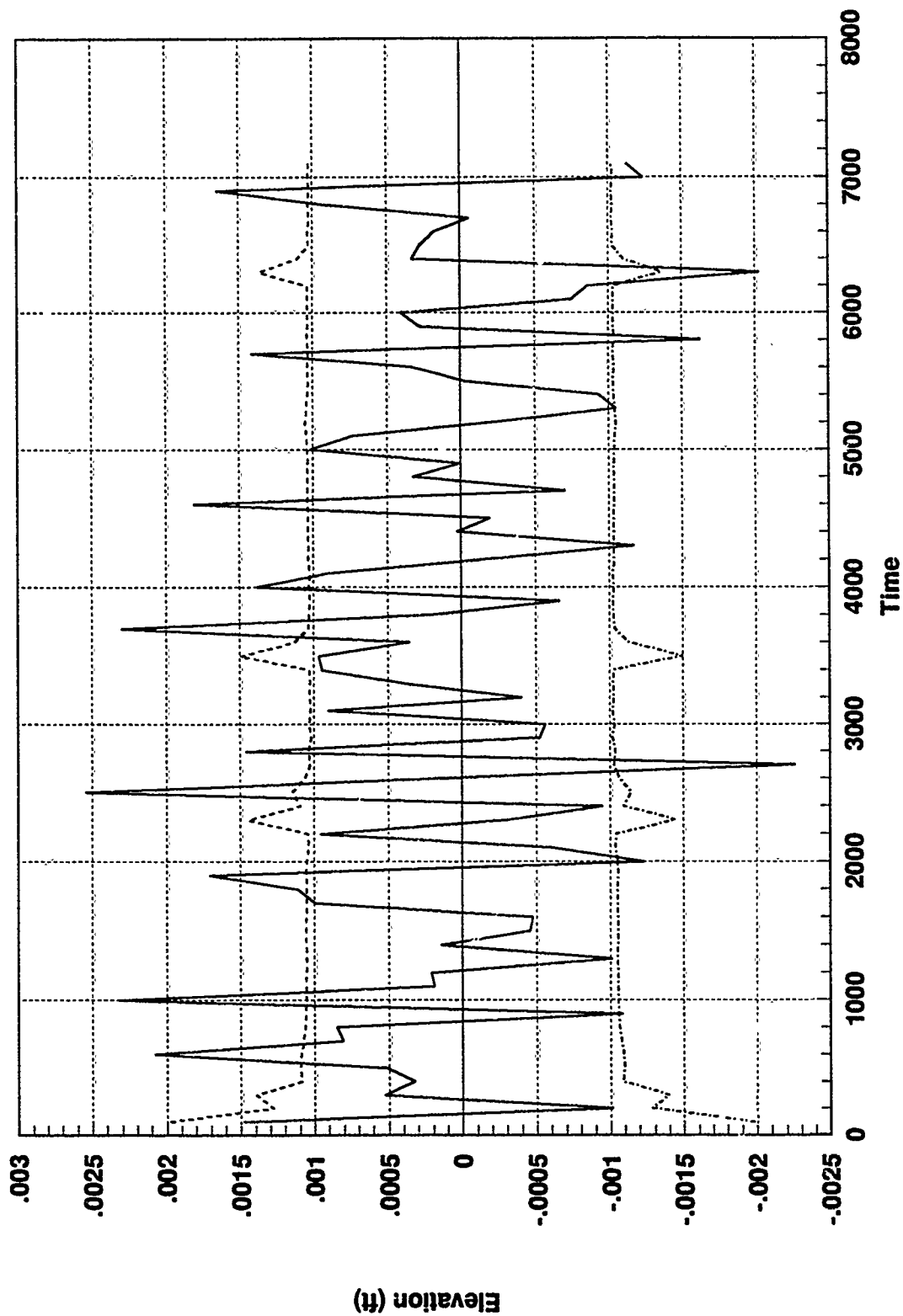
Figure D.14 SAREO Residual, One-Sigma Bound, Normal Conditions
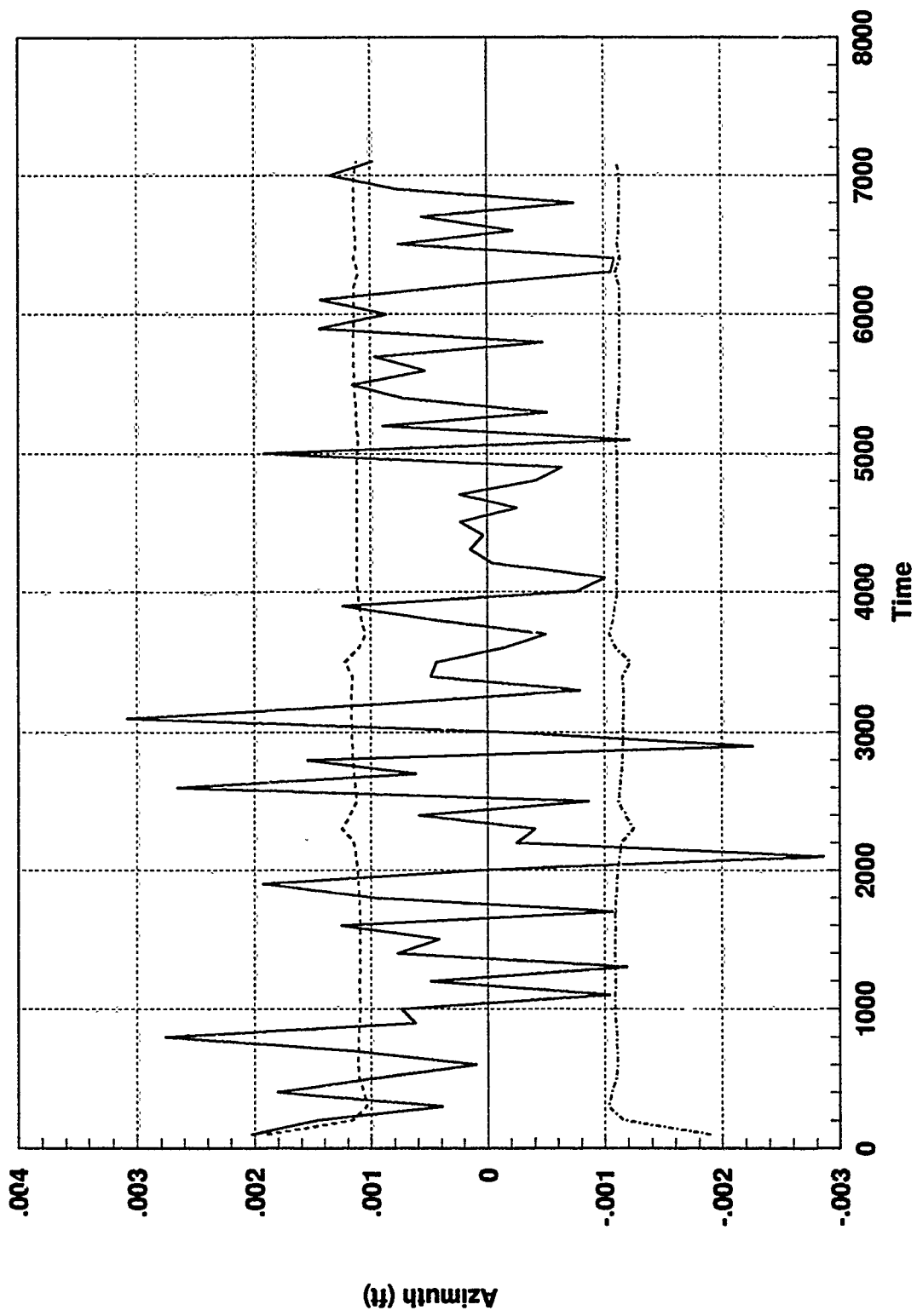
Figure D.15 SAREO Residual, One-Sigma Bound, Normal Conditions

D-16
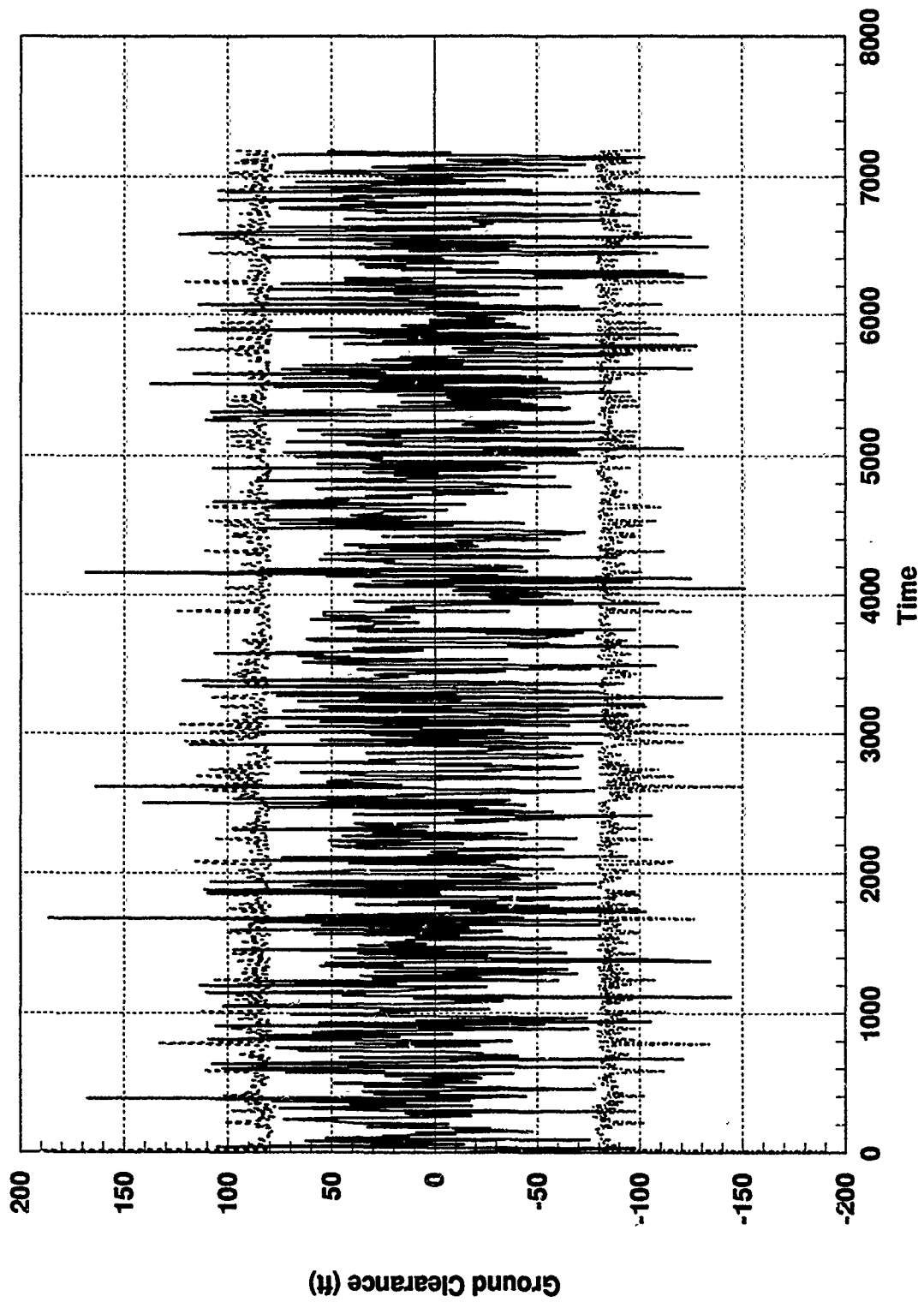
Figure D.16 Federated Filter TAN Residual, One-Sigma Bound, Normal Conditions

Appendix E: Centralized Filter Residual Plots

Soft Failure - Constant Bias

GENERAL INFORMATION

The series of plots in this appendix allows for a comparison of the fault detection considerations of this thesis. Establishing the specific results for the centralized filter residual behavior during simulated failed operating conditions with the failure represented as a constant bias of reasonable magnitude added directly to the residual computation from 1000 to 3500 seconds on the GPS Satellite #1, PseudoRange-Rate residual, and a constant bias added directly to the residual computation from 4500 to 6500 seconds on the TAN Ground Clearance residual.

Each of the residual plots contained in this appendix are obtained from Monte Carlo simulations using DKFSIM Version 1.1. All residual outputs are recorded for the entire 7200 second duration of the simulation. The residual outputs and the upper and lower one-sigma bounds were computed for one Monte Carlo simulation.

There are 16 plots, 8 for the GPS, 7 for the SAR, and one for the TAN sensor measurements. Each plot has the appropriate residual description. A listing of the titles of each these figures is found in the List of Figures at the beginning of this document.

