(1)

MOVING-BANK MULTIPLE MODEL ADAPTIVE ESTIMATION AND CONTROL APPLIED TO A LARGE FLEXIBLE SPACE STRUCTURE

THESIS

Robert Brent Moyle
Captain, USAF

AFIT/GE/ENG/90D-45

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

Wright-Patterson Air Force Base, Ohio

AFIT/GE/ENG/90D-45

MOVING-BANK MULTIPLE MODEL ADAPTIVE ESTIMATION AND CONTROL APPLIED TO A LARGE FLEXIBLE SPACE STRUCTURE

THESIS

Robert Brent Moyle
Captain, USAF

AFIT/GE/ENG/90D-45

AFIT/GE/ENG/90D-45

MOVING-BANK MULTIPLE MODEL ADAPTIVE ESTIMATION AND CONTROL APPLIED TO A LARGE FLEXIBLE SPACE STRUCTURE

THESIS

Presented to the Faculty of the School of Engineering

of the Air Force Institute of Technology

Air University

In Partial Fulfillment of the

Requirements for the Degree of

Master of Science in Electrical Engineering

Robert Brent Moyle, B.S.E.E.

Captain, USAF

December 1990

Preface

The purpose of this thesis was to determine the optimal performance of the moving-bank multiple model adaptive estimation and control algorithms with a reduced order filter against a higher order truth model. The moving-bank mulitple model adaptive estimation and control algorithm is an attempt to reduce the computational loading that would be required for full-scale implementation of the multiple-model adaptive estimator or controller. The results of this thesis showed that the performance of the reduced-order model moving-bank algorithms could be substantially improved over previous research results with the appropriate determination of filter model noise statistics and LQG controller weighting parameters. The determination of the optimal moving-bank logic and controller logic also demonstrated performance comparable to that of a non-adaptive benchmark. This thesis also demonstrated the need for adaptive estimation/control in the case of parameter/filter location mismatch and also for the case of varying parameters for the two-bay truss structure.

I would like to extend my sincere gratitude to Dr. Peter Maybeck for his time, guidance, and encouragement, without which I would not have been able to complete this research. I would like to share the accomplishment of this research with my wife, Laureli. Her love and support, while not apparent in the text of this document, was greatly needed and was sincerely appreciated throughout the duration of this endeavor.

Robert Brent Moyle

Table of Contents

List of Figures

List of Tables

Abstract

The performance of moving-bank multiple model adaptive estimation (MMAE) and control (MMAC) algorithms for large space structure control is analyzed in this thesis. The performance of a six-state filter model and associated controller are evaluated on the basis of estimation/control performance against a 24-state truth model.

A model developed using finite element analysis is used to approximate a large flexible space structure. The space structure is configured as a two-bay truss which is attached to a large central hub, where the mass of the hub is considered to be much larger than the mass of the flexible structure. The model is developed in physical coordinates and then transformed into modal coordinates, where the method of singular perturbations is used to obtain a reduced order filter model. The actual positions and velocities of various physical points on the structure are used in the evaluation of the moving-bank algorithm performance.

Results of the research indicate that appropriate determination of the filter model noise statistics as well as the LQG controller weighting matrices significantly improve performance of the bank throughout the parameter space. The results indicate that the performance of the moving-bank algorithms is seriously degraded by the inclusion of the filter-computed residual covariance in the conditional probability density function for computation of the hypothesis conditional probabilities within the multiple model algorithms. The performance of the moving-bank MMAE/MMAC algorithms using parameter position estimate monitoring for parameter identification, and using a modified MMAC methodology for implementing control, provide performance comparable to an artificially informed non-adaptive controller benchmark. The moving-bank algorithms provide performance comparable to a benchmark controller for the cases of slowly varying and jump parameters, as well as for constant parameter values.

MOVING-BANK MULTIPLE MODEL ADAPTIVE ESTIMATION AND CONTROL APPLIED TO A LARGE FLEXIBLE SPACE STRUCTURE

I. Introduction

In many estimation and control applications, problems arise when the parameters which describe the system model are not known with absolute certainty. The uncertainty in these parameters reduces the degree to which the system model is valid and degrades the accuracy of the algorithms based on that model. The values of these parameters may not remain constant. In fact, quite often these parameters will vary slowly over time (such as due to fuel depletion), or change abruptly (such as due to a structural failure). One method to provide real-time identification of the parameter values required in these situations entails the construction of a bank of Kalman filters; one filter is designed for each possible parameter value. These filters are run in parallel, and their outputs are combined through an appropriate weighting logic. This technique is referred to as Multiple Model Adaptive Estimation (MMAE). This method not only provides estimates of the uncertain parameters, but more importantly, it provides estimates of the system states as well. The system state estimates provided from the MMAE approach are formulated as the probabilistically weighted sum of the elemental filter state estimates.

The basic problem with the full-bank MMAE technique described above is that the number of Kalman filters required to be processed in real time creates an unbearable computer load. This thesis uses the concept of a "moving-bank" MMAE to alleviate the computer burden. This method maintains a subset of the full-bank elemental Kalman filters in a small window in parameter space and dynamically redeclares the filters contained in this window such that the current parameter estimate is continuously surrounded. The states estimated by the moving bank are then provided as inputs to a controller of some form. Several controller designs are appropriate for implementation with the moving-bank

estimator. The control methods examined in this thesis will utilize the "assumed certainty equivalence design" technique, which consists of developing an estimator cascaded with a deterministic full-state feedback optimal controller [17:17]. The controllers so designed will be based on a Linear system with a Quadratic cost control criterion driven by Gaussian noises, or LQG, control synthesis.

In addition to maintaining fewer of the parallel Kalman filters on line, computational loading can be further mitigated by reducing the number of system states upon which the elemental Kalman filters and/or controllers within the moving-bank algorithm are based. However, when a reduced order model is used for the basis of synthesizing a filter and/or controller, the robustness of the adaptation process to the unmodelled effects is a prime consideration.

This thesis will use the moving-bank estimator and LQG control synthesis described above to control a large, flexible, space structure. A Monte Carlo study will be performed to evaluate the performance of a moving-bank algorithm based on steady-state, constant-gain Kalman filters and controllers. In this thesis effort, the control objective is to quell any oscillations in the structure and to "point" the structure in a commanded direction. Structural vibrations can be the result of external disturbances or from commanded maneuvers like slewing.

1.1 Background

The development discussed in this section will provide a brief overview of the following four areas:(1) multiple model adaptive estimation - MMAE, (2) moving-bank MMAE, (3) moving-bank MMAE-based controller, and (4) the system model. The discussions contained in the following sections are not intended to be complete. Rather, the information provided here is intended to allow understanding of the problem statement and approach at the end of this chapter. Chapter 2 discusses the development of the first three algorithms in more detail, whereas Chapter 3 discusses system model development in more detail. The notation used in this thesis is intended to be consistent with the convention found in [15] such that a stochastic process is denoted by $\underline{\mathbf{x}}$ while a deterministic process is denoted by $\mathbf{x}$.

1.1.1 Multiple Model Adaptive Estimation - MMAE. In most control applications, the normal approach for optimal control system stability and robustness is the use of full-state feedback. However, this requires perfect access to all states, which is rarely attainable due to the fact that the states may not be directly measurable. In addition, the states that are measurable may be influenced by uncertain parameters. In either case, a method must be implemented to provide the best possible state estimates to the controller.

The approach that MMAE uses to provide state estimates for system control is to design a Kalman filter for each possible parameter value $\mathbf{a}_k$, for $k = 1, \ldots, K$. The MMAE approach results in a bank of K Kalman filters, where K is the number of possible parameter values. To enhance the feasibility of the MMAE technique, it is assumed that the uncertain parameters can realize only discrete values; either this is physically realistic or discrete values are chosen from the continuous parameter variation range. (Proper discretization of the parameter space is then an important issue, as will be discussed later.) This is necessary in order to reduce the resulting number of possible parameter combinations. For example, if the system of interest had two uncertain parameters that could each realize two discrete values, the resulting parameter space would be comprised of four discrete points. On a larger scale, if the parameters could realize 100 values each, the parameter space would be composed of 10,000 discrete points. Conceptually, a system model would be associated for each of the 10,000 points in the parameter, each requiring a Kalman filter. This would create an unbearable computational load for any real-time control system. Following the development of previous research, this investigation will use two parameters, where each parameter is allowed to realize ten discrete values. This results in a parameter space consisting of 100 points with which to approximate the actual continuous parameter space, and the MMAE based on all 100 points will be referred to as the "full-bank" estimator.

Each Kalman filter in the bank helps to produce a hypothesis conditional probability that its assumed parameter is "correct". These conditional probabilities are based on the characteristics of the residuals of the Kalman filters and are used as a weighting factor for the state estimate produced by each Kalman filter. The Kalman filter based upon the "true" parameter should have residuals consistently smaller (relative to the filter's

own computed residual covariance) than the residuals of the other mismatched filters. Accordingly, the conditional probability associated with the "correct" filter model will increase, causing the others to decrease [16:133]. The probability for each Kalman filter is a function of the conditional probability densities related by

$$p_k(t_i) = \frac{f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_k, \mathbf{Z}_{i-1}) p_k(t_{i-1})}{\sum_{j=1}^{K} f_{\mathbf{z}(t_i)|\mathbf{a},\mathbf{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_j, \mathbf{Z}_{i-1}) p_j(t_{i-1})} \tag{1.1}$$

The first numerator term represents the probability density of the current measurement based on the assumed $\mathbf{a}_k$ parameter and the previous measurement history through time t_{i-1} [16:131]. The second numerator term is the previous p_k value. Equation (1.1), therefore, represents an iterative relation. The probability density function is formed by:

$$\begin{aligned} f_{\underline{\mathbf{z}}(t_i)|\underline{\mathbf{a}},\underline{\mathbf{Z}}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_k, \mathbf{Z}_{i-1}) &= \frac{1}{(2\pi)^{\frac{m}{2}} \mid \mathbf{A}_k(t_i) \mid^{\frac{1}{2}}} \exp\{\cdot\} \\ \{\cdot\} &= \{-\frac{1}{2}\mathbf{r}_k^T(t_i)\mathbf{A}_k^{-1}(t_i)\mathbf{r}_k(t_i)\} \end{aligned} \tag{1.2}$$

where $\mathbf{r}_k(t_i)$ is the filter residual $\left[\mathbf{z}(t_i) - \mathbf{H}_k(t_i)\hat{\underline{\mathbf{x}}}_k(t_i^-)\right]$ in the k^{th} filter, expressed in terms of the measurement matrix $\mathbf{H}_k(t_i)$ and the state estimate before the i^{th} estimate is incorporated, $\hat{\underline{\mathbf{x}}}_k(t_i^-)$. The quantity $\mathbf{A}_k(t_i)$ is a function of $\mathbf{H}_k(t_i)$, the state estimation error covariance matrix before measurement update $\mathbf{P}_k(t_i^-)$, and the noise covariance matrix $\mathbf{R}_k(t_i)$, namely $\mathbf{A}_k(t_i) = \left[\mathbf{H}_k(t_i)\mathbf{P}_k(t_i^-)\mathbf{H}_k^T(t_i) + \mathbf{R}_k(t_i)\right]$. It will be shown in Chapter 2 that the complete evaluation of Equation (1.2) is available from each separate Kalman filter.

The denominator of Equation (1.1) is the sum of all numerator terms for $j = 1, \ldots, K$, so that the sum of all p_k is unity. This property does not let any one filter control the estimation process unless one of the estimator's probability goes to one and all the others go to zero. Steps must be taken to prevent any probabilities from going to zero, since any such probability would become permanently locked onto zero by the iteration performed by Equation (1.1) even if changing conditions were to cause that particular parameter value to become the best value. One way to prevent the probabilities from going to zero is to set a lower limit that the p_k may assume. Other methods are discussed in Chapter 2. The

adaptive state estimate of the MMAE is then determined by summing the probabilistically weighted outputs of each Kalman filter. This technique is referred to as the Bayesian form of the MMAE estimator and is illustrated in Figure 1.1. A second approach for determining the overall state estimate would be to take the state estimate of the filter with the highest conditional probability. This method is referred to as the maximum a posteriori, or MAP, estimator.

1.1.2 Moving-Bank MMAE. The concept behind the moving-bank MMAE is that the state estimate provided by the full-bank MMAE may be approximated by a reduced number of Kalman filters, say J where $J < K$. The moving-bank filters are dynamically redeclared such that the center filter is located as close as possible to the current parameter estimate. The parameter estimate of the moving-bank MMAE algorithm should be nearly as good as the full bank estimate, provided that the vast majority of the full-bank MMAE algorithm parameter probability weight is contained within the moving bank. Figure 1.2 shows how the full-bank MMAE system would look if all 100 filters were used. Each of the blocks represent a discrete parameter point used as the basis for a Kalman filter. Figure 1.3 shows how a moving 3-by-3 bank might appear surrounding the current parameter estimate. The moving bank is composed of nine solid blocks. The "discarded filter" points correspond to a 3-by-3 bank at an earlier time instant when that set of nine points most closely surrounded the estimated parameter location at that time.

There are five basic techniques for deciding when to move the bank, when to contract the bank to a finer discretization, and when to expand the bank to encompass a coarser discretization. "Fine" discretization implies that the filters in the bank are adjacent to one another, as illustrated in Figure 1.3(a). "Coarse " discretization implies that the filters in the bank are not adjacent to one another, which is illustrated in Figure 1.3(b). The five techniques used for decision making are [19]:

- Residual monitoring
- Parameter position estimate monitoring
- Parameter position and "velocity" estimate monitoring

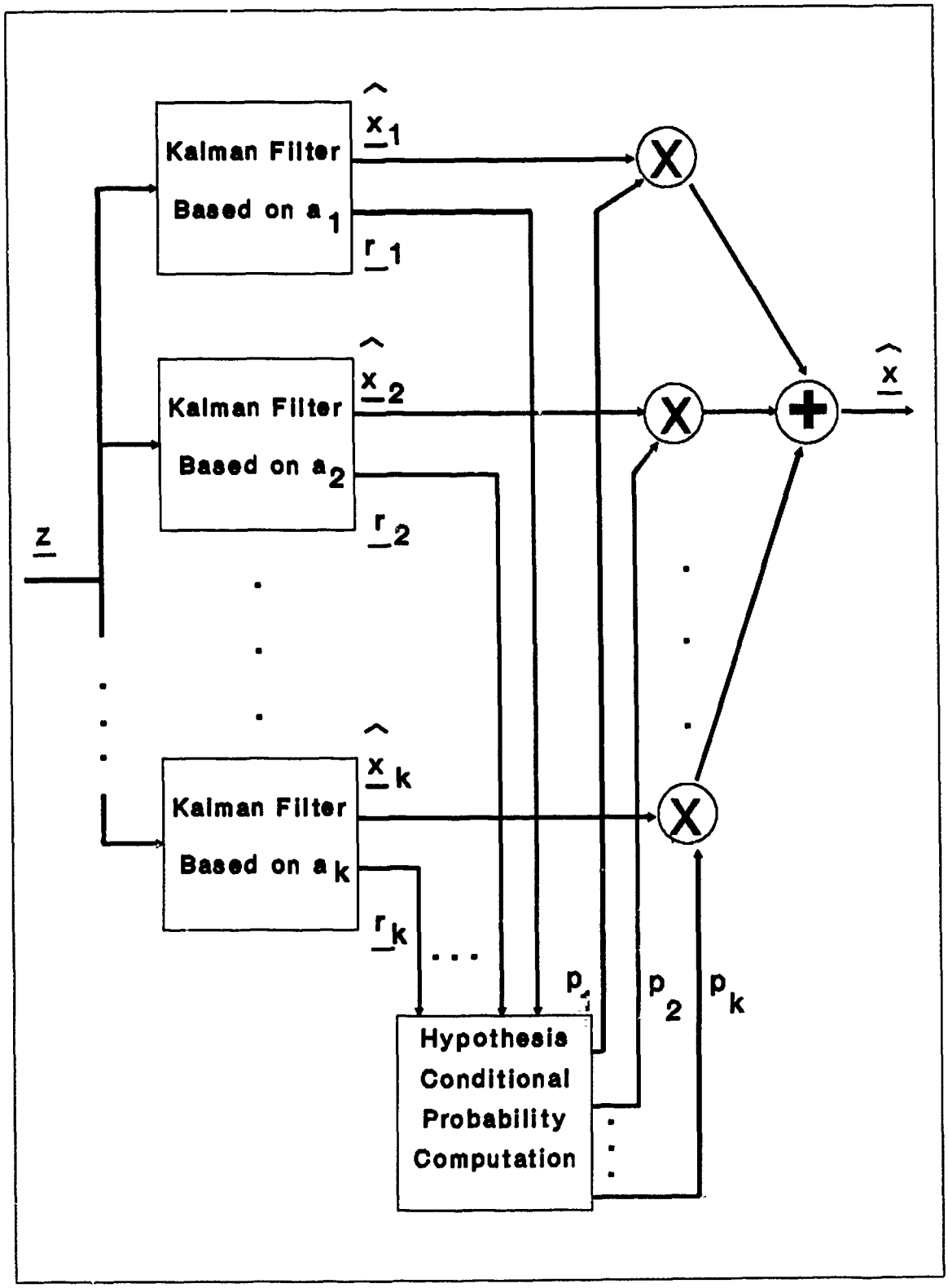

Figure 1.1. Diagram of Multiple Model Adaptive Estimator [16:132]

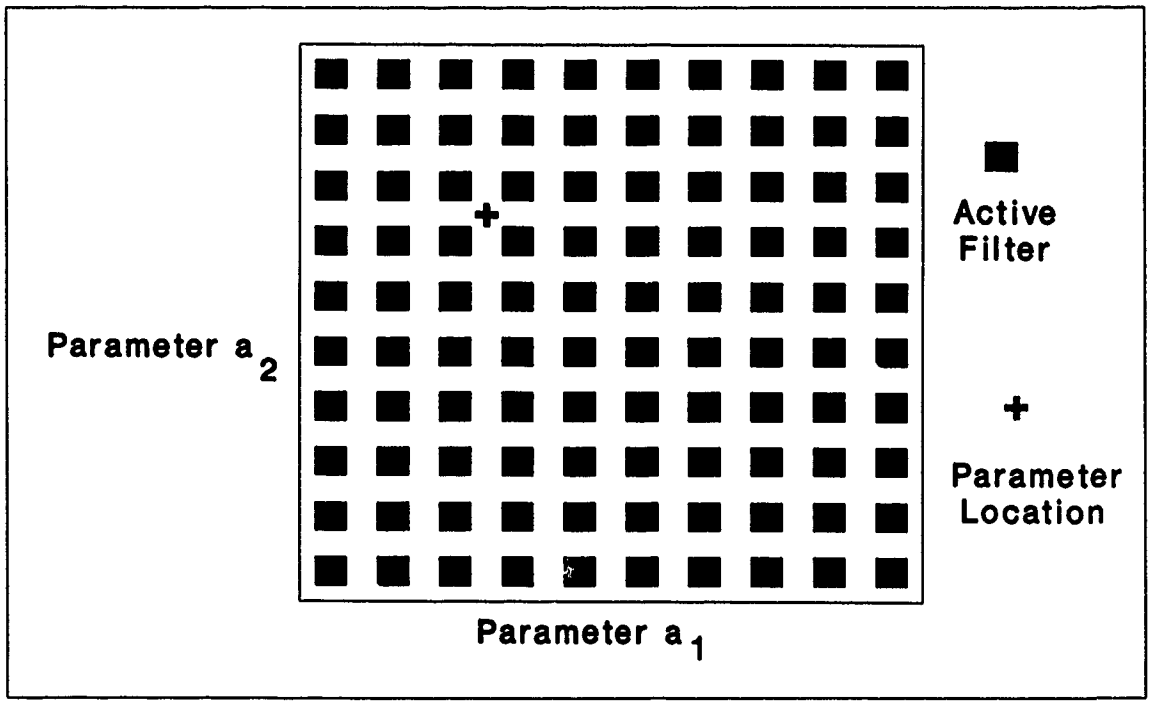

Figure 1.2. Diagram of Full-Bank MMAE

- Probability monitoring
- Parameter estimation error covariance monitoring

Residual monitoring is used for movement of the bank as well as for expansion. Residual monitoring uses a likelihood quotient based on the elemental filters in the bank and is defined as:

$$L_j(t_i) = \mathbf{r}_j^T(t_i)\mathbf{A}_j(t_i)^{-1}\mathbf{r}_j(t_i) \tag{1.3}$$

which is the quadratic form within Equation (1.2). If all the L_j are above a preset move threshold, the bank is moved. This condition indicates that all residuals are larger than anticipated by the associated computed $\mathbf{A}_j(t_i)$ value, and thus that none of the current filters is based upon a particularly good assumed parameter value. In addition, the filter with the smallest likelihood quotient should be the filter nearest the true parameter value, thus providing the direction for movement. If the parameter undergoes a jump change or is changing rapidly, a closely spaced bank may not be able to track the parameter adequately. By comparing the L_j to a preset expansion threshold, the bank can be expanded.

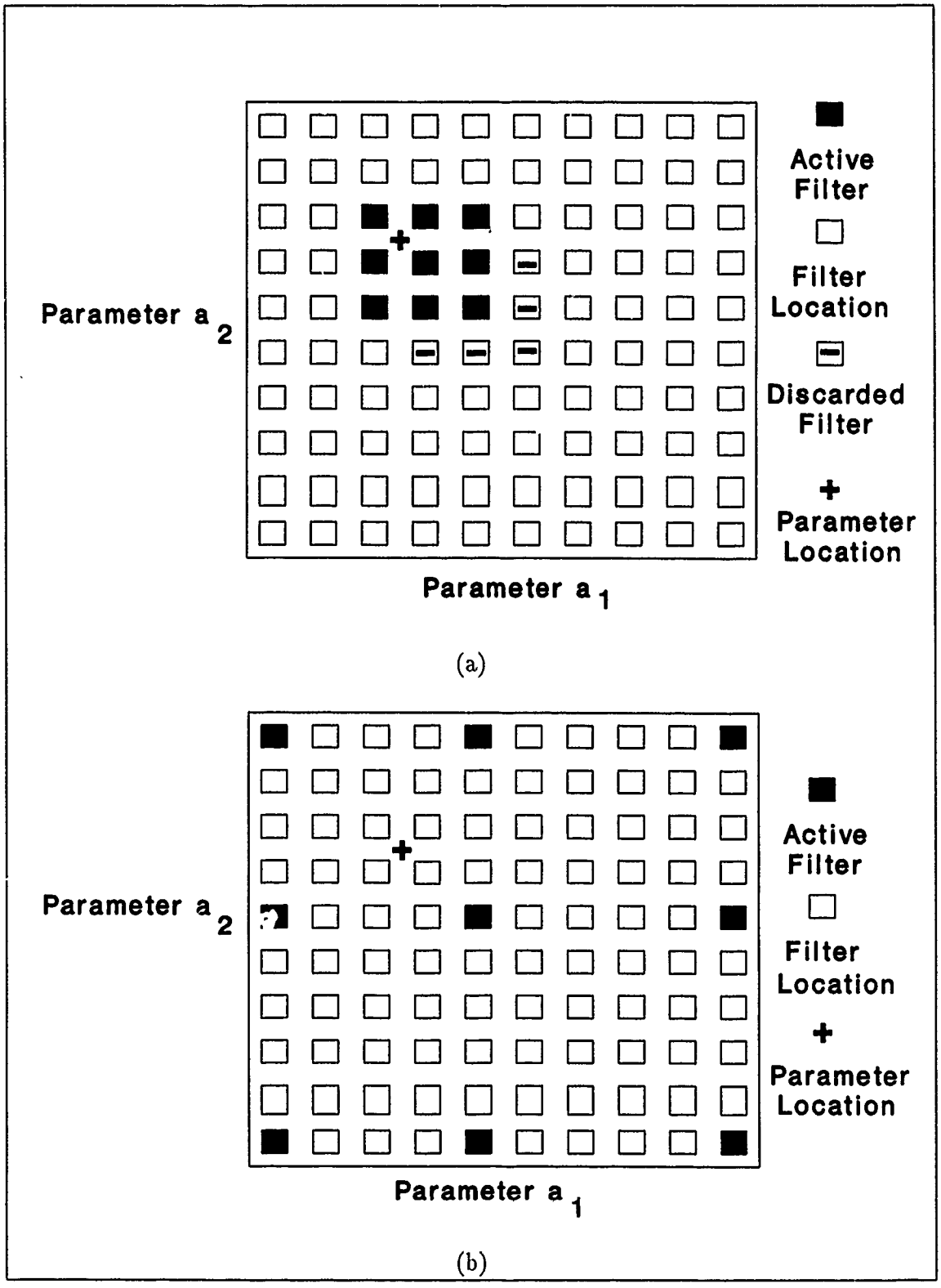

Figure 1.3. Diagram of Moving-Bank MMAE (a) Fine Discretization, and (b) Coarse Discretization

By expanding the filter bank, the parameter value can be reacquired and a decision to contract the bank around the new parameter location can be made. Residual monitoring is susceptible to single large samples of measurement noise and may give false alarms [19:1876], i.e., the bank may be moved or expanded unneccessarily.

Parameter position estimate monitoring is used for movement of the bank and attempts to keep the bank centered on the current estimate of the true parameter. The estimated true parameter is given by:

$$\hat{\mathbf{a}}(t_i) = E\{\mathbf{a}(t_i) \mid \underline{\mathbf{Z}}(t_i)\} = \sum_{j=1}^{J} \mathbf{a}_j \cdot p_j(t_i) \tag{1.4}$$

When the "distance" between the center of the bank and the estimated parameter location becomes larger than a preset move threshold, the filter bank is moved. Since the calculation depends on the time history of measurements, rather than just a single one, this technique is less susceptible to false alarms than the residual monitoring method [19:29].

Parameter position and "velocity" estimate monitoring is used for movement of the bank and is an extension of the previous method. By tracking the "velocity" of a slowly moving parameter through the most recent parameter position estimates, the next position of the parameter may be predicted. If the distance between the predicted location and the current center of the filter bank exceeds a preset move threshold, the bank is moved.

Probability monitoring is used for movement of the bank and uses the computed probabilities provided in Equation (1.1). The computed probabilities are compared to a preset move threshold to determine if the bank should be moved in the direction indicated by the filter producing the highest probability exceeding the threshold.

Parameter estimation error covariance monitoring provides a means for determining whether the bank should be contracted from a coarse to a finer discretization. The parameter estimation error covariance is given by:

$$\begin{aligned} \mathbf{P}_{\hat{\mathbf{a}}}(t_i) &= E\{[\mathbf{a} - \hat{\mathbf{a}}(t_i)][\mathbf{a} - \hat{\mathbf{a}}(t_i)]^T \mid \mathbf{Z}(t_i) = \mathbf{Z}_i\} \\ &= \sum_{j=1}^{J} [\mathbf{a} - \hat{\mathbf{a}}(t_i)][\mathbf{a} - \hat{\mathbf{a}}(t_i)]^T \cdot p_j(t_i) \end{aligned} \tag{1.5}$$

When the norm (or some other suitable measure) of the matrix $\mathbf{P}_{\hat{\mathbf{a}}}$ falls below some preset threshold, the bank may then be contracted around the current parameter estimate.

1.1.3 Moving-Bank MMAE Controller. There are several controller designs appropriate for implementation with the full-bank or moving-bank estimator discussed in Sections 1.1.1 and 1.1.2. All designs considered here assume that the system model is Linear, the control cost criterion is Quadratic, and the system and measurement noises are Gaussian. Using LQG optimal control synthesis for an adaptive control problem is based upon the use of the "assumed certain equivalence property " [17:241]. This property allows the independent development of an estimator cascaded with a deterministic full-state feedback optimal controller. In addition, this property results in the LQG controller being equivalent (in the case in which there is no parameter uncertainty) to the optimal deterministic controller but with the states replaced with the conditional estimates provided by the estimator. This research will investigate a special form of controller known as a "regulator" since the objectives are to drive the position and velocity states to zero.

Each controller developed in this research is a linear, quadratic cost, full-state feedback optimal deterministic controller based upon a specific assumed parameter value of a. The output of the controller is desired to be the optimal control function, u*, such that the quadratic cost function

$$J = E\left\{\sum_{i=0}^{N}\frac{1}{2}\left[\underline{x}^T(t_i)\mathbf{X}(t_i)\underline{x}(t_i)+\mathbf{u}^T(t_i)\mathbf{U}(t_i)\mathbf{u}(t_i)+2\underline{x}^T(t_i)\mathbf{S}(t_i)\mathbf{u}(t_i)\right]+\frac{1}{2}\underline{x}^T(t_{N+1})\mathbf{X}_f\underline{x}(t_{N+1})\right\} \tag{1.6}$$

is minimized [17:73]. This can also be written as

$$J = E\left\{\sum_{i=0}^{N}\frac{1}{2}\left(\begin{bmatrix}\underline{x}(t_i)\\ \mathbf{u}(t_i)\end{bmatrix}^{\mathrm{T}}\begin{bmatrix}\mathbf{X}(t_i) & \mathbf{S}(t_i)\\ \mathbf{S}^{\mathrm{T}}(t_i) & \mathbf{U}(t_i)\end{bmatrix}\begin{bmatrix}\underline{x}(t_i)\\ \mathbf{u}(t_i)\end{bmatrix}\right)+\frac{1}{2}\underline{x}^{\mathrm{T}}(t_{N+1})\mathbf{X}_f\underline{x}(t_{N+1})\right\} \tag{1.7}$$

The cross term, $\mathbf{S}$, between $\underline{x}(t_i)$ and $\mathbf{u}(t_i)$ within Equation (1.7) results from the desire to apply control at discrete time intervals to minimize an appropriate continuous-time quadratic cost. Chapter 2 will provide a more complete discussion of the development.

The matrices of interest in Equation (1.7) are as follows:

- J = cost to be minimized
- $\underline{x}(t_i)$ = n-dimensional state vector
- $\mathbf{X}(t_i)$ = $n - by -$ n-dimensional state weighting matrix
- $\mathbf{X}_f$ = $n - by -$ n-dimensional final state weighting matrix
- $\mathbf{u}(t_i)$ = r-dimensional deterministic input vector
- $\mathbf{U}(t_i)$ = $r - by -$ r-dimensional control weighting matrix
- $\mathbf{S}(t_i)$ = $n - by -$ r-dimensional cross-weighting matrix
- t_{N+1} = final time
- t_N = last time a control is applied and held constant over the next sample period

The final term in Equation (1.7) assigns a quadratic cost penalty to the magnitude of the terminal state deviation from zero. If the cost weighting matrix associated with the final state, $\mathbf{X}_f$, is diagonal, then these diagonal terms are selected to reflect the relative importance of maintaining each component of $\underline{x}(t_{N+1})$ near zero: the more important the state minimization, the larger the associated $\mathbf{X}_f$ term. The same comments can be made for the $\mathbf{X}(t_i)$ weighting matrix, which reflects the importance of maintaining individual state component deviations at small values over each of the $(N+1)$ sample periods. The diagonal entries in the control weighting matrix, $\mathbf{U}(t_i)$, determine the measure of individual control conservation desired over each of the $(N+1)$ sample periods. The larger the matrix entry, the higher will be the penalty for expending more energy. The $\mathbf{X}(t_i)$ and $\mathbf{X}_f$ matrices are assumed to be real symmetric, positive semidefinite. This allows zero cost to be assigned to certain state variables of no significance. The $\mathbf{U}(t_i)$ matrix is assumed to be real symmetric and positive definite. This assumption precludes a controller solution which would require an infinite amount of energy at any time. The cross-weighting matrix, $\mathbf{S}(t_i)$, is chosen so that the resulting symmetric composite matrix in the summation of Equation (1.7) is positive semidefinite. This research will investigate constant weighting matrices, as will be discussed in Chapter 2.

If the assumption of a linear system driven by white Gaussian noise is used, with a quadratic cost function given by Equation (1.7), then the optimal discrete linear feedback

control law, is given by [17:16]:

$$\mathbf{u}^*(t_i) = -\mathbf{G}_c^*(t_i)\hat{\underline{\mathbf{x}}}(t_i^+) \tag{1.8}$$

where full state access has been replaced via assumed certainty equivalence by the state estimate provided by the moving-bank MMAE. The controller gain, $\mathbf{G}_c^*(t_i)$, is found by solving an $n - by - n$ backward Riccati difference equation from an appropriate terminal condition. Chapter 2 will address this development in more detail.

A particularly useful implementation of the control law given in Equation (1.8) can be used for time-invariant systems with stationary noise. If the performance degradation due to ignoring the terminal transient of the $\mathbf{G}_c^*(t_i)$ and the initial transient of the Kalman filter gain is acceptable, one might seek the steady-state constant-gain control law to use during a finite time interval that is long compared to the transient periods. For this case, the optimal discrete linear feedback control law is given by: [17:243]

$$\mathbf{u}^*(t_i) = -\overline{\mathbf{G}}_c^*\hat{\underline{\mathbf{x}}}(t_i^+) \tag{1.9}$$

with $\hat{\underline{\mathbf{x}}}(t_i^+)$ produced by a constant-gain steady-state Kalman filter. The steady-state gain control law provided in Equation (1.9) will be the approach taken in this research. (For the remainder of this discussion, constant-gain algorithms will be assumed and $\overline{\mathbf{G}}_c^*$ will written simply as $\mathbf{G}_c^*$). The formulation of $\mathbf{G}_c^*$ is provided in Chapter 2. As shown in Figure 1.4, the LQG algorithm accepts sampled-data measurements from the system as inputs, and it outputs the optimal control to apply as constant values to the system over the next sample period, from t_i to t_{i+1}.

Six MMAE estimator/controller combinations are presented in this section. These techniques are:

- MMAC control
- Modified MMAC control
- MAP vs. Bayesian MMAC control
- Single fixed-gain controller based on $\mathbf{a}_{nom}$

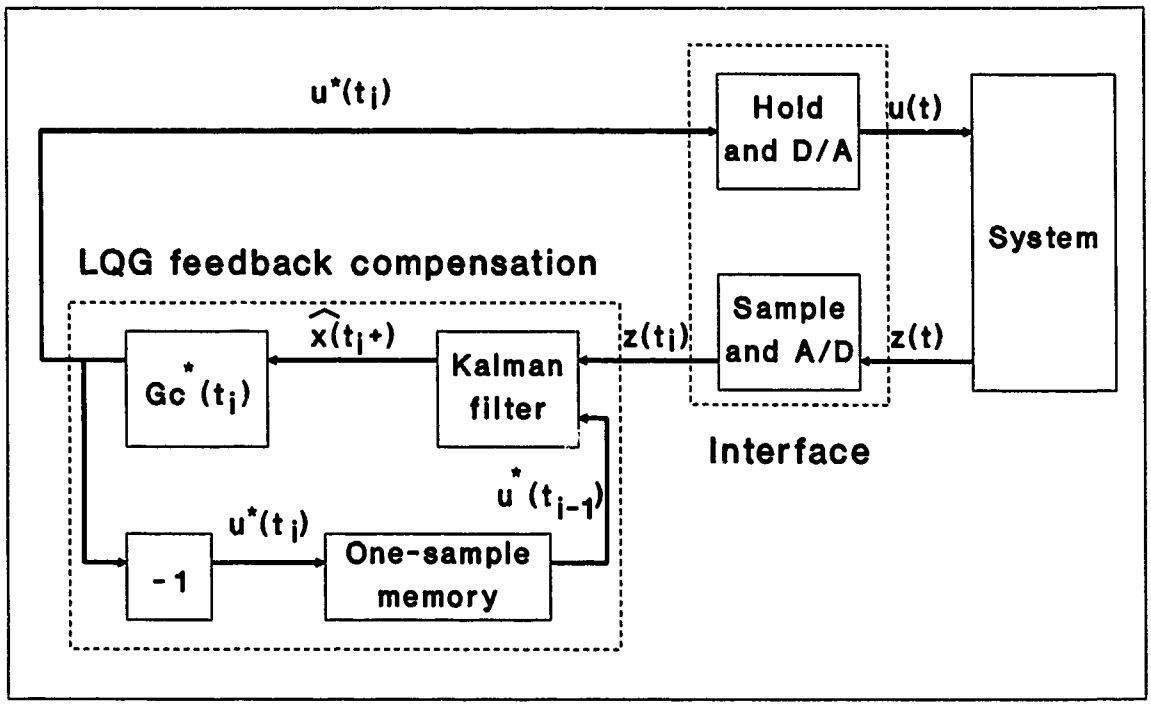

Figure 1.4. Diagram of Sampled-Data LQG State Regulator [17:19]

- Single changeable-gain controller based on $\hat{\mathbf{a}}(t_i)$
- Modified single changeable-gain controller based on $\hat{\mathbf{a}}(t_i)$

MMAC control consists of an elemental controller for each of the elemental filters of the bank, and the control outputs are probabilistically weighted and summed similar to that of the MMAE state estimates. Figure 1.5 illustrates the formation of the control vector for a full-bank estimator/controller combination. The blocks denoted by $-\mathbf{G}_c^*(\mathbf{a}_k)$ are optimal controller gains determined specifically for each discrete parameter value $\mathbf{a}_k$. The only difference between the full-bank MMAC and the moving-bank MMAC method is that the smaller number of estimators/controllers required by the moving-bank MMAC reduces computational loading. The performance of this approach suffers due to the fact that some magnitude (however small) of inappropriate control can be applied by filters based on incorrect models, particularly if lower bounds are placed on computed probabilities as discussed in Section 1.1.1.

Modified MMAC control is similar to the method described above. However, this approach consists of establishing a conditional probability threshold which must be obtained

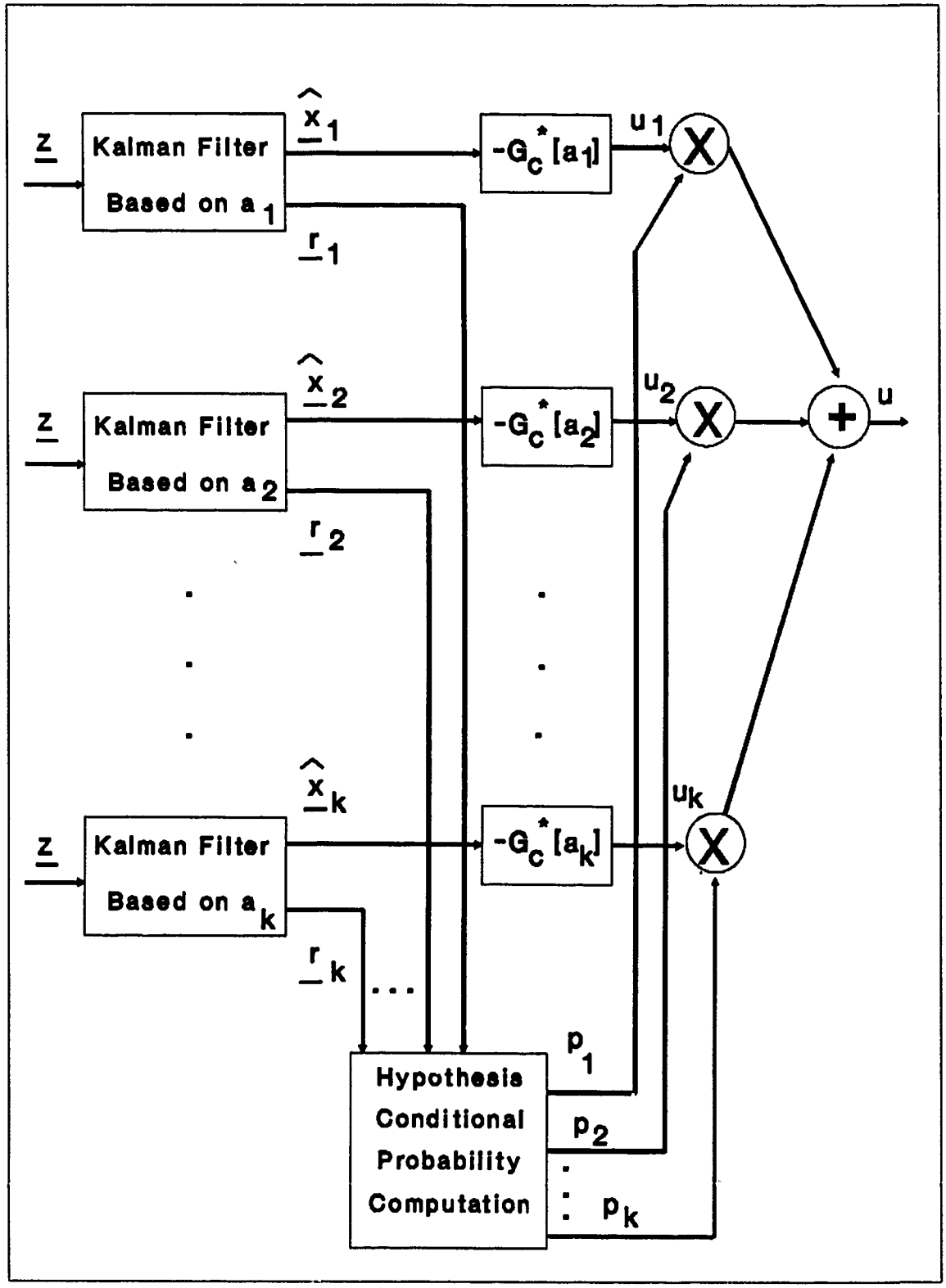

Figure 1.5. Diagram of Multiple Model Adaptive Controller [17:254]

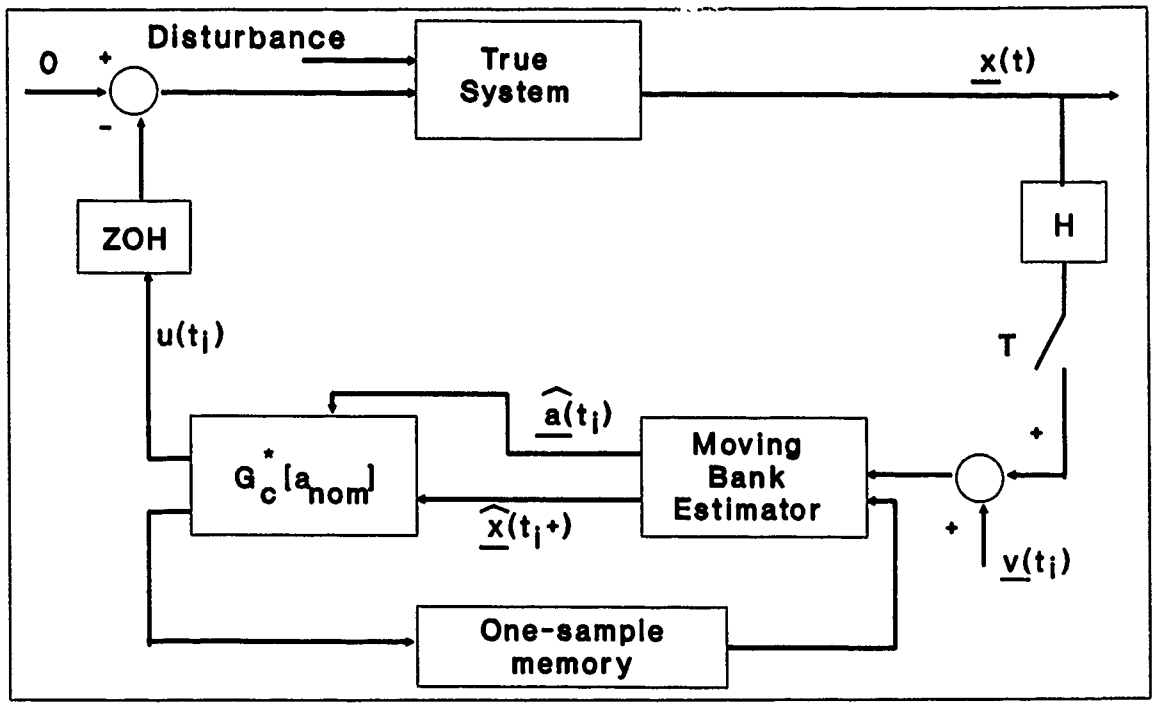

Figure 1.6. Diagram of Single Fixed-Gain Controller [7:41]

before any filter/controller combination of the moving bank can be allowed to apply control. This will reduce the amount of inappropriate control by requiring the probabilities associated with each filter to pass a "goodness" test before the controller associated with that filter can apply control.

MAP vs. Bayesian control is similar to the above approach. However, instead of establishing a probability threshold, this approach uses only the control from the moving-bank filter/controller combination with the highest conditional probability.

Single fixed-gain control is formulated by providing a state vector estimate to a fixed-gain controller, which is designed around a nominal value of the uncertain parameter set, a_{nom}. This method is illustrated in Figure 1.6. This method is reasonable since full-state feedback controllers are inherently robust [7:40]. This controller receives the state estimates from the moving-bank estimator and generates the control as follows:

$$\mathbf{u}^*(t_i) = -\mathbf{G}_c^*[\mathbf{a}_{nom}]\hat{\underline{\mathbf{x}}}(t_i^+) \tag{1.10}$$

The controller parameters are selected such that the controller provides adequate regulation for any true system parameter value [7:40]. The selection of $\mathbf{a}_{nom}$ may not be a trivial task.

Single changeable-gain control is formulated by providing both parameter and state vector estimates to a single controller with gains that are dependent on the parameter estimate. Figure 1.7(a) illustrates this approach. In this method, the control becomes as follows:

$$\mathbf{u}^*(t_i) = -\mathbf{G}_c^*[\hat{\mathbf{a}}(t_i)]\hat{\underline{\mathbf{x}}}(t_i^+) \tag{1.11}$$

For this method, a table of $\mathbf{G}_c[\mathbf{a}_i]$'s, one for each point in the parameter space, is established and used for interpolation in order to generate $\mathbf{G}_c^*[\hat{\mathbf{a}}(t_i)]$ [7:38].

Modified single changeable-gain control is similar to the method above. However, the parameter estimate, $\hat{\mathbf{a}}(t_i)$, from the MMAE is provided to a single filter/controller combination tuned specifically for that parameter value, illustrated in Figure 1.7(b). In this approach, the filter gains, as well as the controller gains, are interpolated from the parameter estimate. This is advantageous in reducing the possibility of underestimating the undamped natural frequency within any part of the algorithm that directly generates control inputs to the system, which has been previously shown to produce instability in this particular application [24].

Now that a brief overview has been provided on the methods used for estimation and control, it is appropriate at this time to provide a brief discussion of the structure which will be investigated in this research.

1.1.4 System Model. The purpose of this section is provide a brief overview of the system under investigation in this research. While this research does not perform the development of the system model being investigated, it is important to understand the concepts that were used in its development. A full description of the physical model and modal model development, as well as order reduction, will be provided in Chapter 3.

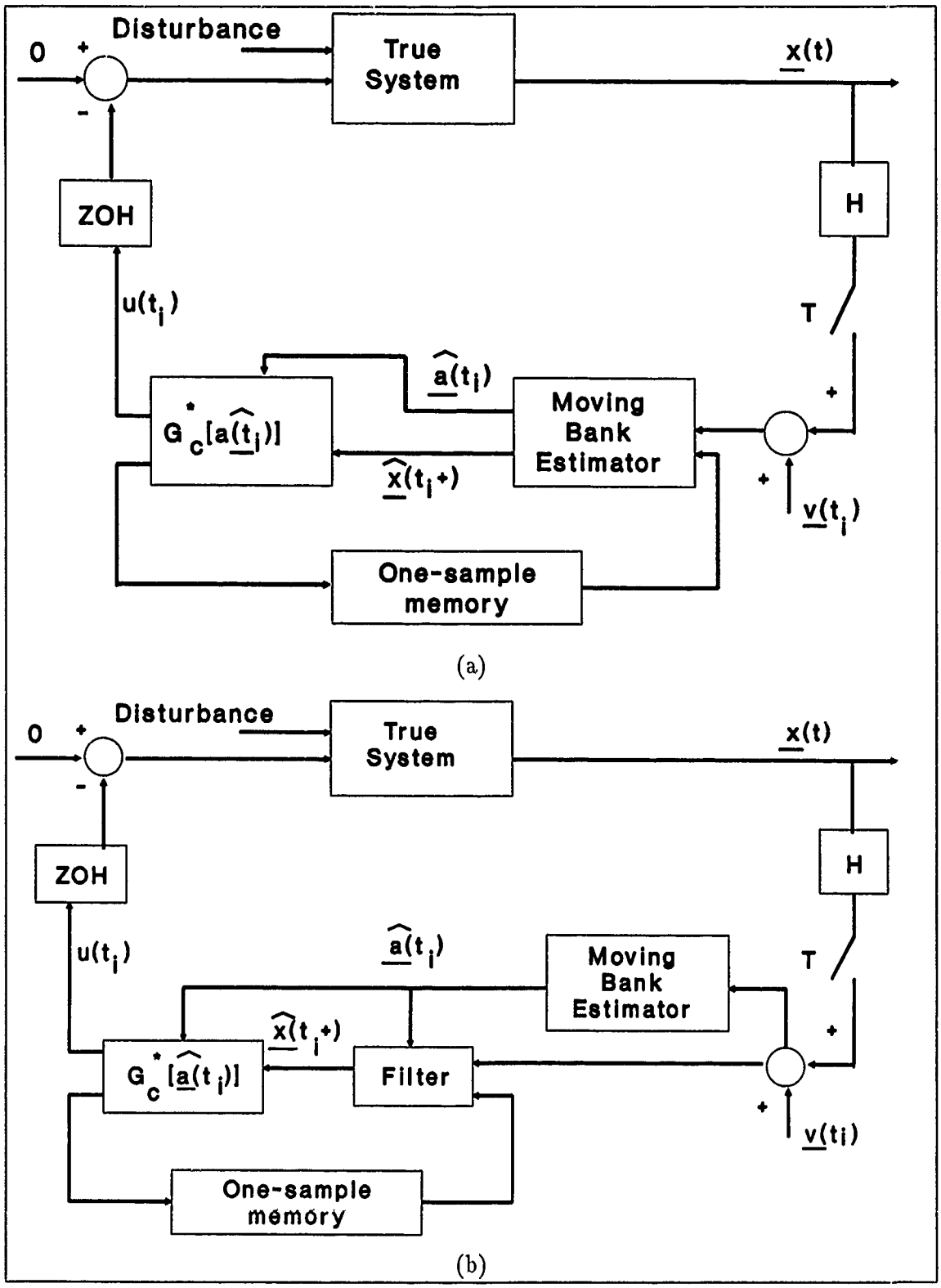

Figure 1.7. Diagram of Single Changeable-Gain Controller (a) Unmodified, and (b) Modified [7:37]

The initial feasibility study performed by Hentz investigated a simple second order system [7:16]. The control ratio of the system is given as:

$$\frac{C(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \tag{1.12}$$

where ζ is the damping ratio and ω is the undamped natural frequency. In state variable form (standard controllable form) the structure dynamics are described by:

$$\dot{\mathbf{x}}(t) = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} 0 \\ \omega_n^2 \end{bmatrix} u(t) \tag{1.13}$$

$$c(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \mathbf{x}(t) \tag{1.14}$$

The state vector, $\mathbf{x}(t)$, is composed of two components: $\mathbf{x}_1(t)$, which represents a position variable, and $\mathbf{x}_2(t)$, which represents a velocity variable.

The structure investigated by Filios consisted of four cantilevered appendages attached to a central hub [3]. This model was obtained from the Draper Laboratory/Rocket Propulsion Laboratory Configuration space structure [3:38]. Basically, this structure represented a satellite with four whip antennas. Unfortunately, Filios discovered that this model did not require on-line adaptation.

Following Filios, Karnick [8] obtained a space structure model that remains as the system used for investigation today. Karnick obtained a fixed, two-bay truss model which was originally developed to study the effects of structural optimization [27, 28] and optimal control design [14]. Karnick incorporated one major modification to the original two-bay truss he obtained. Basically, the truss was attached to a hub in order to enable investigation of rigid body motion [8:45]. Karnick used a six-state truth model and a six-state filter, both of which accounted for the rigid body mode and first two bending modes of the structure.

Figure 1.8 illustrates the physical description of the structure under investigation. This structure could represent an appendage of a larger space structure. The 13 aluminum rods that make up the structure are assumed to be of a certain cross-sectional area and elasticity. The additional non-structural masses attached to the structure act to lower the

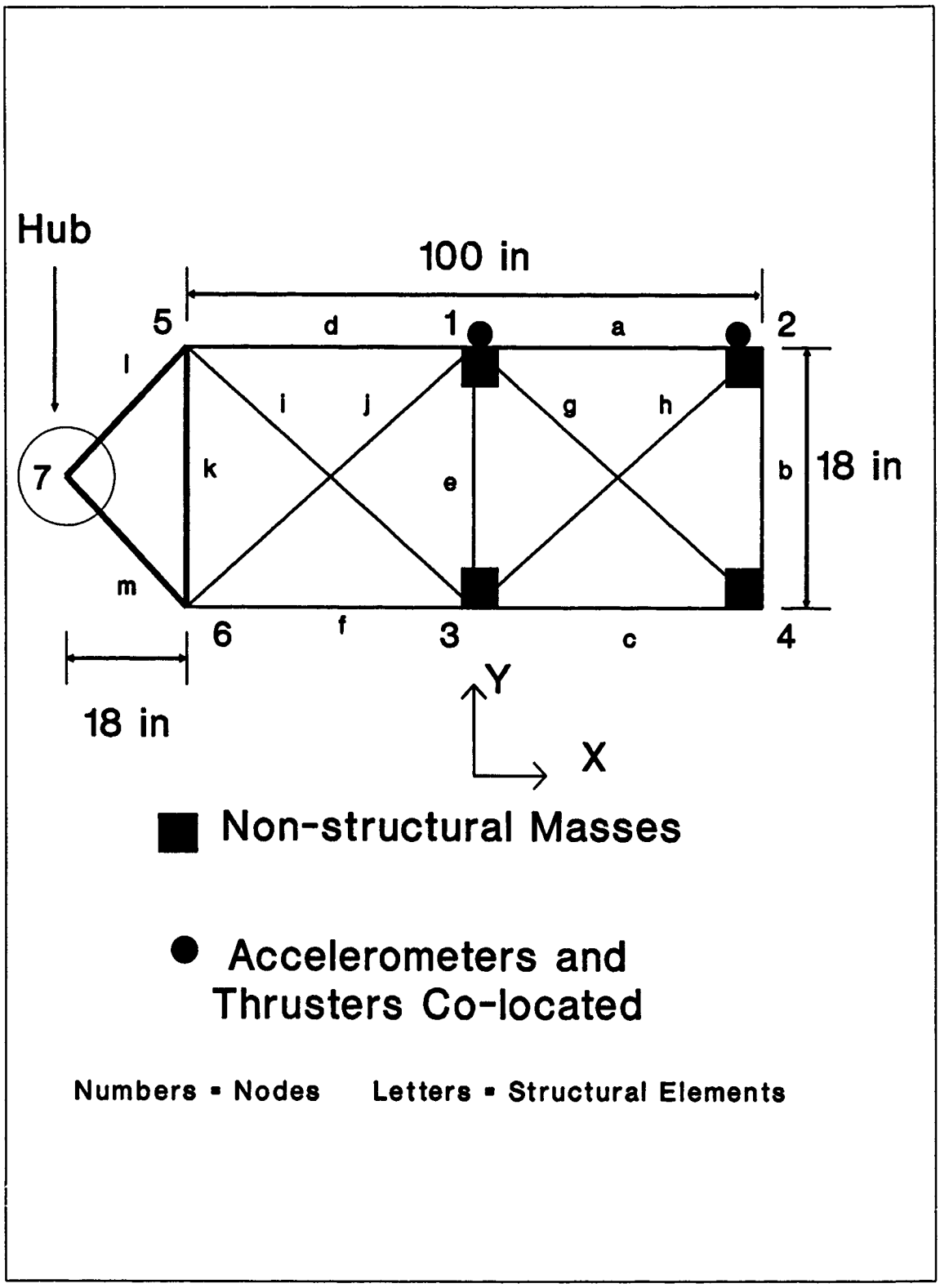

Figure 1.8. Diagram of Rotating Two-Bay Truss Model

structural frequencies, which makes this two-bay truss model representative of large space structures. The mass and stiffness matrices that describe the structure were obtained from a finite element analysis [28]. For the purposes of this research, the parameters which will be varied are the mass and stiffness matrices. It will be shown in Chapter 3 through modal analysis that the mass and stiffness matrix variations will affect the damping ratio and natural frequency of the structure. From the development of the mass and stiffness matrices, the truth model describing the structure contains twelve modes, the rigid body mode and eleven bending modes. Since each mode is represented by two states, one corresponding to position and one to velocity, the truth model for this structure contains 24 states.

In addition to the 13 rods and non-structural masses, the structure uses three sets of sensors (two on the truss and one on the hub) to obtain position and velocity measurements. The sensors on the truss consist of accelerometers and are located at the midpoint and at the end of the structure. Thrusters are co-located with each of the accelerometers to provide control inputs. Two gyroscopes are co-located at the hub; one provides angular displacement and the other provides angular velocity of the rigid structure. Finally, an inertia wheel is co-located at the hub to provide rigid body control inputs.

The work performed by Lashlee [11], Van Der Werken [26], and Schore [22] all continued the use of this model. Like Karnick, Lashlee's research implemented a six-state truth model and a six-state filter model. Van Der Werken, however, was concerned with the effects of unmodelled states. Therefore, the research he conducted implemented the 24-state truth model and a six-state filter model. Schore's research continued the work of Van Der Werken and therefore implemented the previous truth and filter models. As stated previously, the results obtained by Schore indicated that the six-state filter performed adequately when affected by the unmodelled states. Therefore, this research shall continue to use the same truth and filter models as employed by Schore.

Before the objectives of this research are stated, however, it is appropriate to provide a brief overview of the major accomplishments of past research. The next section provides a synopsis of the work that has been accomplished in this area by Hentz [7], Filios [3], Karnick [8], Lashlee [11], Van Der Werken [26], and Schore [22].

1.2 Past Research

In the past six years, research has been conducted in the area of moving-bank MMAE algorithms to reduce the computational loading needed to perform both system state and uncertain parameter estimation associated with the system model. Maybeck and Hentz conducted the initial feasibility study in 1984 [7] and showed, for a simple but physically motivated two state system, the moving-bank MMAE algorithms performed as well as the full-bank MMAE algorithm. In addition, this performance was obtained with an order of magnitude less computational loading. Hentz's study included a performance evaluation of the primary bank movement, expansion and contraction algorithms used by the moving bank as well as appropriate thresholds. Hentz also investigated several of the control techniques described previously. These investigations were conducted for slowly varying parameters and jump changes as well as constant parameters.

Follow-on research conducted by Filios [3] demonstrated that using ambiguity function analysis provides a useful method for evaluating the parameter estimation performance. In addition to formulating a useful estimator evaluation method, Filios also provided much insight into the establishment of movement and contraction thresholds. Unfortunately, the results obtained by Filios indicated that the more complex system he chose to use did not require adaptive control [3:93].

Following Filios, Karnick [8] applied the moving-bank algorithm to a 13-member, two-bay truss. However, the results that Karnick obtained indicated that the moving-bank MMAE was never able to identify the truth model parameters, even though it could sometimes provide accurate state estimates [8:93]. Karnick found that the moving-bank wandered nonsystematically throughout the parameter space and could not converge to a consistent parameter value. The final performance results showed that a coarsely discretized full-bank MMAE could perform as well as a finely discretized moving-bank MMAE [8:92]. The fundamental problem Karnick encountered was that measurement noise was severe enough to hamper the moving-bank adaptation process.

Lashlee [11], following on to Karnick's research, investigated the difficulties experienced by Karnick. Lashlee's research consisted of several studies including investigation of

the dynamics noise strength and measurement noise covariance, state and control weighting matrices, and parameter space discretization. The results of these studies indicated that a clear distinction between good and bad models must be maintained within the moving-bank algorithms. Once the appropriate noise values and parameter space discretization had been found, Lashlee showed that the moving-bank MMAE was able to estimate the parameters accurately and demonstrated greater performance potential than the fixed-bank MMAC he investigated [11:199].

Van Der Werken [26], continuing with the same structure as Karnick and Lashlee, investigated the effects of the order mismatch between a higher order truth model and a reduced order filter model. Whereas Karnick and Lashlee both utilized a filter model and truth model of the same dimensions (both consisted of six states), Van Der Werken conducted research to investigate the effect of the order mismatch between the original 24-state truth model and the reduced order six-state filter model [26:15]. The results obtained by Van Der Werken indicated that the moving-bank algorithm was able to produce neither accurate parameter nor state estimates when the bank was initially centered on a false parameter [26:183]. Van Der Werken concluded that the unmodelled states had a direct, negative impact on the ability of the MMAE algorithm to provide accurate estimates.

The most recent research on the moving-bank MMAE algorithms was conducted last year by Schore [22]. Schore's objective was to continue Van Der Werken's research to determine if the algorithm was truely confounded by the unmodelled states and, if so, determine the minimum states required in the reduced order model for adequate estimation. In order to accomplish this, Schore corrected some flaws in Van Der Werken's simulations and developed a more physically motivated approach to evaluate the performance of the reduced order filter model. Schore's research investigated how well the reduced order filter estimated the true total shape of the truss, as opposed to just the truth model states associated only with the first three modes [22:Chapter 1 pg 22]. The performance of the moving-bank algorithm was evaluated as the affects of the higher order states were gradually allowed to influence the system. In addition, Schore investigated the disturbance rejection performance of the algorithms. Schore's results indicated that, even though the estimation process of the algorithm was "sensitive" to the reduced order mismatch, the

degradation in performance was not sufficient to warrant increasing the six-state filter model dimension [22:Chapter 6 pg 1].

1.3 Problem Statement

The basic problem with the full-bank MMAE/MMAC approach is that the number of Kalman filters (and controllers) required creates an unbearable computational load. The moving-bank MMAE/MMAC approach is an alternative to the full-bank MMAE/MMAC and has previously been shown to be successful for controlling a structure in the face of unmodelled effects [22:Chapter 6 pg 1]. The performance of the moving-bank demonstrated in this research, however, was less than optimal. The tuning of the Kalman filter dynamics noise strength, $\mathbf{Q}(t)$, and the measurement noise covariance, $\mathbf{R}(t)$, the discretization strategy used to generate the parameter points, as well as tuning of the other parameters in the filter algorithm, affect the system substantially. This research will continue the work started by Van Der Werken and Schore by providing insight into performance optimization of the six-state filter evaluated against the 24-state truth model.

1.4 Scope

The research performed investigates problems associated with large space structures. The model space structure selected for this research is represented by a two-bay truss developed by Karnick [8]. The two-bay truss is 100 inches long and 18 inches high. The truss is attached to a hub and is allowed to move only in the x-y plane. Four non-structural masses attached to the truss provide time varying mass problems. The addition of these masses, provided that they are large relative to the structural mass, also produce the low frequency structural model associated with large space structures. The structure is described by mass and stiffness matrices obtained from a finite element analysis performed previously by Karnick [8].

The uncertain parameters of interest for this research are the non structural masses and the structural stiffness [8, 27]. These parameters were chosen because they physically appear in the system dynamic equations.

1.5 Approach

The moving-bank algorithm performance evaluation will be accomplished by measuring how closely the filter estimates of positions and velocities at physical points on the truss match the true values, and then how well the associated controller regulates those states. Research efforts prior to Schore investigated the performance of the moving-bank algorithms based on how well the elemental filters estimated the truth model states associated with only the first three modes [22:Chapter 1 pg 22]. In these investigations, the elemental filters and truth models both were six-state models. Schore's research, however, evaluated the performance of the elemental filters in a more physically motivated manner by investigating how well reduced order filters could estimate (and the regulators could control) the true total shape of the truss, i.e., the shape due to all the modes' effects. For his performance evaluation, Schore defined a new error term as the difference between true and estimated positions and velocities at three locations on the truss. These three locations consist of the accelerometer and thruster positions at the midpoint and at the end of the truss and at the gyroscope and inertia wheel position at the hub (See Figure 1.8). This research will adopt the technique discussed above in evaluating the optimal performance of the moving bank.

Optimal estimation performance of the moving-bank algorithm will be achieved by evaluating the estimation process sensitivities during the following studies:

- Kalman filter dynamics noise strength, $\mathbf{Q}(t)$, and measurement noise covariance, $\mathbf{R}(t_i)$, value determination
- Residual, parameter position estimate, and probability monitoring performance
- Density function "bias" determination
- Parameter space discretization
- Performance evaluation with varying parameters

Optimal control performance of the moving-bank algorithm will be achieved by evaluating the control process sensitivities during the following studies:

- State and control weighting value determination
- Optimal control vector formulation
- Parameter space discretization
- Performance evaluation with varying parameters

For each of the studies, the state estimation and control performance of the moving bank is compared to the performance bound of a single Kalman filter/controller that has artificial knowledge of the true parameter. The primary figure of merit for the estimation process is the state estimation error and the parameter estimation error. The primary figure of merit for the controller process is how well the moving-bank controller regulates the state statistics to zero as compared to the benchmark. The state statistics will be generated by performing a Monte Carlo study. The number of error process samples obtained by simulation will be adequate such that the true process statistics are approximated well by the Monte Carlo study.

Previous research efforts have shown that system complexity [11] and unmodelled effects [22] do not markedly degrade the performance of the algorithm as long as the elemental filters in the bank are readily distinguishable from one another. Previous research has shown that the major areas of concern for preserving this "distinguishability" are the evaluation of the Kalman filter $\mathbf{Q}(t)$ and $\mathbf{R}(t)$ values and determination of the proper discretization of the parameter space. For this research, the parameter space discretization will be investigated as part of both the estimator study and the LQG controller study. Since past research has demonstrated that control performance is significantly affected by the parameter estimation [11], [24], parameter space discretization will be investigated primarily as a means to enhance the performance of the LQG controllers.

Kalman Filter $\mathbf{Q}(t)$ *and* $\mathbf{R}(t)$ *value determination* is of prime importance for the overall performance of the moving bank. (The form of $\mathbf{Q}(t)$ and $\mathbf{R}(t)$ are constant and therefore the time argument will be omitted.) It is important not to mask good versus bad filter models with too much dynamics psuedonoise in an attempt to keep each elemental filter within the MMAE structure from diverging. Each filter must be tuned for best performance when the true value is identical to the assumed value [11]. Previous research

limited the form of the $\mathbf{Q}$ and $\mathbf{R}$ to diagonal matrices. Based upon the use of the same sensor for position and velocity measurements, a measurement noise covariance matrix with non-zero off-diagonal elements might be more appropriate [22:Chapter 3 pg 4]. In addition, previous efforts at determining the appropriate values were conducted by equating $\mathbf{Q}_f$ to $\mathbf{Q}_t$ and $\mathbf{R}_f$ to $\mathbf{R}_t$ [11] (where t denotes truth model and f denotes filter model); this is reasonable if the filter-assumed model and truth model are the same, i.e., no order reduction between the truth model and filter design model (as was the case for Lashlee's research). Van Der Werken's research attempted to evaluate the effects of filter mismatch in computing $\mathbf{R}_f$, where there is an even stronger desire to use a non-diagonal $\mathbf{R}_f$. However, Van Der Werken still used a diagonal matrix to represent $\mathbf{R}_f$ and a single matrix for $\mathbf{Q}$. Schore experienced numerical difficulty with the large values of $\mathbf{Q}$ used by Lashlee and ended up reducing these values by an order of magnitude [22:Chapter 4 pg 10]. Research in this area will begin with the determination of the appropriate values for $\mathbf{Q}_t$ and $\mathbf{R}_t$. This research shall then incorporate the minimum variance reduced order (MVRO) estimator method [16:25] as used previously by Van Der Werken [26]. This method will help in the determination of values for $\mathbf{R}_f$ and $\mathbf{Q}_f$ after appropriate values for $\mathbf{Q}_t$ and $\mathbf{R}_t$ have been determined. This method will account for the unmodeled 18 states and produce non-diagonal matrices.

Residual, parameter position estimate, and probability monitoring will be conducted, as alternatives to monitoring for the bank-moving logic, in order to investigate the performance of the algorithms as suggested by Schore. This research will evaluate the performance of the bank using parameter position estimate monitoring, which Hentz's research determined to provide the best overall performance (along with probability monitoring) [7:87].

Density function "bias" determination will investigate any biasing effect due to the nature of the computation of $p_k(t_i)$. This is motivated by several sources. First, recent research conducted by Stevens [21] indicated that the determination of the $p_k(t_i)$, given in Equation (1.1) was artificially biased by the leading coefficient on the Gaussian density numerator term in the conditional density function computation, given in Equation (1.2). This coefficient contains the term $\mid \mathbf{A}_k(t_i) \mid^{1/2}$ in its denominator. If all elemental

filters' residuals look "equally good" relative to their anticipated covariance, so that all of the exponential terms of Equation (1.2) were essentially the same value, then the $p_k(t_i)$ computations most heavily weight the filter associated with the smallest $|\ \mathbf{A}_k(t_i)\ |$. Since $|\ \mathbf{A}_k(t_i)\ |$ is precomputable and has nothing to do with online adequacy of models, this is an inappropriate bias. Second, previous thesis work by Filios [3:64], Karnick [8:36], and Lashlee [11:43] removed the scale factor associated with the $|\ \mathbf{A}_k(t_i)\ |$ term from the conditional density computation. This was performed to reduce numerical difficulties they were experiencing in determinant evaluations during ambiguity function analysis. However, by so doing, the resulting expression is no longer a true density function since the scale factor is not correct. However, because of the denominator term in Equation (1.1), the probability weightings are still correct since they sum to one [3:65]. Research in this area will entail removing the scale factor term from the $p_k(t_i)$ computations and comparing the performance to a "biased" estimator.

Parameter space discretization is the second area of prime importance for preserving the distinguishability between the elemental filters. The parameters must not be spaced so closely that the elemental filter models cannot be distinguished from one another, however not so far apart that the elemental filters (and associated controllers) must be too robust. Recent research [24] has shown that incorrect parameter estimation has a significant impact on LQG controller performance. Underestimation of the higher modal frequencies resulted in poor controller performance [24]. Hentz's earlier research into this area resulted in a linear parameter space discretization for ζ and a logarithmic discretization for ω_n [7:20]. Following Hentz, a linear space discretization based on the mass and stiffness matrices was developed. Lashlee's research, while investigating the problems Karnick experienced, developed a non-linear discretization [11]. This discretization provided an rms error growth of 20 to 30 percent between each step in the parameter space. Theses three previous space discretizations are provided in Table 1.1. This research shall begin by using the parameter space discretization determined by Lashlee for for performing tuning of the bank filters. Then, the insights gained from the work of Sheldon [24] shall be incorporated to attempt to enhance the discretization and, hence, the performance of the estimator (and controller). The performance enhancement of the controller due to parameter discretization will be

Table 1.1. Discretized Parameter Space for Previous Research

a_k	Hentz [7]		Linear [11]		Lashlee [11]	
⇓	ζ	ω (rad/sec)	Mass	Stiffness	Mass	Stiffness
1	0.0000	6.28	0.50	0.80	0.50	0.50
2	0.1111	8.12	0.60	0.84	0.55	0.60
3	0.2222	10.48	0.70	0.88	0.60	0.70
4	0.3333	13.54	0.80	0.92	0.70	0.80
5	0.4444	17.48	0.90	0.96	0.80	0.90
6	0.5556	22.58	1.00	1.00	0.90	1.00
7	0.6667	29.16	1.10	1.04	1.10	1.16
8	0.7778	37.67	1.20	1.08	1.20	1.26
9	0.8889	48.65	1.30	1.12	1.30	1.40
10	1.0000	62.83	1.40	1.16	1.40	1.50

investigated in the simulations performed for the LQG controller study.

Performance evaluation with varying parameters will investigate the performance of the moving bank estimator when subjected to changing parameters. The majority of past efforts investigated only constant parameters. Hentz and Filios did limited work with evaluating the performance of the moving bank to slowly and smoothly changing parameters and to minor jump changes. This research shall conduct a study to determine the performance due to slowly varying and jump changes throughout the *entire* parameter space. The slowly varying parameter study will be conducted by providing a time history for the parameters. The jump changes will consist of jumping the value of the parameter more than one discretized value away. Research will determine if it is better to expand the bank size to re-acquire or let the bank follow on its own without altering its size. This research will entail re-evaluation of the thresholds for contraction and expansion. As for the space discretization study, the performance of the LQG controllers for varying parameters will be investigated during the LQG controllers studies.

State and control weighting value determination will investigate the LQG controller

performance sensitivity to altered (but constant) cost function definition, as given by Equation (1.6). In previous research [11:87], the state weighting matrix, **X**, values were determined by holding the control weighting matrix, **U**, constant and increasing the **X** values one at a time until the rms values for the true states stopped decreasing drastically. After the **X** values were determined, the **U** values were found by holding the **X** values constant and decreasing the **U** values until the rms values for the true states stopped decreasing drastically. This method was applied to parameter location (7,6), i.e., the mass value at its seventh discrete value and the stiffness parameter at its sixth discrete value. This location accounts for only one of the possible 100 points in the parameter space. These values for **X** and **U** were then used for the other points in the parameter space. The cross weighting matrix, **S**, values for the generalized quadratic cost in the LQG synthesis were also investigated by Lashlee and found to be negligible. Therefore, the values for **S** will be kept at zero for this study. This research, as suggested by Lashlee and Schore, will determine the state and control weighting matrices in a manner similar to that described above. This will be accomplished for each parameter in the parameter space.

Control vector formulation will investigate alternate methods for forming the control vector. All of the previous research efforts since Hentz have investigated forming the control vector by weighted-averaging of the control for all of the filter/controller combinations in the bank using the computed hypothesis conditional probabilities as weighting coefficients (MMAC). This resulted in some magnitude, however small, of inappropriate control being applied, particularly when the computed probabilities were artificially bounded from below to prevent elemental filter lockout. This research shall continue to investigate MMAC in addition to three alternate methods. These methods include the the modified MMAC approach, the MAP vs Bayesian MMAC approach, and the modified single changeable-gain controller approach, which were discussed previously. This study will evaluate an appropriate threshold for the modified MMAC approach as well as to evaluate performance of all four approaches.

Parameter space discretization will be investigated to determine the impact on appropriate adaptations and bank motion and on robustness of the elemental controllers. As stated previously, Sheldon's research [23] has shown that underestimation of the higher

modal frequencies resulted in poor (unstable) controller performance. Using the parameter space discretization previously devised for the parameter estimation studies, the performance of the LQG controllers will be evaluated. For this study, the straightforward MMAC approach will be used to determine the effects of the modified parameter space on the control of the structure.

Performance evaluation with varying parameters will investigate the performance of the moving bank controller when subjected to changing parameters. This research shall conduct a study to determine the controller performance due to slowly varying and jump changes throughout the *entire* parameter space. The varying parameter study for the controller will be the same as for the estimator study, which will allow using the same expansion and contraction thresholds.

