# Discrete Kalman Filtering Equations of Second-Order Form for Control-Structure Interaction Simulations+

K. C. Park[1] and K. F. Alvin[2]
Center for Space Structures and Controls
University of Colorado, Campus Box 429
Boulder, Colorado 80309

and

W. Keith Belvin[3]
Spacecraft Dynamics Branch
NASA Langley Research Center
Hampton, Virginia 23665



## Abstract

A second-order form of discrete Kalman filtering equations is proposed as a candidate state estimator for efficient simulations of control-structure interactions in coupled physical coordinate configurations as opposed to decoupled modal coordinates. The resulting matrix equation of the present state estimator consists of the same symmetric, sparse $N \times N$ coupled matrices of the governing structural dynamics equations as opposed to unsymmetric $2N \times 2N$ state space-based estimators. Thus, in addition to substantial computational efficiency improvement, the present estimator can be applied to control-structure design optimization for which the physical coordinates associated with the mass, damping and stiffness matrices of the structure are needed instead of modal coordinates.

---

+ An earlier version of the present paper without numerical experiments was presented at the AIAA Guidance and Control Conference, Portland, Ore., 20-22 August 1990, Paper No. AIAA 90-3387.

[1] Professor of Aerospace Engineering, University of Colorado. Associate Fellow of AIAA.

[2] Graduate Research Assistant

[3] Structural Dynamics Division, NASA Langley Research Center. Senior Member AIAA.

## Introduction

Current practice in the design, modeling and analysis of flexible large space structures is by and large based on the finite element method and the associated software. The resulting discrete equations of motion for structures, both in terms of physical coordinates and of modal coordinates, are expressed in a second-order form. As a result, the structural engineering community has been investing a considerable amount of research and development resources to develop computer-oriented discrete modeling tools, analysis methods and interface capabilities with design synthesis procedures; all of these exploiting the characteristics of second-order models. Recent work in the area of structural dynamics simulation and massively-parallel processing also rely on the second-order equation forms.

On the other hand, modern linear control theory has its roots firmly in a first-order form of the governing differential equations, e.g., Kwakernaak and Sivan[1]. Thus, several investigators have addressed the issues of interfacing second-order structural systems and control theory based on the first-order form[2-7]. As a result of these studies, it has become straightforward for one to synthesize direct state feedback based control laws within the framework of a first-order control theory and then to recast the resulting control laws in terms of the second-order structural systems.

Unfortunately, controllers based on a first-order state estimator are difficult to express in a pure second-order form because the first-order estimator implicitly incorporates an additional filter equation[7]. However a recent work by Juang and Maghami[8] has enabled the first-order filter gain matrices to be synthesized using only second-order equations. To complement the second-order gain synthesis, the objective of the present paper is to develop a second-order based simulation procedure for first-order estimators. The particular class of first-order dynamic compensation chosen for study are the Kalman Filter based state estimators as applied to second-order structural systems. The proposed procedure permits simulation of first-order estimators with nearly the same solution procedure used for treating the structural dynamics equation. Hence, the reduced size of system matrices and the computational techniques that are tailored to sparse second-order structural systems may be employed. As will be shown, the proposed procedure hinges on discrete time integration formulas to effectively reduce the continuous time Kalman Filter to a set of second-order difference equations.

The *primary goal* of the proposed procedure is the incorporation of this general form of state estimation as a *simulation tool* in partitioned control-structure interaction (CSI) analyses. It is expected that Kalman Filters for real-time control of linear time-invariant systems would be implemented in the most efficient form available, typically a real mode-decoupled state space realization. For analytical studies of CSI systems, however, where the objective is frequently simultaneous optimization of controls and structures as in Belvin[9], the use of such a modal form must be weighed against the preprocessing tasks required to generate the model. In these cases, much more flexible controllers expressed in terms of the physical coordinates instead of the modal coordinates are sought, ones which can readily adapt to iterative changes in the structural parameters. One such control law synthesis has been proposed and demonstrated to be effective for CSI optimization[7]. These studies did not account, however, for dynamic compensation when full state feedback control was utilized. With the discrete Kalman Filter proposed herein, a general form of dynamic compensation can be integrated into CSI simulation and optimization which does not impose limits on the designs of the feedback gains or the filter gains.

The paper first reviews of the conventional first-order representation of the continuous second-order structural equations of motion, in which the state variables are defined as the displacements variables $x$ of the second-order structural model and the velocities $\dot{x}$. An examination of the corresponding first-order Kalman filtering equations indicates that, due to the difference in the derivative of the estimated displacement ($\frac{d}{dt}\hat{x}$) and the

estimated velocity ($\hat{\dot{x}}$), transformation of the first-order estimator into an equivalent second-order estimator requires the time derivative of measurement data, a process not recommended for practical implementation.

Next, a transformation via a generalized momentum is introduced to recast the structural equations of motion in a general first-order setting. It is shown that discrete time numerical integration followed by reduction of the resulting difference equations circumvents the need for the time derivative of measurements to solve Kalman filtering equations in a second-order framework. Hence, the Kalman filter equations can be solved using a second-order solution software package.

Subsequently, computer implementation aspects of the present second-order estimator are presented. Several computational paths are discussed in the context of discrete and continuous time simulation. For continuous time control simulation, an equation augmentation is introduced to exploit the symmetry and sparsity of the attendant matrices by maintaining state dependent control and observer terms on the right-hand-side (RHS) of the filter equations. In addition, the computational efficiency of the present second-order filter as compared to the first-order form is presented.

## Continuous Formulation of State Estimators for Structural Systems

Linear, second-order discrete structural models can be expressed as

$$M\ddot{x} + D\dot{x} + Kx = Bu + Gw\,, \qquad x(0) = x_0\,, \quad \dot{x}(0) = \dot{x}_0 \tag{1}$$

$$u = -Z_1 x - Z_2 \dot{x}$$

with the associated measurements

$$z = H_1 x + H_2 \dot{x} + \nu \tag{2}$$

where $M, D, K$ are the mass damping and stiffness matrices of size $(N \times N)$; $x$ is the structural displacement vector, $(N \times 1)$; $u$ is the active control force $(m \times 1)$; $B$ is a constant force distribution matrix $(N \times m)$; $z$ is a set of measurements $(r \times 1)$; $H_1$ and $H_2$ are the measurement distribution matrices $(r \times N)$; $Z_1$ and $Z_2$ are the control feedback gain matrices $(m \times N)$; $w$ and $\nu$ are zero-mean, white Gaussian processes with their respective covariances $Q$ and $R$; and the superscript dot designates time differentiation. In the present study, we will restrict ourselves to the case wherein $Q$ and $R$ are uncorrelated with each other and the initial conditions $x_0$ and $\dot{x}_0$ are also themselves jointly Gaussian with known means and covariances.

The conventional representation of (1) in a first-order form is facilitated by

$$\begin{cases} x_1 = x \\ x_2 = \dot{x} = \dot{x}_1 \\ M\dot{x}_2 = M\ddot{x} = Bu + Gw - Dx_2 - Kx_1 \end{cases} \tag{3}$$

which, when cast in a first-order form, can be expressed as

$$\begin{cases} E\dot{q} = Fq + \bar{B}u + \bar{G}w, \quad q = \begin{pmatrix} x_1 & x_2 \end{pmatrix}^T \\ z = Hq + \nu \end{cases} \tag{4}$$

where

$$E = \begin{bmatrix} I & 0 \\ 0 & M \end{bmatrix}\,, \quad F = \begin{bmatrix} 0 & I \\ -K & -D \end{bmatrix}\,,$$

$$\bar{B} = \begin{Bmatrix} 0 \\ B \end{Bmatrix}, \quad \bar{G} = \begin{Bmatrix} 0 \\ G \end{Bmatrix} \tag{5}$$