Figure E.1 GPS Sat 1 Residual and One-Sigma Bound, GPS & TAN Constant Bias

Figure E.2 GPS Sat 1 Residual and One-Sigma Bound, GPS & TAN Constant Bias
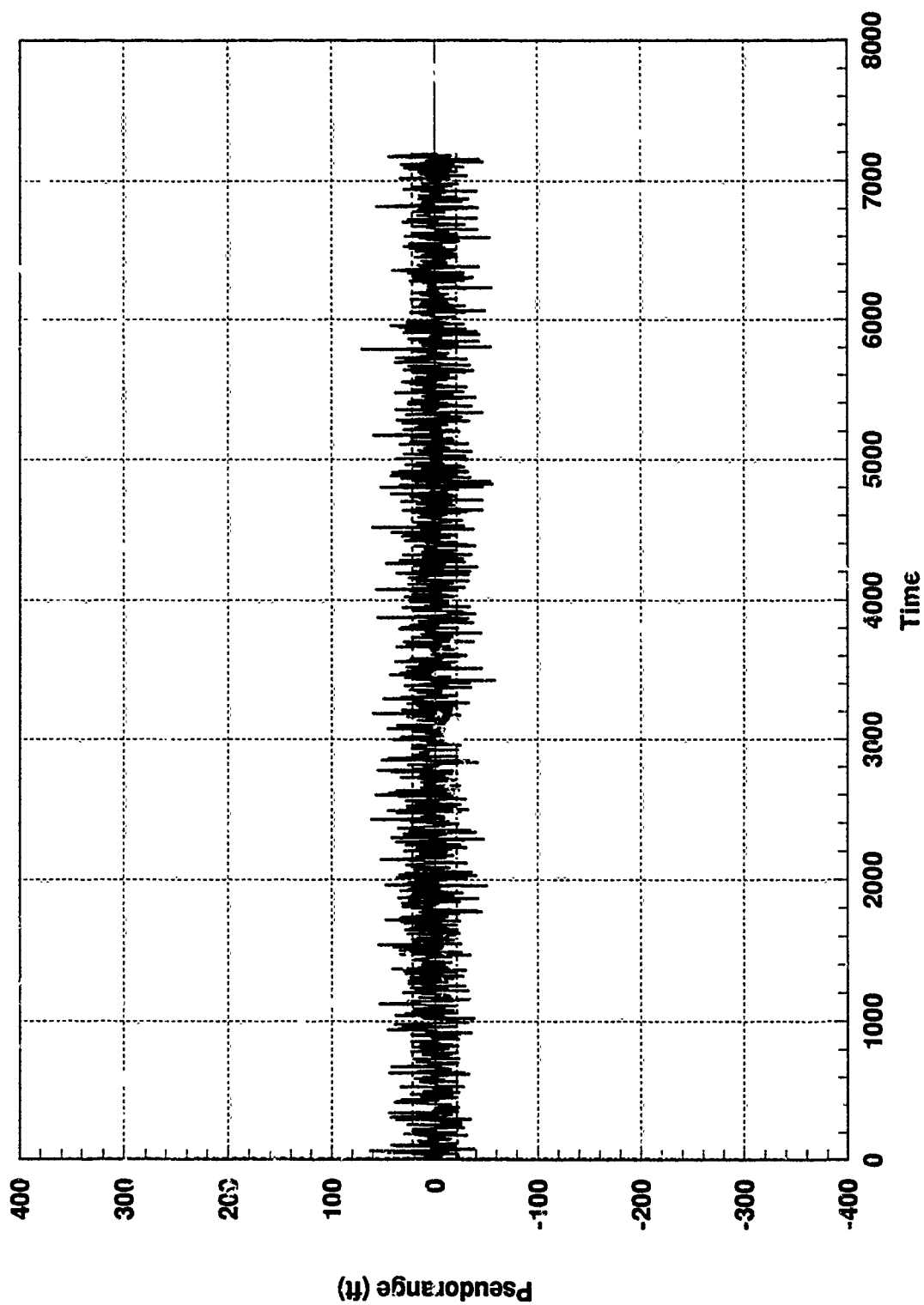
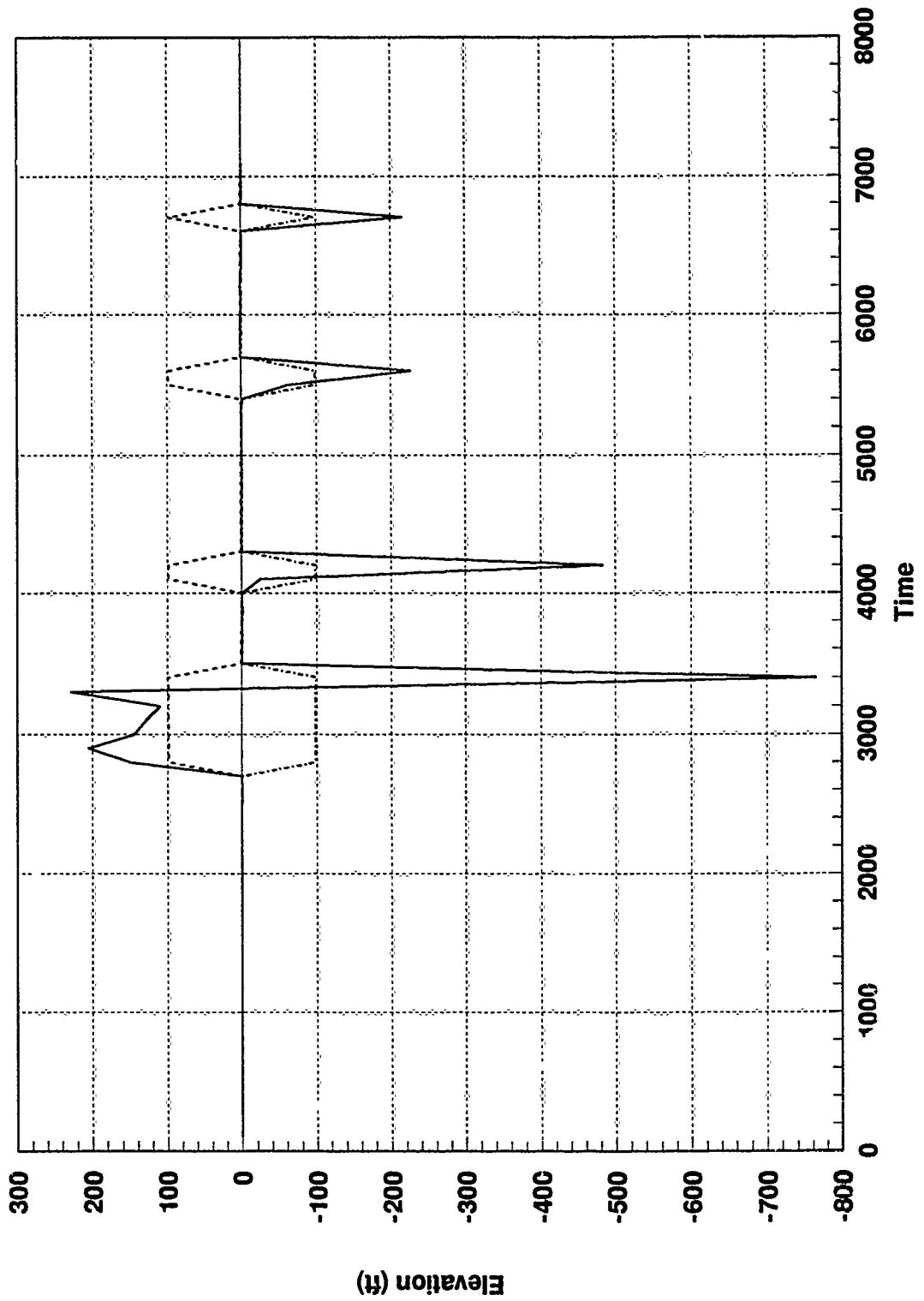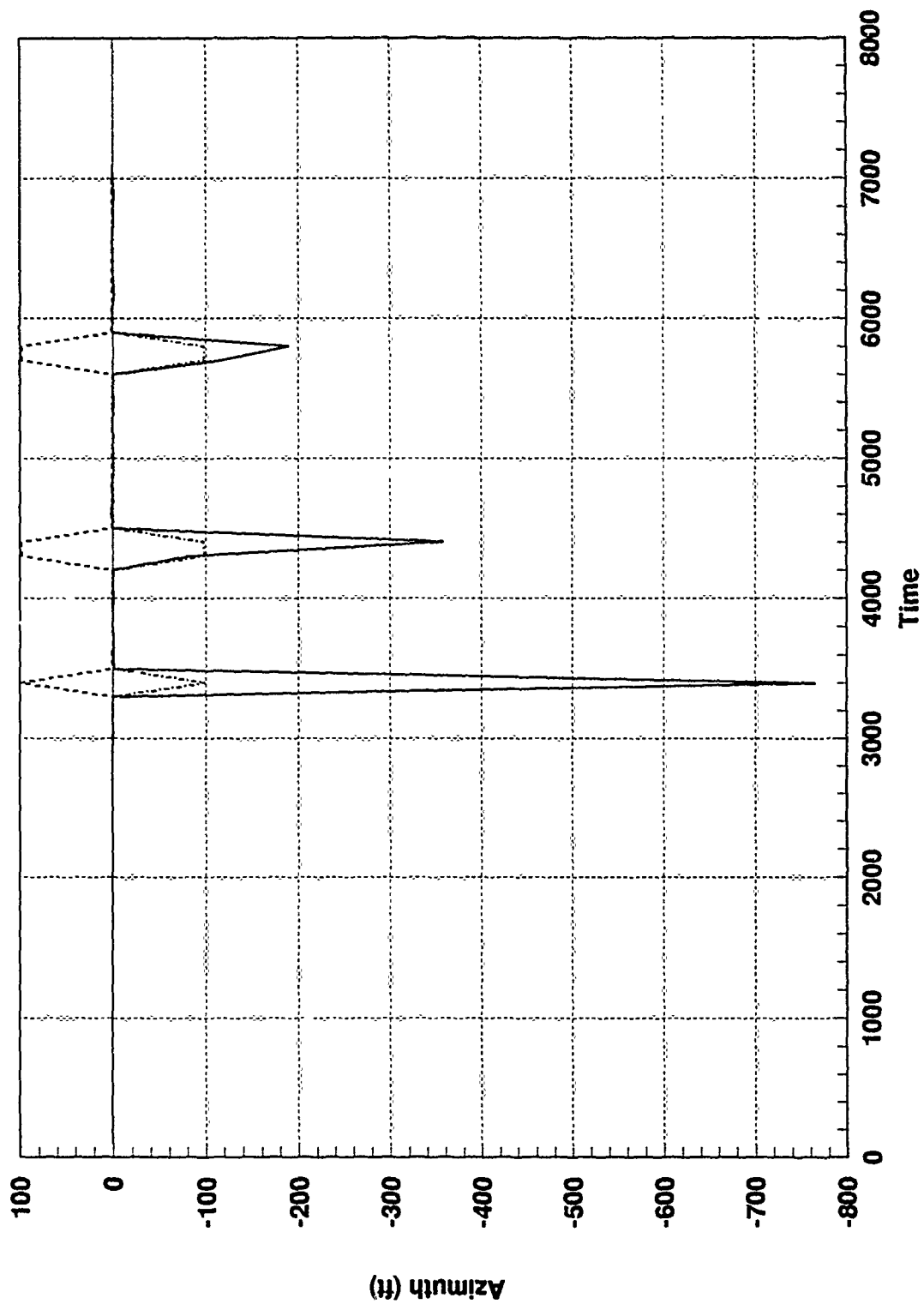Figure E.5 GPS Sat 2 Residual and One-Sigma Bound, GPS & TAN Constant Bias
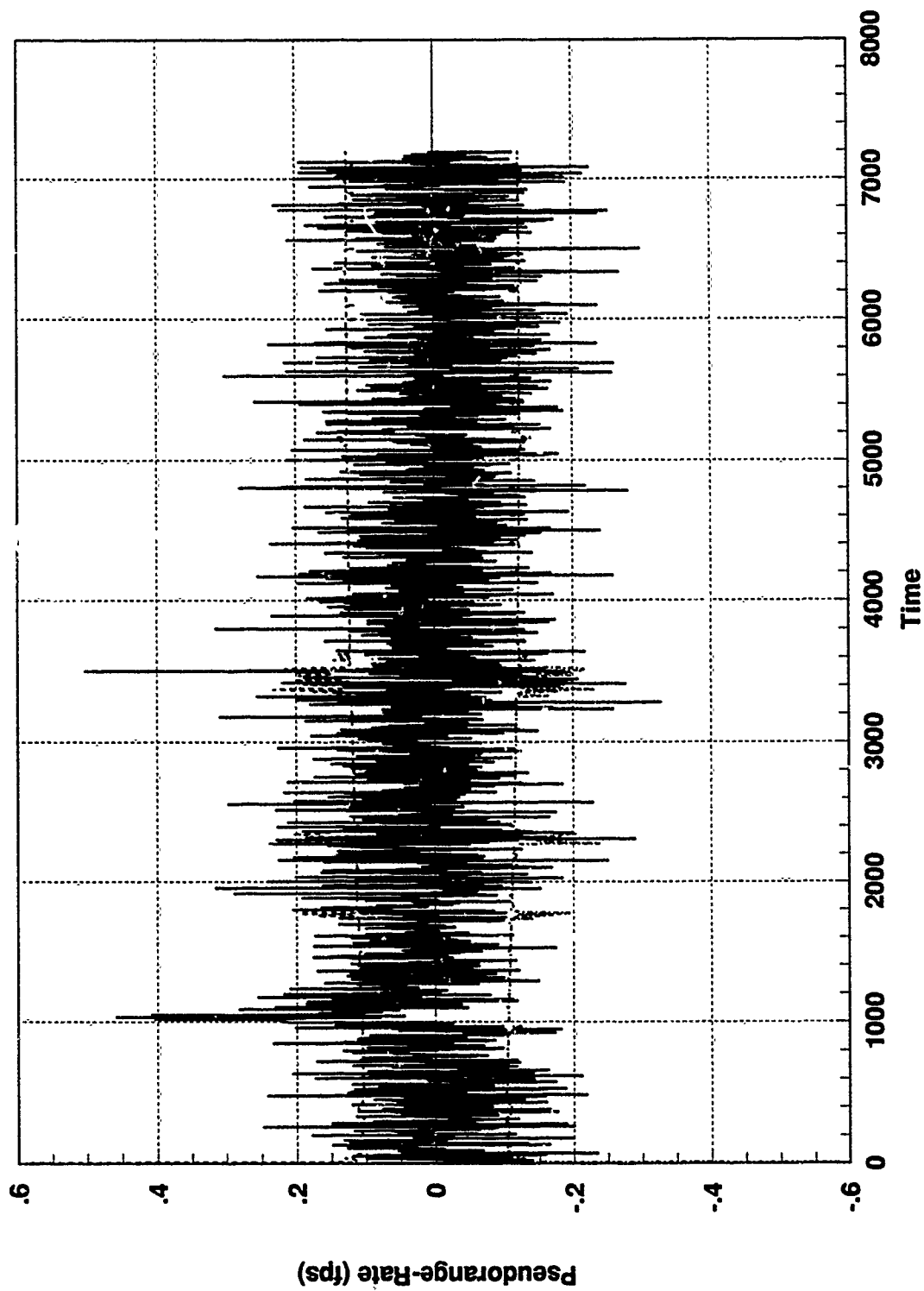
Figure E.4 GPS Sat 2 Residual and One-Sigma Bound, GPS & TAN Constant Bias
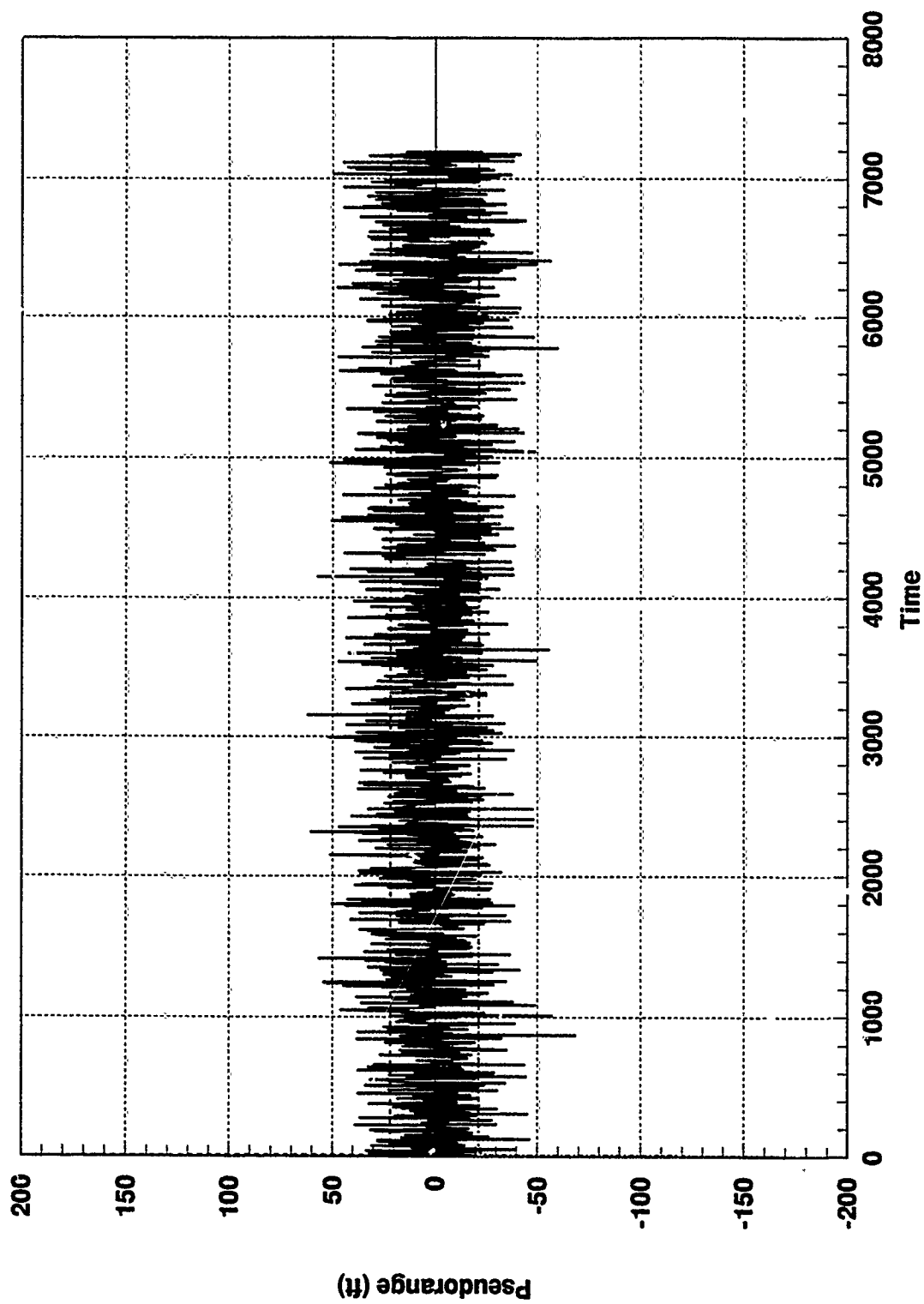
Figure E.5 GPS Sat 3 Residual and One-Sigma Bound, GPS & TAN Constant Bias
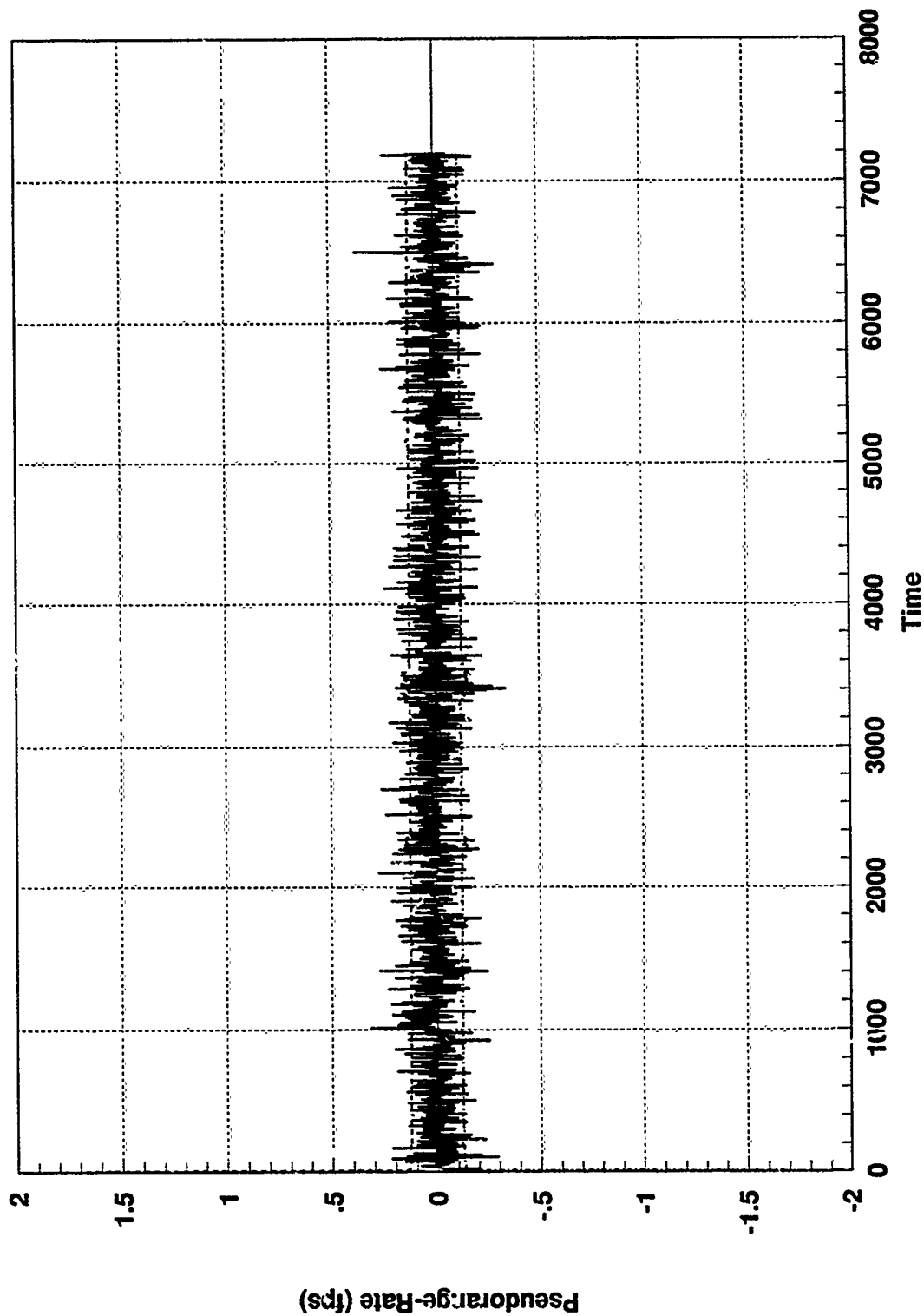
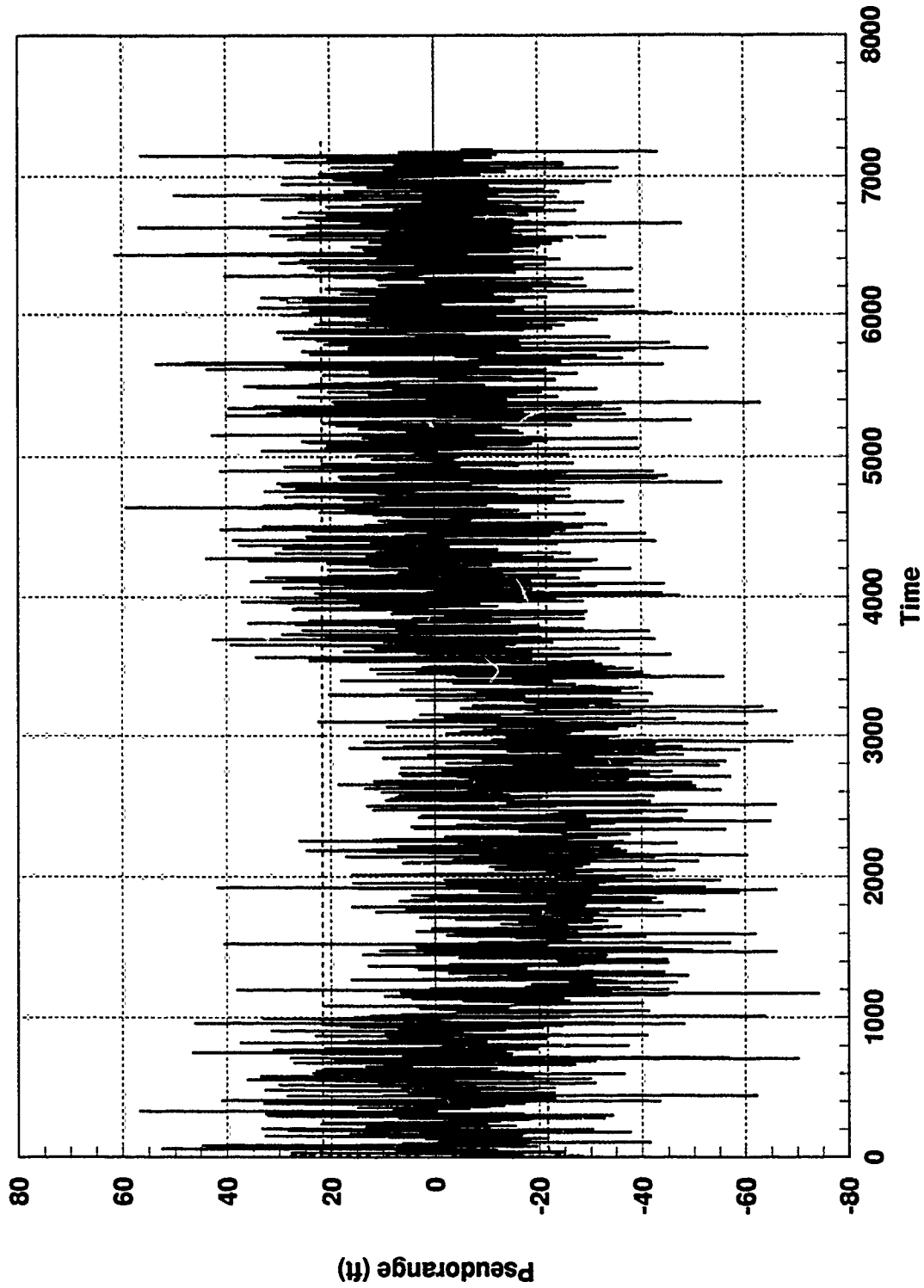Figure E.6 GPS Sat 3 Residual and One-Sigma Bound, GPS & TAN Constant Bias
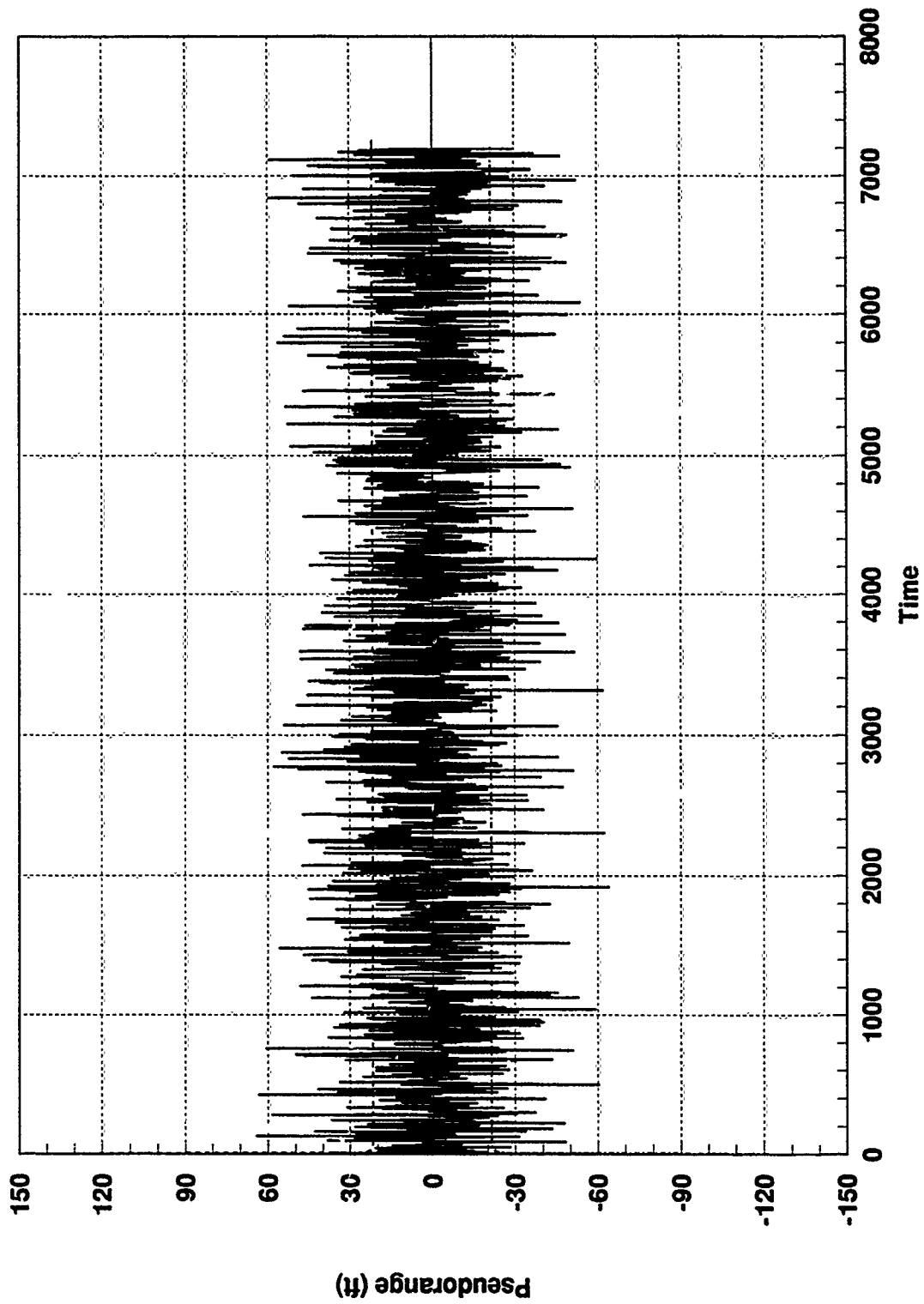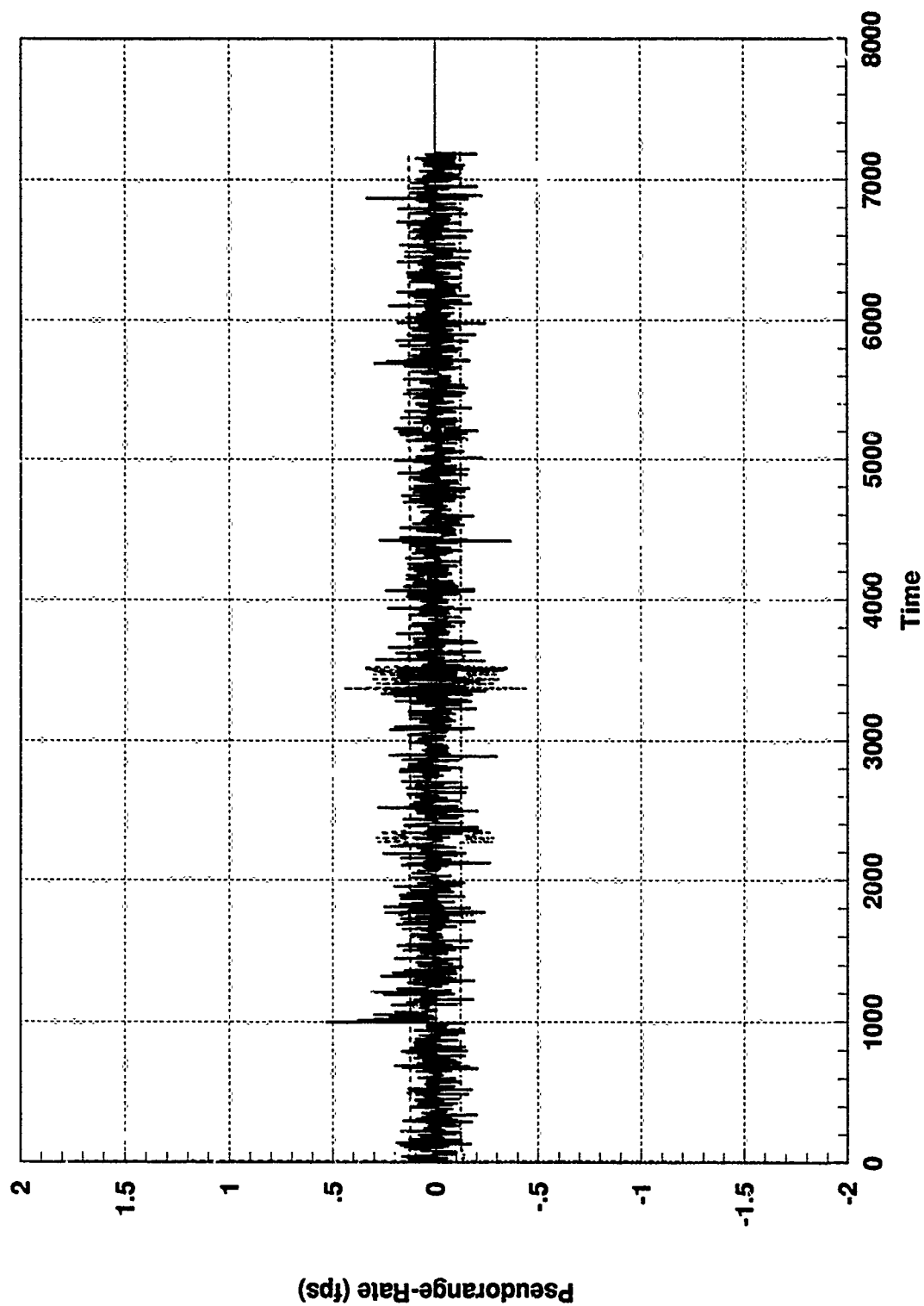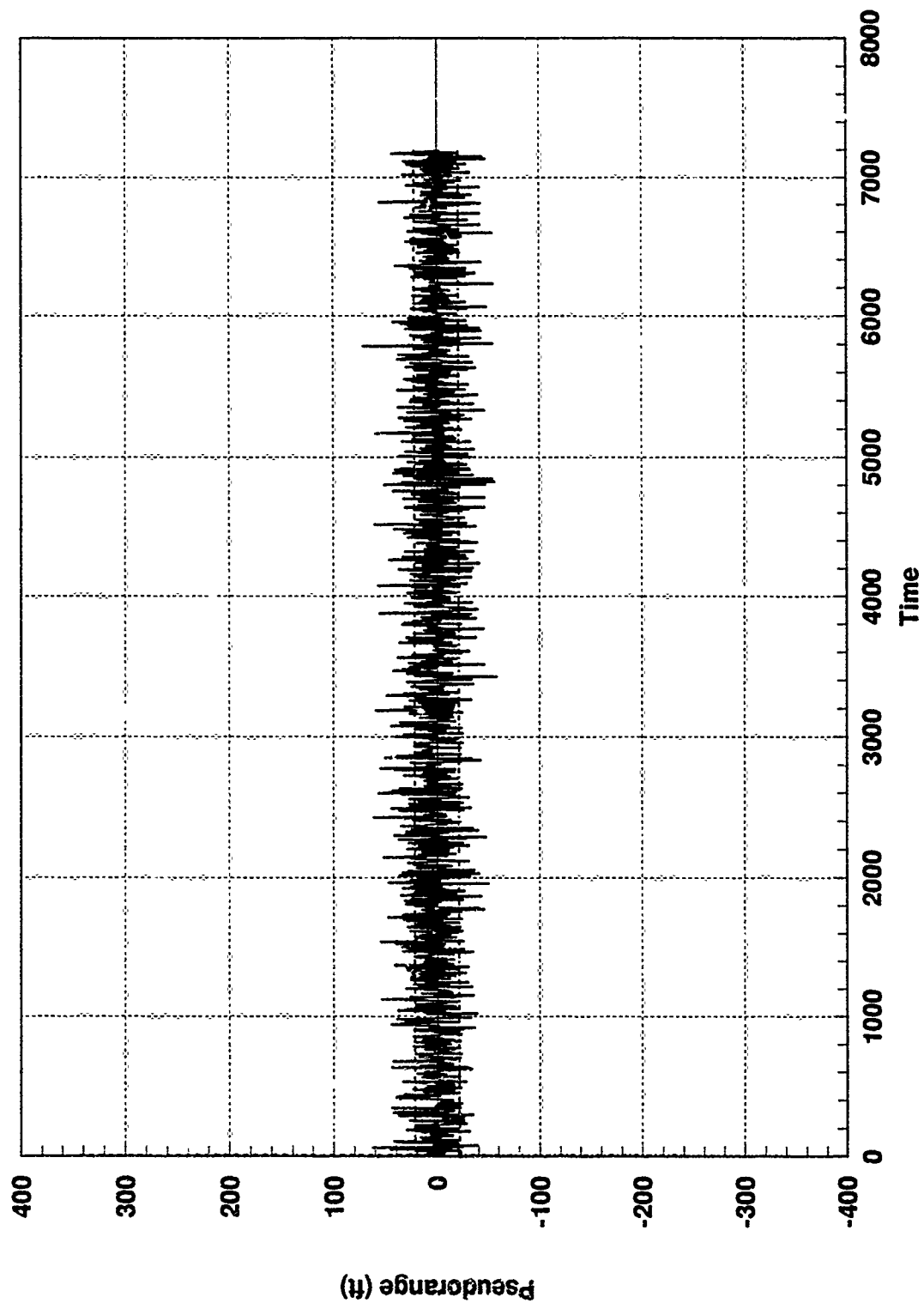
Figure E.7 GPS Sat 4 Residual and One-Sigma Bound, GPS & TAN Constant Bias
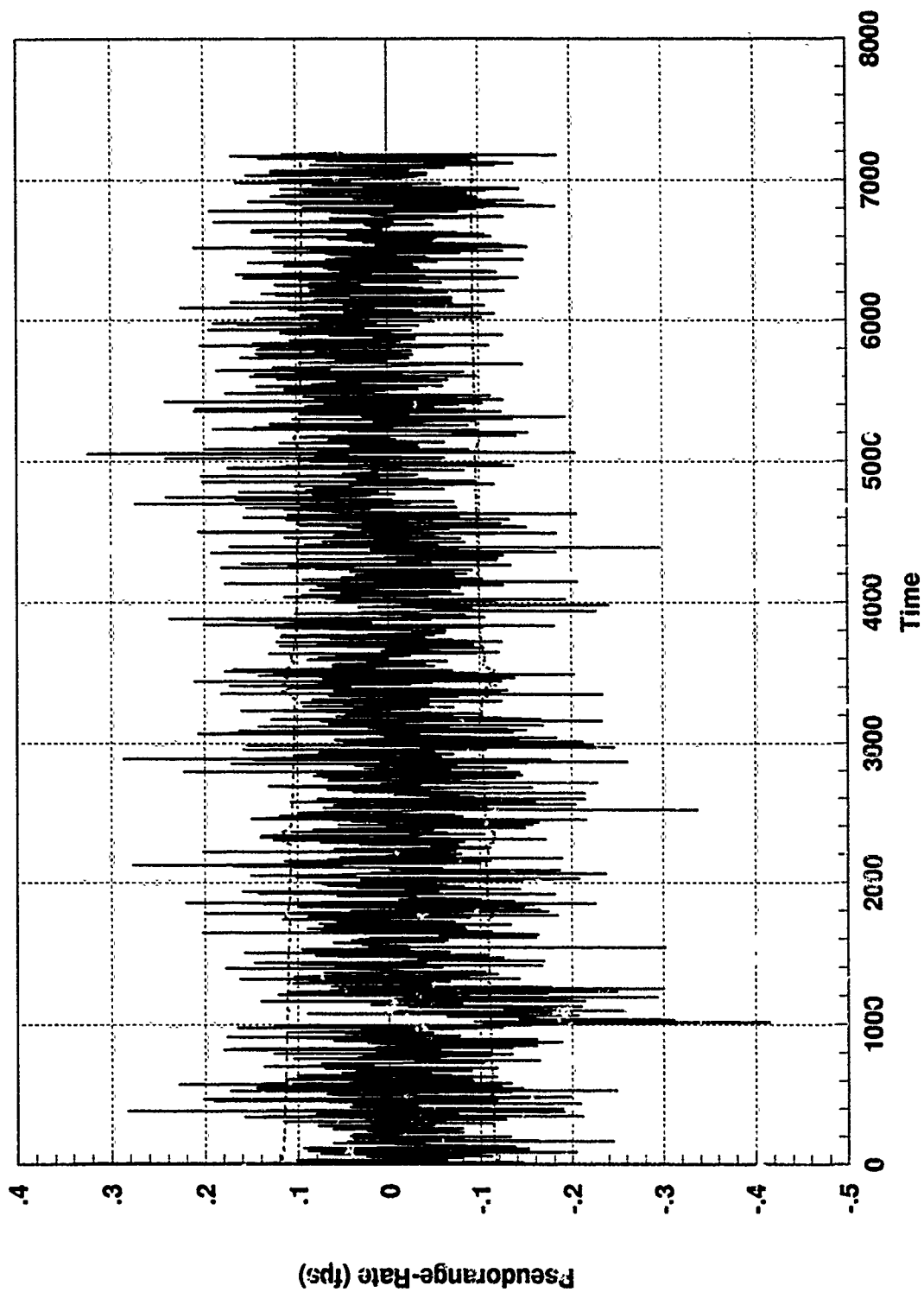
Figure E.8 GPS Sat 4 Residual and One-Sigma Bound, GPS & TAN Constant Bias
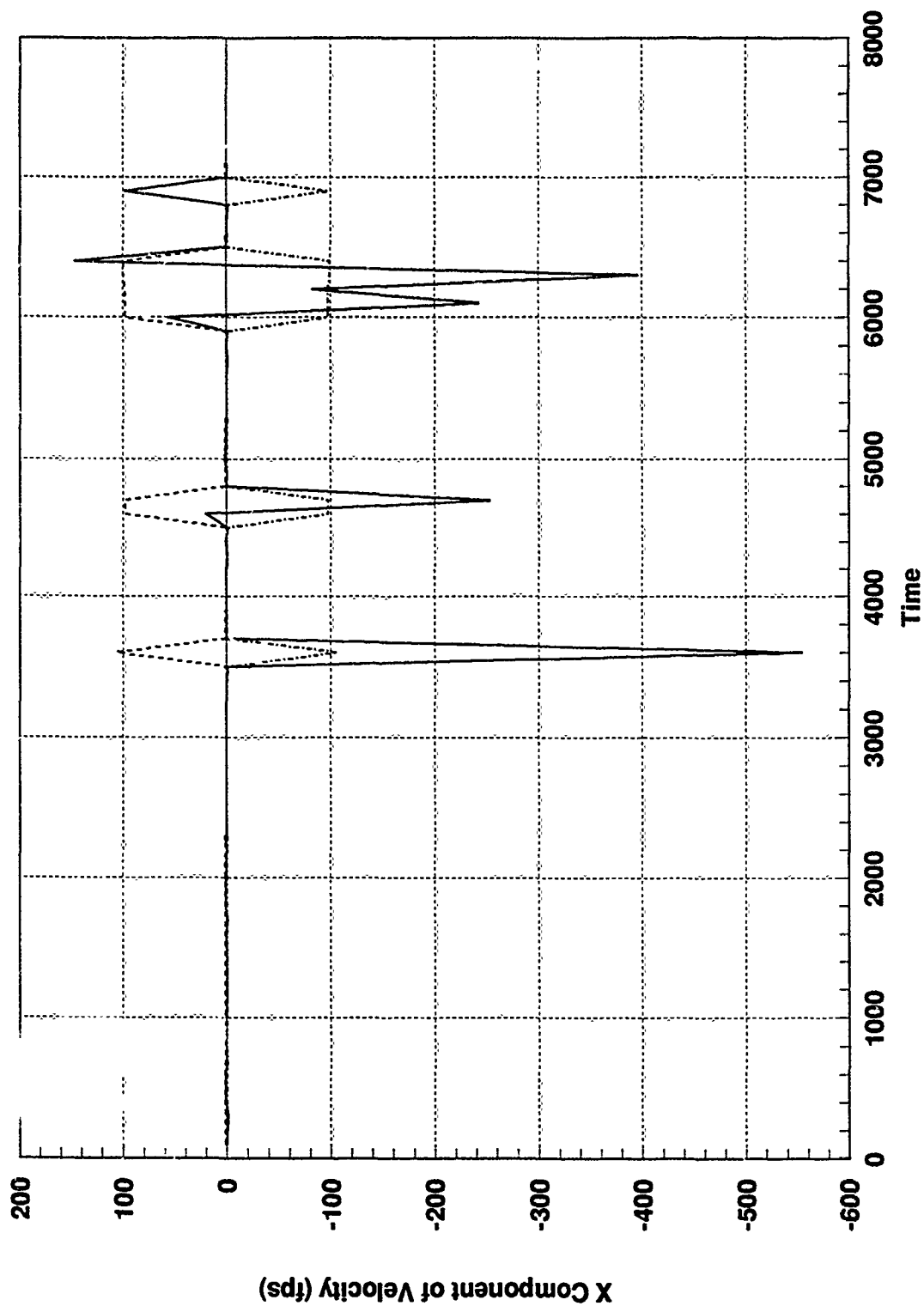
Figure E.9 SARPVU Residual and One-Sigma Bound, GPS & TAN Constant Bias
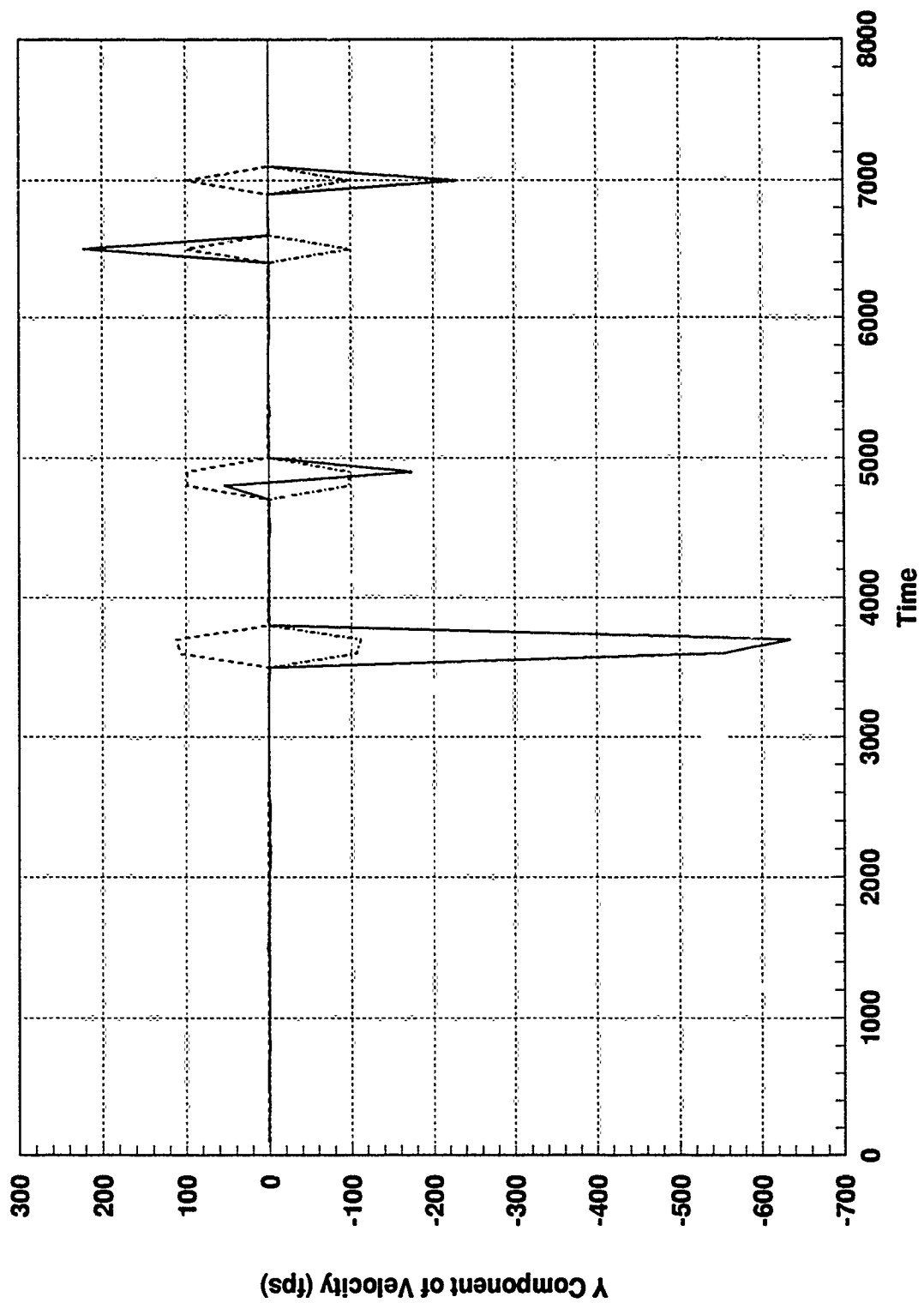
Figure E.10 SARPVU Residual and One-Sigma Bound, GPS & TAN Constant Bias
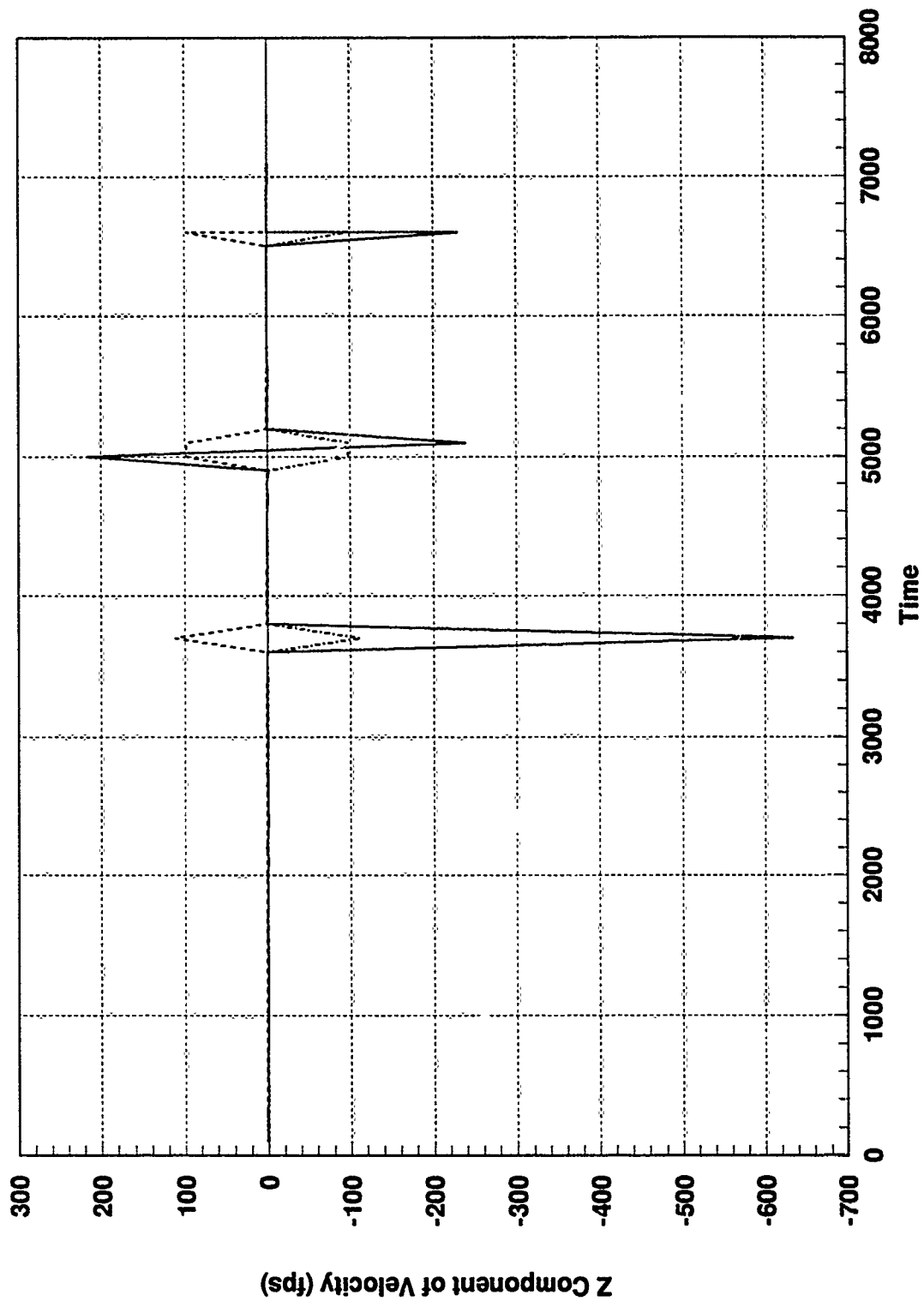
Figure E.11 SARPVU Residual and One-Sigma Bound, GPS & TAN Constant Bias
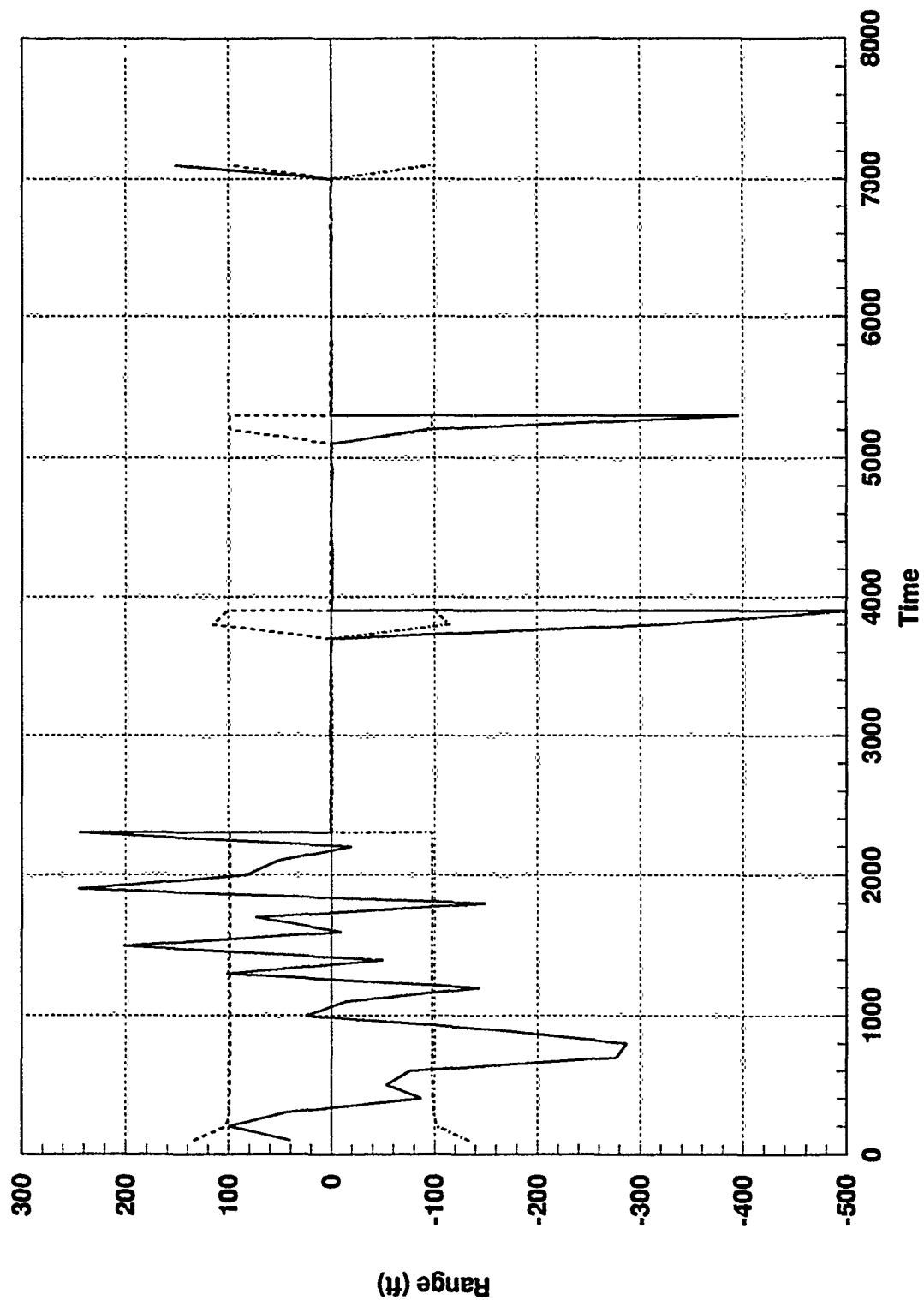
Figure E.12 SAREO Residual and One-Sigma Bound, GPS & TAN Constant Bias
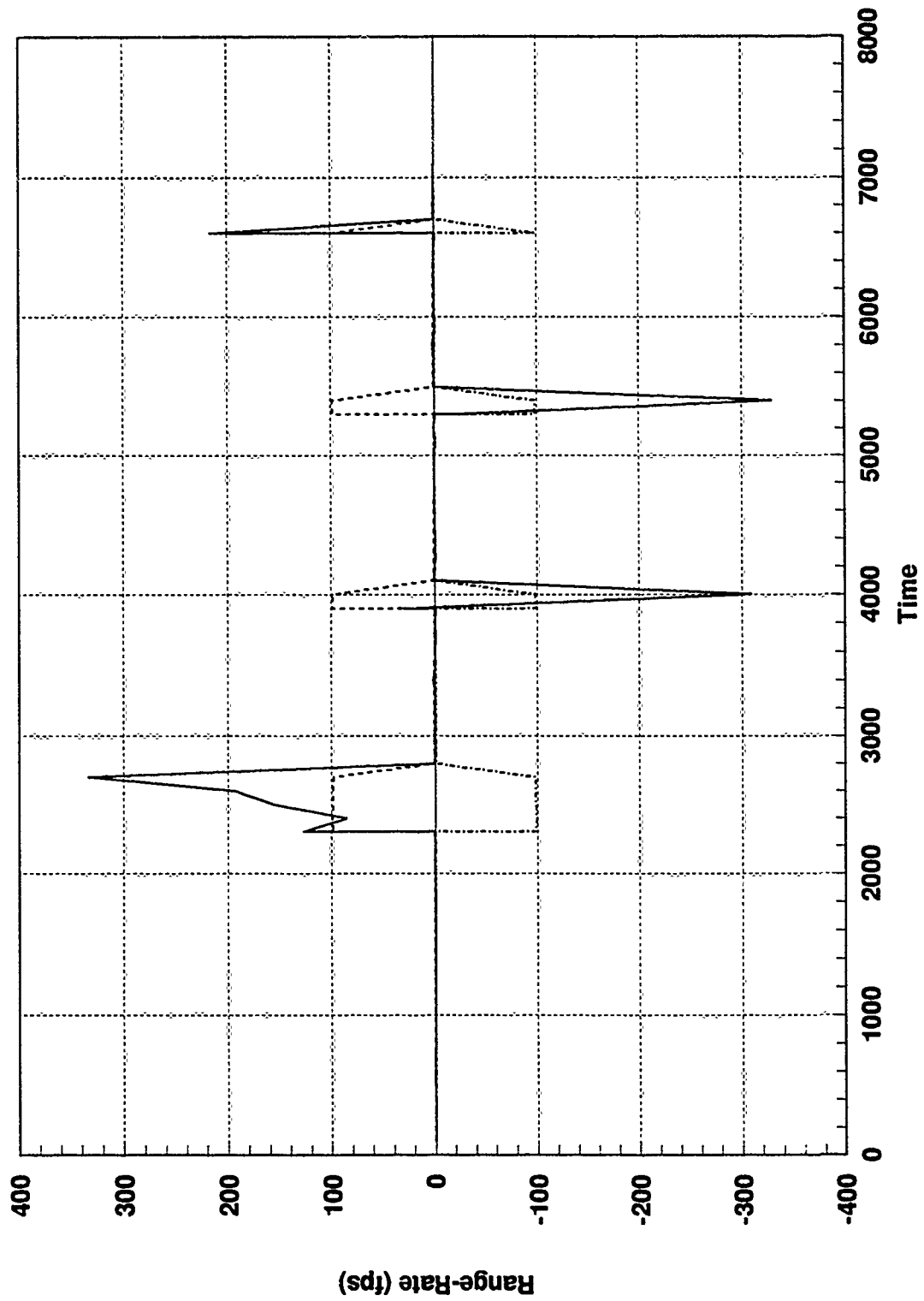
Figure E.13 SAREO Residual and One-Sigma Bound, GPS & TAN Constant Bias
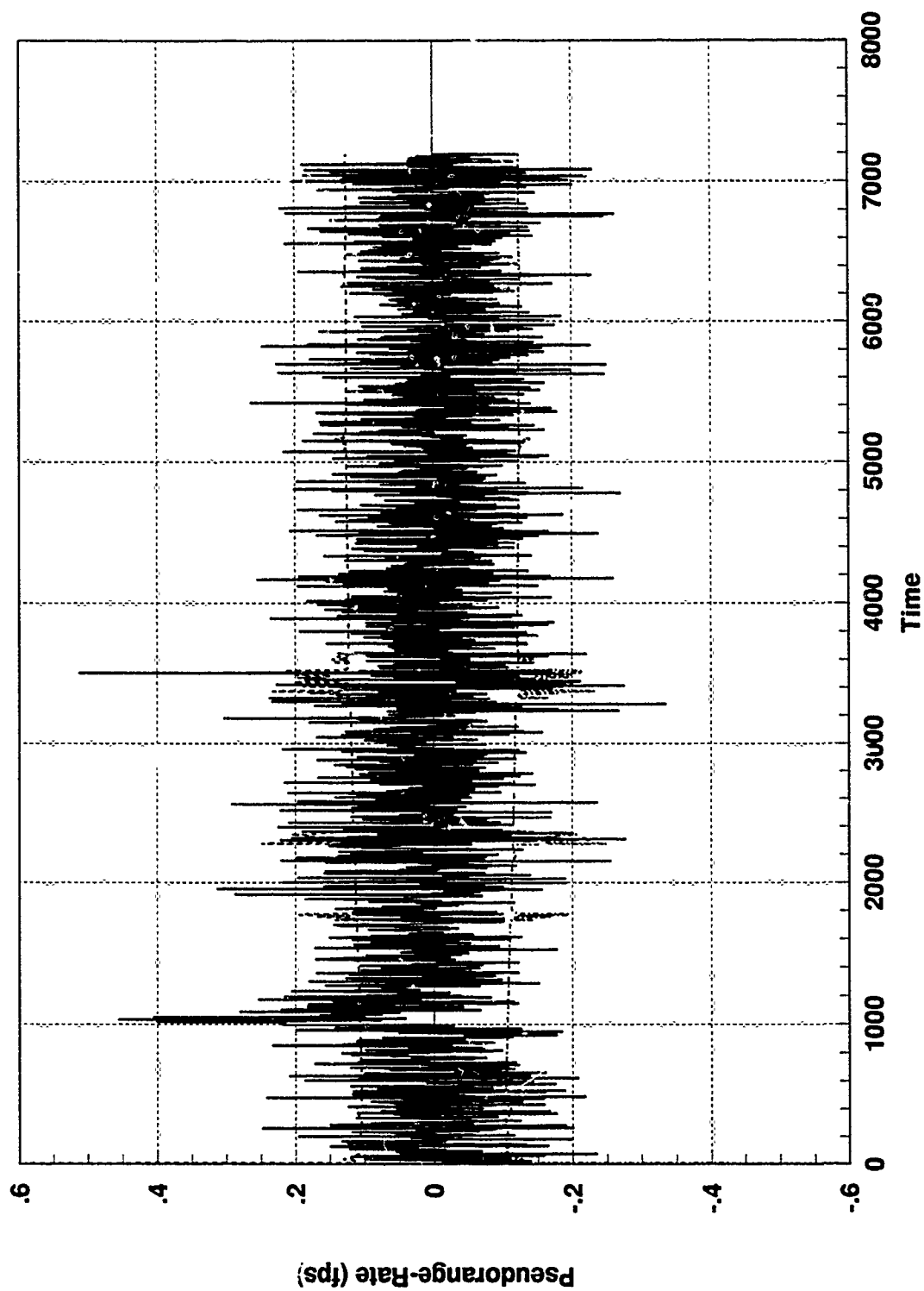
Figure E.14 SAREO Residual and One-Sigma Bound, GPS & TAN Constant Bias