1.6 Summary

The purpose of this chapter was twofold. First, it provided the background necessary to facilitate a basic understanding of the objectives which this research intends to accomplish. Second, this chapter presented the approach which will be adopted to achieve the stated objectives. The background presented concepts of the moving-bank multiple-model adaptive estimation and controller as well as the large space structure which will be used to investigate these concepts. The background also provided a brief synopsis of past research in this area to illustrate the evolution of this research. The remaining chapters of this thesis cover the following areas. Chapter 2 develops the algorithms used in the MMAE and LQG controller portions of this research, including Kalman filter theory and decision logic for moving and expanding/contracting the bank of filters/controllers. The modelling of the large space structure, the two bay truss structure, is developed in Chapter 3. The simulations performed by this thesis will be explained in Chapter 4. Chapter 5 presents the results of this research, with corresponding conclusions and recommendations presented in Chapter 6.

II. Algorithm Development

2.1 Introduction

The background discussion provided in Chapter 1 was intended to provide basic understanding of the problem statement and research approach. However, the purpose of the development provided here is twofold. First, the discussion of Chapter 1 will be embellished to address the general theoretical development relevant to this research. Second, discussion contained here will highlight simplifying assumptions and insights gained from previous research which are pertinent to the accomplishment of the objectives specific to this research. Algorithm development will encompass (1) Kalman filter development, (2) multiple model adaptive estimation development, (3) moving-bank MMAE development, and (4) LQG controller development. While the discussion contained here will add more insight to the discussions of Chapter 1, for an exhaustive development see references [15, 16, 17]. As stated previously, the notation used is intended to be consistent with the convention found in [15] such that a stochastic process is denoted by $\underline{\mathbf{x}}$ while a deterministic process is denoted by $\mathbf{x}$. The cornerstone of the MMAE concept is the Kalman filter. Therefore, it is suitable at this point, before discussing any of the algorithms specific to MMAE, to provide a brief discussion of what has been described simply as "...an optimal, recursive, data processing algorithm [15:4]".

2.2 Kalman Filter Development

It is assumed, for the purposes of this research, that the system under investigation is adequately described by the continuous, linear, stochastic system model given by:

$$\dot{\underline{\mathbf{x}}}(t) = \mathbf{F}(t)\underline{\mathbf{x}}(t) + \mathbf{B}(t)\mathbf{u}(t) + \mathbf{G}(t)\underline{\mathbf{w}}(t) \tag{2.1}$$

driven by deterministic controls and zero-mean white Gaussian noise of strength $\mathbf{Q}(t)$, or more properly,

$$d\underline{\mathbf{x}}(t) = \mathbf{F}(t)\underline{\mathbf{x}}(t)dt + \mathbf{B}(t)\mathbf{u}(t)dt + \mathbf{G}(t)d\underline{\beta}(t) \tag{2.2}$$

driven by Brownian motion $\underline{\beta}(t)$ of diffusion $\mathbf{Q}(t)$. In the above equations, $\underline{\mathbf{x}}(\cdot)$ represents an n-state vector process, $\mathbf{u}(\cdot)$ is an r-vector deterministic control input, $\mathbf{F}(\cdot)$ is an n-by-n system dynamics matrix, $\mathbf{B}(\cdot)$ is an n-by-r deterministic input matrix, and $\mathbf{G}(\cdot)$ is an n-by-s noise input matrix. The model described by Equation (2.1) will be used for the remainder of this discussion. The statistics of $\underline{\mathbf{w}}(\cdot)$ are given by:

$$E\{\underline{\mathbf{w}}(t)\} = \mathbf{0} \tag{2.3}$$

$$E\{\underline{\mathbf{w}}(t)\underline{\mathbf{w}}(t')^T\} = \mathbf{Q}(t)\delta(t-t') \tag{2.4}$$

where $\mathbf{Q}(t)$ is an s-by-s matrix that is symmetric and positive semidefinite and $\delta(t)$ is the Dirac delta function.

The state differential equation given by Equation (2.1) is propagated forward from some initial condition, $\mathbf{x}(t_0)$. Since the initial condition is not generally known precisely a priori, it will be modelled as a random vector with mean $\hat{\mathbf{x}}_0$ and covariance $\mathbf{P}_0$ given by:

$$E\{\underline{\mathbf{x}}(t_0)\} = \hat{\mathbf{x}}_0 \tag{2.5}$$

$$E\left\{[\underline{\mathbf{x}}(t_0) - \hat{\mathbf{x}}_0][\underline{\mathbf{x}}(t_0) - \hat{\mathbf{x}}_0]^T\right\} = \mathbf{P}_0 \tag{2.6}$$

where $\mathbf{P}_0$ is an n-by-n matrix that is symmetric and positive semidefinite. Allowing $\mathbf{P}_0$ to be positive semidefinite provides for the case of perfectly knowing some of the initial states or combinations thereof.

Measurements are available from the system at discrete time points (sampled-data measurements) and are modeled by the relation given by:

$$\underline{\mathbf{z}}(t_i) = \mathbf{H}(t_i)\underline{\mathbf{x}}(t_i) + \underline{\mathbf{v}}(t_i) \tag{2.7}$$

where $\underline{\mathbf{z}}(\cdot)$ is an m-vector discrete-time measurement process, which provides a particular measurement time history for each sample. Measurement noise, $\underline{\mathbf{v}}(t_i)$, accounts for the uncertainty with which the measurements are obtained. The measurement noise is an

m-vector discrete-time, white Gaussian process with statistics given by:

$$E\{\underline{\mathbf{v}}(t_i)\} = \mathbf{0} \tag{2.8}$$

$$E\{\underline{\mathbf{v}}(t_i)\underline{\mathbf{v}}(t_j)^T\} = \mathbf{R}(t_i)\delta_{ij} \tag{2.9}$$

where $\mathbf{R}(t_i)$ is an m-by-m, symmetric, positive definite matrix and δ_{ij} is the Kronecker delta function. Requiring $\mathbf{R}(t_i)$ to be positive definite implies that all components of the measurement vector are noise corrupted. The measurement model in Equation (2.7) also assumes that the system dynamics noise $\underline{\mathbf{w}}(\mathrm{t})$, the measurement noise $\underline{\mathbf{v}}(t_i)$, and $\underline{\mathbf{x}}(t_0)$ are independent of each other. Since all of these quantities are assumed to be Gaussian, this is essentially the same as stating that they are uncorrelated with each other [15:205].

Using the system model given in Equation (2.1), the measurement model from Equation (2.7), and the statistical descriptions of the uncertainties, a Bayesian point of view can be adopted to develop the form of Kalman filter model [15:205]. Since the algorithms developed here will be eventually implemented on a digital computer, it is desirable to formulate discrete-time algorithms. Two methods are available for designing the discrete-time Kalman filter algorithms. First, the designer could take the continuous-time system model Equation (2.1), design the continuous filter, and then discretize the result. Second, the designer could determine an *equivalent* discrete-time model and generate the discrete-time filter from it. The preferable design approach is to discretize the model first and then generate the filter [15:261]. The *equivalent*, stochastic difference equation describing the system model in general is given by:

$$\underline{\mathbf{x}}(t_i) = \mathbf{\Phi}(t_i, t_{i-1})\underline{\mathbf{x}}(t_{i-1}) + \mathbf{B}_d(t_{i-1})\mathbf{u}(t_{i-1}) + \mathbf{G}_d(t_{i-1})\underline{\mathbf{w}}_d(t_{i-1}) \tag{2.10}$$

where the matrices $\mathbf{\Phi}(t_i, t_{i-1})$, $\mathbf{B}_d(t_{i-1})$, $\mathbf{G}_d(t_{i-1})$, and the covariance $\mathbf{Q}_d(t_{i-1})$ of the noise $\underline{\mathbf{w}}_d(t_{i-1})$ are derived from the $\mathbf{F}(\cdot)$, $\mathbf{B}(\cdot)$, $\mathbf{G}(\cdot)$ and $\mathbf{Q}(\cdot)$ matrices provided in Equations (2.1) and (2.4). The following discussion provides general formulation of these matrices.

The state transition matrix, $\mathbf{\Phi}(t_i, t_{i-1})$, is derived using the system dynamics matrix, $\mathbf{F}(\cdot)$. It is assumed in this research that the $\mathbf{F}(\cdot)$ matrix is constant; therefore, the inverse

LaPlace transform of the resolvent matrix can be used as:

$$\Phi(t_i, t_{i-1}) = \Phi(t_i - t_{i-1}) = \mathcal{L}^{-1}\left\{[s\mathbf{I} - \mathbf{F}]^{-1}\right\}|_{(t_i - t_{i-1})} \tag{2.11}$$

Assuming the control $\mathbf{u}$ is held constant over a sample period, the deterministic input matrix, $\mathbf{B}_d(t_{i-1})$, is the discrete-time, deterministic input matrix given by:

$$\mathbf{B}_d(t_{i-1}) = \int_{t_{i-1}}^{t_i} \Phi(t_i, \tau)\mathbf{B}(\tau)d\tau \tag{2.12}$$

The discrete-time, zero-mean, white Gaussian system dynamics noise vector, $\underline{\mathbf{w}}_d(t_{i-1})$, has a strength that is a function of the state transition matrix, the noise input matrix, and the strength of the continuous-time $\underline{\mathbf{w}}(\mathrm{t})$. These statistics are given by:

$$E\{\underline{\mathbf{w}}_d(t_{i-1})\} = 0 \tag{2.13}$$

$$E\{\underline{\mathbf{w}}_d(t_{i-1})\underline{\mathbf{w}}_d(t_j)^T\} = \mathbf{Q}_d(t_i)\delta_{(i-1)j} \tag{2.14}$$

where $\mathbf{G}_d$ is assumed to be the identity matrix and $\delta_{(i-1)j}$ is the Kronecker delta function. $\mathbf{Q}_d(t_{i-1})$ is given by:

$$\mathbf{Q}_d(t_{i-1}) = \int_{t_{i-1}}^{t_i} \Phi(t_i, \tau)\mathbf{G}(\tau)\mathbf{Q}(\tau)\mathbf{G}^T(\tau)\Phi^T(t_i, \tau)d\tau \tag{2.15}$$

Now that the structure of the system model, as well as the uncertainties, have been specified for all times of interest, a Kalman filter can now be completely specified. The Kalman filter algorithms for the conditional mean state estimate and covariance time propagation relations can be written as [15:220]:

$$\hat{\mathbf{x}}(t_i^-) = \Phi(t_i, t_{i-1})\hat{\mathbf{x}}(t_{i-1}^+) + \mathbf{B}_d(t_{i-1})\mathbf{u}(t_{i-1}) \tag{2.16}$$

$$\mathbf{P}(t_i^-) = \Phi(t_i, t_{i-1})\mathbf{P}(t_{i-1}^+)\Phi^T(t_i, t_{i-1}) + \mathbf{G}_d(t_{i-1})\mathbf{Q}_d(t_{i-1})\mathbf{G}_d^T(t_{i-1}) \tag{2.17}$$

These equations provide the propagation relationship between two consecutive state

estimates and covariances from time t_{i-1}^+ to time t_i^-. The"+" and "-" superscripts indicate that the variable of interest is evaluated just after or just before a measurement is incorporated, respectively. In order to incorporate the discrete-time system measurements, given by Equation (2.7), the following update equations are used:

$$\mathbf{K}(t_i) = \mathbf{P}(t_i^-)\mathbf{H}^T(t_i)\left[\mathbf{H}(t_i)\mathbf{P}(t_i^-)\mathbf{H}^T(t_i) + \mathbf{R}(t_i)\right]^{-1} \tag{2.18}$$

$$\hat{\mathbf{x}}(t_i^+) = \hat{\mathbf{x}}(t_i^-) + \mathbf{K}(t_i)\left[\mathbf{z}(t_i) - \mathbf{H}(t_i)\hat{\mathbf{x}}(t_i^-)\right] \tag{2.19}$$

$$\mathbf{P}(t_i^+) = \mathbf{P}(t_i^-) - \mathbf{K}(t_i)\mathbf{H}(t_i)\mathbf{P}(t_i^-) \tag{2.20}$$

In the above equations, there are two quantities of particular interest which need to be highlighted. The first quantity of interest, which is of prime importance in determining the performance of the Kalman filter, is contained in the brackets of Equation (2.19). This bracketed term is called the filter residual (or innovations) [15:228] and is denoted by $\mathbf{r}(t_i)$. The residual indicates how much correction is required by the filter, since it is the difference between the most recent measurement and the best prediction of that measurement based on prior measurements. The residual is weighted by $\mathbf{K}(t_i)$ and added to the previous estimate of the state to arrive at the new estimate of the state. The term $\mathbf{K}(t_i)\mathbf{r}(t_i)$ in Equation (2.19) is referred to as the new information. It can be shown [15:229] that the filter residual sequence is a zero-mean, white Gaussian sequence with known covariance of $\left[\mathbf{H}(t_i)\mathbf{P}(t_i^-)\mathbf{H}^T(t_i) + \mathbf{R}(t_i)\right]$, which is the second quantity of interest. As can be seen from the bracketed term in Equation (2.18), the "expected" covariance of residuals is provided by the filter algorithms. This is the residual covariance $\mathbf{A}_k(t_i)$ term associated with each Kalman filter in the bank. This term is used in calculating the probability density function shown in Equation (1.2). These residual properties can be exploited for checking the reasonableness of measurement data and other forms of adaptation. Explicitly in connection with this research, the residuals, in conjunction with the $\mathbf{A}_k(t_i)$ term discussed above, are of prime importance for moving-bank MMAE residual monitoring mentioned in Section 1.1.2 and described by Equation (1.3).

The previous discussion provided a short and straightforward development of the

Kalman filter. However, "... a substantial amount of engineering insight and experience is required to develop an effective, operational filter algorithm [15:289]". The following discussion provides the engineering insight used to develop an effective algorithm for this research.

Choosing values for $\hat{\mathbf{x}}_0$, $\mathbf{P}_0$, $\mathbf{Q}_d(t_i)$, and $\mathbf{R}(t_i)$, often referred to as "tuning" the Kalman filter, requires judicious selection. For example, increasing $\mathbf{Q}_d(t_i)$ would indicate either stronger noises driving the dynamics or increased uncertainty in the adequacy of the model itself to depict the true dynamics accurately [15:224]. This dictates that the filter should put less confidence in its own dynamics model. Increasing $\mathbf{R}(t_i)$ would indicate that the measurements are subject to a stronger corruptive noise, and so should be weighted less by the filter [15:224]. In this research, conservative approaches which may be possible in other applications cannot be applied. Such practices as adding dynamics pseudonoise to guard against elemental filter divergence will not be possible. It has been shown previously [11:198] that selection of such values will degrade MMAE performance if they are chosen so large as to mask the difference between the discretized parameters. In addition, it has been shown [18:7] that each filter should be tuned for best performance when the "true" values of the parameters are identical to its assumed value for theses parameters. Because of this concern, the technique known as loop transmission recovery (LTR) [14:7] will not be used in this research to determine the appropriate noise values. Chapter 4 will discuss evaluation of the appropriate noise strength values.

Filter performance for systems which are time-invariant with stationary noises can be described by an initial transient in $\mathbf{P}(t_i)$ and $\mathbf{K}(t_i)$ followed by an essentially steady state filter operation [15:224]. In many applications, the transient is short compared to the total time of interest. This suggests a possible approximation of using the steady-state filter for all time if the resulting performance degradation in not excessive. In addition, as can be seen from the $\mathbf{P}(t_i^-)$ in Equation (2.17), $\mathbf{P}(t_i^+)$ in Equation (2.20), and $\mathbf{K}(t_i)$ in Equation (2.18), these quantities can be precomputed and stored for later use, thus reducing the real-time computational loading. Finally, a time-invariant system with stationary noises and a fixed sampling rate allows the one-time computation of the $\mathbf{B}_d(t_{i-1})$ and $\mathbf{Q}_d(t_{i-1})$ matrices in Equations (2.12) and (2.15). For this research, the $\mathbf{R}(t_i)$ matrix will also be

considered constant. Chapter 3 will provide a detailed description of the system and filter matrix development specific to this research.

2.3 Bayesian MMAE Development

The basic concept of the MMAE algorithm was presented in Chapter 1. While the following material is presented for completeness, a thorough development is presented by Maybeck [16:129-136]. The purpose of the Bayesian estimator is to compute the conditional density function:

$$f_{\underline{\mathbf{x}}(t_i),\mathbf{a}|\underline{\mathbf{Z}}(t_i)}(\xi,\alpha \mid \mathbf{Z}_i) = f_{\underline{\mathbf{x}}(t_i)|\mathbf{a},\underline{\mathbf{Z}}(t_i)}(\xi \mid \alpha, \mathbf{Z}_i) f_{\mathbf{a}|\underline{\mathbf{Z}}(t_i)}(\alpha \mid \mathbf{Z}_i) \tag{2.21}$$

Equation (2.21) is the conditional density of $\underline{\mathbf{x}}$ and $\mathbf{a}$ given the measurement history through the current time, where $\mathbf{Z}_i$ is composed of partitions equal to the realizations of $\underline{\mathbf{z}}(t_1), \underline{\mathbf{z}}(t_2), \ldots, \underline{\mathbf{z}}(t_i)$. Let $\mathbf{a}$ be the vector of uncertain parameters for the model under study which can effect any or all of $\mathbf{\Phi}(t_i, t_{i-1})$, $\mathbf{B}_d(t_{i-1})$, $\mathbf{G}_d(t_{i-1})$, $\mathbf{H}(t_i)$, $\mathbf{Q}_d(t_i)$, or $\mathbf{R}(t_i)$. The parameter vector $\mathbf{a}$ can assume values in the continuous range $A \subset R^p$, where R^p is real Euclidean p-dimensional space. The parameter vector may be "uncertain but constant, it may be slowly varying, or it may undergo jump changes" [11:21]. The nature of $\mathbf{a}$ being continuous would cause a problem computationally since the calculations to solve for the conditional density would require an infinite number of separate Kalman filters and integrations that would make online usage of the Bayesian estimator prohibitive. To allow online computation, the parameter space is discretized. The parameter vector might be defined by selecting a finite set of discrete values, $\{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_K\}$, that are dispersed throughout the region of reasonable parameter values. Each discrete value $\mathbf{a}_k$ has an individual system model associated with it, requiring computation of the matrices describing Equations (2.7) and (2.10) for each value of $\mathbf{a}_k$. For this research, 100 discrete parameter values are possible which requires 100 different $\mathbf{\Phi}(t_i, t_{i-1})$, $\mathbf{B}_d(t_{i-1})$, $\mathbf{G}_d(t_{i-1})$, and $\mathbf{H}(t_i)$ system matrices to be calculated and stored.

The conditional density function provided Equation (2.21) can be described by discussing the two terms on the right hand side of the equation. The first density on the

right hand side is produced by each elemental Kalman filter based on the assumption that $\mathbf{a} = \alpha$. The second density on the right hand side is given as $\sum_{k=1}^{K} p_k(t_i)\delta(\alpha - \mathbf{a}_k)$. The state estimate produced by the adaptive filter is given by [16:131]:

$$\begin{aligned}\hat{\mathbf{x}}(t_i^+) &= E\{\underline{\mathbf{x}}(t_i) \mid \underline{\mathbf{Z}}(t_i) = \mathbf{Z}_i\} \\ &= \int_{-\infty}^{\infty} \xi \left[\sum_{k=1}^{K} f_{\underline{\mathbf{x}}(t_i)|\underline{\mathbf{a}},\underline{\mathbf{Z}}(t_i)}(\xi \mid \mathbf{a}_k, \mathbf{Z}_i) p_k(t_i)\right] d\xi \\ &= \sum_{k=1}^{K} \hat{\mathbf{x}}_k(t_i^+) \cdot p_k(t_i) \end{aligned} \tag{2.22}$$

where $\hat{\mathbf{x}}_k(t_i^+)$ is the state estimate produced by a Kalman filter based on the assumption that the parameter equals $\mathbf{a}_k$. The state estimate is the sum of all the probabilistically weighted estimates generated by the K Kalman filters, where the hypothesis conditional probabilities $p_k(t_i)$ are the weighting factors. The generation of $p_k(t_i)$ was provided in Section 1.1.1, Equation (1.1).

The form of the conditional covariance of the state is similar to that of Equation (2.22) and is given by [16:131]:

$$\begin{aligned}\mathbf{P}(t_i^+) &= E\left\{\left[\underline{\mathbf{x}}(t_i) - \hat{\mathbf{x}}(t_i^+)\right]\left[\underline{\mathbf{x}}(t_i) - \hat{\mathbf{x}}(t_i^+)\right]^T \mid \underline{\mathbf{Z}}(t_i) = \mathbf{Z}_i\right\} \\ &= \sum_{k=1}^{K} p_k(t_i)\left\{\mathbf{P}_k(t_i^+) + \left[\hat{\mathbf{x}}_k(t_i^+) - \hat{\mathbf{x}}(t_i^+)\right]\left[\hat{\mathbf{x}}_k(t_i^+) - \hat{\mathbf{x}}(t_i^+)\right]^T\right\} \end{aligned} \tag{2.23}$$

The conditional covariance $\mathbf{P}_k(t_i^+)$ is the "state error covariance" associated with the Kalman filter based on the parameter $\mathbf{a}_k$. The conditional covariance is dependent on $p_k(t_i)$, $\hat{\mathbf{x}}_k(t_i^+)$, and $\hat{\mathbf{x}}(t_i^+)$ as shown in Equation (2.23), and therefore is not precomputable. However, it is not necessary to compute it for online filter implementation.

The other estimate that may be desired or required is that of the parameter vector, $\hat{\mathbf{a}}(t_i)$. The conditional mean of the parameter vector at time t_i is given by [16:132]:

$$\begin{aligned}\hat{\mathbf{a}}(t_i) &= E\{\mathbf{a}(t_i) \mid \underline{\mathbf{Z}}(t_i) = \mathbf{Z}_i\} \\ &= \int_{-\infty}^{\infty} \alpha f_{\mathbf{a}|\underline{\mathbf{Z}}(t_i)}(\alpha \mid \mathbf{Z}_i) d\alpha \\ &= \sum_{k=1}^{K} \mathbf{a}_k \cdot p_k(t_i) \end{aligned} \tag{2.24}$$

The covariance of the estimated parameter vector gives an indication of the precision of the estimate, and it can be estimated via [16:133]:

$$\begin{aligned}\mathbf{P}_{\hat{\mathbf{a}}} &= E\{[\mathbf{a} - \hat{\mathbf{a}}(t_i)][\mathbf{a} - \hat{\mathbf{a}}(t_i)]^T \mid \underline{\mathbf{Z}}(t_i) = \mathbf{Z}_i\} \\ &= \sum_{k=1}^{K} [\mathbf{a}_k - \hat{\mathbf{a}}(t_i)][\mathbf{a}_k - \hat{\mathbf{a}}(t_i)]^T \cdot p_k(t_i) \end{aligned} \tag{2.25}$$

As with the calculation of the state error covariance, Equation (2.23) , the calculations of the parameter estimate, Equation (2.24), and the covariance of the parameter estimate, Equation (2.25), are not required for the calculation of the state estimate [16:133].

The *multiple model adaptive estimation algorithm* developed above is an adaptive filter structure and was shown in Figure 1.1. Discussion of the performance of the MMAE will follow two general topics: first, fundamental performance of the algorithm based on elemental filter residual generation, and second, discussion of performance to varying parameters.

As discussed in Section 1.1.1, the characteristic of the Kalman filter residual is the key to MMAE performance. If a particular filter is nearest to the true parameter location, the residual from that filter would be expected to be smallest in magnitude (relative to the filter-computed covariance $\mathbf{A}_k(t_i)$) of the active bank of filters. Equation (1.2) would then provide the largest conditional density value for this filter and when applied to Equation (1.1), would ultimately yield the highest probability to the "best" filter. However, if residuals were of the same "goodness", i.e. if the quadratic forms within the exponential of Equation (1.2) were all of the same magnitude, the value of $\mid \mathbf{A}_k(t_i) \mid$ would begin to dominate. Under this condition, the $p_k(t_i)$'s are dominated by the filter with the lowest

$| A_k(t_i) |$ due to the leading coefficient in Equation (1.2). However, the $| A_k(t_i) |$ values are independent of not only the residuals, but also of the relative correctness of the K models being used. Therefore, the resulting $p_k(t_i)$ values would be totally erroneous [16:133]. Previous work by Filios [3:64], Karnick [8:36], and Lashlee [11:43] investigated removing this term in performing ambiguity function analysis. More recently, Stevens [21:12] investigated the "bias" caused by this term in the p_k computations. As mentioned previously, this research will investigate this issue.

Another form of conditional density function computation, which may be used if no confidence can be given to the precomputed residual covariance, entails the removal of the A_k term from the exponential, as well as from the coefficient of Equation (1.2). If it is assumed that the filter residuals are Gaussian with a covariance equal to the identity matrix, it is assumed that the residuals follow a "maximally non-committed residual distribution" [23:32]. In this case, the conditional density function computation can be given as:

$$f_{\underline{Z}(t_i)|\underline{a},\underline{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_k, \mathbf{Z}_{i-1}) = \frac{1}{(2\pi)^{\frac{m}{2}} \mid \mathbf{I} \mid^{\frac{1}{2}}} \exp\{-\frac{1}{2}\mathbf{r}_k^T(t_i)\mathbf{I}\mathbf{r}_k(t_i)\} \qquad (2.26)$$

Equation (2.26) is termed a Maximum Entropy with Identity Covariance (ME/I) density computation [23, 24]. The ME/I density computation results in the filter with the lowest residual autocorrelation (i.e., absolute magnitudes, not magnitudes scaled by filter-computed standard deviations) being given the highest $p_k(t_i)$ value. During investigations of the A_k term, the ME/I density function may be investigated if it is determined that less than full confidence can be given to A_k.

Based on the previous discussion, it is essential to ensure that the residuals of "good" versus "bad" filters have very distinguishing characteristics. The "distinguishability" of the elemental filters will be accomplished first by the means of tuning each elemental filter and second, by the manner of discretizing the parameter space.

Tuning of the elemental Kalman filters was discussed in Section 2.2. In addition to the methods related to Kalman filter tuning in general, a simplifying approach will be taken to tune the dynamics driving noise strength for the filters used in the MMAE algorithm. For this research, a single, non-adaptive, artificially informed filter will be tuned based on

the correct parameter and the values selected for this case will be considered representative of all the filters in the entire parameter space.

Description of an appropriate parameter space has been investigated previously by Hentz [7:20] and Lashlee [11:81]. More recently, Sheldon and Maybeck [24] have proposed an optimizing design strategy for parameter space discretization. This strategy chooses the parameter set by minimizing a cost functional representing the average state prediction error autocorrelation for estimator design, and mean square regulation error for controller design. The average state estimation or regulation error autocorrelation is taken as the true parameter range over the admissible parameter set [24:1]. As stated previously, this methodology will be investigated by this research.

The second area of discussion for MMAE performance is performance of the algorithm to varying parameters. A problem which can result from varying parameters has been investigated in previous thesis efforts and has been described as filter "lock out". "Lock out" occurs when the MMAE algorithm gives any one elemental Kalman filter a probability, $p_k(t_i)$ of Equation (1.1), of zero. As seen by the iterative nature of the equation, the $p_k(t_i)$ for this filter cannot take on a value other than zero once the $p_k(t_i)$ equals zero. This means that, even if parameter began to vary and the filter thereby became based on the correct parameter, the MMAE algorithm would continue to ignore the estimates from the elemental filter. Filter "lock out" causes the moving-bank MMAE to lose its ability to produce adaptive state estimates effectively [11:28].

Unfortunately, there are no complete theoretical results available for varying parameters [1, 6, 11]. However, there are two methods available for avoiding "lock out". One method is to add dynamics pseudonoise to the assumed model for each elemental filter [16:25]. However, as discussed previously, pseudonoise would tend to mask the difference between "good" and "bad" models. The second method to prevent filter "lock out" is to establish a lower threshold, p_{min}, on the $p_k(t_i)$. This has been the approach in past research efforts [7, 8, 11, 22, 26]. The hypothesis conditional probabilities are monitored and when one or more probabilities fall below the threshold, they are set to the minimum and the entire bank rescaled to maintain the unity sum of the probabilities.

2.4 *Moving-Bank Algorithm Development*

The MMAE filter algorithm presented in the previous section presents an immense computational burden if implemented in a full bank form, i.e., a filter for every discrete point in the parameter space. As discussed previously, this research will investigate the technique of a moving bank of filters to lessen the computational loading. In general, this moving bank will contain J filters, where $J < K$. Specifically, the moving bank investigated in this research will consist of nine elemental filters (J=9) where the parameter space will consist of 100 filters (K=100). The moving-bank MMAE technique was originally investigated by Maybeck and Hentz [19]. Their research provided the basic techniques for moving this smaller bank. This section discusses the technique for (1) moving the bank, (2) expanding the bank, (3) contracting the bank, and (4) initializing new filters.

2.4.1 Moving the Bank. The moving-bank MMAE is a subset of the full-bank MMAE. Conceptually, the bank is centered around the best estimate of the parameter. The knowledge of the parameter may be uncertain a priori or the parameter location may change. Two courses of action are available when it is detected that the true parameter position is outside the region of the moving bank. The bank can either be moved in the direction of the true parameter point so as to encompass it, or the bank can be expanded so as to encompass it. This section discusses the four basic decision logics used to move the bank: (1) residual monitoring, (2) parameter position monitoring, (3) parameter position and "velocity" monitoring, and (4) probability monitoring. The threshold values used for these techniques must be determined through performance evaluation. This section provides a description of the logic as well as some insight into previous work performed in establishing appropriate thresholds for these methods.

2.4.1.1 Residual Monitoring. The effect of the residuals on the calculation of the probability density function was discussed in Section 1.1.1 and shown in the exponential term of Equation (1.2). In Section 1.1.2, this exponential term was defined as the likelihood quotient, $L_j(t_i)$, based on the active elemental filters in the bank. This was discussed in relation to Equation (1.3). Considered in the scalar case, the likelihood quotient is the ratio of the residual squared divided by the filter-computed variance for the residual. Consider

the effect of the parameter being outside the area of the filter. The residual values would become large and drive all the J likelihood quotients high. A preset threshold could be determined through performance evaluations such that when the smallest L_j exceeds some threshold an appropriate filter movement action could be determined. In other words, a possible detection scheme would be to move the bank when

$$min(L_1, L_2, \ldots, L_j) \geq T_{resid} \tag{2.27}$$

Additionally, the filter closest to the true parameter should have the smallest likelihood quotient and thus provide an indication of where to move the bank. In determining T_{resid}, Hentz noted, as expected, that when the threshold was set too high the moving-bank took longer to identify the parameters [7:61]. When the threshold was set too low, the moving bank failed to maintain estimation of the true parameters. The use of residual monitoring should be limited, however, to situations where the system is not subject to "single large samples of measurement noise" [19:1876]. Examination of Equation (1.3) shows that the values of $L_j(t_i)$ would all rise appreciably in the face of a sudden high value of $\underline{\mathbf{v}}(t_i)$ which affects the residual directly through the realized measurement value. This would result in an erroneous decision to move the bank.

2.4.1.2 Parameter Position Estimate Monitoring. Equation (2.24) gave an expression for the estimated parameter location based on the K discretized parameter locations (now based on the J discrete moving-bank filter locations) and hypothesis conditional probabilities. The estimated parameter location is used in this technique for determining where the center of the moving bank should be. If the parameter location begins to move to the edge of the current bank or to move to some predetermined distance from the current center, then the decision to move the bank could be made. Note that the bank cannot always be centered over the estimated parameter position, as when the estimated position is near the outer edges of the overall parameter space. Centering the bank in this case would require that some elemental filters lie outside the overall parameter space. Reviewing Equation (2.24) shows that the benefit of this technique is that it relies on a history of measurements rather than just on the single current measurement. Thus it

is not prone to the single-large-sample measurement noise problem of residual monitoring.

2.4.1.3 Parameter Position and "Velocity" Estimate Monitoring. This technique is an extension of the previous parameter position monitoring technique. If the parameter is varying slowly, the position and "velocity" can both be monitored. The parameter velocity is the change in parameter location between sample periods or over an extended period of time. The velocity vector could be used to estimate where the parameter will be at the next sample period. If the predicted location is outside the current bank of filters or beyond a certain threshold distance, $T_{predicted\ position}$, from the current center, then a move of the bank is indicated in the direction of the velocity vector. Evaluation of $T_{predictedposition}$ by Hentz indicated that this method showed no real enhancement in speed to acquire, and important destabilization of the bank's position in parameter space [7:62].

2.4.1.4 Probability Monitoring. Monitoring the hypothesis conditional probabilities generated by Equation (1.1) provides insight into which filter is located nearest the true parameter location. Using a preset threshold, T_{prob}, the bank can be moved in the direction of the filter providing the most correct parameter estimate, i.e., the one with the highest $p_j(t_i)$ value which is also above T_{prob}. The bank seeks to center itself on the filter which is based on the most correct assumed parameter value. As with parameter position monitoring, probability monitoring uses a time history of measurements and is less susceptible to radical changes due to individual large realizations of measurement noise. Hentz found that the performance of the bank improved as T_{prob} was decreased from high values to low [7:62]. Essentially, with a low T_{prob}, the bank moved anytime the probability weightings on the perimeter elemental filters exceeded the weighting on the center filter. Hentz found this method and the parameter estimate monitoring method to provide the best performance. However, probability monitoring has the additional benefit of a slightly less costly computational loading than position monitoring [7:87].

2.4.2 Expanding the Bank. The size of the bank need not be fixed and the bank filters need not be at adjacent finely discretized points in the parameter space. If the active filters are not associated with adjacent discrete parameter values, the bank behaves as a

coarsely discretized moving bank. A coarsely discretized bank was illustrated in Figure 1.3(b). The estimates provided by such a bank may not provide as accurate estimates as a bank where the active filters occupy adjacent discrete parameter locations. However, it does offer a higher probability that the true parameter location will be within the bank. This is a good attribute for two specific cases. First, the bank may require expansion to a coarser discretization for the case of a jump change of the true parameter location. Second, a coarse discretization may be desired for initialization of the moving bank. Expansion of the bank due to a jump parameter change is discussed first while the initial acquisition cycle is discussed in the next section.

Using residual monitoring, a jump change in the parameter can be detected and the bank expanded. After a parameter jump, the likelihood ratios of the implemented filters are expected to be large, exceeding some threshold , T_{expand}. Residual monitoring is used because the needed information (that none of the current filters in the bank correspond to a "good" assumed parameter value) is not available from either the covariance of $\hat{\mathbf{a}}_k$ or $\hat{\mathbf{a}}_k$ itself [7:68]. When the bank expands, it is set to an initial coarse discretization with its center at (5,5). Essentially, the moving bank starts the initial acquisition cycle over when an expand decision is made. Evaluation of T_{expand} by Hentz determined that, if it was set too high, the bank waited too long after the parameter jump before expanding. When T_{expand} was set too low, the bank expanded inappropriately during the runs [7:69]. Using residual monitoring for bank expansion has the same problem as that for bank movement; in the face of a sudden high value of $\underline{\mathbf{v}}(t_i)$, the values of $L_j(t_i)$ would all rise appreciably, which would result in an erroneous decision to expand.

2.4.3 Contracting the Bank. As discussed previously, a coarsely discretized bank could result from the decision to expand the bank due to a jump parameter change, or during the initial acquisition cycle, where convergence has a higher probability of occurring. Maybeck and Hentz found that using an initial coarse discretization of the bank improved parameter acquisition [19]. As stated previously, the estimates provided by such a bank may not be as accurate as those provided by a bank where the active filters occupy adjacent discrete parameter locations. Therefore, once the parameter estimate is made, it

is desirable to contract the bank to a finer discretization as soon as possible. The following discussion will highlight the acquisition cycle and the determination of the contraction threshold.

In the initial acquisition cycle, the discrete parameter value spacing is set such that the moving bank covers nearly the entire parameter space. In this way, the true parameter will always be within the region spanned by the large moving bank. The covariance of the parameter estimate is then used to contract the bank as appropriate. The formulation of $\mathbf{P}_{\hat{\mathbf{a}}}$ was given in Equation (2.25). One scheme to contract this coarsely discretized bank is to contract the spacing in two steps (see [7:64] and [3:79]), as opposed to the alternative of being contracted to finest discretization in one step. The bank initially has a spacing of four between the elemental filters. When the covariance of the parameter estimate drops below a specified threshold, $T_{contract1}$, the bank contracts to an intermediate spacing of two. Finally, when the covariance of the parameter estimate drops below a second threshold, $T_{contract2}$, the bank contracts to its finest discretization, a spacing of one. Each time the bank contracts, it contracts such that its center is the discretized parameter point nearest to $\hat{\mathbf{a}}_k$. This has the disadvantage of all the nine filters generally going through an initial transient. Another option would be to contract the bank to the quadrant of the parameter space before contraction that contained the estimated parameter point position. This ensures that four filters of the bank before contraction remain in the bank after contraction. This is desirable in order to minimize the impact of initial transients in the state estimates [3:80]. For this research, the bank will be contracted in one step and will be centered on the discretized parameter point nearest to $\hat{\mathbf{a}}_k$.

It is important to note that the covariance of the parameter estimate in this research is a two-by-two matrix. In order to compare this matrix to the threshold level, a scalar value associated with this matrix is needed. Hentz used the larger of the two diagonal elements, after performing an inverse mapping on the ω estimated variance to give both elements the same relative magnitude [7:64]. (Recall that Hentz's parameter discretization for the ω parameter was logarithmic.) In determining the values for these thresholds, Hentz found that if the thresholds were set too high, the bank contracted before a good parameter estimate could be obtained. If the thresholds were set too low, the bank did not contract

soon enough, degrading performance while it is in the coarse mode. Filios [3] found that requiring *both* variances to be below the thresholds gave a more accurate determination of when to contract. However, this required tradeoffs in the determination of the threshold, since the same threshold was not necessarily the most appropriate for both variances. Filios found that the threshold he used proved to be sensitive to the probability weight lower bound, p_{min}, used to keep the bank from "locking out" any elemental filters. A potentially better idea, and the method which will be incorporated into this research, is to contract in separate directions using separate diagonal terms, which allows for rectangular banks.

Karnick proposed an alternative method in which the probability of a side of the bank was monitored such that

$$p_{\text{side}}(t_i) = \frac{\sum_{\text{side}} f_{\underline{\mathbf{z}}(t_i)|\mathbf{a},\underline{\mathbf{Z}}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_j, \mathbf{Z}_{i-1})}{\sum_{4 \text{ sides}} f_{\underline{\mathbf{z}}(t_i)|\mathbf{a},\underline{\mathbf{Z}}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_j, \mathbf{Z}_{i-1})} \tag{2.28}$$

is the probability associated with each side [8:27-29]. If the side probability falls below some threshold, then the side is contracted. "Conversely, if the probability associated with a side rises above some threshold, the remaining three sides are 'moved in' " [8:29]. The third possibility involves evaluating all four sides, and when the sum of the sides' probabilities fall below an appropriately determined threshold, the bank is contracted. This method is totally ad hoc.

2.4.4 Initialization of New Elemental Filters. When the bank is moved, expanded, or contracted, three processes must occur. First, the matrices $\mathbf{\Phi}$, $\mathbf{B}_d$, $\mathbf{G}_d$, $\mathbf{H}$, $\mathbf{A}_k$, $\mathbf{D}$ (which will be discussed Section 3.4 of Chapter 3), $\mathbf{K}$, $\mathbf{P}(t_i^-)$, and $\mathbf{P}(t_i^+)$ for each elemental filter (and $\mathbf{G}_c^*$ for each elemental controller) are changed to correspond to the new parameter points. Second, the changed filters must be initialized with new values for $\hat{\mathbf{x}}_j(t_i)$. And third, the changed filters must be initialized with new values for $p_j(t_i)$.

The first process is achieved by simply retrieving the matrices which have been previously calculated and stored. The second process can be achieved by selecting $\hat{\mathbf{x}}_j(t_i)$ for the changed filters as the current moving-bank estimate of the system state, $\hat{\mathbf{x}}(t_i^+)$. For the third process, several options have been investigated [8:29-32].

The first option for initialization of $p_j(t_i)$ is to set all of the $p_j(t_i)$'s (*for both changed and unchanged* elemental filters) to $\frac{1}{9}$. This option has been shown to result in sluggish convergence to a parameter estimate. A second option is to redistribute the sum of the discarded filter probabilities equally among the new filters. A third method, proposed by Hentz, is to re-assign the total discarded filter probability based on the "correctness" of the new filters [7:29]. A new filter's "correctness" is based on the computation of its probability density function value for the current measurement relative to the sum of the changed filters probability density functions' values. Hentz's algorithm is given as:

$$p_{j_{ch}}(t_i) = \frac{f_{\underline{z}(t_i)|\underline{a},\underline{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_j, \mathbf{Z}_{i-1})}{\sum_{ch} f_{\underline{z}(t_i)|\underline{a},\underline{Z}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_j, \mathbf{Z}_{i-1})} \left[1 - \sum_{unch} p_j(t_i)\right] \tag{2.29}$$

where ch=changed and unch=unchanged elemental filters. However, this method requires additional computation and has demonstrated no significant performance improvement over equal re-distribution of the discarded probabilities [3:76]. The special case of bank expansion results in the resetting of all the filters in the bank. Dividing the probability among all the filters is appropriate since the old probability weightings may no longer be valid. In all of the methods described above, the sum of the probabilities of the bank after the reset must equal one (as mentioned in Section 1.1.1); this may require a rescaling step.

Finally, in addition to investigating how to initialize new filters, Hentz investigated the need for "warming up" the new filters before they were actually brought on line. This warmup would allow initial transients in state estimates and conditional probabilities to die out. Hentz's results indicated that there was no detectable improvement in performance when filter warm up was used [7:100]. This research will use equal redistribution of the discarded filters and no filter warmup.

2.5 Stochastic Controller Development

As stated in Section 1.1.3, the moving-bank and fixed-bank MMAE algorithms can be used with several stochastic controller designs. The controller implementations investigated in this research will be the MMAC, the modified MMAC, the MAP vs Bayesian MMAC, and the modified single changeable-gain controller. The "assumed certainty equivalence

design" methodology presented by Maybeck [17:241-245] will be used. Each controller developed is a linear, quadratic cost, full-state feedback optimal deterministic controller based upon a specific assumed parameter value of a_k. As discussed previously, the output of the controller is desired to be the optimal control function, such that a quadratic cost function is minimized, in order to cause the closed loop system to behave properly, quickly quelling out any bending vibrations. The following discussion provides the basic development of the general quadratic cost function in order to highlight aspects which are of concern to this research.

As developed in Maybeck [17:73], consider a continuous-time system such as that provided in Equation (2.1) to which it is desired to apply sampled-data control of the form $\mathbf{u}(t) = \mathbf{u}(t_i)$ for all t. What we are really concerned with is the behavior of the continuous-time system for all $t \in [t_0, t_{N+1}]$ and not just at the sample times t_i. Assume that the control objective will be realized when the following continuous-time quadratic cost function is minimized:

$$J = E\left\{\int_{t_0}^{t_{N+1}} \frac{1}{2}\left[\underline{\mathbf{x}}^T(t)\mathbf{W}_{xx}(t)\underline{\mathbf{x}}(t) + \mathbf{u}^T(t)\mathbf{W}_{uu}(t)\mathbf{u}(t) + 2\underline{\mathbf{x}}^T(t)\mathbf{W}_{xu}(t)\mathbf{u}(t)\right]\right.$$
$$\left. + \frac{1}{2}\underline{\mathbf{x}}^T(t_{N+1})\mathbf{X}_f\underline{\mathbf{x}}(t_{N+1})\right\} \tag{2.30}$$

where the $\mathbf{W}_{xx}(t)$ is positive semidefinite and $\mathbf{W}_{uu}(t)$ is positive definite for all t. Following the development in [17], the cost can be equivalently expressed in discrete time by dividing the desired time interval into $(N+1)$ control intervals which yields the following:

$$J = E\left\{\sum_{i=0}^{N} \frac{1}{2}\left[\underline{\mathbf{x}}^T(t_i)\mathbf{X}(t_i)\underline{\mathbf{x}}(t_i) + \mathbf{u}^T(t_i)\mathbf{U}(t_i)\mathbf{u}(t_i) + 2\underline{\mathbf{x}}^T(t_i)\mathbf{S}(t_i)\mathbf{u}(t_i) + L_r(t_i)\right]\right.$$
$$\left. + \frac{1}{2}\underline{\mathbf{x}}^T(t_{N+1})\mathbf{X}_f\underline{\mathbf{x}}(t_{N+1})\right\} \tag{2.31}$$

where

$$\mathbf{X}(t_i) = \int_{t_i}^{t_{i+1}} \Phi^T(t, t_i)\mathbf{W}_{xx}(t)\Phi^T(t, t_i)dt \tag{2.32}$$

$$\begin{aligned} \mathbf{U}(t_i) &= \int_{t_i}^{t_{i+1}} \left[\overline{\mathbf{B}}^T(t,t_i)\mathbf{W}_{xx}(t)\overline{\mathbf{B}}(t,t_i) + \mathbf{W}_{uu}(t) \right. \\ &\quad \left. + \overline{\mathbf{B}}^T(t,t_i)\mathbf{W}_{xu}(t) + \mathbf{W}_{xu}^T(t)\overline{\mathbf{B}}(t,t_i)\right] dt \end{aligned} \tag{2.33}$$

$$\mathbf{S}(t_i) = \int_{t_i}^{t_{i+1}} \left[\Phi^T(t,t_i)\mathbf{W}_{xx}(t)\overline{\mathbf{B}}^T(t,t_i) + \Phi^T(t,t_i)\mathbf{W}_{xu}(t)\right] dt \tag{2.34}$$

and

$$\overline{\mathbf{B}}(t,t_i) \equiv \int_{t_i}^{t} \Phi(t,\tau)\mathbf{B}(\tau)d\tau \tag{2.35}$$

The residual cost, $L_r(t_i)$, in Equation (2.31) ha no bearing on the evaluation of the optimal control function and is therefore not evaluated here [17:76]. Examination of Equation (2.34) reveals that there are two instances where the cross terms in the general cost need to be evaluated. First, if the continuous-time cost matrix $\mathbf{W}_{xu}(t)$ is non-zero, then non-zero $\mathbf{S}(t_i)$ will also exist. Second, due to the desire to apply control to the state over the entire sample period (not just at the sample times), $\mathbf{S}(t_i)$ will still exist even if $\mathbf{W}_{xu}(t) \equiv \mathbf{0}$: see the first of two terms in the integral of Equation (2.34).

If the assumption of a linear system driven by white Gaussian noise is used, with a quadratic cost function given by Equation (2.31), then the optimal discrete linear feedback control law was shown to be of the form given in Equation (1.8), where the gain is given by [17:73]:

$$\mathbf{G}_c^*(t_i) = \left[\mathbf{U}(t_i) + \mathbf{B}_d^T(t_i)\mathbf{K}_c(t_{i+1})\mathbf{B}_d(t_i)\right]^{-1} \left[\mathbf{B}_d^T(t_i)\mathbf{K}_c(t_{i+1})\Phi(t_{i+1},t_i) + \mathbf{S}^T(t_i)\right] \tag{2.36}$$

which is also part of the solution of the backward Riccati equation solving for $\mathbf{K}_c(t_i)$ [17:73]:

$$\begin{aligned} \mathbf{K}_c(t_i) &= \mathbf{X}(t_i) + \Phi^T(t_{i+1},t_i)\mathbf{K}_c(t_{i+1})\Phi(t_{i+1},t_i) \\ &\quad - \left[\mathbf{B}_d^T(t_i)\mathbf{K}_c(t_{i+1})\Phi(t_{i+1},t_i) + \mathbf{S}^T(t_i)\right]^T \mathbf{G}_c^*(t_i) \end{aligned} \tag{2.37}$$

as solved backward from the terminal condition:

$$\mathbf{K}_c(t_{N+1}) = \mathbf{X}_f \tag{2.38}$$

If the assumption of a *time-invariant* linear system driven by white Gaussian noise is used, with a quadratic cost function given by Equation (2.31) with constant weighting matrices, then the optimal *constant-gain* discrete linear feedback control law was shown to be of the form given in Equation (1.9), where the gain is given by [17:242]:

$$\overline{\mathbf{G}}_c^* = \left[\mathbf{U} + \mathbf{B}_d^T \overline{\mathbf{K}}_c \mathbf{B}_d\right]^{-1} \left[\mathbf{B}_d^T \overline{\mathbf{K}}_c \mathbf{\Phi} + \mathbf{S}^T\right] \tag{2.39}$$

where $\overline{\mathbf{K}}_c$ is the solution to the steady state Ricatti equation:

$$\overline{\mathbf{K}}_c = \mathbf{X} + \mathbf{\Phi}^T \overline{\mathbf{K}}_c \mathbf{\Phi} - \left[\mathbf{B}_d^T \overline{\mathbf{K}}_c \mathbf{\Phi} + \mathbf{S}^T\right]^T \left[\mathbf{U} + \mathbf{B}_d^T \overline{\mathbf{K}}_c \mathbf{B}_d\right]^{-1} \left[\mathbf{B}_d^T \overline{\mathbf{K}}_c \mathbf{\Phi} + \mathbf{S}^T\right] \tag{2.40}$$

By assuming the use of constant weighting matrices and assuming steady-state operation, the quadratic cost of Equation (2.31) becomes:

$$J = E\left\{\sum_{i=0}^{\infty} \frac{1}{2}\left[\underline{\mathbf{x}}(t_i)^T \mathbf{X} \underline{\mathbf{x}}(t_i) + \mathbf{u}^T(t_i)\mathbf{U}\mathbf{u}(t_i) + 2\underline{\mathbf{x}}^T(t_i)\mathbf{S}\underline{\mathbf{u}}(t_i)\right]\right\} \tag{2.41}$$

As stated previously, the selection of the weighting matrices found in Equation (2.41) is important in determining the performance of the controller. Tuning of the control and state weighting matrices "is usually required, in analogy to tuning of $\mathbf{Q}_d$ and $\mathbf{R}$..." in the Kalman filter [17:10-11]. This research will investigate the performance enhancement by determining the state weighting matrices, $\mathbf{X}$, for all the discrete parameter points, $\mathbf{a}_k$. As stated previously, the cross weighting, $\mathbf{S}$, will be kept at zero based on previous research which indicated a minimal relative magnitude [11].

2.6 Summary

This chapter highlighted the concepts which are instrumental in this research. First, it discussed general Kalman filter algorithms as well as practical insights which will specifically affect this research. Next, Bayesian MMAE development was discussed as the basis for describing the moving-bank MMAE concept, which includes logic for moving and expanding/contracting the bank of filters/controllers. Finally, LQG controller development was presented. Most important to the discussions of the results in the chapters that follow

is the rational and appropriateness for the simplifying assumptions which will be adopted in this research. Basically, the system under investigation can be adequately represented by time invariant noises and adequate performance is obtained by using steady-state Kalman filters, steady-state optimal LQG controller gains, and constant state and control weighting matrices. Chapter 3 provides the development of the large space structure which is under investigation.

III. Rotating Two-Bay Truss Model Development

3.1 Introduction

The system model discussion provided in Chapter 1 was intended to provide a basic understanding of the system under investigation. The purpose of the current development (like that found in Chapter 2) is twofold. First, the system model description of Section 1.1.4 will be enriched to address the general theoretical development relevant to this research. Second, discussion contained here will highlight simplifying assumptions and insights gained from previous research which are pertinent to the accomplishment of the objectives specific to this research. Rotating two-bay truss model development will encompass (1) a description of the two-bay truss, (2) system model development, (3) reduced-order model development, and (4) truth model/filter model design. For a complete development, see Karnick [8] and Lynch [14].

3.2 Two-Bay Truss Description

The two-bay truss has been discussed in general terms to this point. This section (1) gives a physical description of the model used , (2) describes the sensors and their locations, and (3) relates the physical system parameter uncertainty to the MMAE development discussed previously.

3.2.1 Physical Construction. In 1986, Karnick [8:45] provided the space structure model that has been used since that time and is shown in Figure 1.8. For his research, Karnick obtained a fixed two-bay truss which was originally developed to study the effects of structural optimization on optimal control design [27]. A similar model was used to research active control laws for vibration damping [14]. In order to study the effects of rigid body motion, Karnick modified the fixed two-bay truss by adding a hub of relatively large mass. Rigid body motion is established by holding the hub center fixed while the truss is free to rotate about this point in the x-y plane. Thus, the rotating two-bay truss approximates a space structure that has a hub with appendages extending from the structure, where the hub can be rotated to point the appendage in the commanded direction.

Table 3.1. Structural Member's Cross-Sectional Areas

Member	Area (in^2)	Member	Area (in^2)
a	0.00321	h	0.00328
b	0.00100	i	0.00439
c	0.00321	j	0.00439
d	0.01049	k	0.20000
e	0.00100	l	0.20000
f	0.01049	m	0.20000
g	0.00328		

The thirteen rods that make up the truss structure are listed in Table 3.1 with their cross-sectional areas. The members are made of aluminum which has a modulus of elasticity of 10^7 psi and weight density of 0.1 pounds per cubic inch. The rods connecting the truss to the hub are much larger in diameter than the truss members, causing this link to be very stiff compared to the truss rigidity. The addition of this physical link does introduce high frequency modes into the system but maintains the low frequency modes of the original fixed structure [11:57].

Non-structural masses with a mass of 1.294 $lb \cdot sec^2/in$ are located at nodes 1 through 4 as indicated in Figure 1.8. This mass is very large compared to the member masses, which achieves the low frequencies associated with large space structures [14:14]. The actual mass values were selected based on an optimizing technique that maintains the lowest undamped natural frequency (associated with the lowest mode) of 0.5 Hz [14]. The mass and stiffness matrices, which describe the system model, were obtained using finite element analysis [28]. The mass and stiffness matrices for the nominal structure are listed in Appendix A.

3.2.2 Sensors and Actuators. A combination of gyros and accelerometers is used to provide motion information to the control system. Accelerometers are located at nodes 1 and 2 as shown in Figure 1.8. These sensors measure physical displacements in the y-axis direction. The accelerometers are not located at the node of the bending modes being detected since the "displacement of the truss caused by the bending modes can not be detected" [11:59] under those conditions. The outputs of the accelerometers are integrated once to obtain velocity data and once again to obtain position data. This would argue

for a non-diagonal $\mathbf{R}_t$ matrix. However, for this research, the simplifying assumption will be made that the $\mathbf{R}_t$ matrix can be adequately modelled with a diagonal matrix. Gyros are placed at the hub (node 7) to provide angular displacement and angular velocity information. Actuators are co-located with the accelerometers at nodes 1 and 2. The actuators act along the y-axis only. The co-location is done to simplify the system model, specifically the measurement matrix $\mathbf{H}$. The actuators at nodes 1 and 2 are assumed to be thrusters. The hub also contains an inertia wheel to act as an additional actuator. Dynamics noise, $\underline{\mathbf{w}}$, is assumed to enter the system at the actuators.

3.2.3 Physical System Parameter Uncertainty. The purpose of this thesis is to test the moving-bank MMAE and control algorithms. Therefore, the model must have parameter uncertainty which requires adaptive estimation to be applied. As discussed previously, the parameter space investigated in this research is composed of 100 discrete combinations of the parameters of interest, where the parameters of interest are the non-structural mass and the stiffness matrix. The non-structural mass changes are representative of the depletion of fuel tanks on the structure, refueling of the fuel tanks (where weight is being added), or weight being shifted from one part of the truss to another. The variations in stiffness of the truss members might be caused by structural fatigue in a rod or rods or, in the more extreme case, actual failure of one of the members. Not only does this choice of parameter variations make sense physically, but as will be seen in the next section, the choice of mass and stiffness as parameters is motivated mathematically due to their explicit appearance in describing the dynamics equation of the structure.

3.3 System Model Development

A mathematical model is required to determine the feasibility of the MMAE estimation and control techniques for the large space structure described in Section 3.2. This section will (1) discuss the development of the general system model in state space form and (2) motivate the need for and describe the development of modal decomposition.

3.3.1 Physical Coordinate Form. "The standard second order matrix differential equation, developed through finite element methods, which governs the flexural vibrations

of a structure" is given by [14:3], [8:39] :

$$\mathbf{M}\ddot{\mathbf{r}}(t) + \mathbf{C}\dot{\mathbf{r}}(t) + \mathbf{K}\mathbf{r}(t) = \mathbf{F}(t) = \mathbf{F}_1(\mathbf{u}, t) + \mathbf{F}_2(t) \tag{3.1}$$

where

- $\mathbf{r}(t)$ = n-dimensional vector representing the structure's physical position
- $\mathbf{F}_1(\mathbf{u}, t)$ = r-dimensional deterministic control inputs
- $\mathbf{F}_2(t)$ = r-dimensional disturbances and unmodeled control inputs
- $\mathbf{M} = n - by - n$ constant mass matrix
- $\mathbf{C} = n - by - n$ constant damping matrix
- $\mathbf{K} = n - by - n$ constant stiffness matrix

If the external disturbances are assumed to be white Gaussian noises, then Equation (3.1) becomes [14:4], [8:40]:

$$\mathbf{M}\ddot{\underline{\mathbf{r}}}(t) + \mathbf{C}\dot{\underline{\mathbf{r}}}(t) + \mathbf{K}\underline{\mathbf{r}}(t) = -\mathbf{b}\mathbf{u}(t) - \mathbf{g}\underline{\mathbf{w}}(t) \tag{3.2}$$

where [11:48]

- $\mathbf{u}(t)$ = r-dimensional vector actuator inputs
- $\mathbf{b} = n - by - r$ control input matrix identifying position and relationships between actuators and controlled variables
- $\underline{\mathbf{w}}(t)$ = s-dimensional vector representing the dynamics driving noise, where s is the number of noise inputs
- $\mathbf{g} = n - by - s$ noise input matrix identifying position and relationships between the dynamics driving noise and controlled variables

If the states of the system are taken to be composed of the position and velocity variables, then Equation (3.2) may be written in state space form as [14:4], [8:40]:

$$\dot{\underline{\mathbf{x}}}(t) = \mathbf{F}\underline{\mathbf{x}}(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{G}\underline{\mathbf{w}}(t) \tag{3.3}$$

where the states are stochastic processes since they are driven by noise as well as a deterministic input. The state vector is given by [14:4], [8:41]:

$$\underline{\mathbf{x}}(t) = \begin{bmatrix} \underline{\mathbf{r}}(t) \\ \dot{\underline{\mathbf{r}}}(t) \end{bmatrix}_{2n\times 1} \tag{3.4}$$

The state vector described in Equations (3.3) and (3.4) are position and velocity contributions of the various rigid body and bending modes to physical position and velocity variables at selected points on the structure which are measurable by accelerometers. The uncertain parameters enter the mathematical model via the constant matrices in Equation (3.3). Relating this to the discussion of Section 2.3, Equation (3.3) represents the same system as described by Equation (2.1). The constant system matrices are given by [8:41]:

$$\mathbf{F} = \begin{bmatrix} \mathbf{0}_{n\times n} & \mathbf{I}_{n\times n} \\ -\mathbf{M}^{-1}\mathbf{K}_{n\times n} & -\mathbf{M}^{-1}\mathbf{C}_{n\times n} \end{bmatrix}_{2n\times 2n} \tag{3.5}$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{0}_{n\times r} \\ -\mathbf{M}^{-1}\mathbf{b}_{n\times r} \end{bmatrix}_{2n\times r} \tag{3.6}$$

$$\mathbf{G} = \begin{bmatrix} \mathbf{0}_{n\times s} \\ -\mathbf{M}^{-1}\mathbf{g}_{n\times s} \end{bmatrix}_{2n\times s} \tag{3.7}$$

One of the assumptions taken in this thesis is that the system dynamics driving noise enters the system at the same location as the actuators (thrusters). This assumption causes the $\mathbf{g}$ and $\mathbf{b}$ matrices to be equal, and therefore $\mathbf{G} \equiv \mathbf{B}$.

The model of the discrete-time measurement of the states is given by [8:42]:

$$\underline{\mathbf{z}}(t_i) = \left\{ \begin{bmatrix} \mathbf{H}_\mathrm{p} & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_\mathrm{v} \end{bmatrix}_{m\times 2n} \underline{\mathbf{x}}(t_i) \right\} + \underline{\mathbf{v}}(t_i) \tag{3.8}$$

where:

- m = number of measurements
- $\underline{\mathbf{v}}(t_i)$ = m-dimensional uncertain measurement disturbance modeled as a zero-mean white Gaussian noise of covariance $\mathbf{R}(t_i)$
- $\mathbf{H_p} = (m/2) - by - n$ position measurement matrix in physical coordinates
- $\mathbf{H_v} = (m/2) - by - n$ velocity measurement matrix in physical coordinates

Relating this to the discussion of Section 2.2, Equation (3.8) represents the same measurement model as described by Equation (2.7). For this research, since the position and velocity measurements are co-located, $\mathbf{H_p} \equiv \mathbf{H_v}$.

The equations developed here are highly coupled, that is, the equations are not independent. This situation is not desirable, therefore modal decomposition is used to transform the sets of equations into independent modal equations.

3.3.2 Modal Coordinate Form. The equations just developed are transformed into a system of equations that are decoupled. "The general response of a complicated system can be broken down into the sum of n simple responses" [2:260]. The assumption that the matrices are constant allows the modal decomposition to be useful. If the matrices were variable, then the advantage of using modal decomposition would be lost [2:262].

Following the research performed by Lynch and Banda, the damping matrix is assumed to be a linear combination of the mass and stiffness matrix [14:4]:

$$\mathbf{C} = \alpha\mathbf{M} + \beta\mathbf{K} \tag{3.9}$$

Due to the transformation from physical to modal coordinates, the actual determination of α and β are not required. If the modal coordinates are denoted by $\tilde{\mathbf{r}}$, the relationship between the modal and physical coordinates is given by [14:5]:

$$\mathbf{r} = \mathcal{T}\tilde{\mathbf{r}} \tag{3.10}$$

where $\mathcal{T}$ is the $n-by-n$ transformation matrix composed of eigenvectors determined from the solution of [14:5]:

$$\omega^2 \mathbf{M}\mathcal{T} = \mathbf{K}\mathcal{T} \tag{3.11}$$

Note that Equation (3.11) relates the modal frequency, ω, the mass matrix, and the stiffness matrix. The damping matrix is not involved in the solution of the eigenvectors, therefore, the solution of α and β in Equation (3.9) are not explicitly found since they are not required for the modal decomposition. The values of ω that satisfy Equation (3.11) are called the natural or modal frequencies.

Substituting Equation (3.10) into Equation (3.3) yields the transformed state space equation to [14:5]:

$$\dot{\underline{\tilde{\mathbf{x}}}}(t) = \tilde{\mathbf{F}}\underline{\tilde{\mathbf{x}}}(t) + \tilde{\mathbf{B}}\mathbf{u}(t) + \tilde{\mathbf{G}}\underline{\mathbf{w}}(t) \tag{3.12}$$

where ($\tilde{}$) represents the modal coordinate frame and the state vector is now defined as [14:5]:

$$\underline{\tilde{\mathbf{x}}}(t) = \begin{bmatrix} \underline{\tilde{\mathbf{r}}}(t) \\ \dot{\underline{\tilde{\mathbf{r}}}}(t) \end{bmatrix}_{2n\times 1} \tag{3.13}$$

The matrices describing Equation (3.3) are also transformed and calculated as [14:5]:

$$\tilde{\mathbf{F}} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathcal{T}^{-1}\mathbf{M}^{-1}\mathbf{K}\mathcal{T} & -\mathcal{T}^{-1}\mathbf{M}^{-1}\mathbf{C}\mathcal{T} \end{bmatrix}_{2n\times 2n} \tag{3.14}$$

$$\tilde{\mathbf{B}} = \begin{bmatrix} \mathbf{0} \\ -\mathcal{T}^{-1}\mathbf{M}^{-1}\mathbf{b} \end{bmatrix}_{2n\times r} = \tilde{\mathbf{G}} \tag{3.15}$$

Orthogonality of the modal vectors and the following definition

$$\left[-\mathcal{T}^{-1}\mathbf{M}^{-1}\mathbf{C}\mathcal{T}\right] = -2\zeta_i\omega_i \tag{3.16}$$

allows the open loop plant matrix to be written in terms of the undamped natural frequencies and the damping ratio of the i-th mode. Therefore, $\tilde{\mathbf{F}}$ becomes:

$$\tilde{\mathbf{F}} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ [-\omega_i^2] & [-2\zeta_i\omega_i] \end{bmatrix}_{2n\times 2n} \tag{3.17}$$

where each of the four entries in $\tilde{\mathbf{F}}$ are $n \times n$ and are diagonal. Finally, the equation describing the measurement process may be rewritten as [8:44]:

$$\underline{\mathbf{z}}(t_i) = \begin{bmatrix} \mathbf{H}_\mathbf{p}\mathcal{T} & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_\mathbf{v}\mathcal{T} \end{bmatrix}_{m\times 2n} \tilde{\mathbf{x}}(t_i) + \underline{\mathbf{v}}(t_i) \tag{3.18}$$

The measurement matrix in Equation (3.18) shows the general form and will be referred to as $\tilde{\mathbf{H}}$. The actual implementation of $\tilde{\mathbf{H}}$ is different from that shown in Equation (3.18) due to the method used to incorporate rigid body position and velocity measurements into the matrix; it corresponds to a reordering of the components of $\underline{\mathbf{z}}(t_i)$. The form of $\tilde{\mathbf{H}}$ used for this research is provided in Appendix A.

The use of modal coordinates allows the following assumptions to be made [14:14]. The structural damping is assumed to be uniform throughout the structure. The damping coefficient selection does not play a role in the undamped natural frequencies and therefore may be selected based upon design requirements. The previous theses used a value of $\zeta = 0.005$ based on work performed by Lynch and Banda [14] and as representative of many space structures.

The mathematical structure of the system model has been developed in both physical and modal coordinates. The parameters used in the equations of this section were derived from finite element analysis of the structure. However, large order systems present a problem computationally. As a result, low-order models are desired. The next section provides a detailed explanation of the method used in this research to arrive at a low-order design model.

3.4 Reduced-Order Model Development

The state reduction performed on the system of equations developed follows the work of Kokotovic, et.al. [10:123-124]. The need to reduce the state dimension is due to the large computational load that a 24-state system would place on the MMAE algorithm. The reduced order model is developed in this section from the basis of the system model developed in the previous section.

3.4.1 Order Reduction Development. The continuous, linear, stochastic system model given in Equation (3.3) is reformulated as [10:123], [8:52]:

$$\dot{\underline{x}}(t) = \begin{bmatrix} \dot{\underline{x}}_1(t) \\ \dot{\underline{x}}_2(t) \end{bmatrix} = \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix} \begin{bmatrix} \underline{x}_1(t) \\ \underline{x}_2(t) \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} u(t) + \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} \underline{w}(t) \quad (3.19)$$

where the system is driven by deterministic controls and zero-mean, white Gaussian noise of strength $\mathbf{Q}(t)$. The state vector partition $\underline{x}_1(t)$ corresponds to "slower" modes to be maintained in the design model and $\underline{x}_2(t)$ corresponds to "faster" modes to be ignored. The sampled-data measurement model provided in Equation (3.8) is reformulated as:

$$\underline{z}(t_i) = \begin{bmatrix} H_1 & H_2 \end{bmatrix} \begin{bmatrix} \underline{x}_1(t_i) \\ \underline{x}_2(t_i) \end{bmatrix} + \underline{v}(t_i) \quad (3.20)$$

where the measurement noise is a zero-mean, discrete-time, white Gaussian process with covariance $\mathbf{R}(t_i)$.

If steady state is assumed to be reached instantaneously by the "fast" modes, the $\underline{x}_2(t)$ modes are removed while maintaining the low frequency modes in $\underline{x}_1(t)$. F_{11} and F_{22} are square matrices and $\underline{x}_2(t)$ can be expressed in terms of $\underline{x}_1(t)$ assuming that F_{22}^{-1} exists. Then the higher order modes may be expressed by [10:123], [8:52]:

$$\dot{\underline{x}}_2(t) = F_{21}\underline{x}_1(t) + F_{22}\underline{x}_2(t) + B_2 u(t) + G_2\underline{w}(t) = 0 \quad (3.21)$$

$$\underline{x}_2(t) = -F_{22}^{-1}\left[F_{21}\underline{x}_1(t) + B_2 u(t) + G_2\underline{w}(t)\right] \quad (3.22)$$

Substituting these equations into Equations (3.19) and (3.20) yields [10:123], [8:52] :

$$\begin{aligned}\dot{\underline{x}}_1(t) &= \left[\mathbf{F}_{11} - \mathbf{F}_{12}\mathbf{F}_{22}^{-1}\mathbf{F}_{21}\right]\underline{x}_1(t) + \left[\mathbf{B}_1 - \mathbf{F}_{12}\mathbf{F}_{22}^{-1}\mathbf{B}_2\right]\mathrm{u}(t) \\ &\quad + \left[\mathbf{G}_1 - \mathbf{F}_{12}\mathbf{F}_{22}^{-1}\mathbf{G}_2\right]\underline{\mathrm{w}}(t) \end{aligned} \tag{3.23}$$

$$\begin{aligned}\underline{z}(t_i) &= \left[\mathbf{H}_1 - \mathbf{H}_2\mathbf{F}_{22}^{-1}\mathbf{F}_{21}\right]\underline{x}_1(t_i) - \mathbf{H}_2\mathbf{F}_{22}^{-1}\left[\mathbf{B}_2\mathrm{u}(t_i) + \mathbf{G}_2\underline{\mathrm{w}}(t_i)\right] \\ &\quad + \underline{\mathrm{v}}(t_i) \end{aligned} \tag{3.24}$$

The second term in Equation (3.24) is a direct feedforward term created by the order reduction [10:123], [8:52].

Applying the above order reduction technique to the original system model in modal coordinates (provided in Equations (3.12) through (3.18)) yields a new system dynamics matrix given by [10:123], [8:53] :

$$\tilde{\mathbf{F}} = \begin{bmatrix} 0 & \mathrm{I} & \vdots & 0 & 0 \\ [-\omega_1^2] & [-2\zeta_1\omega_1] & \vdots & 0 & 0 \\ \cdots\cdots & \cdots\cdots & \cdots\cdots & \cdots\cdots & \cdots\cdots \\ 0 & 0 & \vdots & 0 & \mathrm{I} \\ 0 & 0 & \vdots & [-\omega_2^2] & [-2\zeta_2\omega_2] \end{bmatrix} \tag{3.25}$$

The matrix given by Equation (3.25) has obvious partitions. The upper left partition contains the low frequency modes (denoted with subscript 1) to be retained while the lower right partition (denoted with subscript 2) contains the high frequency modes assumed to reach steady state instantaneously. The partitions correspond to the $\mathbf{F}_{11}$ and $\mathbf{F}_{22}$ partitions observed in Equation (3.19). Each of the subpartitions in Equation (3.25) are diagonal matrices, but the entries in any one subpartition are all different. Additionally, the submatrices $\tilde{\mathbf{F}}_{12}$ and $\tilde{\mathbf{F}}_{21}$ are zero. Substituting this information into Equations (3.23) and (3.24) yields [10:123], [8:53]:

$$\dot{\tilde{\underline{x}}}_1(t) = \tilde{\mathbf{F}}_{11}\tilde{\underline{x}}_1(t) + \tilde{\mathbf{B}}_1\mathrm{u}(t) + \tilde{\mathbf{G}}_1\underline{\mathrm{w}}_r(t) = \tilde{\mathbf{F}}_r\tilde{\underline{x}}_1(t) + \tilde{\mathbf{B}}_r\mathrm{u}(t) + \tilde{\mathbf{G}}_r\underline{\mathrm{w}}_r(t) \tag{3.26}$$

$$\begin{aligned}\underline{z}(t_i) &= \tilde{\mathbf{H}}_1\underline{\tilde{x}}_1(t_i) - \tilde{\mathbf{H}}_2\tilde{\mathbf{F}}_{22}^{-1}\tilde{\mathbf{B}}_2\mathbf{u}(t_i) - \tilde{\mathbf{H}}_2\tilde{\mathbf{F}}_{22}^{-1}\tilde{\mathbf{G}}_2\underline{\mathbf{w}}(t_i) + \underline{\mathbf{v}}_r(t_i) \\ &= \tilde{\mathbf{H}}_r\underline{\tilde{x}}_1(t_i) + \tilde{\mathbf{D}}_u\mathbf{u}(t_i) + \tilde{\mathbf{D}}_w\underline{\mathbf{w}}(t_i) + \underline{\mathbf{v}}_r(t_i) \qquad (3.27)\end{aligned}$$

where the subscript r denotes "reduced-order." The only terms associated with the high frequency modes are the direct feedthrough terms in Equation (3.27), $\tilde{\mathbf{D}}_u$ and $\tilde{\mathbf{D}}_w$. These direct feedthrough terms allow direct measurement of the control inputs $\mathbf{u}(t_i)$ as well as the system dynamics driving noise $\underline{\mathbf{w}}(t)$. The noise vectors, $\underline{\mathbf{w}}_r(t)$ in Equation (3.26) and $\underline{\mathbf{v}}_r(t_i)$ in Equation (3.27), represent noise vectors which account for the truncation of the high frequency modes from the system model.

The direct feedthrough terms, $\tilde{\mathbf{D}}_u$ and $\tilde{\mathbf{D}}_w$ in Equation (3.27) are the most complicated since the other terms are readily available. (The other matrices are found by truncating the states associated with $\underline{\mathbf{x}}_2(i)$ from the full state model.) As previously discussed in Section 3.3.1, the $\mathbf{G}$ and $\mathbf{B}$ matrices are equal and therefore, $\tilde{\mathbf{D}}_u = \tilde{\mathbf{D}}_w = \tilde{\mathbf{D}}$. The terms seen in $\tilde{\mathbf{D}}$ show that it is dependent on the state terms that are assumed to reach steady state immediately. The development of $\tilde{\mathbf{D}}$ is provided in Appendix A. Research performed by Karnick and Lashlee did not incorporate the $\tilde{\mathbf{D}}$ terms since the truth model and the filter models they used were of the same dimension. Van Der Werken and Schore both provided development of such matrices for their unmodelled effects investigations, however it is doubtful that their simulations *actually* incorporated these terms. This will be discussed in more detail in Chapter 4.

3.4.2 Order Reduction Selection. For each location in the discretized parameter space, eigenvalues and eigenvectors of the unreduced system may be calculated from the system dynamics matrix $\tilde{\mathbf{F}}$ given in Equation (3.17). Table 3.2 shows the natural frequencies and damping factors associated with the nominal mass and stiffness matrices, i.e., no scaling. The damping factors are all close to the 0.005 value used as an approximation in previous theses. The eigenvalues fall into closely spaced groups. For example, one group of three modes is seen in modes 4 through 6. Research performed by Van Der Werken [26] and Schore [22] were the only previous works to use a 24-state truth model; both Van Der Werken and Schore elected to use the first three modes in the filter model corresponding

Table 3.2. Natural Frequencies and Damping Factors for Nominal Structure

Mode	Eigenvalue	Natural Frequency (Hz)	Damping Factor
1	0.2934 $\pm$ j 0.0000	0.0000	0.000
2	-0.0449 $\pm$ j 8.8982	1.4152	5.0459E-3
3	-0.1127 $\pm$ j 22.5490	3.5888	4.9980E-3
4	-0.1477 $\pm$ j 29.5444	4.7021	4.9993E-3
5	-0.1558 $\pm$ j 31.1511	4.9579	5.0014E-3
6	-0.1640 $\pm$ j 32.7999	5.2203	5.0002E-3
7	-0.2719 $\pm$ j 54.3886	8.6562	4.9992E-3
8	-0.2908 $\pm$ j 58.1586	9.2562	5.0001E-3
9	-4.9296 $\pm$ j 985.904	156.91	5.0000E-3
10	-45.0945 $\pm$ j 9018.79	1435.4	5.0000E-3
11	-57.5800 $\pm$ j 11515.9	1832.8	5.0000E-3
12	-99.7825 $\pm$ j 19956.3	3176.2	5.0000E-3

to a 6-state filter model as done previously by Karnick [8] and Lashlee [11]. As stated previously, this research will continue with the same truth model and filter model as that used by Van Der Werken and Schore. The next section discusses the specific development of the truth model and filter model which will be used.