It is well-known that the Kalman filtering equations[10,11] for (4) can be shown to be (see, e.g., Arnold and Laub[3]):

$$E\dot{\hat{q}} = F\hat{q} + \bar{B}u + EPH^T R^{-1}\bar{z} \tag{6}$$

where

$$\bar{z} = z - H\hat{q}, \quad P = \begin{bmatrix} U & S^T \\ S & L \end{bmatrix}, \quad \hat{q} = \begin{Bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{Bmatrix} = \begin{Bmatrix} \hat{x} \\ \dot{\hat{x}} \end{Bmatrix} \tag{7}$$

in which $U$ and $L$ are positive definite matrices, $\hat{q}$ is the state estimation vector, and the matrix $P$ is determined by the Riccati equation[1,3]

$$E\dot{P}E^T = FPE^T + EPF^T - EPH^T R^{-1} HPE^T + \bar{G}Q\bar{G}^T \tag{8}$$

The inherent difficulty of reducing the first-order Kalman filtering equations given by (6) to second order form can be appreciated if one attempts to write (6) in a form introduced in (3):

$$\begin{cases} a) & \hat{x}_1 = \hat{x} \\ b) & \hat{x}_2 = \dot{\hat{x}} = \dot{\hat{x}}_1 - L_1\bar{z} \\ c) & M\dot{\hat{x}}_2 = -D\hat{x}_2 - K\hat{x}_1 + B\hat{u} + ML_2\bar{z} \end{cases} \tag{9}$$

where

$$L_1 = (H_1U + H_2S)^T R^{-1}, \qquad L_2 = (H_1S^T + H_2L)^T R^{-1}$$

Note from (9b) that $\hat{x}_2 \neq \dot{\hat{x}}_1$. In other words, the time derivative of the estimated displacement ($\dot{\hat{x}}$) is not the same as the estimated velocity ($\hat{\dot{x}}$); hence, $\hat{x}_1$ and $\hat{x}_2$ must be treated as two independent variables, an important observation somehow overlooked in Hashemipour and Laub[12].

Of course, although not practical, one can eliminate $\hat{x}_2$ from (9). Assuming $\dot{\hat{x}}_1$ and $\hat{x}_2$ are differentiable, differentiate (9b) and multiply both sides by $M$ to obtain

$$M\ddot{\hat{x}}_1 = M\dot{\hat{x}}_2 + ML_1\dot{\bar{z}} \tag{10}$$

Substituting $M\dot{\hat{x}}_2$ from (9c) and $\hat{x}_2$ from (9b) in (10) yields

$$M\ddot{\hat{x}}_1 = -D(\dot{\hat{x}}_1 - L_1\bar{z}) - K\hat{x}_1 + Bu + ML_2\bar{z} + ML_1\dot{\bar{z}} \tag{11}$$

which, upon rearrangements, becomes

$$M\ddot{\hat{x}}_1 + D\dot{\hat{x}}_1 + K\hat{x}_1 = Bu + ML_2\bar{z} + ML_1\dot{\bar{z}} + DL_1\bar{z} \tag{12}$$

There are two difficulties with the above second-order estimator. First, the numerical solution of (12) involves the computation of $\ddot{\hat{x}}_1$ when rate measurements are made. The accuracy of this computation is in general very susceptible to errors caused in numerical differentiation of $\dot{\hat{x}}_1$. Second, and most important, the numerical evaluation of $\dot{\bar{z}}$ that is required in (12) assumes that the derivative of measurement information is available which should be avoided in practice. We now present a computational procedure that circumvents the need for computing measurement derivatives and that enables one to construct estimators based on the second-order model form.

## Second-Order Transformation of Continuous Kalman Filtering Equations

This section presents a transformation of the continuous time first-order Kalman filter to a discrete time set of second-order difference equations for digital implementation. The procedure avoids the need for measurement derivative information. In addition, the sparsity and symmetry of the original mass, damping and stiffness matrices can be maintained. Prior to describing the numerical integration procedure, a transformation based on generalized momenta is presented which is later used to improve computational efficiency of the equation solution.

### Generalized Momenta

Instead of the conventional transformation (3) of the second-order structural system (1) into a first-order form, let us consider the following generalized momenta (see, e.g., Jensen[13] and Felippa and Park[14]):

$$\begin{cases} a) & x_1 = x \\ b) & x_2 = AM\dot{x}_1 + Cx_1 \end{cases} \tag{13}$$

where $A$ and $C$ are constant matrices to be chosen. Note that $AM$ should be nonsingular in order to obtain an equivalent form of (1). Time differentiation of (13b) yields

$$\dot{x}_2 = AM\ddot{x}_1 + C\dot{x}_1 \tag{14}$$

Substituting (1) via (13a) into (14), one obtains

$$\dot{x}_2 = A(Bu + Gw) - (AD - C)\dot{x}_1 - AKx_1 \tag{15}$$

Finally, pairing of (13b) and (15) gives the following first-order form:

$$\begin{bmatrix} AM & 0 \\ AD - C & I \end{bmatrix} \begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{Bmatrix} + \begin{bmatrix} C & -I \\ AK & 0 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} = \begin{bmatrix} 0 \\ A(Bu + Gw) \end{bmatrix} \tag{16}$$

The associated Kalman filtering equation can be shown to be of the following form:

$$\begin{bmatrix} AM & 0 \\ AD - C & I \end{bmatrix} \begin{Bmatrix} \dot{\hat{x}}_1 \\ \dot{\hat{x}}_2 \end{Bmatrix} + \begin{bmatrix} C & -I \\ AK & 0 \end{bmatrix} \begin{Bmatrix} \hat{x}_1 \\ \hat{x}_2 \end{Bmatrix} = \begin{Bmatrix} 0 \\ ABu \end{Bmatrix} + \begin{bmatrix} AM & 0 \\ AD - C & I \end{bmatrix} \begin{bmatrix} \bar{L}_1 \\ \bar{L}_2 \end{bmatrix} \bar{z} \tag{17}$$

where

$$\bar{L}_1 = (\bar{H}_1 U + \bar{H}_2 S)^T R^{-1}, \qquad \bar{L}_2 = (\bar{H}_1 S^T + \bar{H}_2 L)^T R^{-1}$$

and $\bar{H}_1$ and $\bar{H}_2$ correspond to a modified form of measurements expressed as

$$z = H_1 x + H_2 \dot{x} = \bar{H}_1 x_1 + \bar{H}_2 x_2 \tag{18}$$

where

$$\bar{H}_1 = H_1 - H_2 M^{-1} A^{-1} C, \qquad \bar{H}_2 = H_2 M^{-1} A^{-1}$$

Clearly, as in the conventional first-order form (9), $\hat{x}_1$ and $\hat{x}_2$ in (17) are now two independent variables. Specifically, the case of $A = M^{-1}$ and $C = 0$ corresponds to (3) with $x_2 = \dot{x}_1$. However, as we shall see below, the Kalman filtering equations based on the generalized momenta (13) offer several computational advantages over (3).

### Numerical Integration

At this juncture it is noted that in the previous section one first performs the elimination of $\hat{x}_1$ in order to obtain a second-order equation, then performs the numerical solution of the resulting equation. This approach has the disadvantage of having to deal with the time derivative of measurement data. To avoid this, we will first integrate numerically the associated Kalman filtering equation (17).

The direct time integration formula we propose to employ is a mid-point version of the trapezoidal rule:

$$\begin{cases} a) & \left\{\begin{matrix}\hat{x}_1\\ \hat{x}_2\end{matrix}\right\}^{n+1/2} = \left\{\begin{matrix}\hat{x}_1\\ \hat{x}_2\end{matrix}\right\}^{n} + \delta \left\{\begin{matrix}\dot{\hat{x}}_1\\ \dot{\hat{x}}_2\end{matrix}\right\}^{n+1/2} \\ b) & \left\{\begin{matrix}\hat{x}_1\\ \hat{x}_2\end{matrix}\right\}^{n+1} = 2\left\{\begin{matrix}\hat{x}_1\\ \hat{x}_2\end{matrix}\right\}^{n+1/2} - \left\{\begin{matrix}\hat{x}_1\\ \hat{x}_2\end{matrix}\right\}^{n} \end{cases} \tag{19}$$

where the superscript $n$ denotes the discrete time interval $t^n = nh$, $h$ is the time increment and $\delta = h/2$.

It should be noted that we have chosen the trapezoidal rule due to its unconditional stability and high accuracy while it does not introduce any numerical damping (see, for example, Dahlquist[15] and Park[16]). Contamination of damping from numerical dissipation can not only adversely affect the solution accuracy but lead to misinterpretation of the simulation results.