Figure E.15 SAREO Residual and One-Sigma Bound, GPS & TAN Constant Bias

E-16
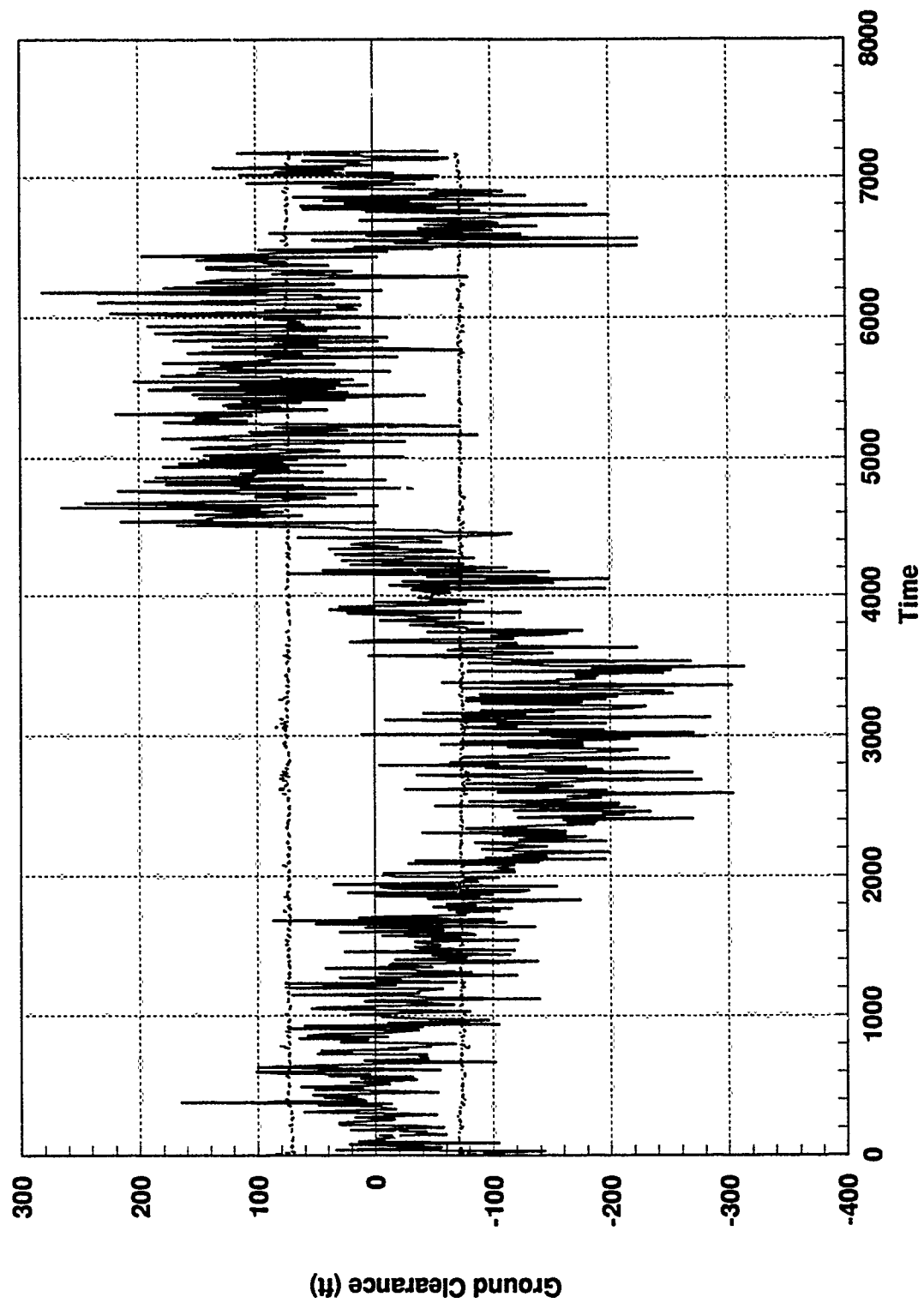
Figure E.16 Centralized Filter TAN Residual, GPS & TAN Constant Bias

Appendix F: Federated Filter Residual Plots

Soft Failure - Constant Bias

GENERAL INFORMATION

The series of plots in this appendix allows for a comparison of the fault detection considerations of this thesis. Establishing the specific results for the federated filter residual behavior during simulated failed operating conditions with the failure represented as a constant bias of reasonable magnitude added directly to the residual computation from 1000 to 3500 seconds on the GPS Satellite #1, PseudoRange-Rate residual, and a constant bias added directly to the residual computation from 4500 to 6500 seconds on the TAN Ground Clearance residual.

Each of the residual plots contained in this appendix are obtained from Monte Carlo simulations using DKFSIM Version 1.1. All residual outputs are recorded for the entire 7200 second duration of the simulation. The residual outputs and the upper and lower one-sigma bounds were computed for one Monte Carlo simulation.

There are 16 plots, 8 for the GPS, 7 for the SAR, and one for the TAN sensor measurements. Each plot has the appropriate residual description. A listing of the titles of each these figures is found in the List of Figures at the beginning of this document.

Figure F.1 GPS Sat 1 Residual and One-Sigma Bound, GPS & TAN Constant Bias

Figure F.2 GPS Sat 1 Residual and One-Sigma Bound, GPS & TAN Constant Bias

F-3

Figure F.3 GPS Sat 2 Residual and One-Sigma Bound, GPS & TAN Constant Bias

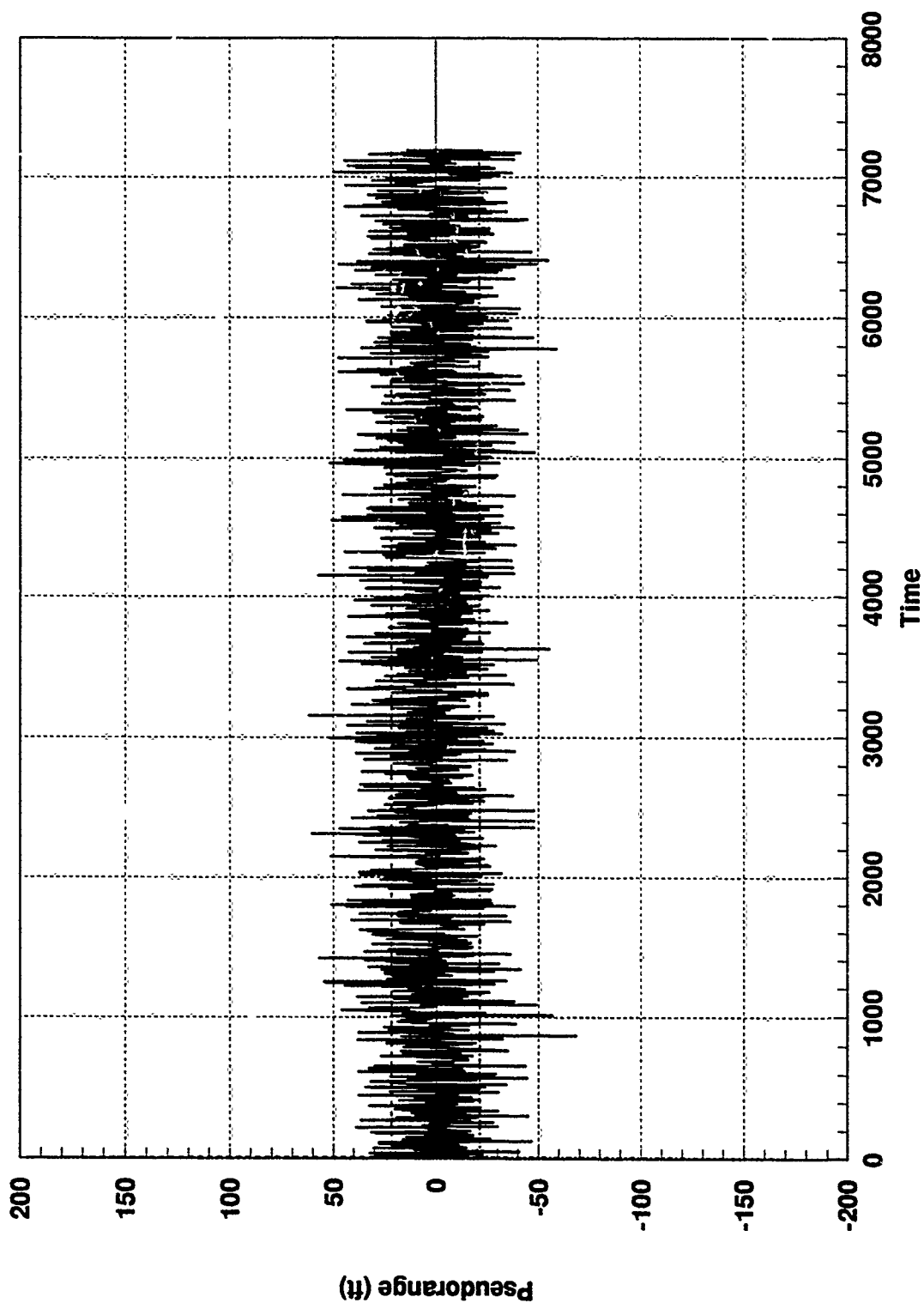Figure F.4 GPS Sat 2 Residual and One-Sigma Bound, GPS & TAN Constant Bias

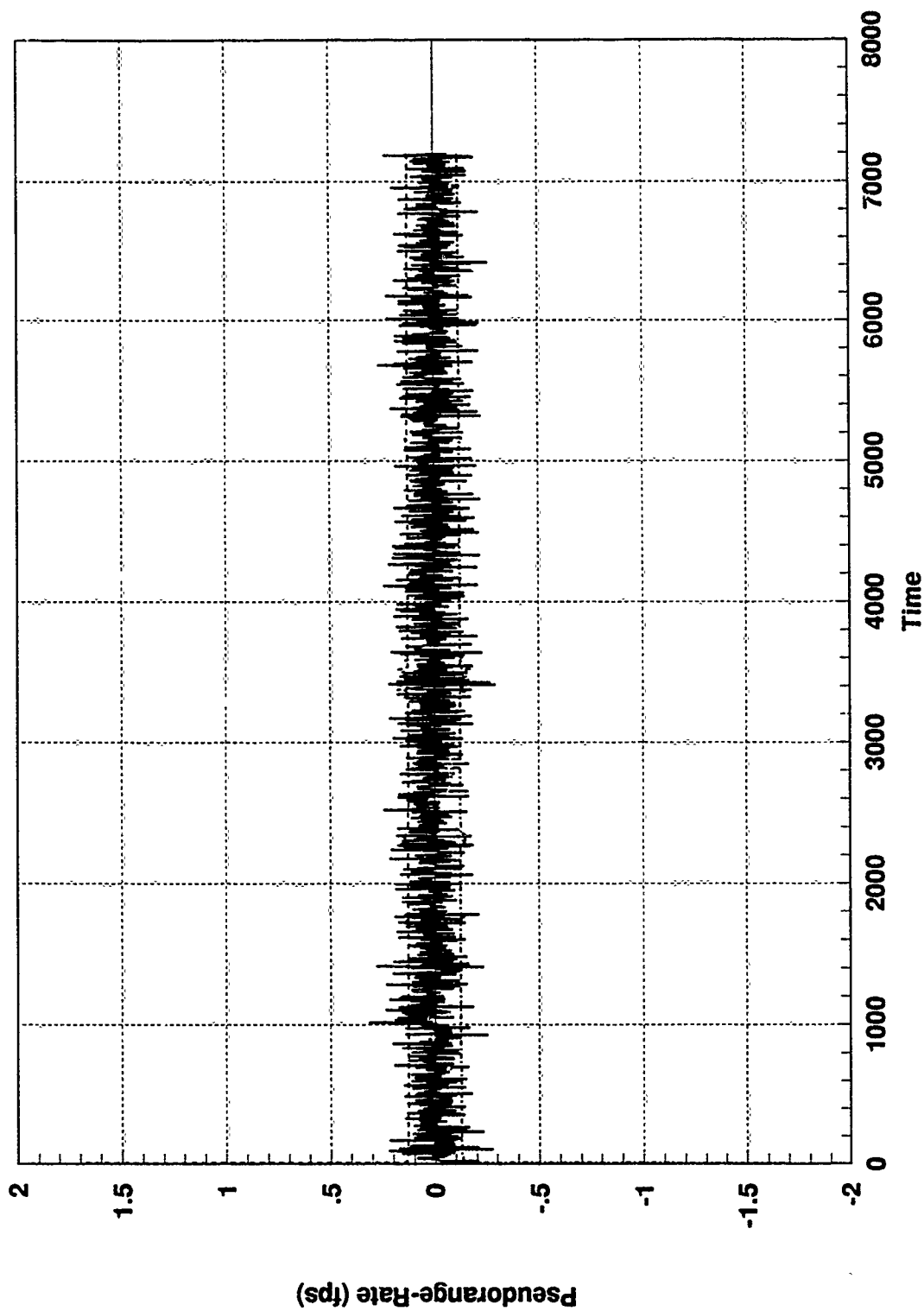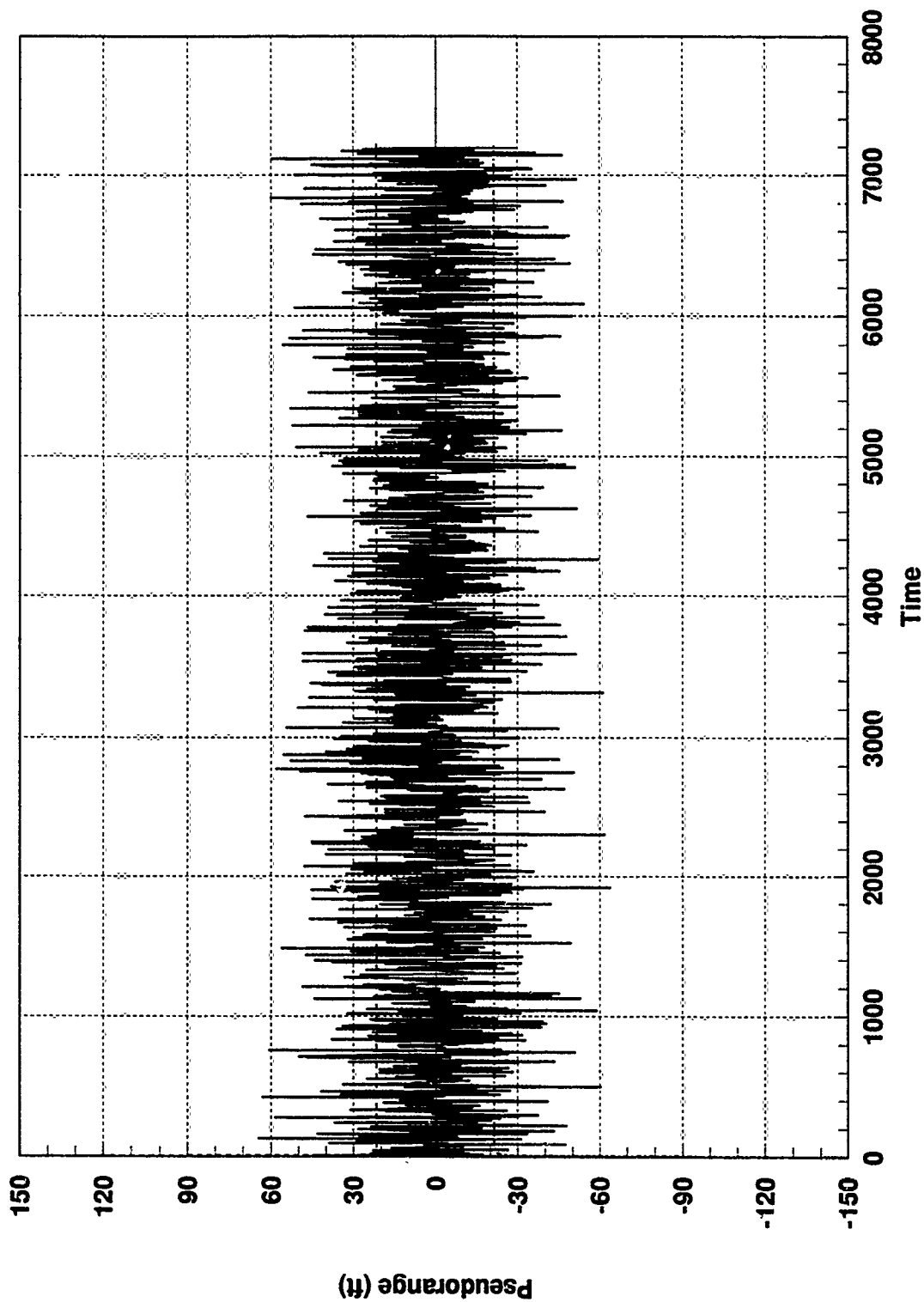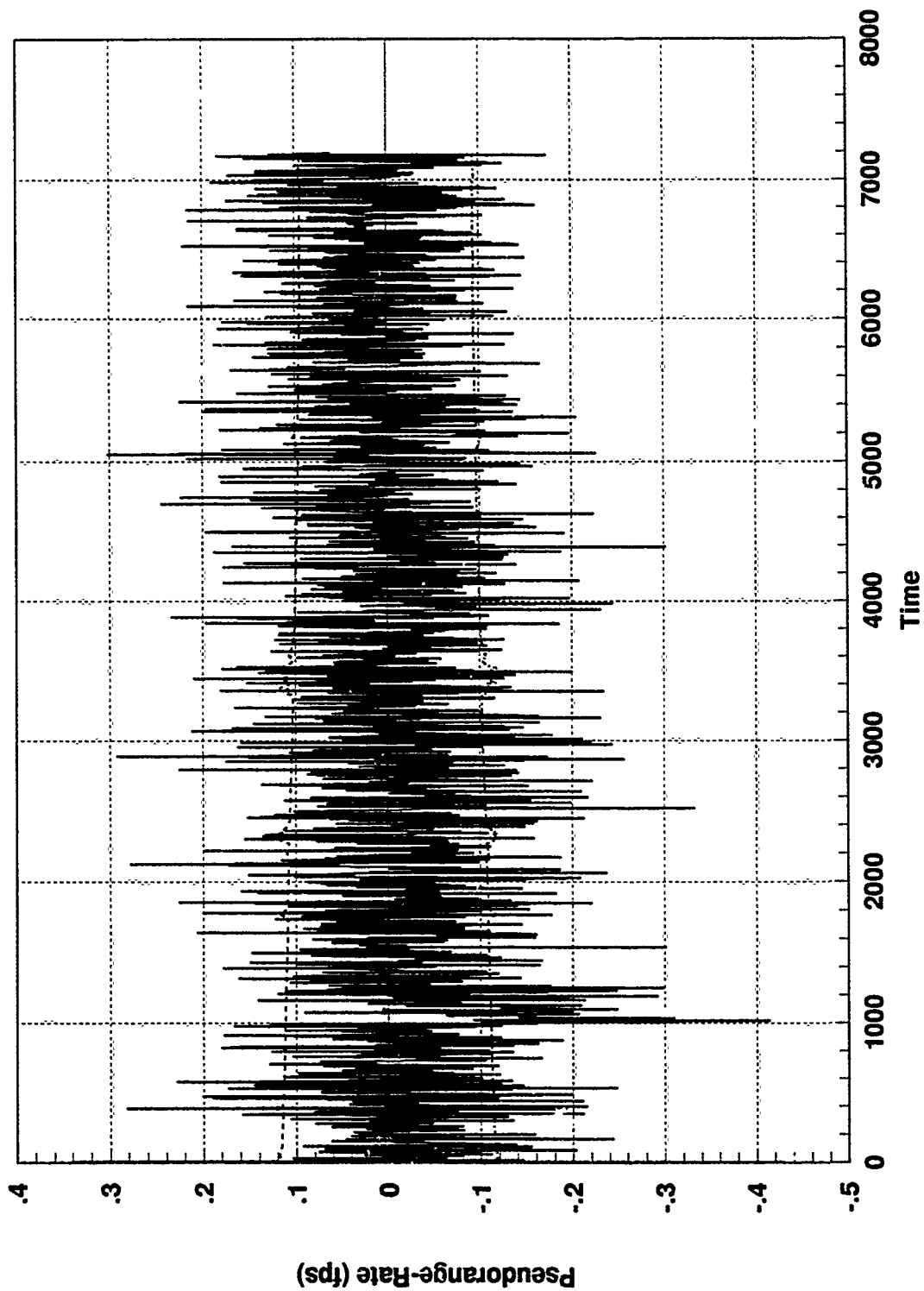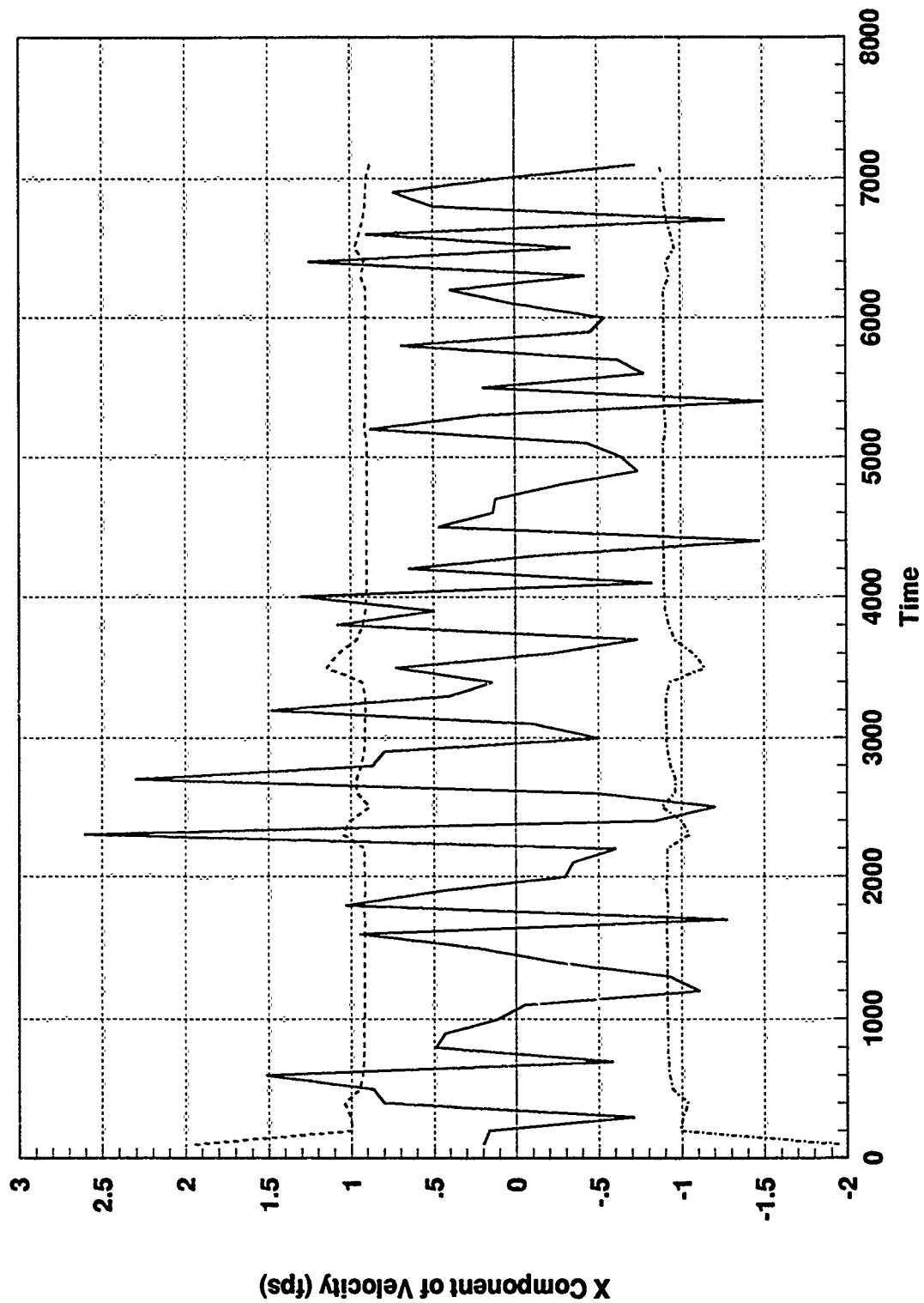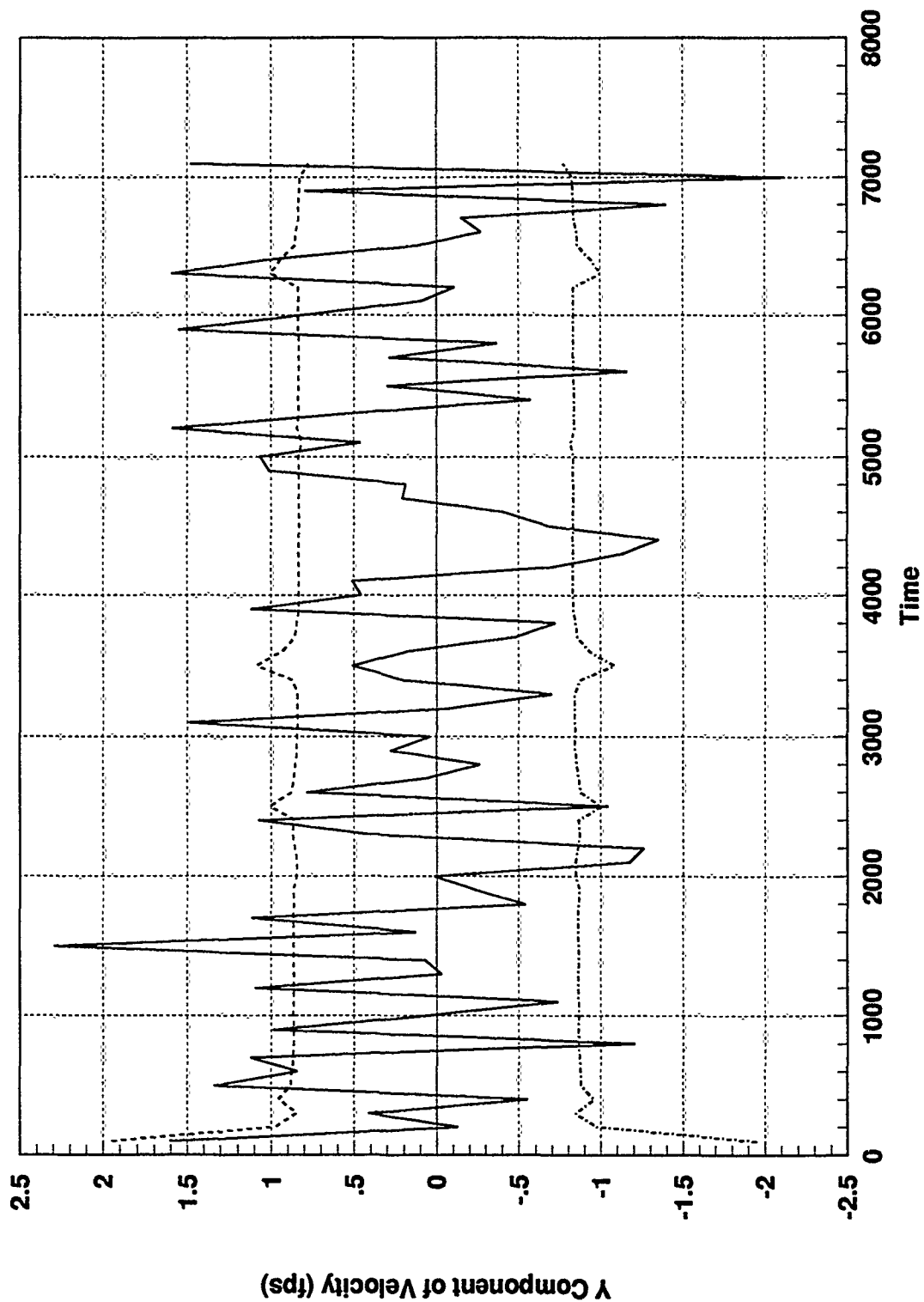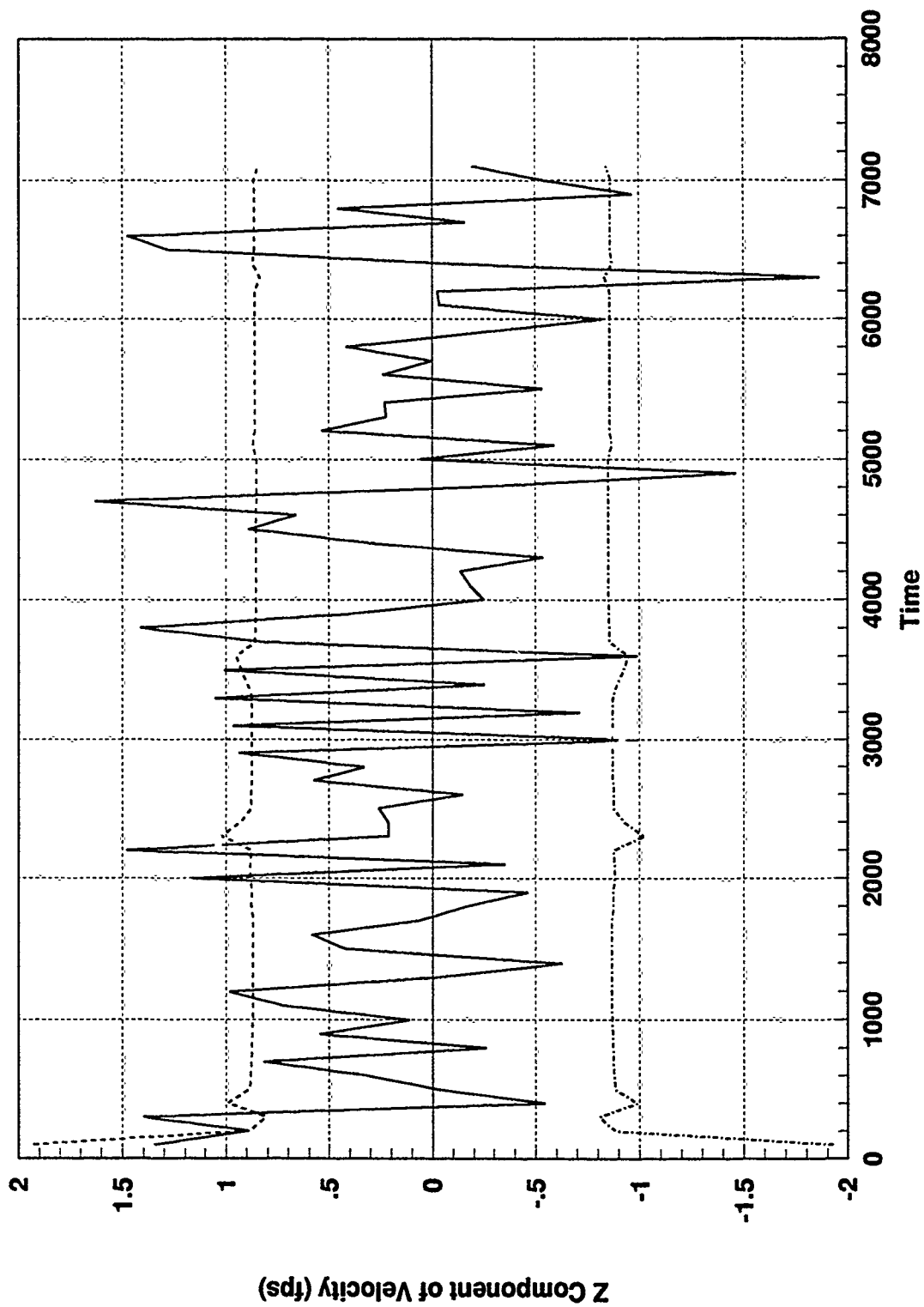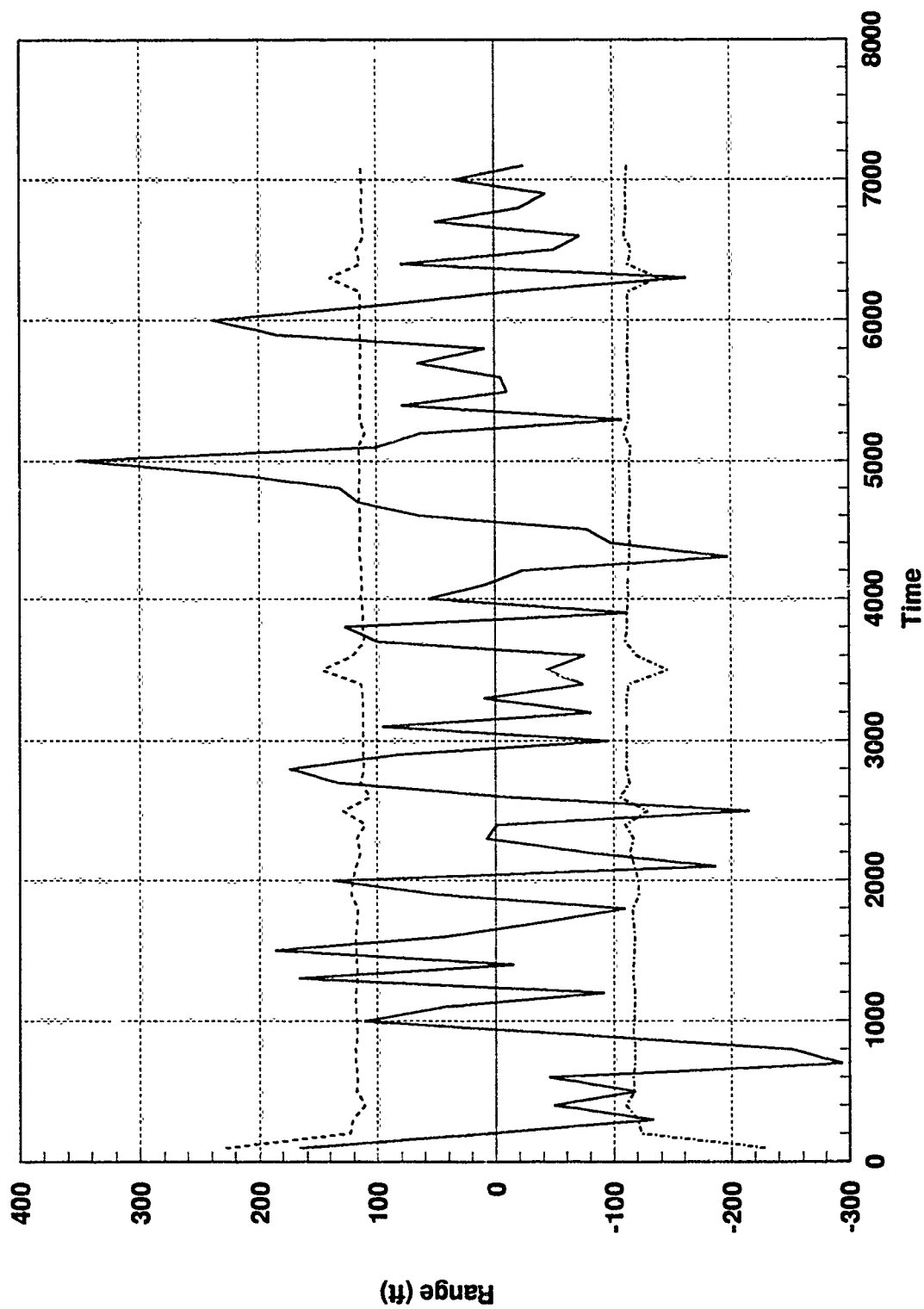Figure F.5 GPS Sat 3 Residual and One-Sigma Bound, GPS & TAN Constant Bias