3.5 Truth Model/Filter Model Design

The previous sections provided a general background for developing a system model for the two-bay truss as well as a method for selecting a suitable reduced order model for basing the Kalman filter and stochastic controller. This section provides specific details for the truth model and Kalman filter model used in this research. This section discusses (1) the form of the truth model and filter state vectors, and (2) determination of the appropriate sampling frequency for the discrete formulation.

Equations (3.14), (3.15), and (3.18) show the form of the matrices needed to describe the system and measurement models. These matrices are functions of the true parameter vector $\mathbf{a}_t$, a two-dimensional vector with scalar components that dictate the value of M and K, respectively, where M and K are the mass and stiffness matrices described earlier. For this research, the truth model vector is assumed to be composed of the rigid body and first eleven bending mode position states, followed by the corresponding twelve velocity

states:

$$\underline{\tilde{\mathbf{x}}}_t = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{12} \\ x_{13} \\ x_{14} \\ \vdots \\ x_{24} \end{bmatrix} = \begin{bmatrix} \text{Rigid body mode position} \\ \text{First bending mode position} \\ \vdots \\ \text{Eleventh bending mode position} \\ \text{Rigid body mode velocity} \\ \text{First bending mode velocity} \\ \vdots \\ \text{Eleventh bending mode velocity} \end{bmatrix} \tag{3.28}$$

Matrices developed to describe the truth model and measurement model corresponding to the $\underline{\tilde{\mathbf{x}}}_t$ vector will be denoted by subscript t.

The filter-model states used in this research and previous research efforts are defined by:

$$\underline{\tilde{\mathbf{x}}}_f = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} = \begin{bmatrix} \text{Rigid body mode position} \\ \text{First bending mode position} \\ \text{Second bending mode position} \\ \text{Rigid body mode velocity} \\ \text{First bending mode velocity} \\ \text{Second bending mode velocity} \end{bmatrix} \tag{3.29}$$

Matrices developed to describe the filter model and measurement model corresponding to the $\underline{\tilde{\mathbf{x}}}_f$ vector will be denoted by subscript f.

In order to evaluate the error between the 24-state truth model and the six-state filter model, Van Der Werken transformed the the truth model so that the first six states correspond to the states of the filter [26:233]. However, for this research, his transformation is not needed. Formation of the error vector for this research will be discussed in Chapter 4. Appendix A provides a listing of the truth model and filter model matrices used in this research.

As discussed in Section 2.2, the algorithms which will be implemented need to be in discrete form. Therefore, selection of a sampling period is required for evaluation of the truth model matrices Φ_t, B_{d_t}, and Q_{d_t} and the filter model matrices Φ_f, B_{d_f}, and Q_{d_f} as developed in Equations (2.10) and (2.15). Based upon the frequencies shown for the states to be retained in the filter model, a sampling period of 0.05 seconds (20 Hz sampling rate) is appropriate.

3.6 Summary

This chapter provided the framework for developing a practical mathematical model for basing a "truth" model and a Kalman filter design model for investigating the multiple-model adaptive estimation and control algorithms. Use of modal decomposition is instrumental in establishing the mathematical model. For this method to be useful, this research assumes that the matrices describing the second order differential equation of the structure are constant. Using the method of singular perturbations, appropriate selection of reduced-order states can be accomplished for basing the Kalman filter model. The development and form of the 24-state truth model and six-state Kalman filter matrices are provided in Appendix A. Chapter 4 discusses the simulations which will be performed in this research.

IV. Simulation

4.1 Introduction

The space structure discussed in Chapter 3 is simulated in order to study the estimation and control capabilities of the MMAE and controller techniques discussed in Chapter 2. To determine the statistical properties of the estimation/control process, Monte Carlo analysis is performed. This chapter provides (1) a discussion of the Monte Carlo analysis, (2) a description of the software, (3) a discussion of the modifications to existing software, and (4) the simulation plan for this research.

4.2 Monte Carlo Analysis

The Monte Carlo analysis performed by the simulation software provides the statistical information about the performance of the MMAE and the controller algorithms described in the previous chapters. If the system under investigation were fully linear, then covariance analysis could be used [15:329]. However, the adaptive nature of the MMAE technique used to control the large space structure requires the employment of Monte Carlo analysis to obtain many samples of the process and evaluate the statistics of the process. As with the work performed by Van Der Werken [26] and Schore [22], the simulation involves two models: a 24-state "truth model" for accurate simulation and performance evaluation and a six-state filter and controller design model based on the development of Chapter 3. The design model is the same one used by Karnick [8], Lashlee [11], Van Der Werken [26], and Schore [22]. The 24-state truth model was developed by Van Der Werken [26] and used by Schore [22]. This section will discuss (1) the error vector formulation and (2) the error vector statistics.

4.2.1 Error Vector Formulation. While previous thesis efforts [8, 11, 26] viewed the estimation and control of the modal states as the primary objective, the goal of this research is to view physically meaningful quantities. The first simulation is concerned with the estimation of the structure positions and velocities at nodes 1, 2, and 7 (i.e., at the midpoint and end of the two bay truss, and at the hub). A general vector of error

signals (which consists of three physical positions and three velocities) is determined by subtracting the filter estimates of these quantities from the true positions and velocities. The error term formulation used in this research is modified from that presented by Schore [22:Chapter 1 pg 22]. The error term is computed as:

$$\underline{e}_x(t_i) = \tilde{\mathbf{H}}'_t \underline{\tilde{\mathbf{x}}}_t(t_i) - \sum_{j=1}^{9} \tilde{\mathbf{H}}_{f_j} \underline{\tilde{\hat{\mathbf{x}}}}_{f_j}(t_i) \cdot p_j(t_i) \tag{4.1}$$

where $\tilde{\mathbf{H}}'_t$ is a modified version of the general truth model measurement matrix, in modal coordinates, and relates the three physical positions and velocities to the truth model states (assumed to be 12 modal positions and velocities, in that order, described by Equation (3.28)). The form of $\tilde{\mathbf{H}}'_t$ is given by [22:Chapter 4 pg 8]:

$$\tilde{\mathbf{H}}'_t = \begin{bmatrix} \tilde{\mathbf{H}}_{f1_{3\times3}} & \lambda\tilde{\mathbf{H}}_{tp_{3\times9}} & 0 & 0 \\ 0 & 0 & \tilde{\mathbf{H}}_{f2_{3\times3}} & \lambda\tilde{\mathbf{H}}_{tv_{3\times9}} \end{bmatrix}_{6\times24} \tag{4.2}$$

where $\tilde{\mathbf{H}}_{tp}$ corresponds to the effect of the upper nine modal position states, and $\tilde{\mathbf{H}}_{tv}$ to the effect of the upper nine modal velocity states. Schore implemented a modified version of this error form to investigate the effects of the higher order bending modes of the structure on the ability of the estimation technique to provide accurate position and velocity estimates. The effects of the $\tilde{\mathbf{H}}_{tp}$ and $\tilde{\mathbf{H}}_{tv}$ partitions of the $\tilde{\mathbf{H}}_t$ matrix can be varied by the scalar variable, λ, between 0 and 1. ($\lambda = 0$ allows no higher order modes whereas $\lambda = 1$ allows the full effect of the higher order modes.) For this research, the full effects of the higher order modes will be investigated, therefore requiring λ equal to one in Equation (4.2) unless otherwise specified.

The filter measurement matrix, $\tilde{\mathbf{H}}_f$, relates the three physical positions and three velocities, in that order, to the filter assumed states (three modal positions and three velocities, in that order). The general form of $\tilde{\mathbf{H}}_f$ is given by:

$$\tilde{\mathbf{H}}_f = \begin{bmatrix} \tilde{\mathbf{H}}_{f1_{3\times3}} & 0 \\ 0 & \tilde{\mathbf{H}}_{f2_{3\times3}} \end{bmatrix}_{6\times6} \tag{4.3}$$

Figure 4.1(a) shows a block diagram of the Monte Carlo analysis simulation which is similar to that used by Schore [22:Chapter 4 pg 3]. To review, the variables of interest are [22]:

- $\underline{\tilde{\mathbf{x}}}_t(t_i)$: the truth model states; 24-dimensional and in modal coordinates, representing the rigid body mode and eleven bending modes
- $\underline{\tilde{\mathbf{x}}}_f(t_i)$: estimates of the system states; 6-dimensional and in modal coordinates, representing the rigid body and the first two bending modes
- $\mathbf{a}_t(t_i)$: the vector representing the true structure mass and stiffness parameters
- $\underline{\hat{\mathbf{a}}}(t_i)$: estimates of the uncertain parameter vector
- $\underline{\mathbf{e}}_a(t_i)$: the error in the parameter estimate defined as $\underline{\mathbf{e}}_a(t_i) = \mathbf{a}_t(t_i) - \underline{\hat{\mathbf{a}}}(t_i)$
- $\underline{\mathbf{e}}_x(t_i)$: the error in the system estimate defined by Equation (4.1). The form is due to the development of the $\tilde{\mathbf{H}}_t$ and $\tilde{\mathbf{H}}_f$ matrices and is given as follows:

$$\underline{\mathbf{e}}_x(t_i) = \begin{bmatrix} e_{x_1} \\ e_{x_2} \\ e_{x_3} \\ e_{x_4} \\ e_{x_5} \\ e_{x_6} \end{bmatrix} = \begin{bmatrix} \text{Node 1 position error} \\ \text{Node 1 velocity error} \\ \text{Node 2 position error} \\ \text{Node 2 velocity error} \\ \text{Node 7 position error} \\ \text{Node 7 velocity error} \end{bmatrix} \tag{4.4}$$

The second simulation performed implements a controller for closed-loop estimation and control. The control simulation is depicted in Figure 4.1(b). The regulation error signals are generated and evaluated statistically as for the estimation study. Here, the signals of most interest are the true structure positions and velocities at the three points of interest (described previously). Basically this is the quantity $\mathbf{H}_t\underline{\mathbf{x}}_t$. This is significant since the goal of the controller is to quell the oscillations of the structure that may be induced by moving the structure, changes in structural characteristics, or changes in non-structural masses on the structure. Also of interest is $\mathbf{u}$ since the amount of control available from

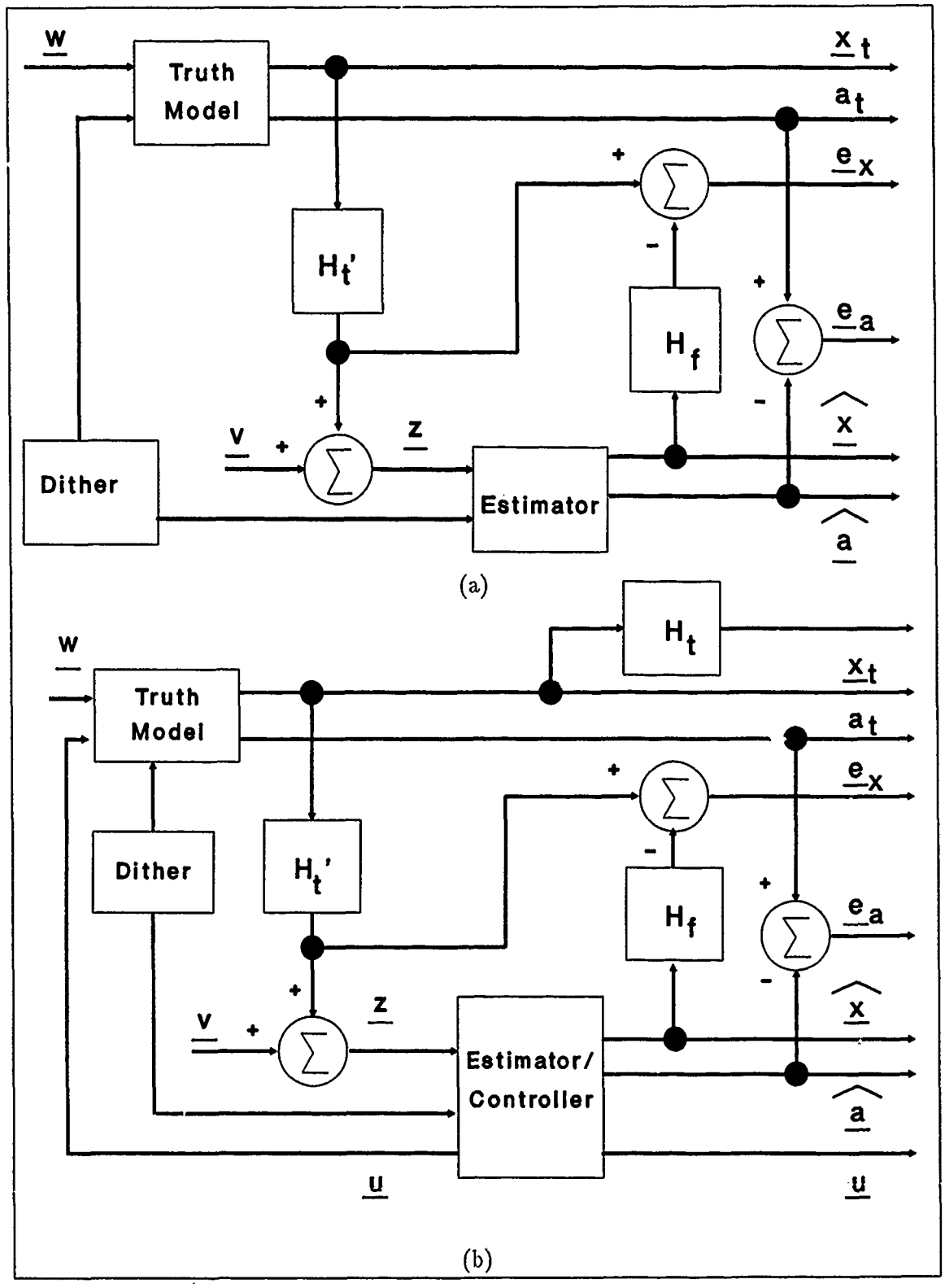

Figure 4.1. Diagram of (a) Estimator Simulation, and (b) Controller Simulation [22]

the actuators is realistically limited. The dither block shown in Figure 4.1(a) and (b) will be described in Section 4.4.1.

4.2.2 Error Vector Statistics. The statistics of primary concern for the estimation error (and control) processes may be defined as the sample mean and covariance of the processes. The mean is determined by [26:67], [11:74]:

$$\mathbf{E}\{\underline{\mathbf{e}}_x(t_i)\} \approx \mathbf{M}_{e_x}(t_i) = \frac{1}{L}\sum_{k=1}^{L} \underline{\mathbf{e}}_{x_k}(t_i) \tag{4.5}$$

where L is the number of Monte Carlo analysis runs made and $\mathbf{e}_{x_k}(t_i)$ is the value of the error signal during the k^{th} simulation run at run time t_i. The covariance may be calculated as [15:130]:

$$\begin{aligned} \mathbf{P}_{e_x}(t_i) &= \mathbf{E}\{[\underline{\mathbf{e}}_x(t_i) - \mathbf{E}\{\underline{\mathbf{e}}_x(t_i)\}][\underline{\mathbf{e}}_x(t_i) - \mathbf{E}\{\underline{\mathbf{e}}_x(t_i)\}]^T\} \\ &\approx \frac{1}{L-1}\sum_{k=1}^{L}\{\underline{\mathbf{e}}_{x_k}(t_i)\underline{\mathbf{e}}_{x_k}^T(t_i)\} - \frac{L}{L-1}\mathbf{M}_{e_x}(t_i)\mathbf{M}_{e_x}^T(t_i) \end{aligned} \tag{4.6}$$

The statistics of the parameter estimation errors, $\underline{\mathbf{e}}_a(t_i)$, or true structure positions, $\mathbf{H}_t\tilde{\underline{\mathbf{x}}}_t(t_i)$, may also be obtained similarly with appropriate substitutions into Equations (4.5) and (4.6). This research will use sample statistics computed on the basis of ten Monte Carlo runs to approximate the true statistics. Each run will be ten seconds in duration.

4.3 Software Description

4.3.1 Introduction. The software used in this research was started by Hentz [7] and then modified and used by Filios [3], Karnick [8], Lashlee [11], Van Der Werken [26] and Schore [22]. The work performed through Lashlee was performed on a CDC Cyber computer (a non-AFIT resource). Van Der Werken [26:69-70] moved the FORTRAN programs to an Elxsi 6400 superminicomputer (in-house). During Schore's research, the programs were moved once again. The programs now reside on two separate computer systems within AFIT: an ELXSI 6400 superminicomputer and a DEC VAX-11/785 superminicomputer. This section will discuss the (1) preprocessor, (2) the processor, and (3) the postprocessor.

4.3.2 The Preprocessor: SETUPS.F. The preprocessor resides on the DEC VAX-11/785 computer. This is due to the availability of the IMSL [4] library routines on this machine. In addition, this program is linked to a software library named LQGLIB [13]. The preprocessor generates the matrices used to describe the structure over a range of mass and stiffness parameter points. The parameter space is determined by multiplying both the mass and stiffness matrices by ten different scalar values, thus generating the 100-point space. The matrices generated are Φ, $\tilde{\mathbf{B}}_d$, $\tilde{\mathbf{H}}$, and $\mathbf{Q}_d$ for the system model, and Φ_f, $\tilde{\mathbf{B}}_{d_f}$, and $\tilde{\mathbf{H}}_f$ for the filter models, as well as the direct feedthrough matrix $\mathbf{D}$, the Kalman filter gains $\mathbf{K}$, the filter computed residual covariance $\mathbf{A}_k$, and the LQ controller gains $\mathbf{G}_c^*$ for the 100-point parameter space. An input file is used to input the state ($\mathbf{X}$) and control ($\mathbf{U}$) weighting matrices, the measurement noise covariance matrix ($\mathbf{R}$), the dynamics driving noise strength matrix ($\mathbf{Q}$), and the time variables (start time, stop time, and time increment). A second input file provides the mass and stiffness matrices that describe the structure. This program is linked to the LQGLIB [13] and IMSL [4] libraries. The truth models and the reduced-state filter models are stored in two output files to be used by the processor.

4.3.3 The Processor: MOVBNK.F. The primary processor uses the information generated by the preprocessor to perform the moving-bank simulation via Monte Carlo analysis. As Figures 4.1(a) and 4.1(b) indicate, the processor propagates the true system, which is a full 24-state model of the structure, and uses measurements of this system to update the estimator (and controller). At the measurement input to the estimator, noise is introduced in the form of a white Gaussian noise vector. The software has the capability to perform bank expansions, contractions, and movements according to the logic described in Section 2.4.1. These functions are used once the states of the filter bank have been updated and various internal parameters are compared to preset thresholds. This program is linked to the LQGLIB [13] library. At the end of each time increment, pertinent data is written to output files to be postprocessed.

4.3.4 The Postprocessor: RESULT.F. Once the simulation is completed, the data must be reduced using the methods described by Equations (4.5) and (4.6), and put into

a presentable format, i.e., graphs of the mean ± one standard deviation of quantities of interest . The most important values observed in this thesis effort are the errors in position and velocity estimation and the corresponding true positions and velocities for control evaluation. This program provides data in the form compatible with the GNUPLOT plotting routine [5].

4.4 Special Simulation Processes

This section discusses two processes depicted in Figure 4.1(a) and (b) which are required for the estimation and control simulations. First, the generation of a dither signal is discussed, and second, generation of appropriate system dynamics and measurement noise is presented.

4.4.1 Dither Signal. To view the effects of the different modes of the large space structure adequately, a dither signal is applied at the beginning of each ten second Monte Carlo analysis run. The dither signal excites the system model and enhances parameter identification [11:74]. (Schore corrected the dither signal used in previous research and this corrected version is used for this research [22:Chapter 4 pg 7]). The sample period used in this and previous thesis research is 0.05 seconds which corresponds to a sample frequency of:

$$f = \frac{1}{2T} = \frac{1}{2 \times 0.05} = 10Hz \approx 63\frac{rad}{sec} \tag{4.7}$$

From Table 3.2, the first eight average frequencies are all below this 10 Hz frequency and should be adequately excited by the dither signal used for this research. The square wave used does contain harmonics of the fundamental frequency with sufficient energy to excite the higher order modes of the structure. The effect of the higher modes of the structure may well be negligible, but major software modifications would be required to change this limitation. The dither signal magnitude of 10 lbs was determined by trial and error to provide adequate excitation and is applied for 0.5 seconds through the control actuators at node 2 and node 7. For the estimation simulations, after 0.5 seconds the dither is turned off and the actuator outputs are zeroed for the remainder of the simulation time. For the control simulations, after 0.5 seconds the dither is turned off and closed-loop control is

allowed to be applied, as shown in Figure 4.1(b).

4.4.2 Dynamics and Measurement Noise. The driving noise $\underline{w}_d(t_i)$ and the measurement noise $\underline{v}(t_i)$ are provided by the simulation. A zero-mean Gaussian random variable with a variance of one can be approximated by [7:45], [3:58]:

$$\underline{r}(t_i) = \sum_{j=1}^{12} \underline{y}_j - 6 \tag{4.8}$$

where $\underline{y}_j$ is a random variable that is evenly distributed uniformly between zero and one (available through a random number generator). In order to simulate a a zero-mean Gaussian random vector with covariance $\mathbf{Q}_d$, the following operation is performed [7:45], [3:58]:

$$\underline{w}_d(t_i) = \mathbf{A} \cdot \underline{r}(t_i) \tag{4.9}$$

where the elements of $\underline{r}(t_i)$ are computed by Equation (4.8) and $\mathbf{A} = \sqrt{\mathbf{Q}_d}$, i.e., $\mathbf{Q}_d = \mathbf{A} \cdot \mathbf{A}^T$. A Cholesky square root is used to generate the transformation matrix A. Analogously, $\mathbf{A} = \sqrt{\mathbf{R}}$ is used to simulate $\underline{v}(t_i)$. Appendix B provides more detail as to the approximations used to compute the A matrix when the $\mathbf{G}_d$ matrix is not the identity matrix, as is the case for this research. The random number generator used in this research is started with the same seed for every ten-run Monte Carlo set to ensure the same noise samples are used for each study.

4.5 Software Modifications

Several problems with the existing software were encountered during this research. The simulations performed to study the effects of the software modifications became an important early objective of this research. This section briefly discusses the major modifications which were investigated and corrected before the major objectives of this research were initiated. These modifications are discussed according to which of the programs were affected.

4.5.1 The Preprocessor: SETUPS.F. During review of the pre-processor program, three problems were investigated.

First, the feedthrough term, **D**, of Equation (3.27) was calculated as a two-by-two matrix instead of a six-by-three matrix. This matrix results from the truth model order reduction and should improve performance of the filters by providing information of the states not incorporated into the filter model.

Second, the development of the reduced order control input matrix, $\mathbf{B}_r$, of Equation (3.26) did not incorporate the rigid body angular velocity input. Correct development of this matrix will allow optimal control gains computed for the rigid body velocity to be transferred through the actuator at the hub.

Third, the calculation of $\mathbf{A}_k$ implemented the truth model measurement noise covariance matrix instead of the filter model measurement noise covariance. Incorporating the filter measurement noise covariance matrix will alter the computation of $\mathbf{A}_k$, described in Equation (2.18), and therefore the likelihood quotient for residual monitoring shown in Equation (1.3) as well as the density function computation of Equation (1.2) will be affected.

4.5.2 The Processor: MOVBNK.F. During review of the processor program, several discrepancies were corrected. First, a major modification which affected the estimation and control simulations rectified the fact that the error vector was incorrectly computed as the center measurement matrix multiplied by the first filter state estimate. A better approach is to use a Bayesian approach and form a sum of probabilistically weighted filter estimate errors, as seen in Equation (4.1). This will be discussed in more detail in the duplication of past research.

Second, the direct feedthrough term, **D**, calculated from the preprocessor program was not incorporated into the filter measurement model. As stated previously, incorporation of this term in the filter measurement model should improve performance of the filters by providing information of the controls not incorporated previously into the filter model.

Third, the probability density computation of Equation (1.2) was affected by initialization of the p_k's for new filters not being correctly implemented as an equal redistribution

of the discarded filter p_k's. Since the density computation relies on the previous probability calculation, this correction is important (especially for probability monitoring). Also, a software coding error was corrected in the exponential term of the density computation.

Fourth, the residual likelihood quotient used for bank movement was implemented as $\mathbf{r}_j^T \mathbf{r}_j$ instead of as shown in Equation (1.3). Correctly implementing this logic, in addition to the correct use of the filter measurement noise covariance in the $\mathbf{A}_k$ computation, warrants re-evaluation of the bank move threshold. Also, the move logic incorrectly compared the computed minimum likelihood quotient by testing if it was less than the move threshold instead of larger, as discussed in Equation (2.27). This is important in correctly re-evaluating the threshold value.

Fifth, the sum of the probabilistically weighted control inputs for the MMAC implementation was incorrectly initialized. Also, new values of $\mathbf{G}_c^*$ for each of the elemental controllers were not correctly implemented in the moving bank after a movement occurred. These two corrections are important for the comparisons which will be made for alternate control vector formulation, which will be investigated in this research.

4.5.3 The Postprocessor: RESULT.F. The post-processor was modified to be compatible with the GNUPLOT plotting program [5].

4.6 Simulation Plan

The main objective of this thesis effort is to determine the performance of MMAE and controller algorithms. A means to observe the capability of the MMAE and controller algorithms to provide useful estimates (in the case of the filter) and/or control (in the case of the controller) is to monitor the position and velocity estimation errors and true position and velocities of the structure. The purpose of this section is to describe the simulations performed which deal with the performance of the estimation of the structure shape and rigid body orientation, and the performance of the control algorithm, in the presence of a higher order truth model. This research entails five studies for evaluating the estimation performance and two studies for evaluating control performance. These seven studies were

previously outlined in Chapter 1. Before these simulations are presented, a brief discussion describing duplication of past work as well as modification investigations is presented.

4.6.1 Duplication of Past Research. Prior to initiating the moving-bank simulations described in the following sections, an effort was made to duplicate the results produced by Schore [22] . Duplication of past research is valuable in two respects. First, duplication of previous research results is desirable to ensure that the software is operating in the manner as for previously obtained results and to gain familiarity with operating the software. Ideally, in order to satisfy this requirement, duplication of previous research should use the exact software and input parameters. However, for this research, the exact software and input parameters could not be discerned. Therefore, the duplication of past research performed in this research will be considered successful if the results demonstrate the major characteristics of past research. Second, duplication provides a benchmark to evaluate the affects of software modifications. Duplication cases for both estimation and control will be investigated. Once duplications have been performed, investigation of the software modifications will be performed to determine affects on both estimation and control.

4.6.2 Dynamics Noise Strength and Measurement Noise Covariance Determination. The first study conducted to investigate the performance of the estimator is the determination of the dynamics noise strength, $\mathbf{Q}_d$, and measurement noise covariance, $\mathbf{R}$. Values for $\mathbf{Q}_{d_t}$, $\mathbf{Q}_{d_f}$, $\mathbf{R}_t$, and $\mathbf{R}_f$ will need to be determined. Determination of $\mathbf{Q}_{d_t}$ and $\mathbf{R}_t$ will be accomplished first, followed by the values of $\mathbf{R}_f$ and $\mathbf{Q}_{d_f}$

The values of $\mathbf{Q}_{d_t}$ must be large enough to excite the system reasonably over a single propagation cycle, but not too large as to mask the difference between the models. The dynamics noise is evaluated as it affects the physical positions at nodes 1, 2, and 7. In order to determine the values for $\mathbf{Q}_t$, the initial form is an identity matrix. The corresponding $\mathbf{Q}_{d_t}$ is computed as in Equation (2.15). The physical positions and velocities are observed open loop to determine when adequate excitation has been achieved.

The utilization of the same sensors to provide position and velocity measurements on the truss and hub imply a highly correlated noise measurement covariance matrix with

corresponding non-zero off-diagonal elements. The form of $\mathbf{R}_t$ will initially be diagonal with later investigations using a non-diagonal matrix. The evaluation of $\mathbf{R}_t$ is determined physically to represent actual sensor precision.

Once the values for $\mathbf{Q}_{d_t}$ and $\mathbf{R}_t$ are determined, the values for $\mathbf{R}_f$ are determined using the MVRO approach as used by Van Der Werken [26]. The form of $\mathbf{R}_f$ is determined by increasing $\mathbf{R}_t$ by an amount equal to the difference between the 24-state matrix product $\mathbf{H}_t\mathbf{P}_t(t_i^-)\mathbf{H}_t^T$ and the six-state matrix product $\mathbf{H}_f\mathbf{P}_f(t_i^-)\mathbf{H}_f^T$ [26:78]. This difference should be symmetric, positive definite, and provides off-diagonal terms.

Once appropriate values for $\mathbf{Q}_{d_t}$, $\mathbf{R}_t$, and $\mathbf{R}_f$ are found, values for $\mathbf{Q}_{d_f}$ can be determined. The values for this matrix represent the effect of omitting the higher order states from the filter model. A conservative approach must be taken so as not to mask the performance of the elemental filters. As with evaluation of the $\mathbf{Q}_{d_t}$ matrix, the values for this matrix are computed after first selecting the entries of the diagonal continuous-time form, $\mathbf{Q}_f$.

4.6.3 Moving-Bank Logic Study. Once the filters have been properly tuned, the bank will be initialized away from the true parameter and the bank will be free to move. For this study, the bank will be initialized in the middle of the parameter space and the true parameter will be initialized several parameter points away. A finely discretized bank will be employed and expansion and contraction will be disabled. Residual monitoring will be the move logic evaluated first and will establish the performance baseline for the moving bank. This benchmark will be used to evaluate both parameter position estimate and probability monitoring. For each of the studies, an appropriate move threshold will be established to obtain the best possible parameter identification, and therefore, the best possible estimation performance.

4.6.4 Density Function "Bias" Study. Following the move logic investigation, a study to determine the effects of the coefficient term on the probability density function will be performed. This will entail removing the constant coefficient from the conditional density computation in Equation (1.2). Therefore, the new probability density function

will be computed as follows:

$$f_{\underline{\mathbf{z}}(t_i)|\underline{\mathbf{a}},\underline{\mathbf{Z}}(t_{i-1})}(\mathbf{z}_i \mid \mathbf{a}_k, \mathbf{Z}_{i-1}) = \exp\{-\frac{1}{2}\mathbf{r}_k^T(t_i)\mathbf{A}_k^{-1}(t_i)\mathbf{r}_k(t_i)\} \tag{4.10}$$

While the the resulting expression is no longer a true conditional density function since the scale factor is not correct and the resulting area under the curve is no longer one, the probability weightings are still correct since they sum to one because of the denominator term in Equation (1.1). This study will be accomplished by determining the effects of the modification on movement of the bank using residual, parameter position estimate, and probability monitoring. If the results of the study indicate that the $\mathbf{A}_k^{-1}$ term in Equation (4.10) affects the performance adversely, then further investigations will be performed to investigate removal of the $\mathbf{A}_k^{-1}$ term in the exponential, basically investigating the ME/I form discussed in Section 2.3 and described by Equation (2.26).

4.6.5 Estimator Parameter Space Discretization Study. Following the density function study, an improved parameter space discretization will be determined using insights gained from Sheldon's research [24]. Since his research determined that underestimating the modal frequencies had a major destabilizing impact on closed-loop controller performance, a modified parameter space will be determined such that the region of the space which corresponds to the higher modal frequencies in the structure will be more densely discretized. Recalling the development of the system model in modal coordinates in Equation (3.11), the region of the space which corresponds to higher modal frequencies will be the region of lower **M** and higher **K** scaling values. For this study, the current boundaries for each of the scaling parameters will be maintained and a logarithmic scale will be implemented to yield nonlinear discretization values. Using the best moving-bank logic method found in the previous investigations, the performance of the moving-bank estimator with the new parameter space discretization will be determined. The performance of the associated controllers with the new parameter space will also be investigated in the simulations performed for the controller study.

4.6.6 Estimator Varying Parameter Study. The goal of this study will be to evaluate the optimal estimation algorithms previously found, for the case of varying the true parameters. Following the parameter space discretization study, the true parameters will be allowed to vary slowly throughout the space. For this research, a "slow" true parameter variation will be implemented by changing the true parameter value by one discrete point at the midpoint of the simulation. For this research, true parameter "jumps" will also be investigated. For this research, a "jump" in the true parameter will be implemented as a change in the value of the true parameter by four discrete values. In addition to maintaining a finely discretized bank during the "slow" and "jump" changes, bank expansions and contractions will be allowed. Therefore, the appropriate thresholds for contraction and expansion will be determined. As for the space discretization study, the performance of the controllers for varying parameters will also be investigated later.

4.6.7 State and Control Weighting Matrix Determination. This study will determine if the moving-bank controller performance could be enhanced by evaluating appropriate values for the **X** and **U** weighting matrices. The state weighting matrix, **X**, and control weighting matrix, **U**, evaluated by Lashlee [11] will be the starting point for this investigation. The goal of this study will be to maintain relatively small values for **U** while increasing the values of **X** until the rms values for the true states stop decreasing drastically. This will be accomplished for each parameter in the parameter space so that each controller will be tuned for a specific parameter value.

4.6.8 Optimal Control Vector Formulation Study. The goal of the control vector study is to compare the performance of a moving-bank controller using several proposed methodologies. The goal is to optimize the controller performance by investigating the MMAC, modified MMAC, MAP vs Bayesian, and the modified single changeable-gain controller approaches (recall Section 1.1.3). Using the optimal moving-bank parameter estimation logic determined previously, the MMAC approach will be accomplished first and will be considered the benchmark for performance. The modified MMAC, MAP vs Bayesian, and the modified single changeable-gain controller will be compared to the MMAC approach.

4.6.9 Controller Parameter Space Discretization Study. Following the control vector study, the effect of improved parameter space discretization on controller performance will be determined using insights gained from Sheldon's research [24]. A parameter space such that the region of the space which corresponds to the higher modal frequencies in the structure are more densely discretized will be investigated to enhance performance. For this study, the parameter space determined for the previous estimator space discretization study will be used. The MMAC approach will be used to investigate the effects of the new parameter space on control of the structure.

4.6.10 Controller Varying Parameter Study. The goal of this study will be to evaluate the optimal control algorithms for the case varying parameters. Following the space discretization study, the true parameters will be allowed to vary slowly throughout the space. True parameter jumps will also be investigated. In addition to maintaining a finely discretized bank and allowing the bank to move, expanding and contracting the bank will be investigated. This study will use the appropriate thresholds for contraction and expansion previously found in the estimation study.

4.7 Summary

This chapter provided a wide variety of information relating to the simulations performed in this thesis. Section 4.2 explained the need for and use of Monte Carlo analysis as it pertains to this thesis. Sections 4.3 through 4.5 describe the software used to implement the simulations and some of the problems encountered with the implementation. Finally, Section 4.6 presented the simulation plan which will be followed to accomplish the objectives of this research. Chapter 5 describes the results derived from this simulation.

V. Results

5.1 Introduction

The purpose of this thesis is to determine the optimal performance which can be expected from the moving-bank estimation and control algorithms when a six-state Kalman filter and/or associated LQG controller is evaluated against a 24-state truth model. This evaluation is conducted in a physically motivated manner by observing physical positions and velocities of the space structure at various selected locations. This chapter provides the results of the Monte Carlo simulations in two major sections: Section 5.2 provides results from the MMAE investigations, beginning with the simulations that duplicate previous results and continuing through the study plan of the previous chapter, and Section 5.3 provides results from the LQG controller investigations, also beginning with the duplication simulations.

5.2 Multiple Model Adaptive Estimation Study

This section discusses results from the studies which investigated estimation performance of the moving-bank MMAE algorithms. These studies are as follows: 1) duplication of past MMAE research; 2) modifications to software; 3) Kalman filter dynamics noise strength and measurement noise covariance determination; 4) residual, parameter position estimate, and probability monitoring performance determination; 5) density function "bias" investigation; 6) parameter space discretization; and 7) varying-parameter estimation performance. The plots discussed in this section are provided after Section 5.2.6.2.

5.2.1 Duplication of Past MMAE Research. Duplication of past MMAE research entails two studies. The first study is the duplication of Schore's MMAE benchmark simulation [22]. The estimation benchmark consists of setting λ equal to one (yielding the full effects of the 24-state truth model) and placing the filter bank and parameter location at (7,6); i.e., the mass parameter is located at its seventh discrete value out of ten and the stiffness matrix is at its sixth discrete location out of ten. Additionally, the probability of filter five in the bank (the center of the finely discretized bank) is set to one,

and the bank move threshold is set extremely high so as not to allow the bank to move. In essence, this is the case of an artificially informed (with correct parameter values), non-adaptive, single filter. Figures 5.1 through 5.3 provide the results for this duplication. The apparent divergence after 5 seconds demonstrated in the position errors for nodes 1 and 2 duplicate the estimation errors of Schore's research very well. This apparent divergence will be addressed later in Section 5.2.2 also. A second form of non-adaptive benchmark which could have been performed for the duplication study is a single-filter worst-case analysis. For this case, the single filter is set to the (5,5) parameter position and is made to be non-adaptive. The true parameter is located at (7,6). This duplication simulation was not performed since it was felt that the non-adaptive, best-case single filter and the moving-bank simulation, described next, provided an adequate demonstration of MMAE performance duplication.

The second study for the MMAE duplication determines the ability of the bank to move within the filter space when the initial parameter estimate and filter location differ from the true parameter location. The duplication consists of setting the bank center filter to the (5,5) parameter position. The true parameter is located at (7,6). The probabilities of all the filters in the bank are initialized to $\frac{1}{9}$ and the lower limit of the probabilities, p_{min}, is set to 0.05. Residual monitoring is used for the movement logic and the move threshold set to 0.25. These threshold values were determined previously by Schore [22] to provide the best performance for his software configuration. The estimation error results of this duplication are shown in Figures 5.4 through 5.6. Figure 5.7 provides the mean bank center location and the mean parameter estimate in the parameter space. Note that the value of λ is set equal to 0.5 for the bank movement duplication since Schore [22] does not provide figures for λ equal to one.

5.2.2 MMAE Modification Study. Due to the fact that several software issues needed to be investigated, this became a substantial part of initial research efforts in order to determine the affects of the modifications. Individual studies were performed for each error in order to determine the performance impact to physical position and velocity estimation error and parameter estimation. This section provides a discussion of the major impacts.

This section concludes by providing simulation results from the cumulative modifications made to the software.

The first major modification that significantly impacts the estimation errors for all of the simulations performed in this research is the computation of the error vector described in Equation (4.1). Instead of using the Bayesian approach demonstrated in Equation (4.1), the previously developed software used a somewhat modified MAP approach. Specifically, the estimated physical positions and velocities were implemented as the bank center filter measurement matrix, $\tilde{\mathbf{H}}_{f_5}$, multiplied by single-filter state estimates from the first filter in the bank, $\tilde{\hat{\mathbf{x}}}_{f_1}(t_i)$. For the non-adaptive, single-filter benchmark case, this had the effect of providing position and velocity estimates from a single filter that was offset by one discrete step in each direction of parameter space from the bank center filter location. Modification of the error vector computation to that shown in Equation (4.1) removes the divergent estimation errors in the node 1 and 2 position estimation errors for the MMAE, non-adaptive benchmark (Figures 5.1 and 5.2). The effect of this modification is apparent in Figures 5.8 and 5.9. Any effect on the node 7 position error, shown in Figure 5.10, is unobservable. Note also the effect of dither that is obvious in Figure 5.9; this will be discussed with respect to the third modification. Figures 5.8 through 5.10 incorporate all modifications investigated.

The second major modification incorporates the rigid body angular velocity actuator into the development of the $\mathbf{B}_r$ matrix described in Equation (3.26) and developed in Appendix A. Modification of this matrix allows the inertia wheel located at the hub to apply control to the rigid body mode. In the previous software, this control capability was inadvertantly removed. The effects of this are more apparent in the MMAC duplication and will be discussed in Section 5.3.2. However, the estimation process benefits from this modification because the dither signal can adequately excite the modes as intended.

The third major modification incorporates the direct feedthrough term, $\mathbf{D}$, as described in Equation (3.27). Since this study incorporates the full 24-state truth model and a six-state filter model, the correct computation of $\mathbf{D}$ was implemented and incorporated into generation of the residuals, as described in Equation (2.19). This modification is apparent in the dither signal direct feedthrough in node 2 position and velocity error

illustrated in Figure 5.9. There were several other software modifications made which were minor compared to the three described above and were discussed in Section 4.5.

The results obtained with the modified software establish a baseline which will be used for comparing the results of varying $\mathbf{Q}$ and $\mathbf{R}$ values and other variations to be considered in the following sections. Simulations to investigate the modification impacts for the moving-bank MMAE algorithms were not performed since the moving-bank logic was itself to be changed as part of this research effort; also this would have required determination of appropriate move thresholds. These studies are more appropriately investigated as part of the MMAE moving-bank logic study presented in Section 5.2.4.

5.2.3 Dynamics Noise Strength and Measurement Noise Covariance Determination. The goal of this study was to determine appropriate values for the 24-state truth model dynamics noise strength matrix, $\mathbf{Q}_{d_t}$, and measurement noise covariance matrix, $\mathbf{R}_t$. In addition, the six-state filter model dynamics noise strength matrix, $\mathbf{Q}_{d_f}$, and measurement noise covariance matrix, $\mathbf{R}_f$ were determined. As a starting point, previous values for these matrices were used. (The values of $\mathbf{Q}$ and $\mathbf{R}$ used to investigate duplications and modifications are provided in Appendix B.) For these studies, the bank was configured to represent an open-loop, non-adaptive single filter with knowledge of the true parameter location, which was selected as (7,6). This represents the nominal parameter values and is assumed to be representative of the rest of the parameter space. For this study, the full impact of the unmodelled effects is incorporated into the analysis, i.e., λ is set equal to one. A primary consideration, in addition to minimizing the rms estimation errors, is the ability of the tuned filters to perform distinguishably different when the assumed true parameter is correct versus when it is wrong. The initial values of $\mathbf{Q}_{d_t}$ (the values which had been used by Schore [22]) were varied by orders of magnitude to investigate the sensitivity of the truth model and filter. However, small values of $\mathbf{Q}_{d_t}$ (reduced by an order of magnitude) result in a large mean error in the position estimates. This "ringing" effect is the result of low energy values being transferred to the structure by means of the wideband (white) noise process, thus allowing the initial dither signal to remain in the slightly damped structure. Since the higher values of $\mathbf{Q}_{d_t}$ better represent the magnitude of noise which a

space structure may experience, and in order to maintain a baseline consistent with past research, the larger values are used for the remainder of this research.

Next, the values for $\mathbf{R}_t$ were investigated. The $\mathbf{R}_t$ matrix used in previous research provided values characteristic of the physical capabilities of current sensors. However, it was noted that the rigid body angular position measurement noise variance was several orders of magnitude larger than the rest of the diagonal elements. In effect, this was "telling" the filter that the measurements from this sensor were extremely poor and should be weighted less. Since there was no reasonable explanation for this situation, the value was decreased so that the ratio of rigid body angular position to rigid body angular velocity was approximately the same as the node 1 and 2 position-to-velocity ratios. The remainder of the matrix was left unchanged.

With the values of $\mathbf{Q}_{d_t}$ and $\mathbf{R}_t$ determined as above, the values for $\mathbf{R}_f$ and $\mathbf{Q}_{d_f}$ were determined. The values for these matrices were determined by adding appropriate amounts, $\mathbf{R}_{add}$ and $\mathbf{Q}_{add}$, to $\mathbf{R}_t$ and $\mathbf{Q}_{d_t}$, respectively, to account for the fewer states used in the filter model. Using the MVRO approach [16:25], an approximation to $\mathbf{R}_{add}$ was determined from the difference between the 24-state matrix product, $\mathbf{H}_t\mathbf{P}_t(t_i^-)\mathbf{H}_t^T$, and the six-state matrix product, $\mathbf{H}_f\mathbf{P}_f(t_i^-)\mathbf{H}_f^T$. For this determination, $\mathbf{P}_t(t_i^-)$ was computed as an average value from the last five seconds of the simulation; this was in order to avoid the transient period of the simulation. With this initial value of $\mathbf{R}_f$ available, and using the initial value of $\mathbf{Q}_f$ as used by Schore, an iterative approach was used to obtain minimum rms estimation errors. The $\mathbf{R}_f$ matrix which resulted indicated very weak coupling between the hub angular position and velocity measurements and highly correlated position and velocity measurements for both node 1 and node 2. The measurements at the hub are dominated by the rigid body mode, whereas the rigid body and bending modes all contribute importantly to measurements at nodes 1 and 2. For the final $\mathbf{Q}$ and $\mathbf{R}$ values, the quality of the final filter tuning was investigated by determining the covariance of the true position and velocity estimation errors from the Monte Carlo simulation. These values are compared to the steady-state, pre-computed, filter indicated performance $\mathbf{H}_f\mathbf{P}_f(t_i^+)\mathbf{H}_f^T$. ($\mathbf{P}_f(t_i^+)$ was used because the estimation errors for the Monte Carlo simulation are computed after measurement update.) This comparison is provided in Figures 5.11 through 5.13 and shows

that the filter is well tuned for the nominal case, i.e., parameter location (7,6). These values of **Q** and **R** are used for all of the filters in the parameter space.

5.2.3.1 Non-Adaptive Single-Filter Best-Case Analysis. The simulations performed for the **Q** and **R** determination study for parameter location (7,6) also represent the performance for the best-case, non-adaptive, single filter with artificial knowledge of the true parameter. Comparing the results shown in Figures 5.11 through 5.13 with that for the untuned filter of Section 5.2.2, it is apparent from Figures 5.11 and 5.12 that the velocity errors for nodes 1 and 2 have been reduced by an order of magnitude. Also, the large mean error which exhibits some phase information has been removed for these two velocities. More importantly, Figure 5.13 shows that the node 7 position error has been reduced by an order of magnitude. The oscillatory mean error previously exhibited also has been removed. Thus, estimation accuracy for the rigid body position of the structure has been improved.

5.2.3.2 Non-Adaptive Single-Filter Worst-Case Analysis. In addition to the best-case benchmark, a worst-case, non-adaptive case was investigated. Previous efforts [11, 22] considered a non-adaptive filter located at parameter location (5,5) with the true parameter located at (7,6) to be the worst case. However, for this configuration, the performance of the filter for this case does not differ significantly from that of the best-case benchmark. Therefore, a non-adaptive single filter was located at (5,5) with the true parameter set to (2,9). This was selected so that the estimator performance could be established for the case of under-estimation of bending mode natural frequencies through assumed parameter values, which will be of major concern for the following controller studies. The results are shown in Figures 5.14 through 5.16. The figures indicate that the node 1 and 2 position error magnitudes are about twice as large as the best-case benchmark.

5.2.3.3 Open-Loop Structure Positions. Parameters which will be highlighted in the following controller performance evaluations are the true positions at selected points on the structure. As described previously, these are found via $\mathbf{H}_t\underline{\mathbf{x}}_t$. The open-loop struc-

ture positions are shown in Figure 5.17 for the nominal parameter value (7,6). Recalling the eigenvalues from Table 3.2, the structure is stable. The error variances shown for node 1 and node 2 have not reached steady state values. Based on the damping coefficient for the highest mode, the settling time is approximately 0.04 seconds. However, the lowest order bending mode has a settling time of approximately 88.91 seconds. The true position for node 7, which represents the rigid body position of the structure, is not damped and can be interpreted as a rigid body rotation in either direction in the x-y plane. These plots provide a baseline of comparison for exhibiting the effectiveness of MMAC control later.

5.2.4 MMAE Moving-Bank Logic Study. The goal of this study was to investigate the best move logic of those discussed previously. The best logic is determined not only by how well the bank identifies the true parameter, but also by how well the states, or in this case structure positions and velocities, are estimated. Although the purpose is not to design a parameter identifier, the state estimation errors are expected to be less for the logic which provides the algorithms with the best parameter identification. This section discusses results from evaluation of the best threshold for each logic, the effects of the density function "bias" on each logic, as well as removing the $\mathbf{A}_k$ matrix of Equation (1.2) entirely from the density function, i.e., using the ME/I approach. The simulation performed for the following studies consisted of centering the bank at parameter location (5,5) and positioning the true parameter at location (7,6). (This simulation will be referred to as case 1). The probabilities of the nine filters were initially set to $\frac{1}{9}$ and the bank was allowed to move. The results for these studies are presented first for residual monitoring, then for probability monitoring, and finally for parameter position estimate monitoring.

5.2.4.1 Residual Monitoring Analysis. The first step taken to investigate this method for bank movement was to establish performance for the bank with the full conditional density function as in Equation (1.2) for probability p_k computation and a likelihood quotient of $\mathbf{r}_k^T(t_i)\mathbf{A}_k^{-1}\mathbf{r}_k(t_i)$ for bank-moving decisions. Performance evaluation is determined by observing the mean parameter estimate and mean center filter location for the Monte Carlo study as well as the physical structure state estimation errors. The significant result from this study is that the incorporation of the $\mathbf{A}_k^{-1}$ matrix in the likelihood quotient

causes the bank to move in an incorrect direction. The movement of the bank is characterized by a dramatic move toward the lowest mass parameter value and a moderate move toward a higher stiffness parameter, as shown in Figure 5.18. It was determined that the scaling of the likelihood quotient quadratic form by the $\mathbf{A}_k^{-1}$ might cause such undesirable behavior, and therefore an alternate likelihood computation should be investigated. This alternate form was to omit the $\mathbf{A}_k^{-1}$ matrix from the likelihood quotient, which resulted in $\mathbf{r}_k^T(t_i)\mathbf{r}_k(t_i)$ as the scalar quantity upon which to base bank-moving decisions. With this alternate form incorporated, the simulation was repeated. Initially, the bank movement is as before, however, the parameter identification improves after the first half of the run and the estimated parameter moves toward the correct parameter. Extending the simulation time from 10 seconds to 15 seconds does not improve the final parameter estimation.

It was thought that the "biased" results obtained from the bank movement could be due to the leading coefficient of the probability density function computations of Equation (1.2). Therefore, as earlier proposed, the coefficient was removed from the density computations and the benchmark simulation performed. The alternate form for the residual likelihood quotient was maintained. The resulting parameter estimation still exhibits a strong bias tendency toward a low mass parameter and a higher stiffness parameter. Finally, the ME/I approach was used to compute the probability density function, which is equivalent to replacing $\mathbf{A}_k$ by $\mathbf{I}$ throughout the entire density of Equation (1.2), and in the likelihood quotient as well. For this move logic, a threshold of 0.09 was established as providing the best results. The results of the associated parameter identification are shown in Figure 5.19.

5.2.4.2 Probability Monitoring Analysis. This study began by incorporating the entire probability density function, as provided in Equation (1.2), into the algorithms. However, with this configuration, the mass parameter estimate (and resulting bank movement) exhibits the same biased movement as for the residual logic study. Therefore, the leading coefficient was removed from the density function. The resulting performance of the bank is not significantly affected by the modified algorithm. Therefore, total removal of the $\mathbf{A}_k^{-1}$ matrix from the density computation, or ME/I density function computation, was

used. This modification improves the mass parameter estimate, although it still exhibits an initial bias tendency. The best performance for this method is obtained for a move threshold of 0.13. The parameter identification results for this configuration are shown in Figure 5.20.

5.2.4.3 Parameter Position Estimate Monitoring Analysis. As for the previous two move logic studies, the initial form for the moving-bank algorithms used the complete form for the probability density function of Equation (1.2). However, as might be anticipated based on the results from the previous two studies, the performance of the bank to this algorithm indicate an obvious bias toward a lower mass and higher stiffness parameter values. Removing the leading coefficient coefficient from the density function computation does not significantly correct the biased movement. Finally, the ME/I form alleviates the severity of the parameter estimation bias. The best results are obtained for a move threshold of 0.01. These results are shown in Figure 5.21.

5.2.4.4 MMAE Moving-Bank Logic Summary. The results obtained for the parameter position estimate logic study represent the best moving-bank logic results for the benchmark case investigated here. While these results indicate slightly better performance for the ME/I density function form, in order to investigate and compare the capabilities of the ME/I and proper density forms fully, both methodologies are investigated for the MMAE moving-bank analysis which follows.

5.2.4.5 MMAE Moving-Bank Analysis. In order to evaluate the performance of the moving-bank algorithms throughout the parameter space, two simulations were performed in addition to the simulation conducted for the moving-bank logic analysis described in Section 5.2.4. These two additional simulations were as follows:

1. Case 2: True parameter at location (2,9) with bank initially at location (5,5)
2. Case 3: True parameter at location (9,2) with bank initially at location (5,5)

For both of the cases listed above, the probabilities of the bank filters are initially set to $\frac{1}{9}$ and the threshold for parameter position estimate monitoring is set to 0.01. For

these simulations, both the ME/I and proper density function computations are used. Case 2 represents the performance of the bank to initial underestimation of the natural frequencies for the bending modes. Case 3 represents the performance of the bank to initial overestimation of the natural frequencies.

For the case 2 study, Figures 5.22 through 5.24 show the state estimation performance using ME/I computations. The position estimation errors for nodes 1 and 2 indicate increasing uncertainty, similar to that seen earlier in the duplication studies in Section 5.2.1. The parameter estimation performance is shown in Figure 5 25. The performance obtained for the proper density computation provides final parameter estimation slightly worse (mass parameter estimate of 3.5 and stiffness estimate of 8.1). The estimation errors also exhibit the increasing uncertainty trends for the node 1 and 2 position errors as for the ME/I computation.

For the case 3 study, the performance using ME/I provides rather poor parameter estimation, with a final mass estimate of 6.7 and stiffness estimate of 4.5. The truss position estimates indicate the same trends of increasing uncertainty. Again, the results for incorporating the proper density are only slightly worse, with a final mass estimate of 6.2 and stiffness estimate of 5.1.

A summary of the results for the three test cases using ME/I computations is provided in the upper half of Tables 5.1 and 5.2. For all three test cases examined, the two cases investigated here and the first case investigated in Section 5.2.4, the moving-bank state estimation errors show increasing uncertainty, which is worse than the non-adaptive benchmark investigated in Section 5.2.3.1. The parameter estimation results presented indicate the performance at the end of the simulation, i.e., at $t = 10$ seconds. This is selected as a representative performance indicator since the parameter estimation throughout the duration of the simulation run is relatively steady, as shown in Figures 5.19 through 5.21. The results indicate that the ME/I computation performs slightly better than the proper density function and is the only form of density computation investigated for the remainder of the moving-bank MMAE studies.

Table 5.1. MMAE Parameter Space Discretization Position Estimation Errors

Old Parameter Space Discretization				
Simulation	Node 1 Error(1)		Node 2 Error(1)	
⇓	Mean	1σ	Mean	1σ
Case 1	0.0003	0.1585	0.0003	0.0901
Case 2	0.0047	0.1042	0.0027	0.0587
Case 3	-.0139	0.1503	-.0078	0.0849
New Parameter Space Discretization				
Case 1	-.0167	0.1445	-.0095	0.0827
Case 2	0.0006	0.0807	0.0004	0.0463
Case 3	0.0009	0.1442	0.0006	0.0828
Note (1): Average values for 10 seconds (inches)				

Table 5.2. MMAE Parameter Space Discretization Position Estimation Errors and Parameter Estimates

Old Parameter Space Discretization				
Simulation	Node 7 Error(1)		$\underline{\hat{a}}$ (2)	
⇓	Mean	1σ	M	S
Case 1	0.0005	0.2611	6.5	6.9
Case 2	0.0078	0.1734	3.2	8.9
Case 3	-.0228	0.2473	6.7	4.5
New Parameter Space Discretization				
Case 1	-.0274	0.2784	3.6	8.5
Case 2	0.0011	0.1332	2.1	6.9
Case 3	0.0016	0.2381	6.1	4.5
Note (1): Average values for 10 seconds (radians)				
Note (2): Mass (M) and Stiffness (S) estimates at $t = 10$ seconds				

5.2.5 Estimator Parameter Space Discretization Study. Motivated by the controller study which follows, the region of the parameter space for higher values of modal frequencies was more densely packed than the region for lower frequencies. This was an attempt to reduce the possibility of applying potentially destabilizing control to the structure from controllers based on incorrect models that underestimated true modal frequencies. The boundaries for the mass and stiffness scaling factors remained the same and a logarithmic scaling was performed for each of the 10 mass and stiffness parameters. The new parameter space, determined in this ad hoc fashion, is provided in Table 5.3. For this study, the three test cases described in Section 5.2.4.5 were performed using parameter position logic and ME/I computations. The results for the three test cases are provided in the bottom half of Tables 5.1 and 5.2. The structure estimation errors and the parameter estimation obtained for this new space do not significantly differ from that of the original space discretization. These results are not unexpected since it wasn't anticipated that the new parameter space discretization would make a significant difference for the estimation process. Therefore, for the remaining varying parameter simulations, the original space is used.

Table 5.3. New Discretized Parameter Space

a_k	Lashlee [11]		New Space	
⇓	Mass	Stiffness	Mass	Stiffness
1	0.50	0.50	0.500	0.500
2	0.55	0.60	0.541	0.801
3	0.60	0.70	0.587	0.977
4	0.70	0.80	0.639	1.102
5	0.80	0.90	0.700	1.199
6	0.90	1.00	0.771	1.278
7	1.00	1.16	0.858	1.345
8	1.20	1.26	0.971	1.403
9	1.30	1.40	1.129	1.454
10	1.40	1.50	1.400	1.500

5.2.6 Estimator Varying Parameter Study. The goal of this study was to investigate the performance of the estimator algorthms to two types of parameter changes: first, slowly moving parameters, where the true parameter does not vary by more than one discrete value at a time, and second, a jump change where the true parameter changes by several discrete values. In order to investigate the performance of the algorithms to slowly varying and jump parameters properly, benchmarks for establishing best-case performance must be performed. The benchmarks for these simulations are similar to the non-adaptive, best-case single-filter study performed in Section 5.2.3.1. However, for these benchmarks, the bank is allowed to move and it is artificially informed of the true parameter change during the simulation. A benchmark for each parameter variation study must be performed in order to analyze the results correctly.

5.2.6.1 Slowly Varying Parameter Analysis. For the slowly varying parameter study, two cases were investigated:

1. Case 1: True parameter and bank initially at parameter location (5,5) with true parameter change to (4,6)
2. Case 2: True parameter and bank initially at parameter location (5,5) with true parameter change to (6,4)

The first test case represents a change toward higher natural frequencies of the bending modes, whereas the second test case represents a change toward lower natural frequencies. For each test case, the true parameter variation was a discrete change implemented at t=5.0 seconds into each 10-second Monte Carlo run. A better approach to simulating slowly varying parameters would consist of interpolating the truth model matrices, i.e. Φ, $\mathbf{H}_t$, and $\mathbf{B}_{d_t}$, between the discrete parameter points. Therefore, the only difference between the slowly varying parameter study and the jump parameter study conducted in this research is the number of discrete points that the true parameter is changed. (Since the jump parameter study is only a more severe case of the slowly varying parameter study, the simulation results will only provide plots for the jump parameter study, while summarizing both sets of results in table form). The move logic consisted of parameter

position estimation with ME/I computations. The results of these simulations, as well as the corresponding benchmark results, are provided in Tables 5.4 and 5.5. For both of the test cases, the parameter estimation is very poor. For both case 1 and case 2, the truss position estimation error results are similar to that obtained for the constant parameter moving-bank analysis of Section 5.2.4.5. Basically, the node 1 and node 2 position errors exhibit increasing uncertainty and are much worse than the benchmarks.

5.2.6.2 Jump Parameter Analysis. For the jump parameter study, two methods were investigated for allowing the bank to adapt to the parameter change. The first method maintains a fine discretization and allows the bank only to move to the new parameter. The second method not only allows the bank to move but also to expand to acquire the new parameter and then contract around it. For this case, appropriate thresholds were determined and are provided in the following discussion. For each of the two bank adaptation methods, two parameter jump cases were investigated:

1. Case 1: True parameter and bank initially at parameter location (5,5) with true parameter change to (2,9)
2. Case 2: True parameter and bank initially at parameter location (5,5) with true parameter change to (9,2)

The first test case represents a change toward higher natural frequencies, whereas the second test case represents a change toward lower natural frequencies. For the first method, the move logic consisted of parameter position estimate monitoring and ME/I computations. The artificially informed single-filter benchmark performance for the case 1 study is provided in Figures 5.26 through 5.29. Figures 5.30 through 5.33 show the state estimation errors and the parameter estimation for the fine-bank move for case 1. (Results for this simulation, as well as the case 2 study, are provided in Tables 5.6 and 5.7). The state estimation errors indicate an increasing uncertainty and do much worse than the benchmark. The stiffness estimate in Figure 5.33 looks as though no jump occurred. However, Figure 5.34 shows that the bank properly reacts when the jump change is in the opposite direction, as is the situation for case 2. For case 2, the estimation performance is similar to that for case 1 in that the node 1 and 2 position errors demonstrate a divergent trend.

Table 5.4. MMAE Varying Parameter Position Estimation Errors

Simulation	Node 1 Error(1)		Node 2 Error(1)	
⇓	Mean	1σ	Mean	1σ
Benchmarks				
Bench 1	0.0001	0.0024	0.0000	0.0015
Bench 2	0.0011	0.0037	0.0002	0.0008
Fine-Bank Movement				
Case 1	-.0095	0.2865	-.0056	0.1623
Case 2	-.0171	0.2494	-.0101	0.1471
Note (1): Average values for $t > 5.0$ seconds (inches)				

Table 5.5. MMAE Varying Parameter Position Estimation Errors and Parameter Estimates

Simulation	Node 7 Error(1)		$\underline{\hat{a}}$ (2)	
⇓	Mean	1σ	M	S
Benchmarks				
Bench 1	0.0000	0.0322	N/A	N/A
Bench 2	0.0001	0.0027	N/A	N/A
Fine-Bank Movement				
Case 1	-.0157	0.4722	4.2	8.0
Case 2	-.0286	0.4157	6.4	5.3
Note (1): Average values for $t > 5.0$ seconds (radians)				
Note (2): Mass (M) and Stiffness (S) estimates at $t = 10$ seconds				

The second method for acquiring the parameter after a jump change allows expansion of the bank. After establishing an appropriate expansion threshold of 0.5 using $\mathbf{r}^T(t_i)\mathbf{r}(t_i)$ as the scalar quantity to monitor for expansion decisions, the simulations were performed with the bank allowed to expand. Two contraction thresholds were investigated: the first threshold contracts from coarse discretization (four parameter locations between filters in the bank) to medium discretization (two parameter locations between filters in the bank) and was established at 10.00; the second threshold contracts from medium discretization to fine discretization (filters in the bank are at adjacent parameter locations) and was established at 5.00. The mass and stiffness estimates were compared separately to the thresholds, which allowed for rectangular banks. The results for both case 1 and case 2, as shown in Tables 5.6 and 5.7, demonstrate that parameter estimation is worse and the resulting structure estimation errors are slightly worse than for the method of only allowing the finely-discretized bank to move.

Table 5.6. MMAE Jump Parameter Position Estimation Errors

Simulation	Node 1 Error(1)		Node 2 Error(1)	
⇓	Mean	1σ	Mean	1σ
Benchmarks				
Bench 1	0.0000	0.0023	0.0000	0.0006
Bench 2	0.0000	0.0035	0.0001	0.0010
Fine-Bank Movement				
Case 1	-.0146	0.2063	-.0081	0.1158
Case 2	0.0027	0.2434	0.0018	0.1378
Expansion/Contraction				
Case 1	-.0429	0.2178	-.0114	0.1654
Case 2	0.0058	0.3561	0.0076	0.1723
Note (1): Average values for $t > 5.0$ seconds (inches)				

Table 5.7. MMAE Jump Parameter Position Estimation Errors and Parameter Estimates

Simulation	Node 7 Error(1)		$\underline{\hat{a}}$ (2)	
⇓	Mean	1σ	M	S
Benchmarks				
Bench 1	0.0000	0.0007	N/A	N/A
Bench 2	0.0003	0.0013	N/A	N/A
Fine-Bank Movement				
Case 1	-.0239	0.3391	3.5	8.7
Case 2	0.0046	0.4008	7.5	4.2
Expansion/Contraction				
Case 1	-.0346	0.3924	4.3	6.1
Case 2	0.0051	0.5195	7.3	5.1
Note (1): Average values for $t > 5.0$ seconds (radians)				
Note (2): Mass (M) and Stiffness (S) estimates at $t = 10$ seconds				

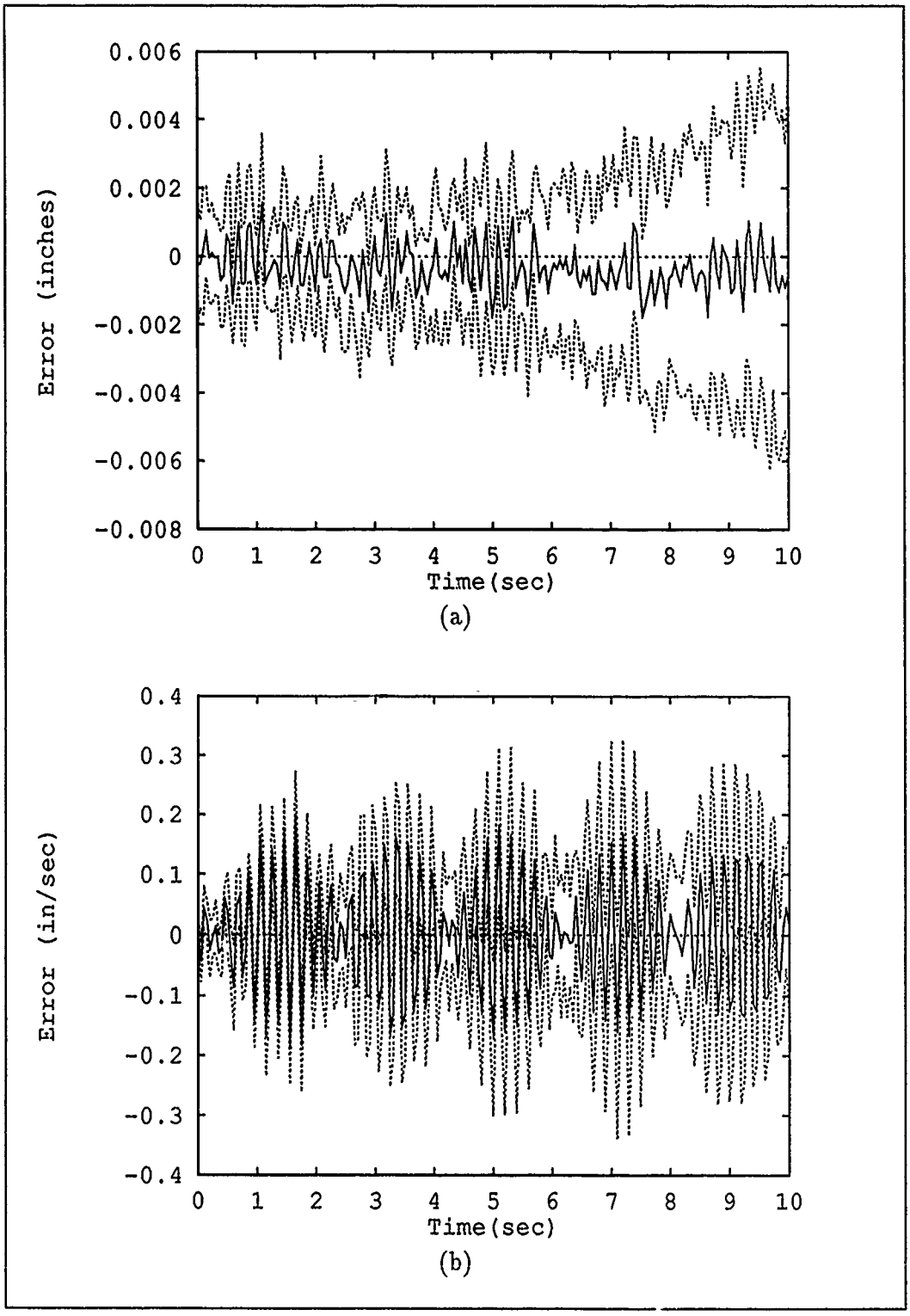

Figure 5.1. Duplication of Single-Filter Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter and Filter at (7,6). (a) Position and (b) Velocity Estimation Errors.

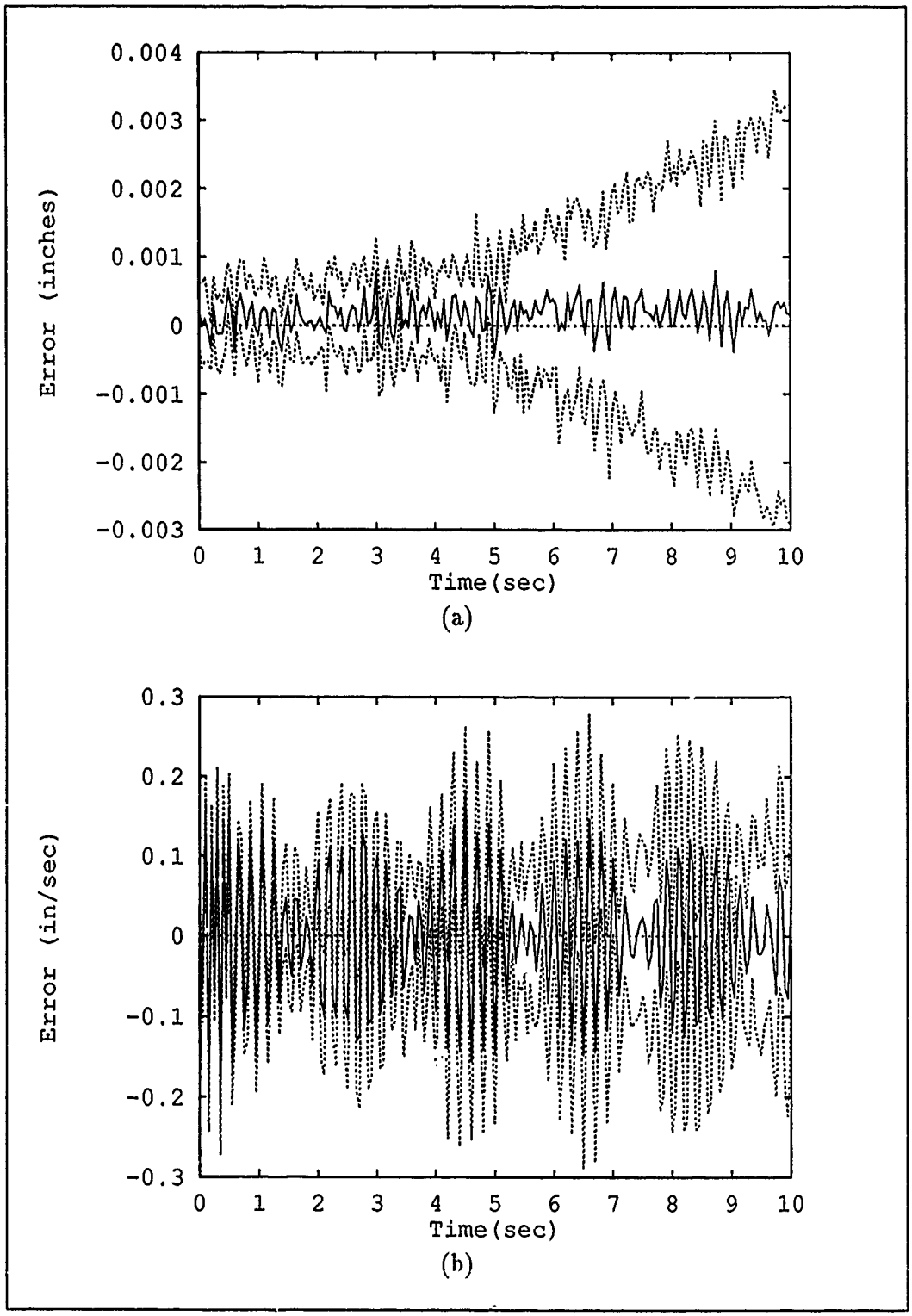

Figure 5.2. Duplication of Single-Filter Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter and Filter at (7,6). (a) Position and (b) Velocity Estimation Errors.

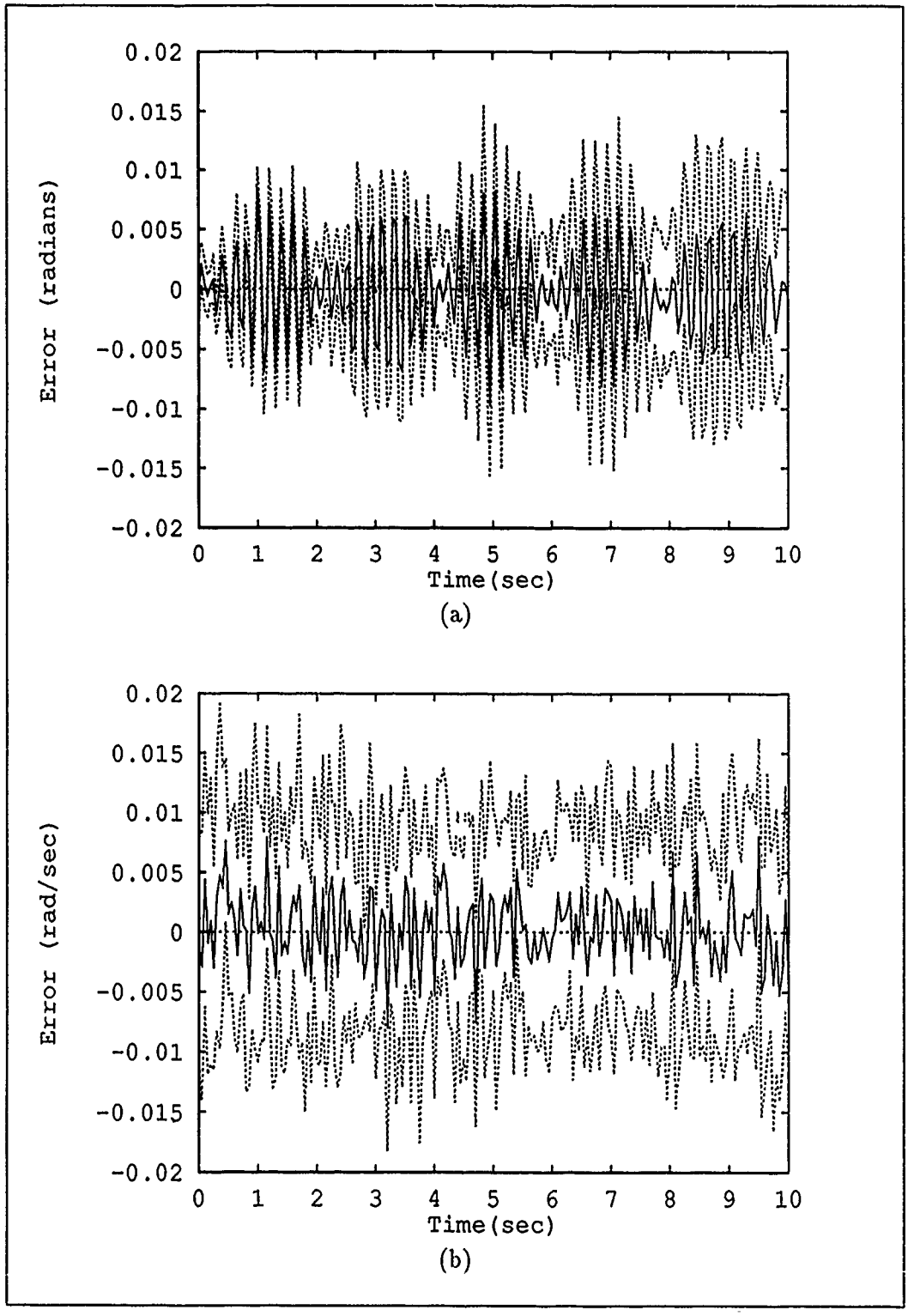

Figure 5.3. Duplication of Single-Filter Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter and Filter at (7,6). (a) Position and (b) Velocity Estimation Errors.