Time discretization of (17) by (19a) at the $n + 1/2$ time step yields

$$\begin{bmatrix} AM & 0 \\ AD - C & I \end{bmatrix} \left\{\begin{matrix}\hat{x}_1^{n+1/2} - \hat{x}_1^n \\ \hat{x}_2^{n+1/2} - \hat{x}_2^n\end{matrix}\right\} + \delta \begin{bmatrix} C & -I \\ AK & 0 \end{bmatrix} \left\{\begin{matrix}\hat{x}_1^{n+1/2} \\ \hat{x}_2^{n+1/2}\end{matrix}\right\}$$
$$= \delta \begin{bmatrix} AM & 0 \\ AD - C & I \end{bmatrix} \begin{bmatrix} \bar{L}_1 \\ \bar{L}_2 \end{bmatrix} \bar{z}^{n+1/2} + \delta \left\{\begin{matrix} 0 \\ ABu^{n+1/2}\end{matrix}\right\} \tag{20}$$

The above difference equations require the solution of matrix equations of $2N$ variables, namely, in terms of the two variables $\hat{x}_2^{n+1/2}$ and $\hat{x}_1^{n+1/2}$, each with a size of $N$. To reduce the above coupled equations of order $2N$ into the corresponding ones of order $N$, we proceed in the following way by exploiting the nature of parametric matrices of $A$ and $C$ as introduced in (13). To this end, we write out (20) as two coupled difference equations as follows:

$$AM(\hat{x}_1^{n+1/2} - \hat{x}_1^n) + \delta(C\hat{x}_1^{n+1/2} - \hat{x}_2^{n+1/2})$$
$$= \delta AM\bar{L}_1\bar{z}^{n+1/2} \tag{21}$$

$$(AD - C)(\hat{x}_1^{n+1/2} - \hat{x}_1^n) + (\hat{x}_2^{n+1/2} - \hat{x}_2^n) + \delta AK\hat{x}_1^{n+1/2}$$
$$= \delta(AD - C)\bar{L}_1\bar{z}^{n+1/2} + \delta\bar{L}_2\bar{z}^{n+1/2} + \delta ABu^{n+1/2} \tag{22}$$

Multiplying (22) by $\delta$ and adding the resulting equation to (21) yields

$$A(M + \delta D + \delta^2 K)\hat{x}_1^{n+1/2} = (AM + \delta(AD - C))\hat{x}_1^n + \delta\hat{x}_2^n$$
$$+\{\delta AM\bar{L}_1 + \delta^2(AD - C)\bar{L}_1 + \delta^2\bar{L}_2\}\bar{z}^{n+1/2} + \delta^2 ABu^{n+1/2} \tag{23}$$

Of several possible choices for matrices $A$ and $B$, we will examine the following two specific cases:

$$\begin{cases} a) & A = I\,, \quad C = D \\ b) & A = M^{-1}\,, \quad C = 0 \end{cases} \tag{24}$$

where the mass matrix $M$ is nonsingular due to its physically positive definite nature since the kinetic energy of structural system is positive for any admissable motion. It is noted that the above two choices, although mathematically equivalent, lead to different computational implementations as discussed below.

The choice of (24a) reduces (23) to:

$$(M + \delta D + \delta^2 K)\hat{x}_1^{n+1/2} = M\hat{x}_1^n + \delta\hat{x}_2^n + \delta^2 B u^{n+1/2}$$
$$+\delta\{M\bar{L}_1 + \delta\bar{L}_2\}\bar{z}^{n+1/2} \tag{25}$$

so that once $\hat{x}_1^{n+1/2}$ is computed, $\hat{x}_2^{n+1/2}$ is obtained from (22) rewritten as

$$\hat{x}_2^{n+1/2} = \hat{x}_2^n + \delta\hat{g}^n - \delta K\hat{x}_1^{n+1/2} \tag{26}$$

where

$$\hat{g}^n = Bu^{n+1/2} + \bar{L}_2\bar{z}^{n+1/2} \tag{27}$$

which is already computed in order to construct the right-hand side of (25). Hence, $K\hat{x}_1^{n+1/2}$ is the only additional computation needed to obtain $\hat{x}_2^{n+1/2}$. It is noted that neither any numerical differentiation nor matrix inversion is required in computing $\hat{x}_2^{n+1/2}$. This has been achieved through the introduction of the general transformation (13) and the particular choice of the parameter matrices given by (24a).

On the other hand, if one chooses the conventional representation (24b), the solution of $\hat{x}_1^{n+1/2}$ is obtained from (23)

$$(M + \delta D + \delta^2 K)\hat{x}_1^{n+1/2} = (M + \delta D)\hat{x}_1^n + \delta M\hat{x}_2^n$$
$$+\delta\{(M + \delta D)\bar{L}_1 + \delta M\bar{L}_2\}\bar{z}^{n+1/2} + \delta^2 B u^{n+1/2} \tag{28}$$

Once $\hat{x}_1^{n+1/2}$ is obtained, $\hat{x}_2^{n+1/2}$ can be computed either by

$$\hat{x}_2^{n+1/2} = (\hat{x}_1^{n+1/2} - \hat{x}_1^n)/\delta - \bar{L}_1\bar{z}^{n+1/2} \tag{29}$$

which is not accurate due to the numerical differentiation to obtain $\dot{\hat{x}}_1^{n+1/2}$, or by (22)

$$\hat{x}_2^{n+1/2} = \hat{x}_2^n + \delta\hat{g}^n - \delta M^{-1}K\hat{x}_1^{n+1/2} -$$
$$M^{-1}D(\hat{x}_1^{n+1/2} - \hat{x}_1^n) + \delta M^{-1}D\bar{L}_1\bar{z}^{n+1/2} \tag{30}$$

which involves two additional matrix-vector multiplications, when $D \neq 0$, as compared with the choice of $A = I$ and $C = D$. Thus (24a) is the preferred representation in a first-order form of the second-order structural dynamics equations (1) and is used in the remainder of this work.

## Decoupling Of Difference Equations

We have seen in the previous section, instead of solving the first-order Kalman filtering equations of $2n$ variables for the structural dynamics systems (1), the solution of the implicit time-discrete estimator equation (25) of $n$ variables can potentially offer a substantial computational saving by exploiting the reduced size

and sparsity of $M, D$ and $K$. This assumes that $\bar{z}^{n+1/2}$ and $u^{n+1/2}$ are available, which is not the case since at the $n^{th}$ time step

$$u^{n+1/2} = -\bar{Z}_1 \hat{x}_1^{n+1/2} - \bar{Z}_2 \hat{x}_2^{n+1/2} \tag{31}$$

$$\bar{z}^{n+1/2} = z^{n+1/2} - \bar{H}_1 \hat{x}_1^{n+1/2} - \bar{H}_2 \hat{x}_2^{n+1/2} \tag{32}$$

requires both $\hat{x}_1^{n+1/2}$ and $\hat{x}_2^{n+1/2}$ even if $z^{n+1/2}$ is assumed to be known from measurements or by solution of (1). Note in (32), the control gain matrices are transformed by

$$\bar{Z}_1 = Z_1 - Z_2 M^{-1} A^{-1} C, \qquad \bar{Z}_2 = Z_2 M^{-1} A^{-1}$$

There are two distinct approaches to decouple (25) and (26) as described in the following sections.

### Discrete Time Update

For systems utilizing discrete-time (i.e. sample and hold) control, equations (31) and (32) become

$$u^{n+1/2} \simeq -\bar{Z}_1 \hat{x}_1^n - \bar{Z}_2 \hat{x}_2^n \tag{33}$$

$$\bar{z}^{n+1/2} \simeq z^n - \bar{H}_1 \hat{x}_1^n - \bar{H}_2 \hat{x}_2^n \tag{34}$$

The time integration step size of the estimator must then be equal to the sample rate of the control, while the continuous structural equations may also be integrated at the same rate or at some fraction of the sampling rate for simulation accuracy considerations. For the present purposes, we have assumed that the sampling interval is the same as the integration time stepsize.

Discrete time simulation is quite simple to implement as the control force and state corrections are treated with no approximation on the right-hand-side (RHS) of (25) and (26). Should continuous time simulation be required, a different approach is necessary.

### Continuous Time Update

To simulate the system given in (25) and (26) in continuous time, strictly speaking, one must rearrange (25) and (26) so that the terms involving $\hat{x}_1^{n+1/2}$ and $\hat{x}_2^{n+1/2}$ are augmented to the left-hand-side (LHS) of the equations. However, this augmentation into the solution matrix $(M + \delta D + \delta^2 K)$ would destroy the computational advantages of the matrix sparsity and symmetry. Thus, a partitioned solution procedure has been developed for continuous time simulation as described in Park and Belvin[17]. The procedure, briefly outlined herein, maintains the control force and state correction on the RHS of the equations as follows.