Figure F.6 GPS Sat 3 Residual and One-Sigma Bound, GPS & TAN Constant Bias

F-7

Figure F.7 GPS Sat 4 Residual and One-Sigma Bound, GPS & TAN Constant Bias

Figure F.8 GPS Sat 4 Residual and One-Sigma Bound, GPS & TAN Constant Bias

Figure F.9 SARPVU Residual and One-Sigma Bound, GPS & TAN Constant Bias

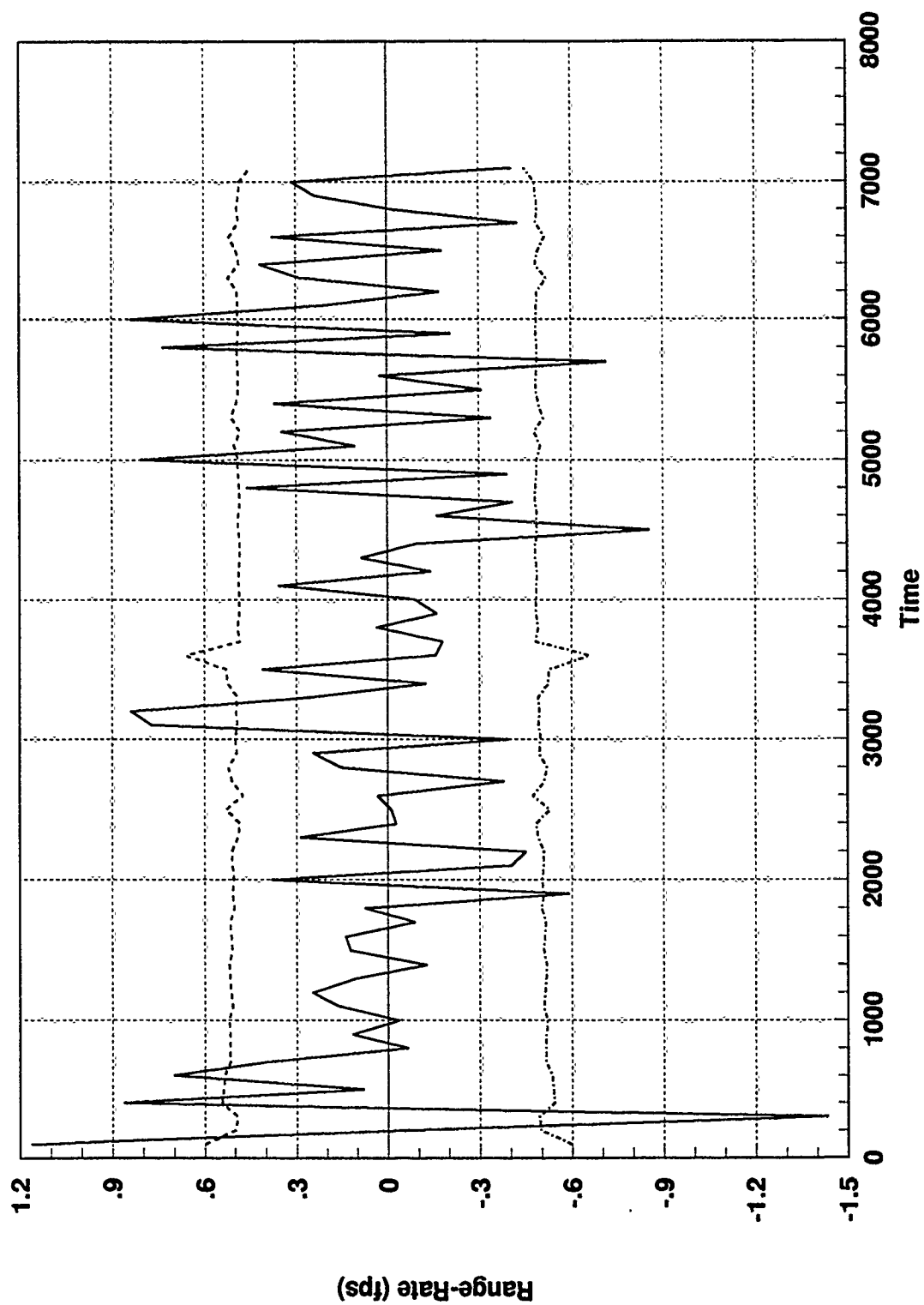Figure F.10 SARPVU Residual and One-Sigma Bound, GPS & TAN Constant Bias

Figure F.11 SARPVU Residual and One-Sigma Bound, GPS & TAN Constant Bias

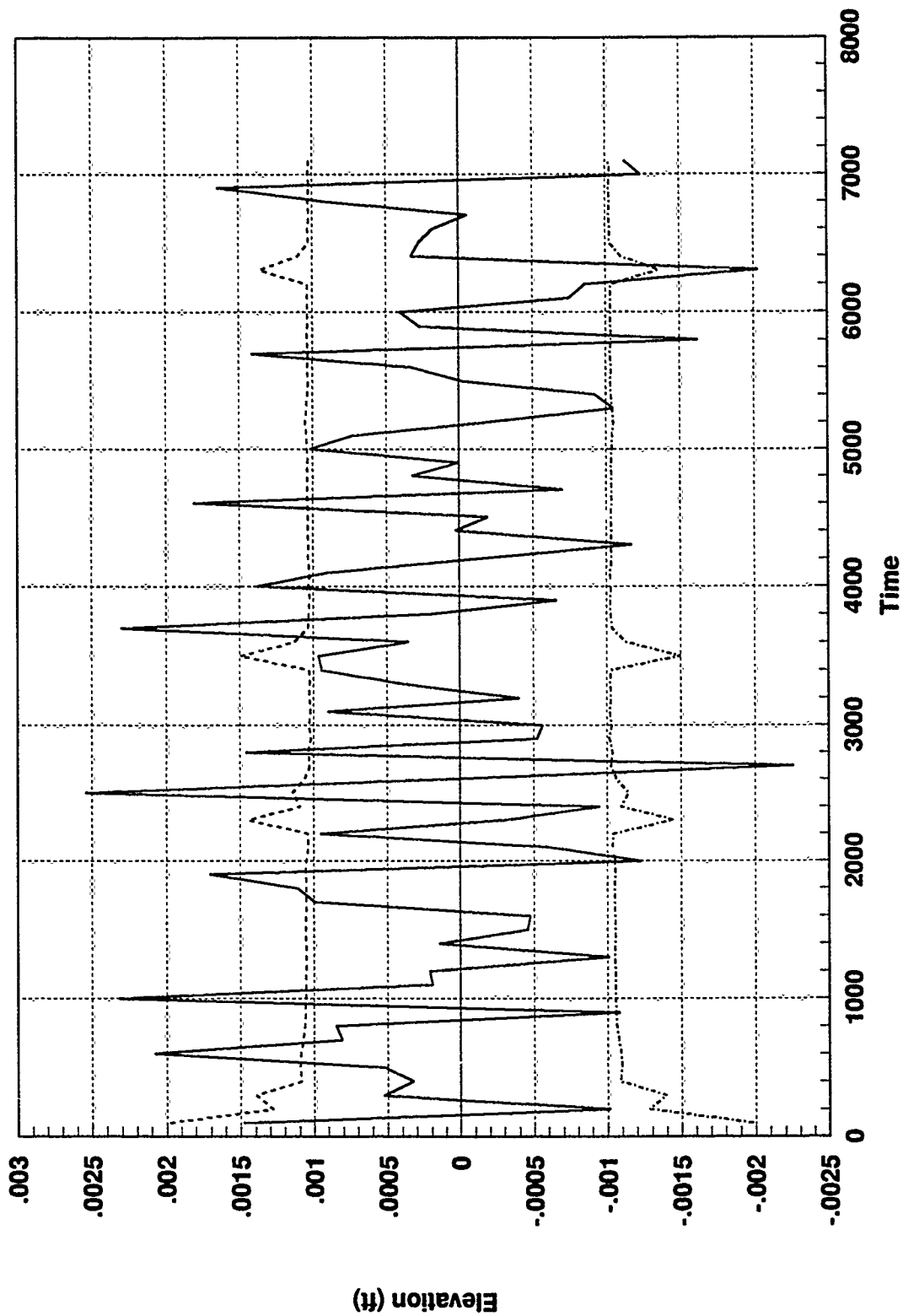Figure F.12 SAREO Residual and One-Sigma Bound, GPS & TAN Constant Bias

F-13

Figure F.13 SAREO Residual and One-Sigma Bound, GPS & TAN Constant Bias

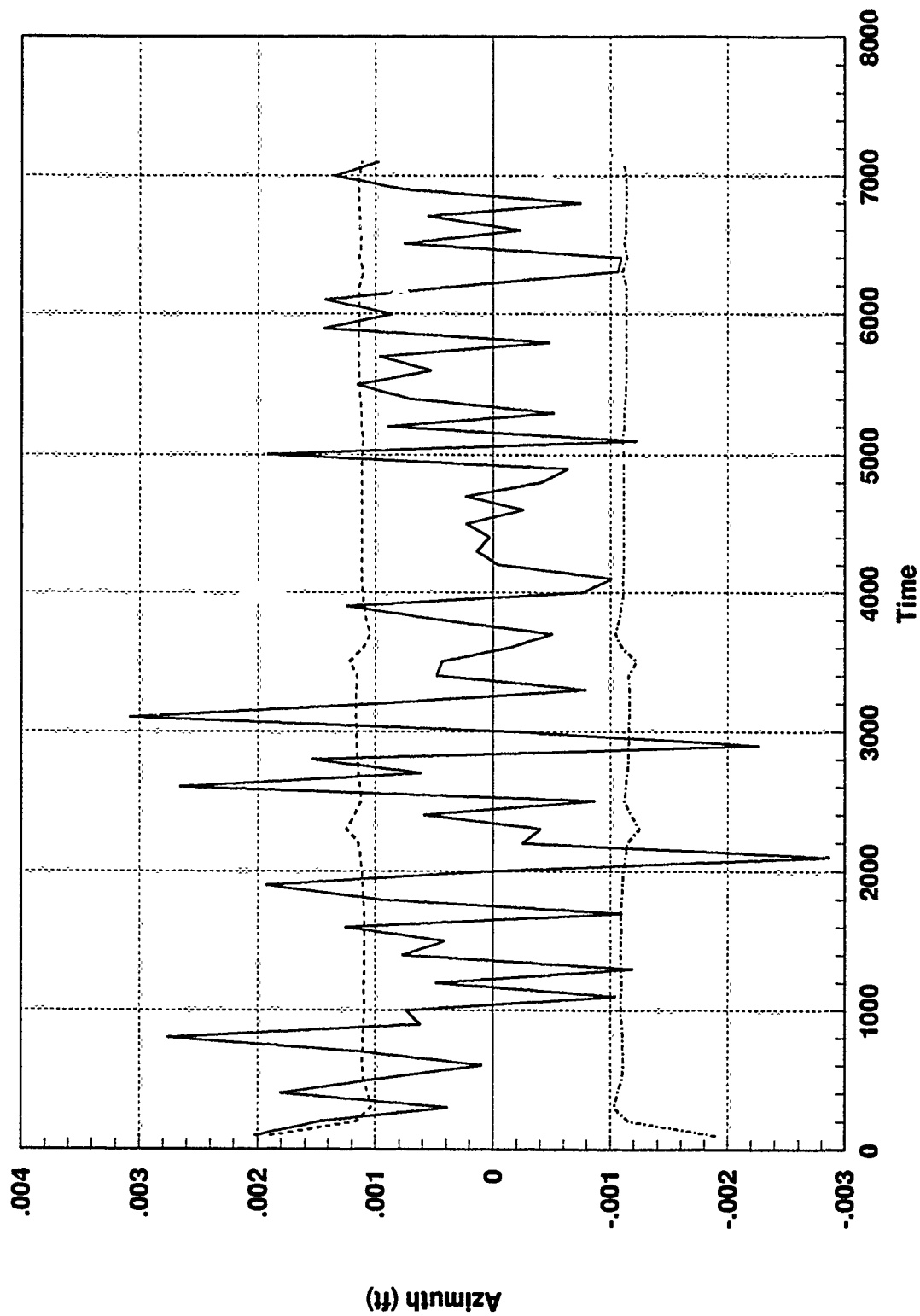**Figure F.14 SAREO Residual and One-Sigma Bound, GPS & TAN Constant Bias**

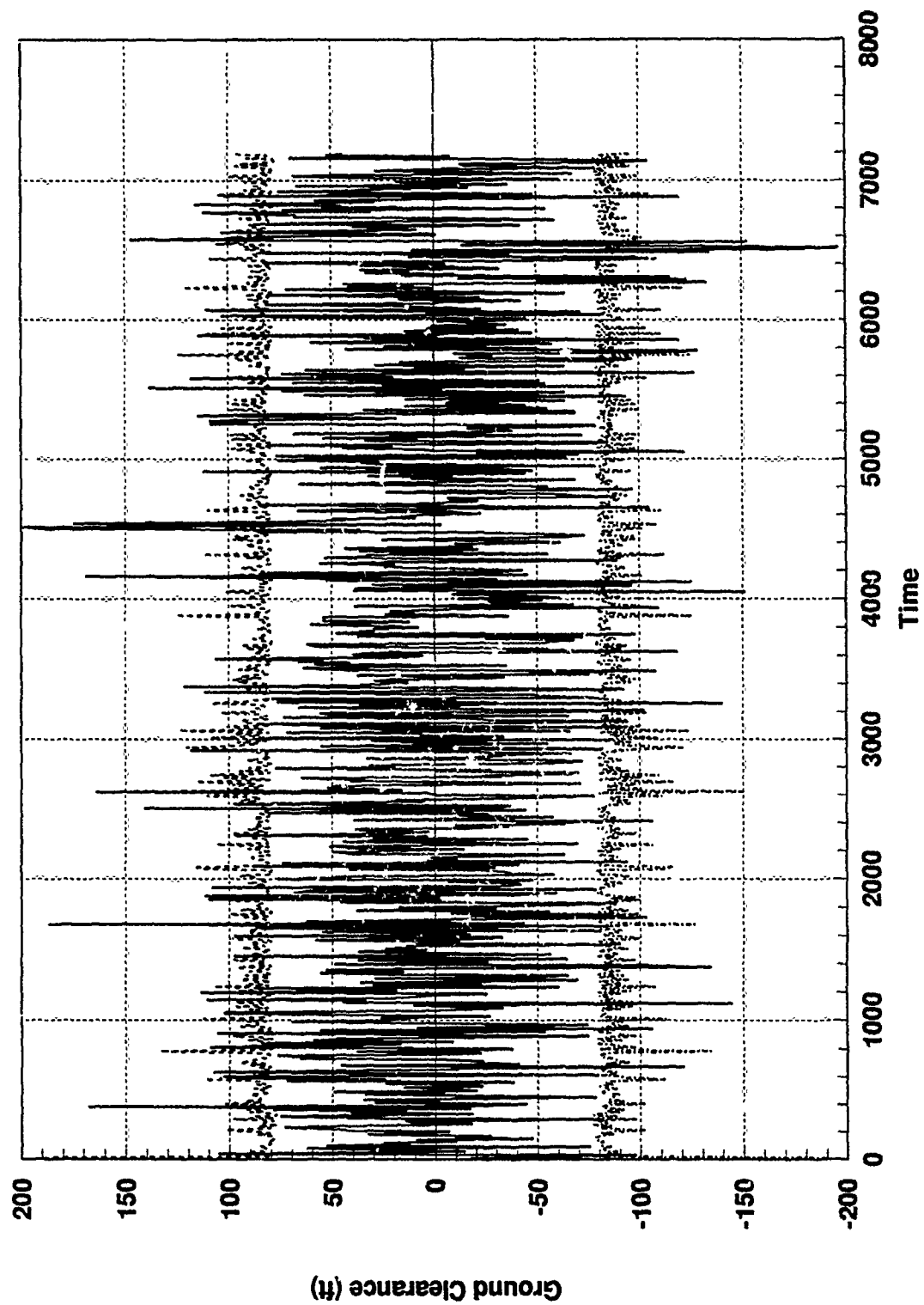Figure F.15 SAREO Residual and One-Sigma Bound, GPS & TAN Constant Bias

Figure F.16 Federated Filter TAN Residual, GPS & TAN Constant Bias

Appendix G: Centralized Filter Residual Plots

Soft Failure - GPS Ramp Bias

GENERAL INFORMATION

The series of plots in this appendix allows for a comparison of the fault detection considerations of this thesis. Establishing the specific results for the centralized filter residual behavior during simulated failed operating conditions with the failure represented as a ramp bias of reasonable magnitude added directly to the residual computation from 1000 to 3500 seconds on the GPS Satellite #1, PseudoRange-Rate residual, and a constant bias added directly to the residual computation from 4500 to 6500 seconds on the TAN Ground Clearance residual. The TAN bias was left unchanged for simplicity.

Each of the residual plots contained in this appendix are obtained from Monte Carlo simulations using DKFSIM Version 1.1. All residual outputs are recorded for the entire 7200 second duration of the simulation. The residual outputs and the upper and lower one-sigma bounds were computed for one Monte Carlo simulation.

There are 16 plots, 8 for the GPS, 7 for the SAR, and one for the TAN sensor measurements. Each plot has the appropriate residual description. A listing of the titles of each these figures is found in the List of Figures at the beginning of this document.
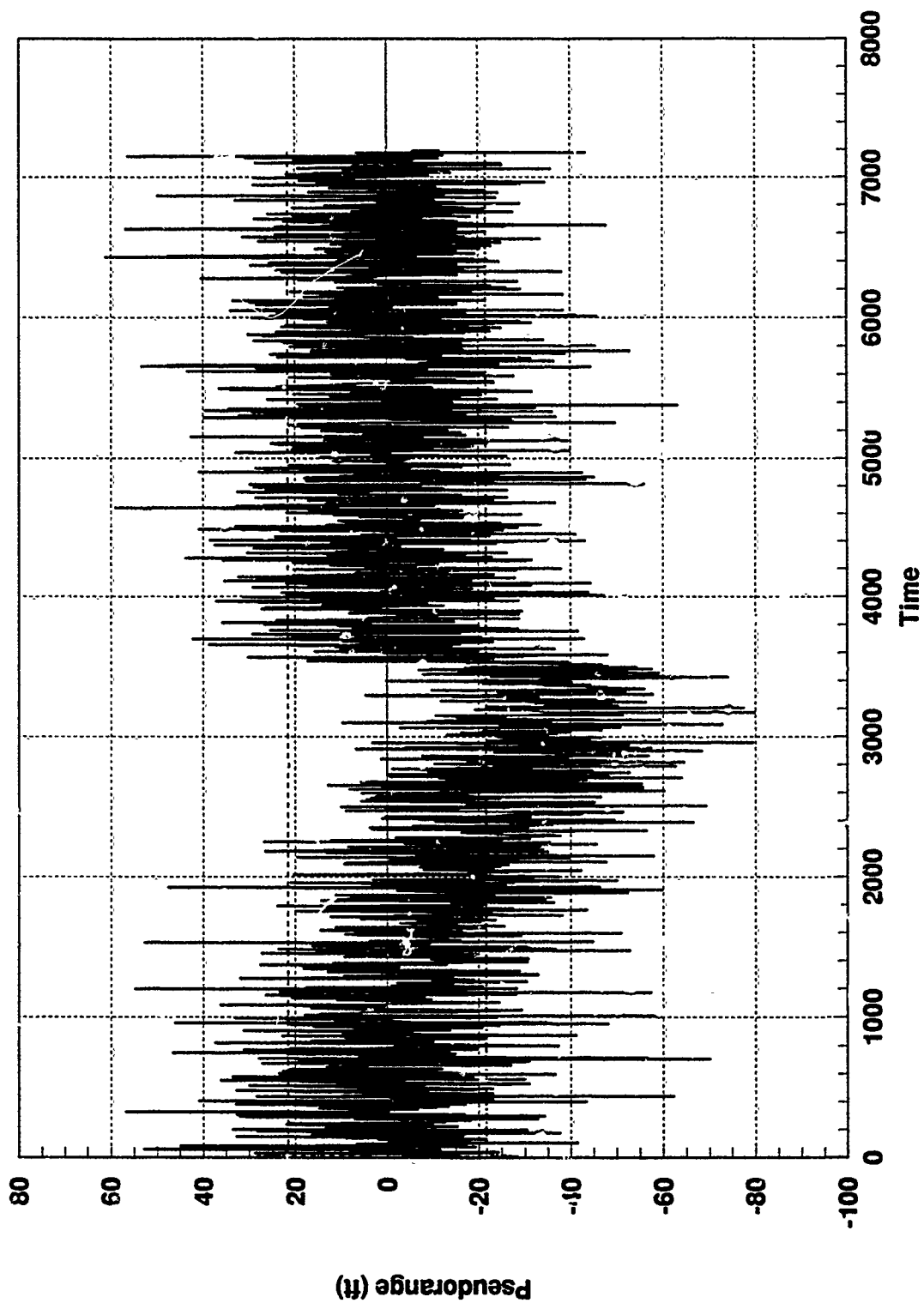
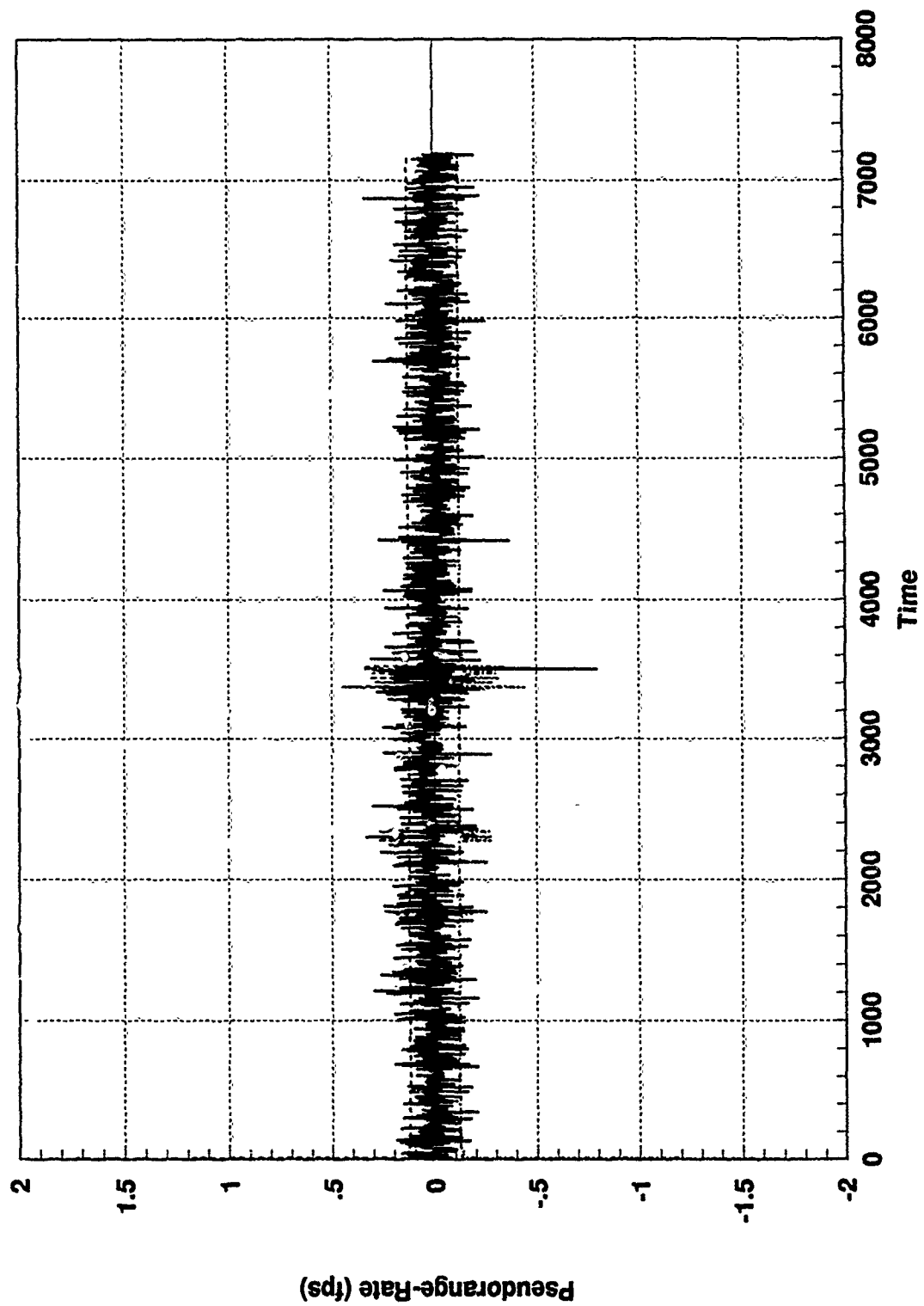Figure G.1 GPS Sat 1 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

Figure G.2 GPS Sat 1 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
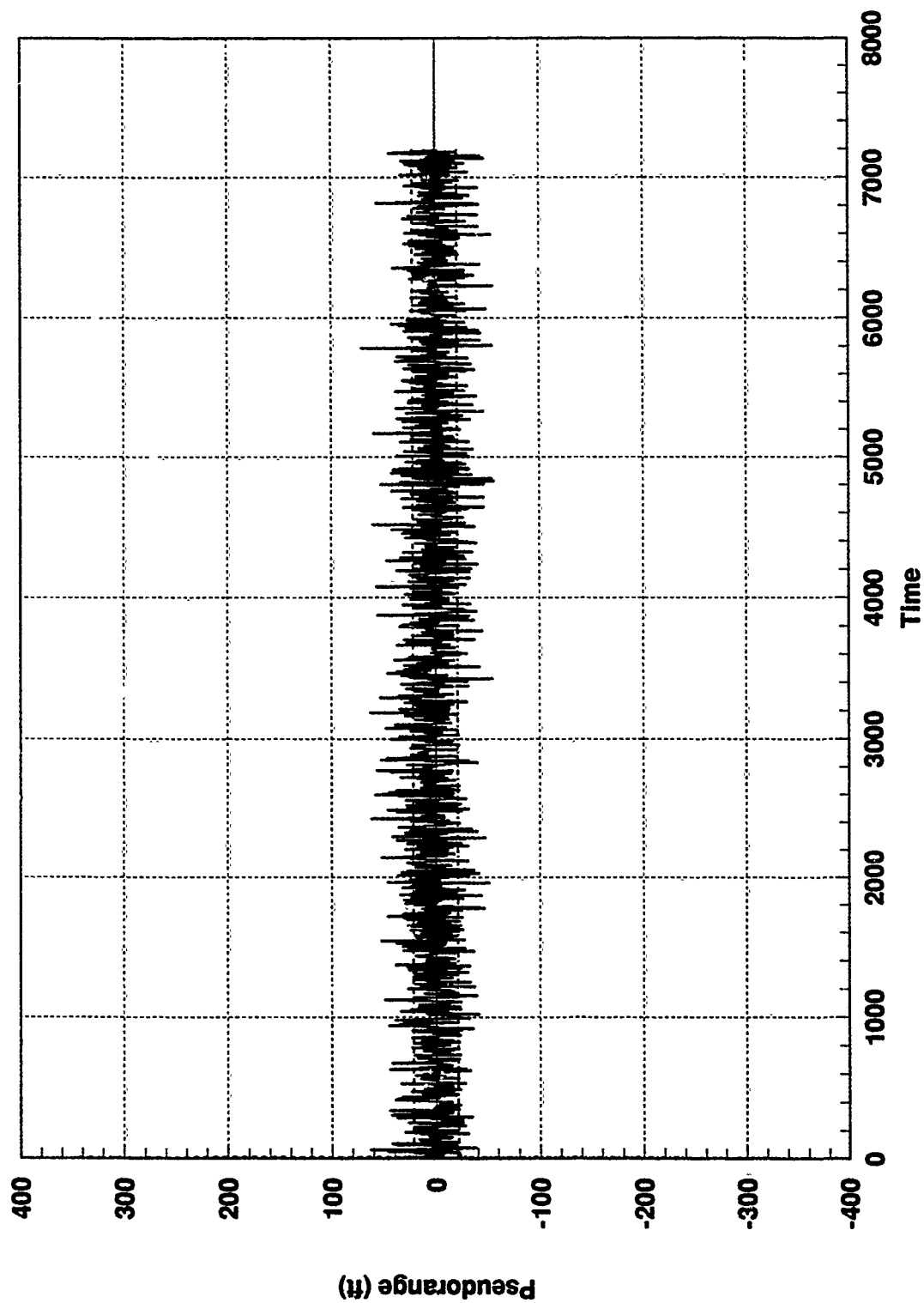
Figure G.3 GPS Sat 2 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

Figure G.4 GPS Sat 2 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

Figure G.5 GPS Sat 3 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

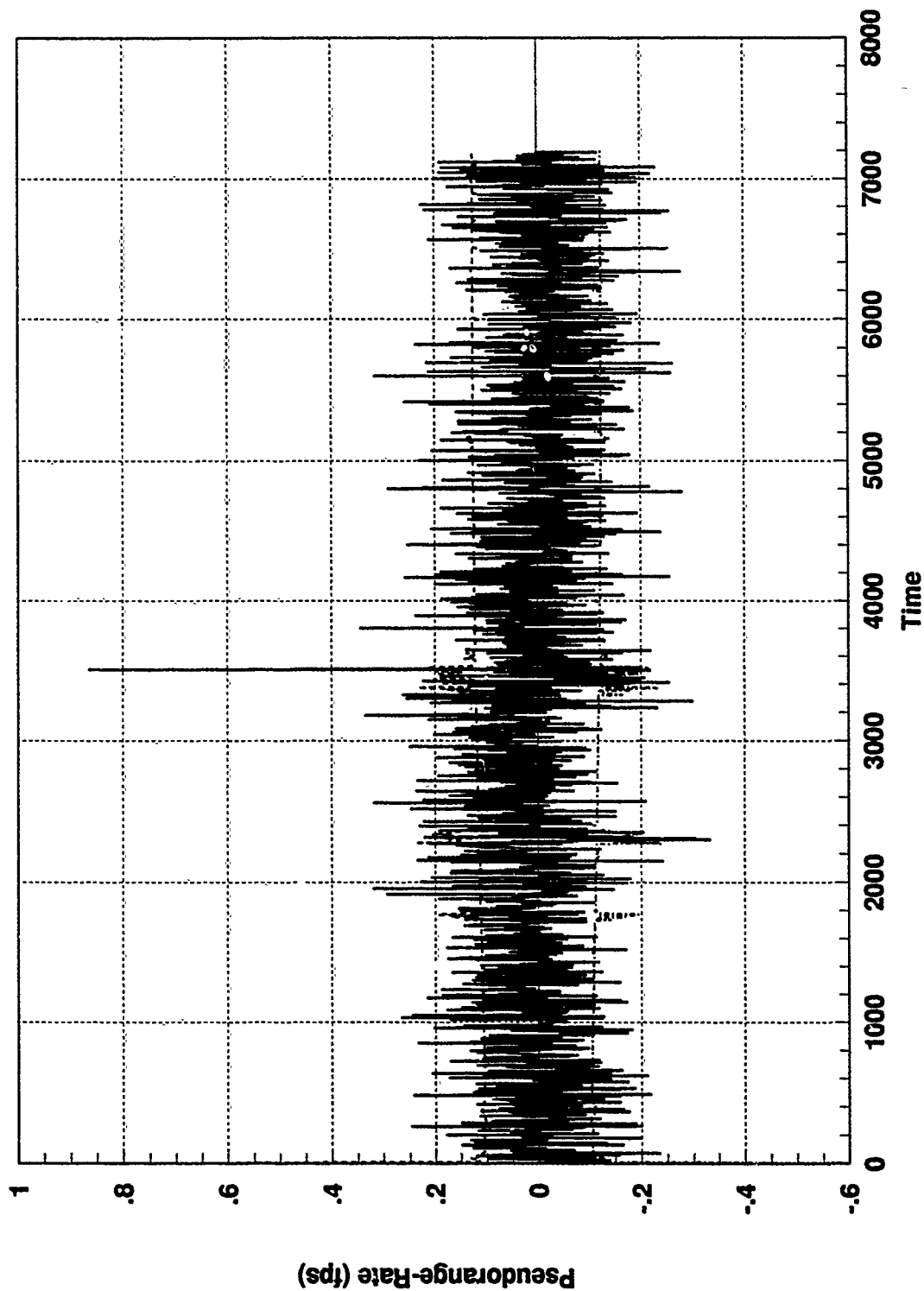Figure G.6 GPS Sat 3 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