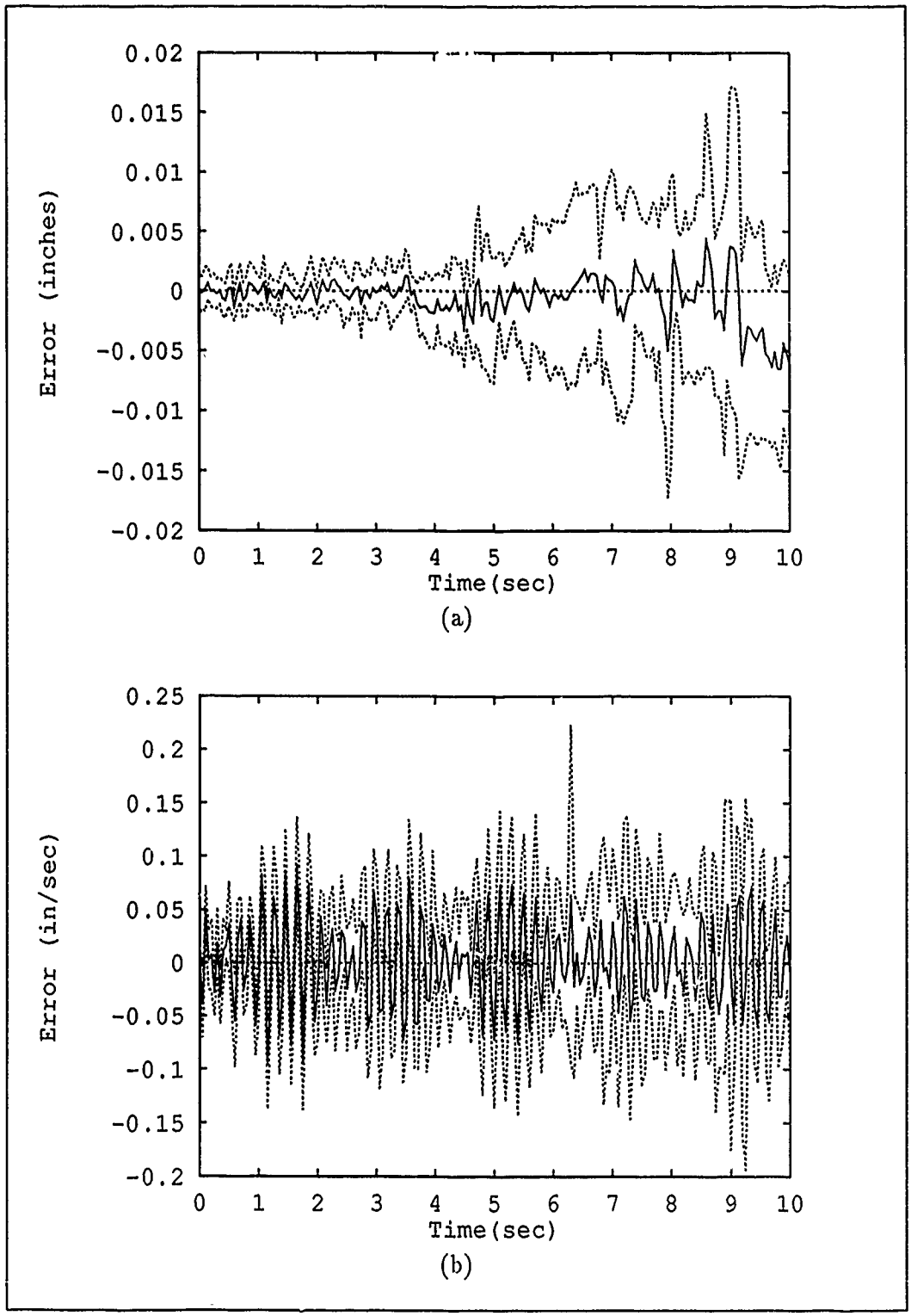

Figure 5.4. Duplication of Moving-Bank Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter at (7,6) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors. λ=0.5.

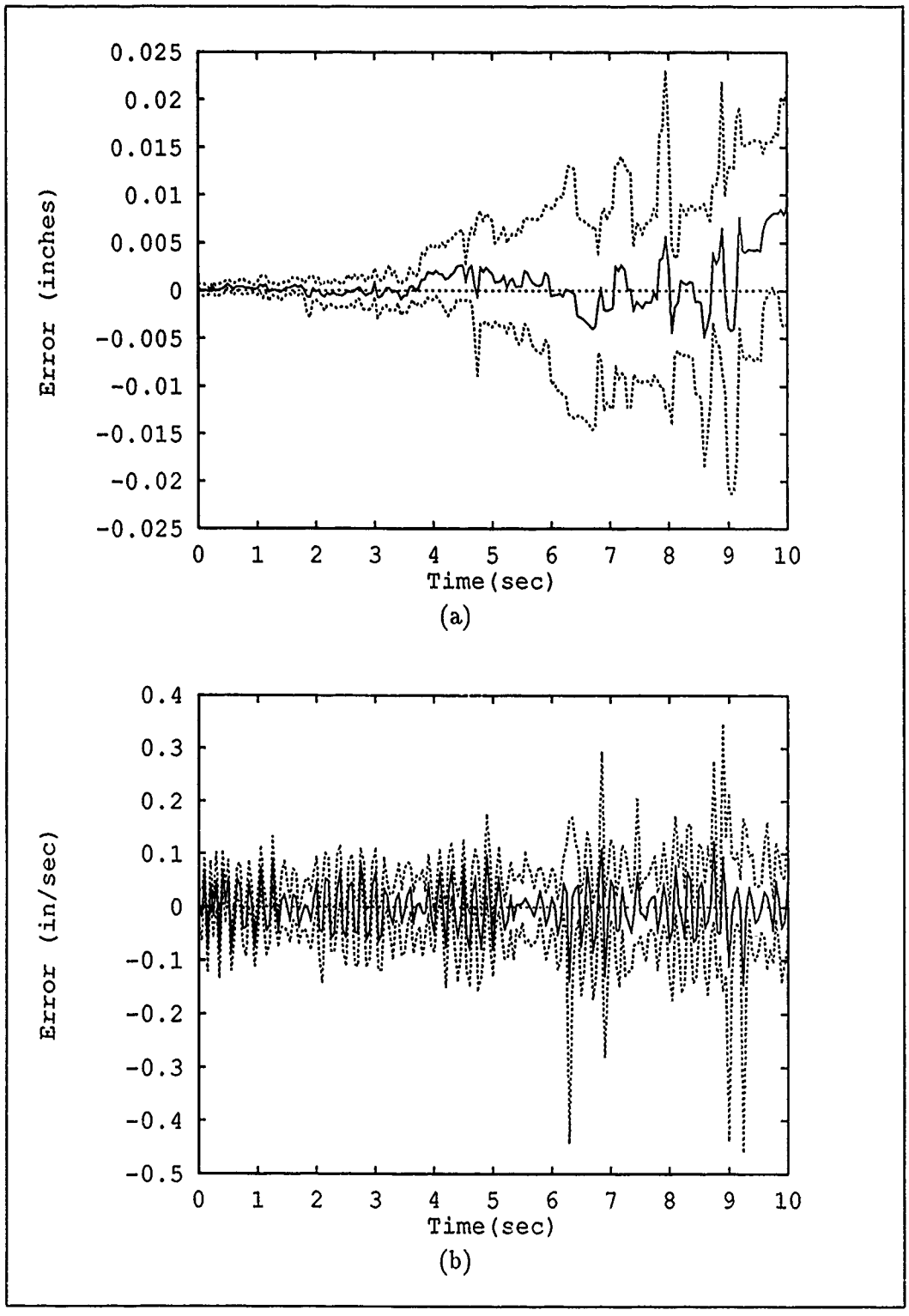

Figure 5.5. Duplication of Moving-Bank Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter at (7,6) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors. λ=0.5.

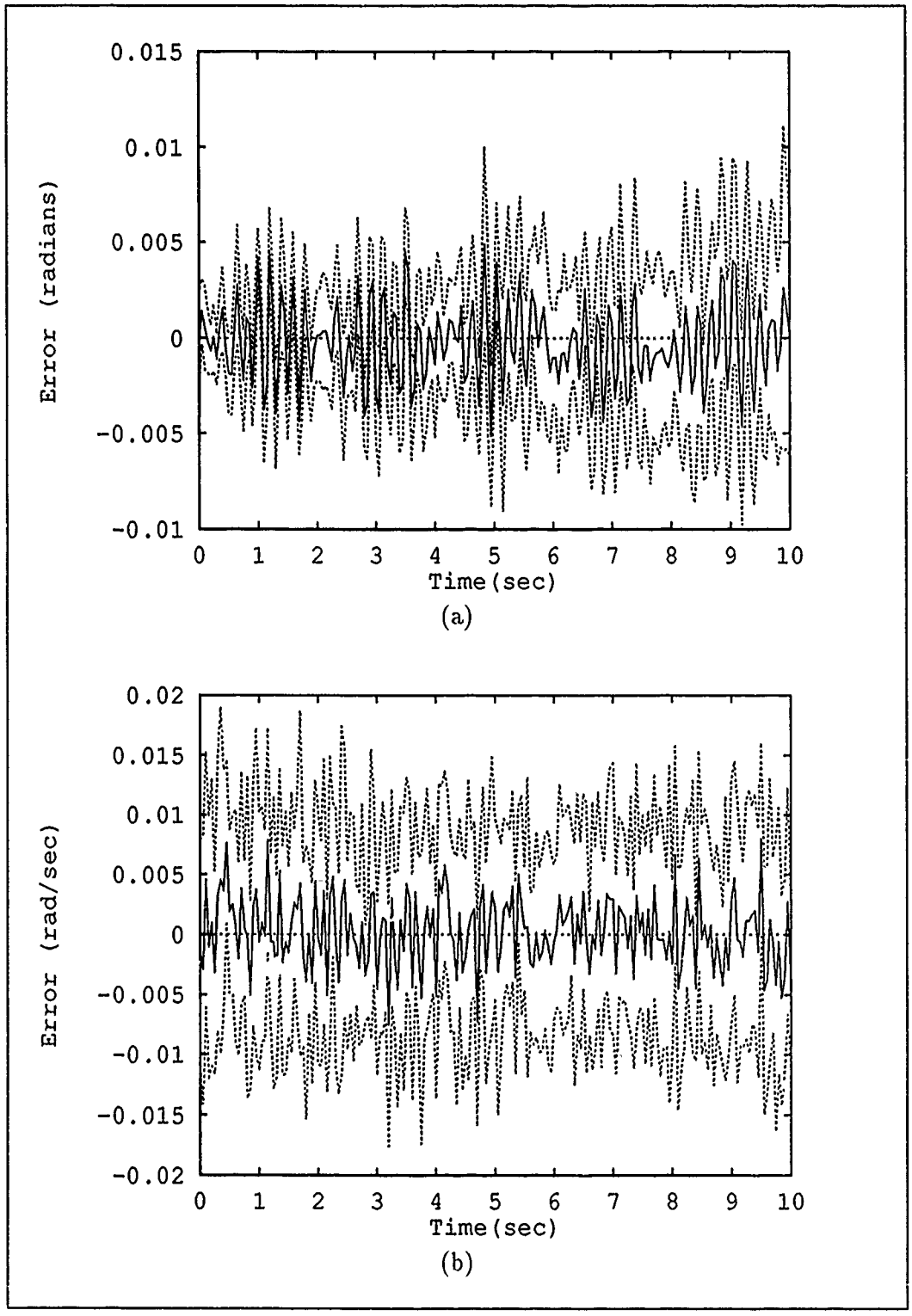

Figure 5.6. Duplication of Moving-Bank Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter at (7,6) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors. λ=0.5.

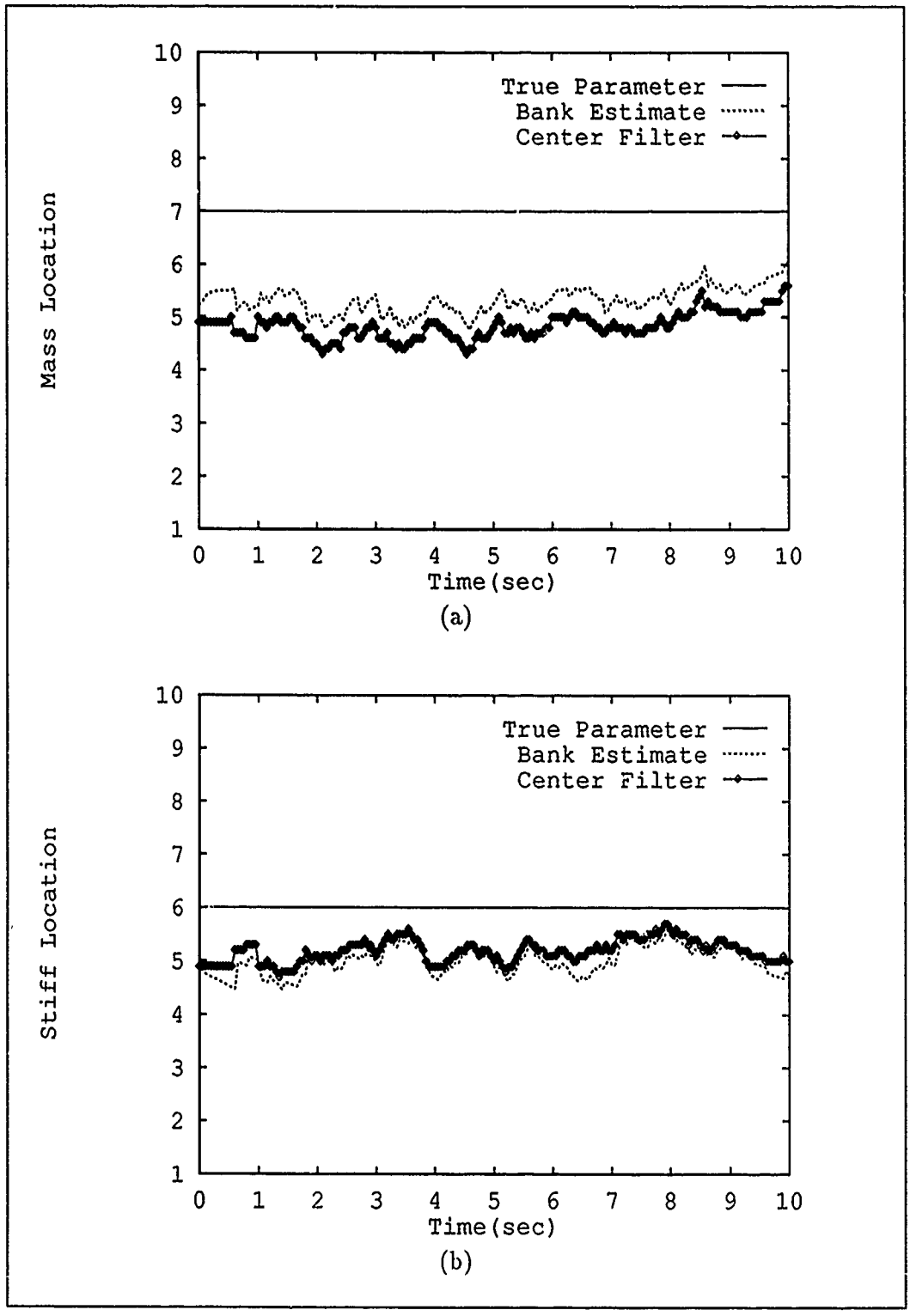

Figure 5.7. Duplication of Moving-Bank Parameter Estimation (Sample Mean) and Actual Filter Bank Location - True Parameter at (7,6) and Bank Initially at (5,5). (a) Mass and (b) Stiffness Parameters. λ=0.5.

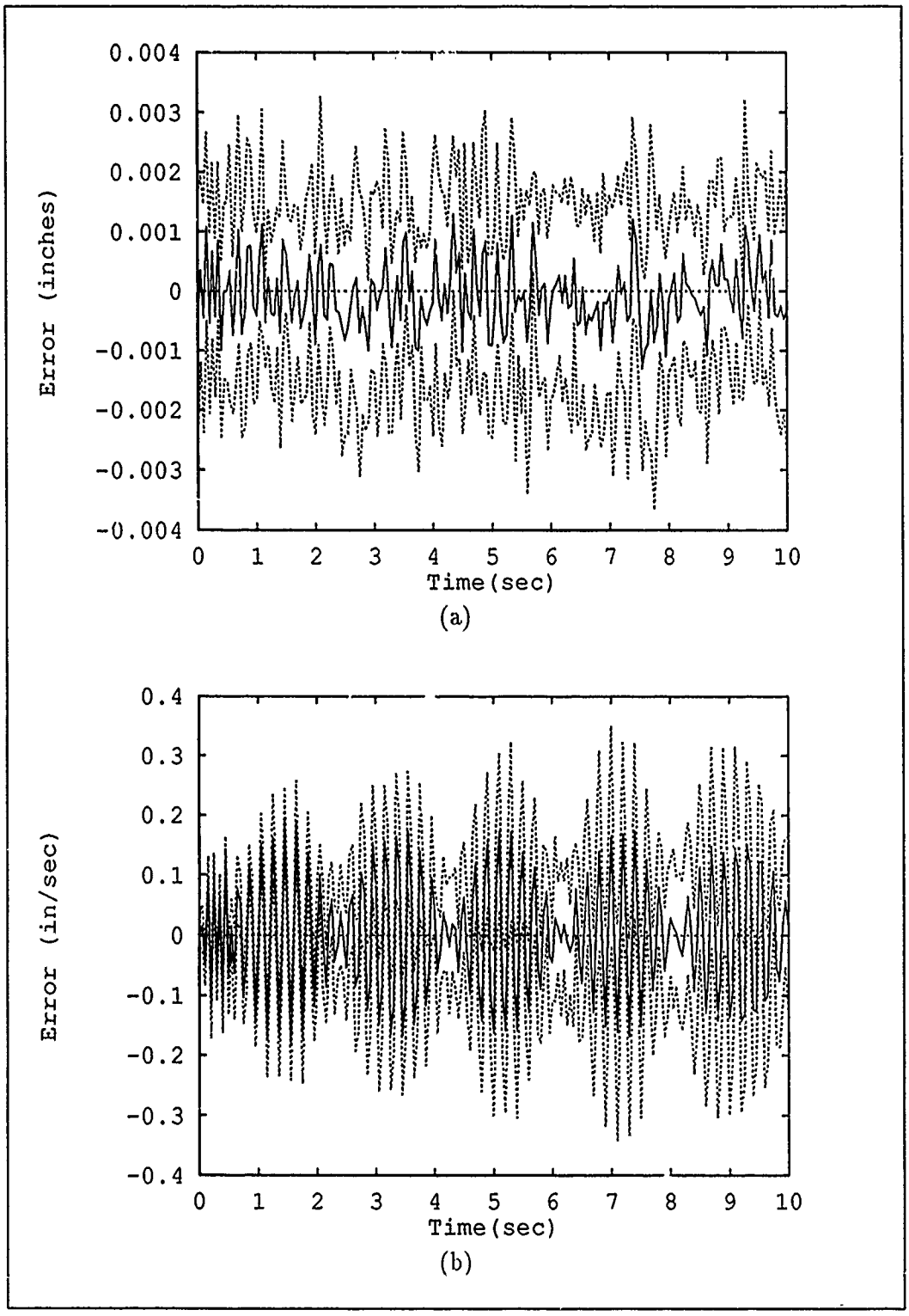

Figure 5.8. Modified Single-Filter Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter and Filter at (7,6). (a) Position and (b) Velocity Estimation Errors.

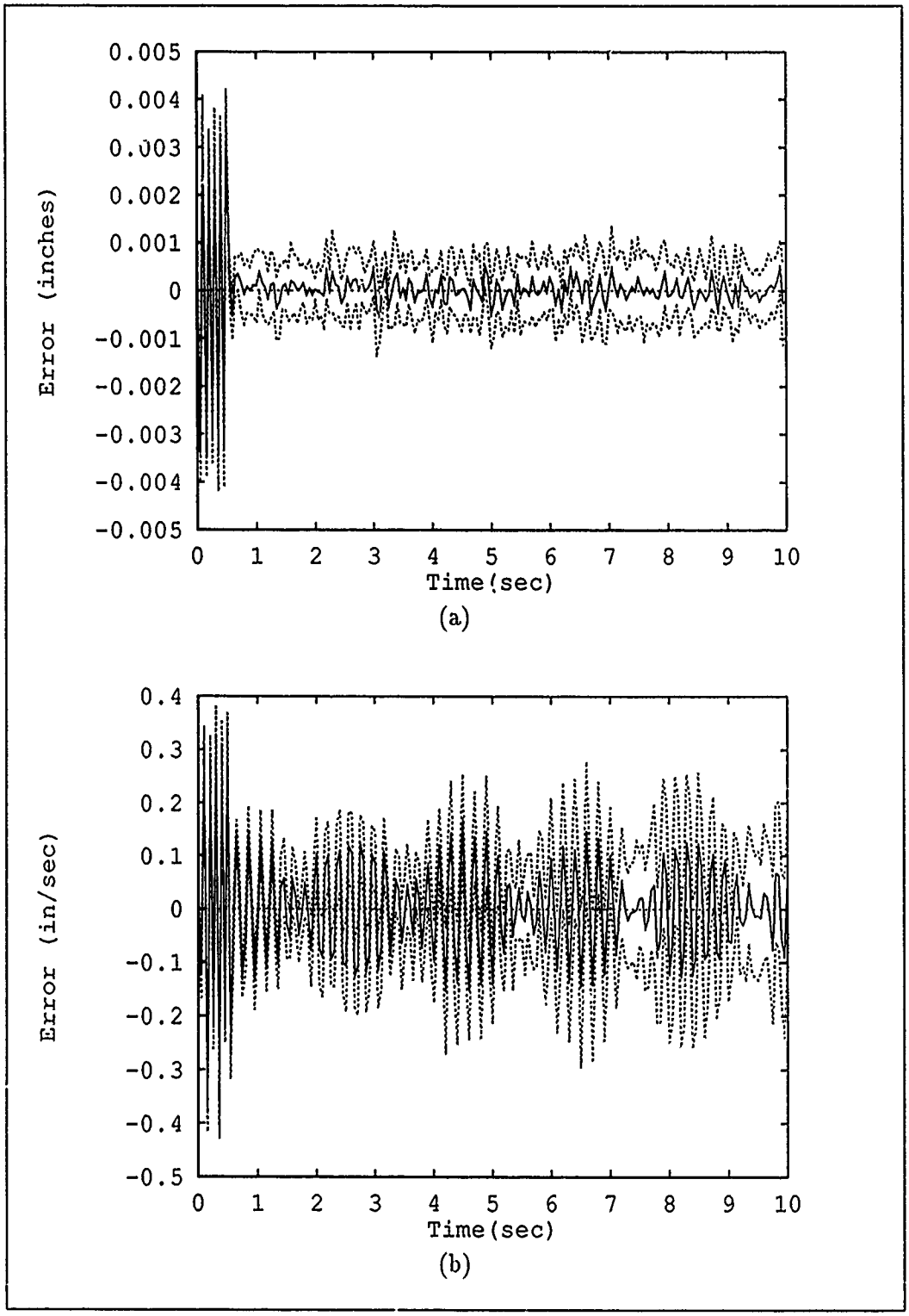

Figure 5.9. Modified Single-Filter Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter and Filter at (7,6). (a) Position and (b) Velocity Estimation Errors.

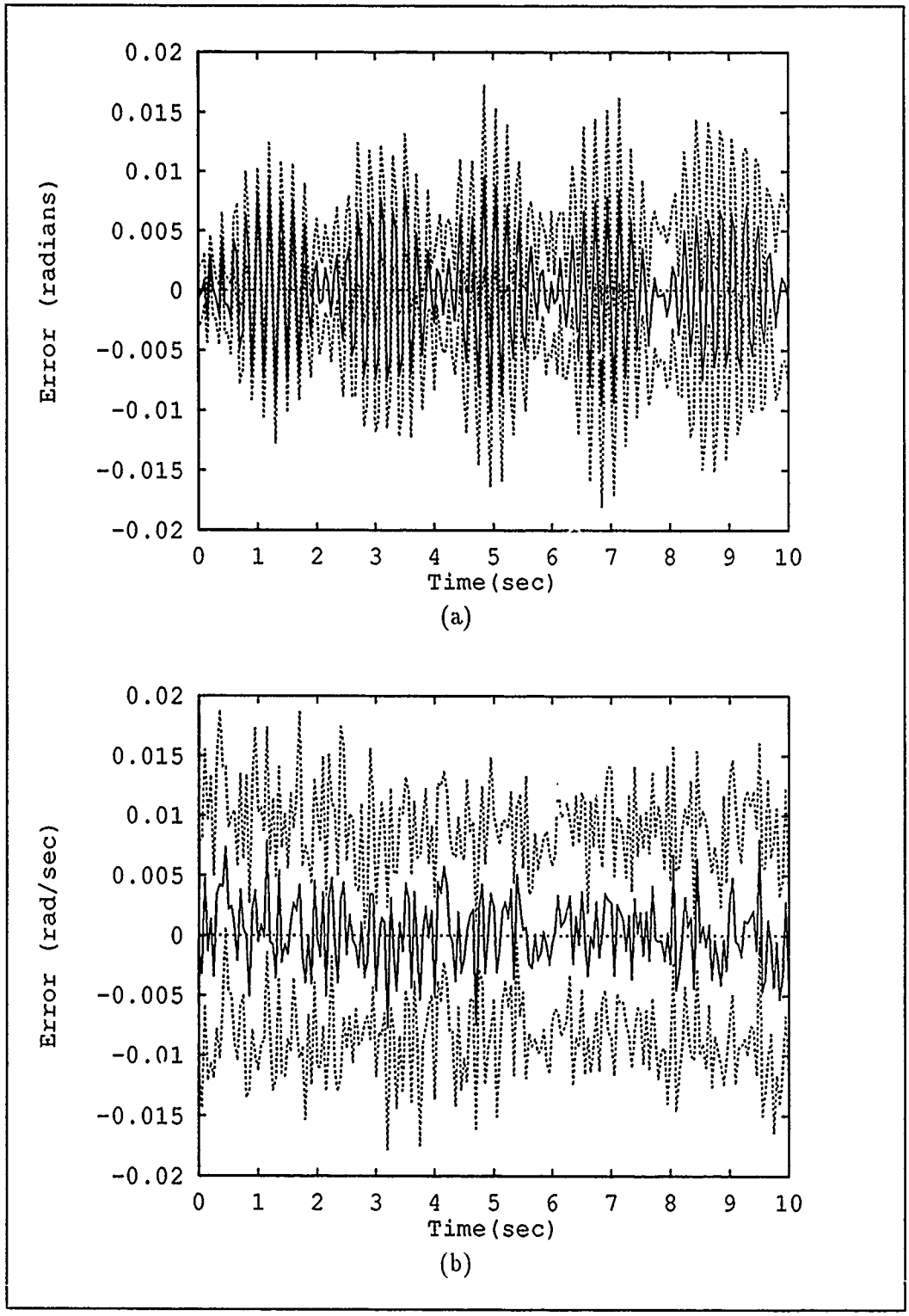

Figure 5.10. Modified Single-Filter Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter and Filter at (7,6). (a) Position and (b) Velocity Estimation Errors.

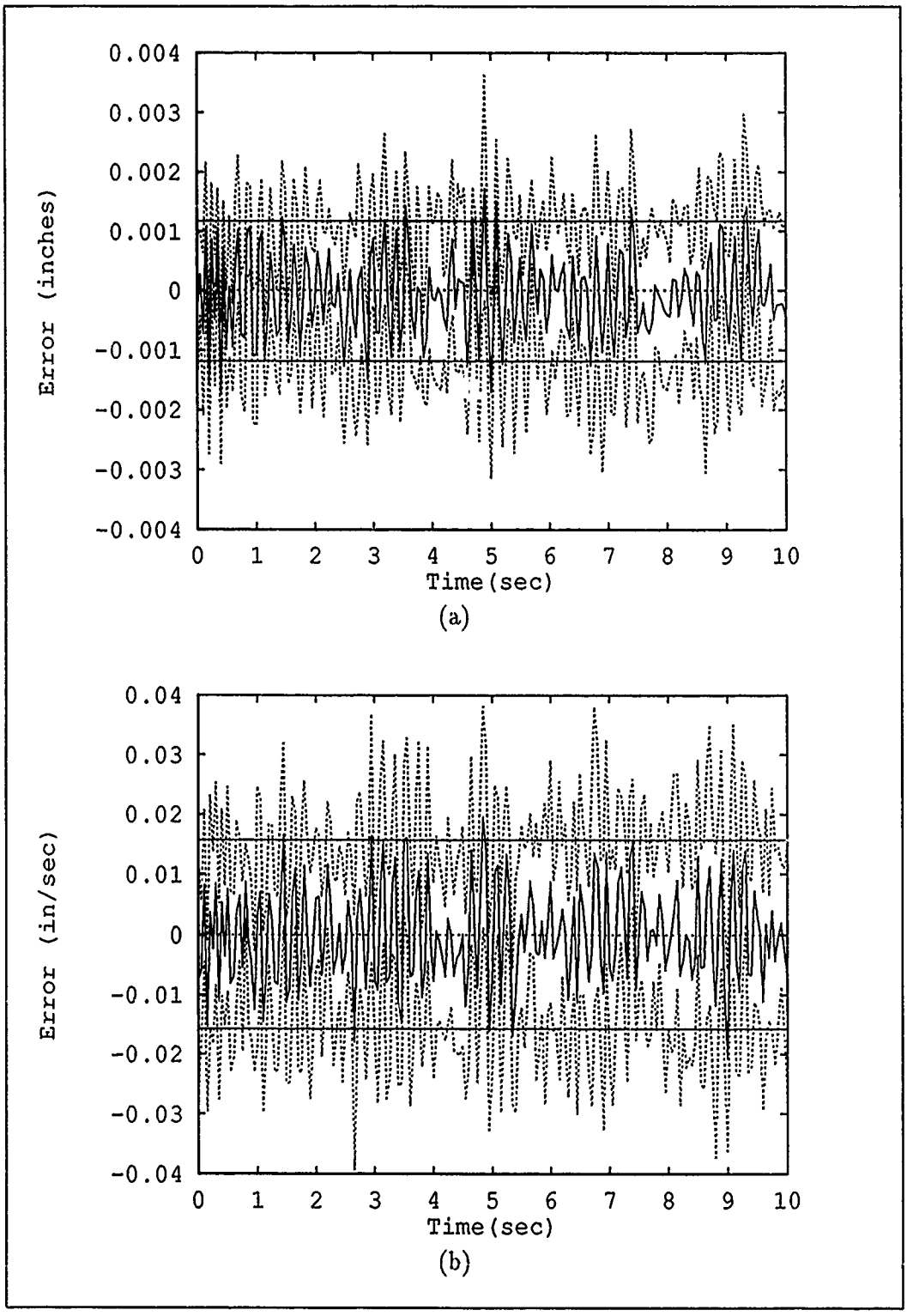

Figure 5.11. Tuned Single-Filter Estimation Errors (Mean $\pm$ One Standard Deviation) with Filter-Computed Errors (Zero $\pm$ One Standard Deviation), Node 1 - True Parameter and Filter at (7,6). (a) Position and (b) Velocity Estimation Errors.

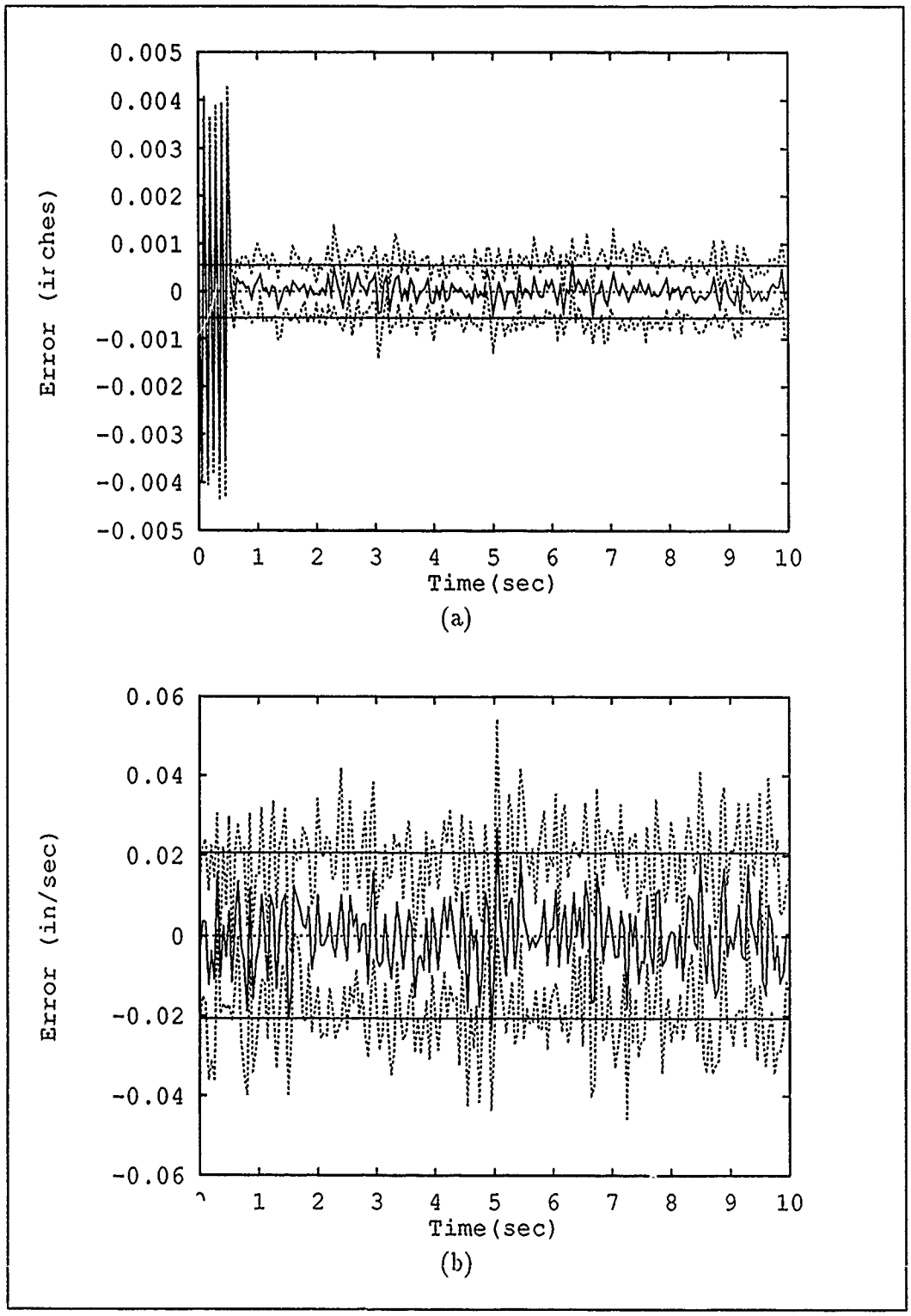

Figure 5.12. Tuned Single-Filter Estimation Errors (Mean $\pm$ One Standard Deviation) with Filter-Computed Errors (Zero $\pm$ One Standard Deviation) , Node 2 - True Parameter and Filter at (7,6). (a) Position and (b) Velocity Estimation Errors.

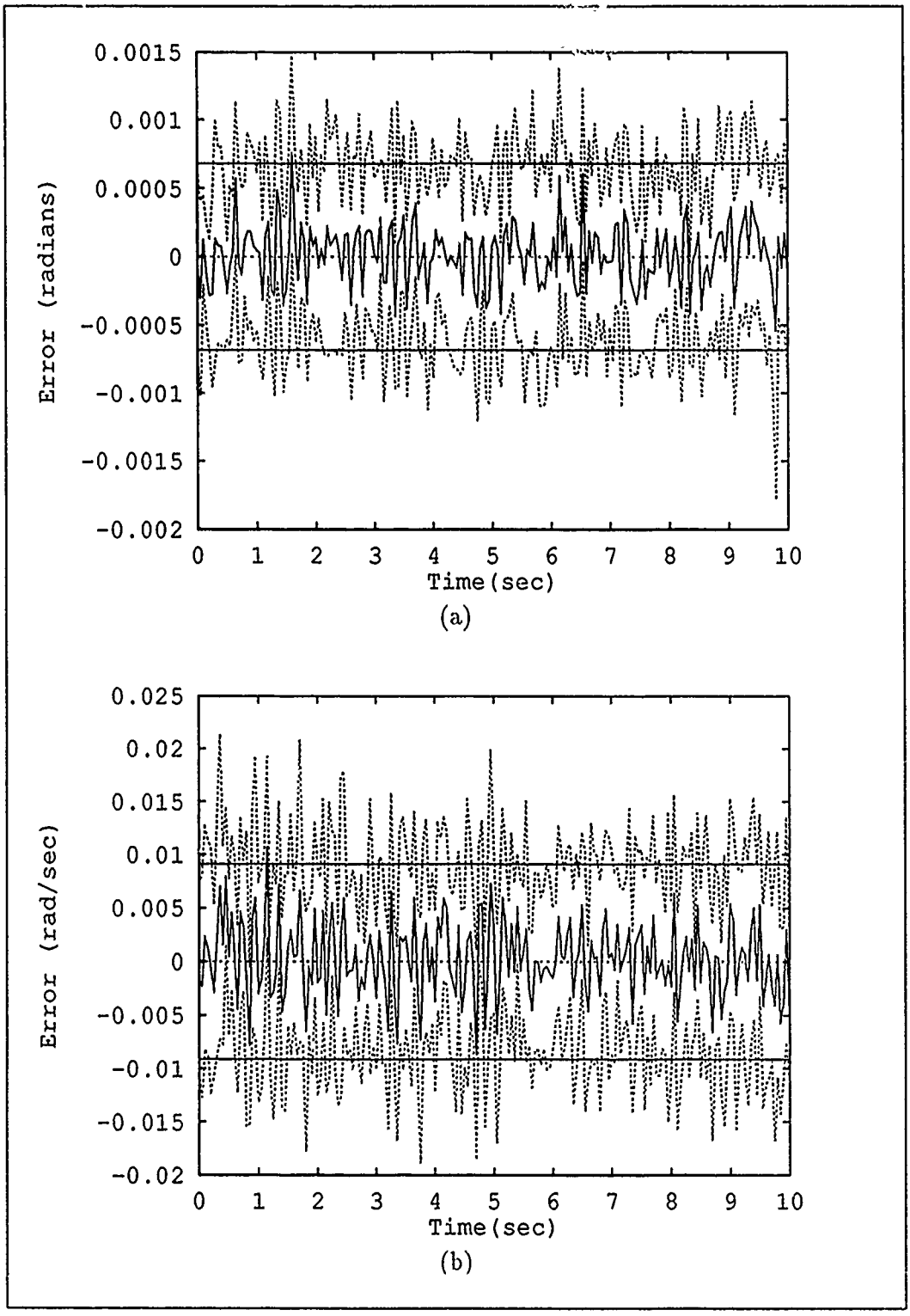

Figure 5.13. Tuned Single-Filter Estimation Errors (Mean $\pm$ One Standard Deviation) with Filter-Computed Errors (Zero $\pm$ One Standard Deviation), Node 7 - True Parameter and Filter at (7,6). (a) Position and (b) Velocity Estimation Errors.

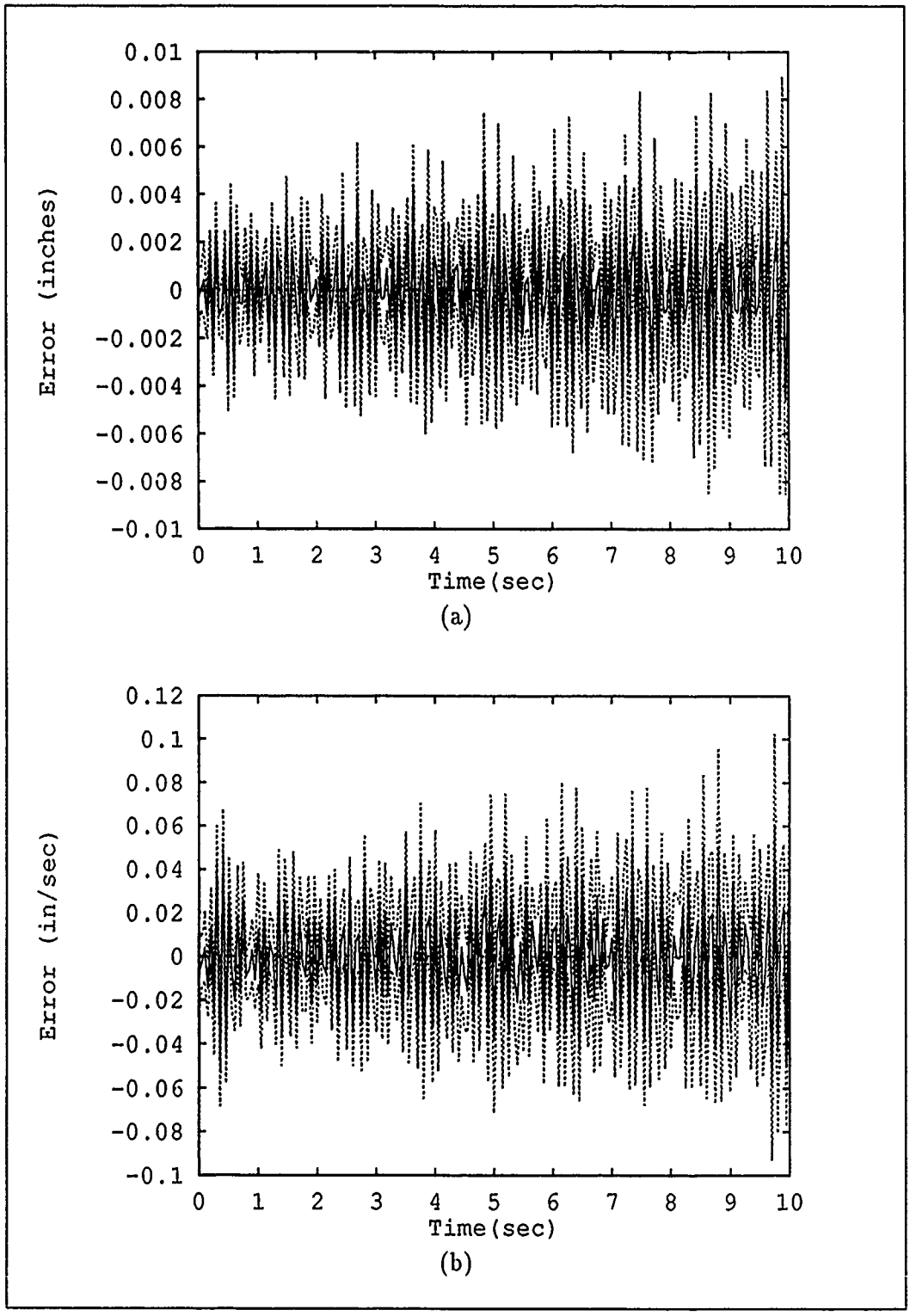

Figure 5.14. Single-Filter Worst-Case Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter at (2,9) and Filter at (5,5). (a) Position and (b) Velocity Estimation Errors.

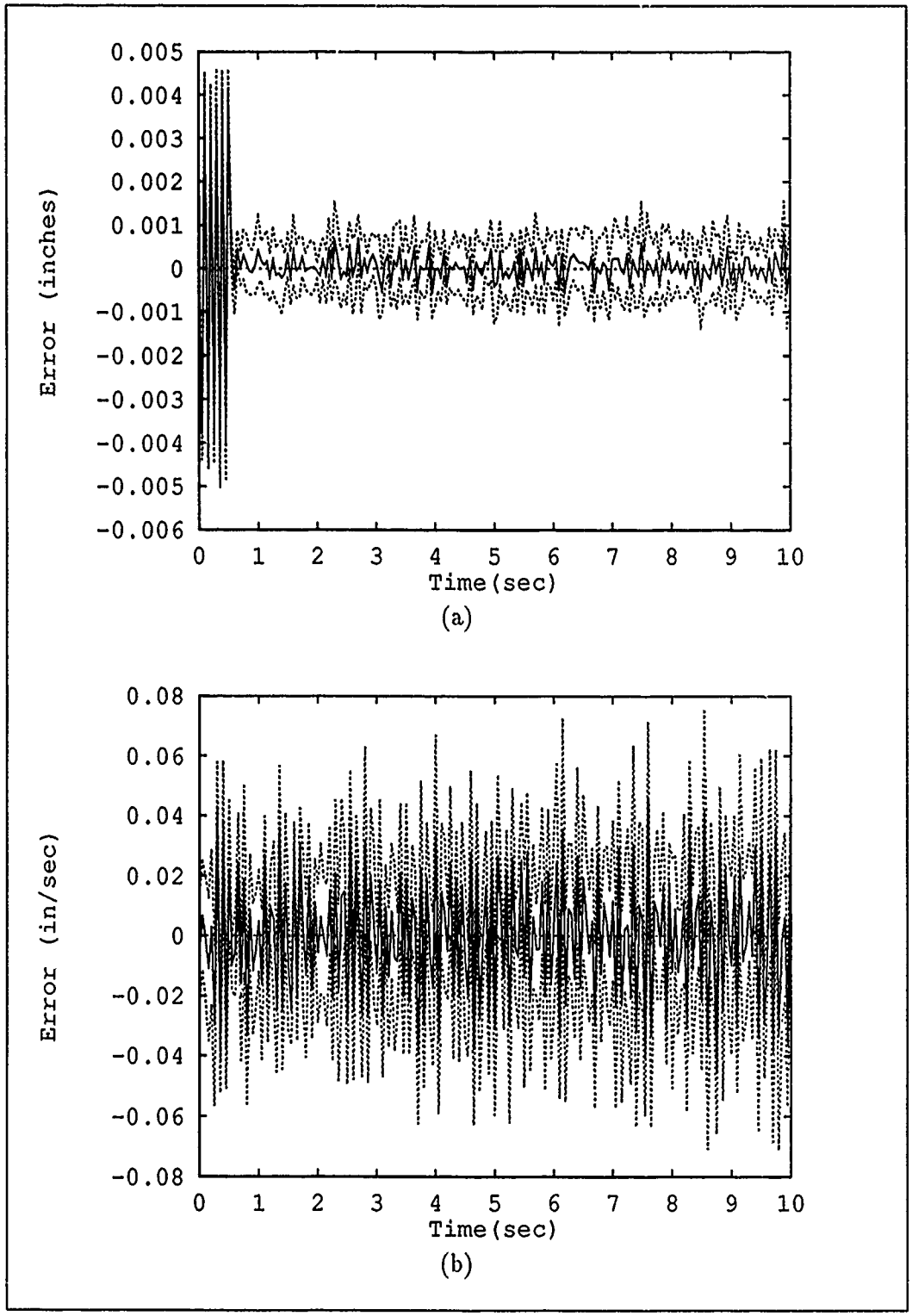

Figure 5.15. Single-Filter Worst-Case Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter at (2,9) and Filter at (5,5). (a) Position and (b) Velocity Estimation Errors.

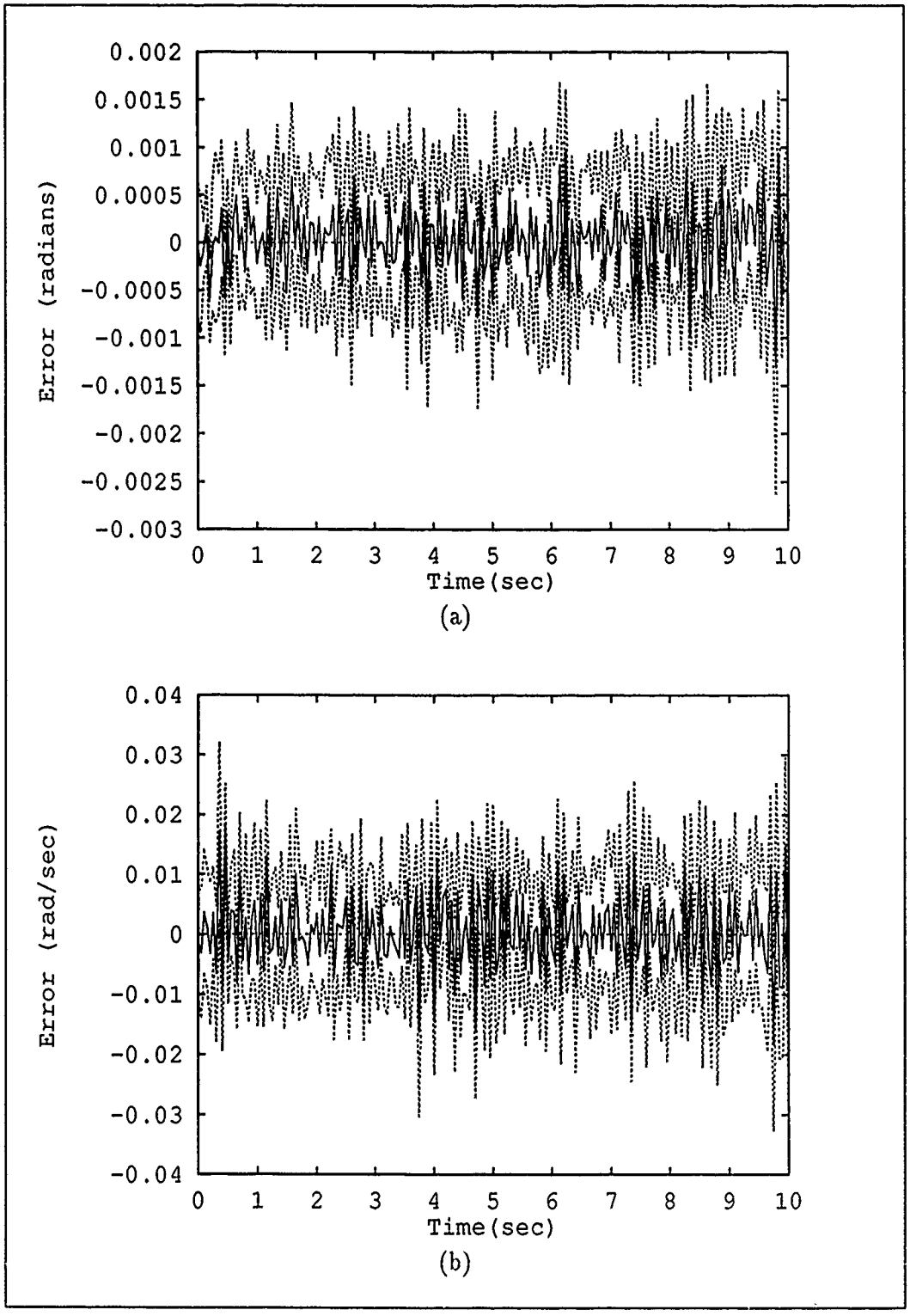

Figure 5.16. Single-Filter Worst-Case Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter at (2,9) and Filter at (5,5). (a) Position and (b) Velocity Estimation Errors.

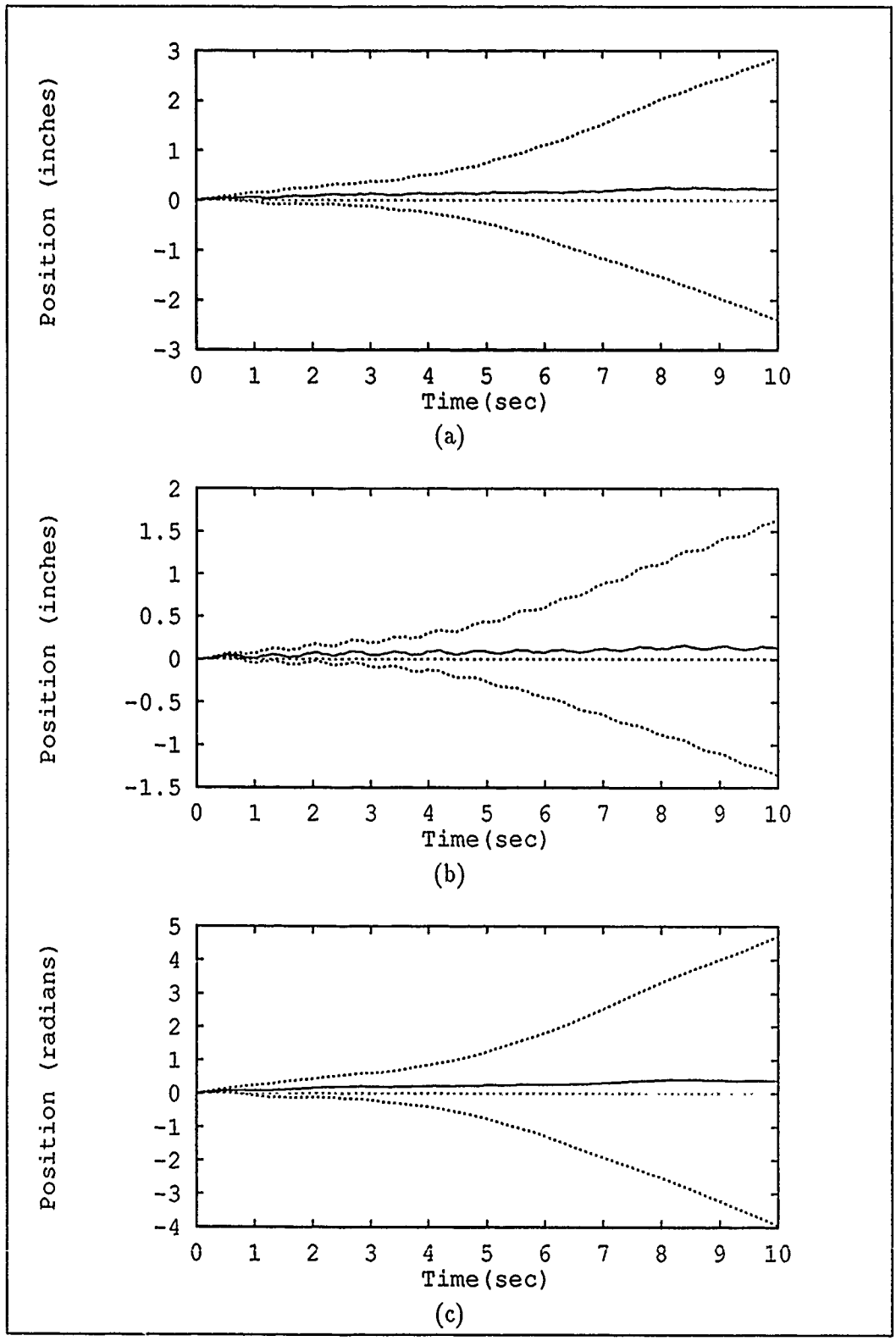

Figure 5.17. True Structure Positions (Mean $\pm$ One Standard Deviation) for Nominal Parameter - (7,6). (a) Node 1 (b) Node 2 and (c) Node 7 Positions.

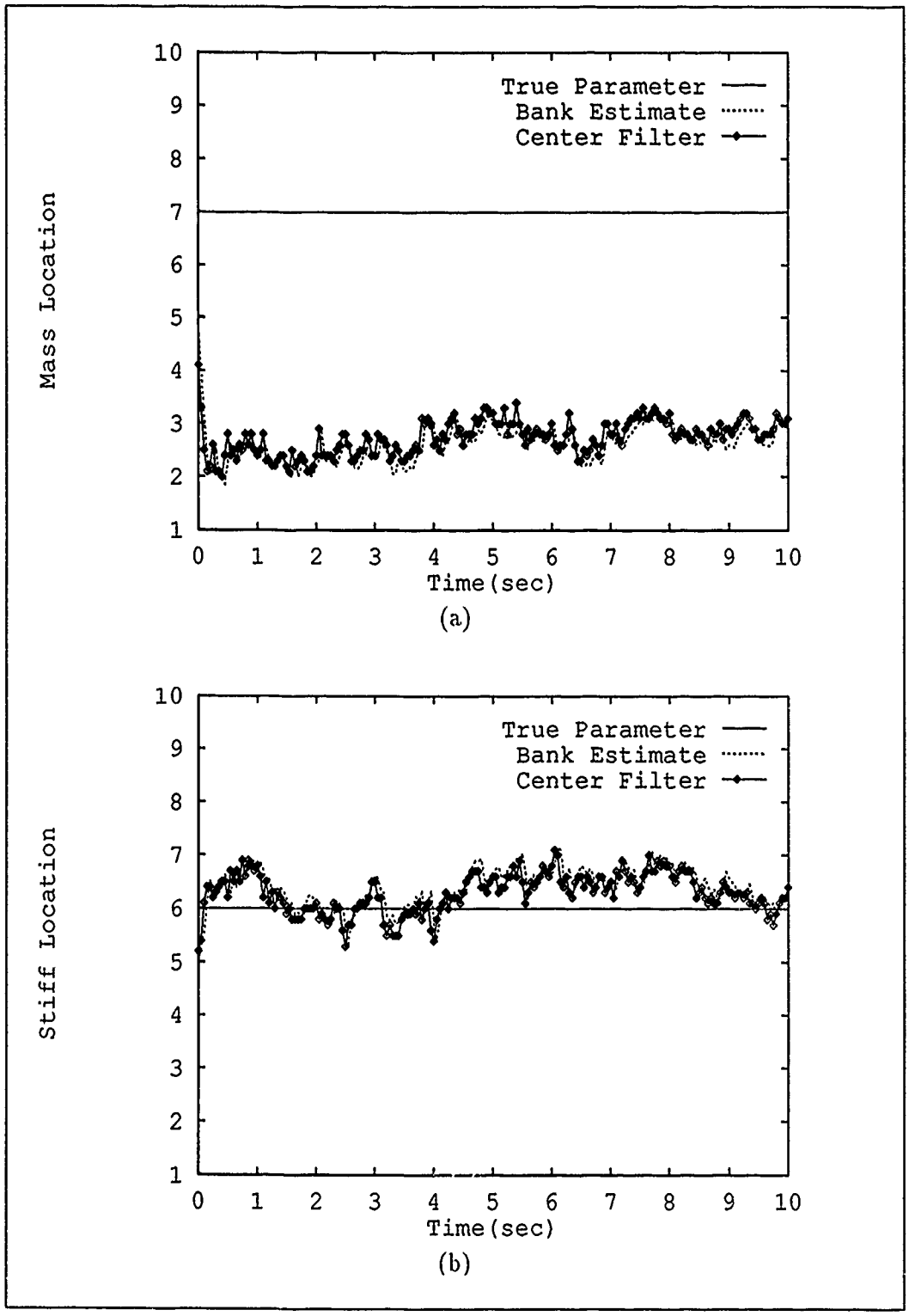

Figure 5.18. Moving-Bank Parameter Estimation (Sample Mean) and Actual Filter Bank Location, Residual Logic with $\mathbf{A}_k$ - True Parameter at (7,6) and Bank Initially at (5,5). (a) Mass and (b) Stiffness Parameters.

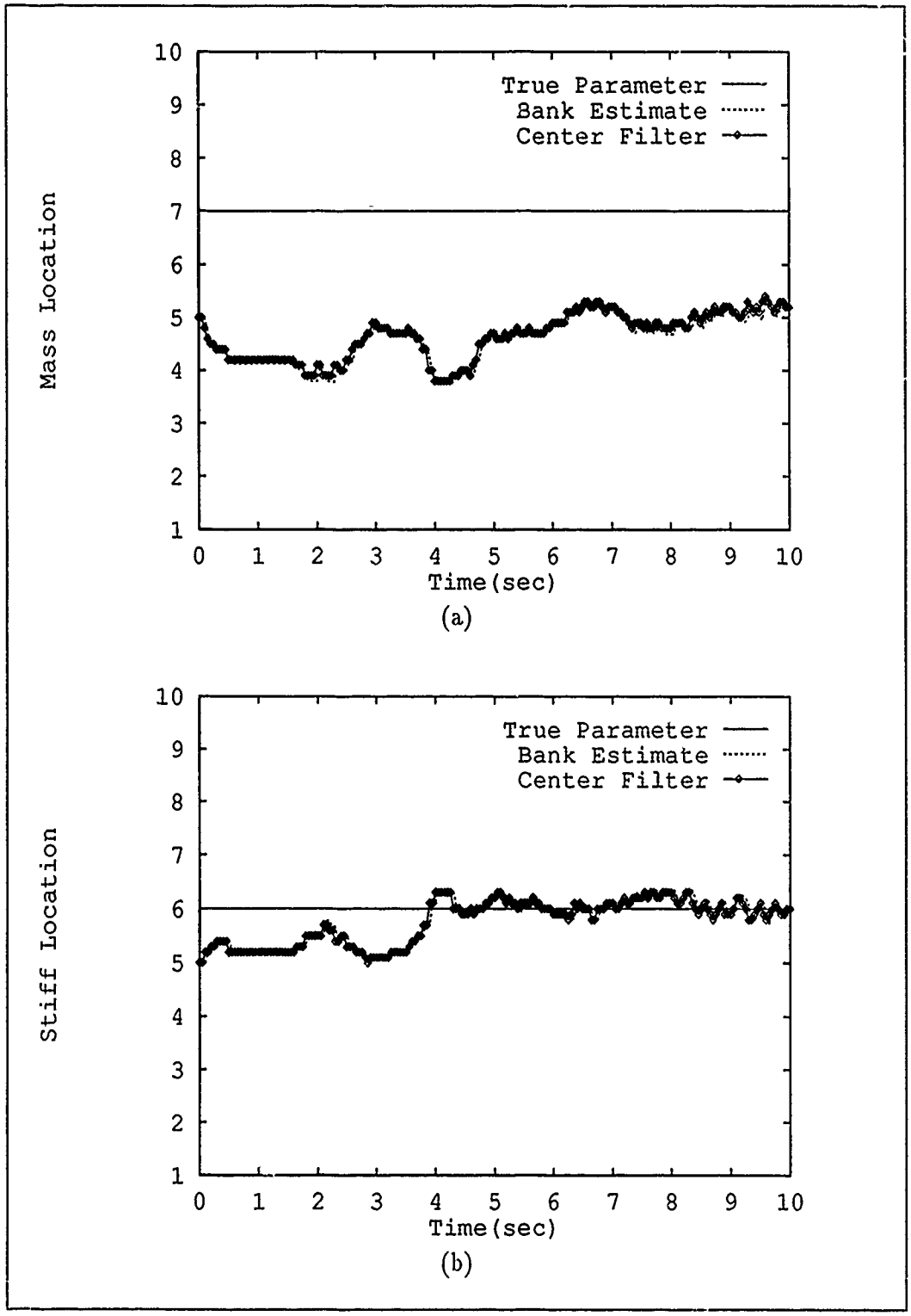

Figure 5.19. Moving-Bank Parameter Estimation (Sample Mean) and Actual Filter Bank Location , Residual Logic without $\mathbf{A}_k$ - True Parameter at (7,6) and Bank Initially at (5,5). (a) Mass and (b) Stiffness Parameters.

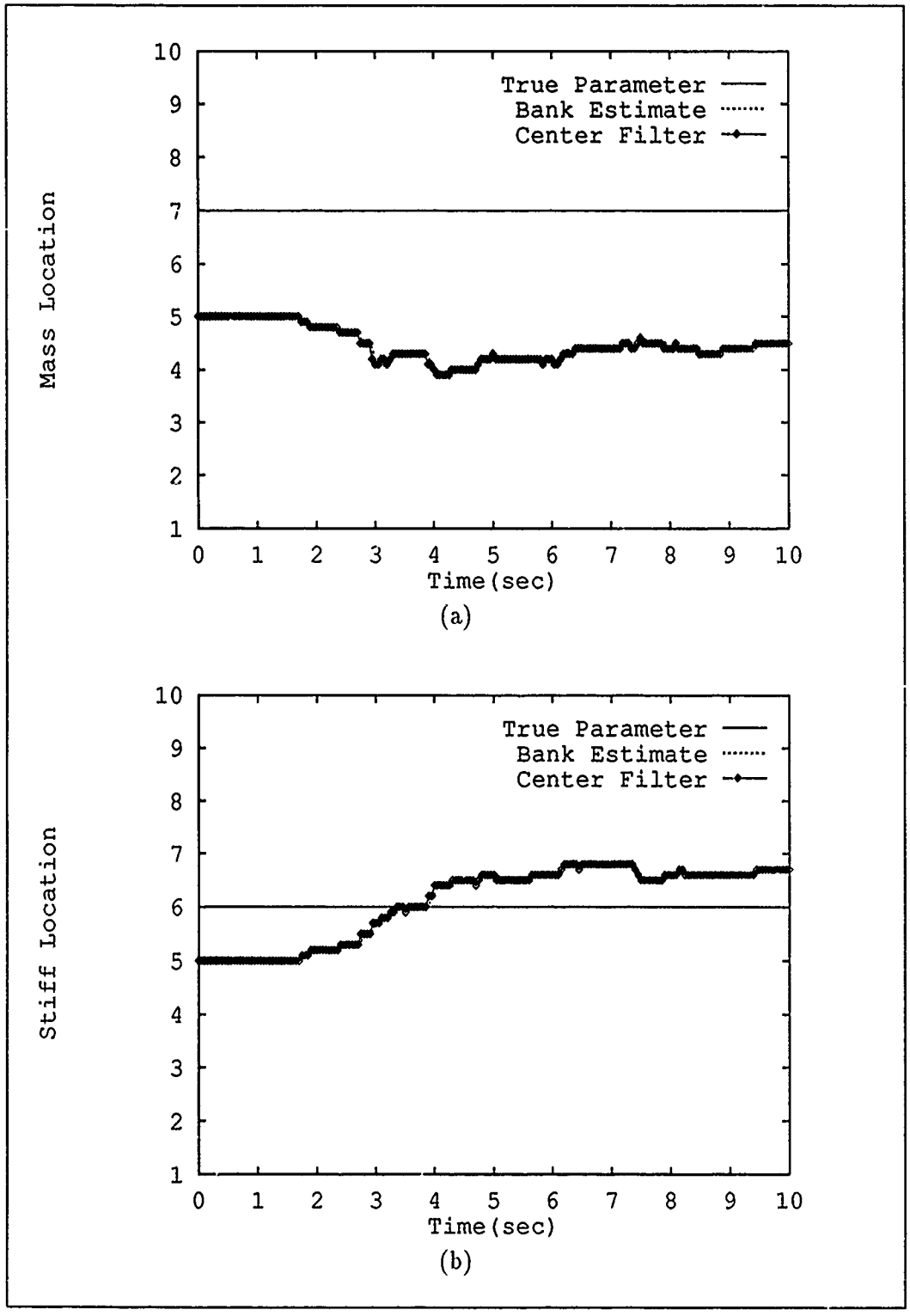

Figure 5.20. Moving-Bank Parameter Estimation (Sample Mean) and Actual Filter Bank Location, Probability Logic - True Parameter at (7,6) and Bank Initially at (5,5). (a) Mass and (b) Stiffness Parameters.

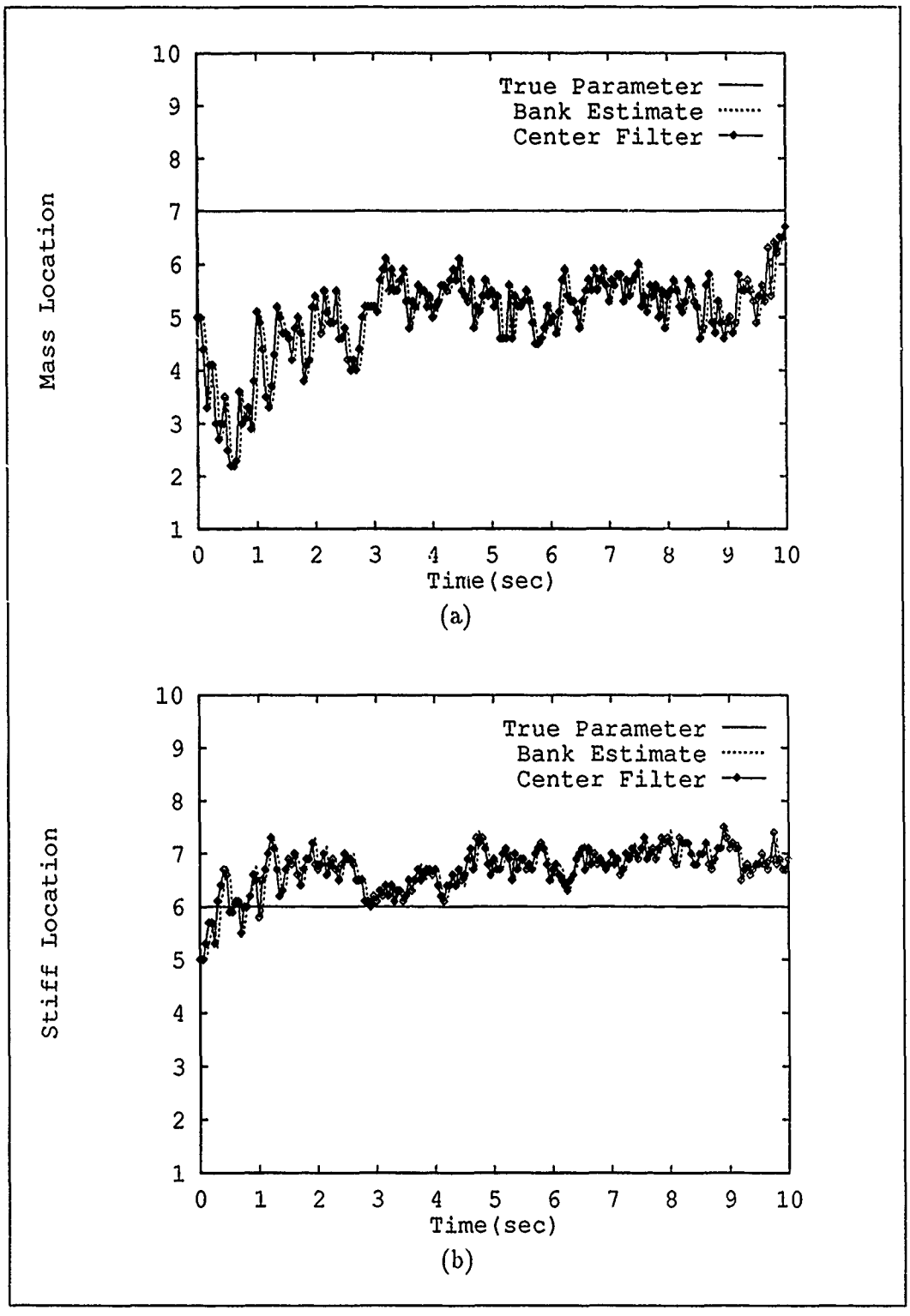

Figure 5.21. Moving-Bank Parameter Estimation (Sample Mean) and Actual Filter Bank Location, Parameter Logic - True Parameter at (7,6) and Bank Initially at (5,5). (a) Mass and (b) Stiffness Parameters.

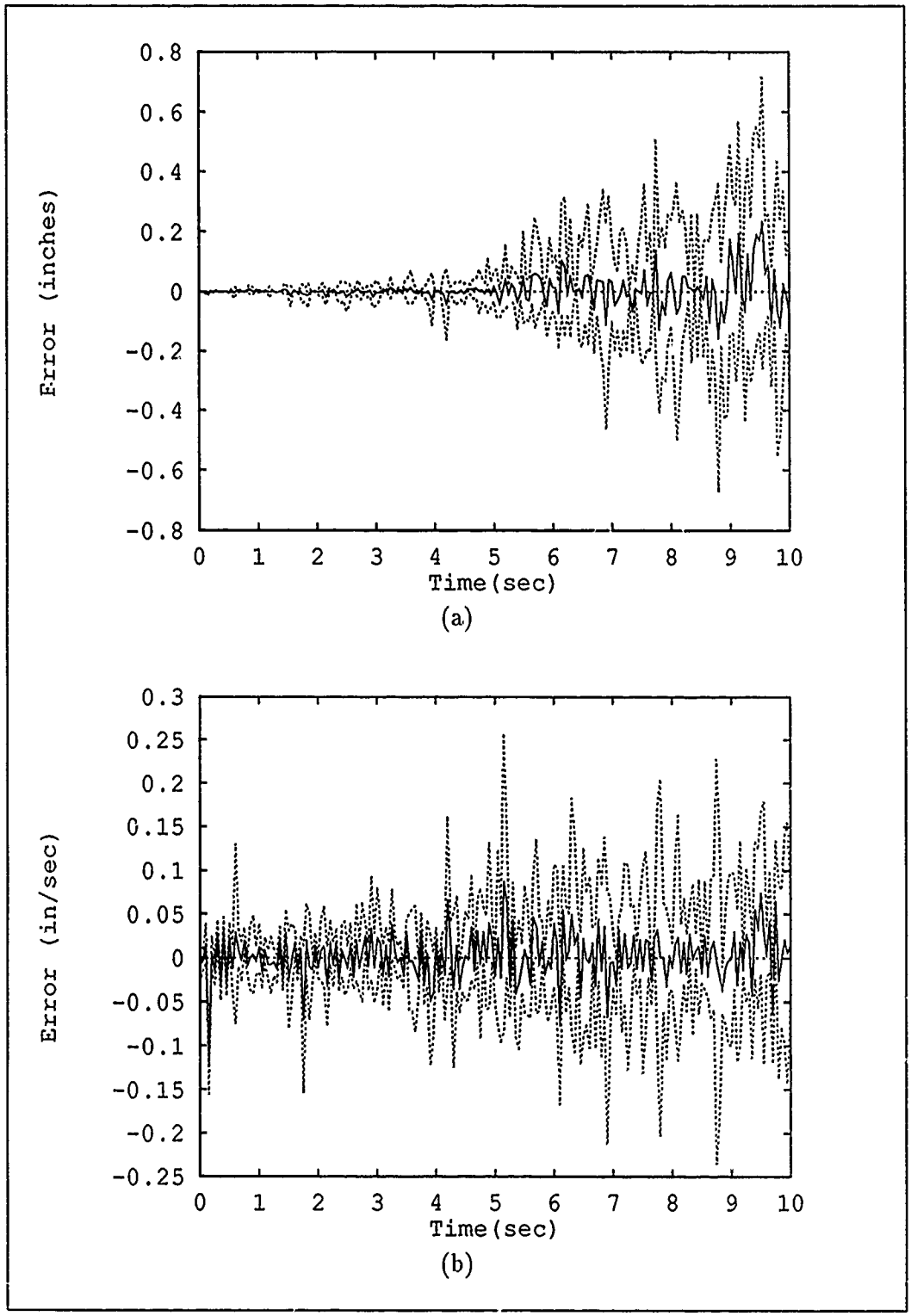

Figure 5.22. Moving-Bank Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter at (2,9) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors.