First, $\hat{x}_1^{n+1/2}$ and $\hat{x}_2^{n+1/2}$ are predicted by

$$\hat{x}_{1p}^{n+1/2} = \hat{x}_1^n, \qquad \hat{x}_{2p}^{n+1/2} = \hat{x}_2^n \tag{35}$$

However, instead of direct substitution of the above predicted quantity to obtain $u_p^{n+1/2}$ and $\bar{z}_p^{n+1/2}$ based on (31) and (32), equation augmentations are introduced to improve the accuracy of $u_p^{n+1/2}$ and $\bar{z}_p^{n+1/2}$. Of several augmentation procedures that are applicable to construct discrete filters for the computations of $u^{n+1/2}$ and $\bar{z}^{n+1/2}$, we substitute (26) into (31) and (32) to obtain

$$\begin{cases} u^{n+1/2} = -\bar{Z}_1 \hat{x}_1^{n+1/2} - \bar{Z}_2 (\hat{x}_2^n - \delta K \hat{x}_1^{n+1/2} + \\ \qquad \delta B u^{n+1/2} + \delta \bar{L}_2 \bar{z}^{n+1/2}) \\ \bar{z}^{n+1/2} = z^{n+1/2} - \bar{H}_1 \hat{x}_1^{n+1/2} - \\ \qquad \bar{H}_2 (\hat{x}_2^n - \delta K \hat{x}_1^{n+1/2} + \delta B u^{n+1/2} + \delta \bar{L}_2 \bar{z}^{n+1/2}) \end{cases} \tag{36}$$

Rearranging the above coupled equations, one obtains

$$\begin{bmatrix} (I+\delta\bar{Z}_2B) & \delta\bar{Z}_2\bar{L}_2 \\ \delta\bar{H}_2B & (I+\delta\bar{H}_2\bar{L}_2) \end{bmatrix} \begin{Bmatrix} u^{n+1/2} \\ \bar{z}^{n+1/2} \end{Bmatrix} =$$

$$\begin{Bmatrix} -\bar{Z}_2\hat{x}_2^n - (\bar{Z}_1 - \delta\bar{Z}_2K)\hat{x}_1^{n+1/2} \\ z^{n+1/2} - \bar{H}_2\hat{x}_2^n - (\bar{H}_1 - \delta\bar{H}_2K)\hat{x}_1^{n+1/2} \end{Bmatrix} \tag{37}$$

which corresponds to a first order filter to reduce the errors in computing $\hat{x}_2 = M\dot{\hat{x}} + D\hat{x}$. A second-order discrete filter for computing $u$ and $\bar{z}$ can be obtained by differentiating $u$ and $\bar{z}$ to obtain

$$\begin{cases} \dot{u} = -\bar{Z}_1\dot{\hat{x}}_1 - \bar{Z}_2\dot{\hat{x}}_2 \\ \dot{\bar{z}} = \dot{z} - \bar{H}_1\dot{\hat{x}}_1 - \bar{H}_2\dot{\hat{x}}_2 \end{cases} \tag{38}$$

and then substituting $\dot{\hat{x}}_1$ and $\dot{\hat{x}}_2$ from (17). Subsequently, (19) is applied to integrate the equations for $u$ and $\bar{z}$ which yields

$$\begin{bmatrix} I+\delta\bar{Z}_2B+\delta^2\bar{Z}_1M^{-1}B & \delta(\bar{Z}_2\bar{L}_2+\bar{Z}_1\bar{L}_1+\delta\bar{Z}_1M^{-1}\bar{L}_2) \\ \delta(\bar{H}_2B+\delta\bar{H}_1M^{-1}B) & I+\delta\bar{H}_1(\bar{L}_1+\delta M^{-1}\bar{L}_2)+\delta\bar{H}_2\bar{L}_2 \end{bmatrix} \begin{Bmatrix} u^{n+1/2} \\ \bar{z}^{n+1/2} \end{Bmatrix} =$$

$$\begin{Bmatrix} u^n \\ \bar{z}^n \end{Bmatrix} - \delta \begin{Bmatrix} \bar{Z}_1M^{-1}(\hat{x}_2^n - \delta K\hat{x}_1^{n+1/2} - D\hat{x}_1^{n+1/2}) + \bar{Z}_2K\hat{x}_1^{n+1/2} \\ \bar{H}_1M^{-1}(\hat{x}_2^n - \delta K\hat{x}_1^{n+1/2} D\hat{x}_1^{n+1/2}) + \bar{H}_2K\hat{x}_1^{n+1/2} \end{Bmatrix} + \begin{Bmatrix} 0 \\ z^{n+1/2} - z^n \end{Bmatrix} \tag{39}$$

The net effects of this augmentation are to filter out the errors committed in estimating both $\hat{x}_1$ and $\hat{x}_2$. Solution of (39) for $u^{n+1/2}$ and $\bar{z}^{n+1/2}$ permits (25) and (26) to be solved in continuous time for $\hat{x}_1^{n+1/2}$ and $\hat{x}_2^{n+1/2}$. Subsequently, (19b) is used for $\hat{x}_1^{n+1}$ and $\hat{x}_2^{n+1}$.

The preceding augmentation (39) leads to an accurate estimate of the control force and state estimation error correction at the (n+1/2) time step. Although (39) involves the solution of an additional algebraic equation, the equation size is relatively small ( size = number of actuators (m) plus the number of measurements (r) ). Thus, (39) is an efficient method for continuous time simulation of the Kalman filter equations provided the size of (39) is significantly lower than the first order form of (4). The next section discusses the relative efficiency of the present method and the conventional first order solution. More details on the equation augmentation procedure (39) may be found in Park and Belvin[17].

Finally, it is noted that by following a similar time discretization procedure adopted for computing $\hat{x}_1^{n+1/2}$ and $\hat{x}_2^{n+1/2}$, the structural dynamics equation (1) can be solved by

$$\begin{cases} (M+\delta D+\delta^2K)x_1^{n+1/2} = Mx_1^n + \delta x_2^n + \delta^2Bu^{n+1/2} \\ x_2^{n+1/2} = x_2^n + \delta Bu^{n+1/2} - \delta Kx_1^{n+1/2} \end{cases} \tag{40}$$

Thus, numerical solutions of the structural dynamics equation (1) and the filter equation (20) can be carried out within the second-order solution context, thus realizing substantial computational simplicity compared with the solution of first-order systems of equations (4) and the corresponding first-order estimator equations (6).

It is emphasized that the solutions of both the structural displacement $x$ and the reconstructed displacement $\hat{x}$ employ the same solution matrix, $(M+\delta D+\delta^2K)$. The computational stability of the present procedure can be examined as investigated in Park[18] and Park and Felippa[19,20]. The result, when applied to the present case, can be stated as

$$\delta^2\lambda_{\max} \leq 1 \tag{41}$$

where $\lambda_{\max}$ is the maximum eigenvalue of

$$(\lambda^2 I + \lambda \bar{Z}_2 B + \bar{Z}_1 M^{-1} B)y = 0 \tag{42}$$

Typically the control laws are formulated in terms of low-frequency response components, viz.,

$$B \propto G^T K G \tag{43}$$

for the displacement feedback case where $G$ is a projection matrix that extracts only low-frequency components from the structural stiffness matrix. Hence, $\lambda_{\max}$ is in practice several orders of magnitude smaller than $\mu_{\max}$ of the structural dynamics eigenvalue problem:

$$\mu M y = K y \tag{44}$$

Considering that a typical explicit algorithm has its stability limit $\mu_{\max} \cdot h \leq 2$, the maximum step size allowed by (42) is in fact several orders of magnitude larger than allowed by any explicit algorithm.

## Computational Efficiency

Solution of the Kalman filtering equations in second-order form is prompted by the potential gain in computational efficiency due to the beneficial nature of matrix sparsity and symmetry in the solution matrix of the second-order estimator equations. There is an overhead to be paid for the present second-order procedure, that is, the additional computations introduced to minimize the control force and state estimation error terms on the right-hand-side of the resulting discrete equations. The following paragraphs show the second-order solution is most advantageous for estimator models with sparse coefficient matrices $M, D$ and $K$.