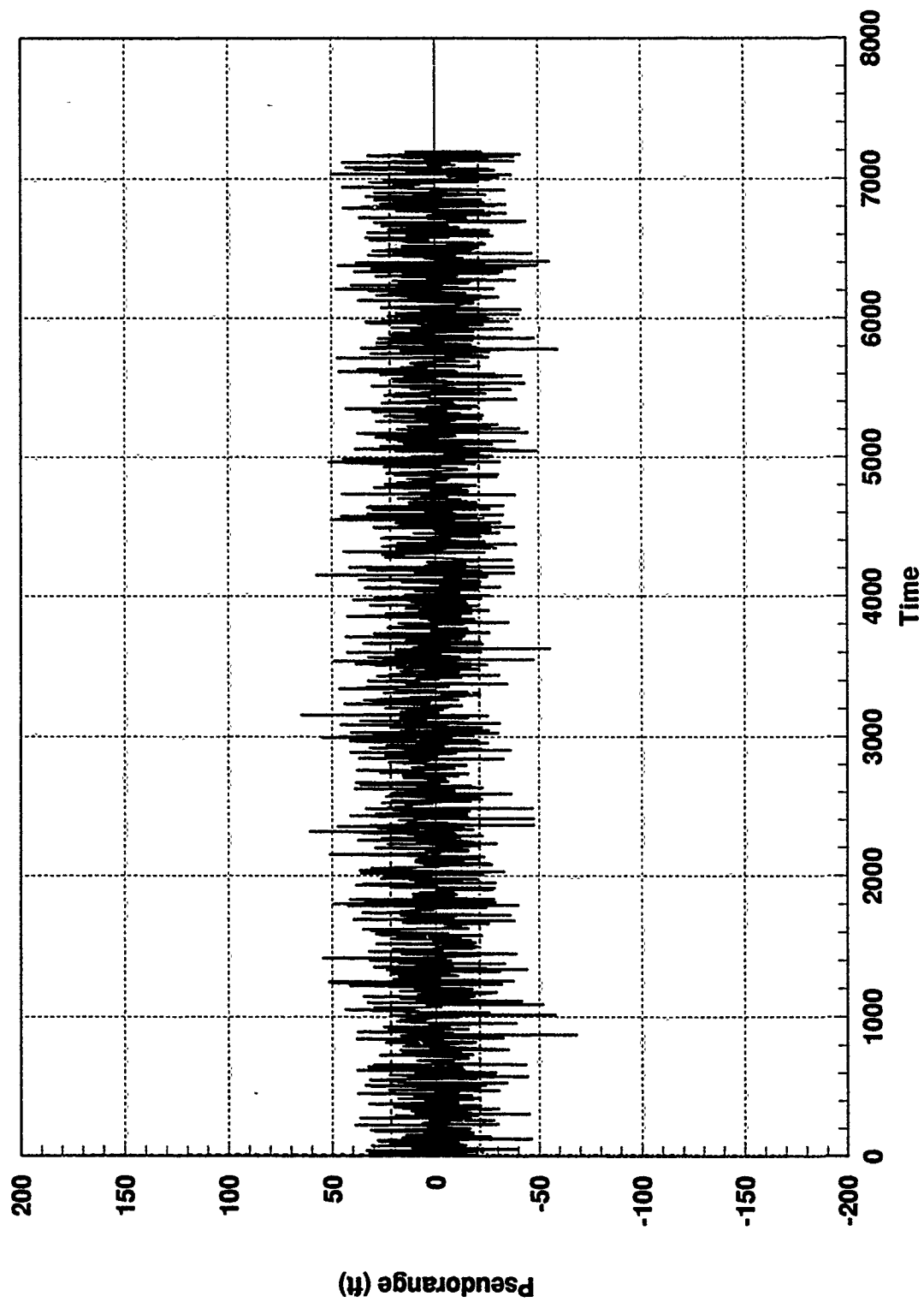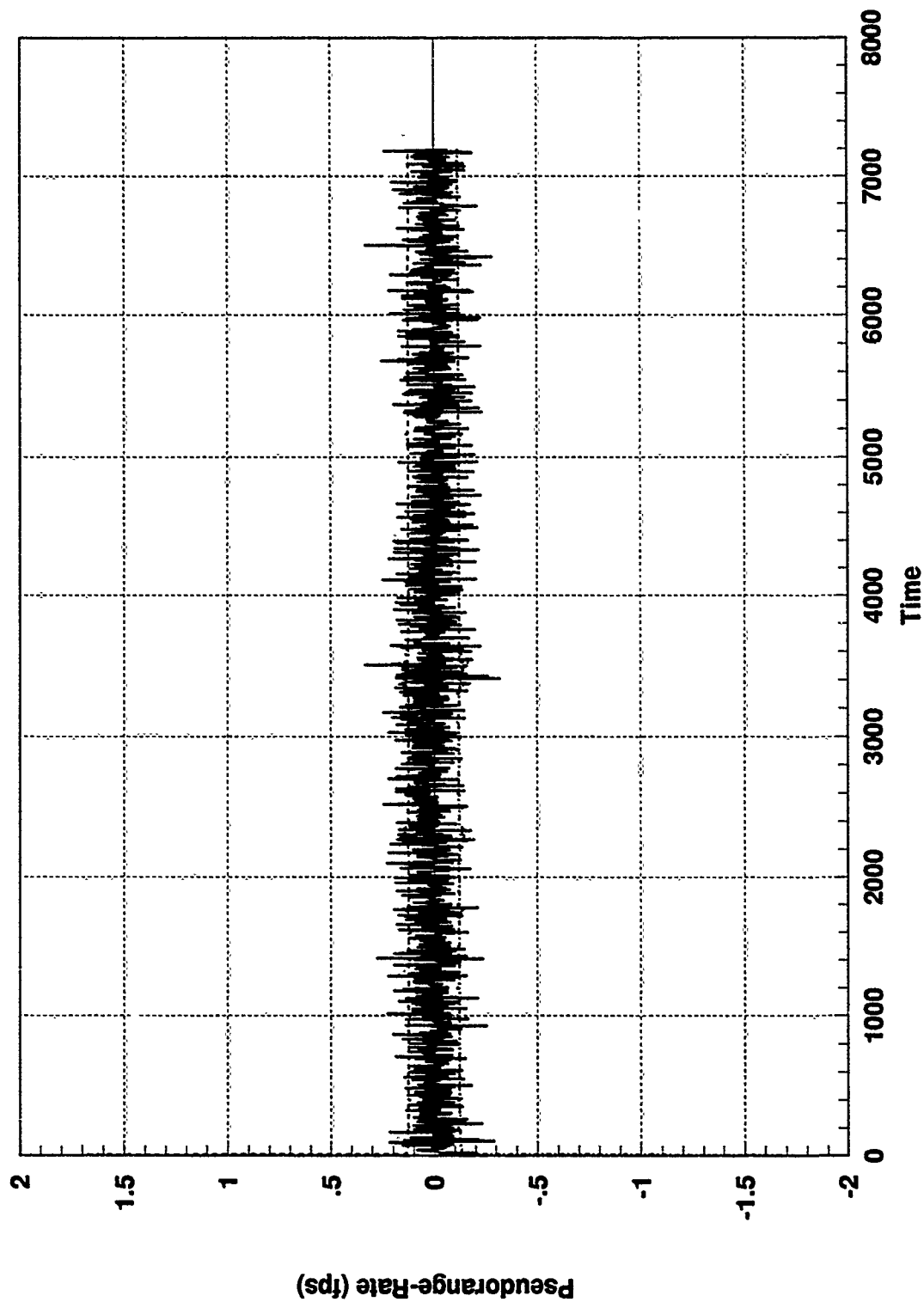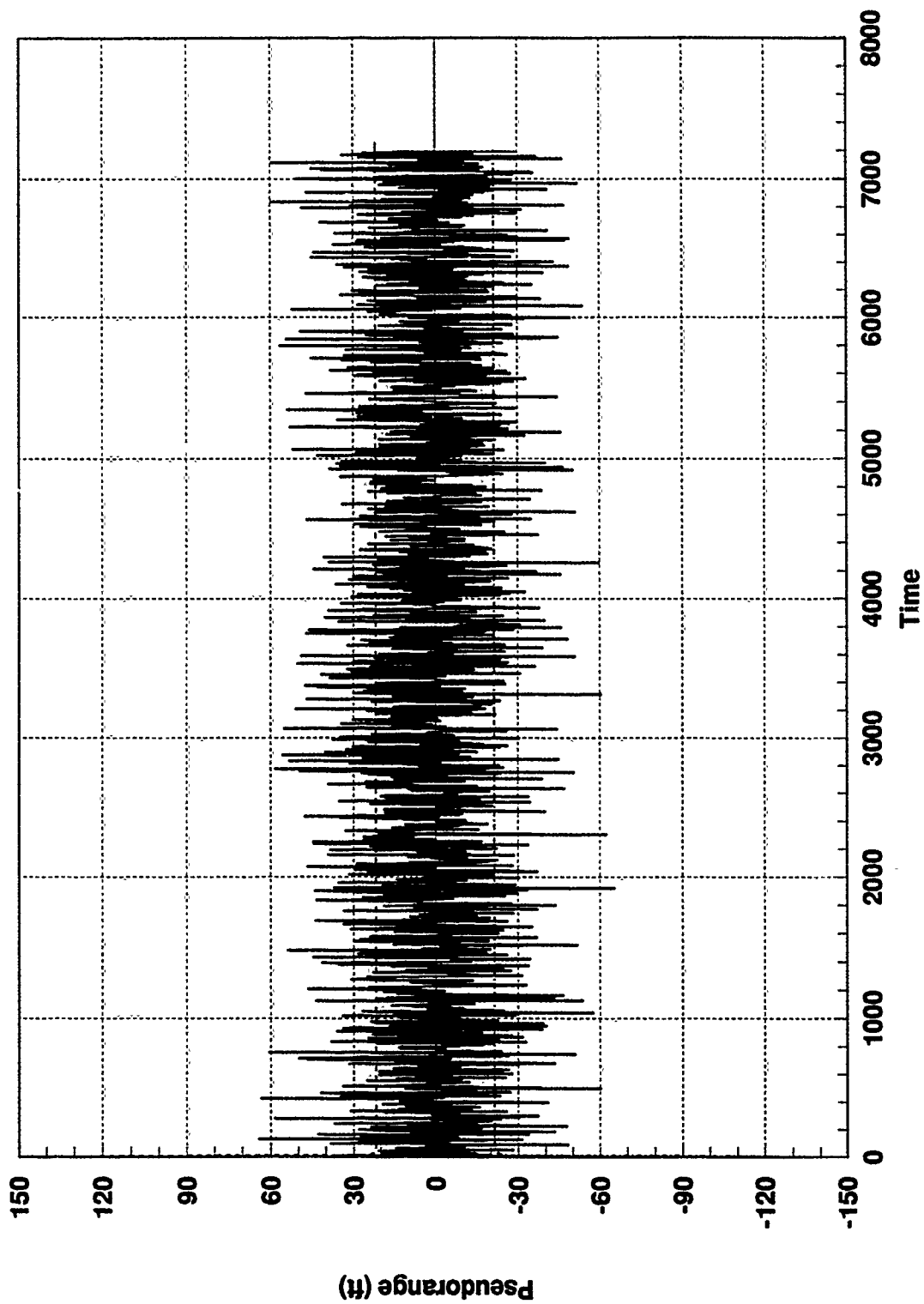Figure G.7 GPS Sat 4 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

Figure G.8 GPS Sat 4 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

Figure G.9 SARPVU Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

Figure G.10 SARPVU Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

G-11

Figure G.11 SARPVU Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
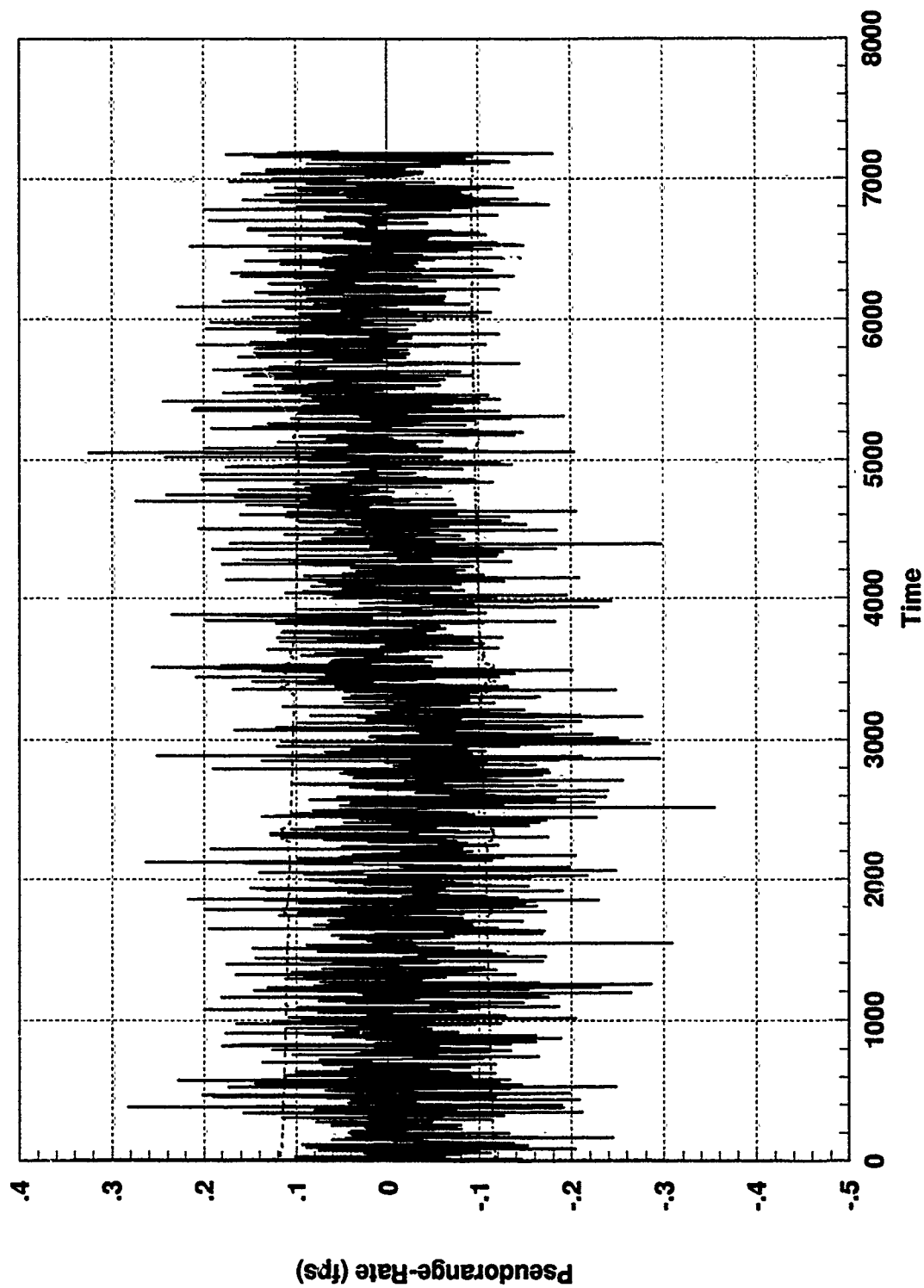
Figure G.12 SAREO Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

Figure G.13 SAREO Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

Figure G.14 SAREO Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

Figure G.15 SAREO Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
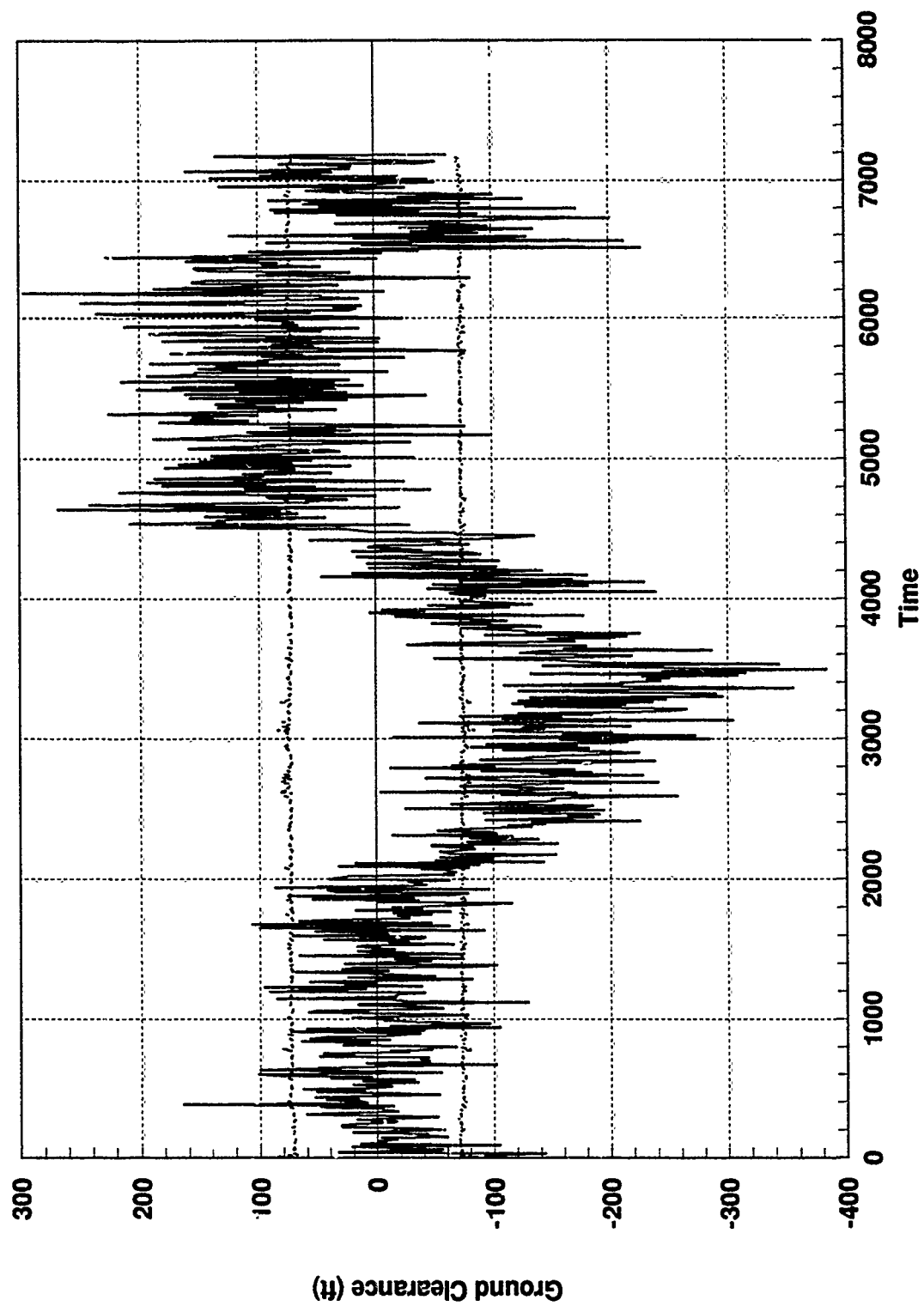
Figure G.16 Centralized Filter TAN Residual, GPS Ramp & TAN Bias

Appendix H: Federated Filter Residual Plots

Soft Failure - GPS Ramp Bias

GENERAL INFORMATION

The series of plots in this appendix allows for a comparison of the fault detection considerations of this thesis. Establishing the specific results for the federated filter residual behavior during simulated failed operating conditions with the failure represented as a ramp bias of reasonable magnitude added directly to the residual computation from 1000 to 3500 seconds on the GPS Satellite #1, PseudoRange-Rate residual, and a constant bias added directly to the residual computation from 4500 to 6500 seconds on the TAN Ground Clearance residual. The TAN bias was left unchanged for simplicity.

Each of the residual plots contained in this appendix are obtained from Monte Carlo simulations using DKFSIM Version 1.1. All residual outputs are recorded for the entire 7200 second duration of the simulation. The residual outputs and the upper and lower one-sigma bounds were computed for one Monte Carlo simulation.

There are 16 plots, 8 for the GPS, 7 for the SAR, and one for the TAN sensor measurements. Each plot has the appropriate residual description. A listing of the titles of each these figures is found in the List of Figures at the beginning of this document.