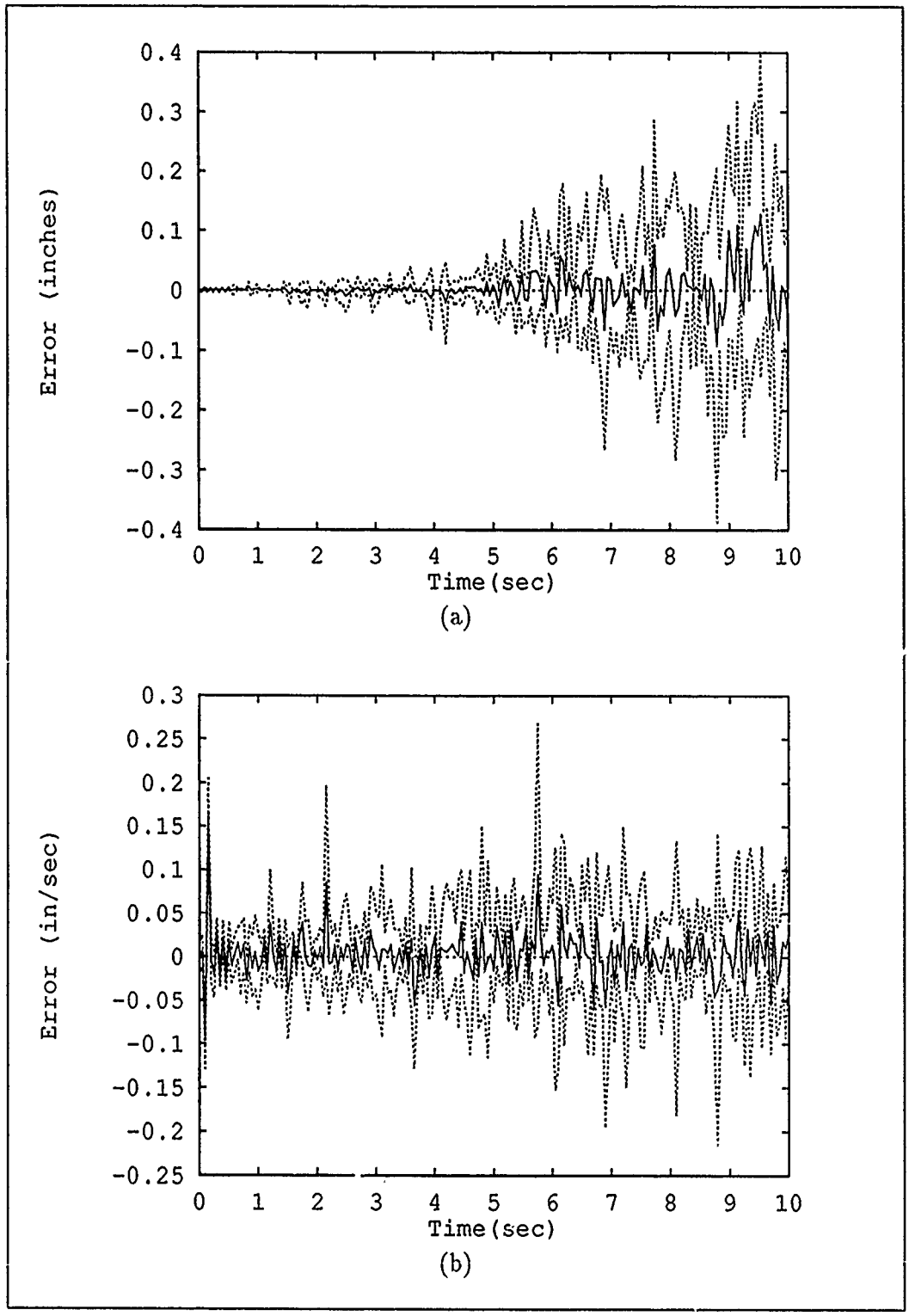

Figure 5.23. Moving-Bank Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter at (2,9) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors.

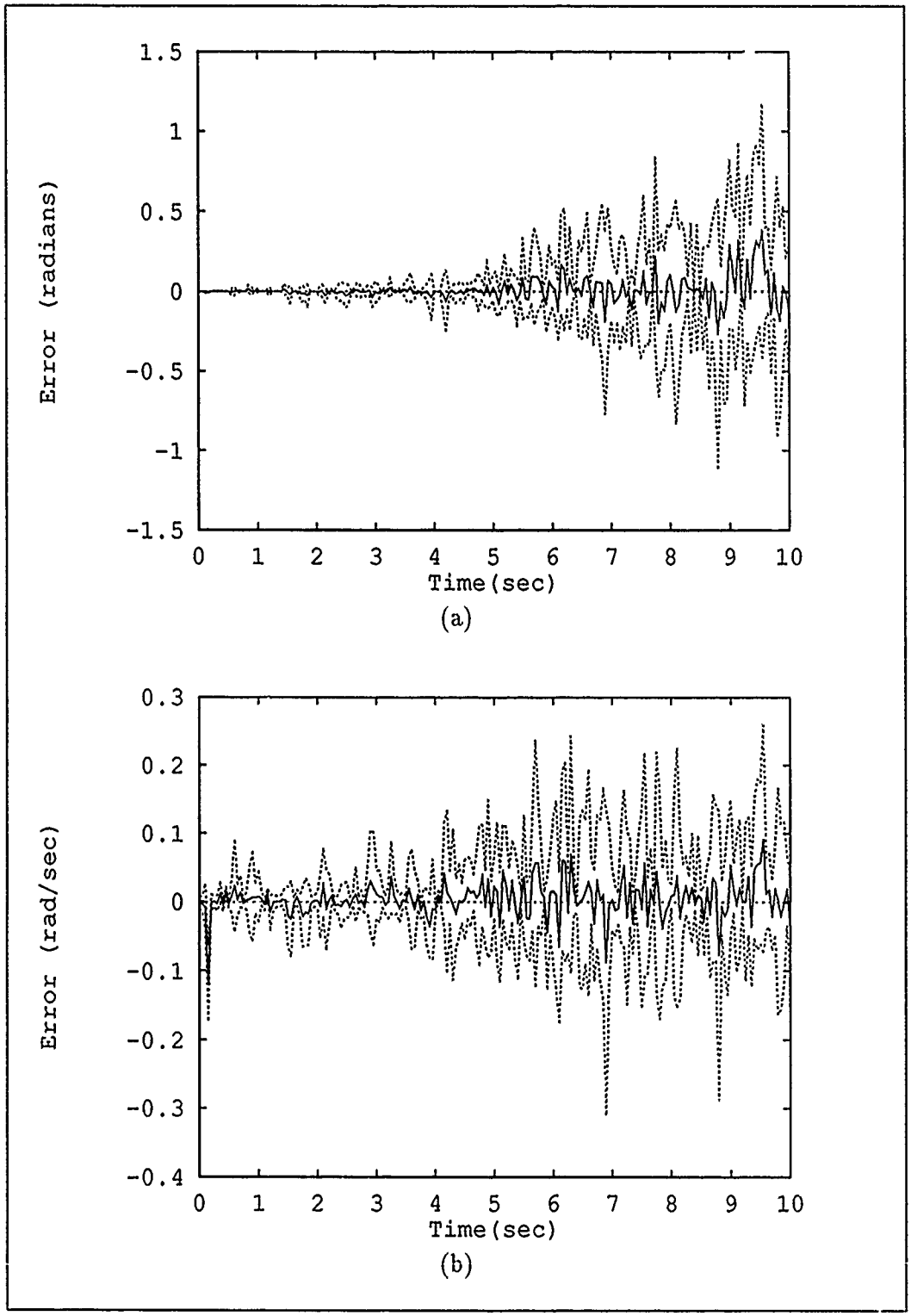

Figure 5.24. Moving-Bank Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter at (7,6) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors.

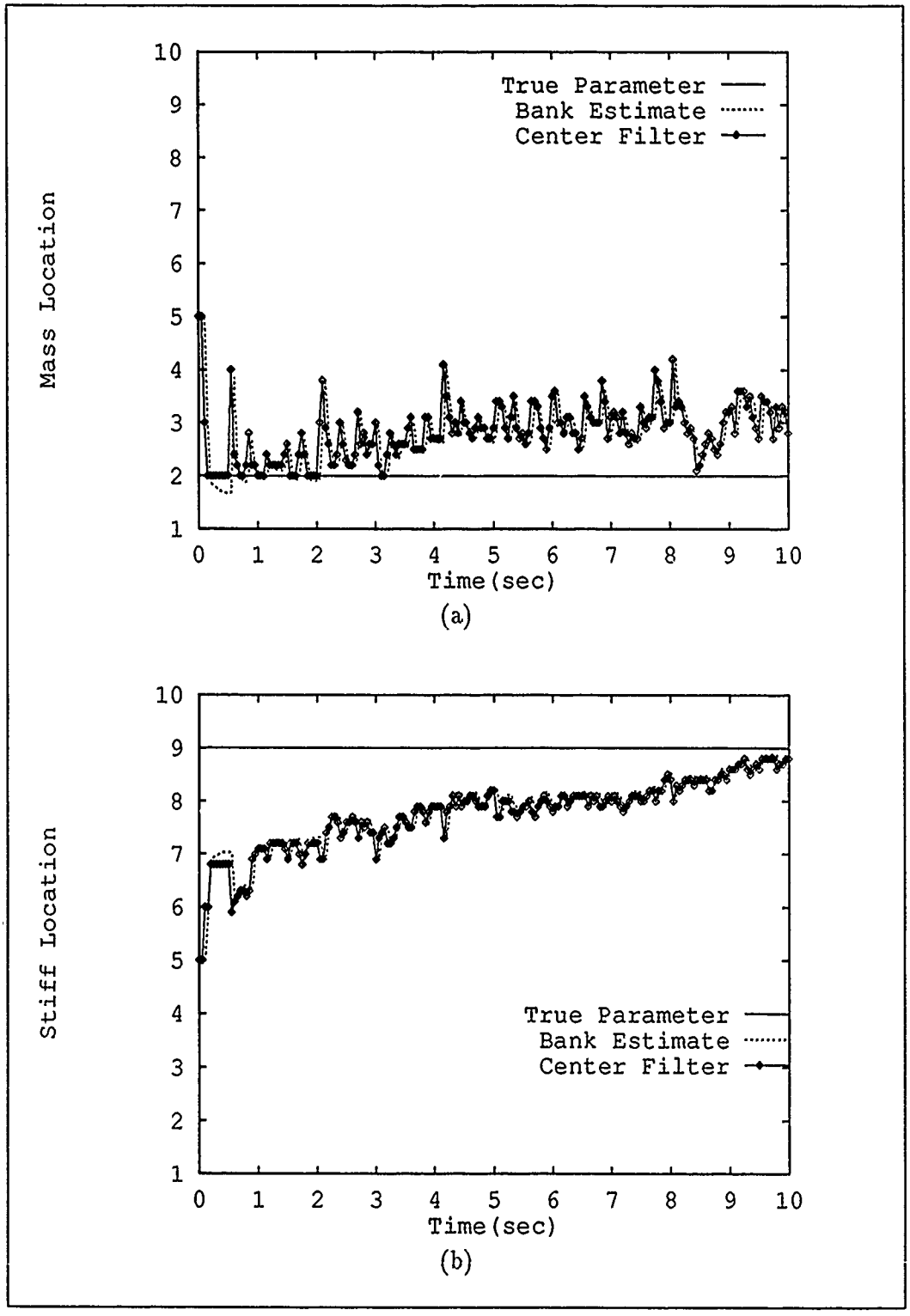

Figure 5.25. Moving-Bank Parameter Estimation (Sample Mean) and Actual Filter Bank Location - True Parameter at (2,9) and Bank Initially at (5,5). (a) Mass and (b) Stiffness Parameters.

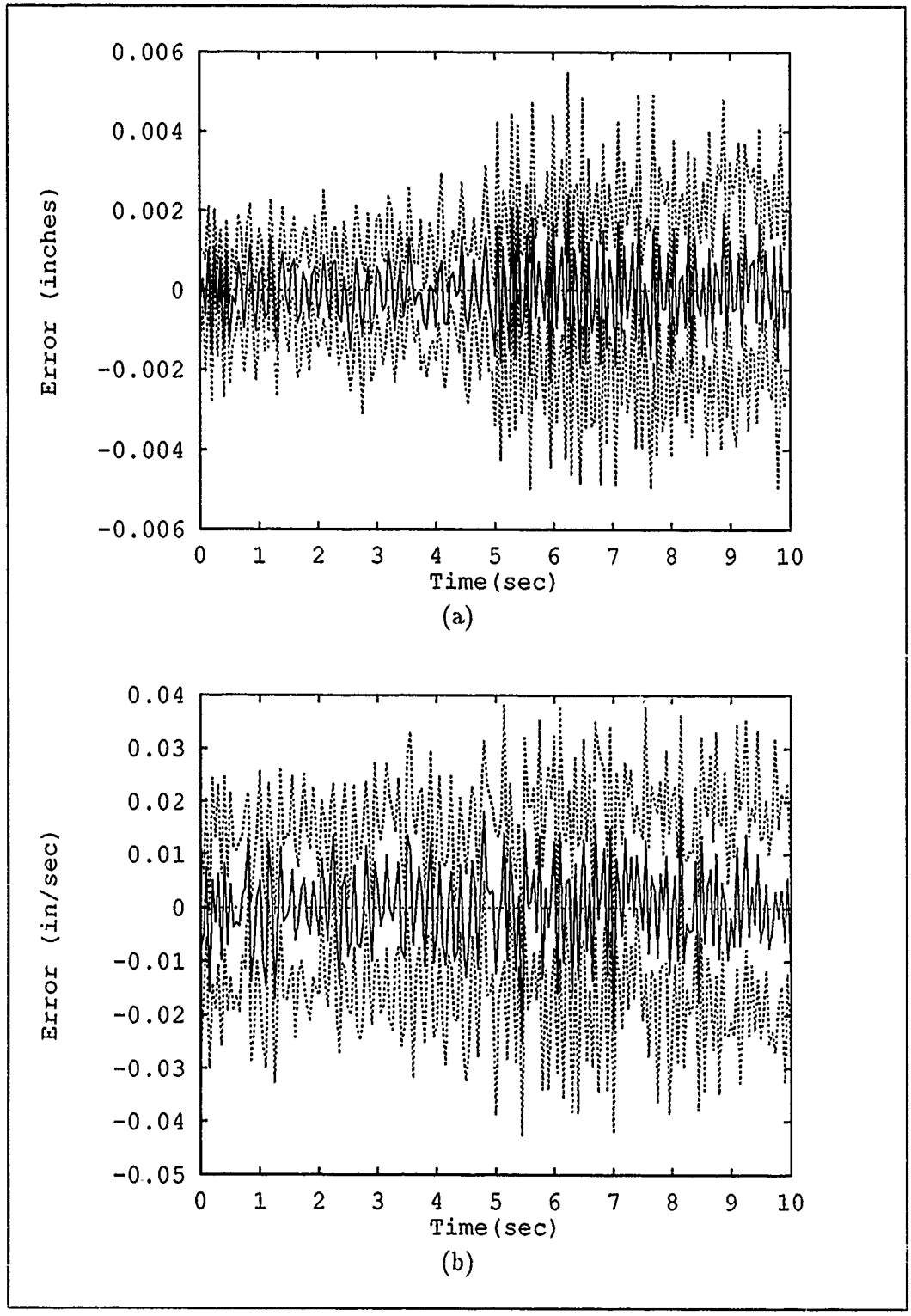

Figure 5.26. Single-Filter Benchmark Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter and Bank Initially at (5,5) with Parameter Jump to (2,9). (a) Position and (b) Velocity Estimation Errors.

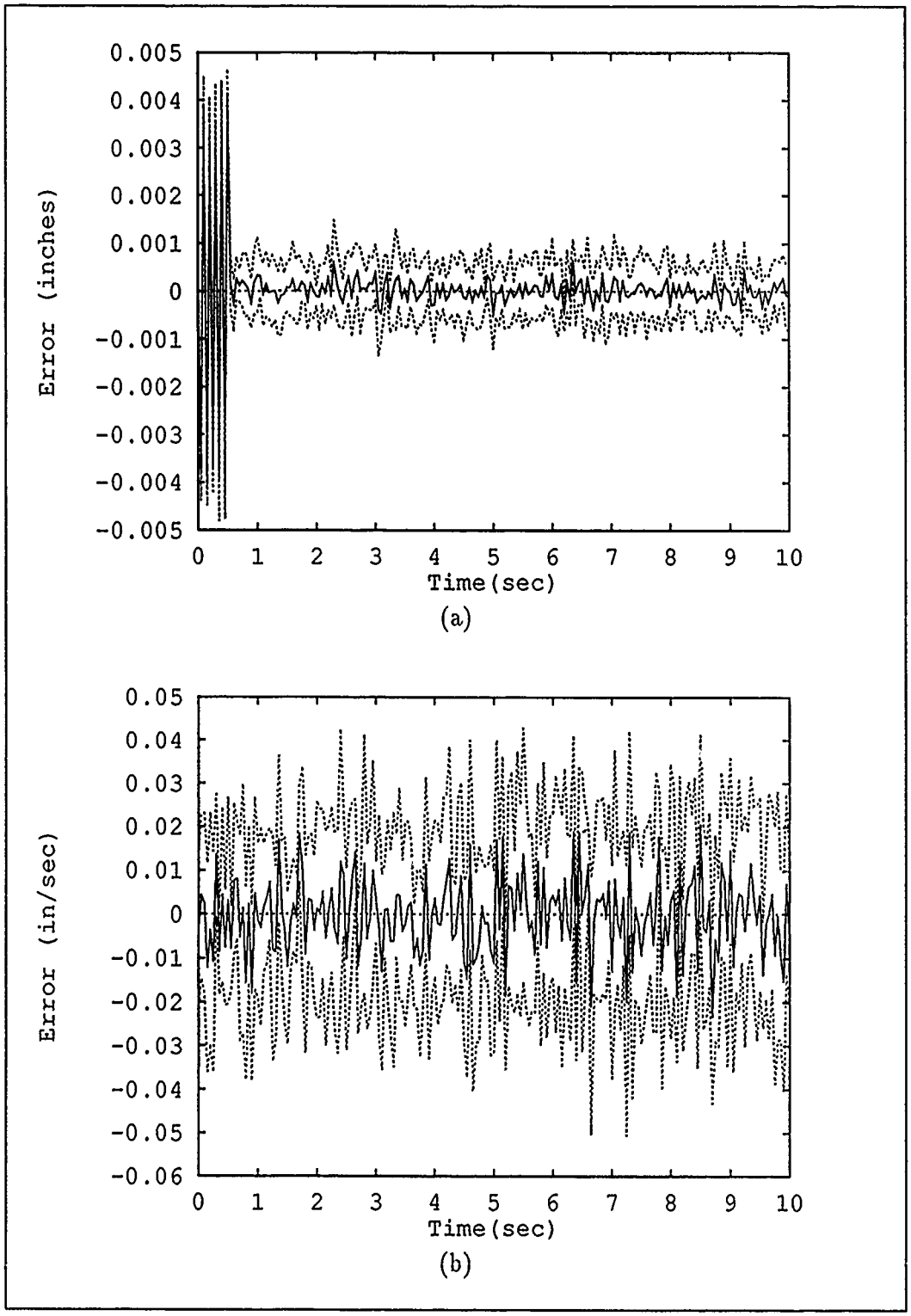

Figure 5.27. Single-Filter Benchmark Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter and Bank Initially at (5,5) with Parameter Jump to (2,9). (a) Position and (b) Velocity Estimation Errors.

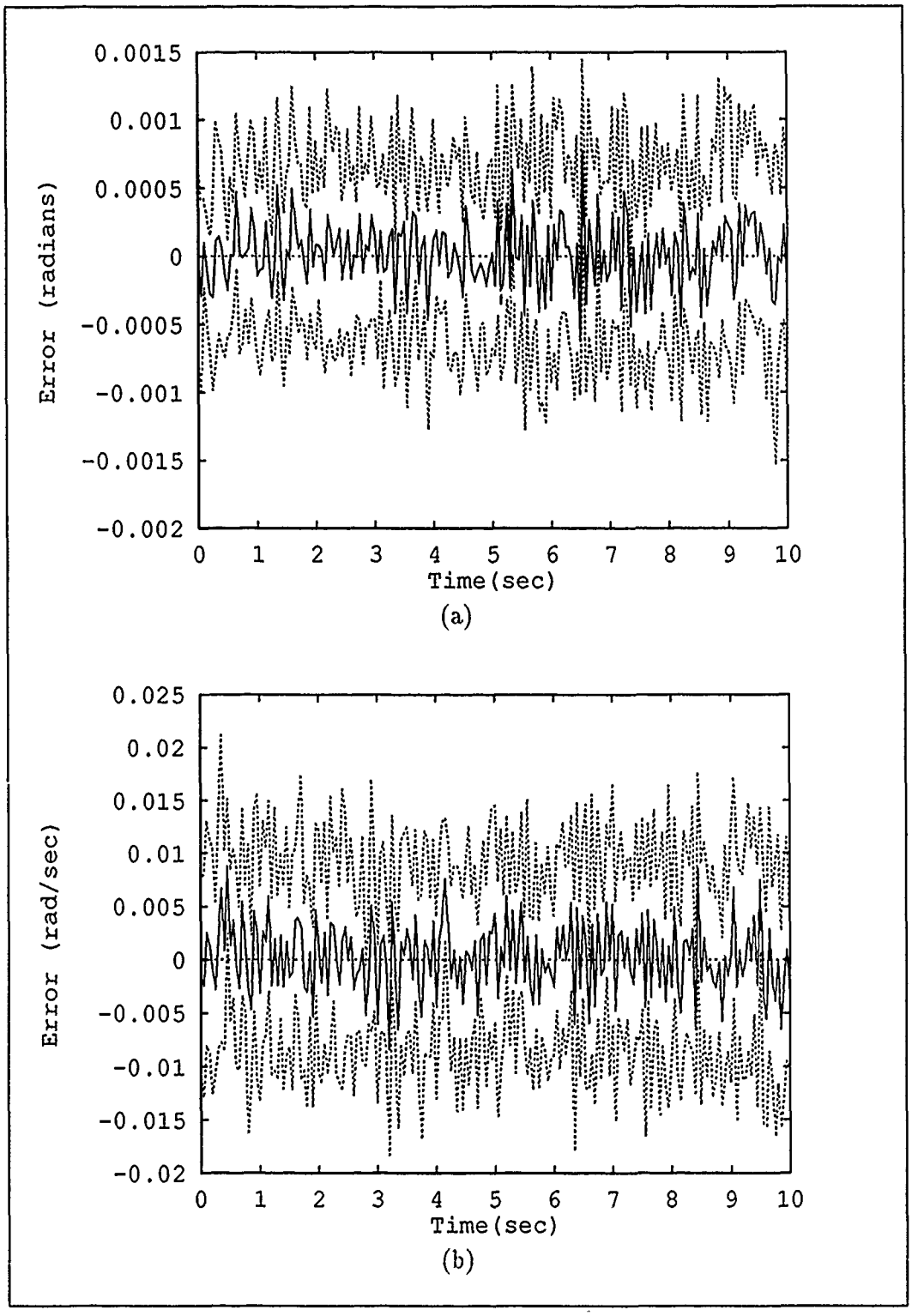

Figure 5.28. Single-Filter Benchmark Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter and Bank Initially at (5,5) with Parameter Jump to (2,9). (a) Position and (b) Velocity Estimation Errors.

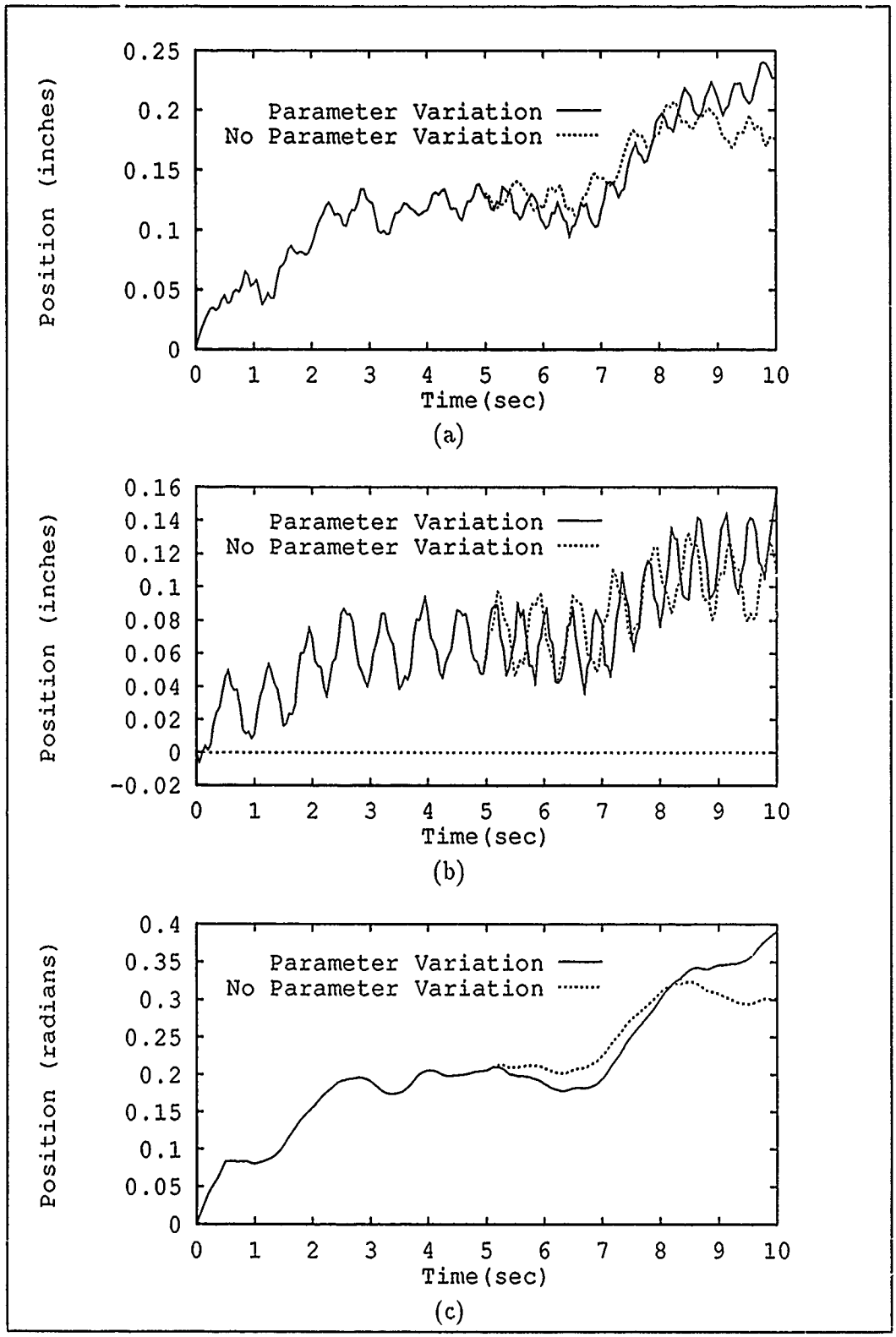

Figure 5.29. True Structure Positions (Sample Mean) for True Parameter Jump from (5,5) to (2,9) with and without Parameter Variations. (a) Node 1 (b) Node 2 and (c) Node 7 Positions.

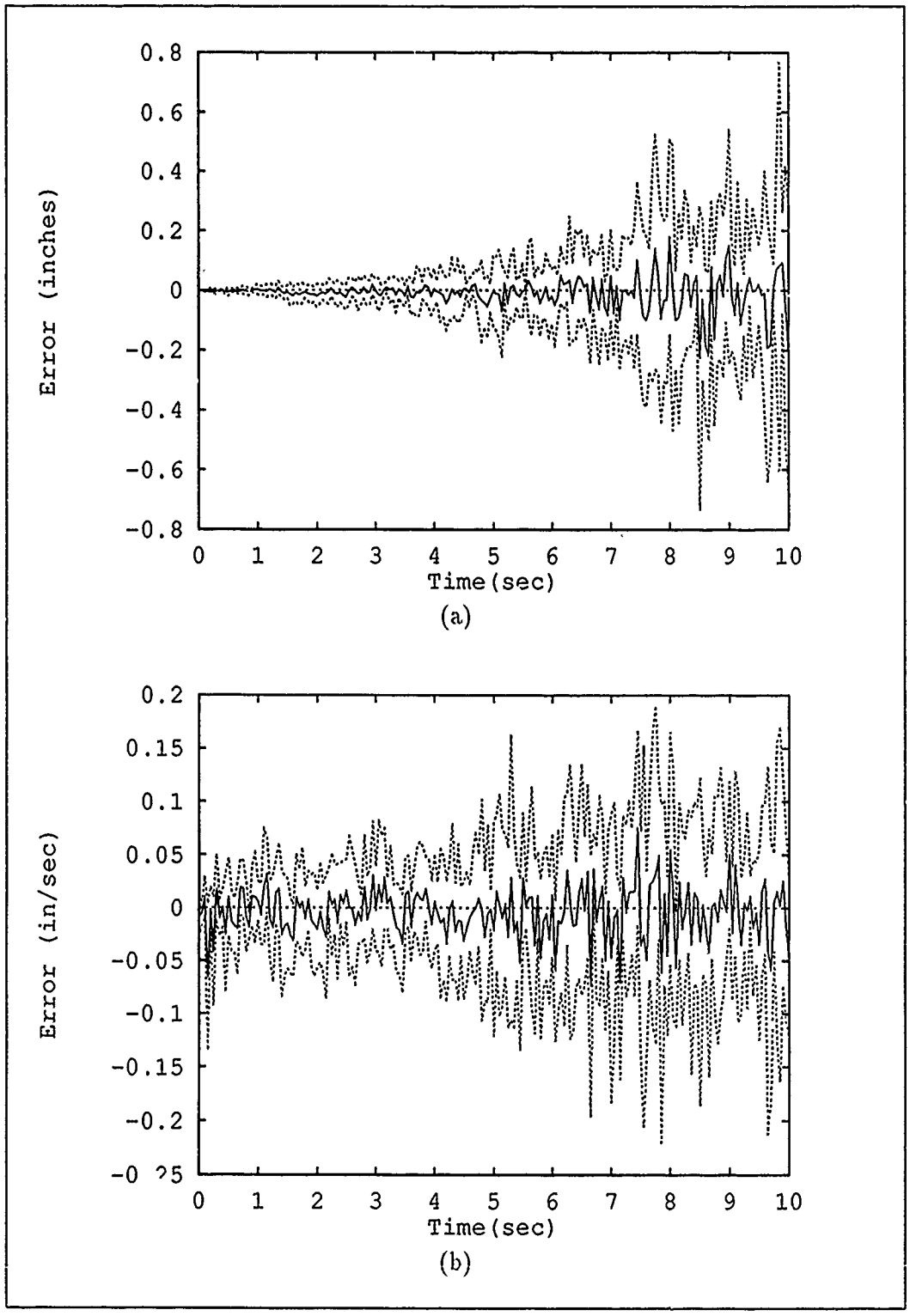

Figure 5.30. Moving-Bank Estimation Errors (Mean $\pm$ One Standard Deviation),Node 1 - True Parameter and Bank Initially at (5,5) with Parameter Jump to (2,9). (a) Position and (b) Velocity Estimation Errors.

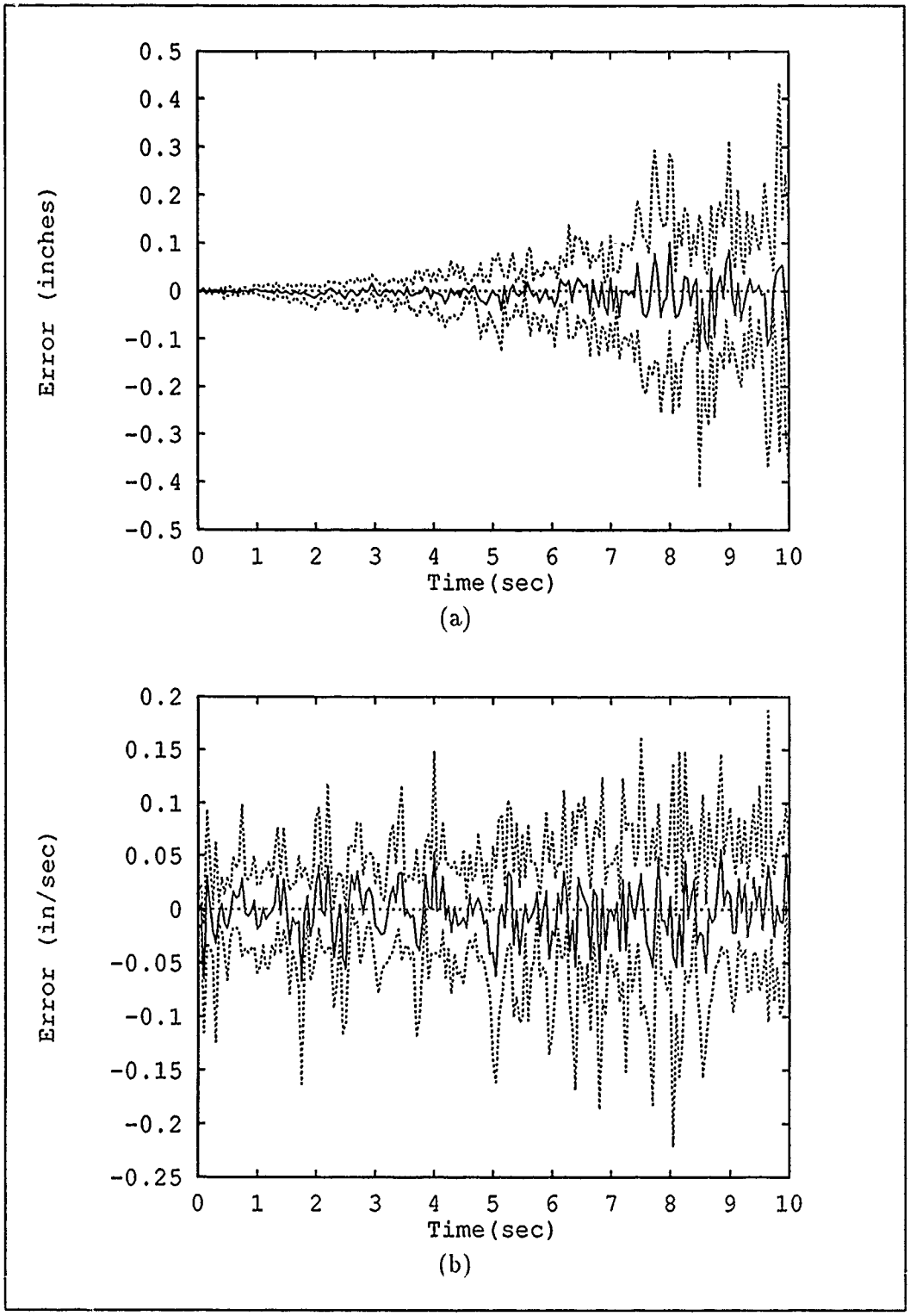

Figure 5.31. Moving-Bank Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter and Bank Initially at (5,5) with Parameter Jump to (2,9). (a) Position and (b) Velocity Estimation Errors.

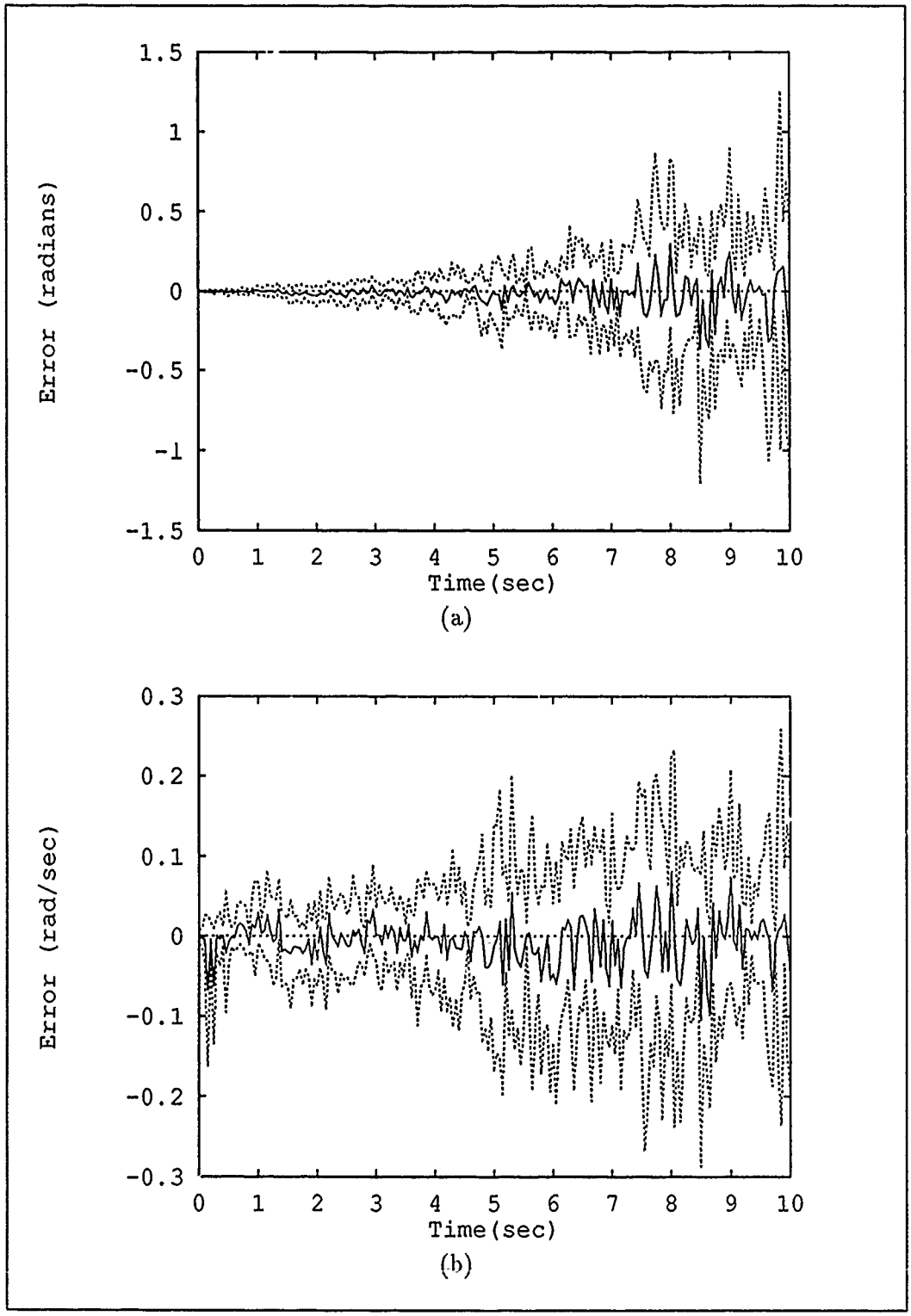

Figure 5.32. Moving-Bank Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter and Bank Initially at (5,5) with Parameter Jump to (2,9). (a) Position and (b) Velocity Estimation Errors.

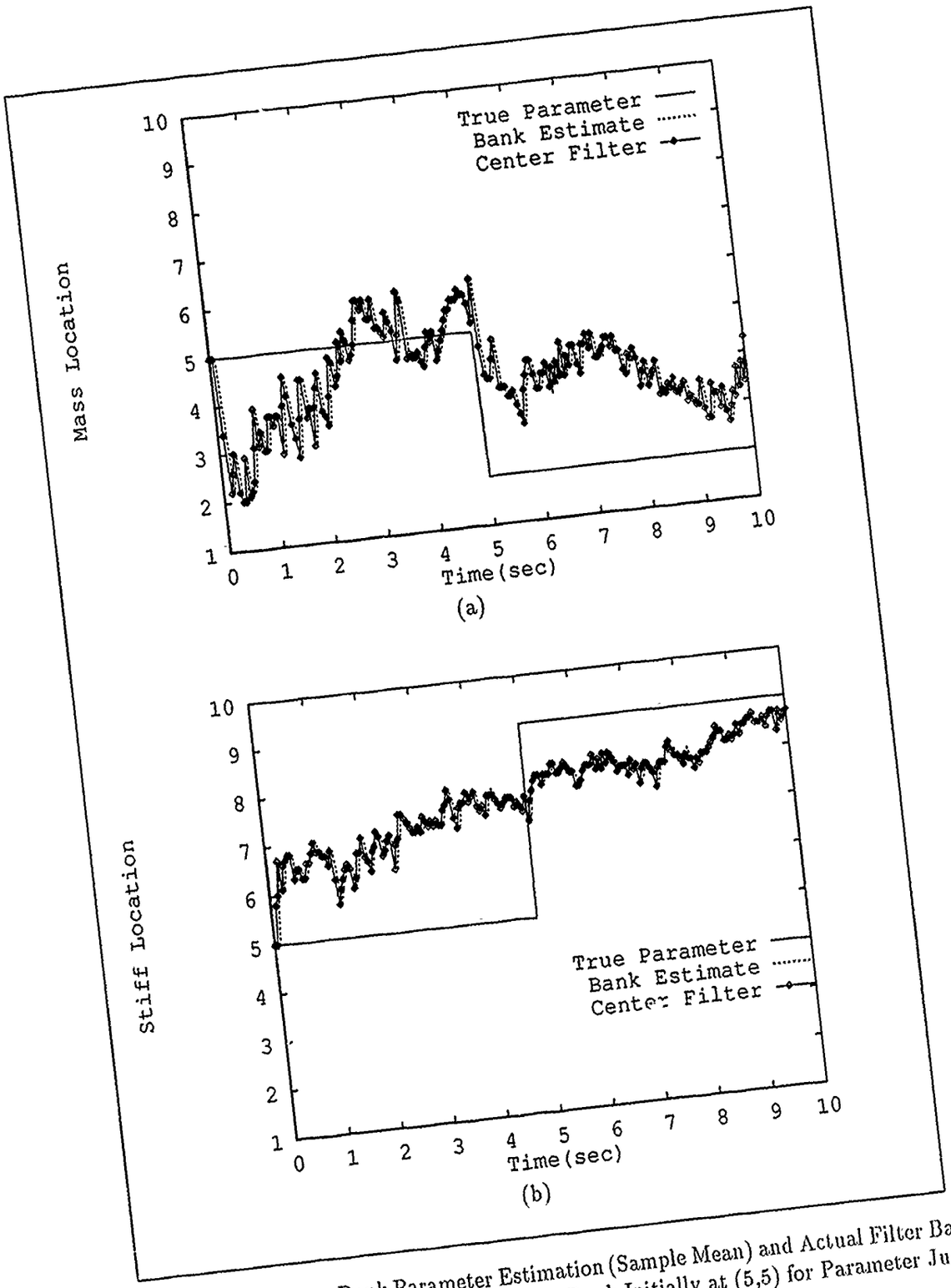

Figure 5.33. Moving-Bank Parameter Estimation (Sample Mean) and Actual Filter Bank Location - True Parameter and Bank Initially at (5,5) for Parameter Jump to (2,9). (a) Mass and (b) Stiffness Parameters.

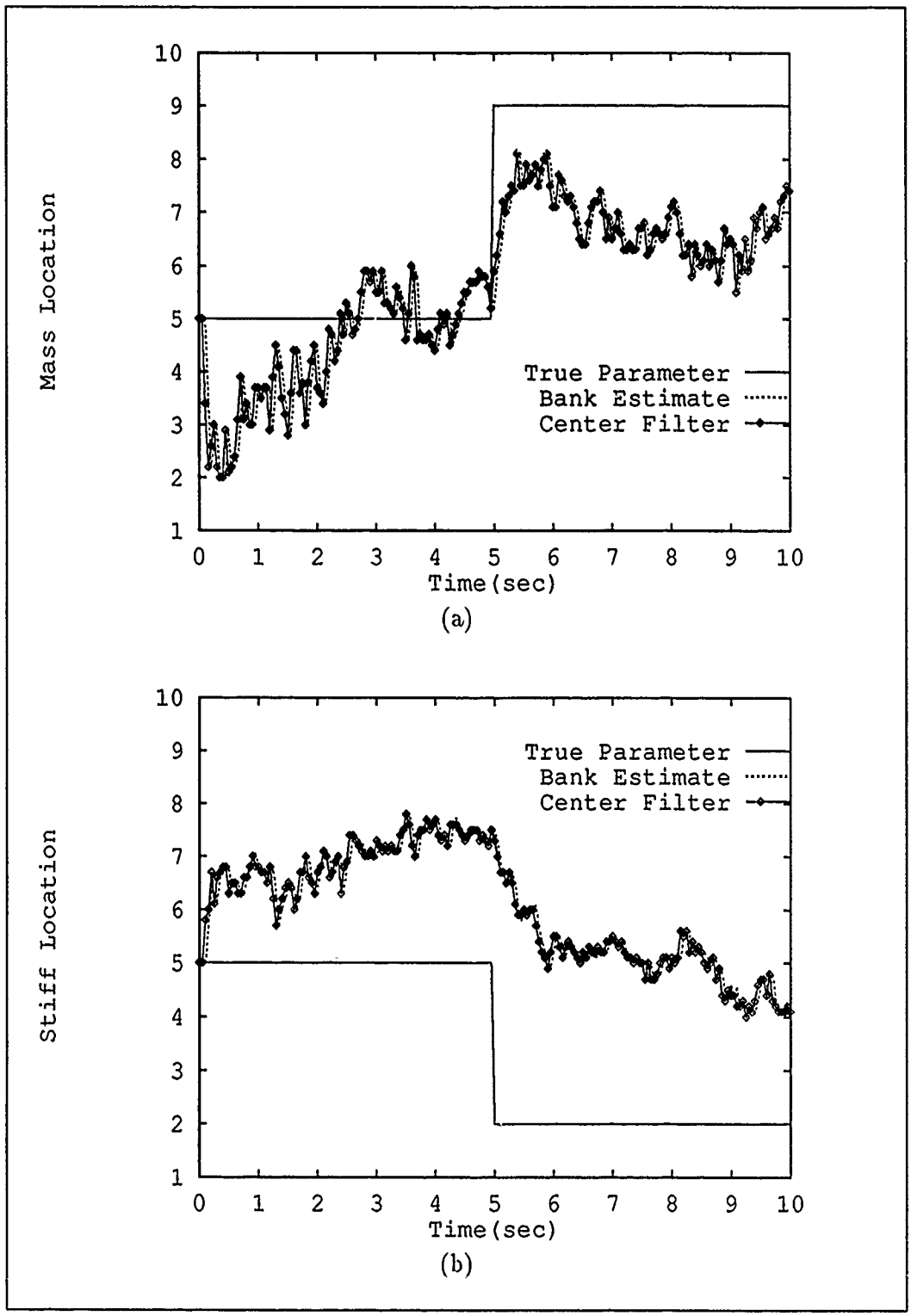

Figure 5.34. Moving-Bank Parameter Estimation (Sample Mean) and Actual Filter Bank Location - True Parameter and Bank Initially at (5,5) for Parameter Jump to (9,2). (a) Mass and (b) Stiffness Parameters.

5.3 *Multiple Model Adaptive Controller Study*

This section discusses results from the studies which investigated control performance of the moving-bank MMAC algorithms. These studies are as follows: 1) duplication of past research; 2) modifications to software; 3) state and control weighting matrix determination; 4) optimal control vector formulation; 5) parameter space discretization; and 6) varying parameter controller performance. The plots discussed in this section are provided after Section 5.4.

5.3.1 Duplication of Past Controller Research. Duplication methodology for the the LQG controller algorithms is similar to the duplications previously performed for the MMAE studies. All duplications use $\mathbf{Q}$ and $\mathbf{R}$ matrices for filter tuning, as well as $\mathbf{X}$ and $\mathbf{U}$ matrices for LQG controller cost definition, used by Schore [22]. Appendix B provides the values for $\mathbf{Q}$ and $\mathbf{R}$ whereas Appendix C provides the values for $\mathbf{X}$ and $\mathbf{U}$ used in the duplication. The duplication simulations consist of the MMAC benchmark and the moving-bank MMAC. The MMAC benchmark is equivalent to a non-adaptive, single LQG controller with artificial knowledge of the true parameter. For this study, the center filter/controller and true parameter location are set to parameter (7,6). The probability of the center filter/controller is set equal to one and the bank is not allowed to move. The estimation error results of the duplication for λ equal to one are shown in Figures 5.35 through 5.37. The mean true positions for the structure are provided in Figure 5.38. The control inputs applied to the system are also of interest and are presented in Figure 5.39. The results provided here correspond very well with the estimation errors, true positions, and actuator responses obtained by Schore [22].

As with the MMAE duplication, the second duplication analysis of the MMAC algorithm determines the ability of the bank to move within the filter space when the initial parameter estimate and filter location differ from the true parameter location. The duplication is generated by setting the center filter/controller of the bank to the (5,5) parameter location while the true parameter is located at parameter (7,6). The probabilities of all the filter/controllers in the bank are initialized to $\frac{1}{9}$ and the lower limit of the probabilities, p_{min}, is set to 0.05. Residual monitoring is used for the movement logic and the move

threshold set to 0.25 [22]. The results of this duplication for the estimation errors, true positions, and actuator responses are shown in Figures 5.40 through 5.42, Figure 5.43, and Figure 5.44, respectively. As with the single filter/controller duplication previously performed, these results compare very well with the results obtained by Schore [22].

There are several items to note from these duplication results. The first item to note from these results is the form of the dither signal applied to the structure. The dither signal is apparent from the control input shown for nodes 2 and 7 in Figure 5.39. Also, since the control input matrix was not correctly formed for the unmodified software, actuator 3 does not apply control to the structure after the dither signal is turned off. The actuators at nodes 1 and 2, therefore, expend considerable amounts of control energy, approximately 66 lbs of thrust for actuator 1 and approximately 33 lbs of thrust for actuator 2, to quell the oscillations of the structure as quickly as possible. Comparing the true structure positions shown in Figure 5.38 to the uncontrolled structure positions discussed earlier in Section 5.2.3.3 and provided in Figure 5.17, the control algorithm provides stable control of the structure. However, the large control outputs impact the estimation errors for node 1 and 2 velocities by creating large transient effects, as shown in Figures 5.35 and 5.36(note the effect of dither for the first half second in Figure 5.36(b) followed by the onset of large magnitude control with a tapering transient; this is highly correlated with Figure 5.39(b)). It will be shown that, with the correct form of the control input matrix, the actuator 1 and 2 outputs will be substantially reduced with corresponding improved performance in the estimation errors. Note that the rigid body position in Figure 5.38 is adequately controlled without the use of actuator 3.

5.3.2 Controller Modification Study. Section 5.2.2 described the results of the major modifications for the MMAE simulations. This section describes the affects of the modifications on the MMAC simulations. The major investigation is the modification of the control input matrix, $\mathbf{B}_r$, to allow control from the hub to affect the structure. The $\mathbf{B}_r$ modification, combined with the modifications discussed previously, caused numerical instability problems such that the simulation could not be performed before the appropriate values of $\mathbf{Q}$ and $\mathbf{R}$ were determined. After determination of these values, as described

in Section 5.2.3 and provided in Appendix B, simulations to investigate the modifications for the non-adaptive, artificially informed single LQG controller were performed. For this case, the bank and true parameter are set to parameter (7,6) and the bank is not allowed to move. Figures 5.45 through 5.47 provide the estimation error results. The steady state estimation errors for velocities at nodes 1 and 2 seem to be reduced by an order of magnitude compared to Figures 5.35 and 5.36. Figure 5.48 shows that the true positions are basically unchanged compared to Figure 5.38. However, note the control inputs from all three actuators are significantly affected, as shown in Figure 5.49, which are directly comparable to Figures 5.39. The initial control energy expended by the actuators located on the truss has been dramatically reduced; actuator 1 now expends approximately 12.5 lbs of thrust whereas actuator 2 expends approximately 9.5 lbs of thrust. Actuator 3 is now operating effectively after the dither is removed. The control modifications have resulted in less control being required from each of the truss actuators while providing the same control authority over the structure. This could have significant implications for determining the capabilities required of the actuators. While the increased actuator outputs from actuators 1 and 2 demonstrated in the duplication analysis do not indicate excessive actuator thrusting, the corrected simulations indicate thrusters with much less capability can be used. As for the MMAE modification study, no modification analysis was performed for the moving-bank MMAC algorithm. The study performed in Section 5.3.3.3 will investigate the performance of the moving-bank MMAC using the move logic established for the estimation studies.

5.3.3 State and Control Weighting Determination. Determination of the appropriate state and control weighting matrices for the LQG controllers is similar to the determination of the dynamics noise strength and measurement noise covarince matrices for the Kalman filters. For this study, the values of $\mathbf{X}$ and $\mathbf{U}$ were iteratively increased until the rms errors of the corresponding states or controls stopped decreasing substantially. This method was performed for the (7,6) parameter and is the method used to determine appropriate tuning for the remainder of the controllers. Appendix C provides the $\mathbf{X}$ and $\mathbf{U}$ matrices which were found previously for the (7,6) parameter location [11].

This study began with re-evaluation of the single controller performance for parameter location (7,6). Performance of this controller is unaffected by alterations to the weighting matrices as large as one order of magnitude, which is also true for the region of the discretized parameter space for the lower natural frequencies, i.e., values for M (mass) parameters greater than the seventh discrete location and values for the S (stiffness) parameter less than the sixth discrete location. An indication of how well the controller is tuned for the (7,6) parameter is provi: ed by the small true position magnitudes, which are on the same order of magnitude as the estimation errors. Following re-evaluation of the controller based on the (7,6) location, the controllers for the parameters in the high natural frequency range were investigated, i.e., values for M parameters less than the seventh discretized location and values for S parameters greater than the sixth discretized location. It was d' 'rmined that the controller performance in this region, which had previously used the **X** and **U** matrices determined for the (7,6) location, could be improved substantially. Several parameter points were selected in this region for direct investigation while the remaining points were interpolated from the results. For example, Figures 5.50 through 5.52 show the state estimation performance for the parameter location (1,9) using the **X** and **U** values for location (7,6), i.e., the "untuned" performance for this controller. The estimation errors for the positions and velocities demonstrate large mean errors. Figure 5.53 show the true positions of the structure are oscillating due to the effect of the relatively high natural frequencies. Figure 5.54 shows the associated actuator outputs. In order to tune this controller, the values of **U** were held constant while the values of **X** were increased. Then the values of **X** were held constant while the values of **U** were increased. (The values of **U** were increased to avoid excessive control magnitudes). The appropriate weighting matrices were determined by scaling the values for Lashlee's matrices. The tuned controller for parameter location (1,9) is provided in Figures 5.55 through 5.59. These figures show that the oscillations previously characteristic of the estimation position errors are quelled to steady-state levels in approximately 2.0 seconds. More importantly, the true positions of the structure are effectively regulated to zero in the same time period without the expenditure of excessive control power. Especially note the improved control of the structure for node 2 position shown for the tuned controller in Figure 5.58 compared

to the untuned controller performance shown in Figure 5.53. Appendix C provides the parameter locations and the associated state and control weighting matrix scaling factors for the controllers tuned in this section.

5.3.3.1 Non-Adaptive Single-Controller Best-Case Analysis. The simulations performed for the controller modification study for parameter location (7,6) in Section 5.3.2 also represent the non-adaptive, single controller best-case benchmark. This is selected as the benchmark because it represents the nominal case and is in the center of the space. The controller for this location, as discussed in the previous section, did not require any additional tuning from that obtained from using the weighting matrices from previous research. The results are presented in Figures 5.45 through 5.49 and were discussed in Section 5.3.2.

5.3.3.2 Non-Adaptive Single-Controller Worst-Case Analysis. In order to establish a worst-case benchmark, a non-adaptive, single LQG controller with the incorrect parameter was investigated. For this study, the true paramater was located at (2,9) and the bank was fixed at several locations throughout the space. Results show that the fixed elemental filter/controller performs well within a boundary region close to the true parameter location. For non-adaptive controllers within the "boundary" locations, performance is adequate for stable control although only slightly degraded from the best-case benchmark. Once the single filter/controllers are fixed outside of this "boundary", instability problems prevent completion of the simulations. A worst-case "boundary" was determined such that a non-adaptive, single filter/controller outside of this "boundary " could not be evaluated. The boundary parameter points consist of the following: (4,1), (5,3), (6,6), (7,8), and (8,10).

5.3.3.3 MMAC Moving-Bank Analysis. As for the MMAE moving-bank analysis, three simulations were performed to investigate the performance of the tuned controller algorithms throughout the parameter space:

1. Case 1: True parameter at location (7,6) with bank initially at location (5,5)
2. Case 2: True parameter at location (2,9) with bank initially at location (5,5)

3. Case 3: True parameter at location (9,2) with bank initially at location (5,5)

All three cases above used parameter position estimate monitoring with a threshold of 0.01. Both the ME/I and complete density function algorithms were investigated. The results for the case 1 study using ME/I computations demonstrate that the algorithm provides stable control, i.e., the true positions are quelled to approximately zero in less than 2.0 seconds. The results for the proper density computations provide similar control of the structure, also quelling the true positions of the structure to zero within the same amount of time.

The results for case 2 using ME/I are shown in Figures 5.60 through 5.65. Note that the bank moves rather quickly to the correct region of the parameter space. In this case, parameter identification is enhanced since the bending modes are greatly excited by the (inappropriate) control based on incorrectly assumed parameter values. For this simulation, the actuators provide an initial transient after which the steady state estimation errors and true structure positions perform as well as the non-adaptive best-case benchmark. Comparing these results to the MMAE simulations (Figures 5.22 through 5.24), the estimation errors are quickly brought to steady state values. For case 2, the MMAC simulation using the proper density function could not be completed due to instability.

For the case 3 study using ME/I computations, the MMAC algorithm provides adequate control even though the bank parameter estimate (and resulting bank location) is not in close proximity to the true parameter (final parameter estimate (6.5,4.5)). Stable control is possible in this situation due to the fact that the bank estimation and movement maintained control based on controllers which overestimated the natural frequencies of the bending modes. The case 3 performance using the proper density function provides parameter estimation slightly worse than for the ME/I, however, the control of the structure is not degraded due to the overestimation of the bending mode frequencies. The top half of Tables 5.8 and 5.9 provides a summary of the MMAC performance for the three test cases using ME/I computations. Comparing these results to the corresponding MMAE simulation results shown in Tables 5.1 and 5.2 show that the position estimation errors are reduced by two orders of magnitude.

Table 5.8. MMAC Parameter Space Discretization Position Estimation Errors

Old Parameter Space Discretization				
Simulation	Node 1 Error(1)		Node 2 Error(1)	
⇓	Mean	1σ	Mean	1σ
Case 1	0.0000	0.0014	0.0000	0.0009
Case 2	0.0000	0.0025	0.0000	0.0011
Case 3	0.0001	0.0014	0.0000	0.0009
New Parameter Space Discretization				
Case 1	-.0002	0.0031	0.0001	0.0011
Case 2	-.0001	0.0024	0.0000	0.0009
Case 3	-.0001	0.0014	0.0000	0.0010
Note (1): Average values for $t > 5.0$ seconds (inches)				

Table 5.9. MMAC Parameter Space Discretization True Positions

Old Parameter Space Discretization				
Simulation	Node 1 True(1)		Node 2 True(1)	
⇓	Mean	1σ	Mean	1σ
Case 1	0.0001	0.0012	-.0004	0.0014
Case 2	0.0019	0.0028	-.0003	0.0013
Case 3	-.0007	0.0016	-.0002	0.0011
New Parameter Space Discretization				
Case 1	0.0007	0.0015	0.0002	0.0012
Case 2	0.0009	0.0028	-.0001	0.0013
Case 3	-.0004	0.0013	-.0003	0.0011
Note (1): Average values for $t > 5.0$ seconds (inches)				

Finally, an additional simulation was performed to investigate the performance of the MMAC moving-bank algorithm when the bank is initially located beyond the boundary of stable control as discussed in Section 5.3.3.2. From the results presented in Section 5.3.3.2, it was determined that a non-adaptive filter/controller at location (7,6) with the true parameter at location (2,9) was beyond the bounds of stable controller algorithm performance. However, when the bank is allowed to move from this location, very good estimation error performance using ME/I computation results with very rapid acquisition of the true parameter location. These results are characteristic of the performance obtained for the case 2 study.

5.3.4 Optimal Control Vector Formulation Study. The goal of this study was to determine if moving-bank modified MMAC, MAP vs. Bayesian, or modified single-fixed gain control formulation could provide better control than the conventional moving-bank MMAC method. The primary basis for comparison are the true positions, $\mathbf{H}_l \underline{\mathbf{x}}_l(l_i)$, since this determines how effective the controller is in quelling the bending modes and pointing of the structure. Estimation errors and control input magnitudes also provide a basis for performance comparison. For each of the control options investigated, the estimator uses parameter position estimate monitoring and a threshold of 0.01. All three test cases described in Section 5.3.3.3 are investigated. In addition, both the ME/I and the proper density function are considered.

5.3.4.1 Modified MMAC Analysis. For this study, a lower bound to be surpassed in order for control to be included in the modified MMAC computations was determined to be p_{mmac}=0.10. Recall the lower probability bound, p_{min}, was set to 0.05. For the case 1 study using ME/I computations, no significant difference is obtained from the standard MMAC method in the previous section. The structure positions for nodes 1 and 2 are quelled to zero in approximately the same time. The proper density function provides similar stable control results.

Figures 5.66 through 5.71 show the performance for the case 2 study using ME/I computations. The settling time of the true positions for nodes 1 and 2 in Figure 5.69 are improved when compared to the standard MMAC results, shown in Figure 5.63. The

settling of the true position for nodes 1 and 2 is reduced from 2.0 seconds to less than 1.0 second. This indicates that the controllers with a substantial probability (versus any nonzero probability; recall that lower bounding is used) provide appropriate control. This more appropriate control also affects computation of the residuals and allows better parameter estimation for both ME/I and the proper density function computations. Figure 5.71 shows quicker good stiffness parameter estimation when compared to the conventional parameter estimation performance shown in Figure 5.65. The case 2 study using the proper density function provided a similar improvement from the standard MMAC using ME/I (whereas the proper density computations for the standard MMAC for case 2 results in instability).

The results for case 3 are not much different from those for the MMAC analysis; stable control is obtained within 2.0 seconds. Using the proper density function, stable control is obtained similar to that for the standard MMAC.

An additional simulation was performed to determine if the parameter estimation (and therefore structure control) could be further improved if the same probability threshold, p_{mmac}, is used for computing $\hat{\mathbf{a}}_k$. This investigation was performed for case 2 using ME/I computations. The resulting parameter estimation performance is severely degraded. The final parameter estimation is (1.6,1.4). The resulting structure control is degraded by an increased settling time (approximately 3.6 seconds) and increased actuator outputs for this implementation.

5.3.4.2 MAP vs Bayesian Analysis. This approach uses the moving-bank elemental filter/controller associated with highest computed conditional probability rather than generating a probability-weighted average. For all three test cases, the performance of the controller in quelling the truss positions is similar to that obtained for the corresponding standard MMAC results. The use of the proper density function also provides similar results (the case 2 study results in instability, as is the case for the standard MMAC).

5.3.4.3 Modified Single Fixed-Gain Analysis. This method was described in Section 1.1.3 and illustrated in Figure 1.7. Using ME/I computations, the control results

for the three test cases are not significantly different from the standard MMAC method (the structure node 1 and 2 positions are quelled to zero in about 2.0 seconds). However, for the proper density computations, test case 1 and 2 both result in instability. These results indicate that the method is extremely sensitive to underestimation of the bending mode frequencies. Case 3 results (where the controller overestimated the bending mode frequencies) provide stable control, similar to the standard MMAC.

5.3.4.4 Optimal Control Vector Summary. The results indicated here show that the moving-bank modified MMAC provides noticeable performance improvement over the MMAC method and performance comparable to that of the artificially informed non-adaptive benchmark. The use of ME/I computations provides slightly better performance than use of the proper density function. In addition, instability problems result with the proper density function implementation for case 2 in every study conducted (except for the modified MMAC method). Therefore, the remainder of the controller simulations use the ME/I method.

5.3.5 Control Parameter Space Discretization Study. For this study, the region of the parameter space for high values of modal frequencies was "densely packed" in order to reduce the destabilizing affects of understimating the modal frequencies during the controller synthesis. The boundaries for the mass and stiffness scaling factors remained the same as before, while the space was discretized using a logarithmic scaling. The new parameter space, determined in an ad hoc fashion, is provided in Table 5.3. For this study, the three test cases described in Section 5.3.3.3 were investigated using parameter position estimate monitoring with a threshold of 0.01 and ME/I computations. The controller logic implemented the conventional MMAC algorithm instead of the modified MMAC logic to determine the impact of the new space without omitting any controllers in the bank. The results for all three cases indicate no significant difference in controller performance when compared to the moving-bank MMAC simulation performed for the original space discretization. These results are provided in the bottom half of of Tables 5.8 and 5.9. This is understandable since there are two major differences between Sheldon's work [23] and the application investigated here. First, the discretized parameter space investigated by Shel-

don had only three coarsely discretized points. Underestimation of the natural frequency modes for the structure with such a coarse space results in significantly inappropriate control. Second, the moving-bank algorithm quickly moves the controllers to the appropriate region of parameter space (unlike a non-moving-bank MMAC with lower bounding on computed p_k's, which ensures that some nonzero control contribution is generated on the basis of very incorrect parameter assumptions), thereby reducing the magnitude of inappropriate control. Based on these results, the remaining parameter variations investigations use the original parameter space discretization.

5.3.6 Controller Varying Parameter Study. The goal of this study was to investigate the performance of the moving-bank controller algorithms to two types of parameter changes. The test cases investigated here are the same as those used previously for the estimator studies in Section 5.2.6. In order to investigate the performance of the controller algorithms to slowly varying paramters and jump parameters properly, benchmarks for best-case performance must be established. The benchmarks for these simulations are similar to the non-adaptive, artificially informed best-case single-controller study performed in Section 5.3.3.1. However, for these benchmarks, the bank is allowed to move and it is artificially informed of the true parameter change during the simulation. A benchmark for each parameter variation study must be performed in order to analyze the results correctly.

5.3.6.1 Slowly Varying Parameter Analysis. The first test case, as described previously in Section 5.2.6.1, represents a change toward higher natural frequencies of the bending modes, whereas the second test case represents a change toward lower natural frequencies. For each test case, the true parameter variation was a discrete change implemented at t=5.0 seconds into each 10-second Monte Carlo run. The move logic consisted of parameter position estimation with ME/I calculations. The controller logic used modified MMAC with a threshold of 0.10. Results of the controller performance in controlling the structure for case 1 indicate performance comparable to the benchmark. Tables 5.10 and 5.11 provide a summary of these results. Comparing the results to the MMAE slowly varying parameter study, provided in Tables 5.4 and 5.5, the control is very good.

Table 5.10. Modified MMAC Varying Parameter Position Estimation Errors

Simulation	Node 1 Error(1)		Node 2 Error(1)	
⇓	Mean	1σ	Mean	1σ
Benchmarks				
Bench 1	0.0002	0.0011	0.0000	0.0001
Bench 2	0.0000	0.0012	0.0000	0.0003
Fine-Bank Movement Simulations				
Case 1	-.0001	0.0014	0.0000	0.0001
Case 2	0.0000	0.0014	0.0000	0.0010
Note (1): Average values for $t > 5.0$ seconds (inches)				

Table 5.11. Modified MMAC Varying Parameter True Positions

Simulation	Node 1 True (1)		Node 2 True (1)	
⇓	Mean	1σ	Mean	1σ
Benchmarks				
Bench 1	0.0001	0.0012	-.0002	0.0001
Bench 2	-.0001	0.0013	-.0003	0.0003
Fine-Bank Movement Simulations				
Case 1	-.0001	0.0015	-.0004	0.0009
Case 2	0.0001	0.0015	-.0001	0.0012
Note (1): Average values for $t > 5.0$ seconds (inches)				

For the case 2 study, the bank performed as well, as demonstrated by the small 1 σ estimation errors and true position values provided in Tables 5.10 and 5.11.

5.3.6.2 Jump Parameter Analysis. In order to evaluate the controller performance for jump parameters, as for the estimation studies of Section 5.2.6.2, two methods were investigated for allowing the bank to adapt. The first method disables the expansion/contraction capability to investigate the performance of the finely discretized bank to move to the correct location. Figures 5.72 through 5.76 show the single-controller benchmark for jump parameter case 1. Figures 5.77 through 5.82 show the estimation errors and the controller performance for the fine-bank move for the case 1 jump using parameter position estimate monitoring, the ME/I density, and modified MMAC. Tables 5.12 and 5.13 provide a summary of these results. The resulting truss control compares favorably to the artificially informed benchmark. Comparing these results to the corresponding MMAE studies for the jump parameter (provided in Tables 5.6 and 5.7 in Section 5.2.6.2), this method provides very good control over the truss positions. The results for the case 2 study also indicate that the fine-bank movement provides very tight control over the truss positions.

The second method for acquiring the parameter after a jump change allows expansion and contraction of the bank. Using an expansion threshold of 0.5, the bank was allowed to expand after 1.0 seconds into the run to avoid erroneous expansion due to the large initial control output from the actuators. The first contraction threshold was set to 10.00. The second contraction threshold was 5.00. As for the first method, parameter position estimate monitoring, the ME/I density, and modified MMAC are implemented. The results provide slightly worse control performance compared to the fine-bank move results. The results for the case 2 jump also provide performance slightly worse than for the fine-bank movement, as seen in Tables 5.12 and 5.13. Therefore, the results of this study indicate estimation errors and controller performance are better for bank movement alone than allowing expansion and contraction.

Table 5.12. Modified MMAC Jump Parameter Position Estimation Errors

Simulation	Node 1 Error(1)		Node 2 Error(1)	
⇓	Mean	1σ	Mean	1σ
Benchmarks				
Bench 1	0.0000	0.0019	0.0000	0.0001
Bench 2	0.0000	0.0012	0.0000	0.0000
Fine-Bank Movement				
Case 1	-.0001	0.0019	0.0000	0.0010
Case 2	0.0000	0.0014	0.0000	0.0011
Expansion/Contraction				
Case 1	0.0000	0.0029	0.0001	0.0011
Case 2	0.0001	0.0017	0.0001	0.0012
Note (1): Average values for $t > 5.0$ seconds (inches)				

Table 5.13. Modified MMAC Jump Parameter True Positions

Simulation	Node 1 True(1)		Node 2 True(1)	
⇓	Mean	1σ	Mean	1σ
Benchmarks				
Bench 1	0.0012	0.0022	-.0002	0.0012
Bench 2	0.0009	0.0008	0.0001	0.0007
Fine-Bank Movement				
Case 1	0.0007	0.0021	-.0004	0.0010
Case 2	-.0001	0.0013	0.0000	0.0001
Expansion/Contraction				
Case 1	0.0001	0.0023	-.0001	0.0011
Case 2	0.0003	0.0016	0.0005	0.0004
Note (1): Average values for $t > 5.0$ seconds (inches)				

5.4 *Summary*

The results presented in this chapter covered the performance of the moving-bank MMAE and MMAC algorithms. The modification studies, which became a substantial part of this study, demonstrated significant improvement in the performance of the algorithms and established baseline performance results difficult to improve upon. The use of the filter-computed residual covariance, $\mathbf{A}_k$, was found to provide biased results in the p_k computation and was therefore not used in conjunction with parameter position estimate monitoring, which was determined to provide the best moving-bank estimation results. After appropriate determination of the filter $\mathbf{Q}$ and $\mathbf{R}$ matrices and the LQG controller state and control weighting matrices for the high bending-mode natural frequency region of the space, a modified MMAC methodology provided results comparable to that obtained from a non-adaptive artificially informed benchmark. The evaluation of a parameter space discretization densely packed in the high natural frequency region provided results not significantly different from the original discretization. Finally, the moving-bank algorithm performance to slowly varying and jump parameters indicate that the best estimation and control results were obtained by a fine-bank move without the use of expansion and contraction logic. Chapter 6 provides a detailed discussion of these results and provides recommendations for future research.

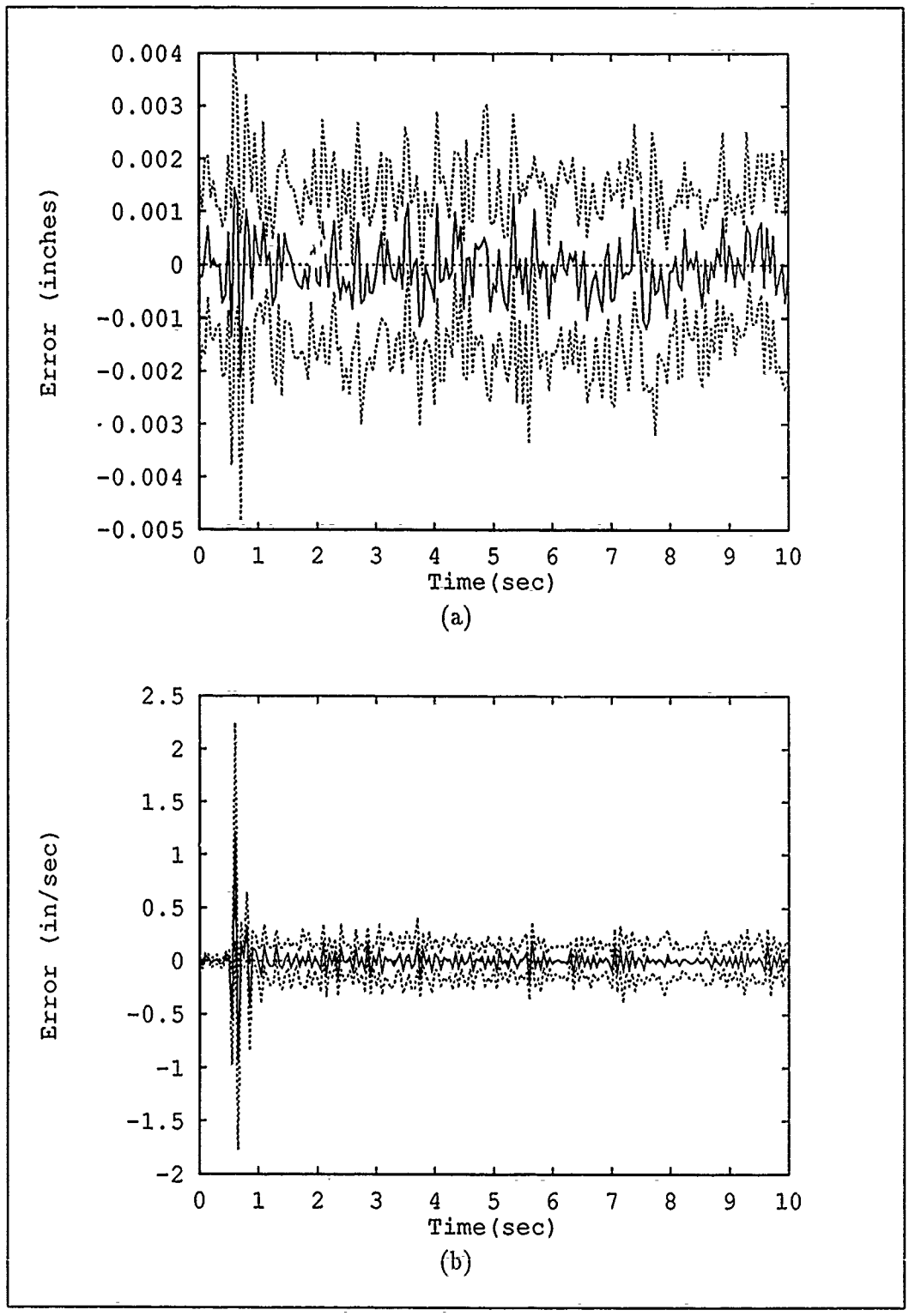

Figure 5.35. Duplication of Single Controller Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter and Controller at (7,6). (a) Position and (b) Velocity Estimation Errors.