Solution of the first order Kalman filter equation (6) or the second-order form (25-26, 39) may be performed using a time discretization as given by (19). For linear time invariant (LTI) systems, the solution matrix is decomposed once and subsequently upper and lower triangular system solutions are performed to compute the estimator state at each time step. Thus, the computations required at each time step result from calculation of the RHS and subsequent triangular system solutions. For the results that follow, the number of floating point operations are estimated for LTI systems of order O(N). In addition, it is assumed that the mass, damping and stiffness matrices ($M, D$ and $K$) are symmetric and banded with bandwidth $\alpha N$, where $0 \leq \alpha \leq (0.5 - \frac{1}{2N})$.

The first-order Kalman filter equation (6) requires $(4N^2 + 2Nr + O(N))$ operations at each time step. The discrete time second-order Kalman filter solution (25-26, 33-34) require $(8\alpha^2 N^2 + 2\alpha N^2 + 3Nm + 4Nr + O(N))$ operations and the continuous time second-order Kalman filter (25-26, 39) require $(8\alpha N^2 + 2\alpha N^2 + 5Nm + 6Nr + (r+m)^2 + O(N))$ operations at each time step. To examine the relative efficiency of the first-order and second-order forms, several cases are presented as follows.

First, a worst case condition is examined whereby $M, D$ and $K$ are fully populated ($\alpha = 0.5 - \frac{1}{2N}$) and $r = m = N$. Only for this extreme condition with large numbers of sensors and actuators relative to the system order, the first order Kalman filter becomes somewhat more efficient than the second-order discrete Kalman filter solution presented herein.

For typical structural systems, $M$ and $K$ are almost always banded. In addition, the number of sensors and actuators is usually small compared to the system order N. If the number of actuators (m) and the

number of measurements (r) are proportional to the bandwidth ( $r = m = \alpha N$), the second-order discrete Kalman filtering equations become computational attractive as long as $\alpha \leq 0.394$. It should be noted that the larger the size of the structural systems, the smaller the bandwidth becomes, with the range of $\alpha$ to be $0.05 \leq \alpha \leq 0.15$.

Finally, for the special case of modal-based structural models, one has $\alpha \rightarrow 0$. For this case, as long as sensors and actuators are sufficiently smaller than the modal degrees of freedom, the present second-order state estimator can be substantially more efficient than the classical first-order form. This is because the conventional state space-based estimator must deal with a fully coupled nonsymmetric $2N \times 2N$ system whereas the present second-order estimator deals with a diagonal $N \times N$ system. A more detailed discussion can be found in Belvin[9].

## Implementation and Numerical Evaluations

The second-order discrete Kalman filtering equation derived in (25) and (26) have been implemented along with the stabilized form of the controller $\mathbf{u}$ and the filtered measurements $\bar{\mathbf{z}}$ in such a way the estimator computational module can be interfaced with the partitioned control-structure interaction simulation package developed previously by Belvin9, Park and Belvin[17] Alvin and Park[21]. It is emphasized that the solution procedure of the present second-order discrete Kalman filtering equations (25) and (26) follows exactly the same steps as required in the solution of symmetric, sparse structural systems. It is this attribute that makes the present discrete filter attractive from the simulation viewpoint. For a succinct comparison between the present CSI simulation procedure and conventional state space-based simulation procedures, the equations that need to be implemented in both of the procedures are summarized below.

### Partitioned Control-Structure Interaction Equations

The partitioned procedure for simulating the control-structure interaction problems developed in Belvin[9] and Park and Belvin[17] exploits the second-order diferential equation form whenever possible as shown below.

$$
\begin{cases}
\text{Structure:} \quad a) & \mathbf{M}\ddot{\mathbf{q}} + \mathbf{D}\dot{\mathbf{q}} + \mathbf{K}\mathbf{q} = \mathbf{f} + \mathbf{B}\mathbf{u} + \mathbf{G}\mathbf{w} \\
& \mathbf{q}(0) = \mathbf{q}_0, \qquad \dot{\mathbf{q}}(0) = \dot{\mathbf{q}}_0 \\
\text{Sensor Output:} \quad b) & \mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{v} \\
\text{Estimator:} \quad c) & \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{Bmatrix} \dot{\tilde{\mathbf{q}}} \\ \tilde{\mathbf{p}} \end{Bmatrix} + \begin{bmatrix} \mathbf{D} & -\mathbf{I} \\ \mathbf{K} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \tilde{\mathbf{q}} \\ \tilde{\mathbf{q}} \end{Bmatrix} = \begin{Bmatrix} \mathbf{0} \\ \mathbf{f} + \mathbf{B}\mathbf{u} \end{Bmatrix} + \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{L}_1 \\ \mathbf{L}_2 \end{bmatrix} \dot{\gamma} \\
& \tilde{\mathbf{q}}(0) = 0, \quad \tilde{\mathbf{p}}(0) = \mathbf{f}(0) + \mathbf{B}\mathbf{u}(0) \\
\text{Control Force:} \quad d) & \dot{\mathbf{u}} + \mathbf{F}_2\mathbf{M}^{-1}\mathbf{B}\mathbf{u} = \mathbf{F}_2(\mathbf{M}^{-1}\dot{\tilde{\mathbf{p}}} + \mathbf{L}_2\gamma) + \mathbf{F}_1\dot{\tilde{\mathbf{q}}} \\
\text{Estimation Error:} \quad e) & \dot{\gamma} + \mathbf{H}_v\mathbf{L}_2\gamma = \dot{\mathbf{z}} - \mathbf{H}_v\mathbf{M}^{-1}(\dot{\tilde{\mathbf{p}}} - \mathbf{B}\mathbf{u}) - \mathbf{H}_d\dot{\tilde{\mathbf{q}}}
\end{cases}
\tag{45}
$$

In addition, notice that the control laws ($\mathbf{u}$) and the estimation error ($\gamma$) are parabolically stabilized and solved in a separate software module from the estimator and the structural analyzers, thus effectively rendering a computaionally efficient and accurate procedure.

### Conventional Control-Structure Interaction Equatioons

In contrast to the partitioned procedure summarized above, conventional control-structure interaction simulation employs a first-order differential equation form as shown below, thus requiring the solution of $2n \times 2n$)-system equations for structures and the observer. In addition, the control laws and the estimation errror are not stabilized, which can give rise to an accumulation of computational errors.

$$\begin{cases} \text{Structure:} & a) & \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{E}\mathbf{f} + \bar{\mathbf{B}}\mathbf{u} + \bar{\mathbf{G}}\mathbf{w} \\ & & \mathbf{x}(0) = \mathbf{x}_0 \\ \text{Sensor Output:} & b) & \mathbf{z} = \mathbf{H}\mathbf{x} + \mathbf{v} \\ \text{Estimator:} & c) & \dot{\tilde{\mathbf{x}}} = \mathbf{A}\tilde{\mathbf{x}} + \mathbf{E}\mathbf{f} + \bar{\mathbf{B}}\mathbf{u} + \mathbf{L}\boldsymbol{\gamma} \\ & & \tilde{\mathbf{x}}(0) = 0 \\ \text{Control Force:} & d) & \mathbf{u} = -\mathbf{F}\tilde{\mathbf{x}} \\ \text{Estimation Error:} & e) & \boldsymbol{\gamma} = \mathbf{z} - \mathbf{H}\tilde{\mathbf{x}} \end{cases} \tag{46}$$

where

$$\mathbf{x} = \begin{Bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{Bmatrix}, \qquad \tilde{\mathbf{x}} = \begin{Bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \end{Bmatrix}$$

and

$$\mathbf{H} = [\mathbf{H}_d \quad \mathbf{H}_v], \qquad \mathbf{L} = \begin{bmatrix} \mathbf{L}_1 \\ \mathbf{L}_2 \end{bmatrix}, \qquad \mathbf{E} = \begin{bmatrix} 0 \\ \mathbf{M}^{-1} \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{D} \end{bmatrix}, \qquad \bar{\mathbf{B}} = \begin{bmatrix} 0 \\ \mathbf{M}^{-1}\mathbf{B} \end{bmatrix}, \qquad \mathbf{F} = [\mathbf{F}_1 \quad \mathbf{F}_2]$$

### Numerical Experiments

The first example is a truss beam shown in Fig. 1, consisting of 8 bays with nodes 1 and 2 fixed for cantilevered motions. Actuator and sensor locations, as well as their orientation, are given in Table 1. In the numerical experiments reported herein, we have relied on the Matlab software package[22] for the synthesis of both the control law gains and the discrete Kalman filter gain matrices. Figures 2, 3 and 4 show the vertical displacement time response at node 9 for open-loop, full state feedback, and dynamically compensated feedback cases, respectively. In the present pepper, a full state feedback corresponds to the case for which the number of sensors are the same as the total system degrees of freedom whereas the dynamically compensated case corresopnds to a smaller number of sensors as compared with the total system degrees of freedom. Note the effectiveness of the dynamically compensated feedback case with four actuators and six sensors as indicated in Table 1 by the present second-order discrete Kalman filtering equations as compared with the full state feedback cases.