Figure H.1 GPS Sat 1 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
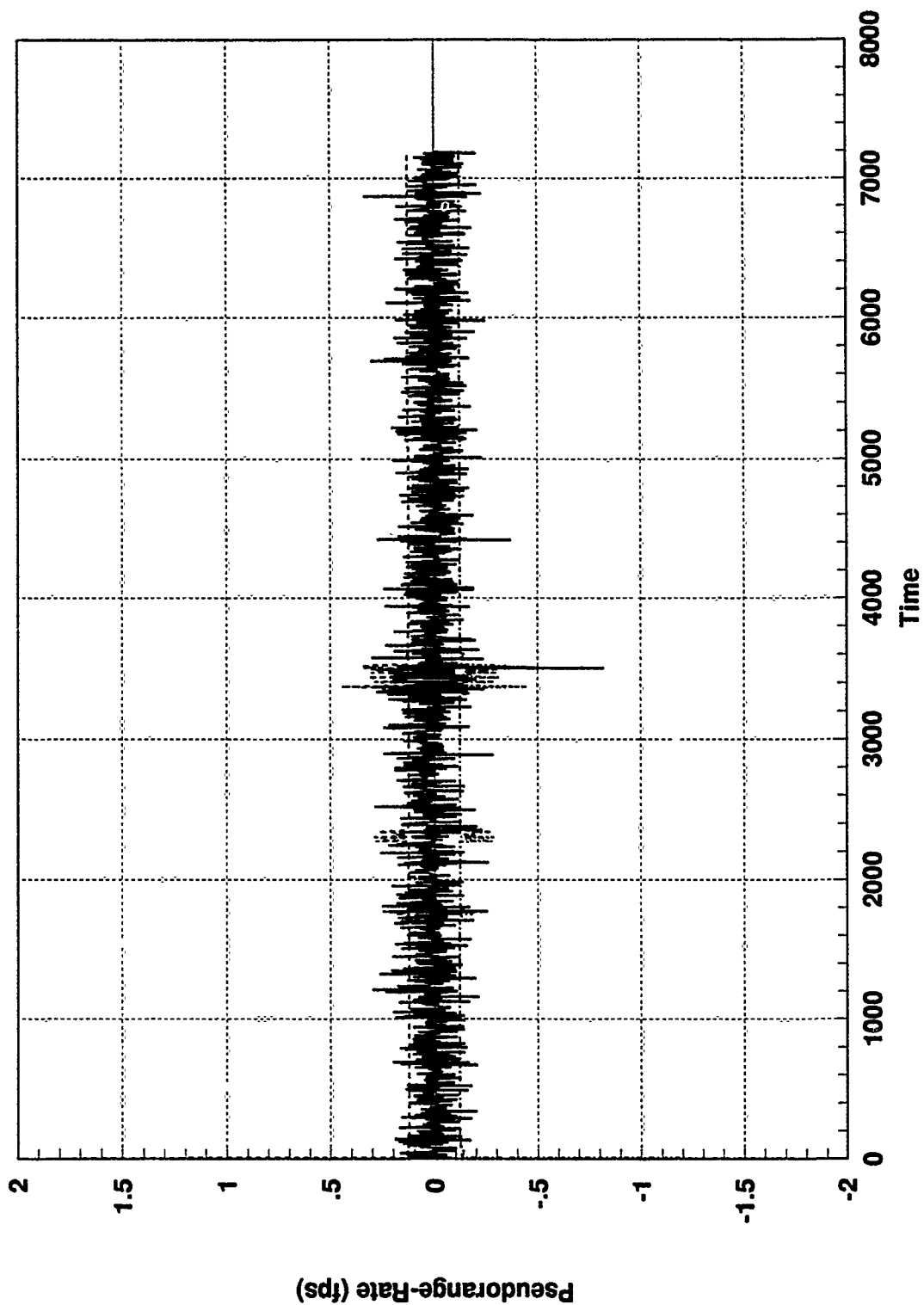
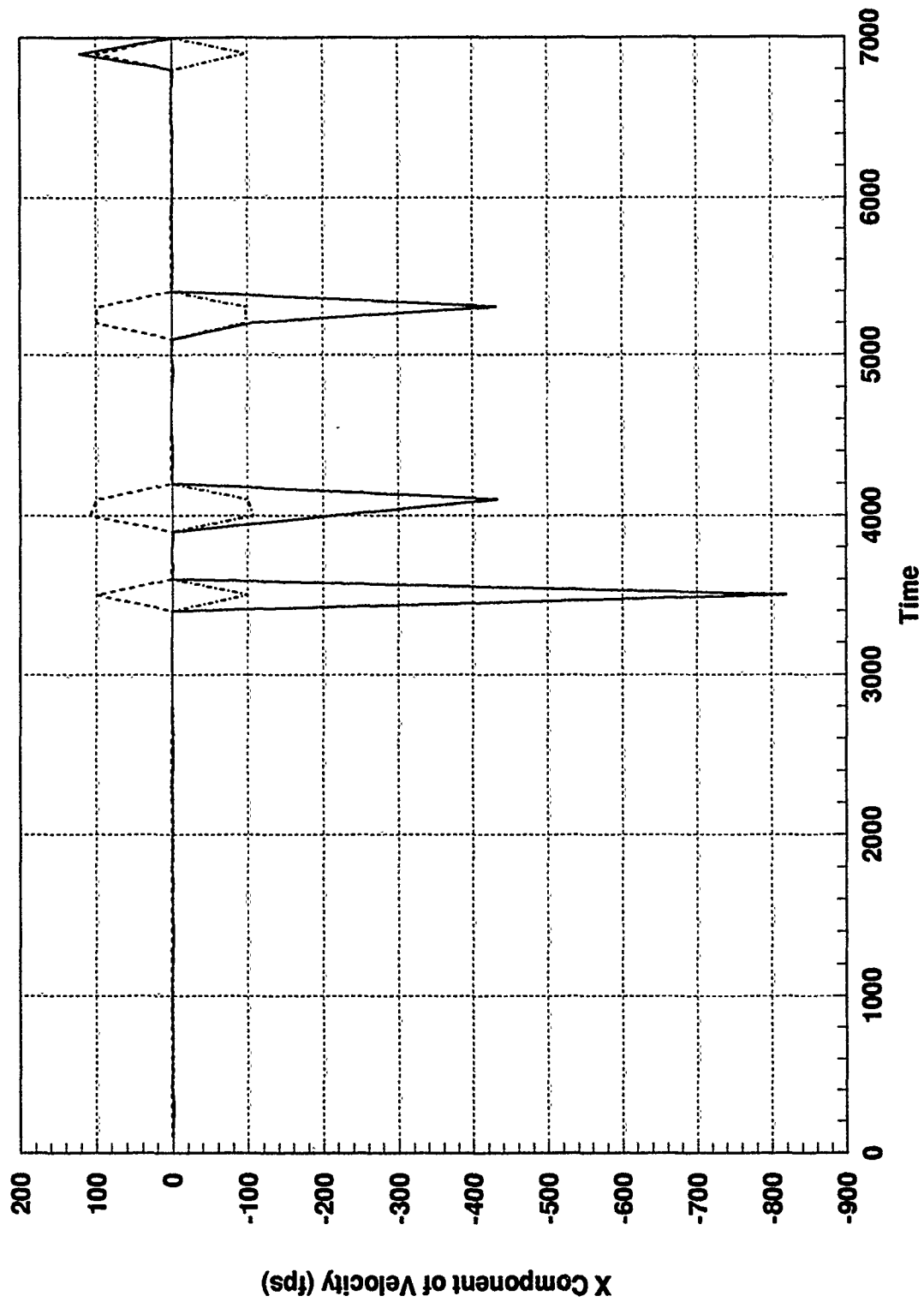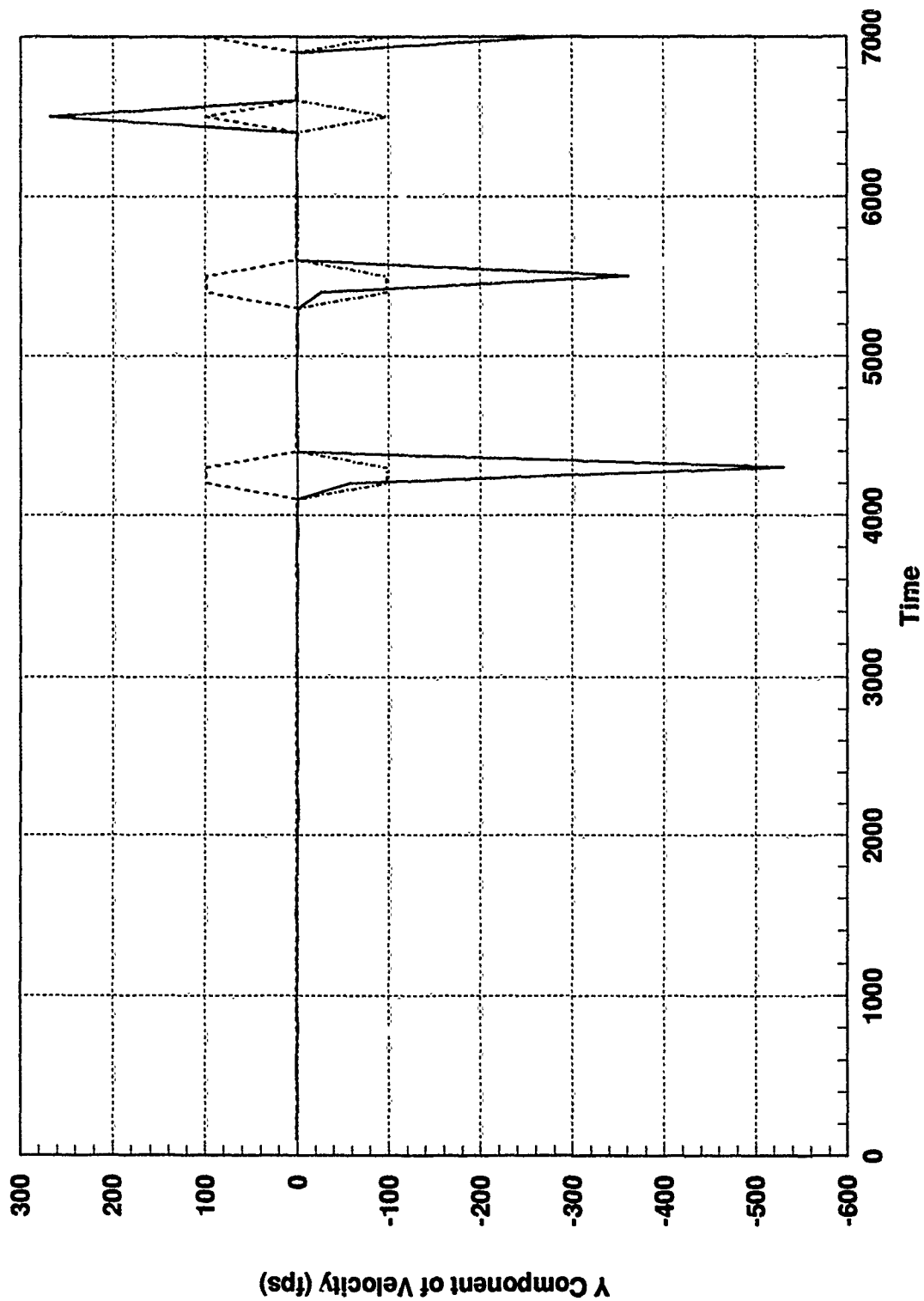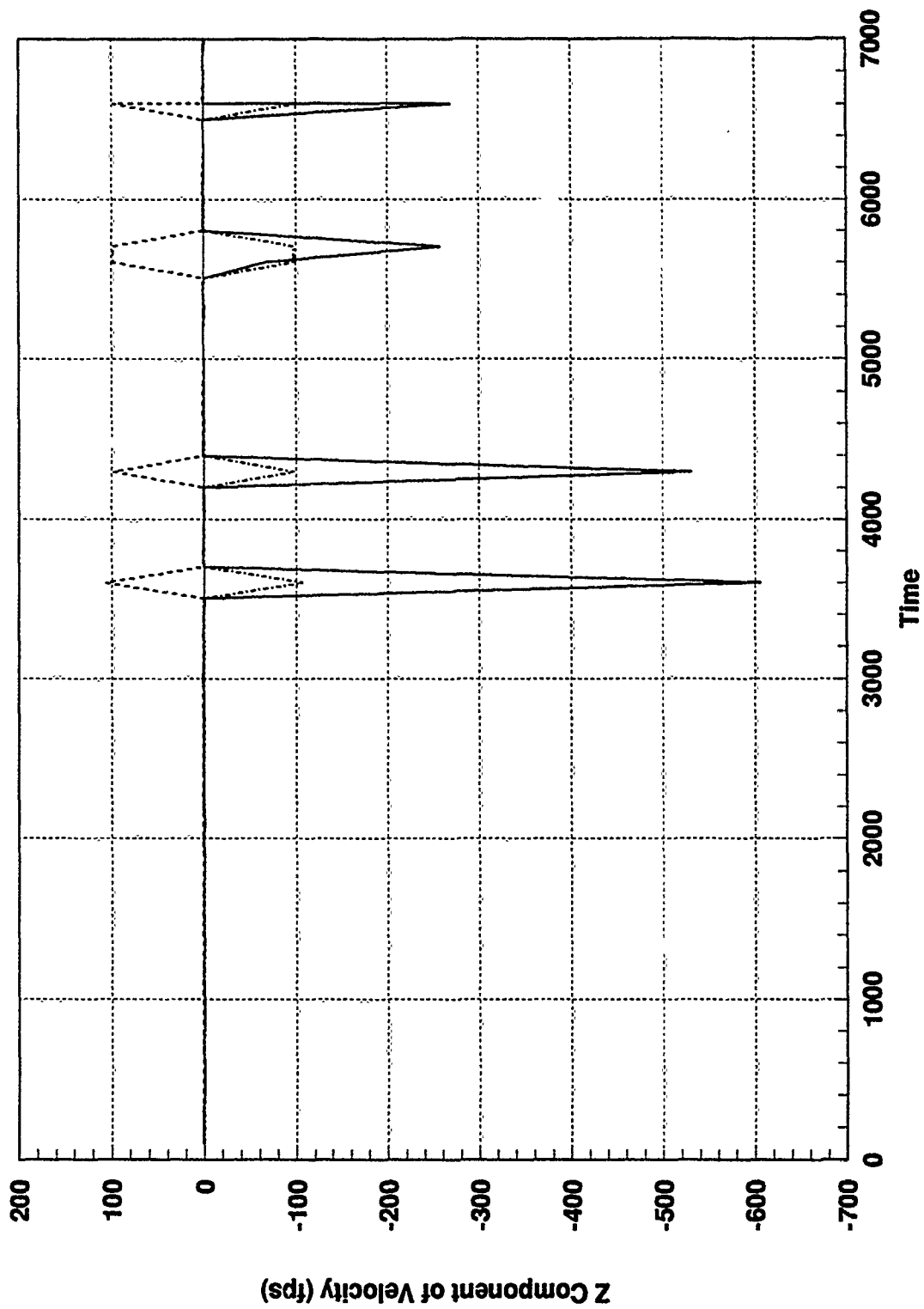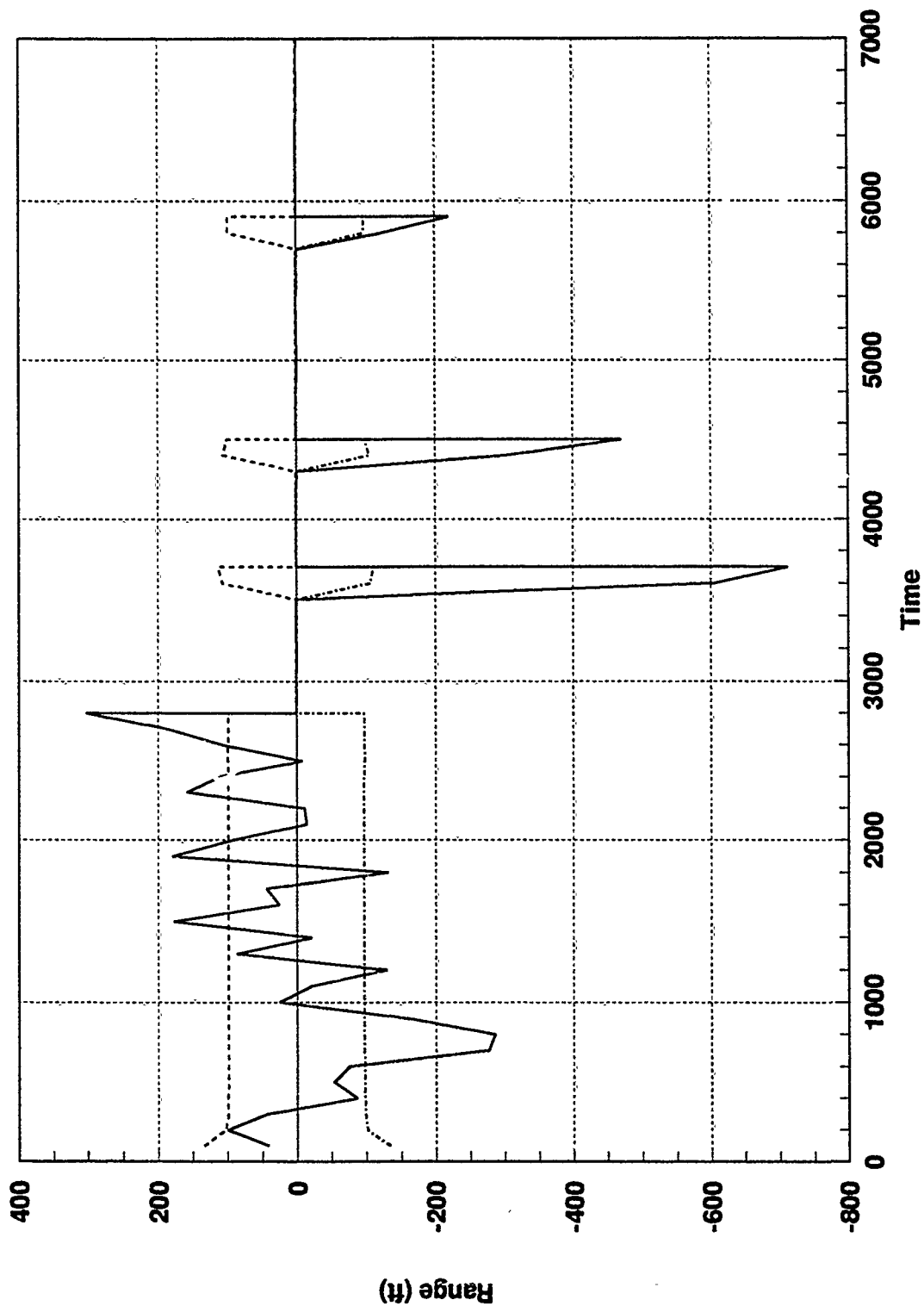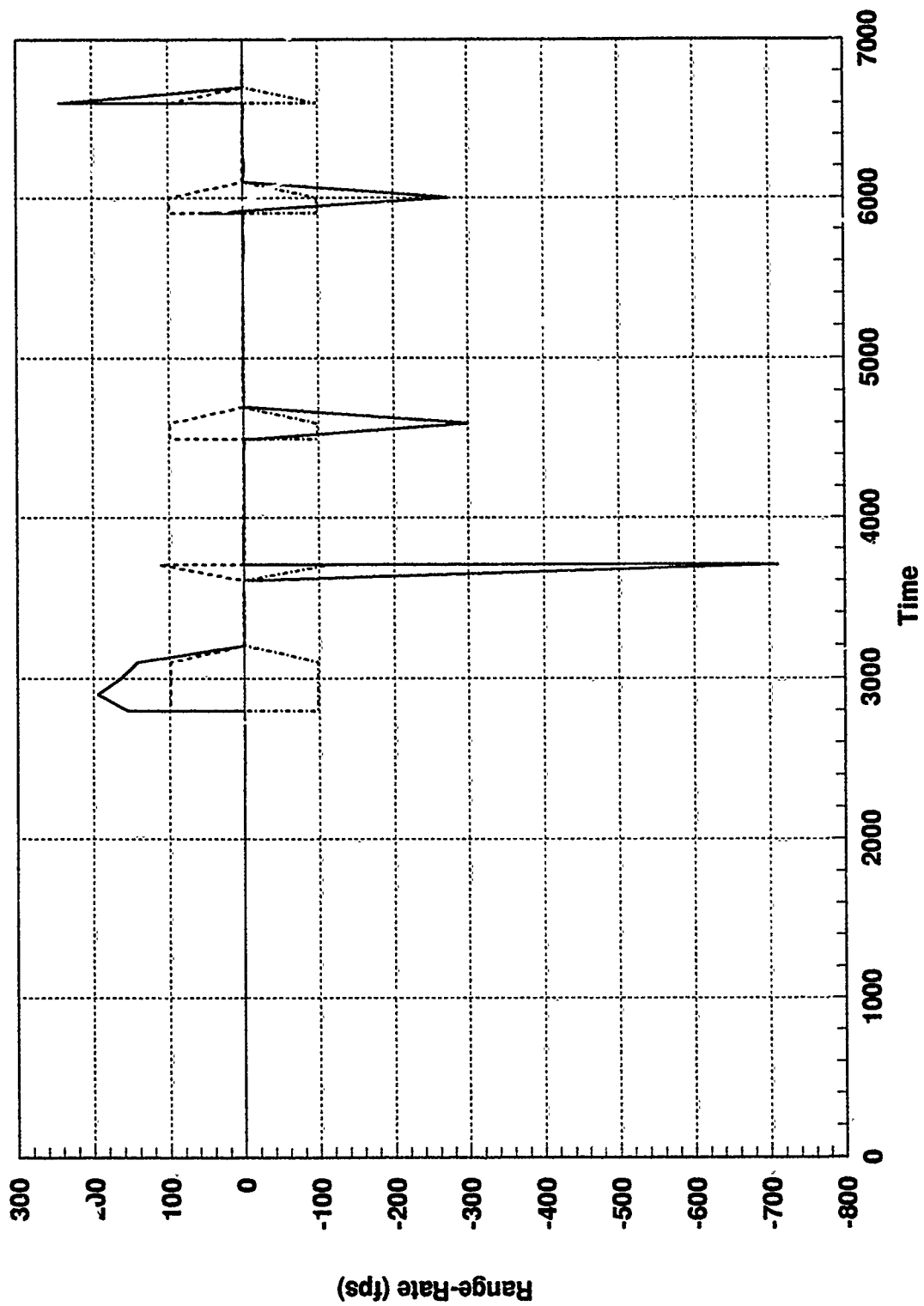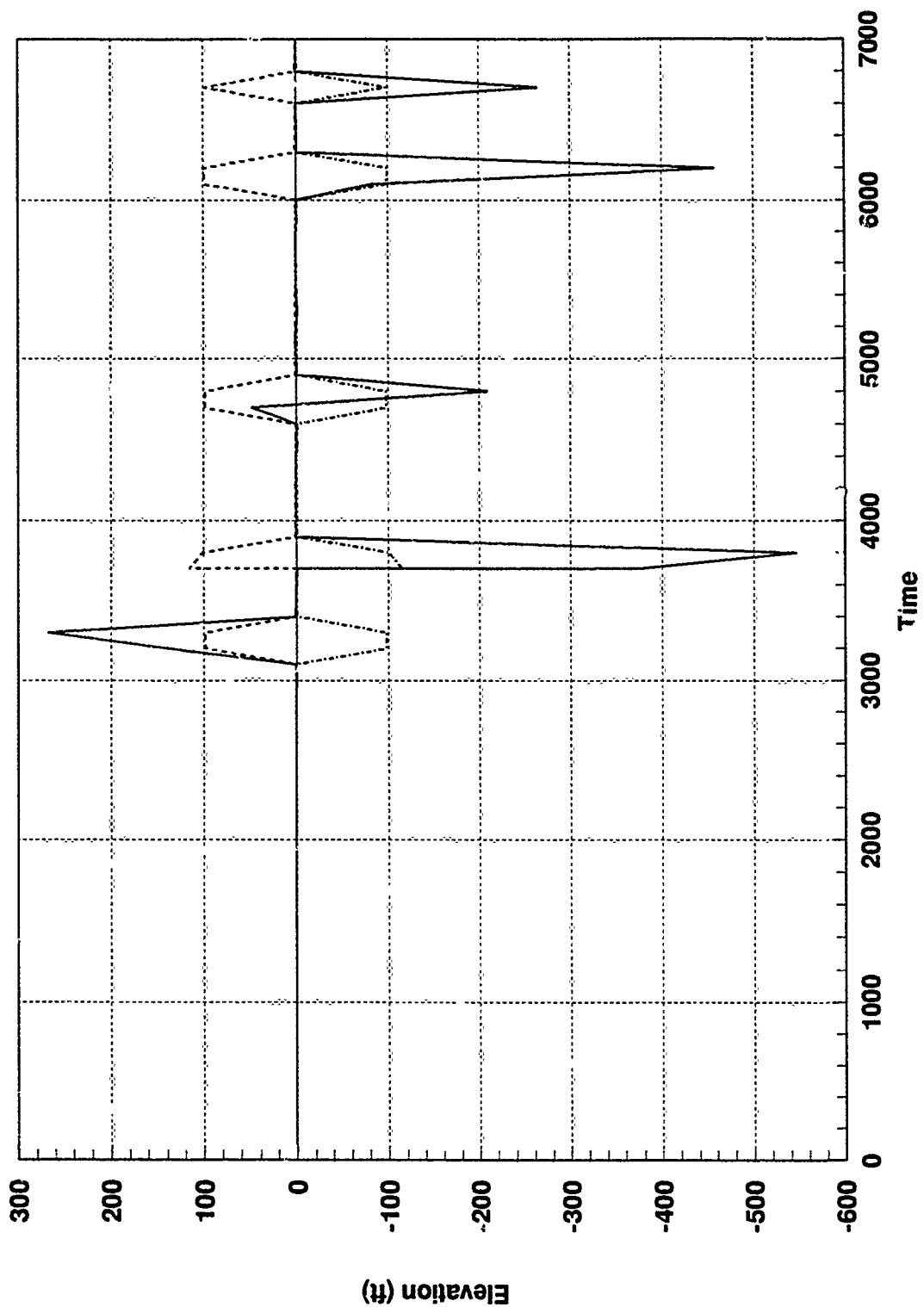Figure H.2 GPS Sat 1 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
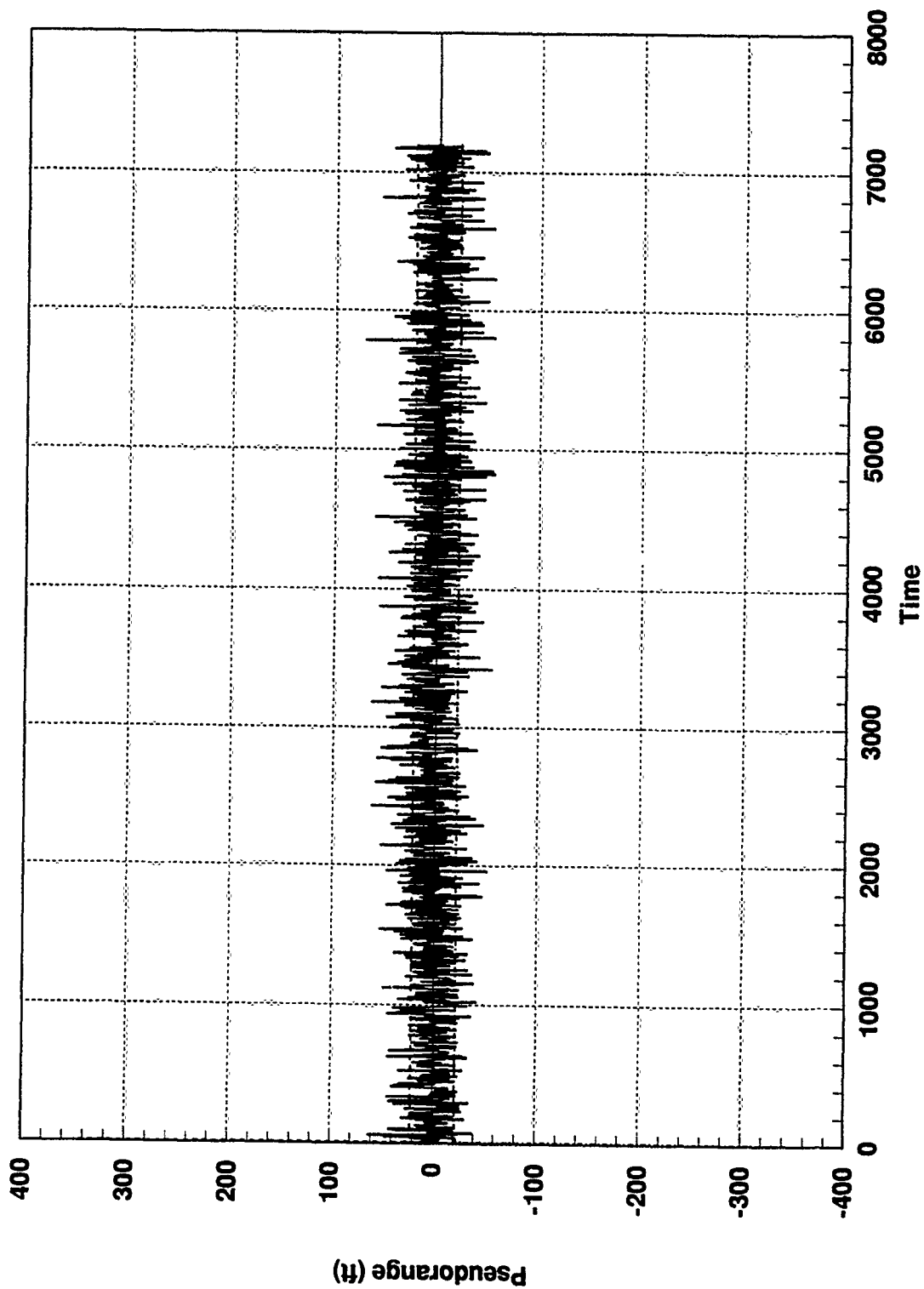
Figure H.3 GPS Sat 2 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
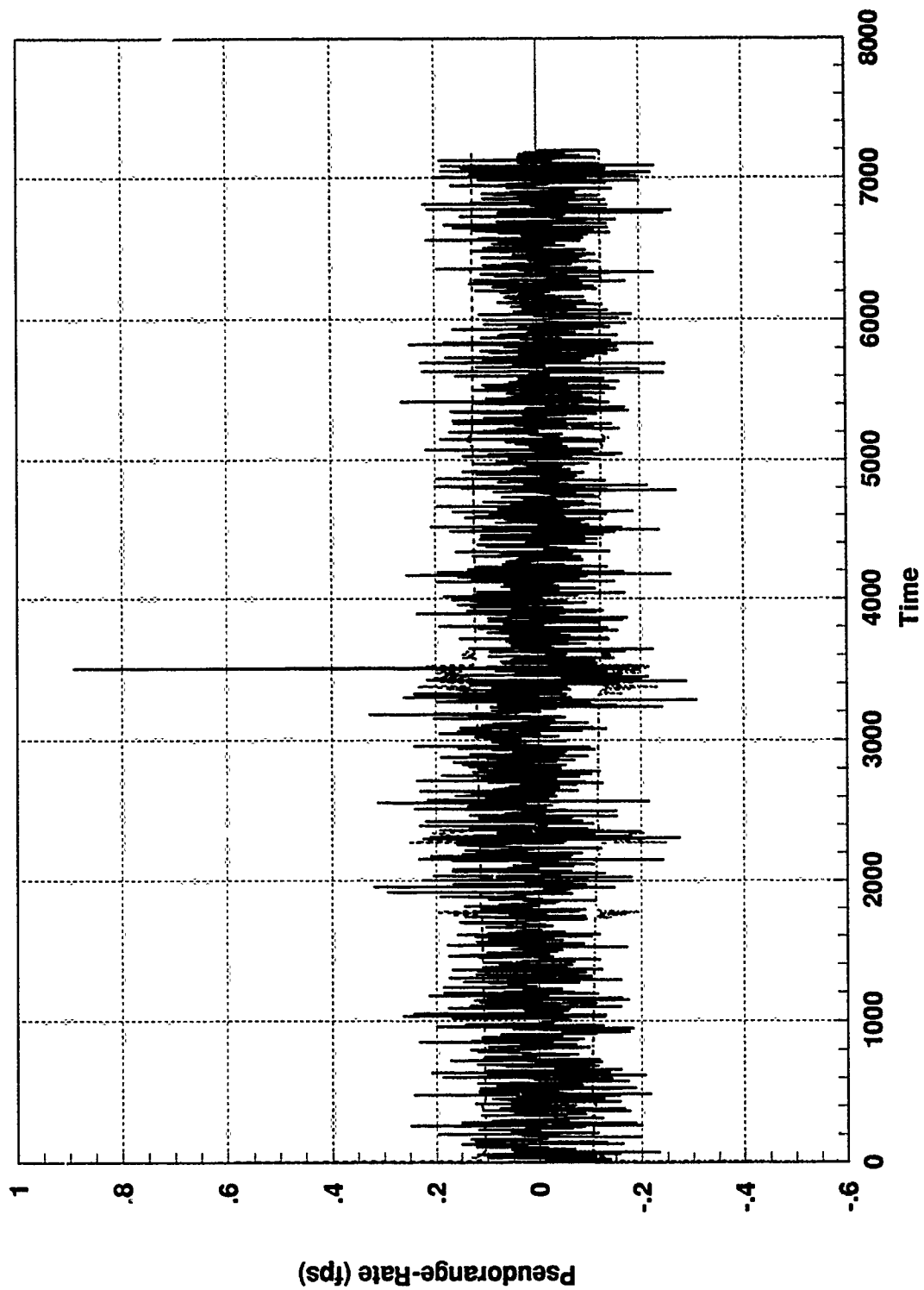
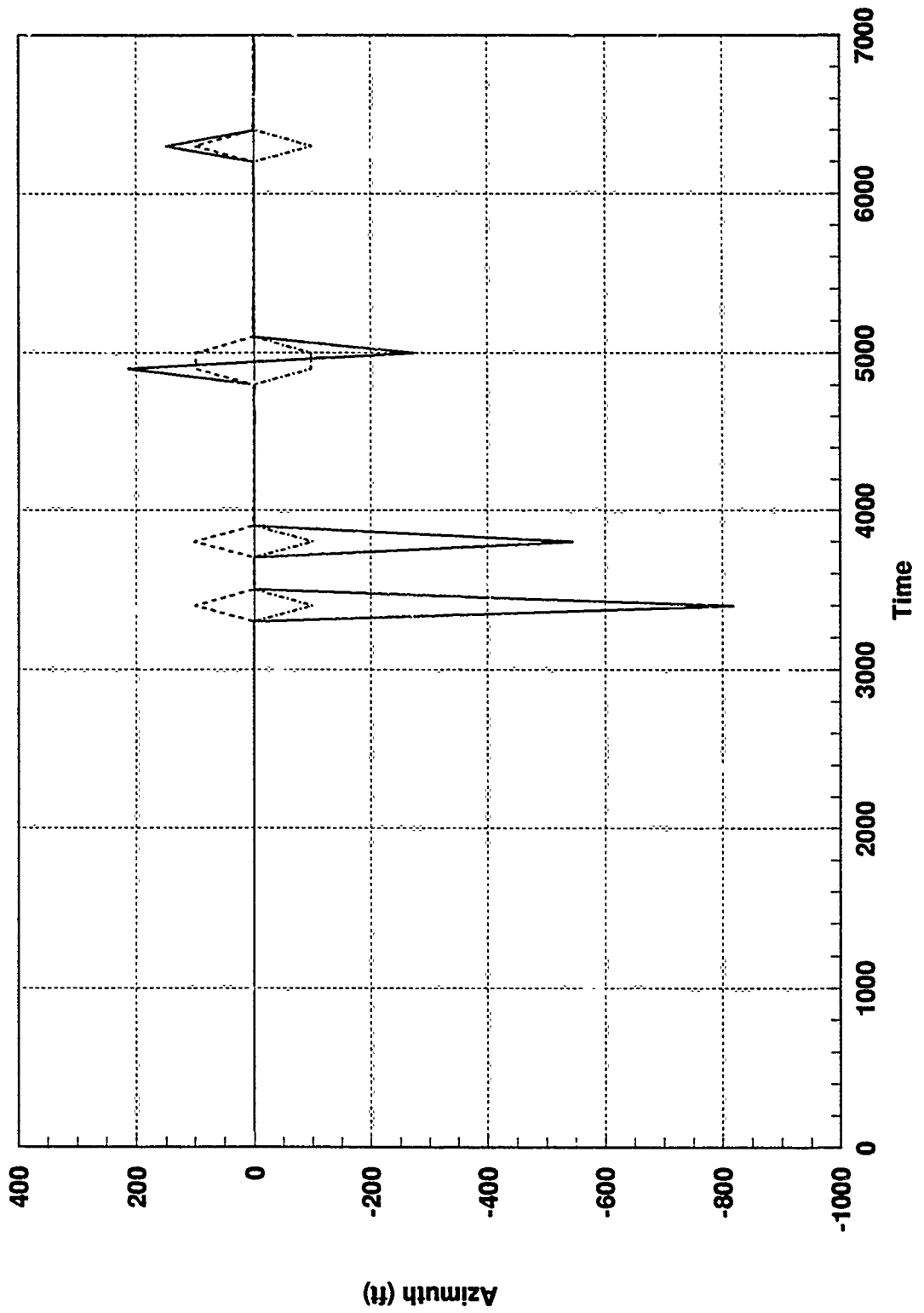Figure H.4 GPS Sat 2 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
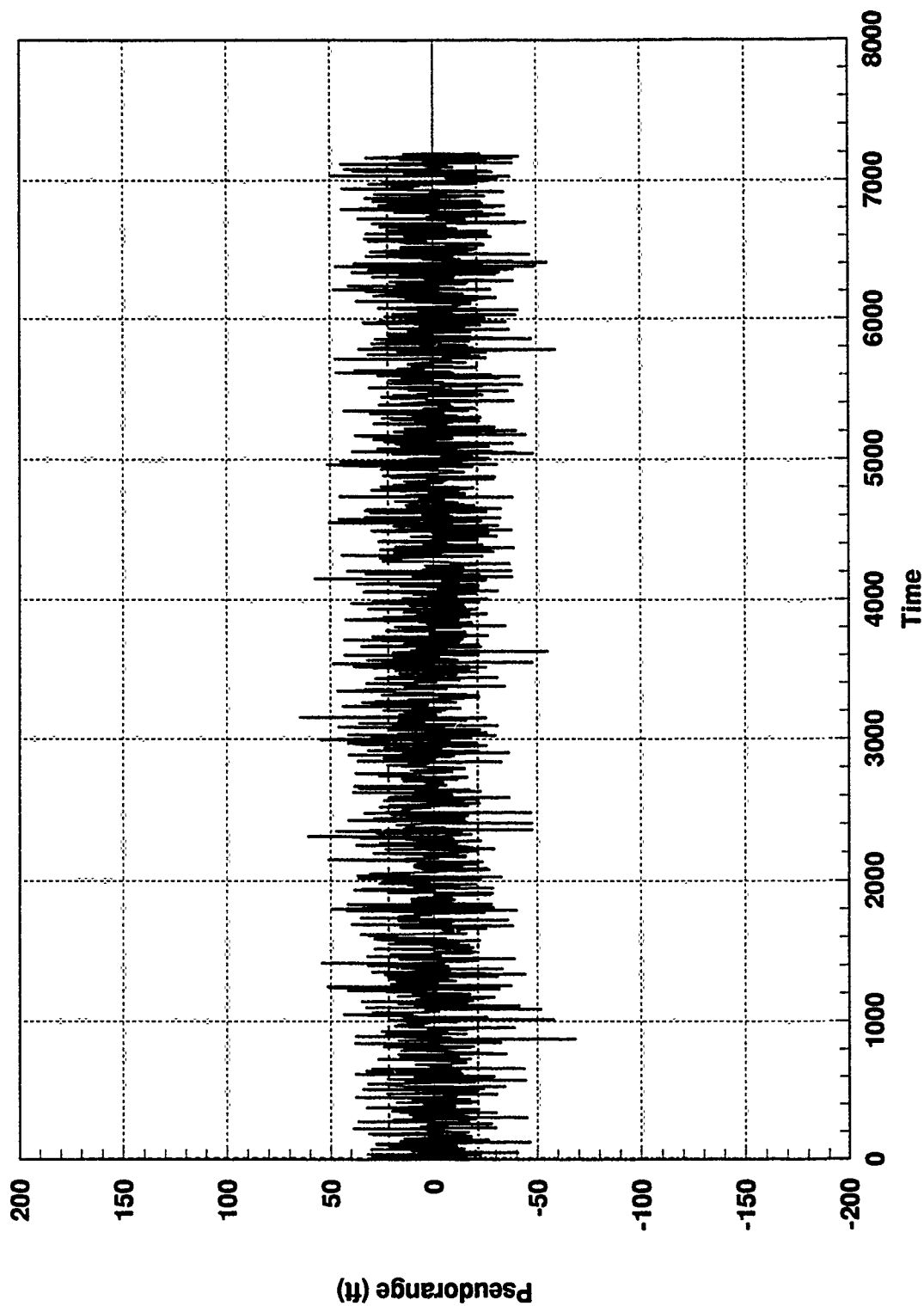
Figure H.5 GPS Sat 3 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
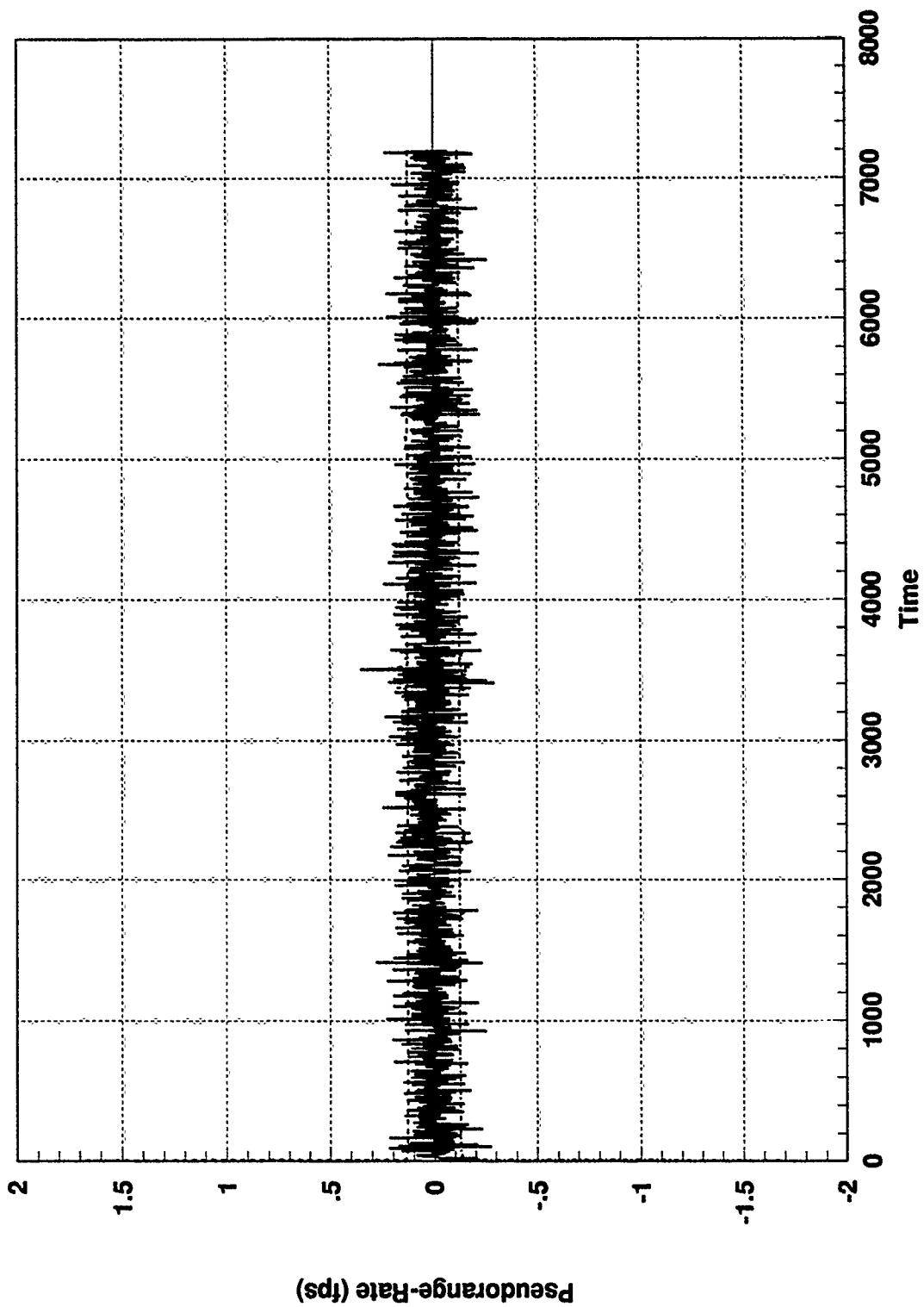
Figure H.6 GPS Sat 3 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
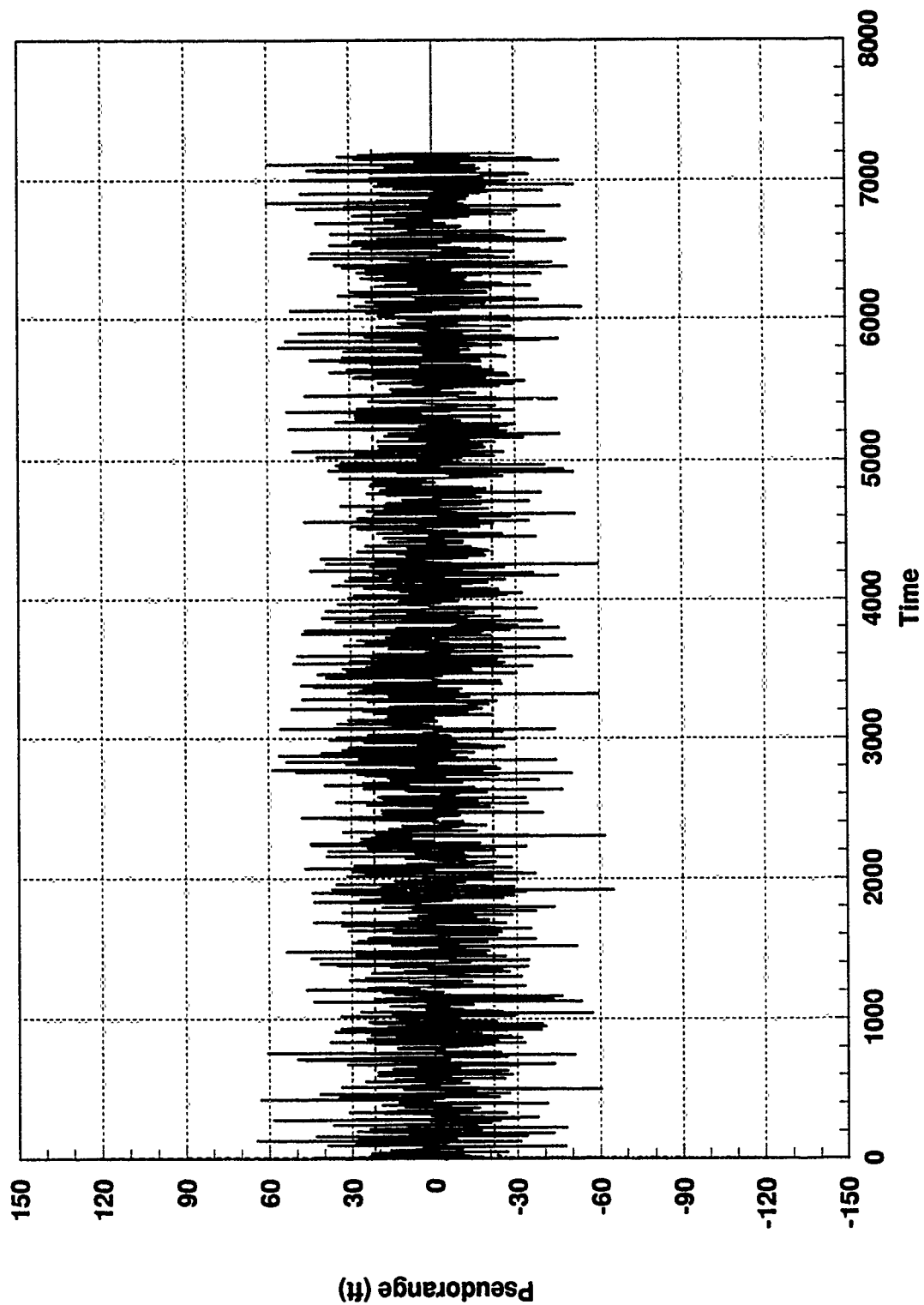
Figure H.7 GPS Sat 4 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
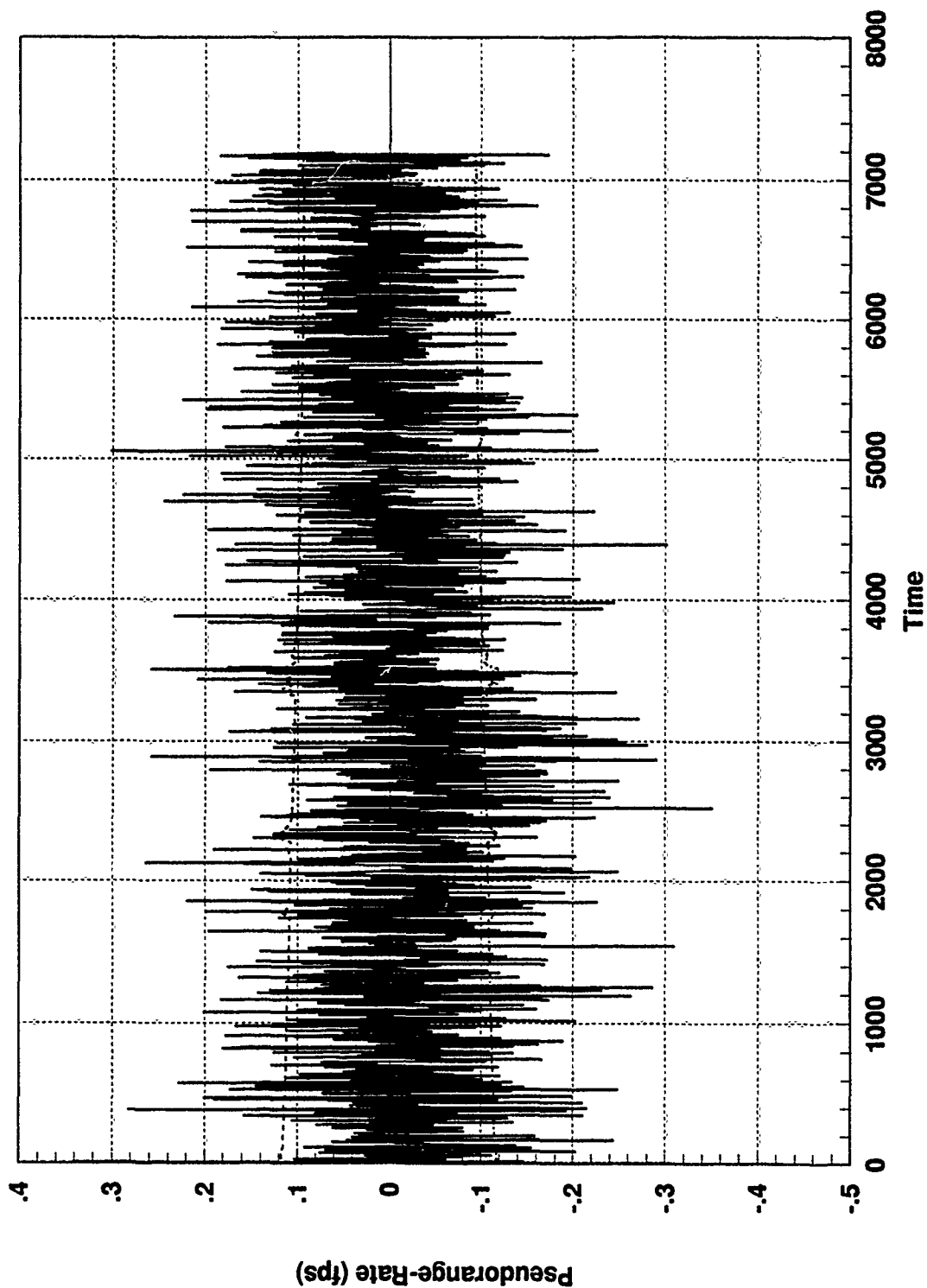
Figure H.8 GPS Sat 4 Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
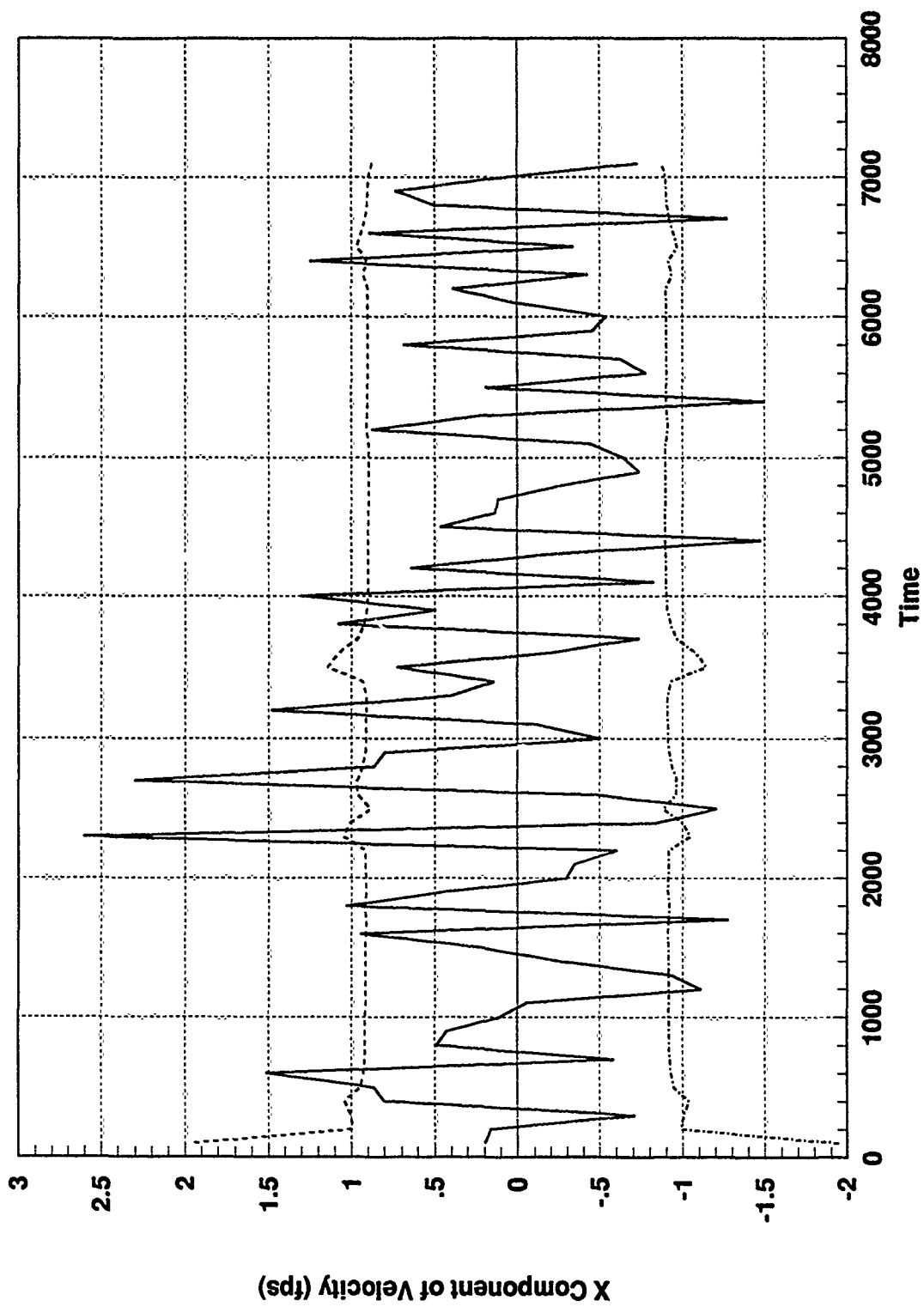
Figure H.9 SARPVU Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