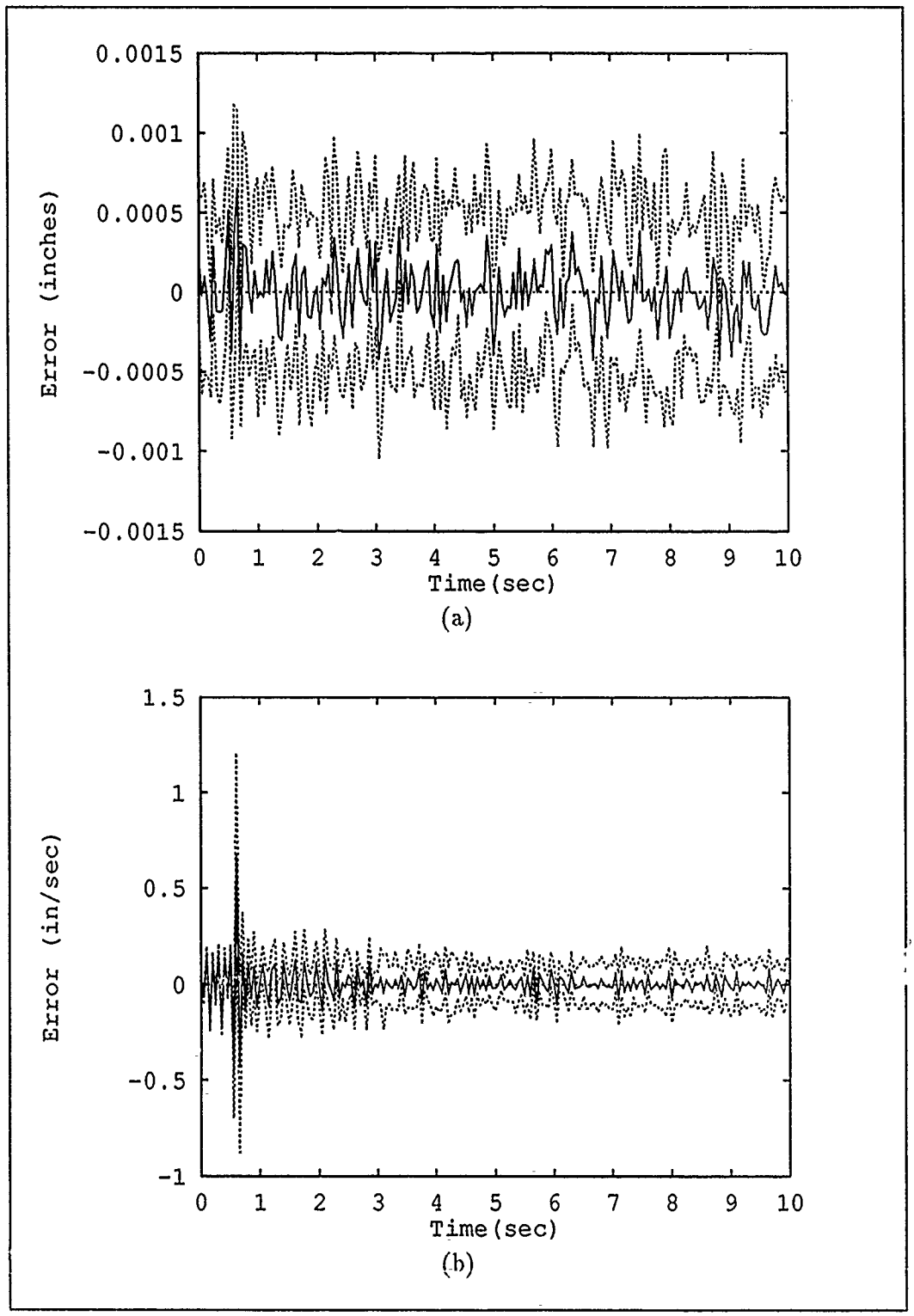

Figure 5.36. Duplication of Single Controller Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter and Controller at (7,6). (a) Position and (b) Velocity Estimation Errors.

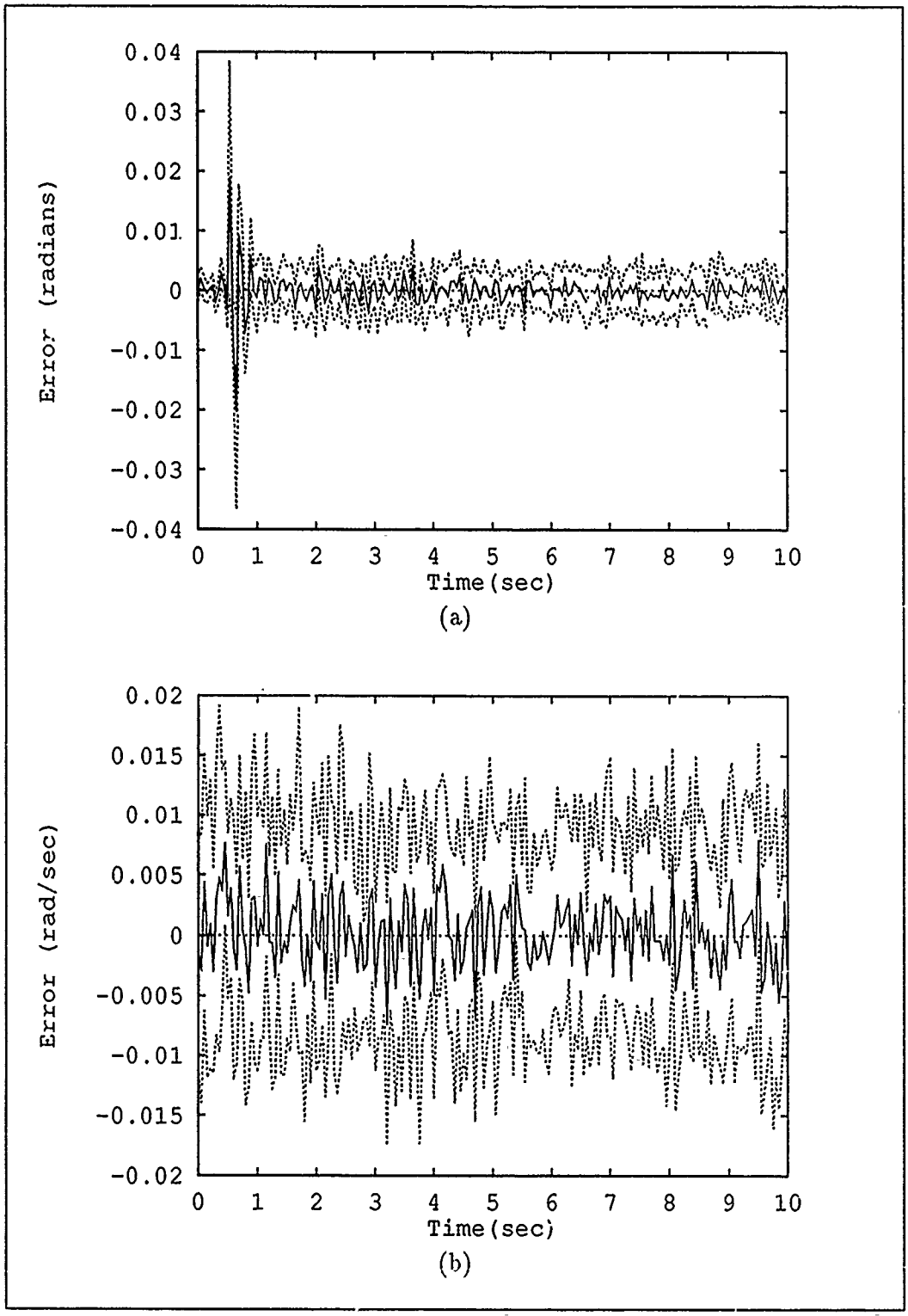

Figure 5.37. Duplication of Single Controller Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter and Controller at (7,6). (a) Position and (b) Velocity Estimation Errors.

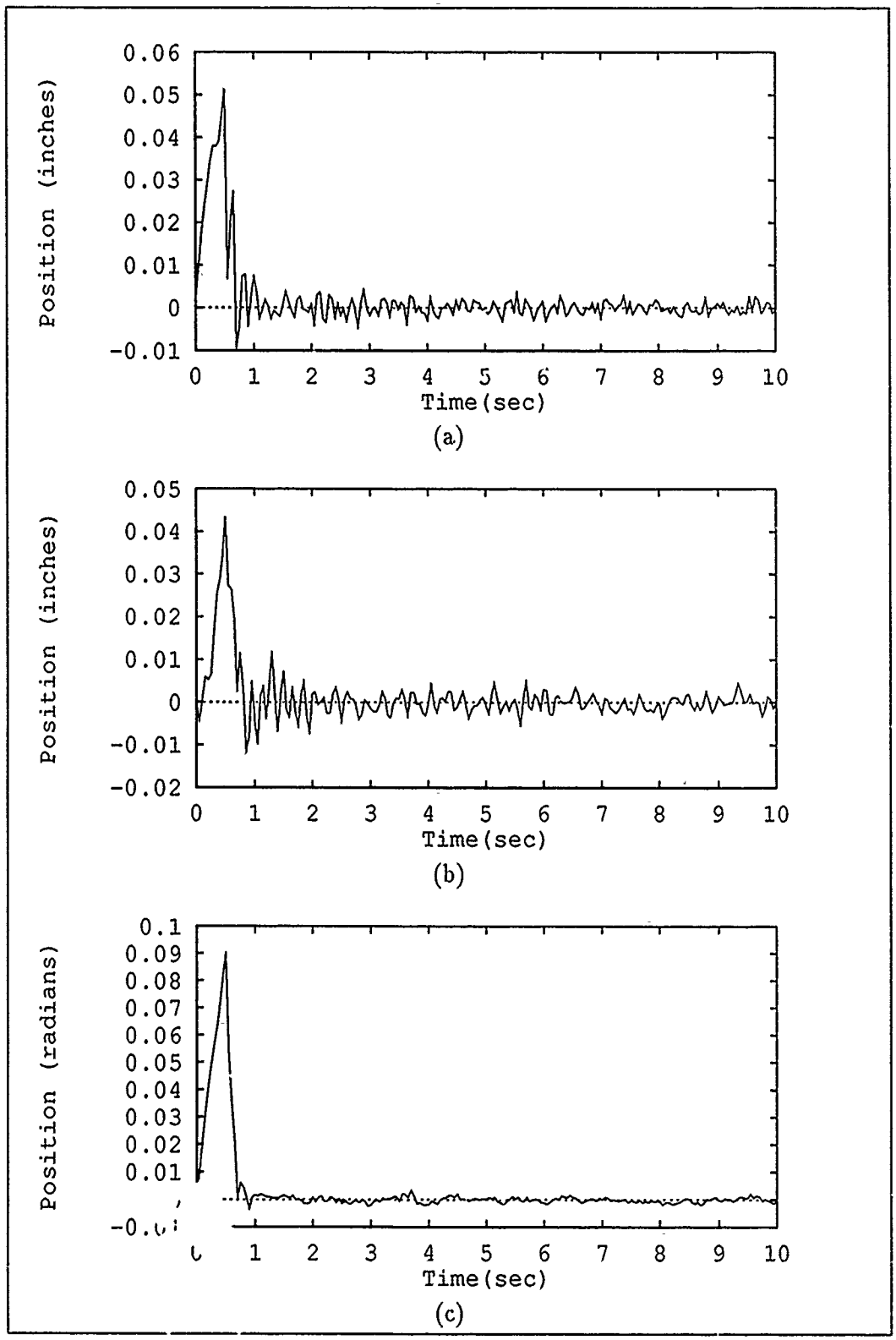

Figure 5.38. Duplication of Single Controller Structure Positions (Sample Mean) - True Parameter and Controller at (7,6). (a) Node 1 (b) Node 2 and (c) Node 7 Positions.

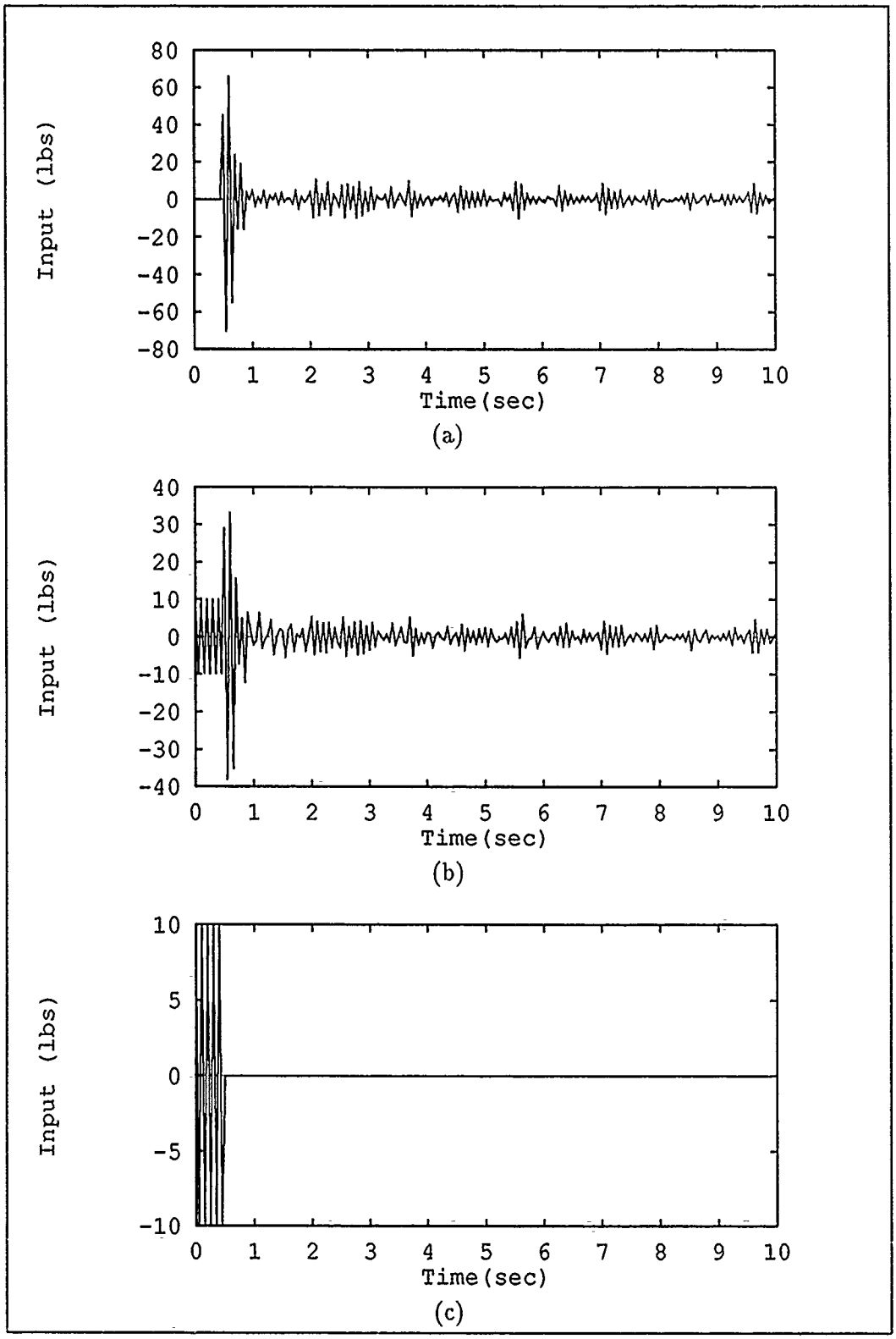

Figure 5.39. Duplication of Control Inputs for Single Controller (Sample Mean) - True Parameter and Control' r at (7,6). (a) Node 1 (b) Node 2 and (c) Node 7 Actuators.

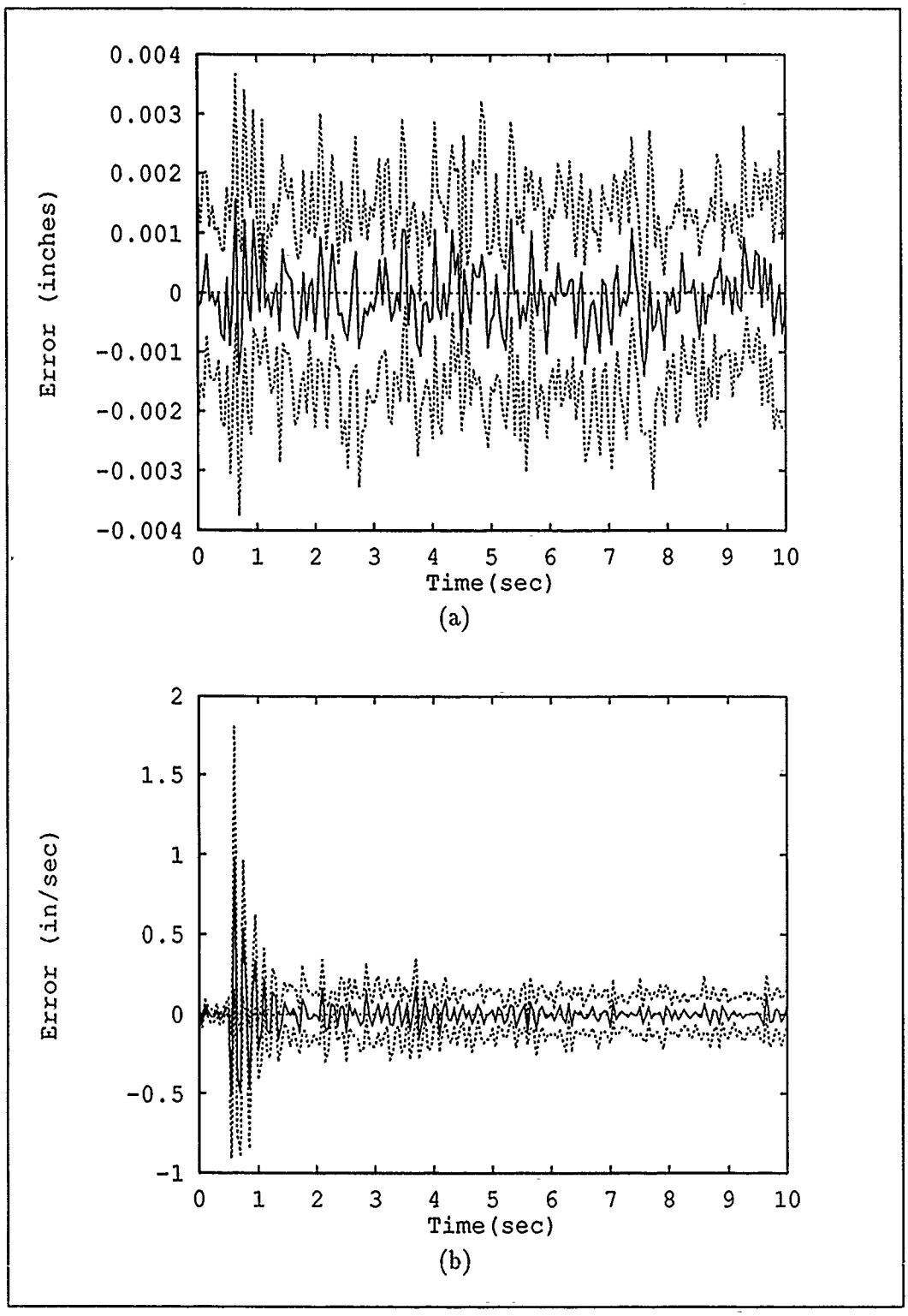

Figure 5.40. Duplication of Moving-Bank MMAC Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter at (7,6) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors.

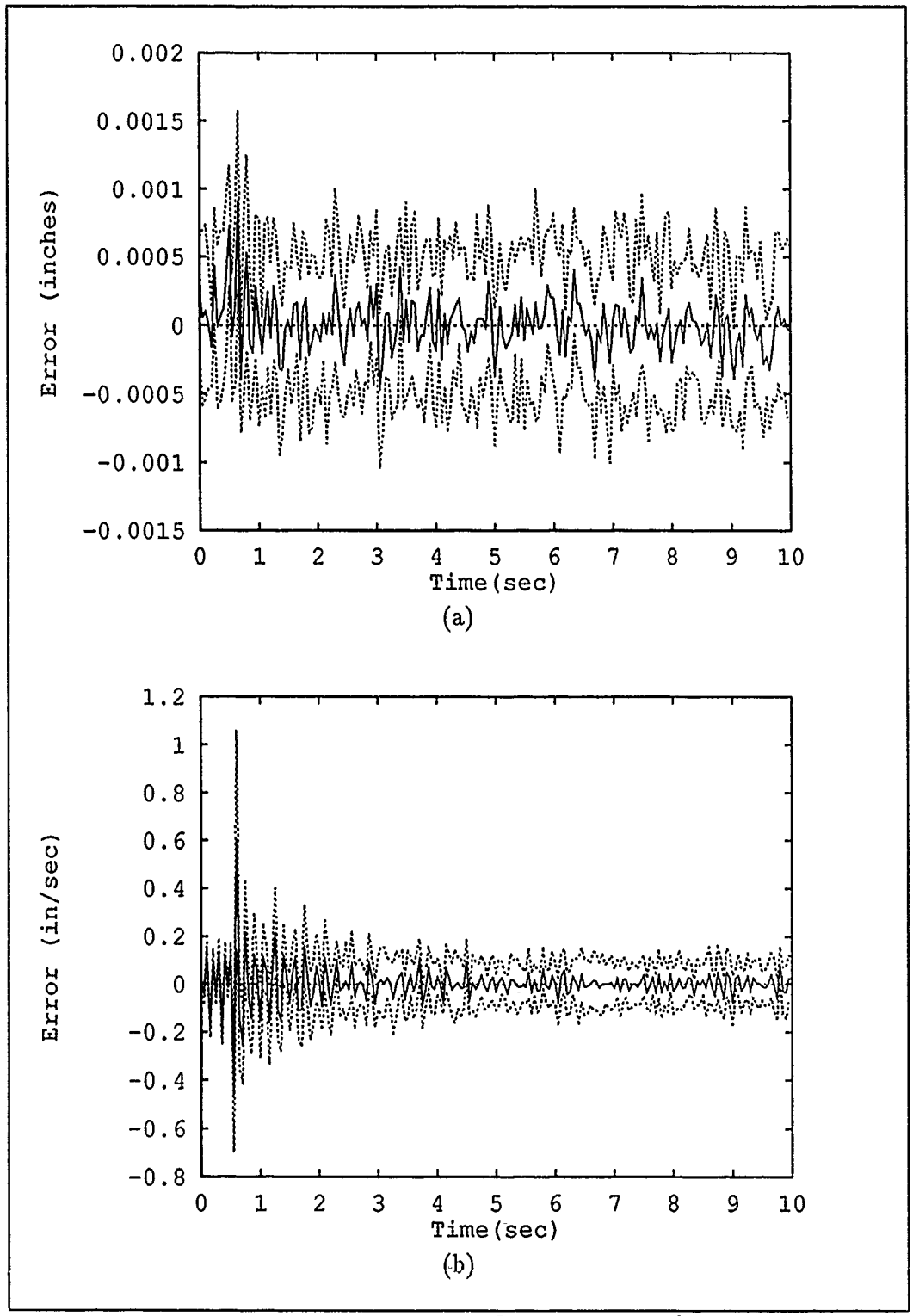

Figure 5.41. Duplication of Moving-Bank MMAC Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter at (7,6) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors.

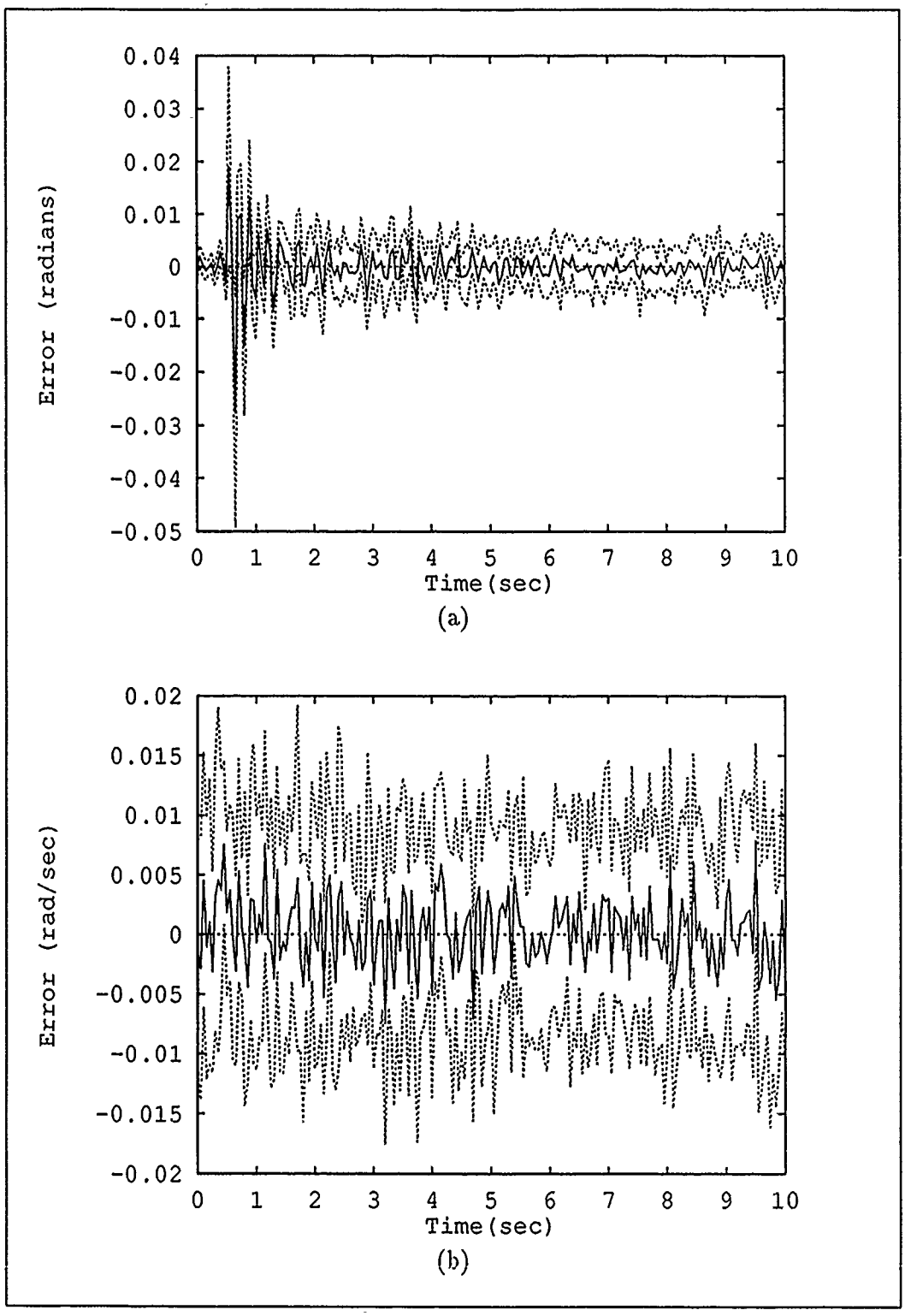

Figure 5.42. Duplication of Moving-Bank MMAC Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter at (7,6) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors.

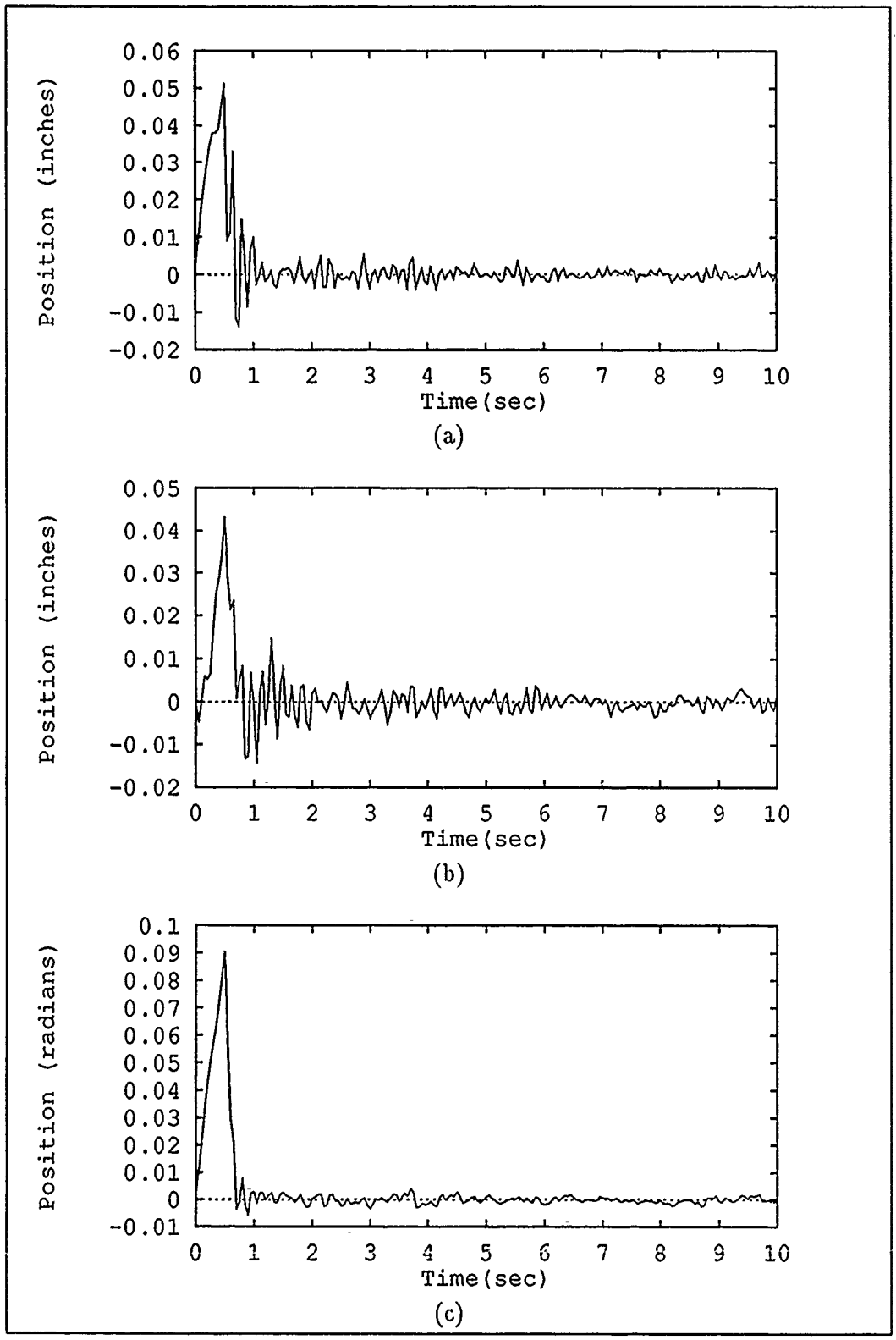

Figure 5.43. Duplication of Moving-Bank MMAC Structure Positions (Sample Mean) - True Parameter at (7,6) and Bank Initially at (5,5). (a) Node 1 (b) Node 2 and (c) Node 7 Positions.

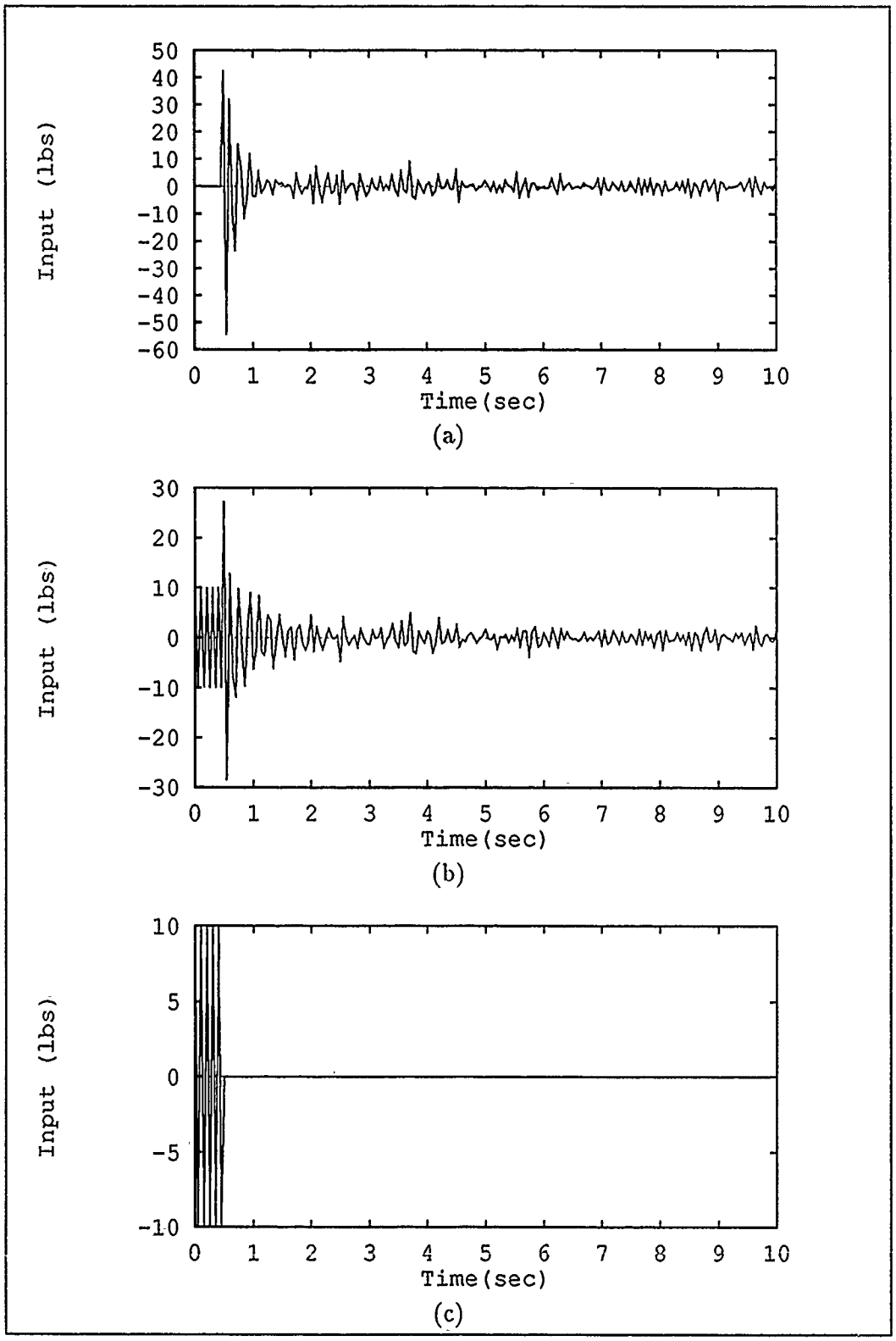

Figure 5.44. Duplication of Moving-Bank MMAC Control Inputs (Sample Mean) - True Parameter at (7,6) and Bank Initially at (5,5). (a) Node 1 (b) Node 2 and (c) Node 7 Actuators.

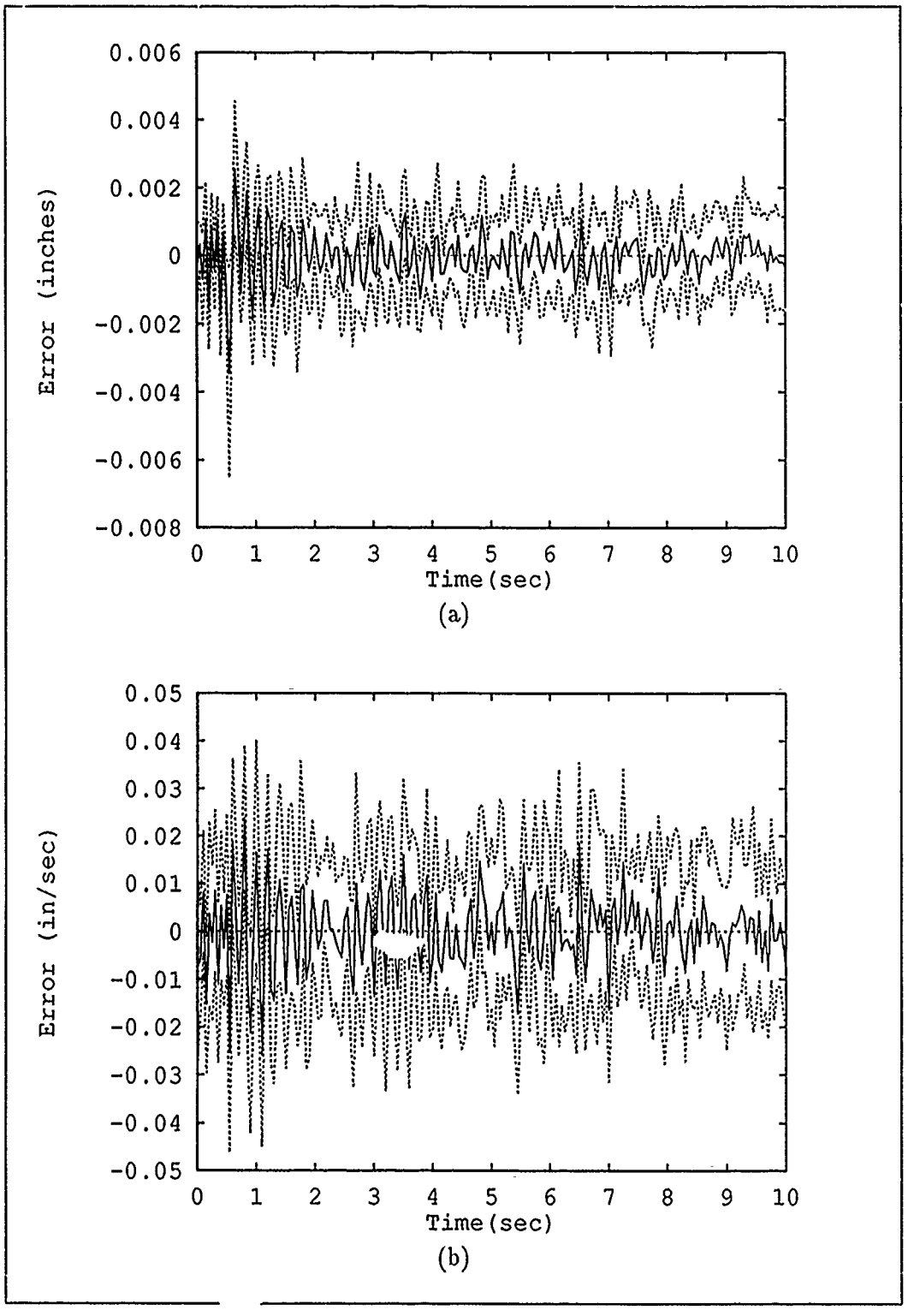

Figure 5.45. Modified Single Controller Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter and Controller at (7,6). (a) Position and (b) Velocity Estimation Errors.

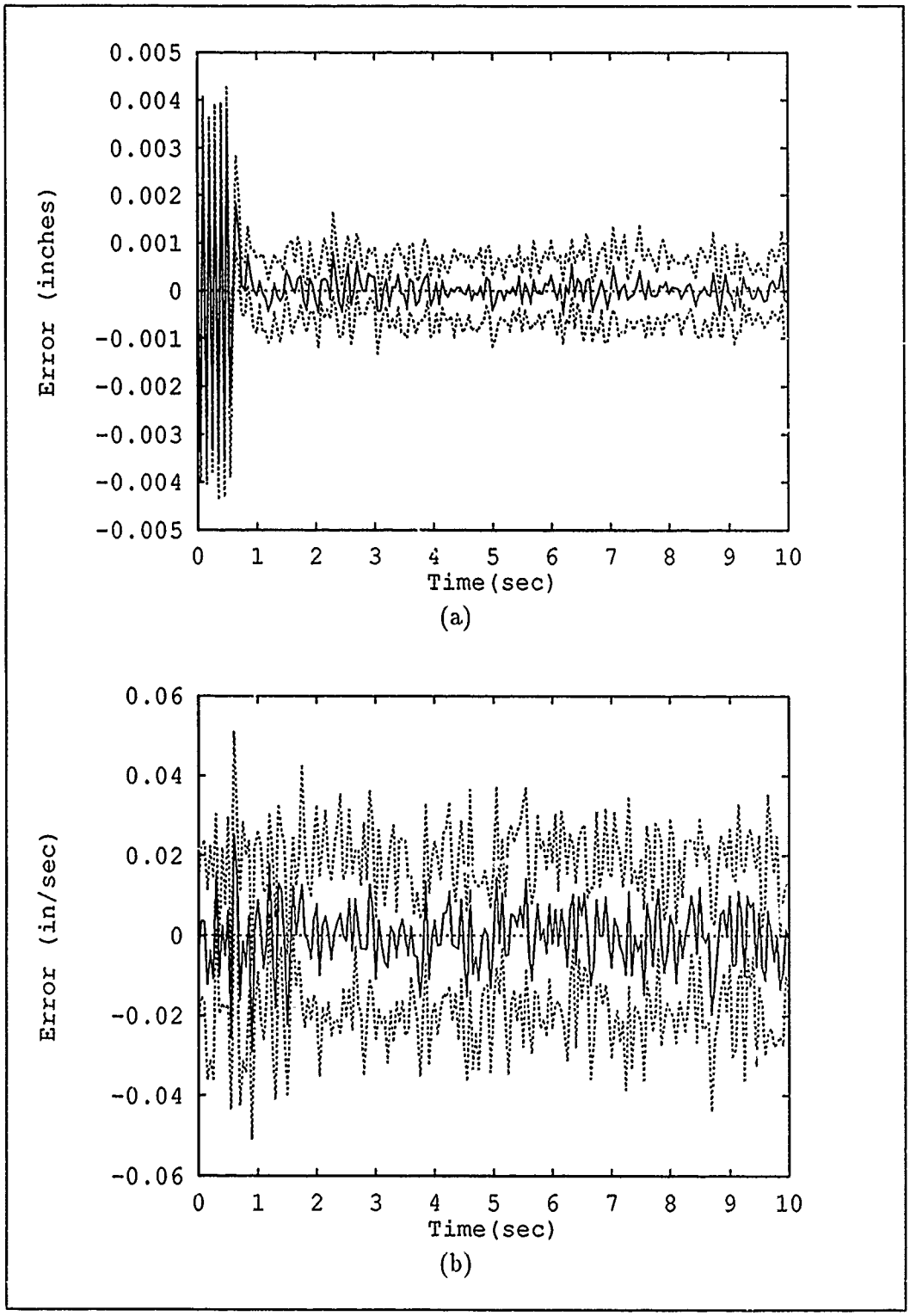

Figure 5.46. Modified Single Controller Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter and Controller at (7,6). (a) Position and (b) Velocity Estimation Errors.

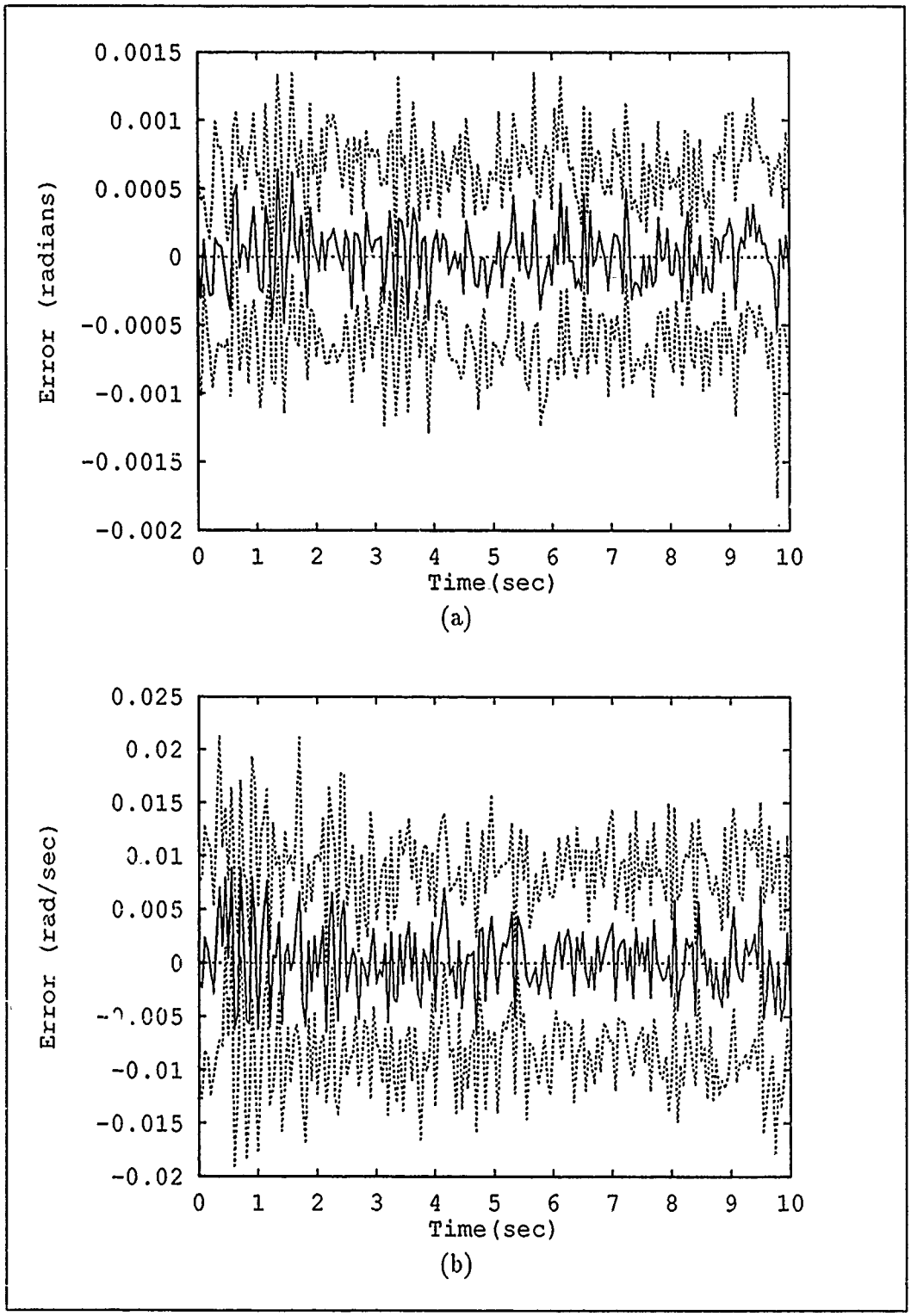

Figure 5.47. Modified Single Controller Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter and Controller at (7,6). (a) Position and (b) Velocity Estimation Errors.

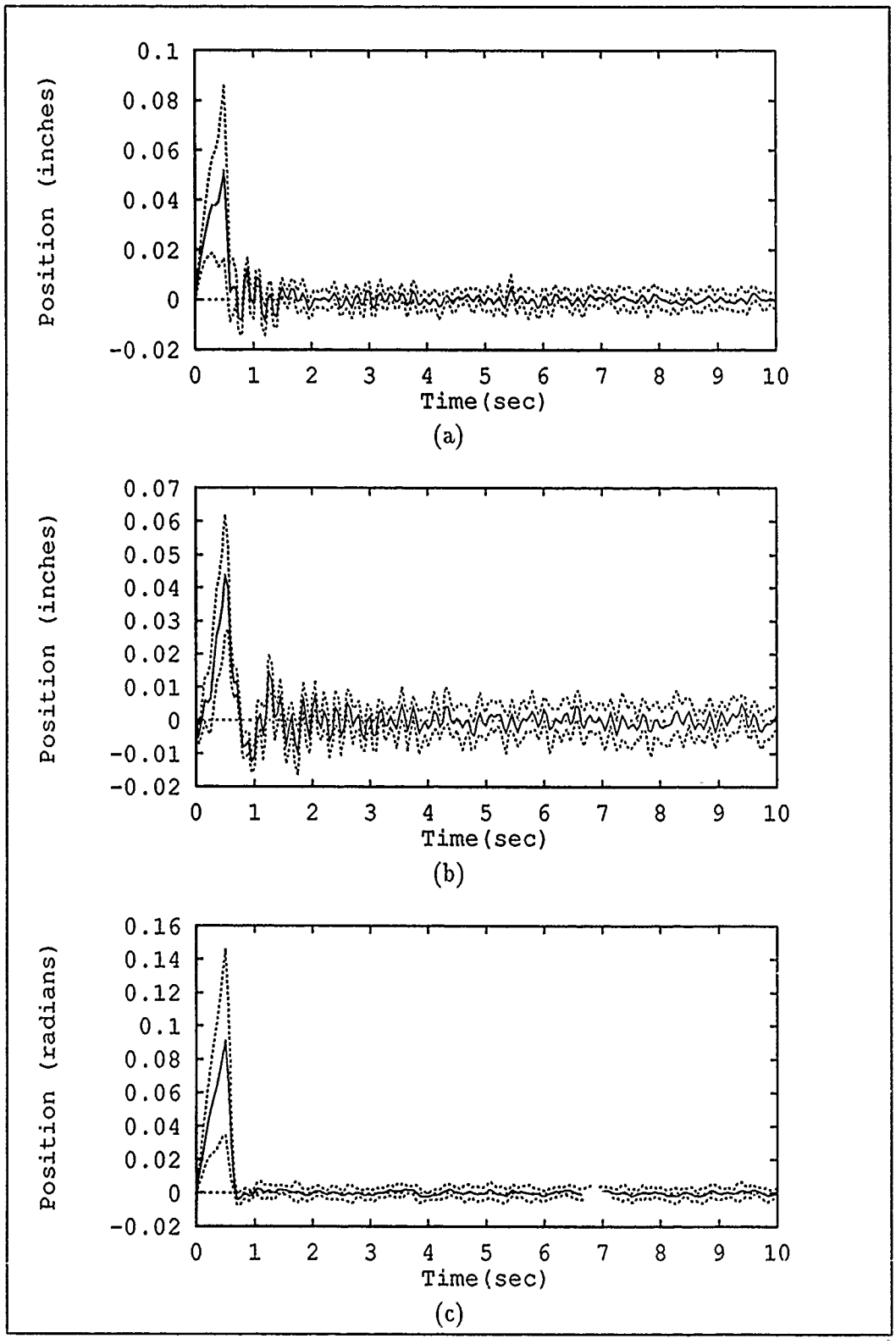

Figure 5.48. Modified Single Controller Structure Positions (Mean $\pm$ One Standard Deviation) - True Parameter and Controller at (7,6). (a) Node 1 (b) Node 2 and (c) Node 7 Positions.

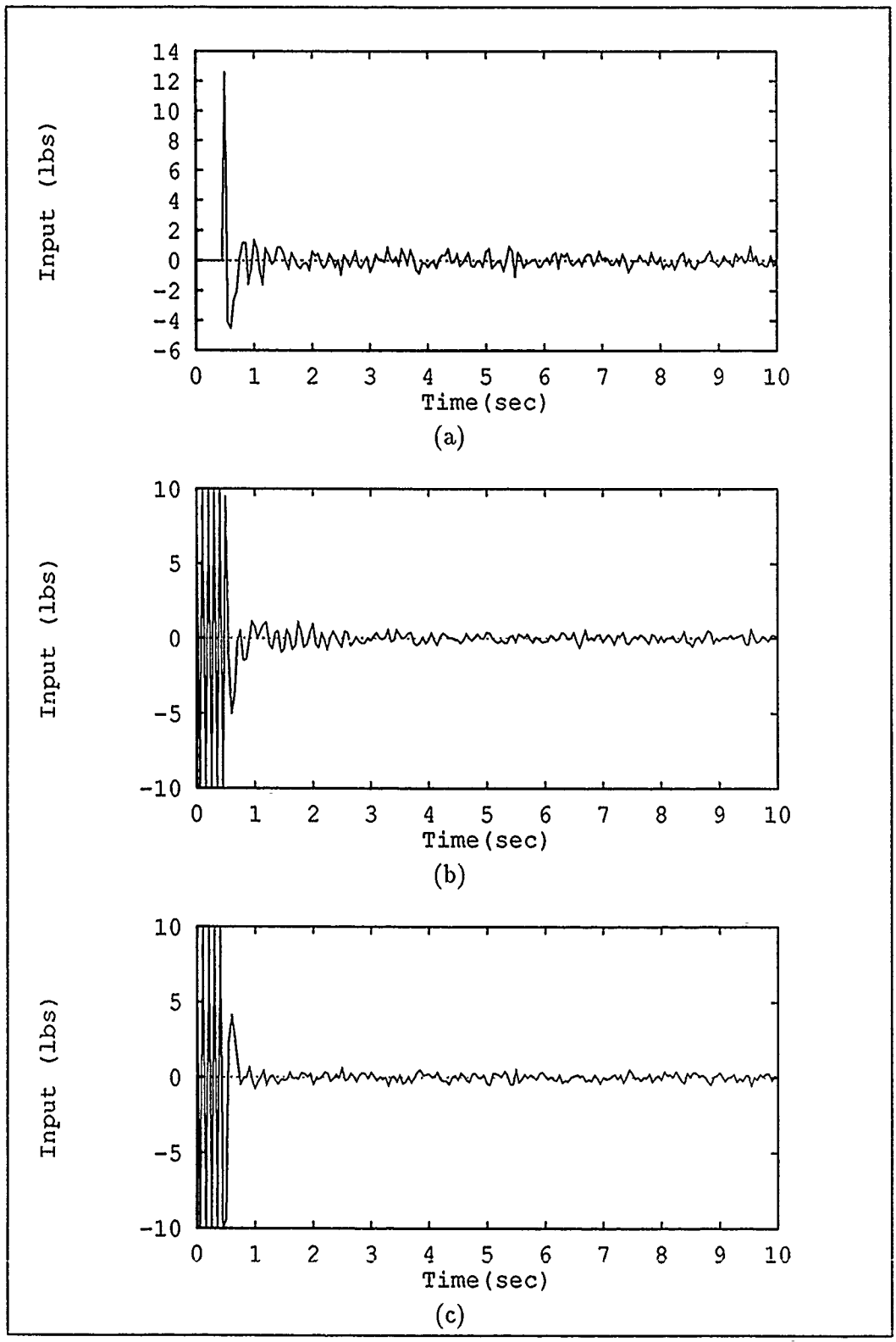

Figure 5.49. Modified Single Controller Control Inputs (Sample Mean) - True Parameter and Controller at (7,6). (a) Node 1 (b) Node 2 and (c) Node 7 Actuators.

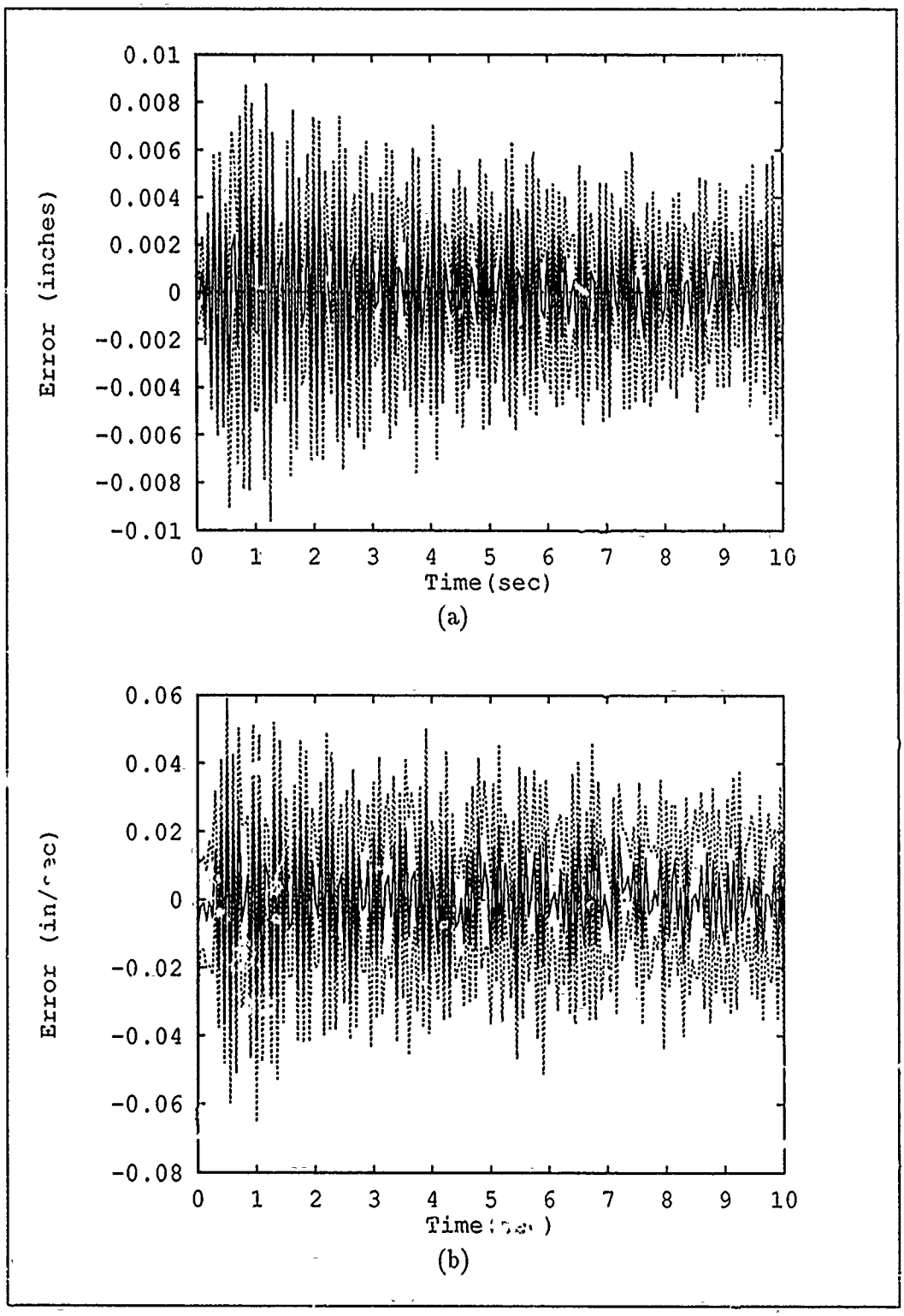

Figure 5.50. Untuned Single Controller Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter and Controller at (1). (a) Position and (b) Velocity Estimation Errors.

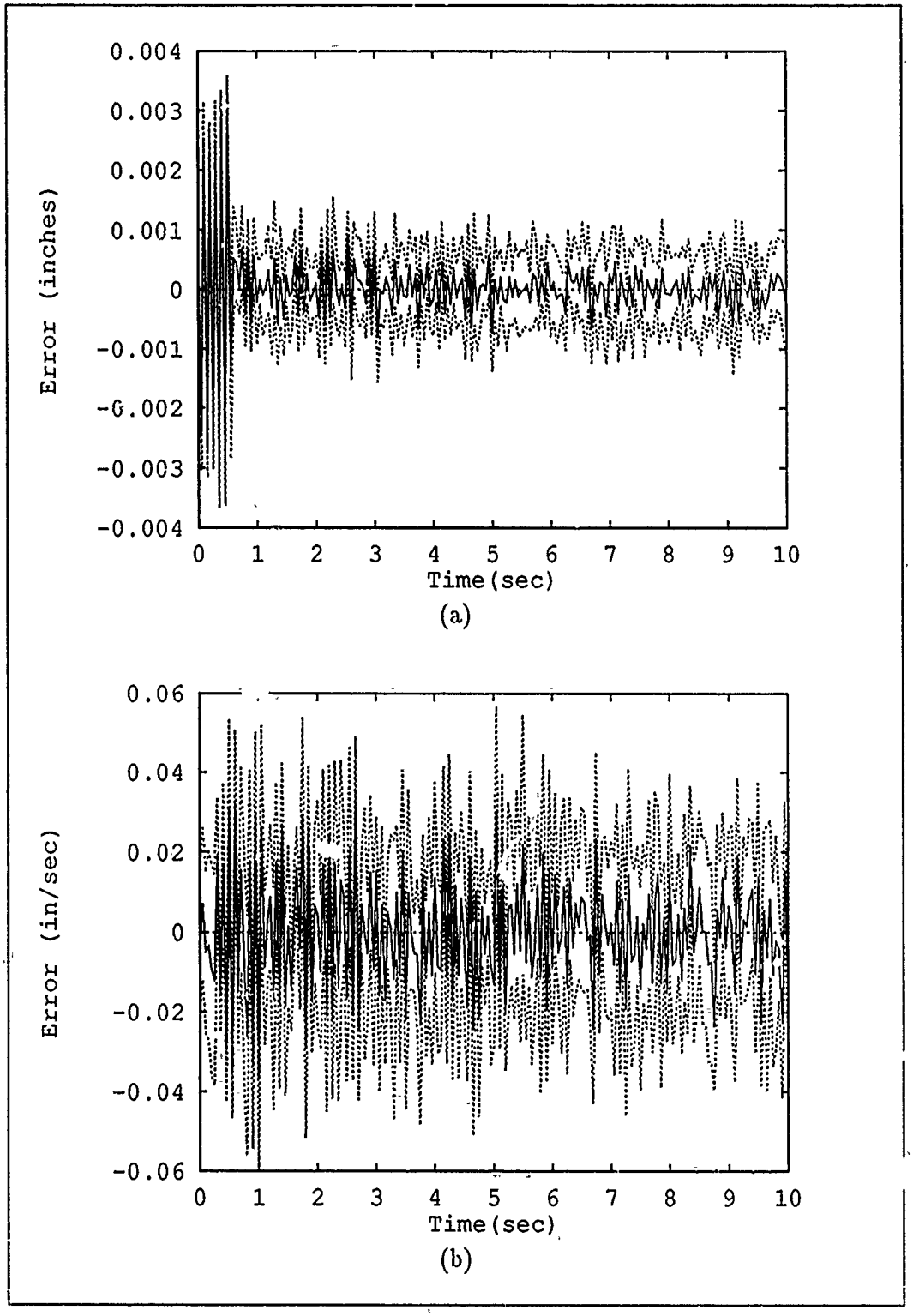

Figure 5.51. Untuned Single Controller Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter and Controller at (1,9). (a) Position and (b) Velocity Estimation Errors.

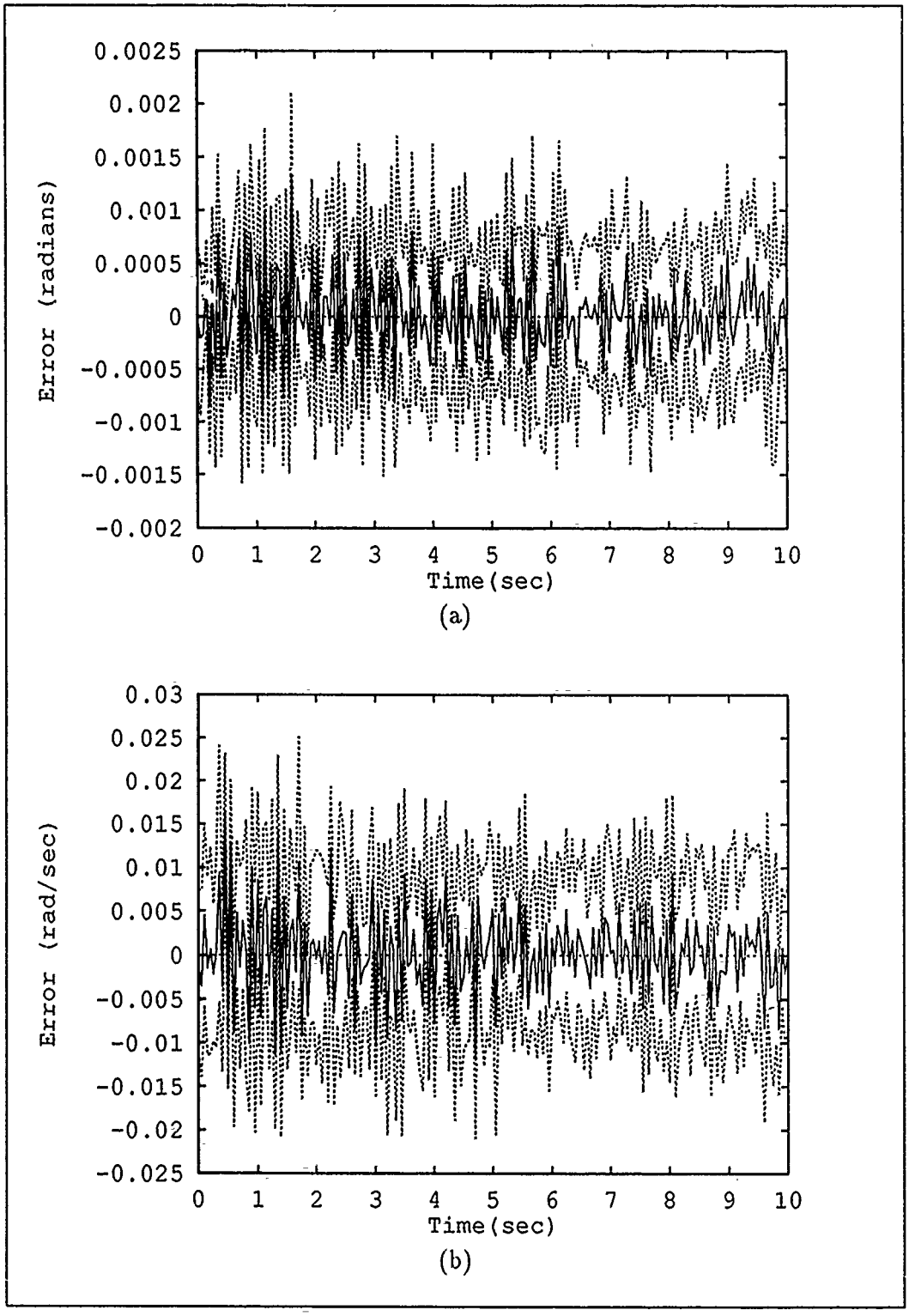

Figure 5.52. Untuned Single Controller Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter and Controller at (1,9). (a) Position and (b) Velocity Estimation Errors.

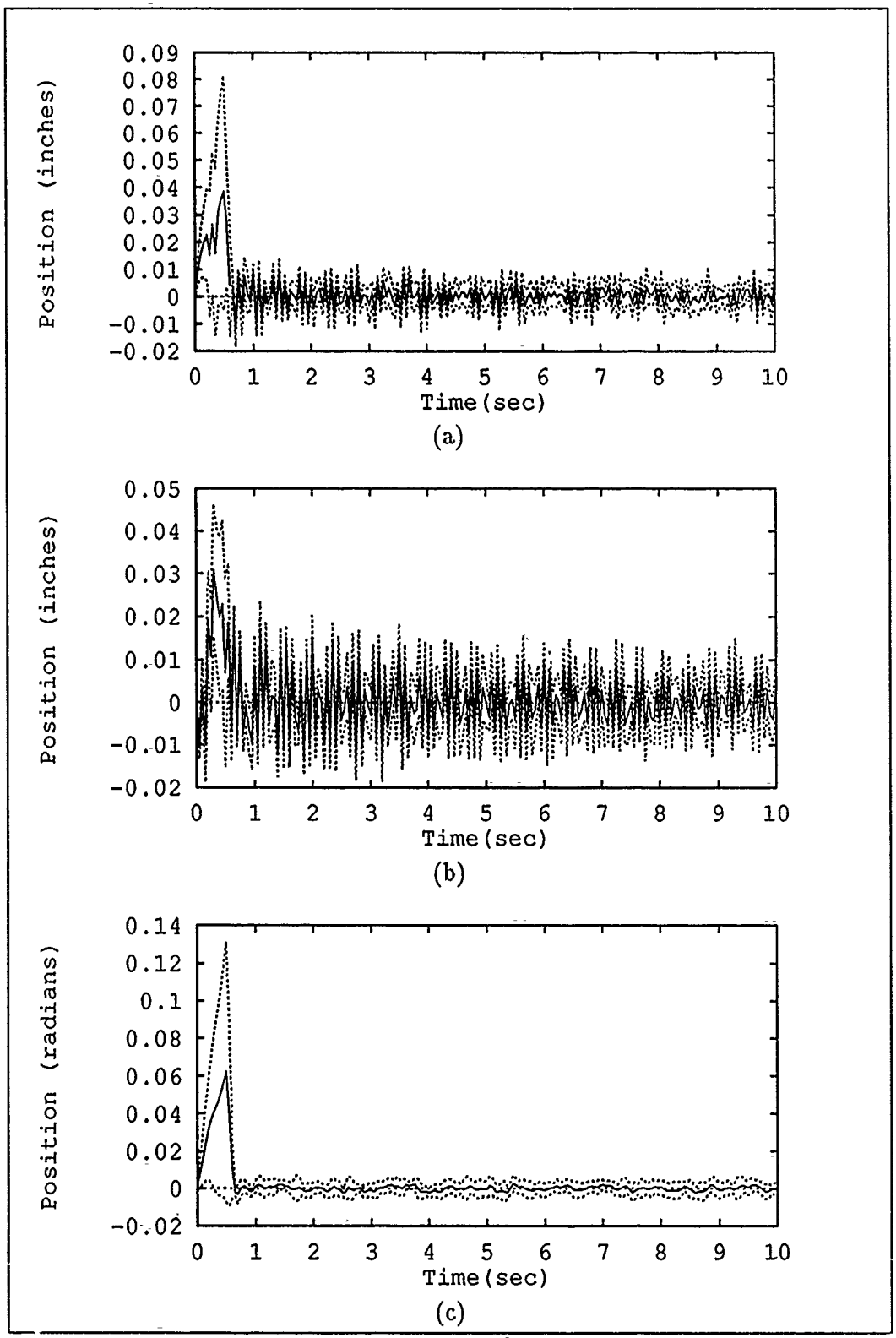

Figure 5.53. Untuned Single Controller Structure Positions (Mean $\pm$ One Standard Deviation) - True Parameter and Controller at (1,9). (a) Node 1 (b) Node 2 and (c) Node 7 Positions.

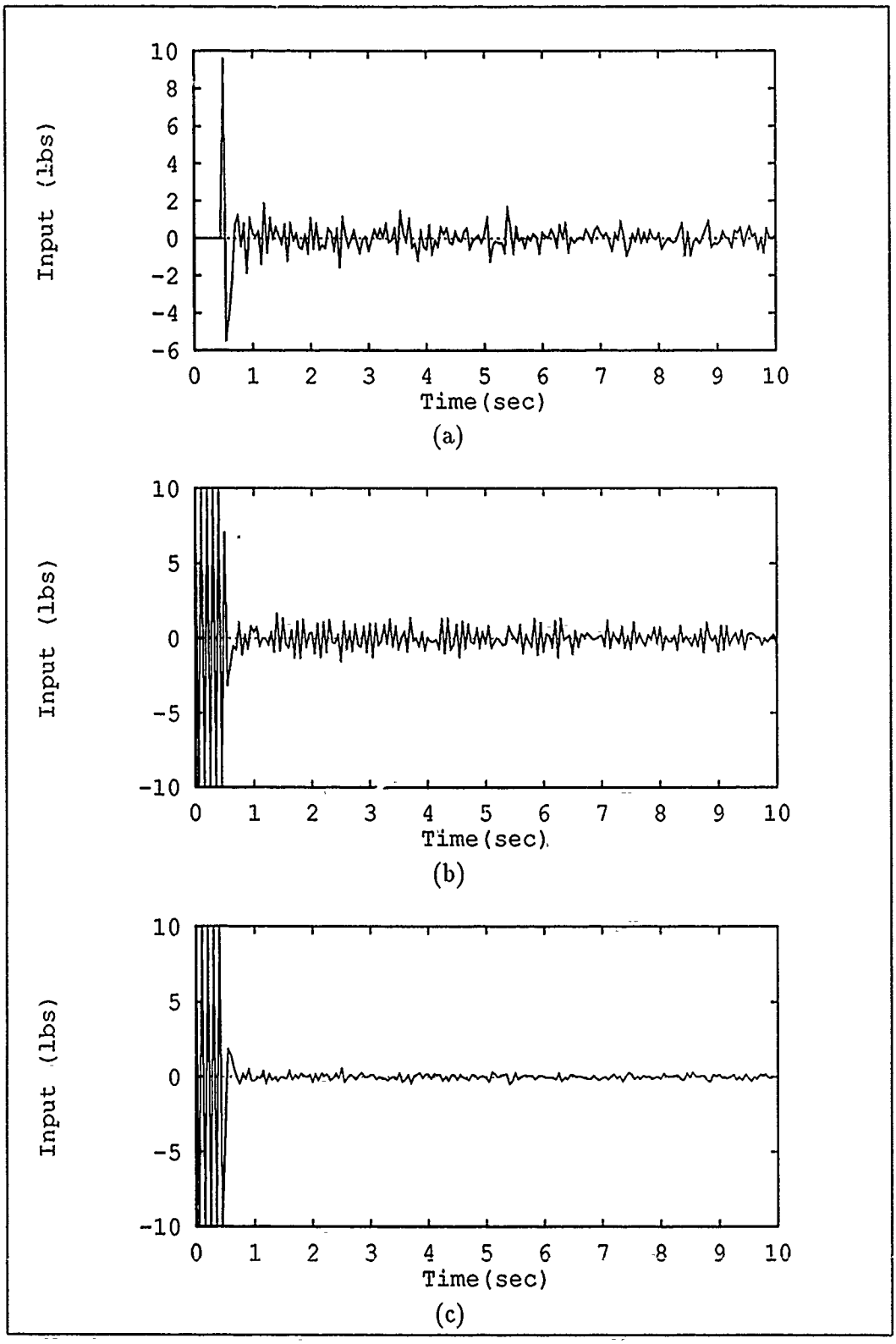

Figure 5.54. Untuned Single Controller Control Inputs (Sample Mean) - True Parameter and Controller at (1,9). (a) Node 1 (b) Node 2 and (c) Node 7 Actuators.