Figure 5 illustrates a testbed model of an Earth-pointing satellite. For vibration control, 18 actuators and 18 sensors are configured throughout the system; their locations are provided in Tables 2 and 3. Figures 6, 7, and 8 are a representative of the responses for open-loop, full state feedback, and dynamically compensated cases, respectively. In both examples, the estimator states are the estimated physical displacements and generalized momenta as previously developed, and thus the number of effective states is equal to $2N$, where $N$ is the number of physical displacement variables of the second-order structural system. Therefore, the Kalman filter for the truss example has 108 states, and the filter for the satellite has 1164 states, a substantial increase over typical estimator orders for such systems. Further simulations with the present procedure should

shed light on the performance of dynamically compensated feedback systems for large-scale systems as they are computationally more feasible than heretofore possible.

The computational overhead associated with the full state feedback vs. the use of a dynamic compensation scheme by the present Kalman filtering equations is reported in Table 4. It is seen that the use of the present second-order discrete Kalman filtering equations for constructing dynamically compensated control laws adds computational overhead, only an equivalent of open-loop transient analysis of symmetric sparse systems of order N instead of $2N \times 2N$ dense systems. This is evidenced in Table 4 in that the normalized CPU time for the dynamically compensated case (designated as K. Filter) is 63.16 whereas the total CPUs for the full state feedback case (FSFB) plus that of the open loop dynamic response (Transient) is 64.18.

## Summary

The present paper has addressed the advantageous features of employing the same direct time integration algorithm for solving the structural dynamics equations and for integrating the associated continuous Kalman filtering equations. The time discretization of the resulting Kalman filtering equations is further facilitated by employing a canonical first-order form via a generalized momenta. When used in conjunction with the previously developed stabilized form of control laws in Park and Belvin[17], the present procedure offers a substantial computational advantage over the simulation methods based on a first-order form when computing with large (i.e. nearly full system dynamics) and sparse estimator models.

In order to minimize the deleterious effect of numerical damping and phase distortion in the solution of the discrete Kalman filtering equations, the trapezoidal rule is employed. This is due to the wellknown fact that the trapezoidal rule conserves the system energy with minimum phase error among all the time integration formulas of second-order accuracy[15,16]

Computational stability of the present solution method for the filter equation has been assessed based on the stability analysis result of partitioned solution procedures[18]. To obtain a sharper estimate of the stable integration step size, a more rigorous computational stability analysis is being carried out and will be reported in the future.

## Acknowledgements

The work reported herein was supported by a grant from NASA/Johnson Space Center, NAG 9-574 and a grant from NASA/Langley Research Center, NAG1-1021. The authors thank Dr. John Sunkel of NASA/Johnson Space Center and Dr. Jer-Nan Juang of NASA/Langley Research Center who encouraged us to work on second-order observers.

## References

1. Kwakernaak, H. and Sivan, R., *Linear Optimal Control Systems*, Wiley-Interscience, New York, 1972.
2. Hughes, P. C. and Skelton, R. E., "Controllability and observability of linear matrix second-order systems," *Journal of Applied Mechanics, Trans. ASME*, Vol. 47, 1980, 415-420.
3. Arnold, W. F. and Laub, A. J., "Generalized Eigenproblem Algorithms and Software for Algebraic Riccati Equations," *Proceedings of the IEEE*, Vol. 72, No. 12, 1984, 1746-1754.
4. Bender, D. J. and Laub, A. J., "Controllability and Observability at Infinity of Multivariable Linear Second-Order Models," *IEEE Transactions on Automatic Control*, Vol. AC-30, 1985, 1234-1237.

5. Oshman, Y., Inman, D. J. and Laub, A. J., "Square-Root State Estimation for Second-Order Large Space Structures Models," *Journal of Guidance, Control and Dynamics*, Vol. 12, no. 5, 1989, 698-708.

6. Belvin, W. K. and Park, K. C., "On the State Estimation of Structures with Second Order Observers," *Proc. the 30th Structures, Dynamics and Materials Conference*, AIAA Paper No. 89-1241, 1989.

7. Belvin, W. K. and Park, K. C., "Structural Tailoring and Feedback Control Synthesis: An Interdisciplinary Approach," *Journal of Guidance, Control and Dynamics*, Vol. 13, No. 3, 1990, 424-429.

8. Juang, J. N. and Maghami, P. G., "Robust Eigensystem Assignment for Second-Order Estimators," *Proc. of the Guidance, Navigation and Control Conference*, AIAA Paper No. 90-3474, 1990.

9. Belvin, W. K., "Simulation and Interdisciplinary Design Methodology for Control-Structure Interaction Systems," PhD Thesis, Center for Space Structures and Controls, University of Colorado at Boulder, CO., Report No. CU-CSSC-89-10, July 1989.

10. Kalman, R. E., "On the General Theory of Control Systems," *Proc. 1st International Congress on Automatic Control*, Butterworth, London, Vol. 1, 1961, 481-491.

11. Kalman, R. E. and Bucy, R. S., "New results in linear filtering and prediction theory," *Trans. ASME J. Basic Engineering*, Vol. 83, 1961, 95-108.

12. Hashemipour, H. R. and Laub, A. J., "Kalman filtering for second-order models," *J. Guidance, Control and Dynamics*, Vol. 11, No. 2, 1988, 181-185.

13. Jensen, P. S., "Transient Analysis of Structures by Stiffly Stable Methods," *Computers and Structures*, Vol. 4, 1974, 67-94.

14. Felippa, C. A. and Park, K. C., "Computational Aspects of Time Integration Procedures in Structural Dynamics, Part 1: Implementation," *Journal of Applied Mechanics*, Vol. 45, 1978, 595-602.

15. Dahlquist, G., "A Special Stability Problem for Linear Multi-step Methods," *BIT*, 3, 1963, pp. 27-43.

16. Park, K. C., "Practical Aspects of Time integration," *Comp. & Struct.*, 7, 1977, 343-353.

17. Park, K. C. and Belvin, W. K., "Stability and Implementation of Partitioned CSI Solution Procedures," *Journal of Guidance, Control, and Dynamics*, 14, January-February 1991, 59-67.

18. Park, K. C., "Partitioned Analysis Procedures for Coupled-Field Problems: Stability Analysis," *Journal of Applied Mechanics*, Vol. 47, 1980), 370-378.

19. Park, K. C. and Felippa, C. A., "Partitioned Analysis of Coupled Systems," in: *Computational Methods for Transient Analysis*, T. Belytschko and T. J. R. Hughes (eds.), Elsevier Pub. Co., 1983, 157-219.

20. Park, K. C. and Felippa, C. A.,"Recent Developments in Coupled-Field Analysis Methods," in: *Numerical Methods in Coupled Systems*, Lewis, R. W. et al(editors), John Wiley & Sons, 1984, 327-352.

21. K. F. Alvin and K. C. Park, "Implementation of A Partitioned Algorithm for Simulations of Large CSI Problems," Center for Space Structures and Controls, University of Colorado at Boulder, CO., Report No. CU-CSSC-91-4, March 1991.

22. *PRO-Matlab User's Guide*, The MathWorks, Inc., 1989.