X Component of Velocity (fps)

Time

Y Component of Velocity (fps)

Time
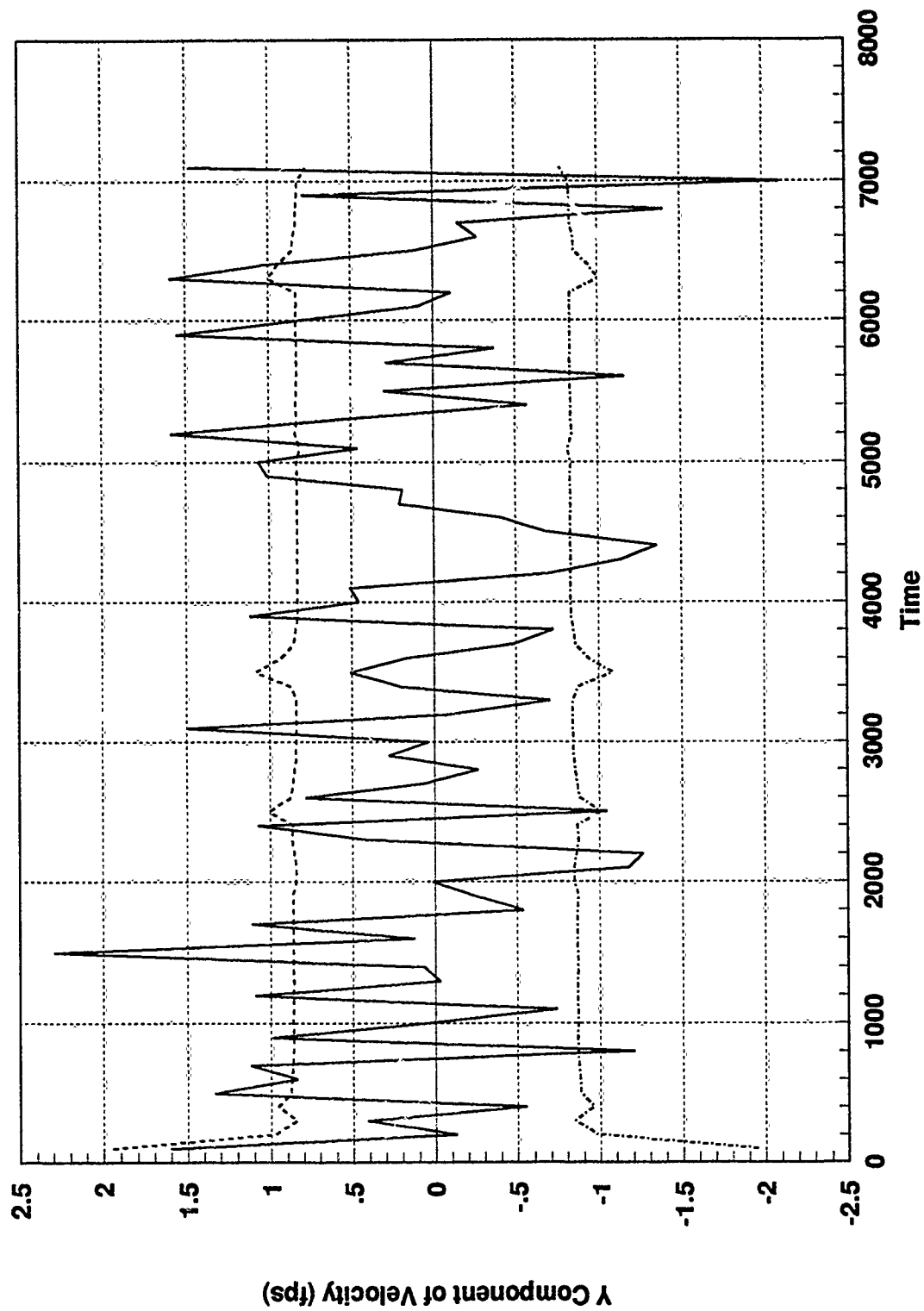
Figure H.10 SARPVU Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
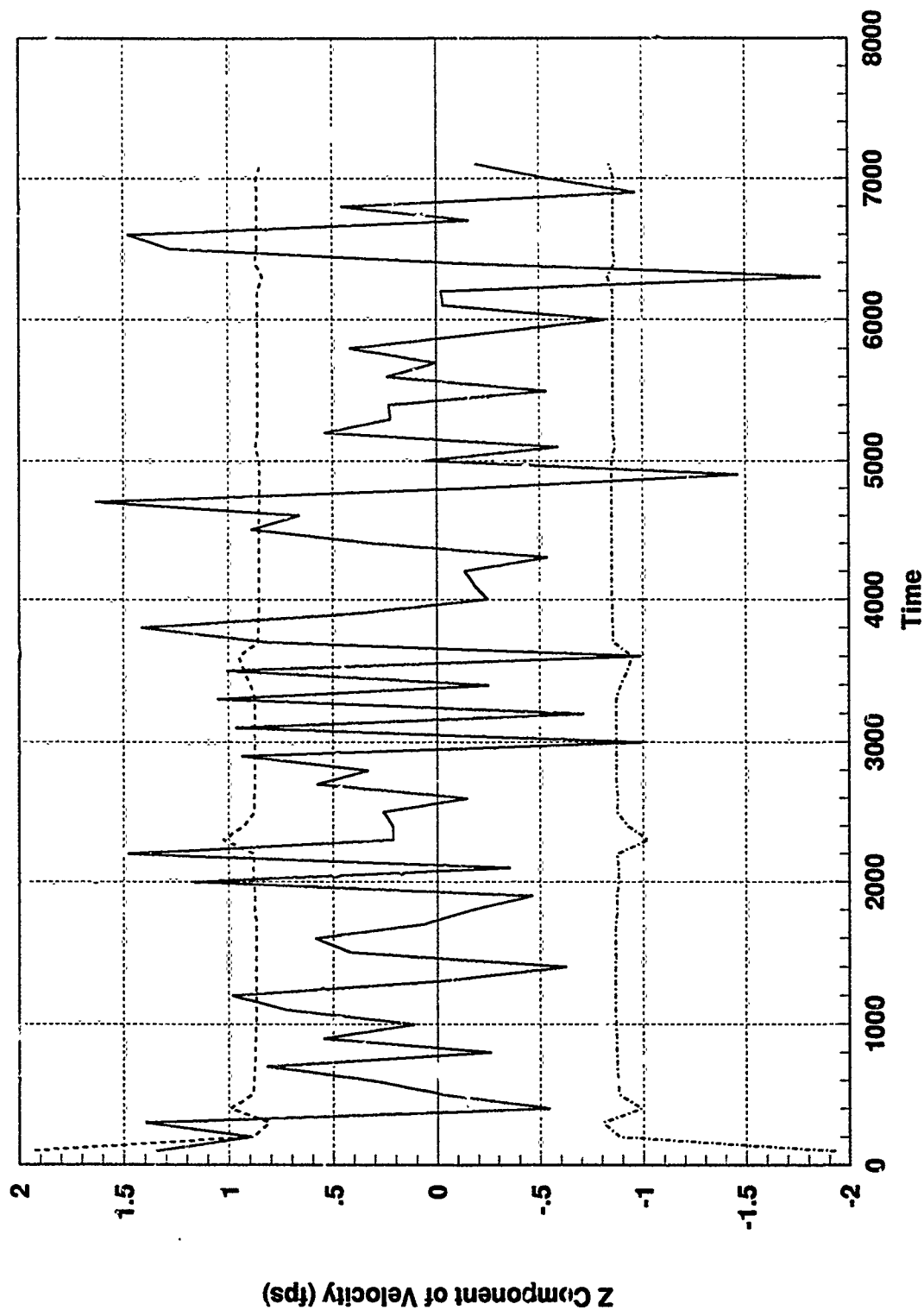
Figure H.11 SARPVU Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

Z Component of Velocity (fps)

Time
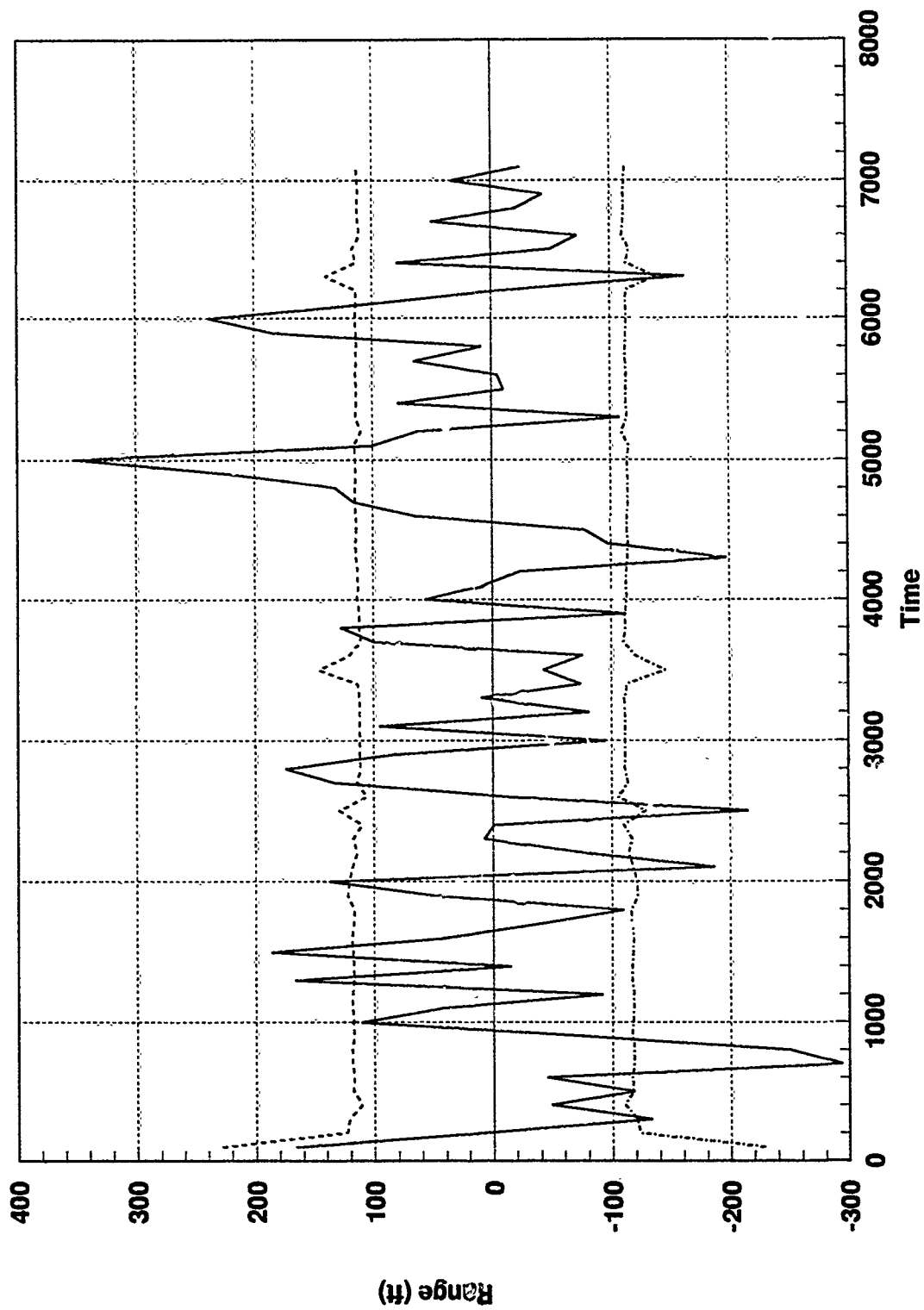
Figure H.12 SAREO Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
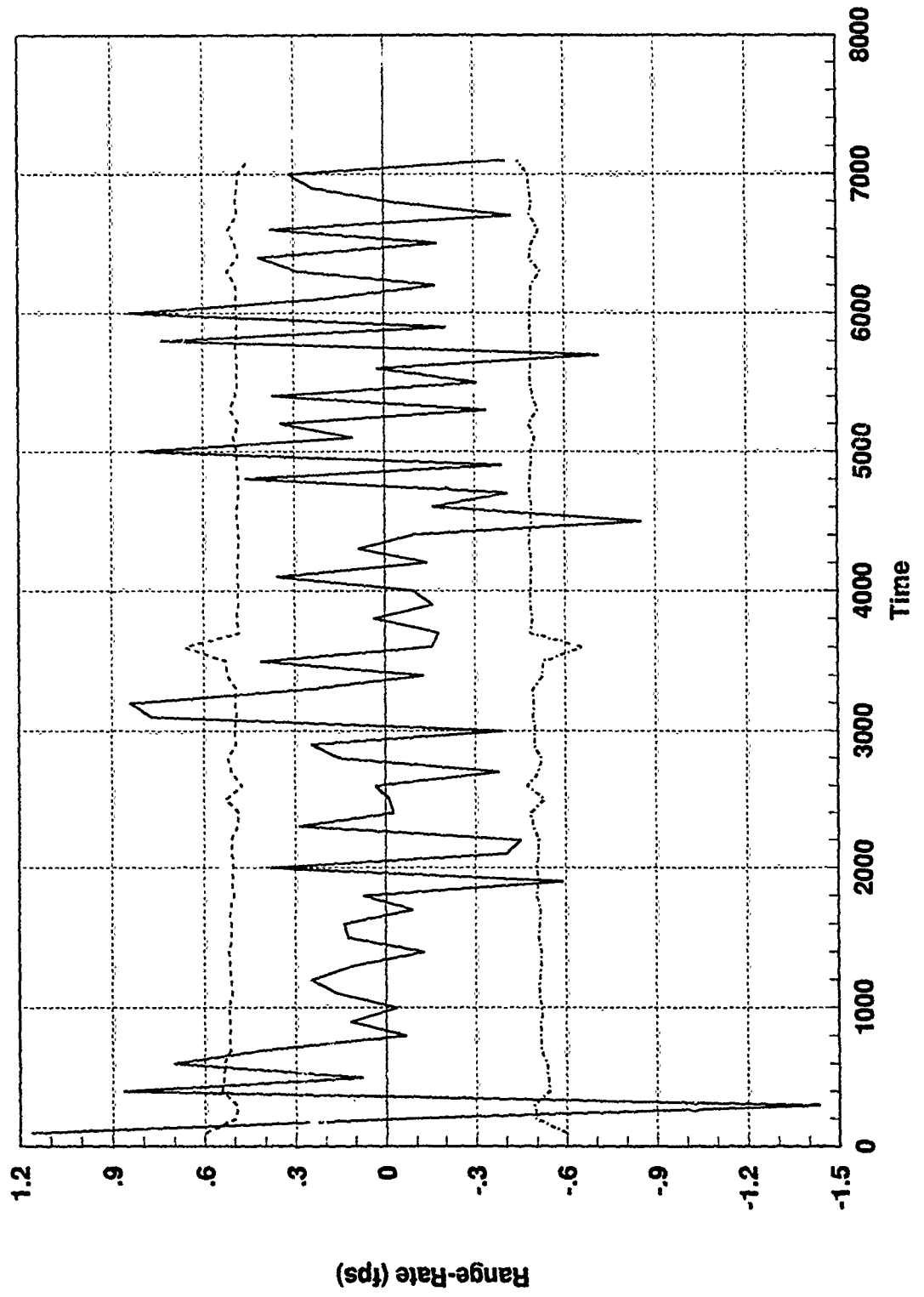
Figure H.13 SAREO Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
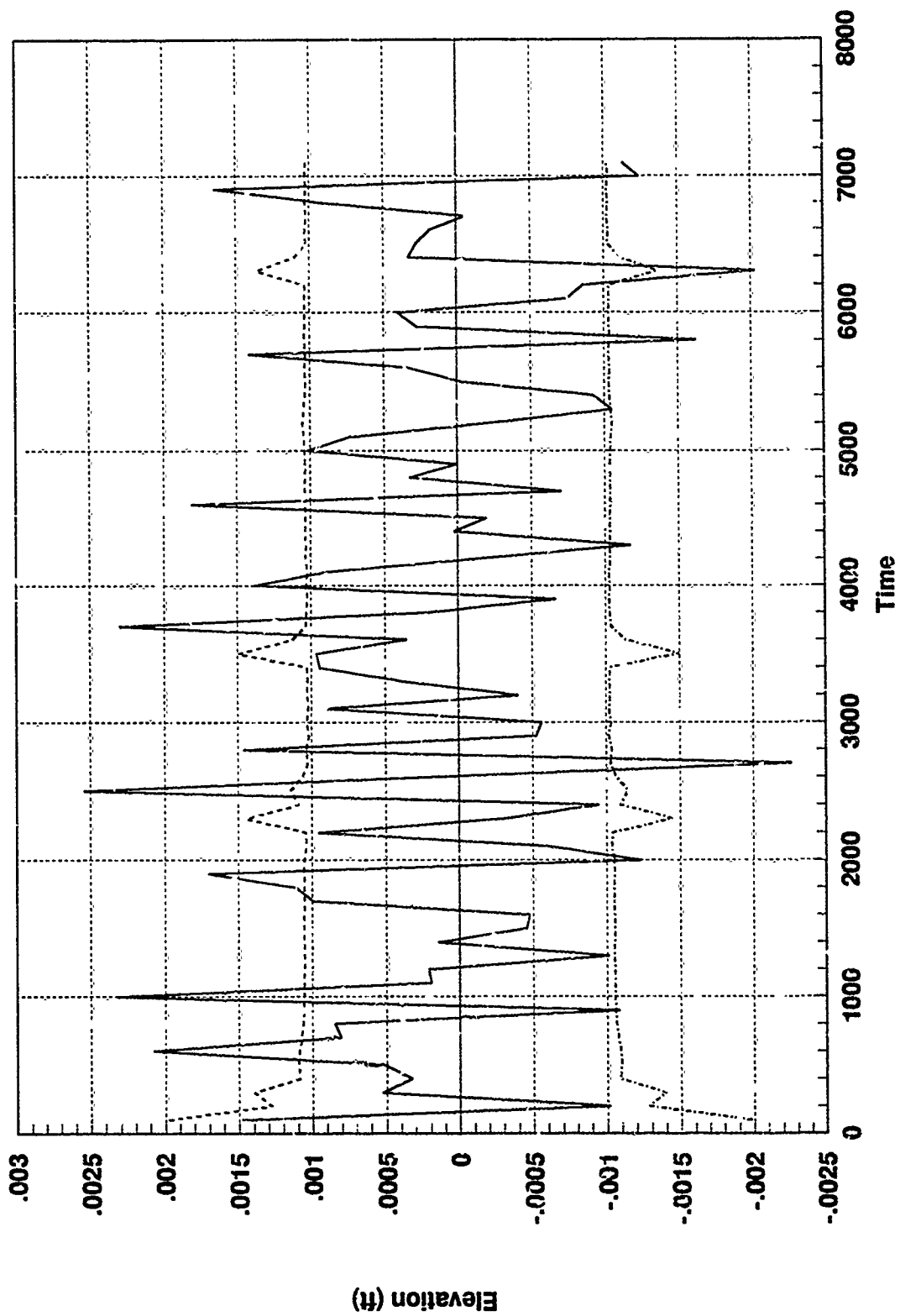
Figure H.14 SAREO Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias

H-15
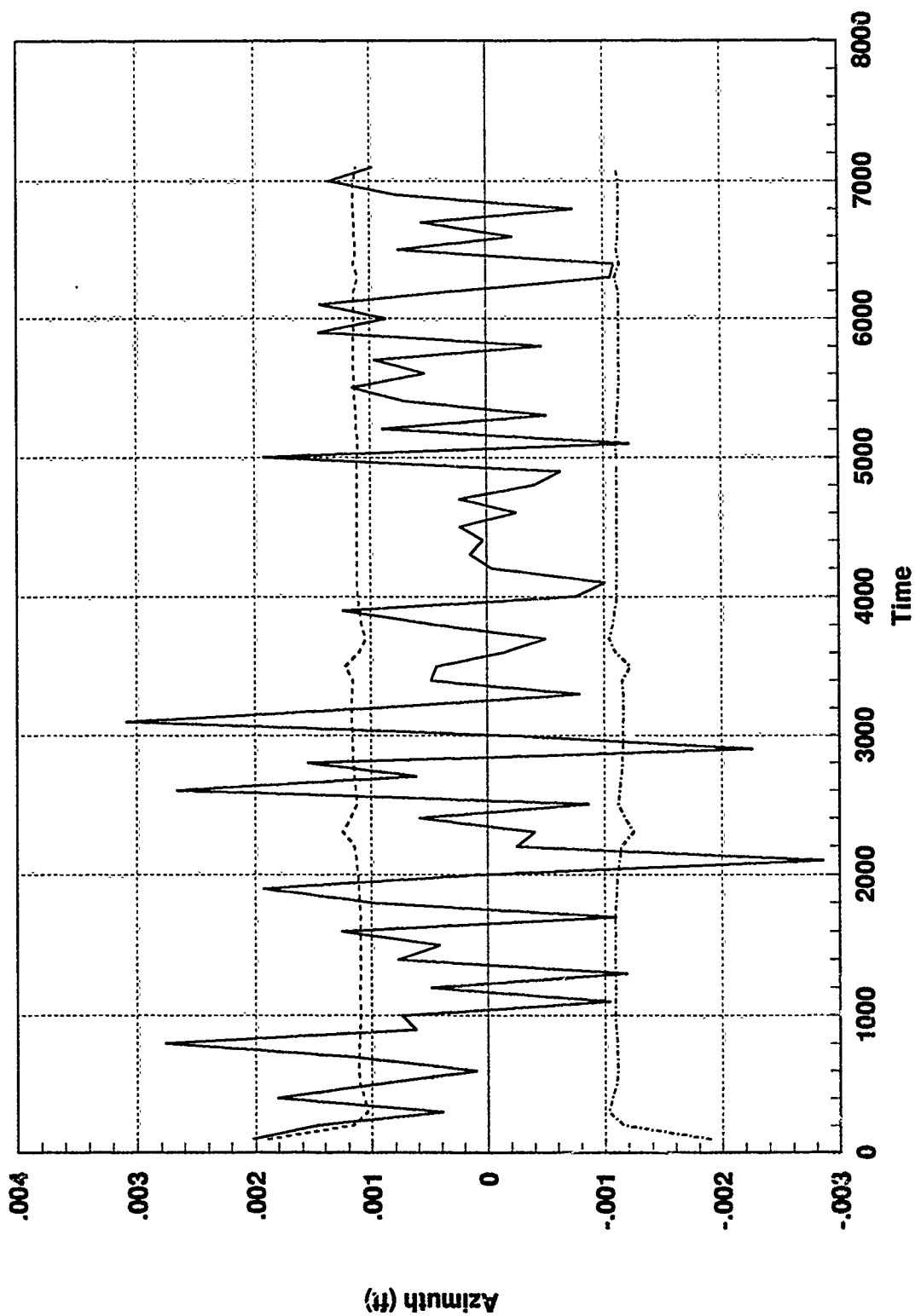
Figure H.15 GAREO Residual and One-Sigma Bound, GPS Ramp & TAN Constant Bias
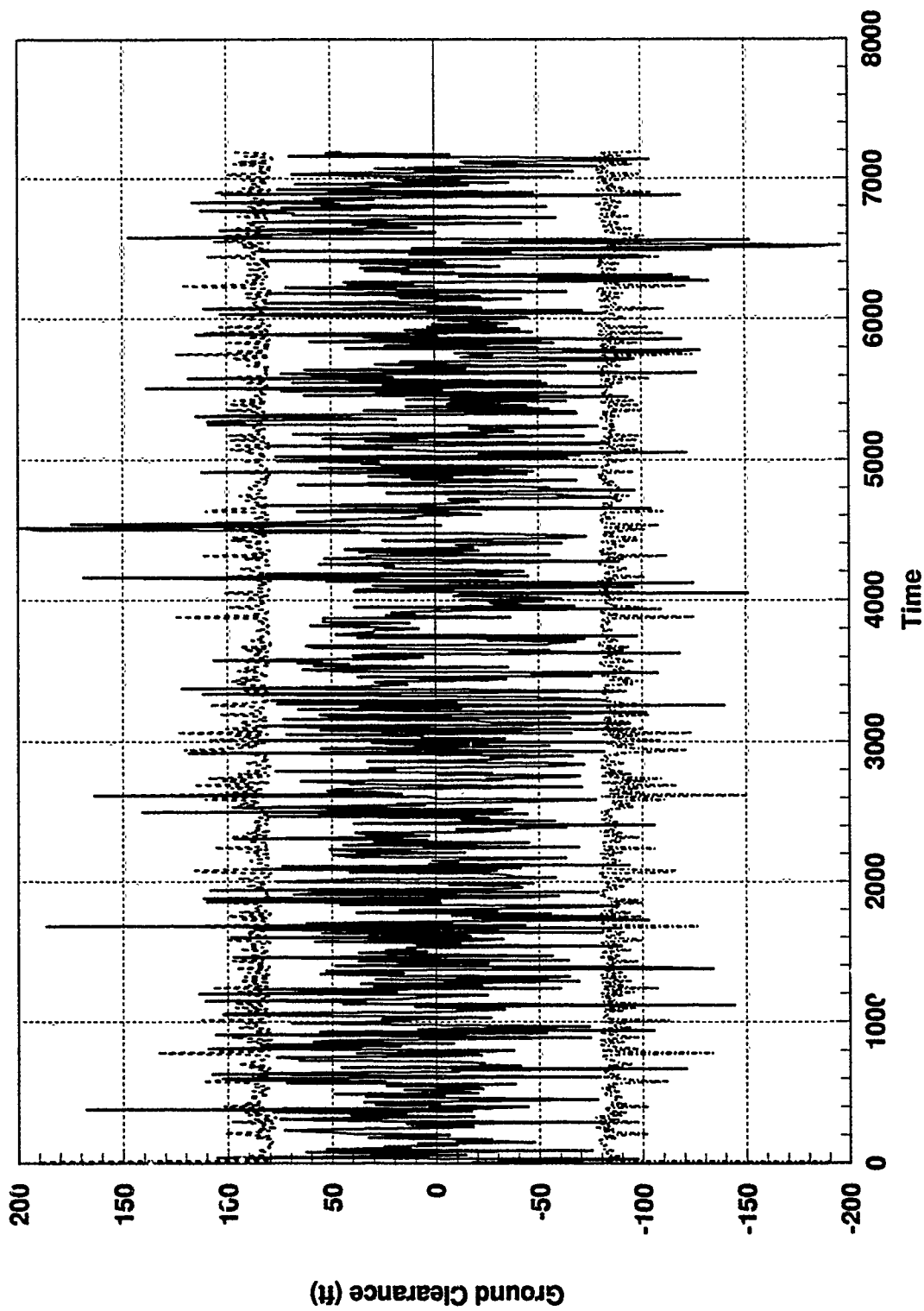
Figure H.16 Federated Filter TAN Residual, GPS Ramp & TAN Bias

Appendix I: Centralized and Federated Filter

Chi-Squared Detection Algorithm Plots

GENERAL INFORMATION

The series of plots in this appendix allows for a comparison of the fault detection considerations of this thesis. Establishing the specific characteristics of the centralized and federated filter residual behavior during simulated failure-mode conditions is the intent.

Each of the plots contained in this appendix are obtained from an application of the chi-squared algorithm on the Monte Carlo simulations from DKFSIM Version 1.1. All chi-squared values are recorded and computed for the entire 7200 second duration of the simulation.

There are 6 plots, two for the GPS, two for the SAR, and two for the TAN sensor measurements. The first set of detection plots applies to the centralized filter residual output data files corresponding to the GPS, SAR, and TAN sensors. The second set of detection plots applies to the federated filter output data files corresponding to the GPS, SAR, and TAN sensor dedicated filter residuals. A ramp bias failure was added to the PseudoRange-Rate residual for GPS Satellite #1 for both filter designs. Both sets of data include the effects of the constant bias added to the TAN sensor filter residual.

Each plot has the appropriate description. A listing of the titles of each these figures is found in the List of Figures at the beginning of this document.
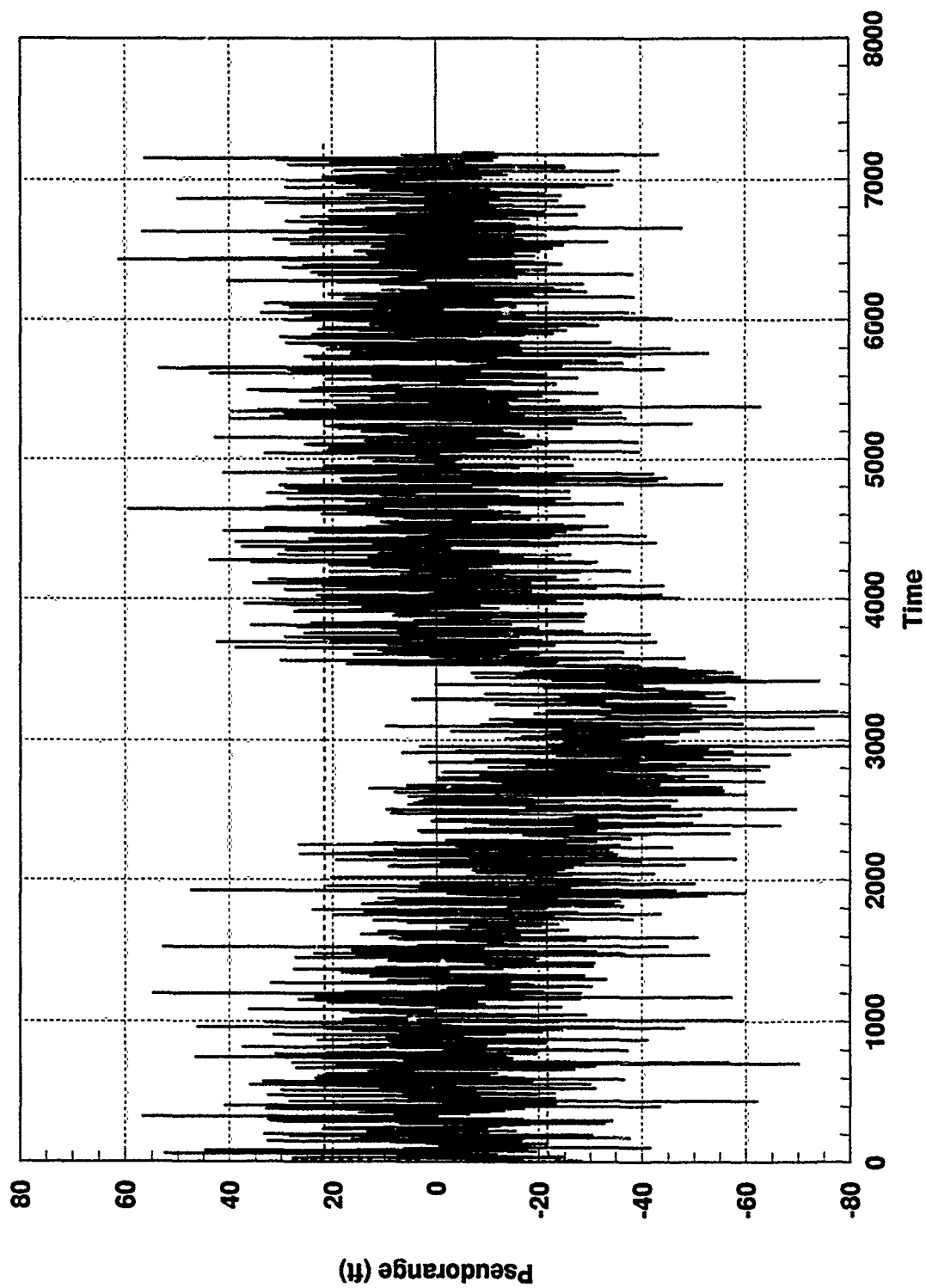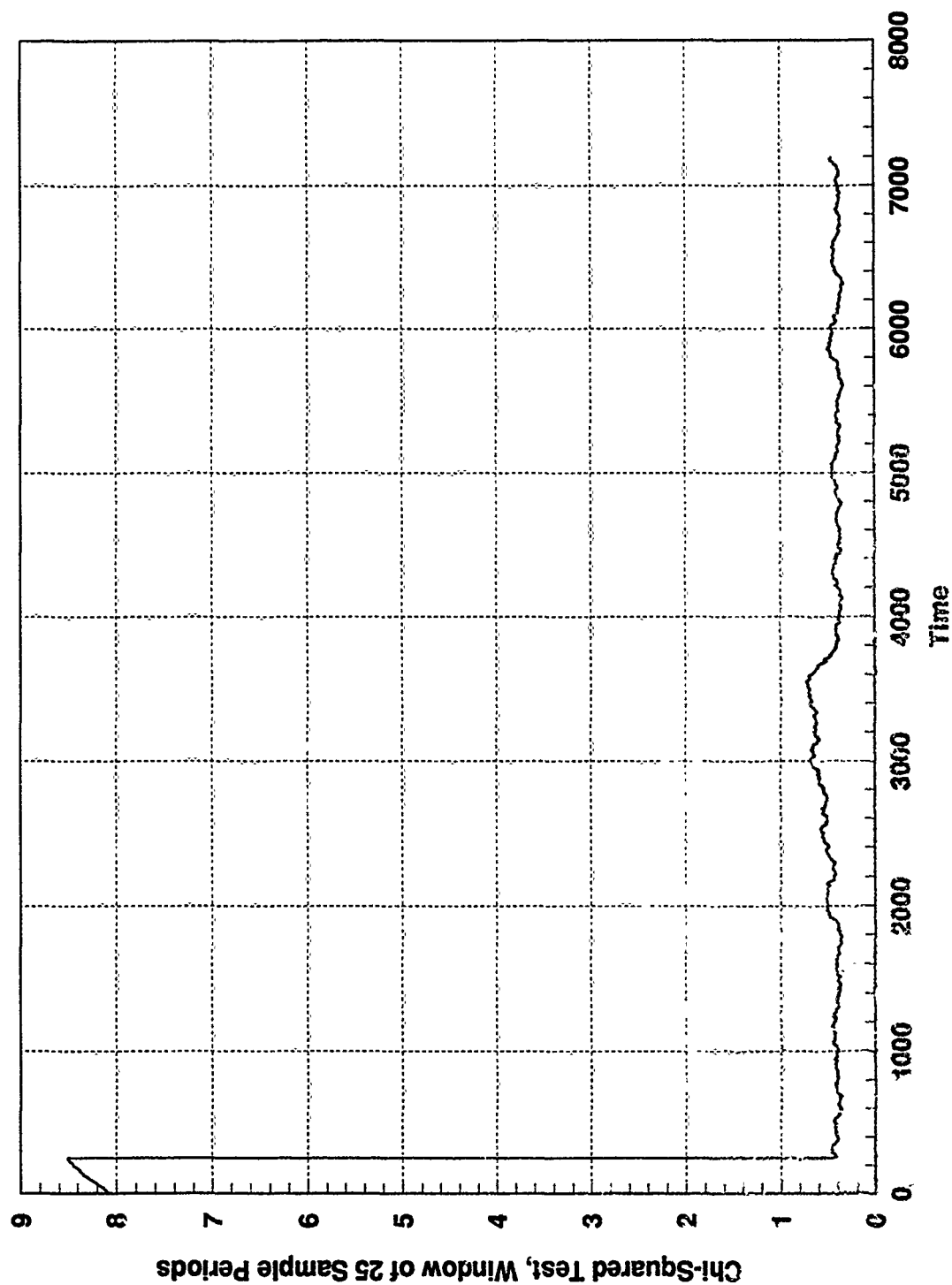
Figure I.1 Centralized Filter, GPS Residuals Chi-Squared Test