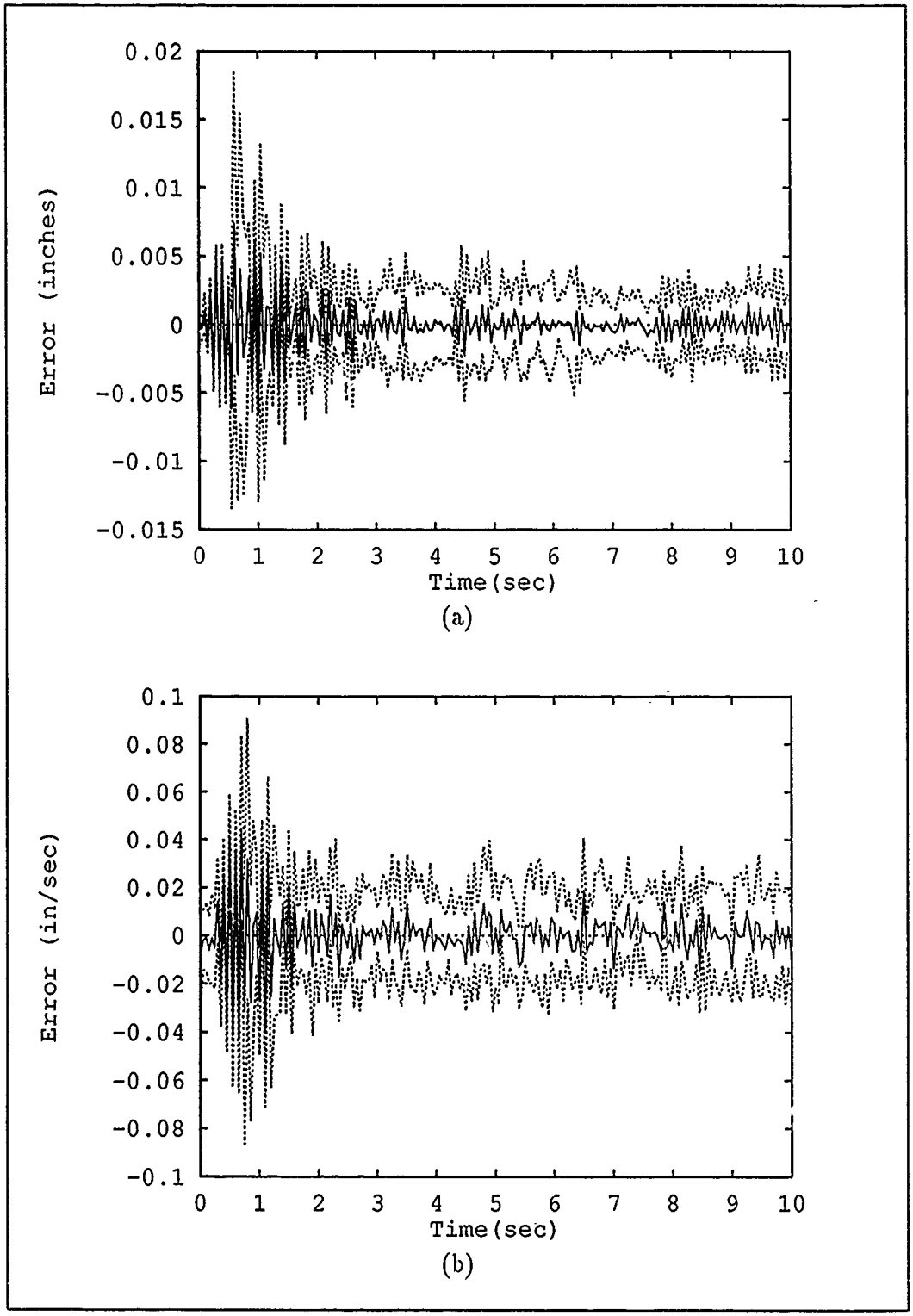

Figure 5.55. Tuned Single Controller Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter and Controller at (1,9). (a) Position and (b) Velocity Estimation Errors.

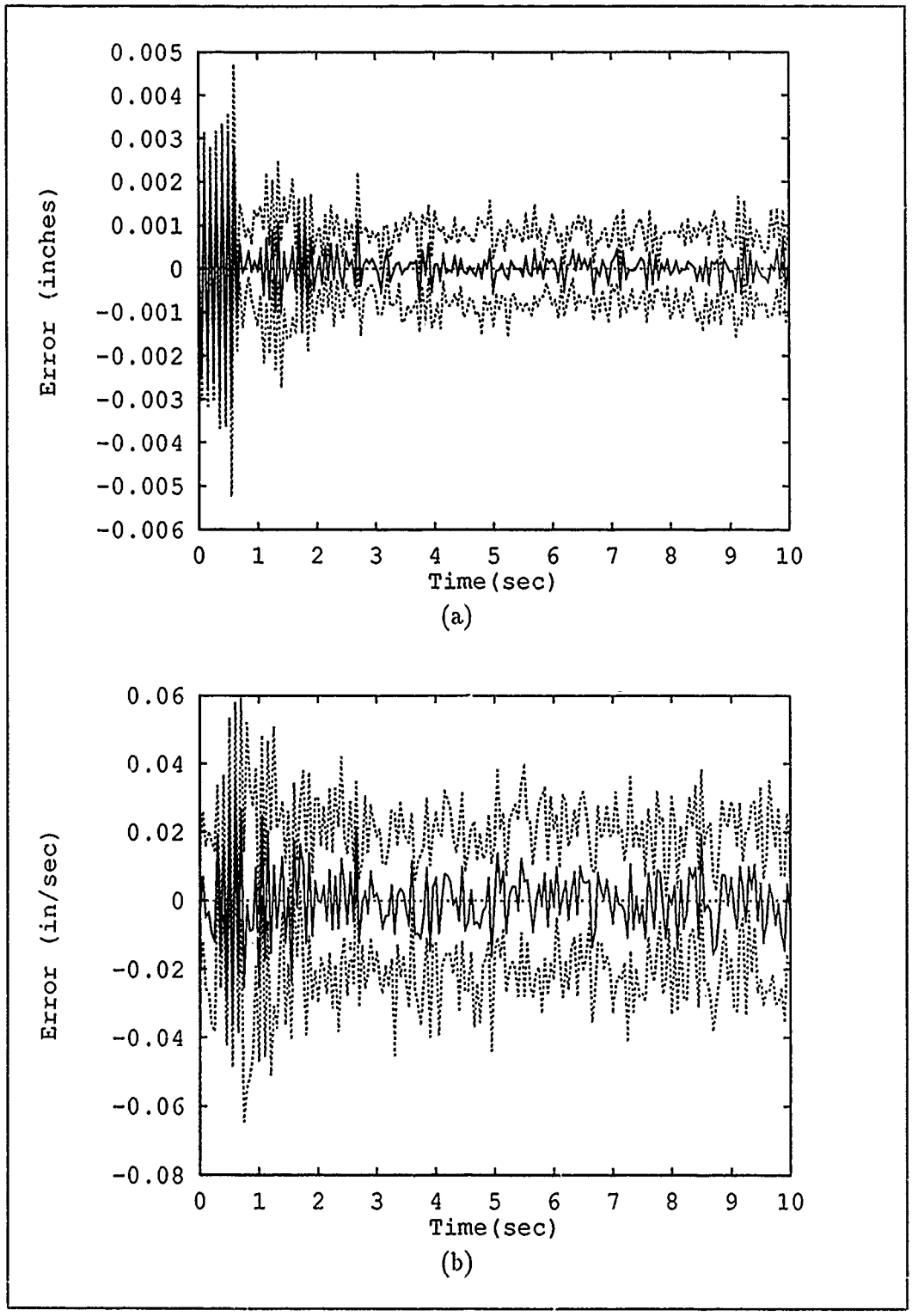

Figure 5.56. Tuned Single Controller Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter and Controller at (1,9). (a) Position and (b) Velocity Estimation Errors.

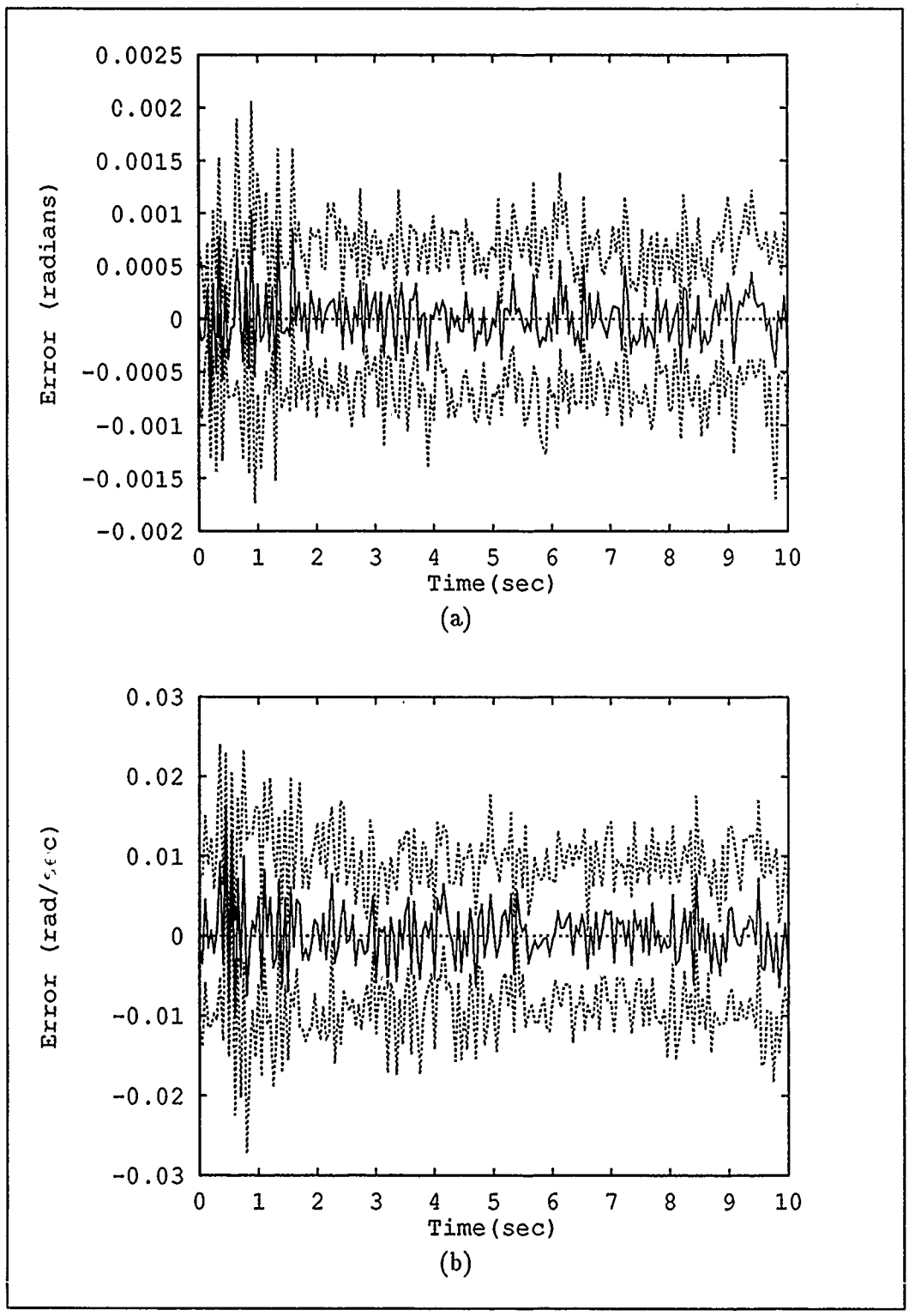

Figure 5.57. Tuned Single Controller Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter and Controller at (1,9). (a) Position and (b) Velocity Estimation Errors.

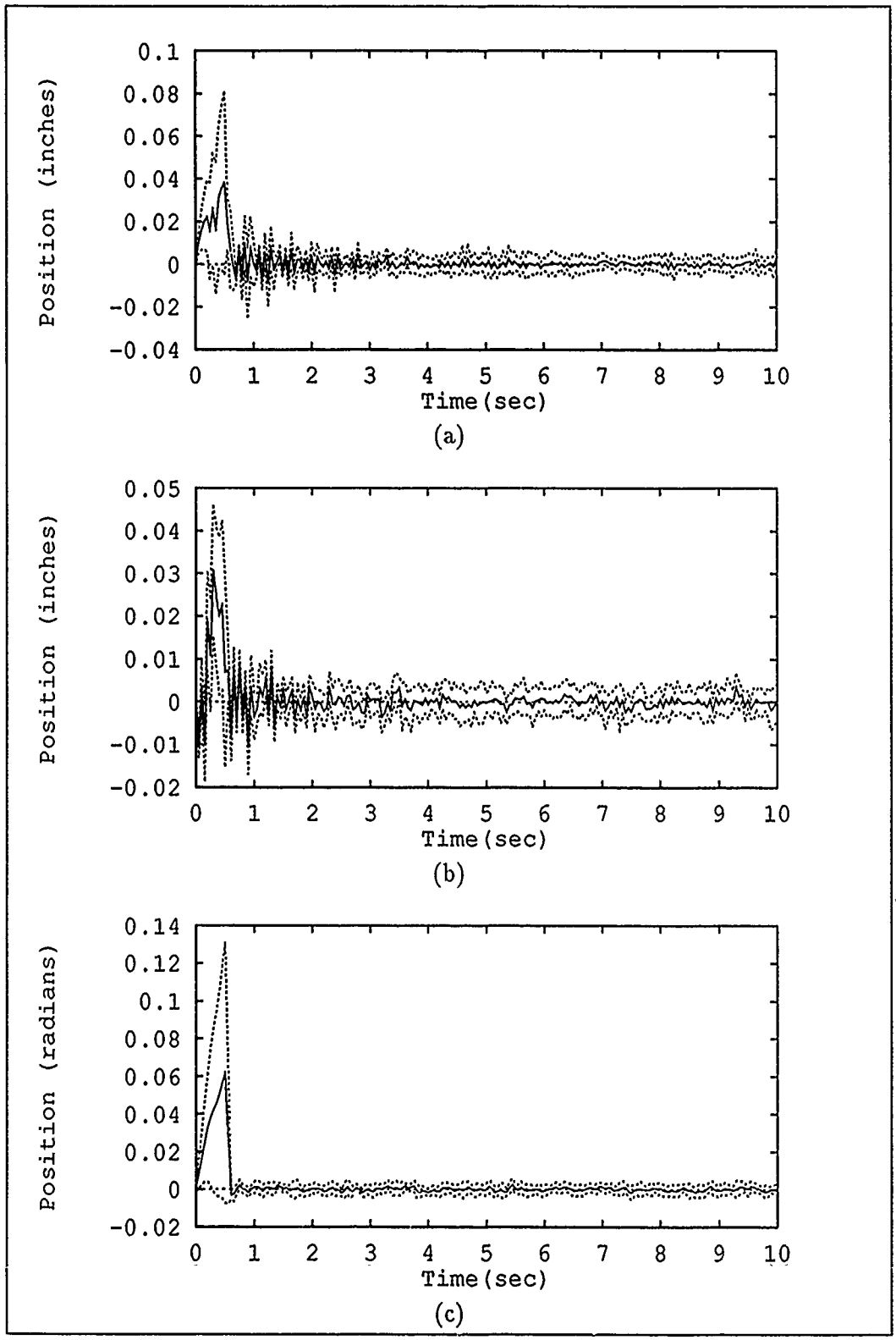

Figure 5.58. Tuned Single Controller Structure Positions (Mean $\pm$ One Standard Deviation) - True Parameter and Filter at (1,9). (a) Node 1 (b) Node 2 and (c) Node 7 Positions.

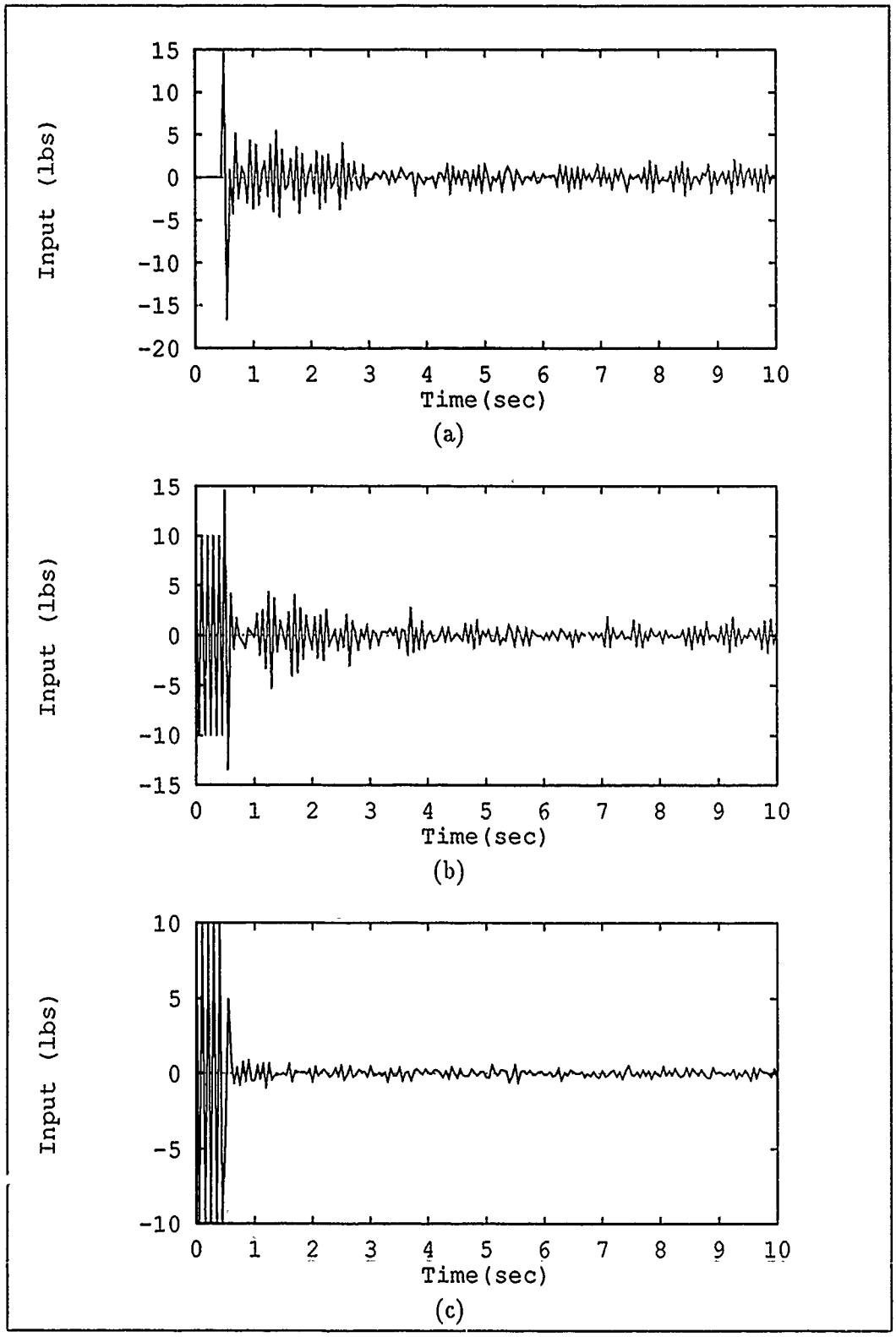

Figure 5.59. Tuned Single Controller Control Inputs (Sample Mean) - True Parameter and Controller at (1,9). (a) Node 1 (b) Node 2 and (c) Node 7 Actuators.

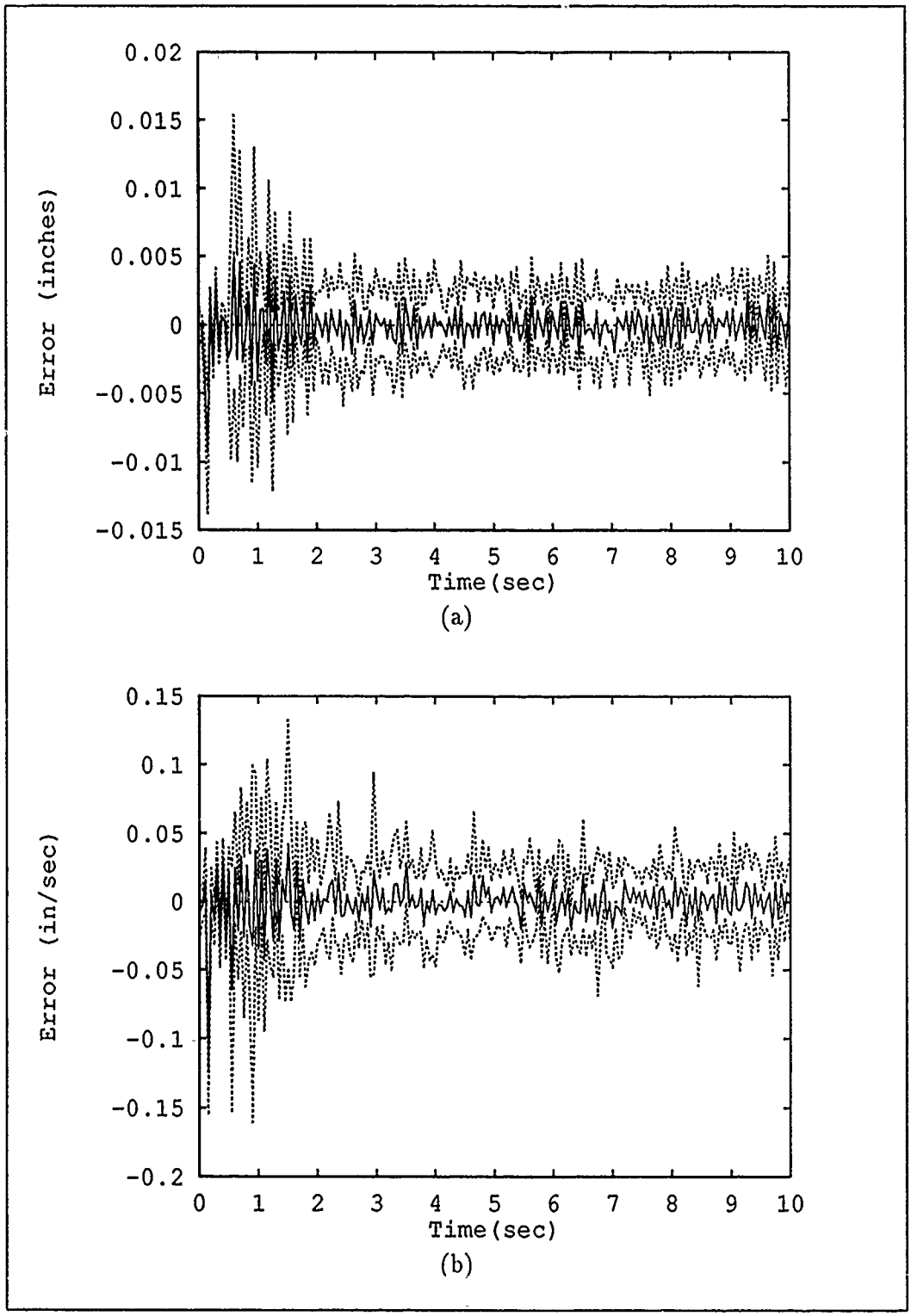

Figure 5.60. Moving-Bank MMAC Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter at (2,9) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors.

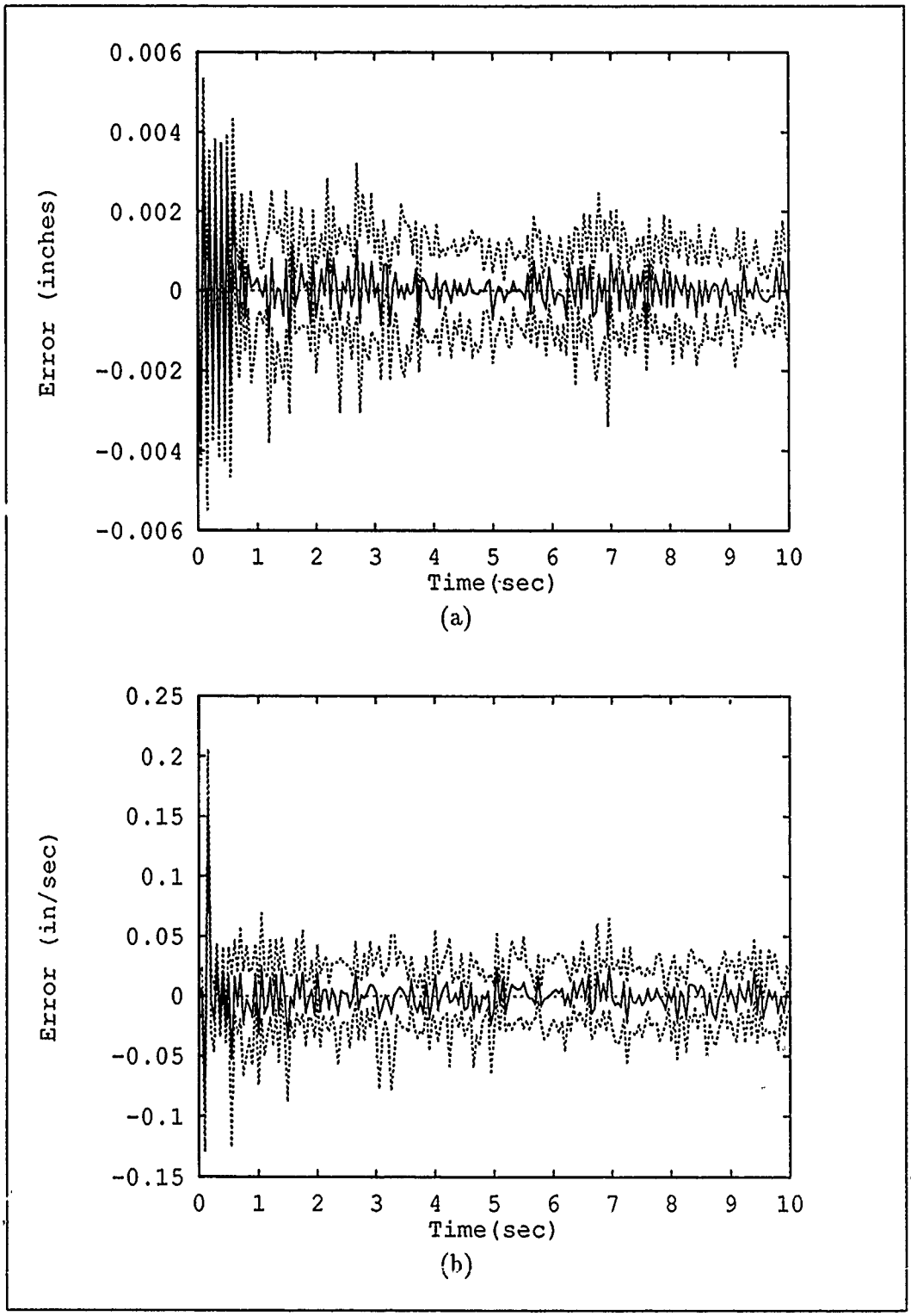

Figure 5.61. Moving-Bank MMAC Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter at (2,9) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors.

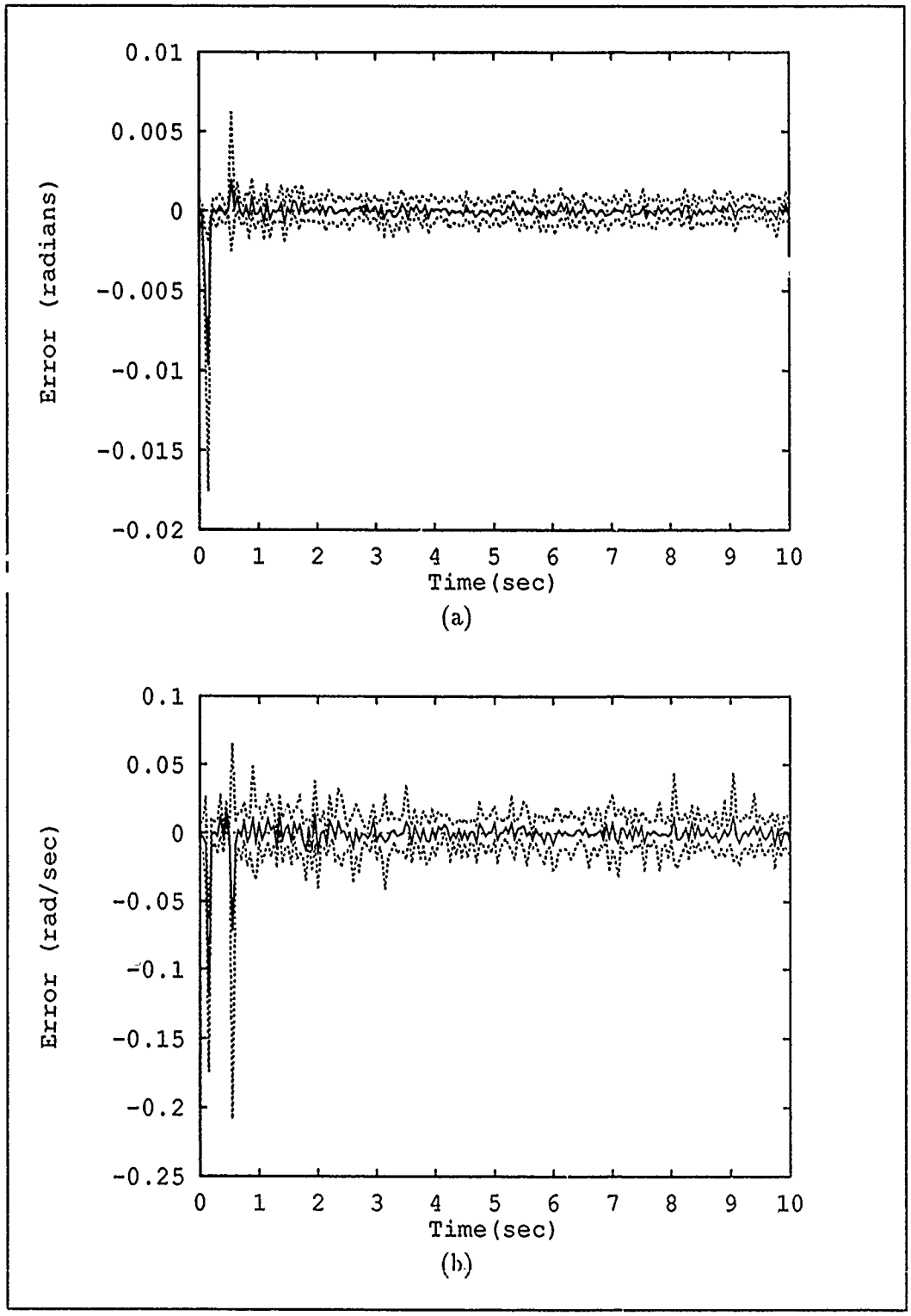

Figure 5.62. Moving-Bank MMAC Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter at (2,9) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors.

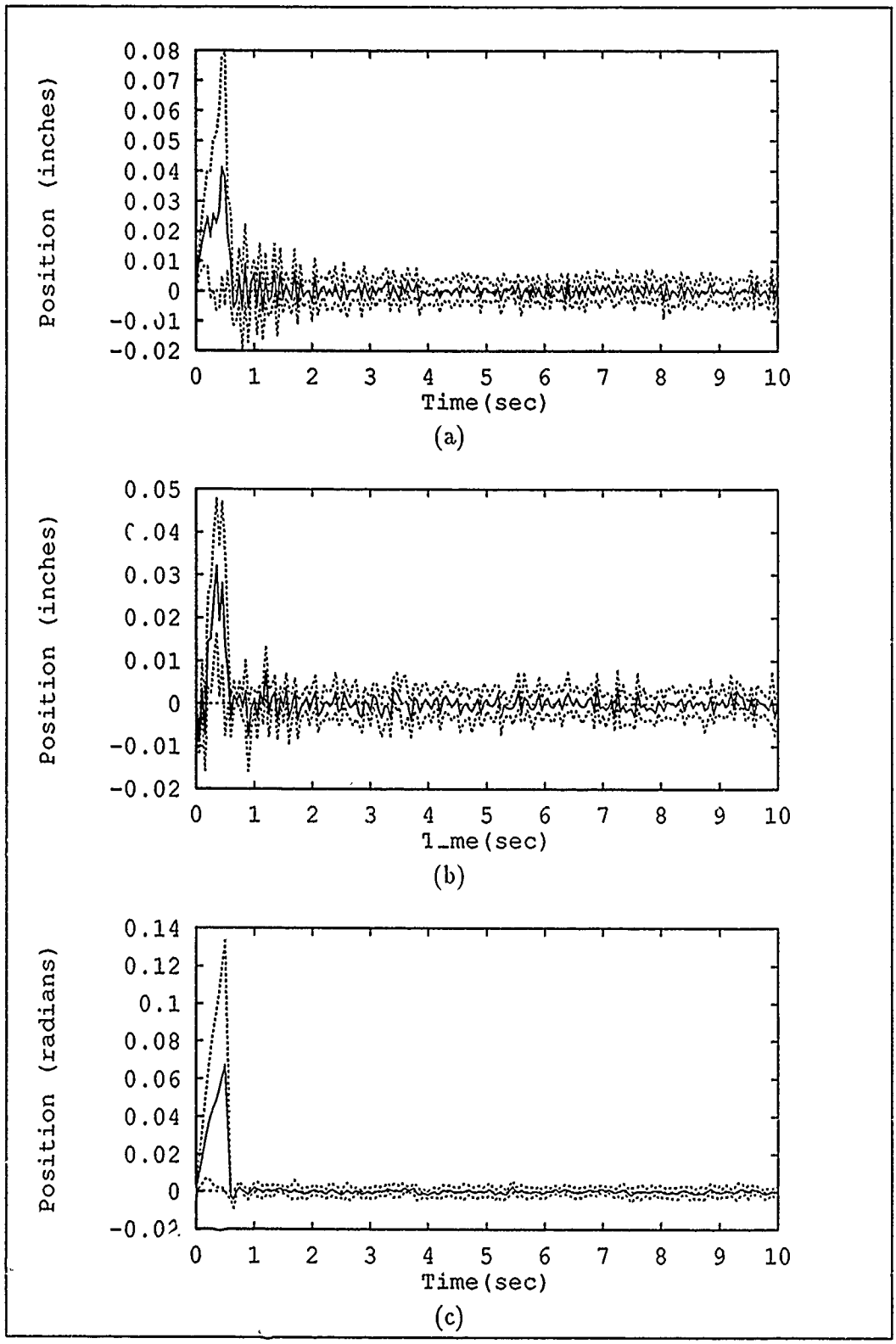

Figure 5.63. Moving-Bank MMAC Structure Positions (Mean $\pm$ One Standard Deviation) - True Parameter at (2,9) and Bank Initially at (5,5). (a) Node 1 (b) Node 2 and (c) Node 7 Positions.

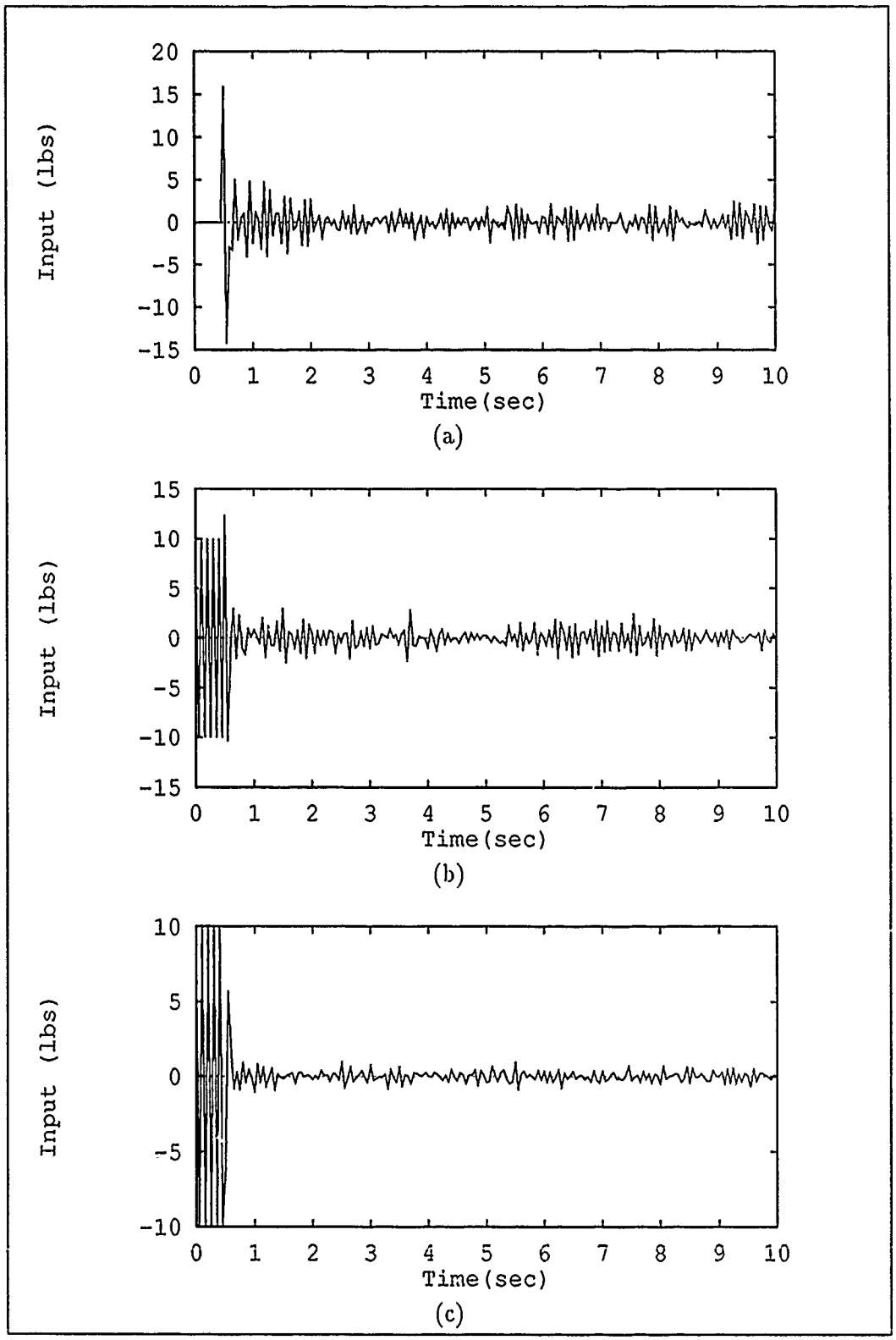

Figure 5.64. Moving-Bank MMAC Control Inputs (Sample Mean) - True Parameter at (2,9) and Bank Initially at (5,5). (a) Node 1 (b) Node 2 and (c) Node 7 Actuators.

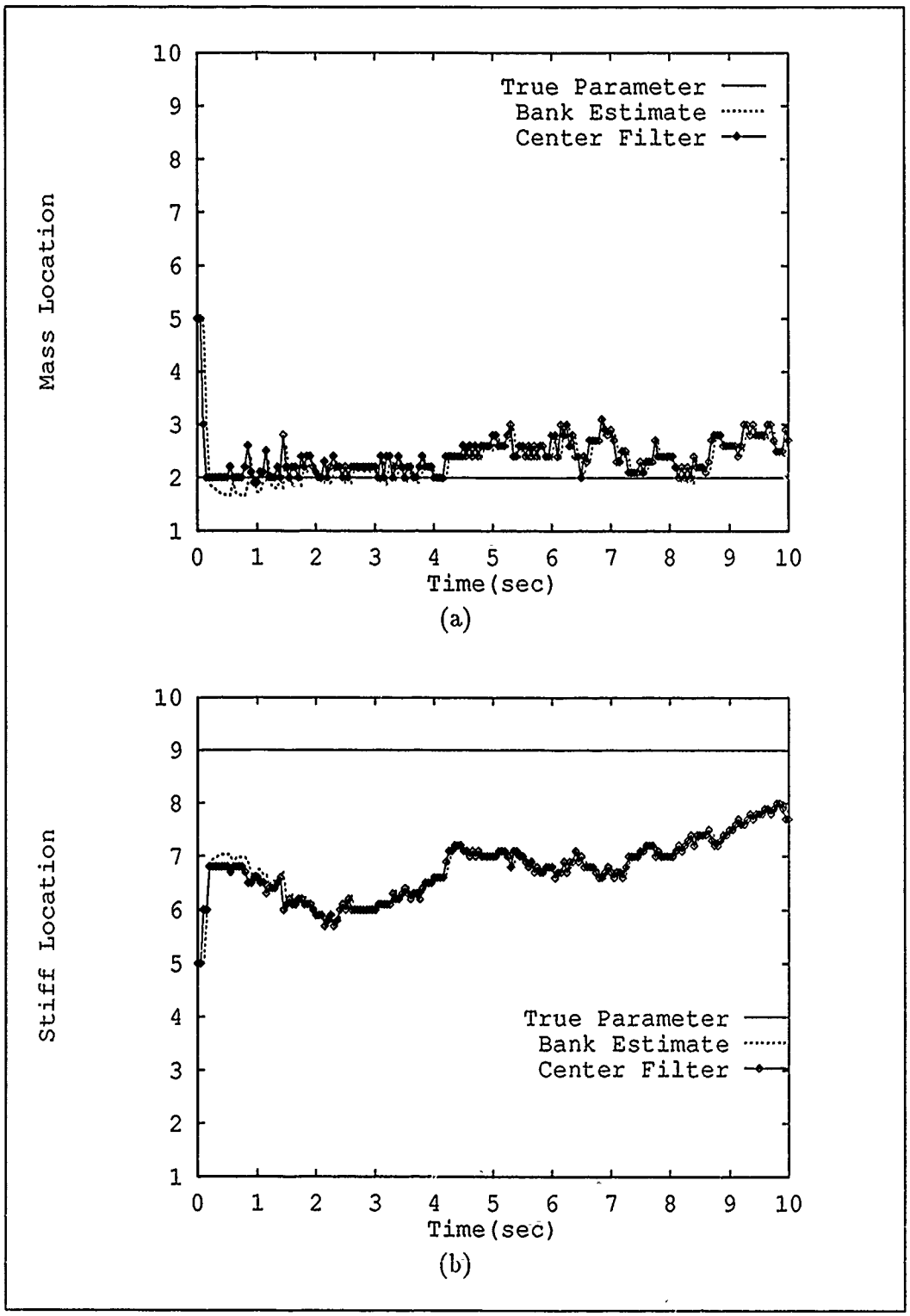

Figure 5.65. Moving-Bank MMAC Parameter Estimation (Sample Mean) and Actual Filter Bank Location - True Parameter at (2,9) and Bank Initially at (5,5). (a) Mass and (b) Stiffness Parameters.

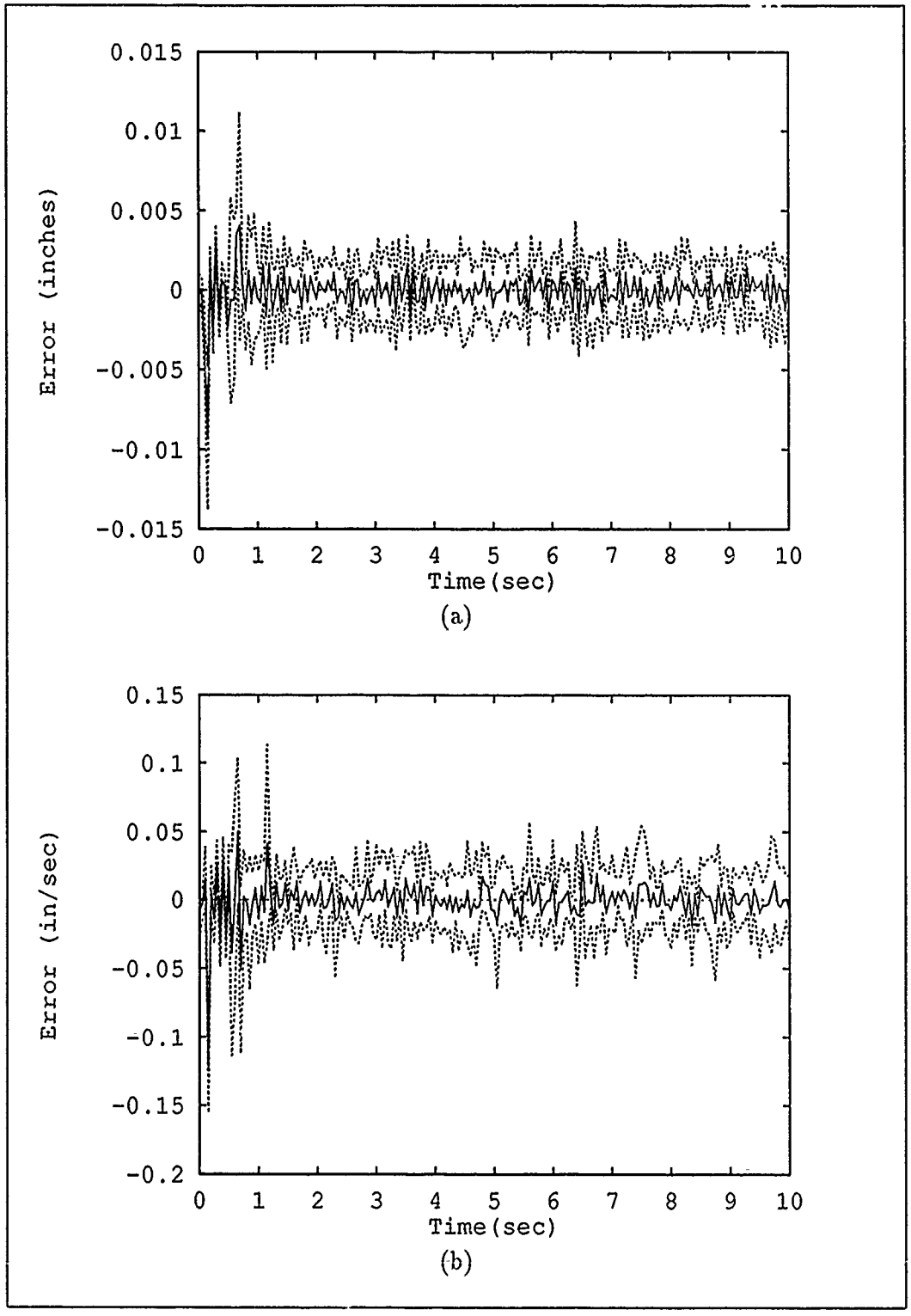

Figure 5.66. Moving-Bank Modified MMAC Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter at (2,9) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors.

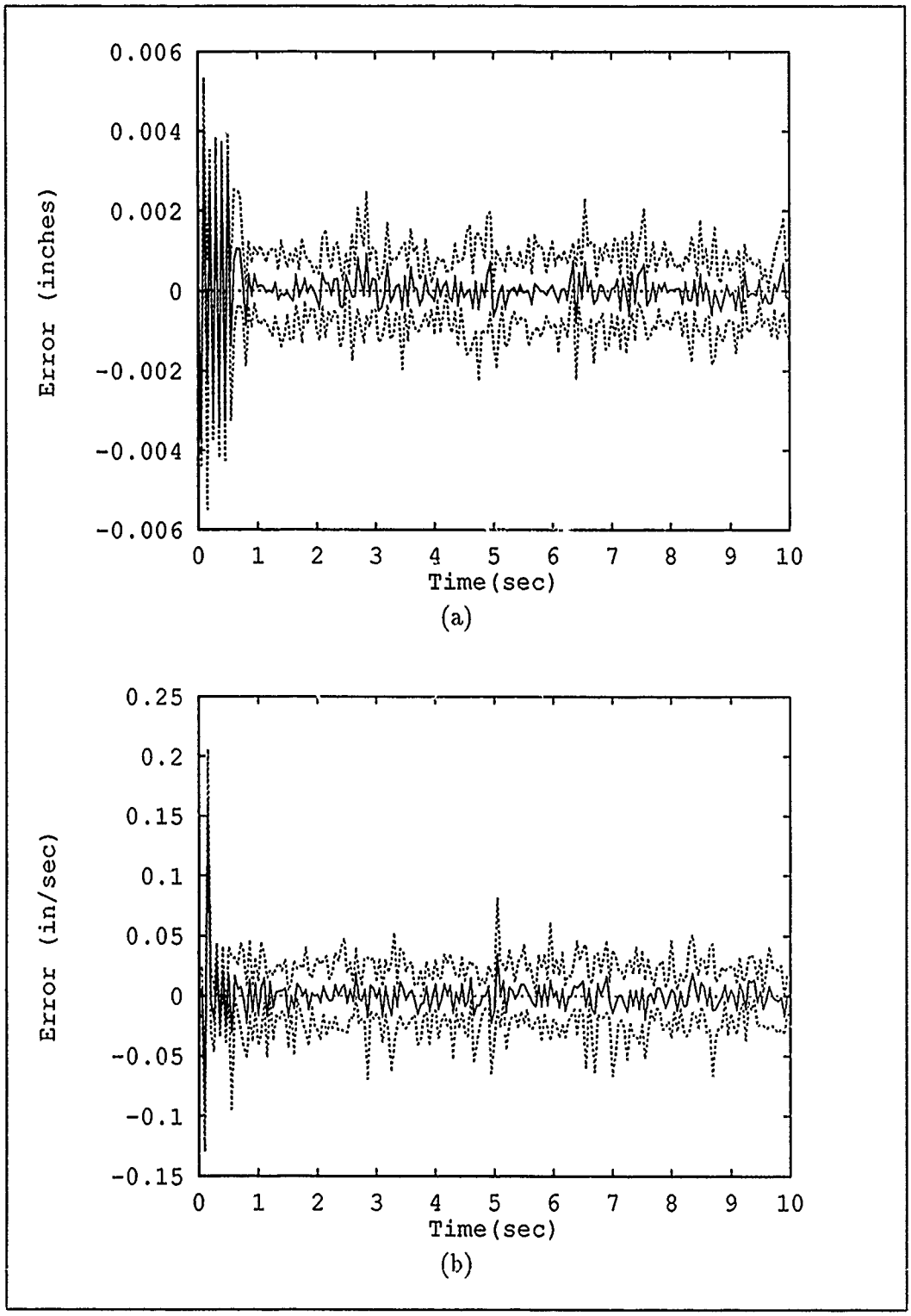

Figure 5.67. Moving-Bank Modified MMAC Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter at (2,9) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors.

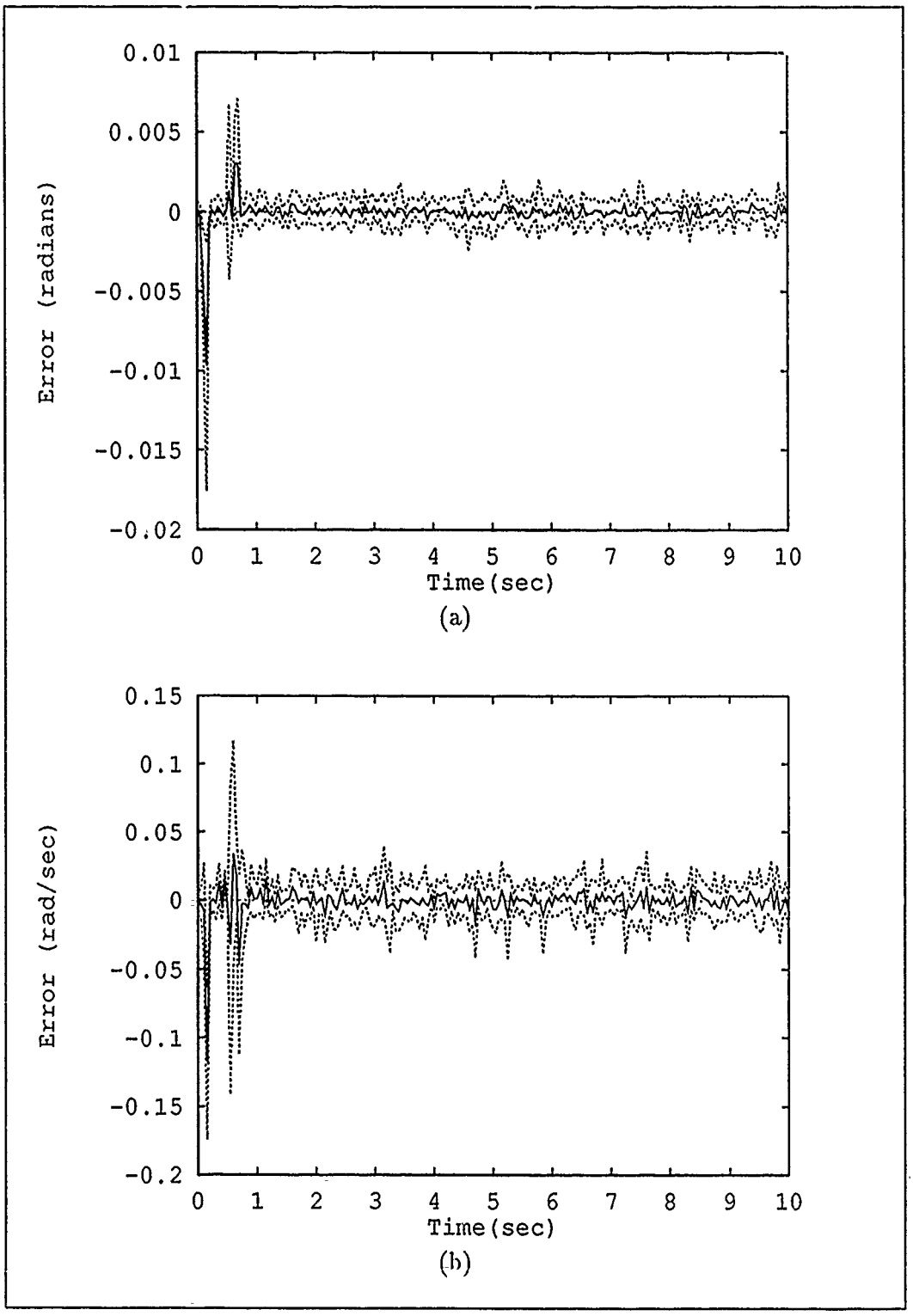

Figure 5.68. Moving-Bank Modified MMAC Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter at (2,9) and Bank Initially at (5,5). (a) Position and (b) Velocity Estimation Errors.

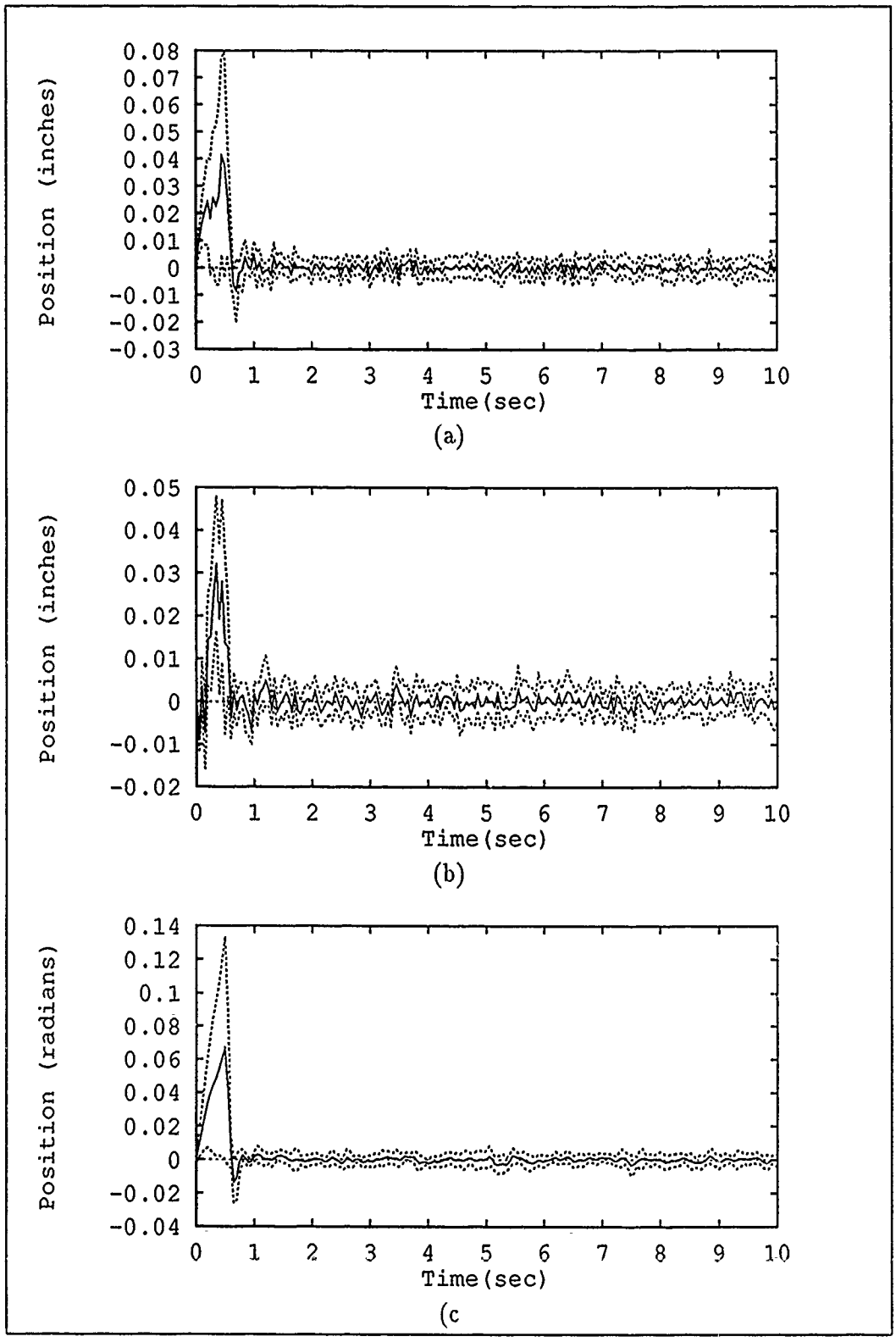

Figure 5.69. Moving-Bank Modified MMAC Structure Positions (Mean $\pm$ One Standard Deviation) - True Parameter at (2,9) and Bank Initially at (5,5). (a) Node 1 (b) Node 2 and (c) Node 7 Positions.

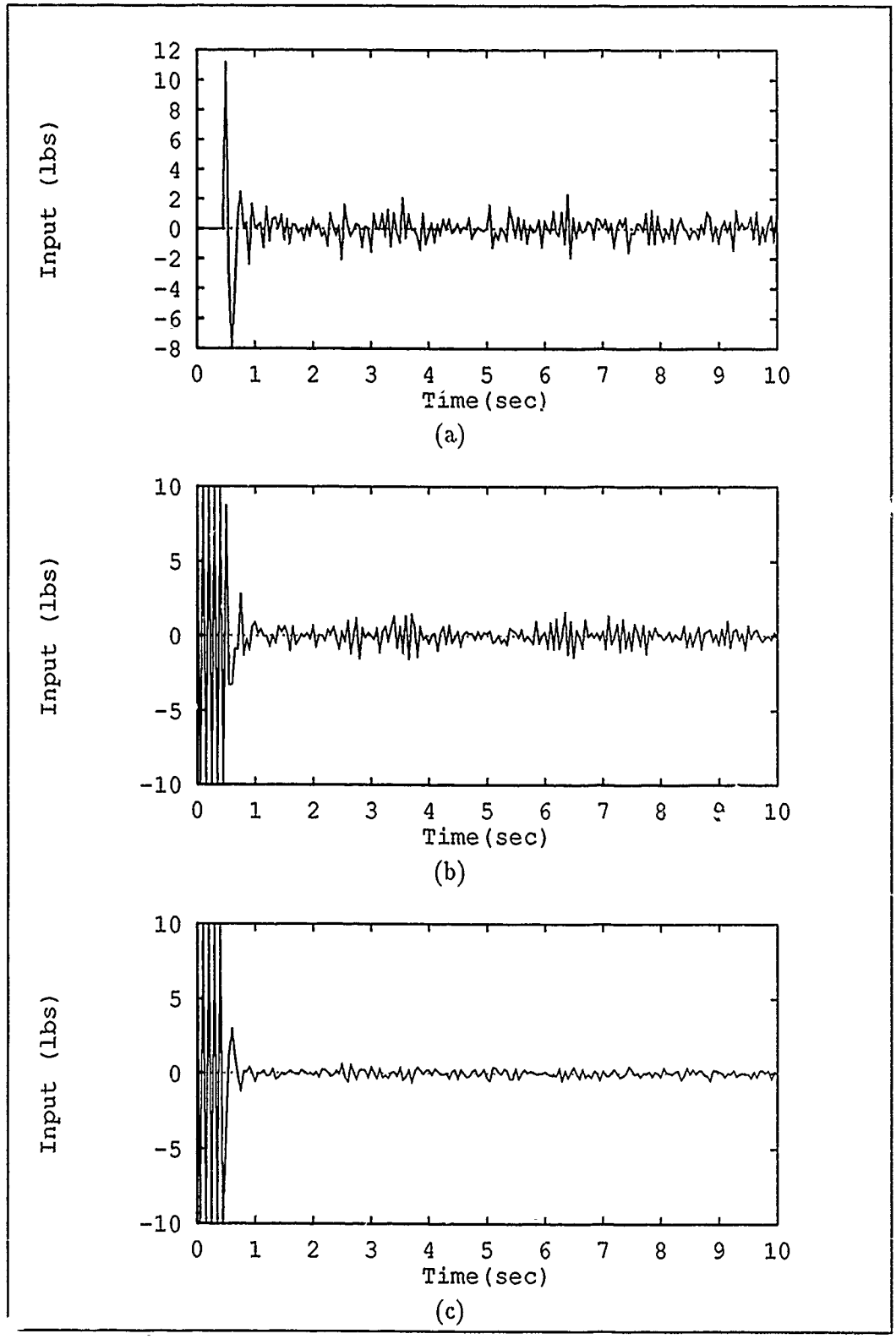

Figure 5.70. Moving-Bank Modified MMAC Control Inputs (Sample Mean) - True Parameter at (2,9) and Bank Initially at (5,5). (a) Node 1 (b) Node 2 and (c) Node 7 Actuators.

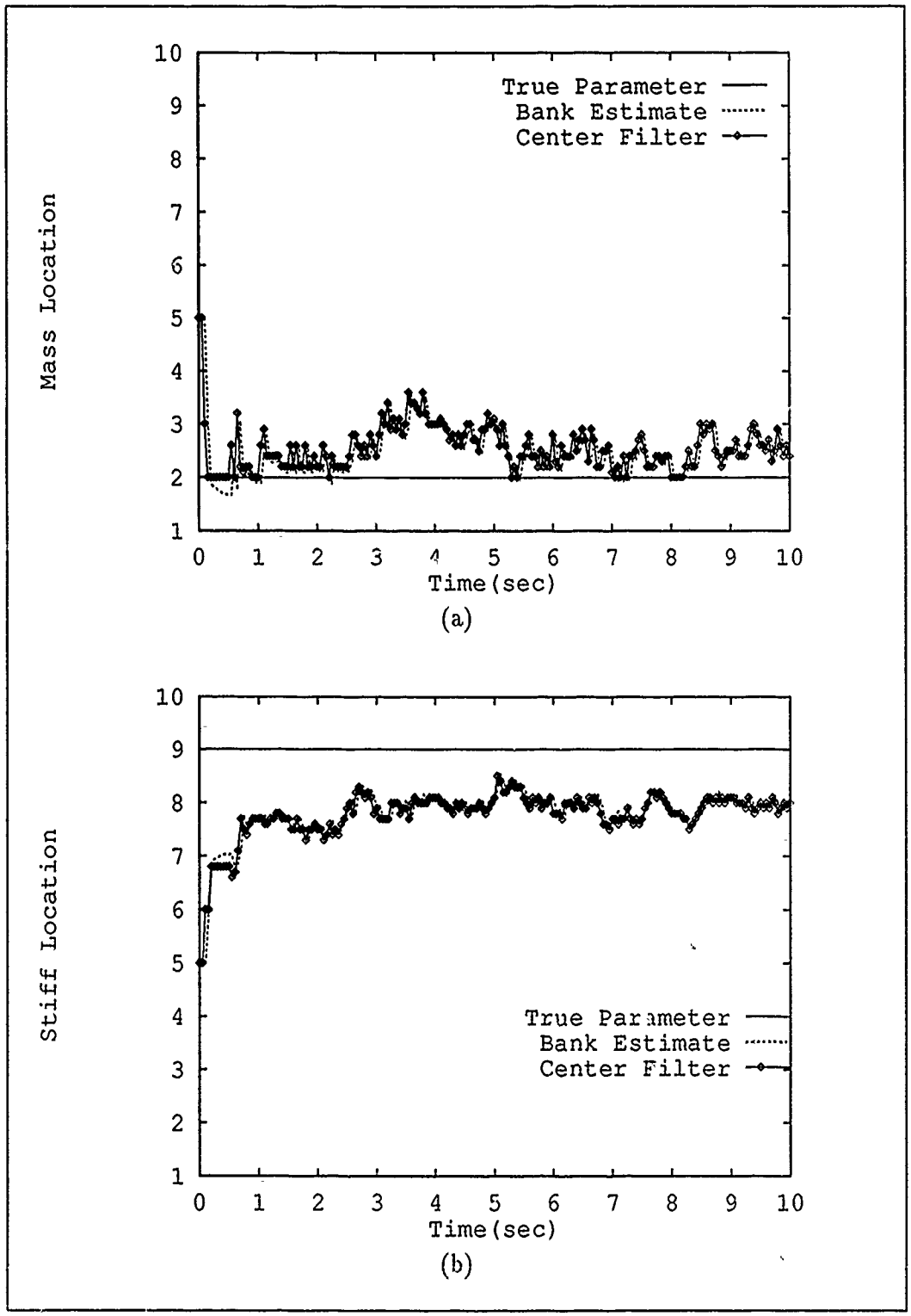

Figure 5.71. Moving-Bank Modified MMAC Parameter Estimation (Sample Mean) and Actual Filter Bank Location - True Parameter at (2,9) and Bank Initially at (5,5). (a) Mass and (b) Stiffness Parameters.

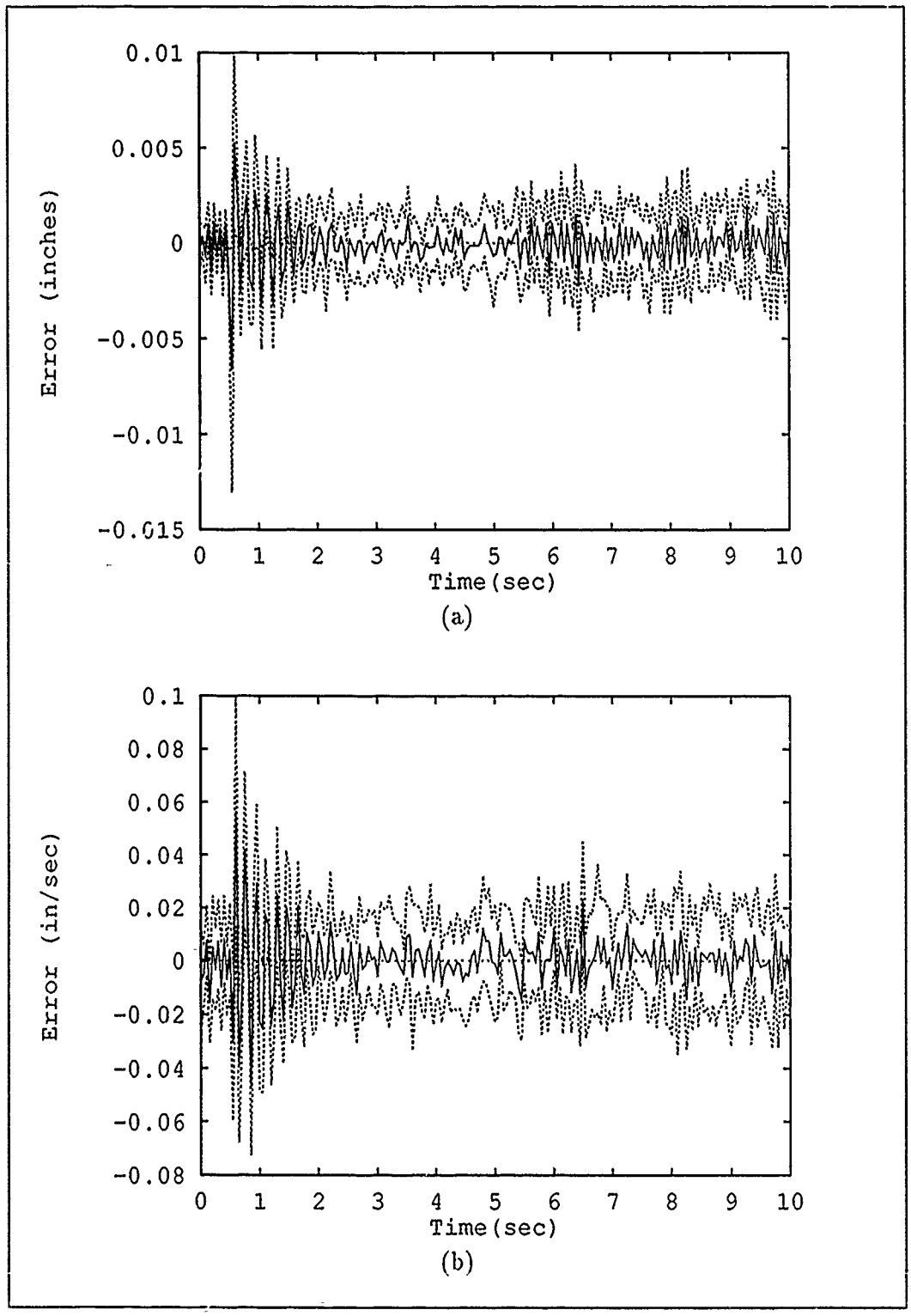

Figure 5.72. Single-Controller Benchmark Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter and Bank Initially at (5,5) with True Parameter Jump to (2,9). (a) Position and (b) Velocity Estimation Errors.

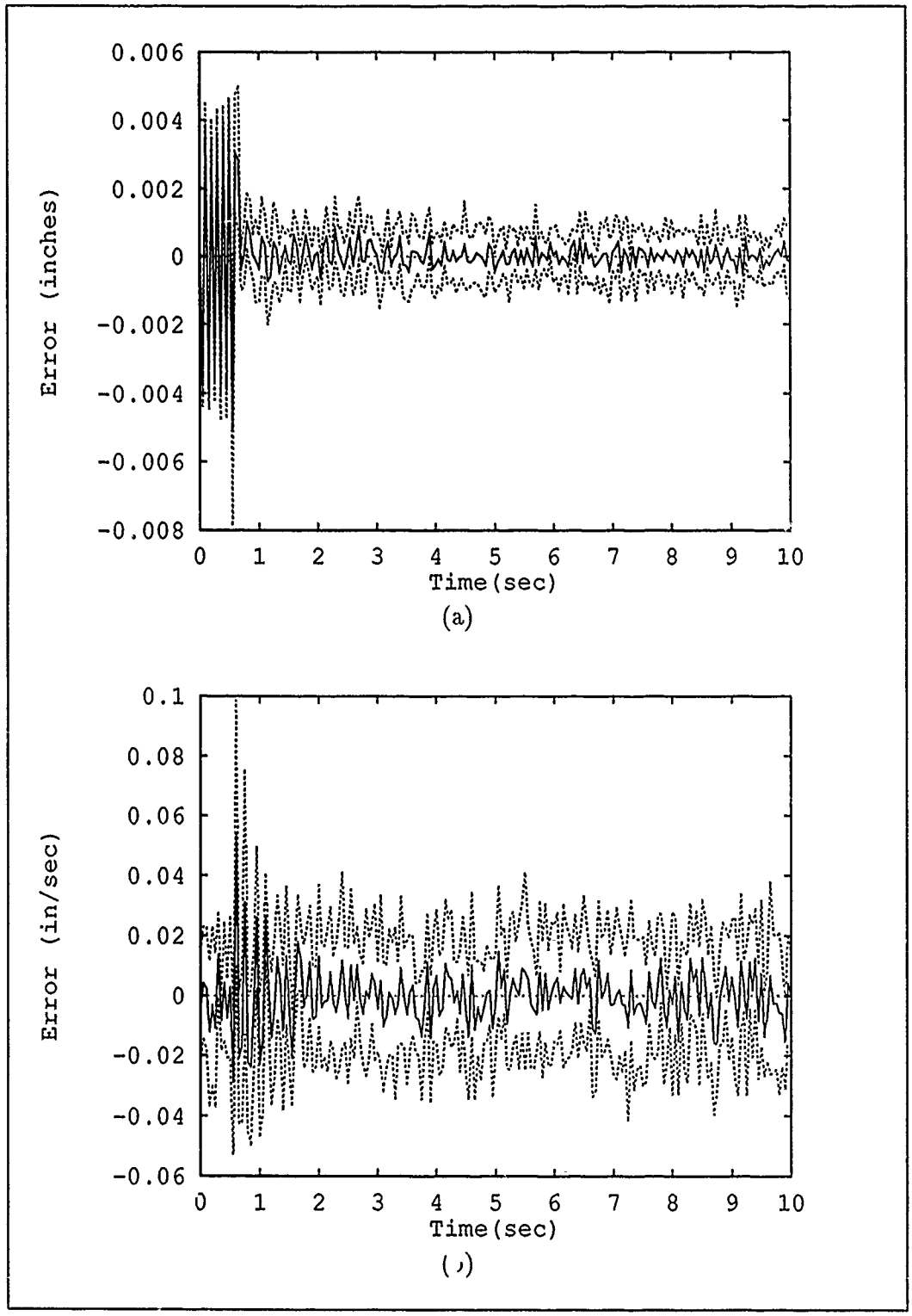

Figure 5.73. Single-Controller Benchmark Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter and Bank Initially at (5,5) with True Parameter Jump to (2,9). (a) Position and (b) Velocity Estimation Errors.

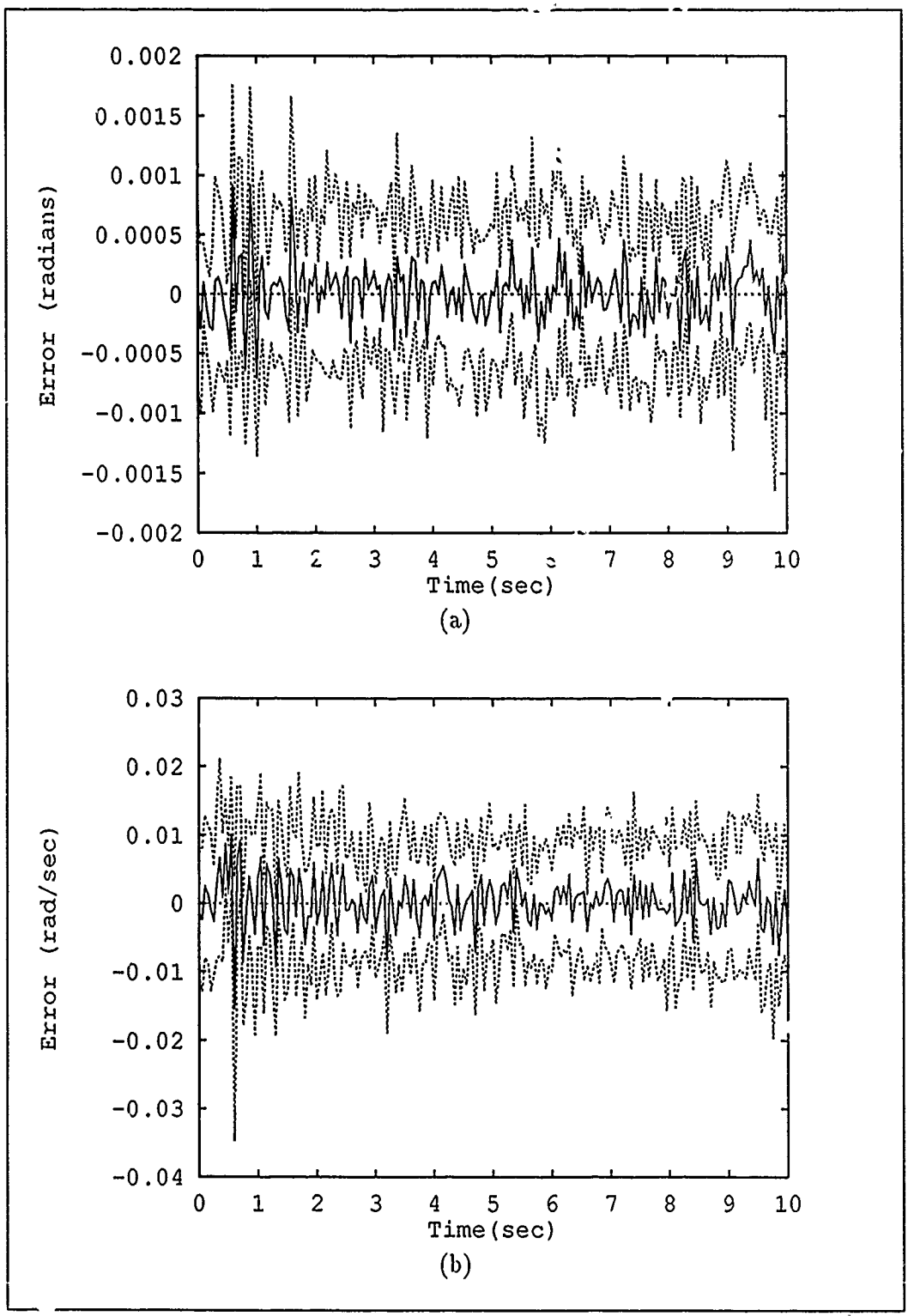

Figure 5.74. Single-Controller Benchmark Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter and Bank Initially at (5,5) with True Parameter Jump to (2,9). (a) Position and (b) Velocity Estimation Errors.

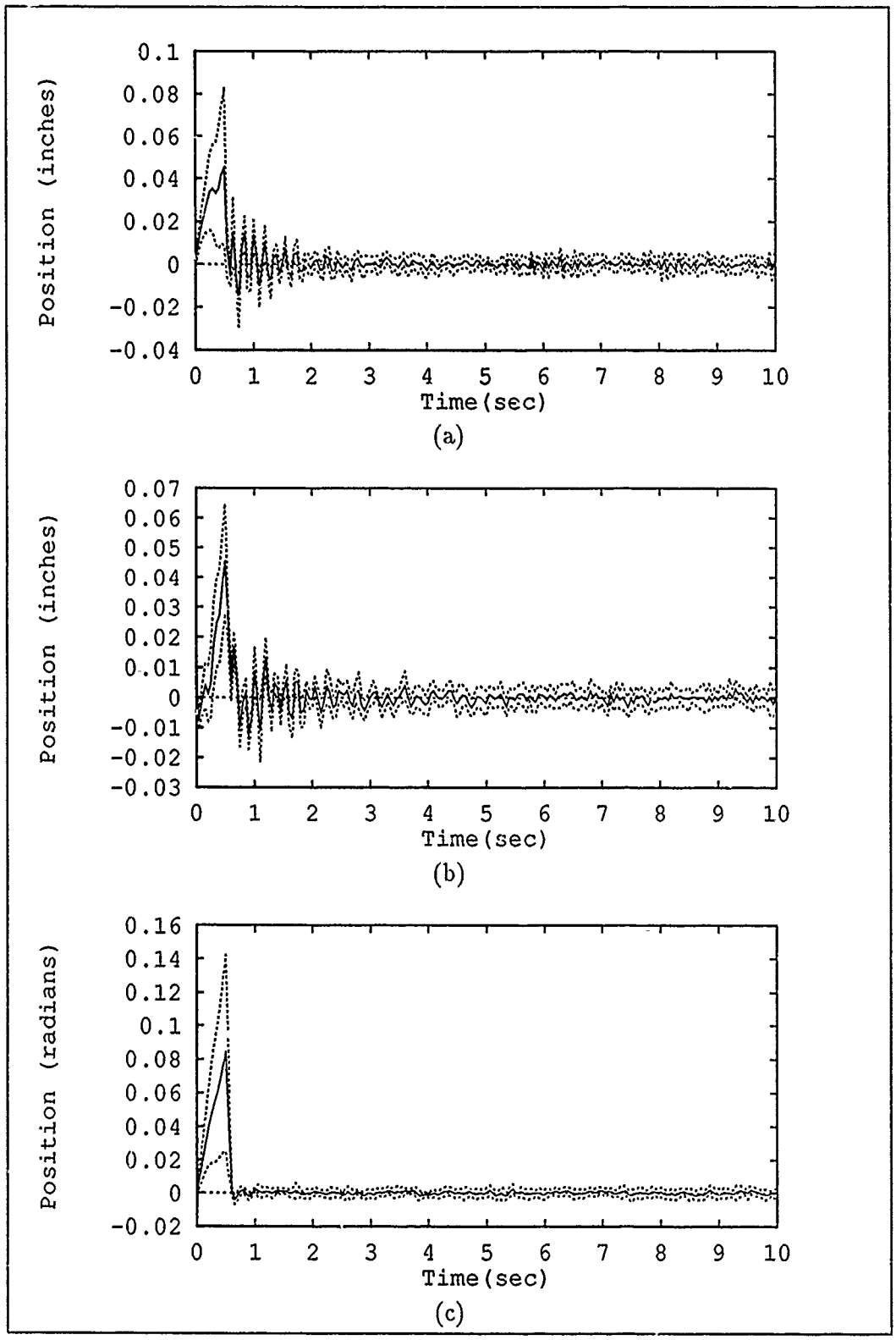

Figure 5.75. Single-Controller Benchmark Structure Positions (Mean $\pm$ One Standard Deviation) - True Parameter and Bank Initially at (5,5) with True Parameter Jump to (2,9). (a) Node 1 (b) Node 2 and (c) Node 7 Positions.

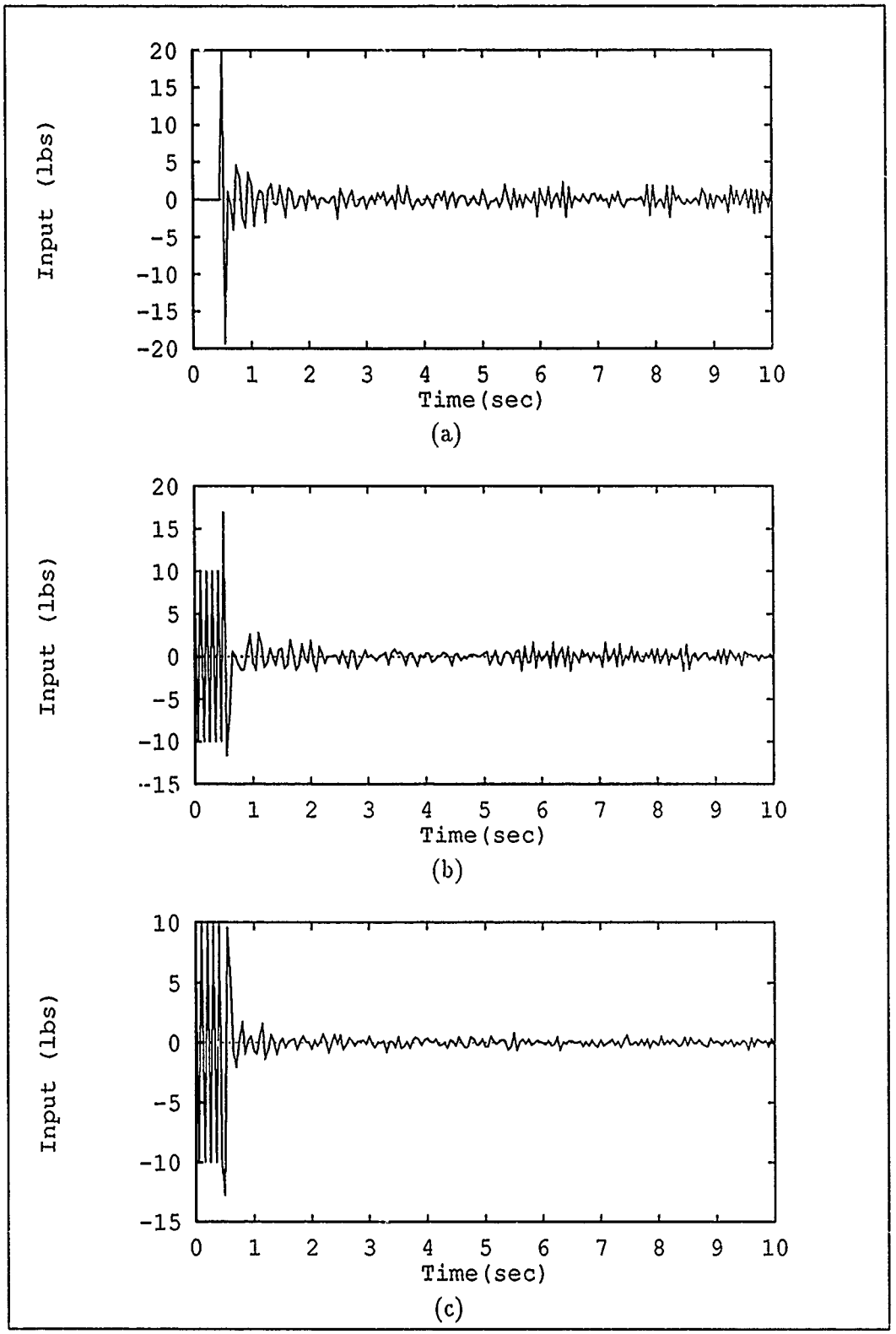

Figure 5.76. Single-Controller Benchmark Control Inputs (Sample Mean) - True Parameter and Bank Initially at (5,5) with True Parameter Jump to (2,9). (a) Node 1 (b) Node 2 and (c) Node 7 Actuators.

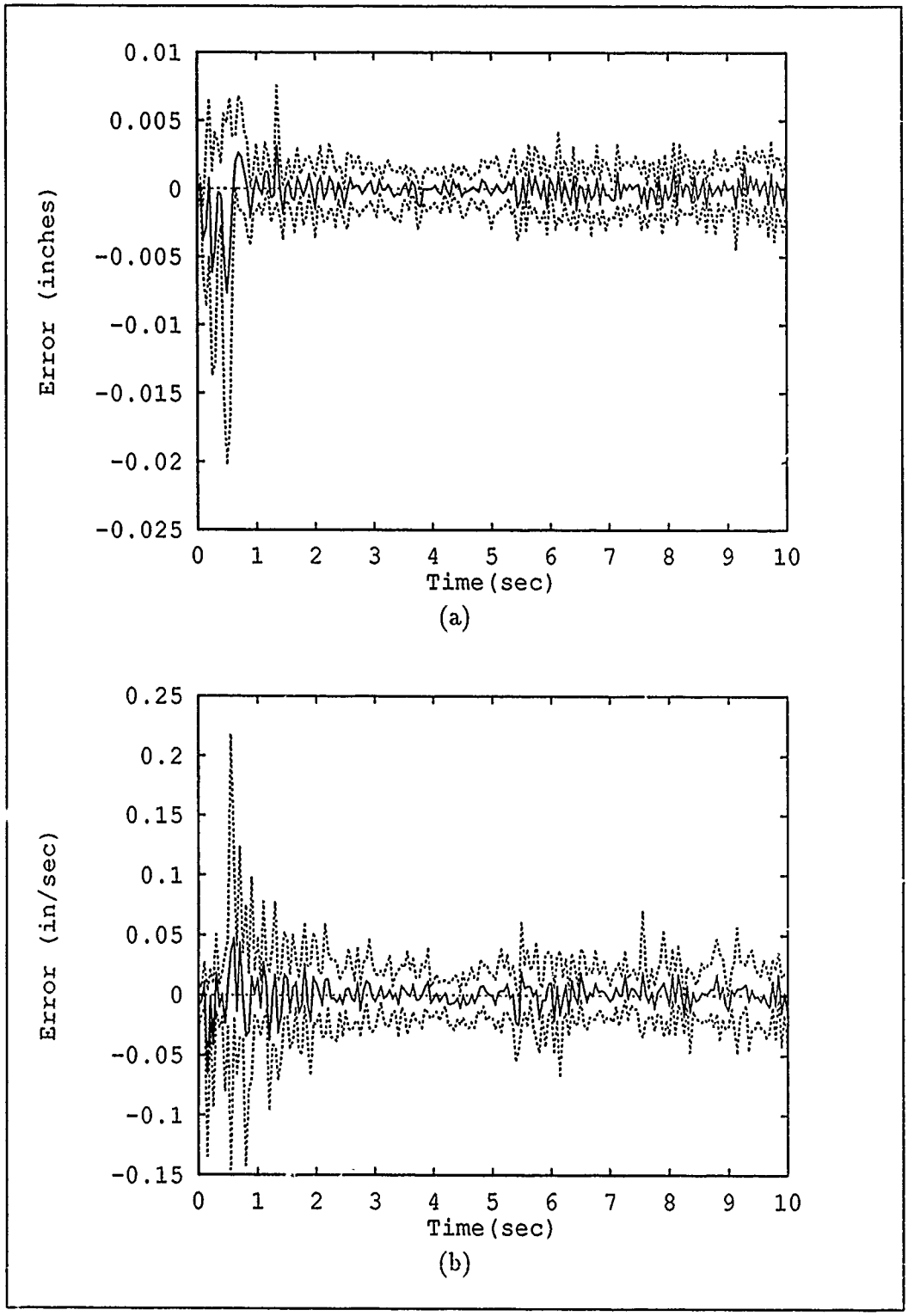

Figure 5.77. Moving-B nk Modified MMAC Estimation Errors (Mean $\pm$ One Standard Deviation), Node 1 - True Parameter and Bank Initially at (5,5) with True Parameter Jump to (2,9). (a) Position and (b) Velocity Estimation Errors.

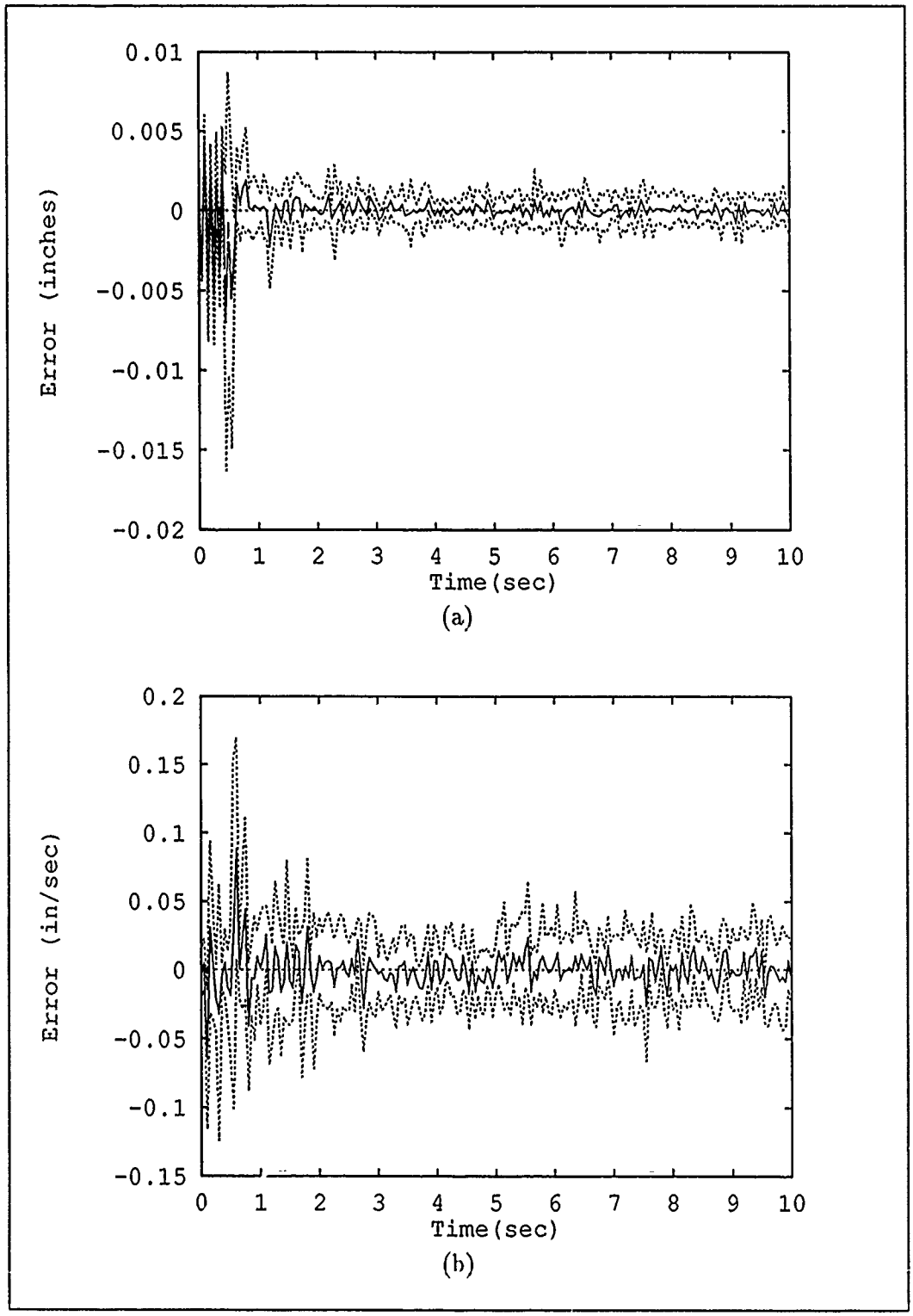

Figure 5.78. Moving-Bank Modified MMAC Estimation Errors (Mean $\pm$ One Standard Deviation), Node 2 - True Parameter and Bank Initially at (5,5) with True Parameter Jump to (2,9). (a) Position and (b) Velocity Estimation Errors.

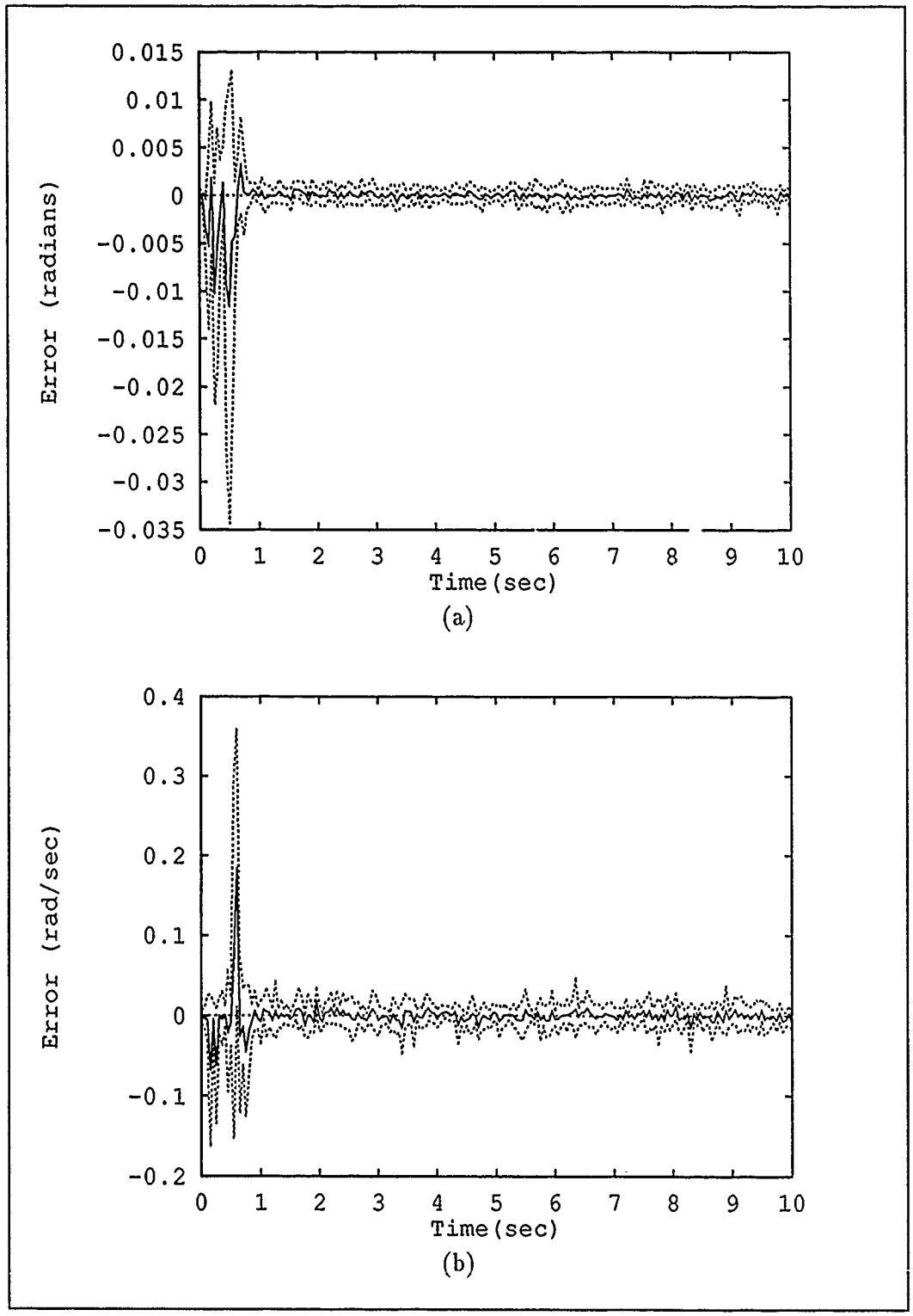

Figure 5.79. Moving-Bank Modified MMAC Estimation Errors (Mean $\pm$ One Standard Deviation), Node 7 - True Parameter at (2,9) and Bank Initially at (5,5). (a) Position and (o) Velocity Estimation Errors.

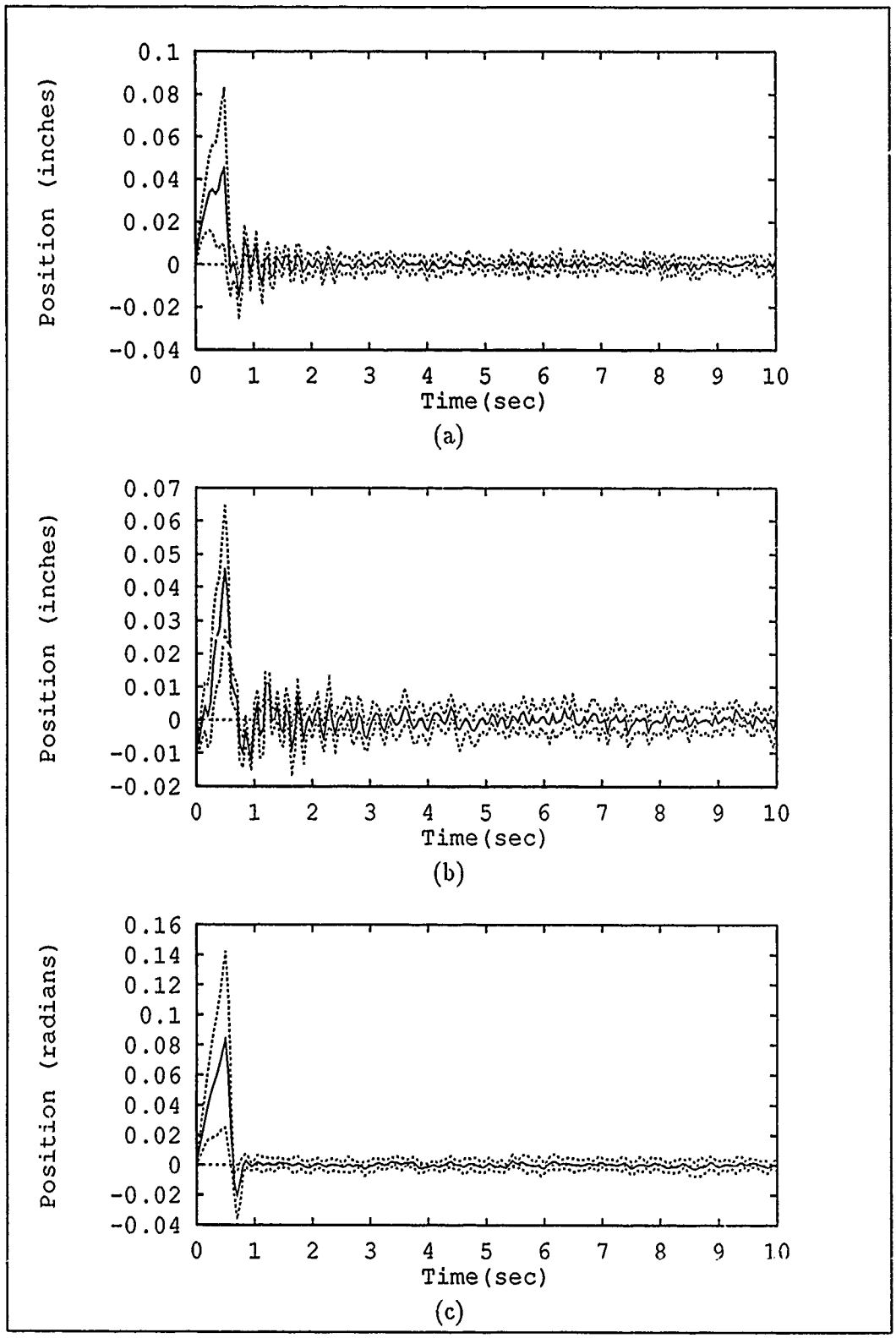

Figure 5.80. Moving-Bank Modified MMAC Structure Positions (Mean $\pm$ One Standard Deviation) - True Parameter and Bank Initially at (5,5) with True Parameter Jump to (2,9). (a) Node 1 (b) Node 2 and (c) Node 7 Positions.

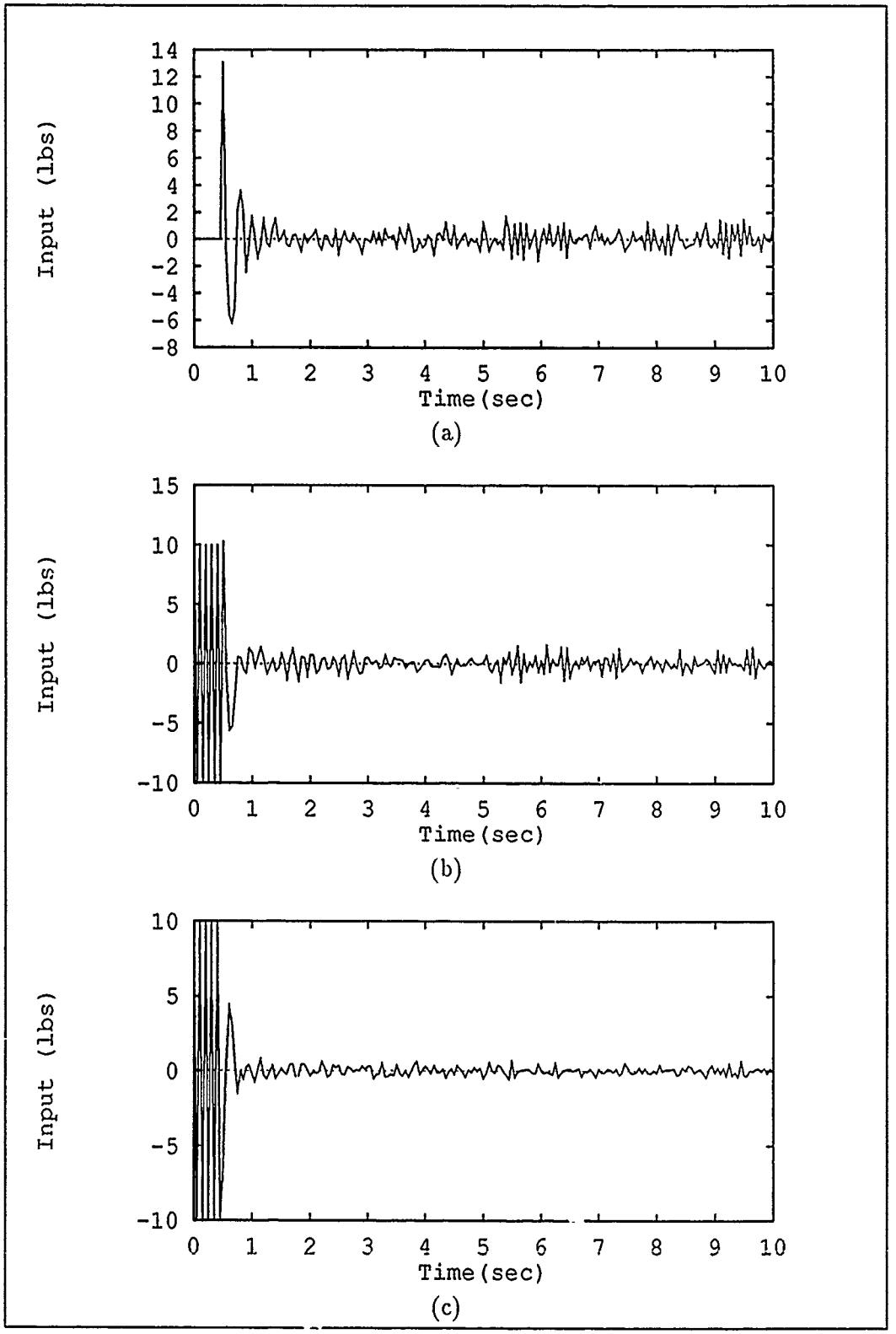

Figure 5.81. Moving-Bank Modified MMAC Control Inputs (Sample Mean) - True Parameter and Bank Initially at (5,5) with True Parameter Jump to (2,9). (a) Node 1 (b) Node 2 and (c) Node 7 Actuators.

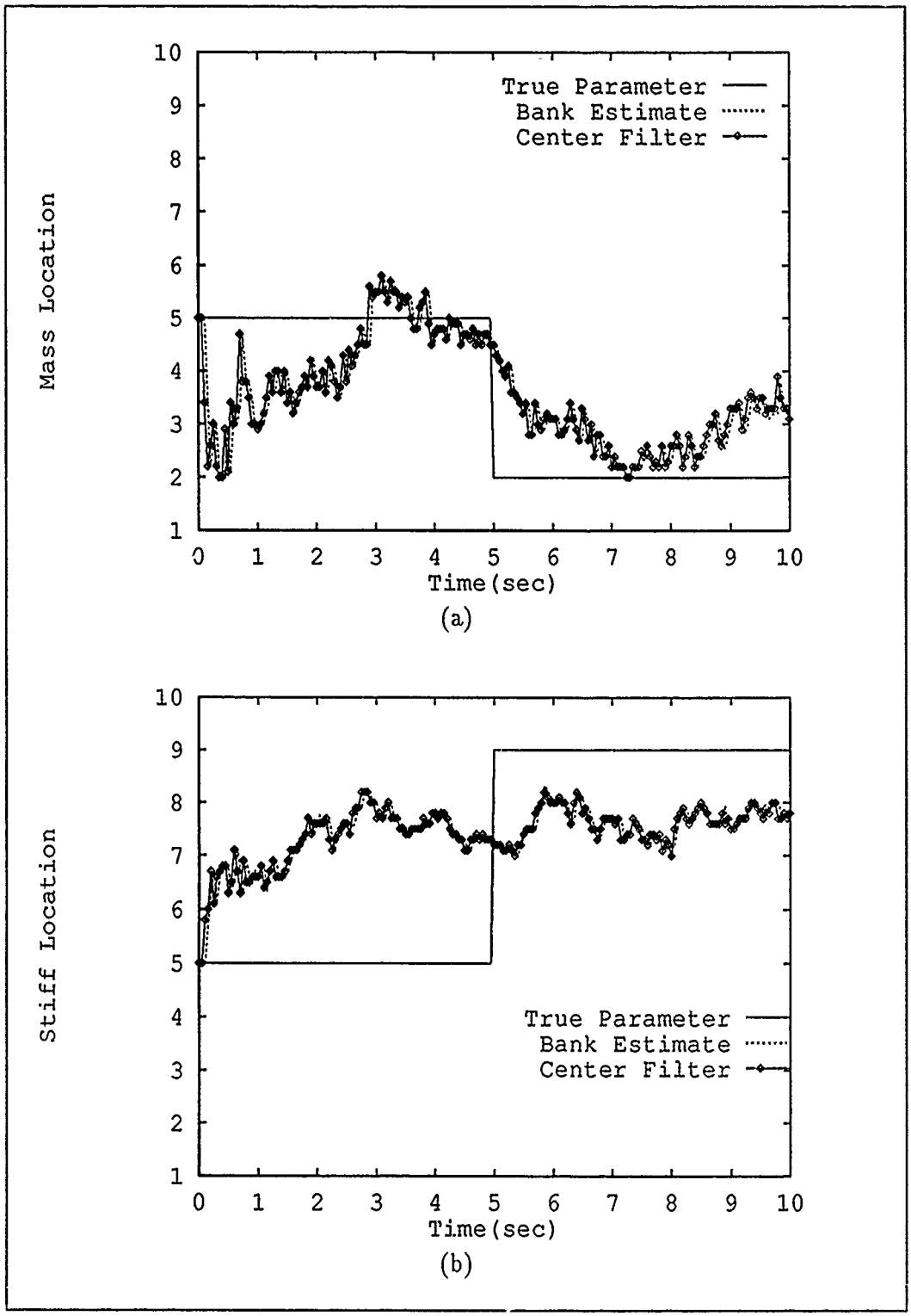

Figure 5.82. Moving-Bank Modified MMAC Parameter Estimation (Sample Mean) and Actual Filter Bank Location - True Parameter and Bank Initially at (5,5) with True Parameter Jump to (2,9). (a) Mass and (b) Stiffness Parameters.

VI. Conclusions and Recommendations

6.1 Introduction

The purpose of this research was to determine the full capabilities of moving-bank adaptive estimation and control algorithms using a six-state filter model against a 24-state filter model. The measure of performance was the estimation and control of a two-bay truss at various physical locations on the structure. This chapter preserts conclusions based on the results presented in Chapter 5 and recommendations for future research.

6.2 Conclusions

The modification studies, which became a substantial part of this study, refined the implementation software and demonstrated significant improvement in the performance of both the estimation and control algorithms. The modification results established a baseline for performance significantly different from past research using the same filter/controller and truth models. For the non-adaptive artificially informed single-filter studies, the estimation errors no longer showed trends of increasing parameter uncertainty. For the non-adaptive single-controller studies, the modified control input matrix allowed the output from the actuator at the hub to relieve the load requirements on the truss actuators.

Proper tuning of the Kalman filters and the LQG controllers provided significant performance improvement. For the estimator study, proper determination of the filter noise statistics, dynamics noise strength $\mathbf{Q}$ and measurement noise covariance $\mathbf{R}$, improved performance by an order of magnitude for the truss velocities as well as for the rigid body position. This performance was obtained without conservative tuning approaches, and thus while successfully avoiding "masking" the differences between the multiple models. For the LQG study, controllers based on higher natural frequencies for the bending modes performed significantly better after the appropriate determination of the state, $\mathbf{X}$, and control, $\mathbf{U}$, weighting matrices for the quadratic cost function. Investigation of a worst-case non-adaptive benchmark for the controllers resulted in stability problems. Therefore, for the non-adaptive single controller benchmark, a region of the parameter space was determined about a nominal parameter value beyond which the control algorithms caused

stability problems. Underestimating the true natural frequencies of the bending modes was found to be critical in inducing instabilities.

The moving-bank logic study indicated that using the filter-computed residual covariance matrix, $\mathbf{A}_k$, in the probability density computation required in calculating the hypothesis conditional probabilities (p_k's) resulted in poor performance of the bank. Poor performance was also demonstrated when this matrix was incorporated into the residual likelihood quotient for residual-monitoring move logic as well as for bank expansion. Using the best methodology investigated in this research, maximum entropy with identity assumed covariance (ME/I) computations and parameter position estimate monitoring, the moving-bank estimator compared poorly to non-adaptive artificially informed benchmarks. The incorporation of the residual covariance in the probability density computation resulted in numerical precision problems for the moving-bank controller performance. However, using ME/I computations and parameter position estimate monitoring and a modified MMAC approach, the moving-bank controller provided performance nearly as good as a non-adaptive artificially informed benchmark.

The space discretization study, with densely discretized parameters in the high natural frequency region of the space, did not provide any improvement for the moving-bank estimation or control simulations. These results are primarily due to the relatively fine discretization of the parameter space and the ability of the moving-bank algorithms to place the bank within adequate proximity of the true parameter.

For the case of varying parameters, the controller results show that the algorithms provide control nearly comparable to an artificially informed benchmark. The results for the jump parameter showed that the performance was better when the finely discretized bank was only allowed to move rather than expand and contract for parameter acquisition. For the problem investigated here, with only 100 discretized values closely spaced, the acquisition time with a fine-bank move was acceptable. (The additional computation associated with the expansion and contraction logic was not warranted). However, for applications where the number of discrete parameters is substantially more or the discretization is finer, the performance of the expansion and contraction algorithms may be substantially better than those that do not allow alteration of bank size.

The moving-bank MMAE/MMAC algorithms provide stabilizing control over the two-bay truss structure investigated in this research. However, there remain several topics of research that should be investigated to determine the full capabilities of the algorithms. The next section outlines recommendations that should be performed in future research.

6.3 Recommendations

The scope of this and past research performed was sufficient to cover all areas of interest for the present structure, filter/controller models, and truth models. Future research should be continued to address the full capabilities of the algorithms on a more complex space structure. The following recommendations are made for future research:

1. Modify the current two-bay structure by adding at least a third bay, or adopt a model representative of an entire flexible space structure. This would entail development of an appropriate truth model using finite element analysis with possible addition of sensors and actuators. Special attention should be paid to the development of the measurement and control input matrices in physical coordinates.
2. Perform an analysis for determining the lowest order filter/controller capable of adequately controlling the structure. Investigate performance of moving-bank MMAE/MMAC algorithms in this new application, considering the same issues as explored in this research. Continue with a physical interpretation of the structure positions for determining the algorithm performance.

Appendix A. *Rotating Two-Bay Truss Truth and Filter Model Matrices*

This appendix lists (1) the nominal mass, M, and stiffness matrices, K, (2) the 24-state truth model matrices, (3) the reduced order six-state filter and controller design model matrices, and (4) the implemented error vector formulation. The matrices developed here are for the nominal structure, i.e., no scaling of the mass and stiffness matrices. (Recall that the two parameters are scalar multipliers on these matrices.) In addition, the truth model matrices are associated with the truth model vector described by Equation (3.28).

Mass and Stiffness Matrices

The development of the truth model was provided in Sections 3.3.1 and 3.3.2. The mass and stiffness matrices, which describe the system model, were obtained using finite element analysis [28]. Finite element analysis models a structure as consisting of a finite number of nodes connected by elements. The finite element program produces mass and stiffness matrices with dimension equal to the number of degrees of freedom (DOF's) associated with the model. Each row of the mass and stiffness matrices is associated with a specific node and DOF. For the rotating two-bay truss in Figure 1.8, row 1 of each mass and stiffness matrix is associated with the x-axis DOF of node 1. Each node has three translational DOF's. Only planar motion is being considered; therefore, the nodes are modeled with only two DOF's. For this research, node 7 is fixed. Therefore, all three DOF's associated with this mode are eliminated, thereby reducing the dimensionality of the mass and stiffness matrices to 12 states. Accounting for both position and velocity of each of the 12 states yields a 24-state truth model. *These are the nominal matrices from which parameter variations are considered for this research.* Parameter variations are obtained by scaling these nominal matrices. The $12 - by - 12$ mass and stiffness matrices for the specifications previously defined are provided on the following pages [8:99-105], [11:205- 210], [26:215-223].

Mass Matrix

Row 1	.118E+4	.196E+3	0.0	0.0	-.642E+3	0.0
	-.546E+3	-.196E+3	0.0	0.0	0.0	0.0
Row 2	.196E+3	.626E+3	0.0	-.555E+3	0.0	0.0
	-.196E+3	-.707E+2	0.0	0.0	0.0	0.0
Row 3	0.0	0.0	.118E+4	-.196E+3	-.546E+3	.196E+3
	-.642E+3	0.0	0.0	0.0	0.0	0.0
Row 4	0.0	-.555E+3	-.196E+3	.626E+3	.196E+3	-.707E+2
	0.0	0.0	0.0	0.0	0.0	0.0
Row 5	-.642E+3	0.0	-.546E+3	.196E+3	.401E+4	.669E+2
	0.0	0.0	-.209E+4	0.0	-.732E+3	-.263E+3
Row 6	0.0	0.0	.196E+3	-.707E+2	.669E+2	.721E+3
	0.0	-.555E+3	0.0	0.0	-.263E+3	-.9487E+2
Row 7	-.546E+3	-.196E+3	-.642E+3	0.0	0.0	0.0
	.401E+4	-.669E+2	-.732E+3	.263E+03	-.209E+04	0.0
Row 8	-.196E+3	-.707E+2	0.0	0.0	0.0	-.555E+3
	-.669E+2	.7212E+3	.263E+3	-.948E+2	0.0	0.0
Row 9	0.0	0.0	0.0	0.0	-.209E+4	0.0
	-.732E+3	.263E+3	.861E+5	.478E+5	0.0	0.0
Row 10	0.0	0.0	0.0	0.0	0.0	0.0
	.263E+3	-.948E+2	.478E+5	.139E+6	0.0	-.111E+6
Row 11	0.0	0.0	0.0	0.0	-.732E+3	-.263E+3
	-.209E+4	0.0	0.0	0.0	.861E+5	-.478E+5
Row 12	0.0	0.0	0.0	0.0	-.263E+3	-.948E+2
	0.0	0.0 .	0.0	-.111E+6	-.478E+5	0.139E+6

Note that the first eight elements of the mass matrix are essentially the values of the non-structural mass because the non structural mass is large compared to the structure mass. The units mass matrix elements used here are $lb \cdot sec^2/in.$

Stiffness Matrix

Row 1	.129E+1	-.239E-5	0.0	0.0	.692E-5	0.0
	.665E-5	.239E-5	0.0	0.0	0.0	0.0
Row 2	-.239E-5	.129E+1	0.0	.776E-6	0.0	0.0
	.239E-5	.862E-6	0.0	0.0	0.0	0.0
Row 3	0.0	0.0	.129E+1	.239E-5	.665E-5	-.239E-5
	.692E-5	0.0	0.0	0.0	0.0	0.0
Row 4	0.0	.776E-6	.239E-5	.129E+1	-.239E-5	.862E-6
	0.0	0.0	0.0	0.0	0.0	0.0
Row 5	.692E-5	0.0	.665E-5	-.239E-5	.129E+1	-.815E-6
	0.0	0.0	.226E-4	0.0	.891E-5	.321E-5
Row 6	0.0	0.0	-.239E-5	.862E-6	-.815E-6	.129E+1
	0.0	.776E-6	0.0	0.0	.321E-5	.115E-5
Row 7	.665E-5	.239E-5	.692E-5	0.0	0.0	0.0
	.129E+1	.815E-6	.891E-5	-.321E-5	.226E-4	0.0
Row 8	.239E-5	.862E-6	0.0	0.0	0.0	.776E-6
	.815E-6	.129E+1	-.321E-5	.115E-5	0.0	0.0
Row 9	0.0	0.0	0.0	0.0	.226E-4	0.0
	.891E-5	-.321E-5	.881E-3	-.640E-4	0.0	0.0
Row 10	0.0	0.0	0.0	0.0	0.0	0.0
	-.321E-5	.115E-5	-.640E-4	.834E-3	0.0	.155E-3
Row 11	0.0	0.0	0.0	0.0	.891E-5	.321E-5
	.226E-4	0.0	0.0	0.0	.881E-3	.640E-4
Row 12	0.0	0.0	0.0	0.0	.321E-5	.115E-5
	0.0	0.0	0.0	.155E-3	.640E-4	.834E-3

Note that both the mass and stiffness matrices are symmetric due to the way the finite element analysis generated the data. The units of stiffness matrix elements used here are *lb/in.*

Truth Model Matrices

The truth model matrices are provided here. These matrices are shown in Equatic (3.14), (3.15), and (3.18).

$\tilde{\mathbf{F}}_t$ *Matrix*

Row 1	0	0	0	0	0	0	0	0	0	0	0	0
	1.0	0	0	0	0	0	0	0	0	0	0	0
Row 2	0	0	0	0	0	0	0	0	0	0	0	0
	0	1.0	0	0	0	0	0	0	0	0	0	0
Row 3	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	1.0	0	0	0	0	0	0	0	0	0
Row 4	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	1.0	0	0	0	0	0	0	0	0
Row 5	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	1.0	0	0	0	0	0	0	0
Row 6	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	1.0	0	0	0	0	0	0
Row 7	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	1.0	0	0	0	0	0
Row 8	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	1.0	0	0	0	0
Row 9	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	1.0	0	0	0
Row 10	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	1.0	0	0
Row 11	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	1.0	0
Row 12	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	1.0

Row 13	0.000E+0	0	0	0	0	0
	0	0	0	0	0	0
	0.000E+0	0	0	0	0	0
	0	0	0	0	0	0
Row 14	0	-7.917E+1	0	0	0	0
	0	0	0	0	0	0
	0	-8.898E-2	0	0	0	0
	0	0	0	0	0	0
Row 15	0	0	-5.084E+2	0	0	0
	0	0	0	0	0	0
	0	0	-2.254E-1	0	0	0
	0	0	0	0	0	0
Row 16	0	0	0	-8.728E+2	0	0
	0	0	0	0	0	0
	0	0	0	-2.954E-1	0	0
	0	0	0	0	0	0
Row 17	0	0	0	0	-9.704E+2	0
	0	0	0	0	0	0
	0	0	0	0	-3.115E-1	0
	0	0	0	0	0	0
Row 18	0	0	0	0	0	-1.076E+3
	0	0	0	0	0	0
	0	0	0	0	0	-3.280E-1
	0	0	0	0	0	0

Row 19	0	0	0	0	0	0
	-2.958E+3	0	0	0	0	0
	0	0	0	0	0	0
	-5.438E-1	0	0	0	0	0
Row 20	0	0	0	0	0	0
	0	-3.382E+3	0	0	0	0
	0	0	0	0	0	0
	0	-5.815E-1	0	0	0	0
Row 21	0	0	0	0	0	0
	0	0	-9.720E+5	0	0	0
	0	0	0	0	0	0
	0	0	-9.859E+0	0	0	0
Row 22	0	0	0	0	0	0
	0	0	0	-8.134E+7	0	0
	0	0	0	0	0	0
	0	0	0	-9.018E+1	0	0
Row 23	0	0	0	0	0	0
	0	0	0	0	-1.326E+8	0
	0	0	0	0	0	0
	0	0	0	0	-1.151E+2	0
Row 24	0	0	0	0	0	0
	0	0	0	0	0	-3.982E+8
	0	0	0	0	0	0
	0	0	0	0	0	-1.995E+2

Note that frequency of the the rigid body mode, or the $\tilde{\mathbf{F}}_t$ matrix elements corresponding to that mode, is set to zero in the $24 - by - 24$ matrix and therefore is unaffected by the parameter variations [8:89].

$\tilde{B}_t = \tilde{G}_t$ *Matrix*

$$\begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0 \\
-4.762E-1 & -2.747E-1 & 1.0 \\
2.584E-1 & -4.742E-1 & 0 \\
9.694E-2 & 1.578E-1 & 0 \\
-4.848E-1 & 2.519E-1 & 0 \\
-8.862E-2 & 9.464E-2 & 0 \\
2.295E-1 & 4.591E-1 & 0 \\
2.931E-2 & 2.683E-2 & 0 \\
4.325E-2 & 3.949E-3 & 0 \\
-4.322E-5 & 1.182E-1 & 0 \\
4.354E-8 & 4.148E-3 & 0 \\
6.177E-7 & 2.348E-3 & 0 \\
-4.210E-9 & -2.567E-4 & 0
\end{bmatrix}$$

The first column of the $24-by-3$ $\tilde{B}_t$ matrix represent actuator inputs located at node 1 on the truss; the second column represents actuator inputs located at node 2 on the truss; and the third column is due to the actuator located at node 7, i.e, the hub. The nonzero portion of the first two columns follows the development of previous theses [11, 22, 26] and was designed in physical coordinates as follows:

$$\mathbf{b} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T$$

and then transformed into modal coordinates. The nonzero entries correspond to physical positions and velocities on the truss. The angular input actuator was then added by augmenting a third column and entering a one in the row corresponding to the rigid body angular velocity state.

$\tilde{\mathrm{H}}_t$ *Matrix*

Row 1	6.075E-1	-3.287E-1	-1.231E-1	6.276E-1	1.147E-1	-2.993E-1
	-3.194E-2	-5.658E-2	2.029E-7	2.435E-9	2.143E-9	-3.475E-10
	0	0	0	0	0	0
	0	0	0	0	0	0
Row 2	3.440E-1	5.904E-1	-2.036E-1	-3.257E-1	-1.156E-1	-5.929E-1
	-2.914E-2	-5.342E-3	-1.204E-4	-3.734E-6	-2.003E-6	1.695E-7
	0	0	0	0	0	0
	0	0	0	0	0	0
Row 3	0	0	0	0	0	0
	0	0	0	0	0	0
	6.075E-1	-3.287E-1	-1.231E-1	6.276E-1	1.1471-1	-2.993E-1
	-3.194E-2	-5.658E-2	2.029E-7	2.435E-9	2.143E-9	-3.475E-10
Row 4	0	0	0	0	0	0
	0	0	0	0	0	0
	3.4408E-1	5.9043E-1	-2.0361E-1	-3.2572E-1	-1.1565E-1	-5.9294E-1
	-2.9144E-2	-5.3428E-3	-1.2048E-4	-3.7340E-6	-2.0033E-6	1.6959E-7

Row 5	1.000E+0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0
Row 6	0	0	0	0	0	0	0	0	0	0	0	0
	1.000E+0	0	0	0	0	0	0	0	0	0	0	0

The first and second rows of the $6 - by - 24$ $\tilde{\mathbf{H}}_t$ matrix represent position measurements from the sensors located at nodes 1 and 2. The third and fourth rows represent velocity measurements from the sensors located at nodes 1 and 2. The fifth and sixth rows represent angular position and velocity measurements at the hub, node 7. The development of the non-zero portions of rows one and two (as well as the non-zero portions of rows three and four) of the matrix follows the development of previous theses [11, 22, 26] and is calculated in physical coordinates as follows:

$$\mathbf{H_p} = \mathbf{H_v} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

and then transformed into modal coordinates. The dimension of the matrix partition is $2 - by - 12$ due to the fact that the truth model state vector is partitioned into 12 modal positions followed by 12 modal velocities. The partitions are arranged according to the form shown in Equation (3.18). Angular position and velocity measurements were then added by augmenting with rows five and six and entering a one in the columns corresponding to the rigid body angular position and velocity states. Note that the $2 - by - 12$ partitions in rows one and two are identical to the partitions in rows three and four because of the co-location of the position and velocity sensors. Note that the form of the $\tilde{\mathbf{H}}_t$ provided here is different from that provided in the general development of Equation (3.18) due to the way in which the rigid body angular position and velocity measurements are incorporated.

Reduced Order Matrices

The reduced order matrices for the design of the Kalman filter and LQG controller are now provided. The matrices are developed by retaining the first three modes in the $\tilde{\mathbf{F}}_f$, $\tilde{\mathbf{B}}_f$, and $\tilde{\mathbf{H}}_f$ matrices in Equations (3.26) and (3.27). These matrices are associated with the filter model state vector given in Equation (3.29).

The $6-by-6$ $\tilde{\mathbf{F}}_f$ matrix is developed similar to the $\tilde{\mathbf{F}}_t$ matrix, but only the first three modes are retained:

$$\tilde{\mathbf{F}}_f = \begin{bmatrix} 0 & 0 & 0 & 1.0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.0 \\ 0.0 & 0 & 0 & 0.0 & 0 & 0 \\ 0 & -7.918E-1 & 0 & 0 & -8.898E-2 & 0 \\ 0 & 0 & -5.084E+2 & 0 & 0 & -2.254E-1 \end{bmatrix}$$

The $6-by-3$ $\tilde{\mathbf{B}}_f = \tilde{\mathbf{G}}_f$ matrix is developed similar to that for the $\tilde{\mathbf{B}}_t$ matrix:

$$\tilde{\mathbf{B}}_f = \tilde{\mathbf{G}}_f = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ -4.762E-1 & -2.747E-1 & 1.0 \\ 2.584E-1 & -4.742E-1 & 0 \\ 9.694E-2 & 1.578E-1 & 0 \end{bmatrix}$$

The $6-by-6$ $\tilde{\mathbf{H}}_f$ matrix is developed similar to that for the $\tilde{\mathbf{H}}_t$ matrix:

$$\tilde{H}_f = \begin{bmatrix} 6.075E-1 & -3.287E-1 & -1.231E-1 & 0 & 0 & 0 \\ 3.440E-1 & 5.904E-1 & -2.031E-1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 6.075E-1 & -3.287E-1 & -1.231E-1 \\ 0 & 0 & 0 & 3.440E-1 & 5.904E-1 & -2.036E-1 \\ 1.0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.0 & 0 & 0 \end{bmatrix}$$

Previous theses have shown, that for the reduced order model investigated [8:55-56], [11:65-66], [26:59-60], $\mathbf{D}$ of Equation (3.27) is given by:

$$\begin{aligned} \tilde{D} &= -\tilde{H}_2 \tilde{F}_{22}^{-1} \tilde{B}_2 \\ &= -\begin{bmatrix} \tilde{H}_{2p_{2\times 9}} & 0_{2\times 9} \\ 0_{2\times 9} & \tilde{H}_{2v_{2\times 9}} \\ 0_{2\times 9} & 0_{2\times 9} \end{bmatrix}_{6\times 18} \begin{bmatrix} [-\omega_2^2][2\zeta_2\omega_2]_{9\times 9} & [-\omega_2^2]_{9\times 9}^{-1} \\ I_{9\times 9} & 0_{9\times 9} \end{bmatrix}_{18\times 18} \begin{bmatrix} 0_{9\times 3} \\ \tilde{b}_{2_{9\times 3}} \end{bmatrix}_{18\times 3} \\ &= \begin{bmatrix} \tilde{H}_{2p}[-\omega_2^2]^{-1}\tilde{b}_{2_{2\times 3}} \\ 0_{4\times 3} \end{bmatrix}_{6\times 3} \end{aligned} \tag{A.1}$$

where the unmodeled position states for nodes 1 and 2 are represented by $\tilde{H}_{2p}$ and the unmodelled velocity states for nodes 1 and 2 are represented by $\tilde{H}_{2v}$. The $\tilde{b}_2$ matrix contains the unmodelled portion of the control input matrix. These matrix partitions were derived from matrix development provided previously in this appendix as well as Equations (3.19) and (3.20) in Section 3.4.1. The resulting $m-by-r$ $\tilde{D}$ matrix (which is $6-by-3$ in this research) is given as:

$$\tilde{D} = \begin{bmatrix} -4.239E-4 & 6.423E-5 & 0.0 \\ 6.461E-5 & -3.586E-4 & 0.0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Appendix B. *Dynamics Noise Strength and Measurement Noise Covariance Matrices*

The purpose of this appendix is to provide the truth model and filter model dynamics noise strength and measurement noise covariance values used for the simulations. These matrices were initial described in Section 4.4.2. In addition, this appendix describes in more detail how the truth model dynamics noise strength is incorporated into the truth model propagation simulation.

Dynamics Noise Strength Matrices

This section expands on two ideas related to determination and implementation of dynamics niose strength matrices. First, iterations to find appropriate values actually changed $\mathbf{Q}_t$ and $\mathbf{Q}_f$ and then discrete versions were computed. The values used for duplication and those finally determined are provided below.

Duplication Simulations. The value of $\mathbf{Q}_t$ used in the duplication and modification simulations is as follows:

$$\mathbf{Q}_t = \begin{bmatrix} 0.05 & 0.00 & 0.00 \\ 0.00 & 0.05 & 0.00 \\ 0.00 & 0.00 & 0.05 \end{bmatrix}$$

The units of $\mathbf{Q}$ are in^2/sec (rad^2/sec) for position states and in^2/sec^3 (rad^2/sec^3) for velocity states.

The value for $\mathbf{Q}_{add}$, as described in Section 5.2.3, used in the duplication and modification simulations is as follows:

$$\mathbf{Q}_{add} = \begin{bmatrix} 7.95 & 0.00 & 0.00 \\ 0.00 & 7.95 & 0.00 \\ 0.00 & 0.00 & 4.95 \end{bmatrix}$$

The value for $\mathbf{Q}_f$ used in the duplication and modification simulations is as follows:

$$\mathbf{Q}_f = \begin{bmatrix} 8.00 & 0.00 & 0.00 \\ 0.00 & 8.00 & 0.00 \\ 0.00 & 0.00 & 5.00 \end{bmatrix}$$

Dynamics Noise Strength Tuned Values. The tuned value of $\mathbf{Q}_t$ found in this research, described in Section 5.2.3, is as follows:

$$\mathbf{Q}_t = \begin{bmatrix} 0.05 & 0.00 & 0.00 \\ 0.00 & 0.05 & 0.00 \\ 0.00 & 0.00 & 0.05 \end{bmatrix}$$

The tuned value for $\mathbf{Q}_{add}$ found in this research is as follows:

$$\mathbf{Q}_{add} = \begin{bmatrix} 79.95 & 00.00 & 00.00 \\ 00.00 & 79.95 & 00.00 \\ 00.00 & 00.00 & 49.95 \end{bmatrix}$$

The tuned value for $\mathbf{Q}_f$ found in this research is as follows:

$$\mathbf{Q}_f = \begin{bmatrix} 80.00 & 00.00 & 00.00 \\ 00.00 & 80.00 & 00.00 \\ 00.00 & 00.00 & 50.00 \end{bmatrix}$$

The value for $\mathbf{Q}_f$ is found by adding $\mathbf{Q}_{add}$ to the $\mathbf{Q}_t$ matrix in each case. The values for $\mathbf{Q}_f$ are those used by Lashlee in his research [11].

Second, implementing the truth model propagation noise is slightly different from that provided in Equation (2.15). For this research, a first order approximation is made to the $\mathbf{Q}_{d_t}$ computation provided in Equation (2.15) due to the fact that $\underline{\mathbf{w}}_{d_t}(t_i)$ will be transformed by the $\mathbf{G}_{d_t}$ matrix (as shown in Equation (2.10), which is not the identity matrix). In this case, a first order approximation is given as follows:

$$\mathbf{Q}_d \approx \mathbf{G}\mathbf{Q}\mathbf{G}^T\Delta t \tag{B.1}$$

Using a first order approximation of $\mathbf{G}_d \approx \mathbf{G}\Delta t$, the covariance of the quantity $\mathbf{G}_d\underline{\mathbf{w}}_d(t_i)$ becomes:

$$E\{\mathbf{G}_d\underline{\mathbf{w}}_d(t_i)\underline{\mathbf{w}}_d(t_i)^T\mathbf{G}_d^T\} \approx \mathbf{G}\Delta t E\{\underline{\mathbf{w}}_d(t_i)\underline{\mathbf{w}}_d(t_i)^T\}\mathbf{G}^T\Delta t \tag{B.2}$$

In order for Equation (B.2) to equal (B.1) (to first order), the following must be true:

$$E\{\underline{\mathbf{w}}_d(t_i)\underline{\mathbf{w}}_d(t_i)^T\} = \frac{\mathbf{Q}}{\Delta \mathbf{t}} \tag{B.3}$$

Therefore, implementing this in the simulation requires the noise transformation matrix, A, given in Equation (4.9) to be calculated as $\mathbf{A} = \sqrt{\frac{\mathbf{Q_t}}{\Delta \mathbf{t}}}$, where in this research $\Delta t = 0.05 sec$.

Measurement Noise Covariance Matrices

Duplication Simulations. Implementing the measurement noise is straightforward as described in Equation (4.9). The value of $\mathbf{R}_t$ used in the duplication and modification simulations is as follows:

$$\mathbf{R}_t = \begin{bmatrix} 2.7E-06 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2.7E-07 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2.5E-04 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4.2668E-04 & 0 & 0 \\ 0 & 0 & 0 & 0 & 12.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 8.55E-05 \end{bmatrix}$$

Units of $\mathbf{R}$ are $in^2 \cdot sec$ $(rad^2 \cdot sec)$ for position measurements and in^2/sec (rad^2/sec) for velocity measurements [11:94].

The value of $\mathbf{R}_{add}$ used in the duplication and modification simulations is as follows:

$$\mathbf{R}_{add} = \begin{bmatrix} -6.24E-08 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.048E-06 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4.45 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3.4E-01 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 9.77E-15 \end{bmatrix}$$

The value of $\mathbf{R}_f$ used in the duplication and modification simulations is as follows:

$$\mathbf{R}_f = \begin{bmatrix} 2.6376 - 06 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.318E-06 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4.4502 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3.3957E-01 & 0 & 0 \\ 0 & 0 & 0 & 0 & 12.0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 8.55E-05 \end{bmatrix}$$

Note: The 4-4 entry used for $\mathbf{R}_{add}$ resulted in a $\mathbf{R}_f$ matrix which is not positive definite.

Measurement Noise Tuned Values. The tuned value of $\mathbf{R}_t$ used in this research, as described in Section 5.2.3, is as follows:

$$\mathbf{R}_t = \begin{bmatrix} 2.7E-06 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2.7E-07 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2.5E-04 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4.2668E-04 & 0 & 0 \\ 0 & 0 & 0 & 0 & 4.8875E-07 & 0 \\ 0 & 0 & 0 & 0 & 0 & 8.55E-05 \end{bmatrix}$$

Note that the 5-5 entry is the only difference between the $\mathbf{R}_t$ matrix used in this research and the $\mathbf{R}_t$ matrix used in previous research. This difference is discussed in Section 5.2.3. The value of $\mathbf{R}_{add}$ used in this research is as follows:

$$\mathbf{R}_{add} = \begin{bmatrix} 1.1325E-08 & 1.5944E-08 & 1.0296E-06 & 7.1844E-07 & 8.6427E-09 & 5.3644E-09 \\ 1.5944E-08 & 4.3735E-08 & 5.6047E-07 & 4.2312E-07 & 3.6657E-08 & 5.5507E-09 \\ 1.0295E-06 & 5.6043E-07 & 1.9865E-05 & 1.6783E-06 & 1.6763E-06 & 2.1497E-07 \\ 7.1842E-07 & 4.2306E-07 & 1.6783E-06 & 1.8856E-05 & 1.2182E-06 & 1.1496E-07 \\ 8.6427E-09 & 3.6657E-08 & 1.6763E-06 & 1.2183E-06 & 0.0000E+00 & 0.0000E+00 \\ 5.3644E-09 & 5.5507E-09 & 2.1497E-07 & 1.1496E-07 & 0.0000E+00 & 0.0000E+00 \end{bmatrix}$$

The value of $\mathbf{R}_f$ used in this research is as follows:

$$\mathrm{R}_f = \begin{bmatrix} 2.7113E-06 & 1.5944E-08 & 1.0296E-06 & 7.1844E-07 & 8.6427E-09 & 5.3644E-09 \\ 1.5944E-08 & 3.1373E-07 & 5.6047E-07 & 4.2312E-07 & 3.6657E-08 & 5.5507E-09 \\ 1.0295E-06 & 5.6043E-07 & 2.6986E-04 & 1.6783E-06 & 1.6763E-06 & 2.1497E-07 \\ 7.1842E-07 & 4.2306E-07 & 1.6783E-06 & 4.4553E-04 & 1.2182E-06 & 1.1496E-07 \\ 8.6427E-09 & 3.6657E-08 & 1.6763E-06 & 1.2183E-06 & 4.8875E-07 & 0.0000E+00 \\ 5.3644E-09 & 5.5507E-09 & 2.1497E-07 & 1.1496E-07 & 0.0000E+00 & 8.5500E-05 \end{bmatrix}$$

The value for $\mathbf{R}_f$ is found by adding $\mathbf{R}_{add}$ to the $\mathbf{R}_t$ matrix in each case. The $\mathbf{R}_f$ matrix is a symmetric, positive definite matrix.

Appendix C. *LQG State and Control Weighting Matrices*

The purpose of this appendix is to provide the state weighting, X , and control weighting, U, matrices used in the duplication of past research and those matrices found in the controller tuning study described in Section 5.3.3. These matrices were initially described in Section 1.1.3.

State Weighting Matrices

Duplication Simulations. The values of **X** used in the duplication and modification simulations are as follows:

$$\mathbf{X} = \begin{bmatrix} 7.62E+5 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 7.91E+3 & 0.00 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 5.08E+2 & 0.00 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 8.20E+01 & 0.00 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 1.00E+0 & 0.00 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 1.00E+0 \end{bmatrix}$$

This matrix corresponds to the parameter point (7,6) and represents the nominal case for tuning (i.e., a scale factor of 1.0).

State Weighting Matrix Tuned Values. The appropriate values for the state weighting matrices in this research were obtained by selecting a few parameter points in the high natural frequency region of the parameter space and determining an appropriate scaling factor for the state weighting matrix provided above. The remaining state weighting matrix scaling factors were interpolated from those determined by simulations. The parameter locations used for state weighting matrix tuning and the associated scaling factors are provided as follows: parameter (1,6): scale 34.5; parameter (1,10): scale 70.0; parameter (5,6): scale 8.5; parameter (5,9): scale 26.0; and parameter (7,9): scale 1.4.

Control Weighting Matrices

Duplication Simulations. The values of U used in the duplication and modification simulations are as follows:

$$\mathbf{U} = \begin{bmatrix} 3.0 & 0.0 & 0.0 \\ 0.0 & 3.0 & 0.0 \\ 0.0 & 0.0 & 9.0 \end{bmatrix}$$

This matrix corresponds to the parameter point (7,6) and represents the nominal case for tuning (i.e., a scale factor of 1.0).

Control Weighting Matrix Tuned Values. The methodology for determining the appropriate control weighting matrices is the same as for the state weighting matrices. The appropriate values were obtained by selecting a few parameter points in the high natural frequency region of the parameter space and determining an appropriate scaling factor for the control weighting matrix provided above. The remaining control weighting matrix scaling factors were interpolated from those determined by simulations. The parameter points used for control weighting matrix tuning are the same as those for the state weighting matrix determination. The parameter locations and the associated scaling factors are as follows: parameter (1,6): scale 5.0; parameter (1,10): scale 10.0; parameter (5,6): scale 1.2; parameter (5,9): scale 3.7; and parameter (7,9): scale 2.0.

Bibliography

1. Baram, Y., and Sandell, N. R., Jr. "An Information Theoretic Approach to Dynamic System Modeling and Identification," *IEEE Transactions on Automatic Control, AC-23 (1)*: 61-66 (1978).
2. Brogan, William L. *Modern Control Theory.* Englewood Cliffs, New Jersey: Prentice-Hall, Inc., 1985.
3. Filios, Capt Paul G. *Moving-Bank Multiple Model Adaptive Algorithms Applied to Spacecraft Control.* MS Thesis AFIT/GE/ENG/85D-14. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, Ohio, December 1985 (AD-A164 016).
4. FORTRAN Subroutines for Mathematical Applications Software. Houston: IMSL Inc., Houston, Texas (1989).
5. GNUPLOT USER'S MANUAL. "An Interactive Plotting Program." Williams, Thomas, and Colin Kelley.
6. Hawkes, Robert M., and John B. Moore. "Performance Bounds for Adaptive Estimation," *Proceedings of IEEE, 64*: 1143-1150 (August 1976).
7. Hentz, 1Lt Karl P. *Feasibility Analysis of Moving Bank Multiple Model Adaptive Estimation and Control Algorithms.* MS Thesis AFIT/GE/ENG/84D-32. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, Ohio, December 1984 (AD-A152 015).
8. Karnick, 2Lt Drew A. *Moving Bank Multiple Model Adaptive Estimation Applied to Flexible Spacestructure Control.* MS Thesis AFIT/GE/ENG/86D-41. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, Ohio, December 1986 (AD-A178 870).
9. Karnick, Drew A., and Peter S. Maybeck. "Moving Bank Multiple Model Adaptive Estimation Applied to Flexible Spacestructure Control." *Proceedings of the 26th IEEE Conference on Decision and Control*: 1249-1257. Los Angeles, California (December 1987).
10. Kokotovic, P.V., O'Malley, R.E. Jr., and Sannuti, P. "Singular Pertubations and Order Reduction in Control Theory – An Overview," *Automatica, 12*: 123-132 (1976).
11. Lashlee, Capt Robert W. *Moving Bank Multiple Model Adaptive Estimation Applied to Flexible Spacestructure Control.* MS Thesis AFIT/GE/ENG/87D-36. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, Ohio, December 1987.
12. Lashlee, Robert W., and Peter S. Maybeck. "Spacestructure Control Using Moving Bank Multiple Model Adaptive Estimation," *Proceedings of the 27th IEEE Conference on Decision and Control*: 712-717. Austin, Texas (December 1988).
13. LQGLIB USER'S MANUAL. "A Description of Computer Routines for Use in Linear Systems Studies." Air Force Institute of Technology (May 1984).

14. Lynch, P.J., and Siva S. Banda. "Active Control for Vibration Damping," *Damping: 1986 Proceedings*, Technical Report. Flight Dynamics Laboratory, Air Force Wright Aeronautical Laboratory, Wright-Patterson AFB, Ohio, 1986 (AFWAL-TR-86-3509).

15. Maybeck, Peter S. *Stochastic Models, Estimation, and Control, Volume 1.* New York: Academic Press, 1979.

16. -----. *Stochastic Models, Estimation, and Control, Volume 2.* New York: Academic Press, 1982.

17. -----. *Stochastic Models, Estimation, and Control, Volume 3.* New York: Academic Press, 1982.

18. -----. "Moving-Bank Multiple Model Adaptive Estimation and Control Algorithms: An Evaluation," *Control and Dynamic Systems: Advances in Aerospace System Dynamics and Control Systems, Edited by C. T. Leondes, 31*: 1-31. Academic Press, San Diego CA. (1989).

19. Maybeck, Peter S., and Capt Karl P. Hentz. "Investigation of Moving- Bank Multiple Model Adaptive Algorithms," *Proceedings of the 24th Conference on Decision and Control*: 1874-1881. Ft. Lauderdale, FLorida (December 1985).

20. Maybeck, Peter S., and Michael Roger Schore. "Robustness of A Moving-Bank Multiple Model Adaptive Algorithm For Control of a Flexible Spacestructure". *Proceedings of the IEEE National Aerospace and Electronics Conference*: 368-374. Dayton, Ohio (May 1990).

21. Maybeck, Peter S., and Richard D. Stevens. "Reconfigurable Flight Control Via Multiple Model Adaptive Control Methods," *Proceedings of IEEE Conference on Decision and Control.* Honolulu, Hawaii (December 1990).

22. Schore, Capt Michael Roger. *Robustness of a Moving-Bank Multiple Model Adaptive Controller for a Large Space Structure.* MS Thesis AFIT/GE/ENG/89D-46. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, Ohio, December 1989.

23. Sheldon, Stuart N. *An Optimizing Design Strategy for Multiple Model Adaptive Estimation and Control.* PhD dissertation. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, Ohio, December 1989.

24. Sheldon, Stuart N., and Peter S. Maybeck. "An Optimizing Design Strategy for Multiple Model Adaptive Estimation and Control." *Proceedings of IEEE Conference on Decision and Control.* Honolulu, Hawaii (December 1990).

25. Stevens, Richard D. *Characterization of a Reconfigurable Multiple Model Controller Using a STOL F-15 Model.* MS Thesis AFIT/GE/ENG/89D-52. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, Ohio, December 1989 (AD-B139 226).

26. Van Der Werken, Capt Daniel F., Jr. *A Robustness Analysis of Moving-Bank Multiple Model Adaptive Estimation and Control of a Large Flexible Space Structure.* MS Thesis AFIT/GE/ENG/88D-59. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB, Ohio, December 1988.

27. Venkayya, Vipperla B., and Victoria A. Tischler. "Frequency Control and Its Effect on the Dynamic Response of Flexible Structures," *AIAA Journal, 23*: 1768-1774 (November 1985).

28. Venkayya, Vipperla B. *'Analyze' - Analysis of Aerospace Structures with Membrane Elements.* Technical Report. Analysis and Optimization Branch, Structural Mechanics Division, Air Force Flight Dynamics Laboratory, Air Force Wright Aeronautical Laboratories, Wright-Patterson AFB, Ohio, December 1978 (AFFDL-TR-78-170).

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	December 1990	Master's Thesis

4. TITLE AND SUBTITLE

Moving-Bank Multiple Model Adaptive Estimation and Control Applied to a Flexible Space Structure

5. FUNDING NUMBERS

6. AUTHOR(S)

Robert B. Moyle
Captain, USAF

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

Air Force Institute of Technology
WPAFB OH 45433-6583

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GE/ENG/90D-45

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

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for Public Release; Distribution Unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT (Maximum 200 words)

The performance of moving-bank multiple model adaptive estimation (MMAE) and control (MMAC) algorithms is analyzed in this thesis. The performance of a six-state filter/controller model is evaluated on the basis of estimation/control performance against a 24-state truth model. A model developed using finite element analysis is used to approximate a large flexible space structure. The space structure is configured as a two-bay truss which is attached to a large central hub. Results indicate that appropriate determination of the filter model noise statistics as well as the LQG controller weighting matrices significantly improve performance of the bank throughout the parameter space. The performance of the moving-bank algorithms is seriously degraded by the inclusion of the filter-computed residual covariance in the conditional probability density function for computation of the hypothesis conditional probabilities within the multiple model algorithms. The performance of the moving-bank MMAE/MMAC algorithms using parameter position estimate monitoring for parameter identification, and using a modified MMAC methodology for implementing control, provide performance comparable to an artificially informed controller benchmark for the cases of slowly varying and jump parameters, as well as for constant parameter values.

14. SUBJECT TERMS

Multiple Model Adaptive Estimation, Multiple Model Adaptive Control, LQG Control, Flexible Space Structure

15. NUMBER OF PAGES

234

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