Table 1a

**Actuator Placement for Truss Example Problem**

| Actuator | Node | Component |
|---|---|---|
| 1 | 2 | $y$ |
| 2 | 18 | $y$ |
| 3 | 9 | $y$ |
| 4 | 9 | $x$ |

Table 1b

**Sensor Placement for Truss Example Problem**

| Sensor | Type | Node | Component |
|---|---|---|---|
| 1 | Rate | 2 | $y$ |
| 2 | Rate | 18 | $y$ |
| 3 | Rate | 9 | $y$ |
| 4 | Rate | 9 | $x$ |
| 5 | Position | 9 | $y$ |
| 6 | Position | 9 | $x$ |

Table 2

**Actuator Placement for EPS Example Problem**

| Actuator | Node | Component |
|---|---|---|
| 1 | 97 | $x$ |
| 2 | 97 | $z$ |
| 3 | 96 | $x$ |
| 4 | 96 | $z$ |
| 5 | 65 | $y$ |
| 6 | 68 | $y$ |
| 7 | 59 | $y$ |
| 8 | 62 | $y$ |
| 9 | 45 | $y$ |
| 10 | 45 | $z$ |
| 11 | 70 | $y$ |
| 12 | 70 | $z$ |
| 13 | 95 | $x$ |
| 14 | 95 | $y$ |
| 15 | 95 | $z$ |
| 16 | 95 | $\phi_x$ |
| 17 | 95 | $\phi_y$ |
| 18 | 95 | $\phi_z$ |

Table 3

**Sensor Placement for EPS Example Problem**

| Sensor | Type | Node | Component |
|---|---|---|---|
| 1 | Rate | 97 | $x$ |
| 2 | Rate | 97 | $z$ |
| 3 | Rate | 96 | $x$ |
| 4 | Rate | 96 | $z$ |
| 5 | Rate | 65 | $y$ |
| 6 | Rate | 68 | $y$ |
| 7 | Rate | 59 | $y$ |
| 8 | Rate | 62 | $y$ |
| 9 | Rate | 45 | $y$ |
| 10 | Rate | 45 | $z$ |
| 11 | Rate | 70 | $y$ |
| 12 | Rate | 70 | $z$ |
| 13 | Position | 95 | $x$ |
| 14 | Position | 95 | $y$ |
| 15 | Position | 95 | $z$ |
| 16 | Position | 95 | $\phi_x$ |
| 17 | Position | 95 | $\phi_y$ |
| 18 | Position | 95 | $\phi_z$ |

Table 4

**CPU Results for ACSIS Sequential and Parallel Versions**

| Model | Problem Type | Sequential | Parallel |
|---|---|---|---|
| 54 DOF | Transient | 4.5 | 5.6 |
| Truss | FSFB | 9.4 | 10.2 |
| | K. Filter | 13.0 | 10.7 |
| 582 DOF | Transient | 98.6 | 100.3 |
| EPS7 | FSFB | 190.2 | 294.5 |
| | K. Filter | 284.2 | 321.5 |

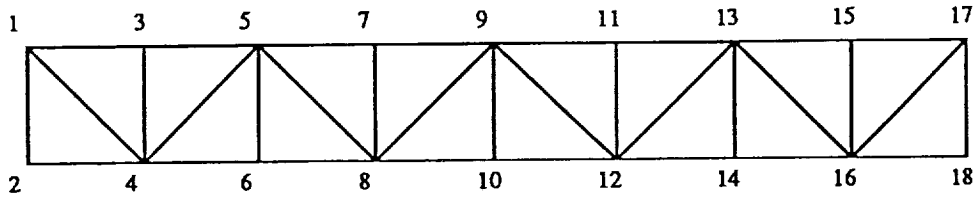



**Figure 1: TRUSS BEAM PROBLEM**

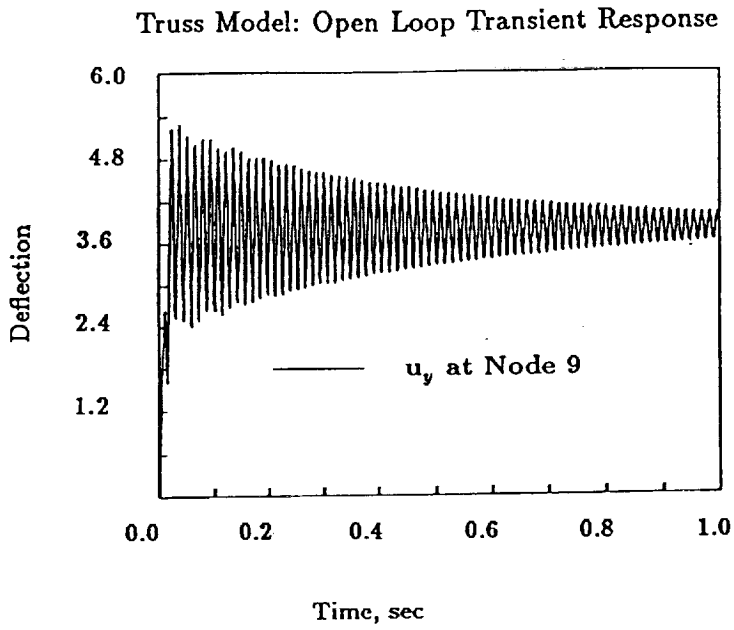



**Figure 2: Truss Transient Response**

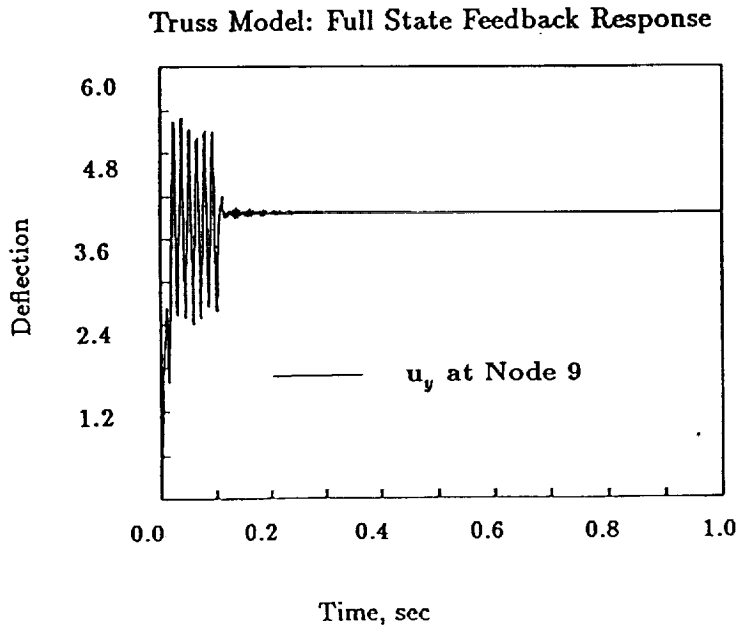



**Figure 3: TRUSS FULL STATE FEEDBACK RESPONSE**

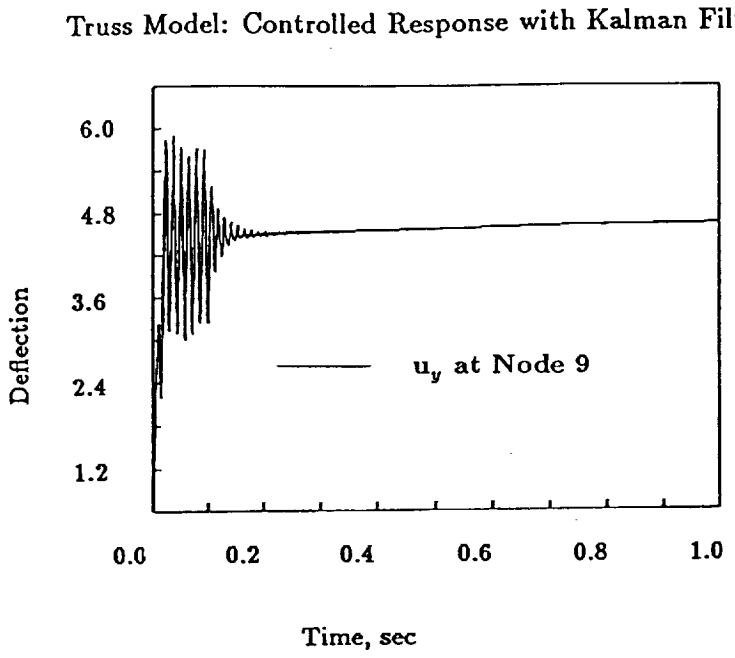



**Figure 4: Truss Response with Filter**

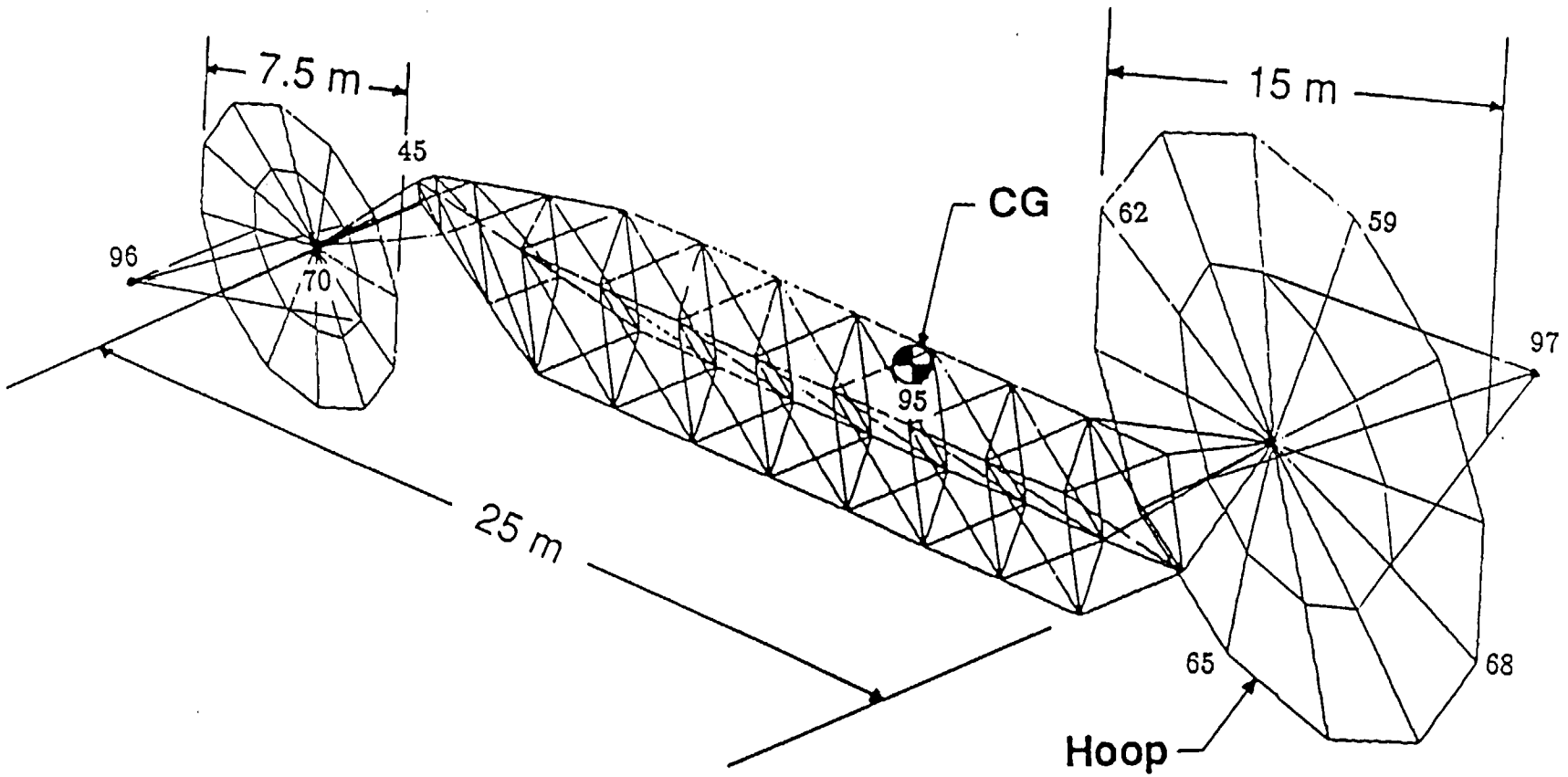



Figure 5: GENERIC EARTH POINTING SATELLITE EXAMPLE

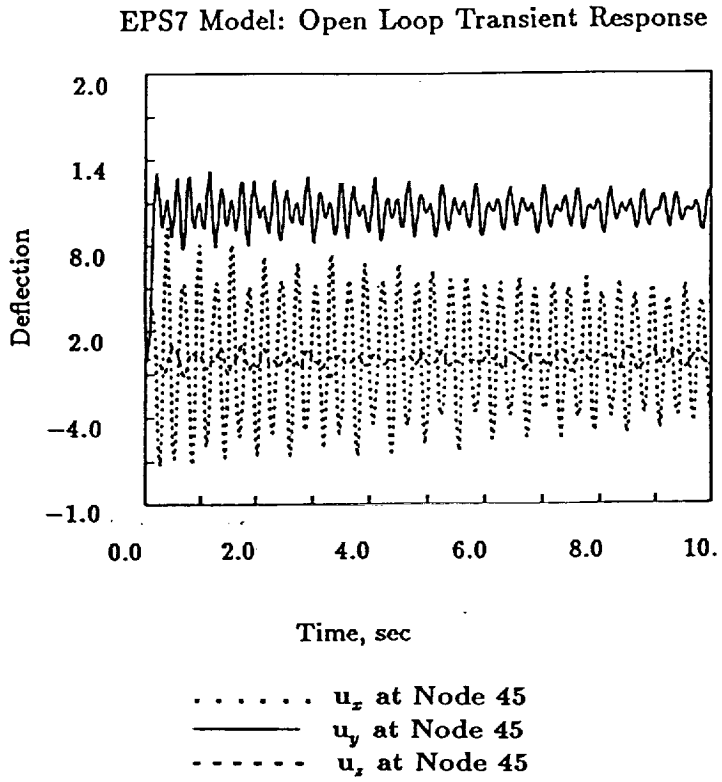
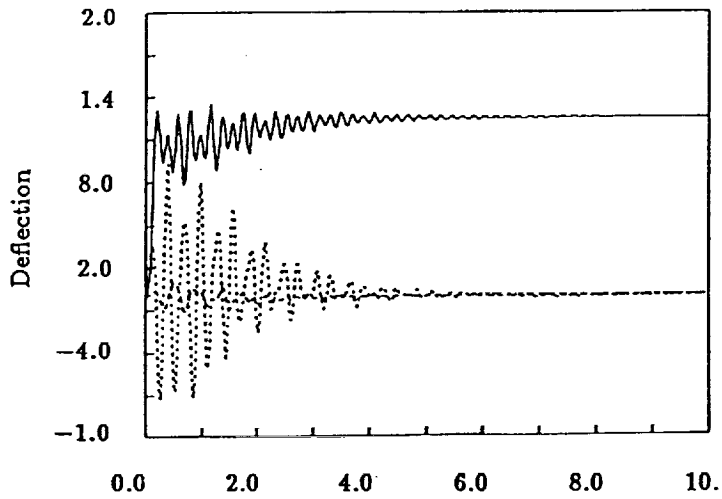



**Figure 6: EPS TRANSIENT RESPONSE**

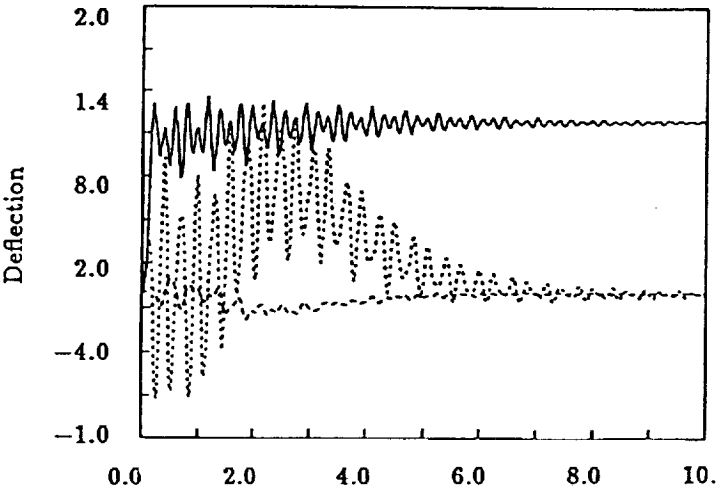
EPS7 Model: Full State Feedback Response




. . . . . $u_x$ at Node 45
——— $u_y$ at Node 45
- - - - - - $u_z$ at Node 45

**Figure 7: EPS FULL STATE FEEDBACK RESPONSE**

Model: Controlled Response w/Kalman Filter




. . . . . $u_x$ at Node 45
——— $u_y$ at Node 45
- - - - - - $u_z$ at Node 45

**Figure 8: EPS RESPONSE with KALMAN FILTER**