Figure I.2 Centralized Filter, SAR Residuals Chi-Squared Test

Chi-Squared Test, Window of 5 Sample Periods
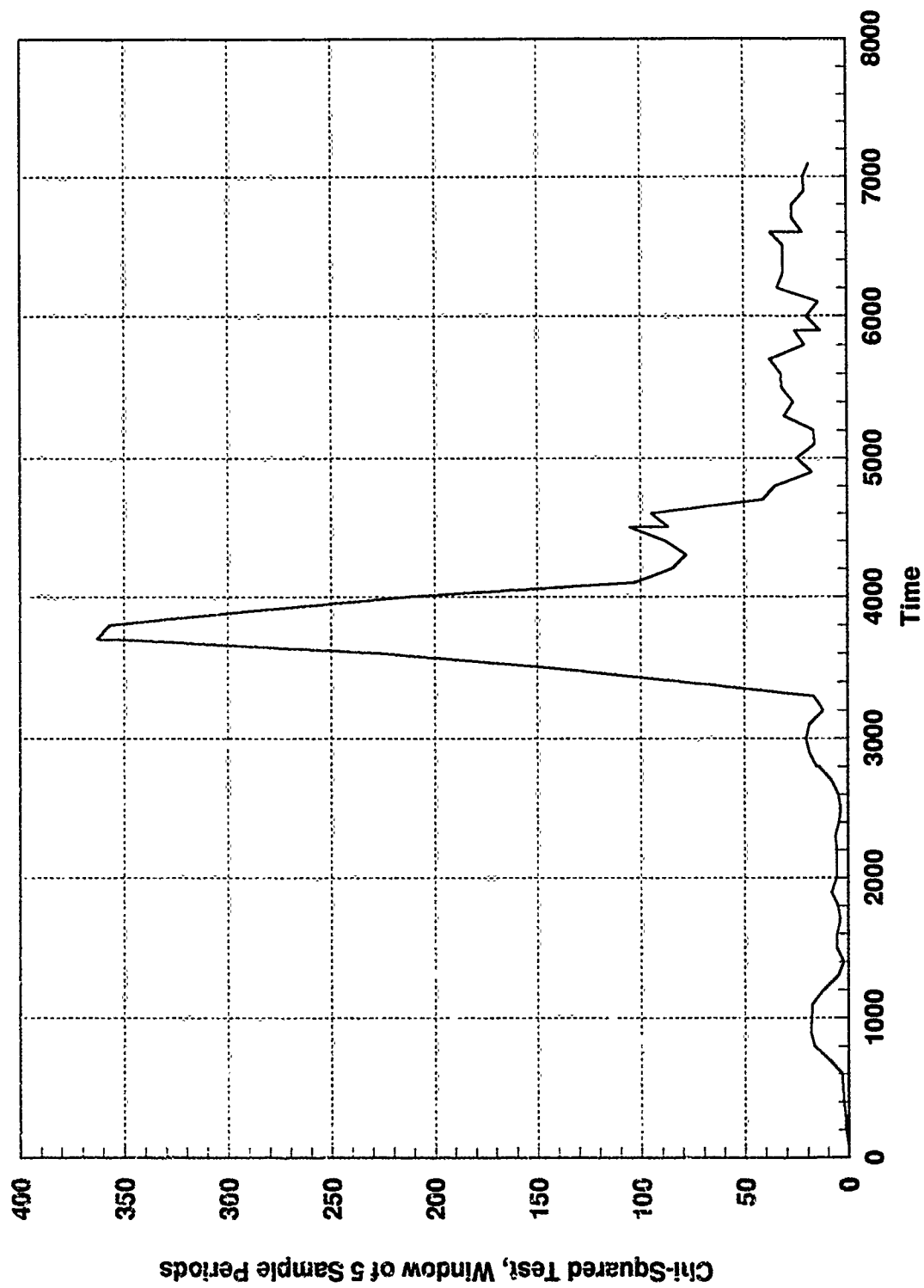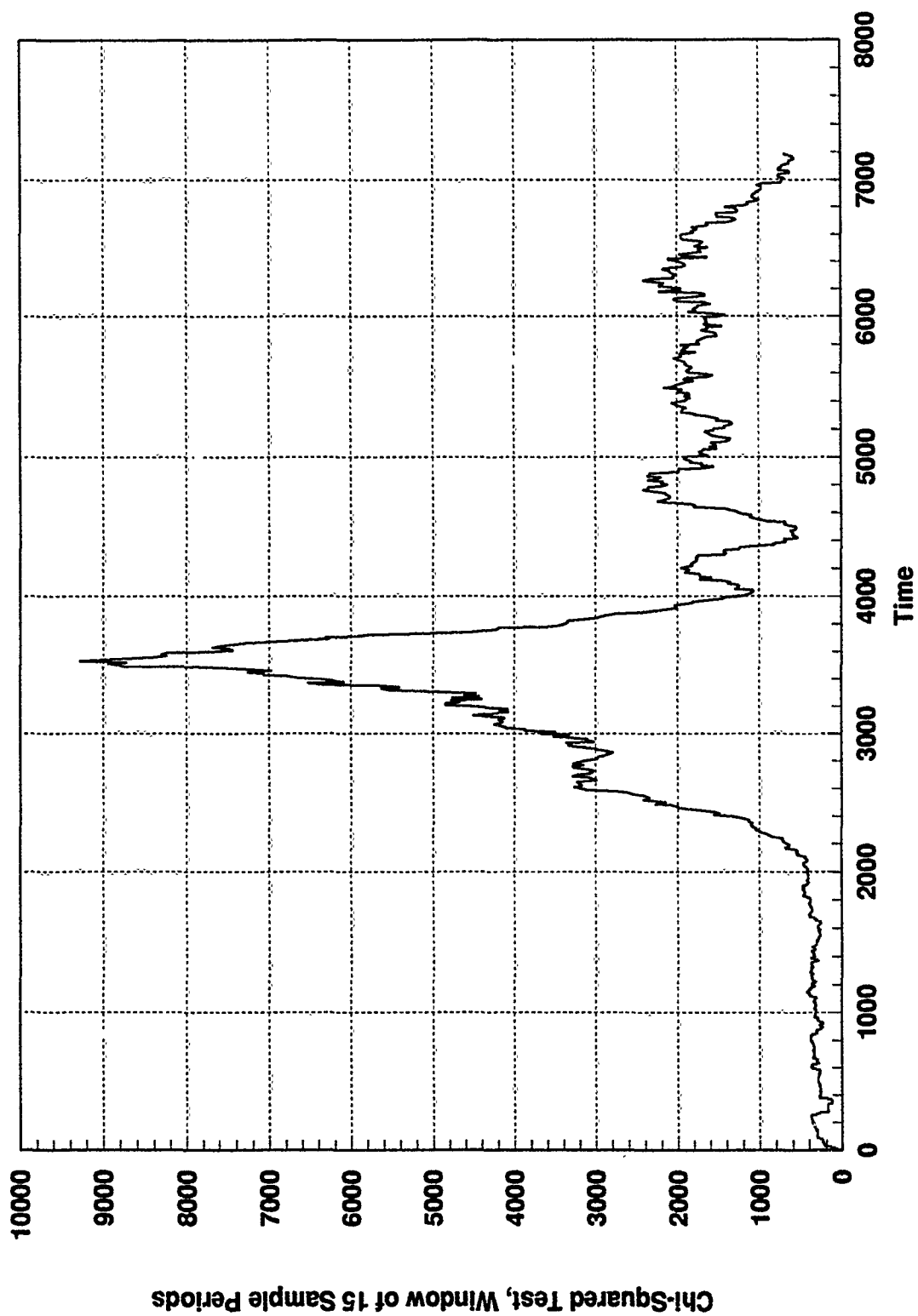
Figure I.3 Centralized Filter, TAN Residual Chi-Squared Test
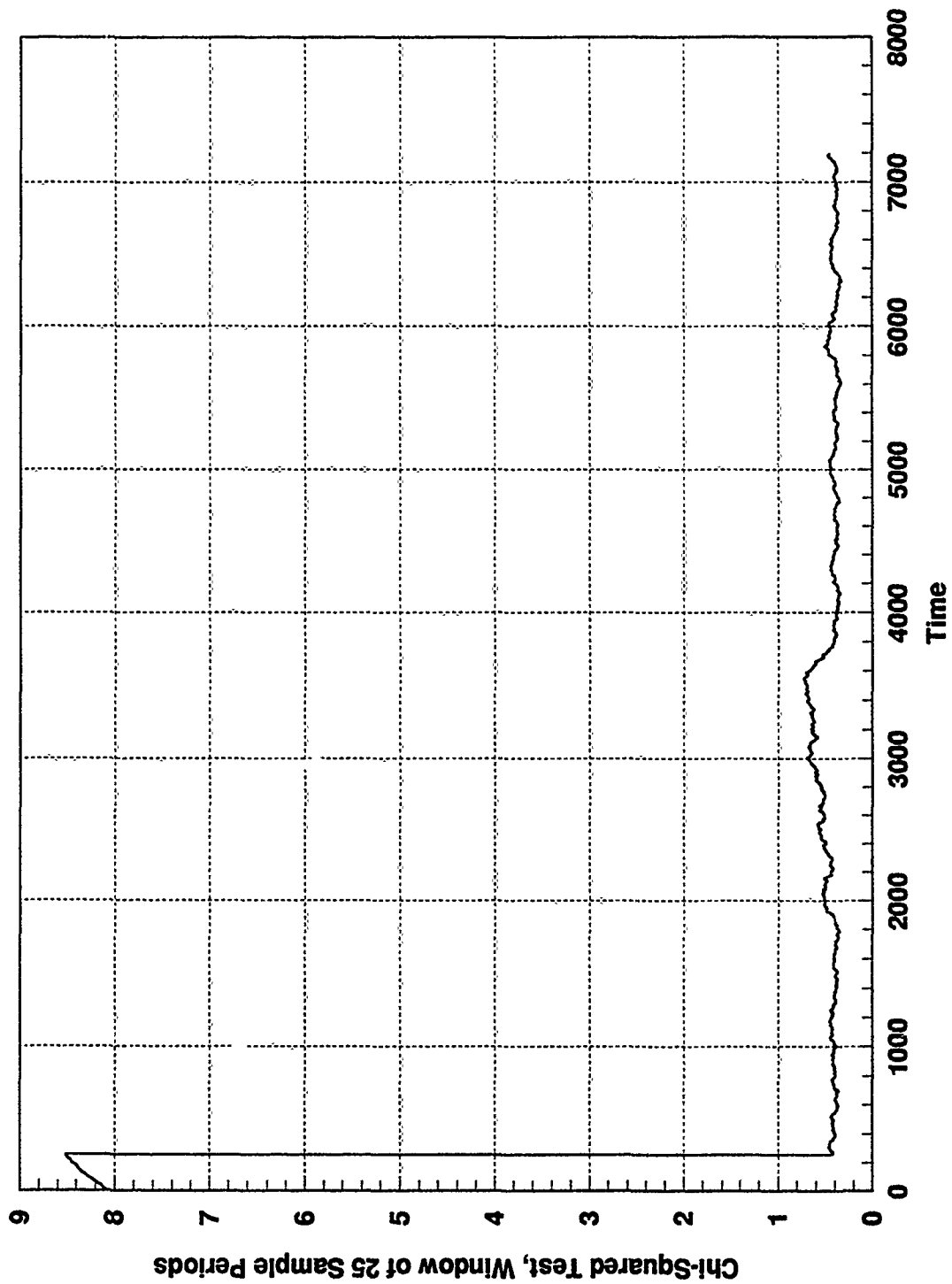
Figure I.4 Federated Filter, GPS Residuals Chi-Squared Test
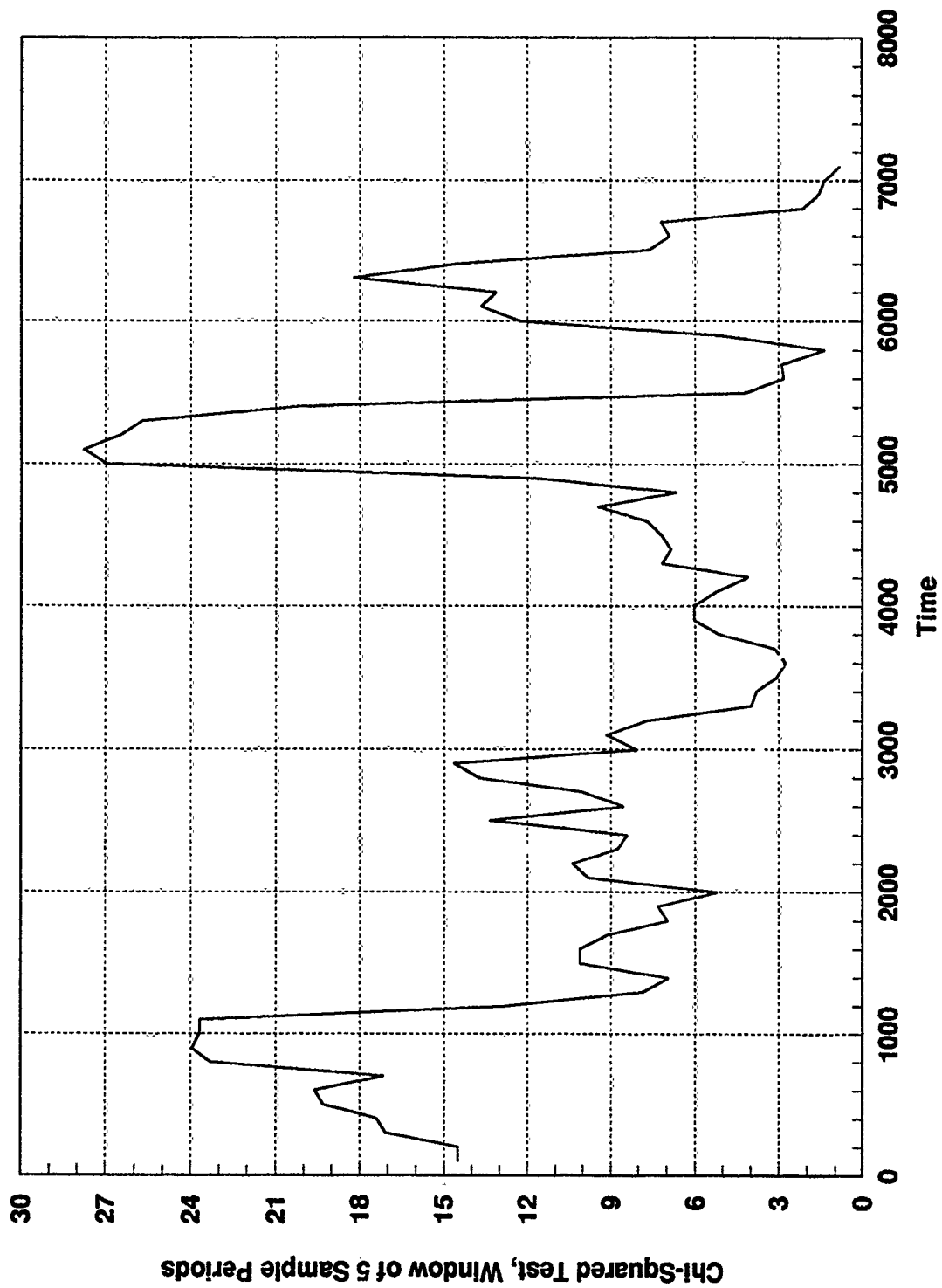
Figure I.5 Federated Filter, SAR Residuals Chi-Squared Test

Chi-Squared Test, Window of 5 Sample Periods
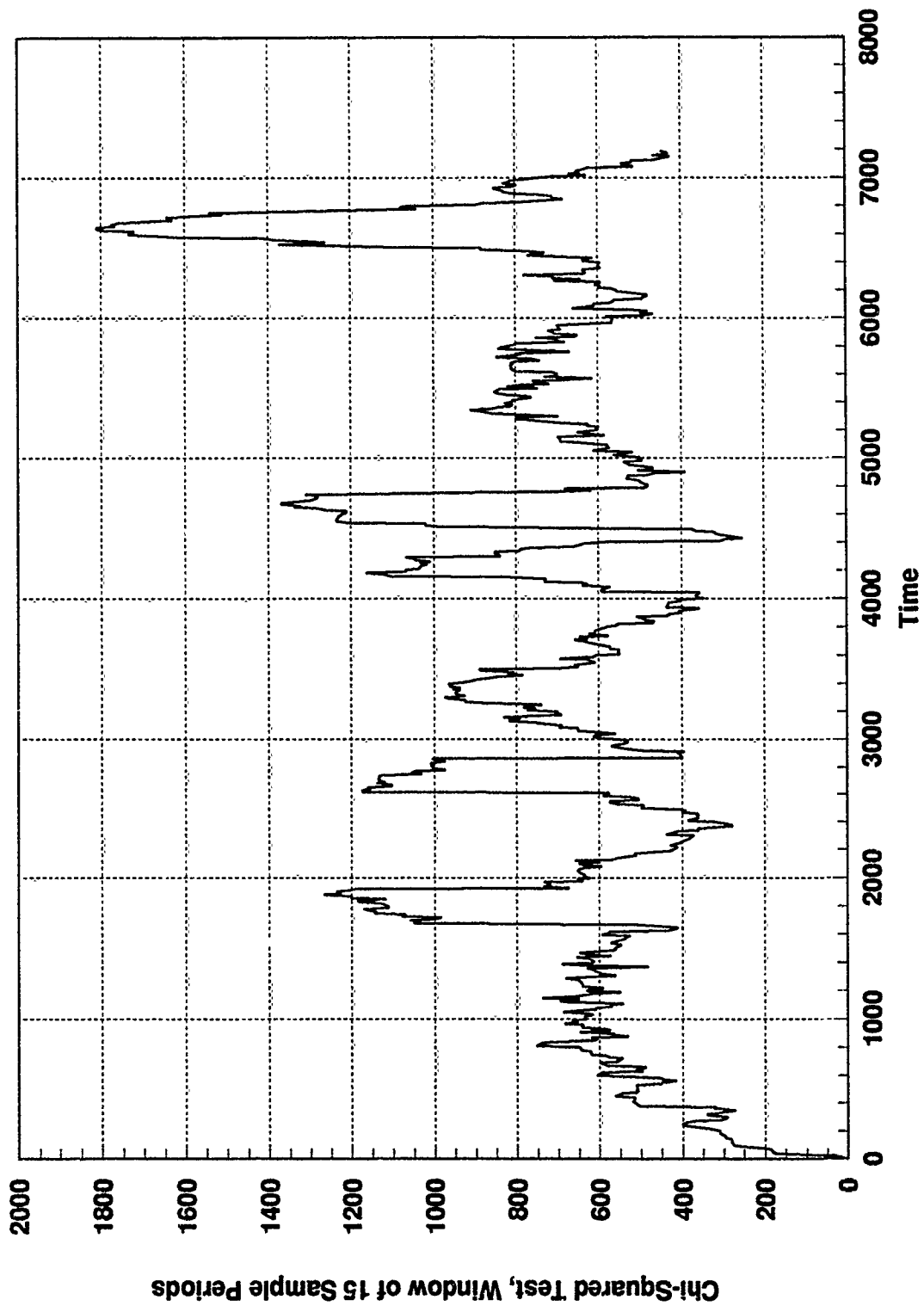
Figure I.6 Federated Filter, TAN Residual Chi-Squared Test

Appendix J: Federated Filter

Input Data Control File Example

GENERAL INFORMATION

The data file in this appendix is a sample of the input data control files, named indata.dat, used for this thesis effort. The parameter values and switches are indicative of the simulations performed for the federated filter design.

This particular input data file requires the first state and its one-sigma values output in ASCII format. Ten runs are accomplished for durations of 7200 seconds each. Local filter #1 is dedicated to the GPS sensor. Local filter #2 is dedicated to the SAR sensor array. Local filter #3 is dedicated to the TAN sensor. The no-reset mode is selected for the federated filter's operating mode.

There are a maximum of 29 states selectable per local filter and 16 states for the master filter. The INU model used is the AGBIAS model, implying a 16-state INU filter model. The GPS model used is the GPS5 model, implying a 5-state GPS filter model. Various tolerances and time segments are specified in this example to show the possible variations. For a description which elaborates on the meaning and options for each variable in this data file, please refer to the DKFSIM user's manual, reference (9).

```
'DKF SIMULATION TITLE = ', 'DKFSIM PRELIMINARY PRODUCTION RUN',

SIMULATION PARAMETERS, INPUT RANDOM NUMBER SEEDS:
$SIMCON
IRUN1   =        1,     IRUNL  =       10,
TSTART  =        0.,    TFINAL =     7200.,
FSEED   =        0,     VSEED  =        9,
LTRAJ   =        F,     FLTFIL =   'FLIGHT'
$END


OUTPUT AND POST PROCESSING CONTROL:
$OUTPPR
SCASE = 'DKF',
DHEADL(1) = 'LF/GPS = GPS5/INU16                             ',
DHEADL(2) = 'LF/SAR = SAR/INU16                              ',
DHEADL(3) = 'LF/TAN = TAN/INU16                              ',
DHEADL(4) = '                                                ',
DHEADM  =      'MF = GPS5/SAR/TAN/INU16                       ',

STYP = 'NONE',        UPSTAT =        F,

MPLOT   =        F,
OUTASC  =        T,
OUTERR  =        T,    OUTSIG =        T,
EFSAV   =        T,

DTSTAT  =     1000.,
LPLOT   =        F,    LPLMES =        F,
LPTRAJ  =        F,    LPTASK =        F,

LFSOUT(1,1) = 1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
              0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
LFSOUT(1,2) = 1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
              0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
LFSOUT(1,3) = 1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
              0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
LFSOUT(1,4) = 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
              0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,

MFSOUT(1)   = 1,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
              0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
$END   INDEX--> 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
               17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32
```

FILTER DEFINITION
```
$FILDEF
MFON = T,
NLF = 4,   LREF = 4,
LFON(1)   =        T,         T,         T,         F,
LF2MF(1) =         1,         1,         1,         0,
BETLL0(1) =      1.0,       1.0,       1.0,       1.0,
BETLC0(1) =     0.33,      0.33,      0.33,        0.,
BETML0(1) =       0.,        0.,        0.,        0.,
LNAME(1) =   'GPS5',     'SAR',     'TAN',     'REF',
BETMC0   =        0.,
DTLF = 5., DTMF = 5., DTFUS0 = 10., DTFDI = 0.,
FBKMD0 = 'NRESET',
MFMOD0 = 'MFLOCAL',
LSHAP = 'LOWER',
MSHAP = 'LOWER'

NSEN = 5,
SNAME(1)  =   'GPS',    'SARPV',   'SAREO',     'TAN',    'BARO',
TSEN0(1)  =      0.,        0.,        0.,        0.,        0.,
S2LF(1,1) =       1,         0,         0,         0,        0.,
S2LF(1,2) =       0,         1,         1,         0,         0,
S2LF(1,3) =       0,         0,         0,         1,         0,
S2LF(1,4) =       0,         0,         0,         0,         0,
S2MF(1)   =       0,         0,         0,         0,         0,
$END
```

INERTIAL NAVIGATION UNIT (INU) PARAMETERS
```
$INUPR
DTINUR =   1.,
SGGBR  =   3.889E-8,  TCGBR  =     3600.,
SGGSR  =       2.E-6,  SGGMR  =    15.E-6,  SGGADR = 8.6E-11,
SGABR  =   1.283E-3,  TCABR  =     3600.,
SGASR  =      60.E-6,  SGAMR  =    50.E-6,
NPHIR  = 3.4028E-13,  NVR    =     .0001,

SGGVR  =   1.123E-3,  SGGHR  =   .802E-3,
DCGR(1) =    5.804E4,  5.804E4,   3.696E5,

DP0(1)  =       10.,       10.,       10.,
DV0(1)  =       .05,       .05,       .05,
DTH0(1) =     10E-6,     10E-6,    100E-6,
ALNMOD = 'SIMALN',

BARDMP = F,
KBAR1 = .02, KBAR2 = 1.03E-4,
$END
DP0(1)  =      100.,      100.,      100.,
DV0(1)  =       1.0,       1.0,       1.0,
DTH0(1) =    100E-6,    100E-6,    500E-6,
ALNMOD = 'AIRALN',
```

```
$INUPC
 INUMOD = 'AGBIAS',
 SGGBC  =    3.889E-8,  TCGBC  =    3600.,
 SGGSC  =        2.E-6,  SGGMC  =   15.E-6,   SGGADC = 8.6E-11,
 SGABC  =    1.283E-3,  TCABC  =    3600.,
 SGASC  =       60.E-6,  SGAMC  =   50.E-6,
 NPHIC  =        8E-12,  NVC    =     .0004,

 SGGVC  =    1.123E-3,  SGGHC  =  .802E-3,
 DCGC(1) =    5.804E4,  5.804E4,  3.696E5,

 TCHAGC = 10.,   TCHGGC = 10.,   TCVTHC = 10.,

 DTIFBC = 10.,  KIFBC = 1.,

 SGPNOC(1) =     100.,    100.,    100.,
 SGVNOC(1) =     1.0,     1.0,     1.0,
 SGFNOC(1) =   100E-6,  100E-6,  500E-6,
 $END



GLOBAL POSITIONING SYSTEM (GPS) TRUTH PARAMETERS
 $GPSPR
 NSATR = 5,
 TREFG =  2200.,  LONG = -84.3,  LATG =  47.2,
 RGPS = 84E6,  IGPSD    =   55.,    GELMIN = -90.,
 AZMOD(1) = -80.,      80.,    180.,    180.,    180.,
 ELVOD(1) =  20.,      20.,     20.,     70.,     45.,

 NGPSSG    =       5,
 DTGPS(1) =    5.0,    5.0,    5.0,    5.0,    5.0,
 TGPSSF(1) = 1600.,  2300.,  2400.,  3600.,  9999.,

 SGCPOT = 10000.,
 SGCFT  =   0.05,  TGCFT = 1800.,
 SGCFAT =  .0015,
 SGIRT  =    10.,   TGIRT  = 3600.,   ZGIRT = 0.7,
 SGSRT  =    10.,   TGSRT  = 3600.,   ZGSRT = 0.3,
 SGTRT  =     5.,   TGTRT  = 3600.,   ZGTRT = 0.7,
 SGRPNT =    20.,   SGRFNT = 0.075,
 USESAT(1) =      T,     T,      T,      T,      F,
 USEPR  =        T, USEPRR =      T,
 OUTSAT =        F,
 $END
```

GPS FILTER PARAMETERS:
```
$GPSPC
GPSMOD = 'GPS5',
SGCPOC = 10000.,   SGCFOC =    0.05,
SGCFC  =    0.05,  TGCFC  = 1800.,
SGCFAC = 0.0015,
SGRPNC =    20.,   SGRFNC =    0.075,
SGTRC  =     5.,   TGTRC  = 3600.,
SGIRC  =    10.,
SGSRC  =    10.,
NGCPC  =    1.0,   NGSRC  =    1.0,
TOLGPR =     5.,   TOLGRR =    5.,
$END
```

SAR-EO SYSTEM PARAMETERS, REAL-WORLD
```
$SEOPR
NSEOSG    =        2,
DTSEO(1)  =     300.,    300.,
TSEOSF(1) =    3000.,   9999.,

SSRSFR = .0004,
SSLMR  =    50.,    DSLMR  = 50000.,
SSRBR  =   100.,    TSRBR  =   600.,
SSRRBR =     .1,    TSRRBR =   600.,
SSABR  =   .001,    TSABR  =   600.,
SSEBR  =   .001,    TSEBR  =   600.,
SSRNR  =    50.,    SSRRNR =    0.1,   SSANR  = .0005,   SSENR  = .0005,
UREOR  =     T,     URREOR =     T,    UAZEOR =      T,  UELEOR =       T,
SRNGMX = 500000.,   SAZMMX =     4.,   SELVMX =     4.,  WBMAX  =   100.,

NSEOTG = 4,
DR(1)  =  125000.,  125000.,  125000.,  125000.,
CT(1)  =   25000.,  -25000.,   25000.,  -25000.,
ALTLM(1)=   5000.,    5000.,    6000.,    8000.,
DTTARG(1) =  100.,     100.,     100.,     100.,
$END
```

SAR-EO SYSTEM PARAMETERS, COMPUTER
```
$SEOPC
SSRSFC = .0004,
SSLMC  =    50.,
SSRBC  =   100.,    TSRBC  =   600.,
SSRRBC =     .1,    TSRRBC =   600.,
SSABC  =   .001,    TSABC  =   600.,
SSEBC  =   .001,    TSEBC  =   600.,
SSRNC  =    50.,    SSRRNC =    0.1,   SSANC = .0005,   SSENC = .0005,
TOSRN  =     5.,    TOSRRN =     5.,   TOSAN =    5.,   TOSEN =    5.,
$END
```

SAR-PVU SYSTEM PARAMETERS, REAL-WORLD
```
$SPVPR
NSPVSG      =        2,
DTSPV(1)    =    300.,    300.,
TSPVSF(1)   =   3000.,   9999.,

SSVNR    =     .5,
SSVMR    =   .001,
SSVSR    =   .001,   TSVSR =  1000.,
SSVBR    =     1.,   TSVBR =  1000.,
UPV(1)   =      T,       T,       T,
SALTMX   =  1.E6, VMIN =  100.,
$END
```

SAR-PVU SYSTEM PARAMETERS, COMPUTER
```
$SPVPC
SSVNC    =     .5,
SSVMC    =   .001,   TSVMC =  1000.,
SSVSC    =   .001,   TSVSC =  1000.,
SSVBC    =     1.,   TSVBC =  1000.,
TOSVN    =    5.,
$END
```

BARO-ALTIMETER INPUT DATA 4-10-89 HRM
```
$BARPR
NBARSG      =        1,
DTBAR(1)    =     10.0,
TBARSF(1)   =    9999.,

SBBR     =    200.,
DBAR     =   6.8E6,
SBSR     =     .03,
SBTDR    =     .25,
SBSPR    =  1.5E-4,
SBNR     =     10.,
UBARO    =       T,
$END
SBBR     =    500.,
DBAR     =  1.25E6,

$BARPC
SBBC     =    500.,
SBSC     =     .03,
SBTDC    =     .25,
SBSPC    =  1.5E-4,
SBNC     =     10.,
DBAC     =  1.25E6,
TOBAN    =      5.,
$END
```

```
$TANPR
NTANSG     =        1,
DTTAN(1)   ..       20,
TTANSF(1)  =     9999.,

NMAPI  =  3,
SRABR  =       20.,   TRABR  =  600.,    SRANR  =  10.,
SRASFR  =  .001,
TERTYP  =     -1,
STMHR  =    150.,    DTMHR  =   5000.,
STMFER  =    45.,    DTMFER  =  5000.,    STMVER  =  15.,
DLATR  =   10E-6,
DLONR  =   14E-6,
UTANO  =        T,

ELBOR  =  -2000.,
NTERSG     =        3,
STSAR(1)   =      .05,        .10,        .05,
DTEAR(1)   =   1800.,      1800.,      1800.,
ZTEAR(1)   =      2.,        2.,        2.,
STSCR(1)   =      .05,       .10,       .05,
DTECR(1)   =   1800.,      1800.,      1800.,
TTERSF(1)  =   2600.,      3300.,      9999.,
$END


$TANPC
SRABC  =       20.,   TRABC  =  600.,    SRANC  =  50.,
SRASFC  =  .001,
STSC  =        0.1,
STMHC  =    150.,
STMEC  =    45.,    DTMEC  =   5000.,
TLTANA  =  5.,
$END
```

Vita

Captain Paul J. Lawrence, Jr. was born 30 June, 1959 in Jacksonville, North Carolina. He graduated in 1977 from Lakewood High School in Roseboro, North Carolina. He joined the United States Air Force directly after high school and served as a Defensive Aerial Gunner in B-52D aircraft for the 9th Bombardment Squadron, 7th Bombardment Wing, Carswell Air Force Base, Texas. He was selected for the Airman Education and Commission Program in 1983 and began study at the University of Texas at Arlington in May, 1984. He graduated in Dec, 1986 with a Bachelor of Science in Electrical Engineering. Subsequently, he graduated Officer Training School in Apr, 1987 and then went on to become an engineer for the Air Force Logistics Command, F-111 Directorate. He was selected to hold the position of project and production engineer for the avionics modification of the F-111 aircraft. As project engineer, he was responsible for integration and modification of the hardware and software aspects of the avionics upgrade effort. As production engineer, he was responsible for ensuring the smooth operation of the production lines by solving aircraft tail number peculiar problems plus a host of other manufacturing problems. He was selected to attend the Air Force Institute of Technology in 1991 to obtain a Master of Science degree in Electrical Engineering with specialization in Guidance, Navigation, and Control.

Permanent Address: 11144 South Vernon Avenue

Chicago, Illinois   60628

REPORT DOCUMENTATION PAGE

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	30 July, 1993	Master's Thesis

4. TITLE AND SUBTITLE

Comparison of a Distributed Kalman Filter Versus a Centralized Kalman Filter With Fault Detection Considerations

5. FUNDING NUMBERS

6. AUTHOR(S)

Paul J. Lawrence, Jr.
Captain, USAF

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Air Force Institute of Technology
Wright-Patterson Air Force Base, OH 45433

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GE/ENG/93S-06

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Michael P. Berarducci
Air Force Wright Laboratory, Avionics Directorate
WL/AAAI-3
Wright-Patterson Air Force Base, OH 45433

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Distribution Unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

This project examined the results obtained by simulating an aircraft navigation system with a partial complement of a typical avionics sensor array. Two different techniques of estimation processes were utilized and compared: the conventional Kalman and the federated filter architectures. Areas of interest include error state estimation accuracy and overall performance, residual behavior under normal and induced sensor failure conditions, and potential for failure detection and isolation. Several simulations were accomplished for each filter design and the results were compared in order to verify the validity of the newly developed federated filter architecture.

Comparison of the error state estimation accuracies of the two filter designs revealed excellent overall performances for both. The identification of failures showed a definite advantage in the federated filter design. Having sensor-dedicated local filters allowed for easy sensor failure identification for the federated filter, while the centralized filter design suffered from navigation solution corruption under the same circumstances.

Once established as a valuable estimation technique, the federated filter will add significantly to the viable alternatives when choosing a particular filter architecture for current avionics modifications or future avionics implementations.

14. SUBJECT TERMS

Kalman Filter, Distributed Kalman Filter, Federated Filter, Residual Monitoring, Fault Detection and Isolation

15. NUMBER OF PAGES
280

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL