

DETC2015-47812

DISCRETE TIME FINITE ELEMENT TRANSFER MATRIX METHOD DEVELOPMENT  
FOR MODELING AND DECENTRALIZED CONTROL

**Nick Cramer\***

Baskin School of Engineering  
UC Santa Cruz, CA, USA  
School of Engineering  
Email: nrcramer@ucsc.edu

**Sean Swei**

Intelligent Systems Division,  
NASA Ames Research Center  
Naval Air Station, Moffett Field  
Mountain View, CA  
Email: sean.s.swei@nasa.gov

**Kenny Cheung**

NASA Ames Research Center  
Naval Air Station, Moffett Field  
Mountain View, CA  
Email: kenny@nasa.gov

**M. Teodorescu**

Baskin School of Engineering  
UC Santa Cruz, CA, USA  
School of Engineering  
Email: mteodorescu@soe.ucsc.edu

**ABSTRACT**

*The current emphasis on increasing aeronautical efficiency is leading the way to a new class of lighter more flexible airplane materials and structures, which unfortunately can result in aeroelastic instabilities.*

*To effectively control the wings deformation and shape, appropriate modeling is necessary. Wings are often modeled as cantilever beams using finite element analysis. The drawback of this approach is that large aeroelastic models cannot be used for embedded controllers. Therefore, to effectively control wings shape, a simple, stable and fast equivalent predictive model that can capture the physical problem and could be used for in-flight control is required.*

*The current paper proposes a Discrete Time Finite Element Transfer Matrix (DT-FETMM) model beam deformation and use it to design a regulator. The advantage of the proposed approach over existing methods is that the proposed controller could be designed to suppress a larger number of vibration modes within the fidelity of the selected time step. We will extend the discrete*

*time transfer matrix method to finite element models and present the decentralized models and controllers for structural control.*

**Nomenclature**

$A_n$  = Acceleration integration scaling value for the  $n^{th}$  node  
 $B_n$  = Acceleration integration constant for the  $n^{th}$  node  
 $C_{ij}$  = Galerkin finite element damping matrix sub-block  
 $D_n$  = Velocity integration constant for the  $n^{th}$  node  
 $E_n$  = Velocity integration scaling value for the  $n^{th}$  node  
 $f_n$  = Control input force at  $n^{th}$  node  
 $F_n$  = Forward propagation matrix of the right force for the  $n^{th}$  node  
 $H_n$  = Reverse propagation matrix of the left force for the  $n^{th}$  node  
 $J_n$  = Reverse propagation matrix for the  $n^{th}$  node  
 $K_{ij}$  = Galerkin finite element elastic matrix sub-block  
 $M_{ij}$  = Galerkin finite element mass matrix sub-block  
 $P_n$  = Forward propagation matrix for the  $n^{th}$  node  
 $Q_n$  = Transfer matrix from left boundary condition to  $n^{th}$  node  
 $T_n$  = Transfer matrix from right boundary condition to  $n^{th}$  node  
 $v_n$  = Propagation vector

\*Address all correspondence to this author

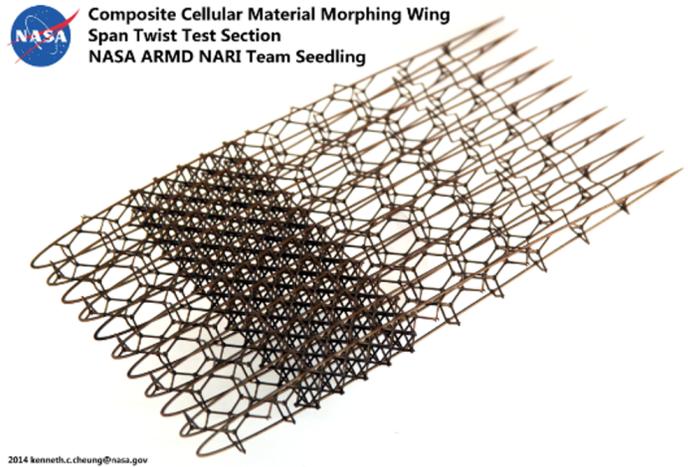
$x_n = n^{th}$  node position states  
 $\dot{x}_n = n^{th}$  node velocity states  
 $\ddot{x}_n = n^{th}$  node acceleration states  
 $\tau_n =$  Internal forces at the  $n^{th}$  node for either left or right side

## 1 Introduction

Aerodynamic efficiency is one of the most sought after goals for aerodynamic design. It has been shown that active in-flight aeroelastic shape control of wing twist and deflection at local wing sections can improve overall aerodynamic efficiency [1–3]. Two recent NASA studies to realize in-flight aeroelastic wing shape control are the Variable Camber-Continuous Trailing Edge Flap (VCCTEF) and Distributed Electric Propulsion (DEP) concepts. The VCCTEF strategy divides a trailing edge into many flap sections, each of which can be individually controlled subject to the constraints of an interstitial elastomeric interpolating surface. Chord-wise and span-wise adaptive wing shaping for various flight scenarios can be achieved in this manner. DEP proposes to accomplish similar shape morphing modes, but with differential control of spatially distributed electric propulsion systems.

A significant number of direct actuation methods proposed in literature focus either on developing a variable camber techniques [4] or on twisting the wing to morph in a suitable shape. For example, Majji *et al.* [5, 6] proposed a variable twist wing that had three sections that were all capable of independent variable twist actuation. The advantage of this approach is that the slope of lift drag line for a given angle of attack could be controlled by the set point of the other twist positions. Vos *et al.* [7] addressed the issue of the skin by using more traditional wind box and a carbon-fiber-reinforced polymer. They also showed that the lift, drag, and lift drag ratio for given angle of attacks could be controlled with the amount of twist.

The current paper proposes a new approach for the modeling and control of cellular based composite materials. [8] We view cellular based composite materials as an enabling technology for wing shaping control and are therefore critical to develop techniques to model and control them. Cellular composite materials are individual components that can be incrementally assembled, modified, and repaired without requiring custom tooling, for high-performance structures. Fiber composites have been widely used to construct truss cores [9] and structural frames [10]. These have all entailed the development of processes to continuously wind the fibers through a structure, and subsequently cure a resin matrix around them. Instead, the approach taken with the cellular composite materials, uses parts with carbon-fiber beams and loops that are reversibly linked. The design of the reversible linked components is very important step to achieving the stated goals by allowing the global material characteristics to be governed by the way in which the components are assembled. This provides the necessary design fidelity to



**FIGURE 1.** Lattice-based composite cellular wing structure.

achieve the shape morphing wing, while the components once assemble do act as continuous mediums. [11]

Modeling of high dimensional lattice structures using conventional FEM approach can be a challenge, mainly because it is difficult to analyze and visualize the integrated lattice structure in real-time. In this paper, we propose to use the concept of discrete-time finite element transfer matrix method (DT-FE-TMM) to model and analyze large structural systems. The wing structure shown in Figure 1 has a dense wing box and shows the general structure that allows the uses of global finite element. The damping incorporates extremely low internal damping of the carbon fiber and the friction due to the reversible attachment of the parts. The basic concept behind this approach was inspired by the work of Tan *et al.* [12], where the notion of modified transfer matrix method (M-TMM) approach was developed by utilizing the dynamic stiffness matrix of finite element. The primary goal there was to reduce the computational efforts involved in structural analysis. In this paper, in addition to the incorporation of the notion of M-TMM, we also utilize the numerical integration approach proposed in Kumar and Sankar [13] and develop a reduced order discrete-time state-space model that is best suited for control synthesis. An optimal decentralized LQR controller is then designed for the reduced order model, and it is demonstrated that the overall structural system performance is comparable with that of the optimal LQR controller with full order model. In previous work Cramer *et al.* used a similar approach with the lumped-mass system that showed very promising results. [14]

Work has been done previously using the DT-TMM method as a means of control for flexible robots [15, 16] as well as multi-body systems. [17–19] These works have shown that the transfer matrix method can be used to design and create efficient controllers, though to the authors knowledge none of them has specifically addressed the use of transfer matrix method for struc-

tural control.

This paper is organized as follows. Section 2 introduces the mathematical modeling techniques used in the paper: the discrete-time transfer matrix method, the reduced-order model as well as the proposed transfer matrix method. Section 3 focuses on developing and formulating the decentralized controls, and Section 4 contains numerical simulations and discussions. Concluding remarks are made in Section 5.

## 2 Modeling

The general discrete time transfer matrix method (DT-TMM) is a discrete time extension of the statistical mechanics transfer matrix method. DT-TMM uses the repetitive nature of adjoining subsystems to determine the contribution of each node. From that point there is a simple and natural expansion detailing the importance of each nodes contributions to the overall system. The adaptation of this method to work with the Galerkin finite element method (GFEM) that will be shown in this section needs to be derived from the basic equation of motion for the GFEM

$$\begin{bmatrix} M_{n-1,n-1} & M_{n-1,n} & 0 \\ M_{n,n-1} & M_{n,n} & M_{n,n+1} \\ 0 & M_{n+1,n} & M_{n+1,n+1} \end{bmatrix} \ddot{X} + \begin{bmatrix} C_{n-1,n-1} & C_{n-1,n} & 0 \\ C_{n,n-1} & C_{n,n} & C_{n,n+1} \\ & C_{n+1,n} & C_{n+1,n+1} \end{bmatrix} \dot{X} + \begin{bmatrix} K_{n-1,n-1} & K_{n-1,n} & 0 \\ K_{n,n-1} & K_{n,n} & K_{n,n+1} \\ 0 & K_{n+1,n} & K_{n+1,n+1} \end{bmatrix} X + \begin{bmatrix} f_{n-1} \\ f_n \\ f_{n+1} \end{bmatrix} = \begin{bmatrix} F_{n-1} \\ F_n \\ F_{n+1} \end{bmatrix} \quad (3)$$

Using equation 3 together with the configuration of equation 2  $\tau_n^L(t_i)$  and  $\tau_n^R(t_i)$  can be determined as shown below in equations 4 and 5.

$$\tau_n^L = M_{n,n-1} \ddot{x}_{n-1} + [C_{n,n-1} \ C_{n,n}] \begin{bmatrix} \dot{x}_{n-1} \\ \dot{x}_n \end{bmatrix} + [K_{n,n-1} \ K_{n,n}] \begin{bmatrix} x_{n-1} \\ x_n \end{bmatrix} \quad (4)$$

$$\tau_n^R = M_{n,n+1} \ddot{x}_{n+1} + [C_{n,n} \ C_{n,n+1}] \begin{bmatrix} \dot{x}_n \\ \dot{x}_{n+1} \end{bmatrix} + [K_{n,n} \ K_{n,n+1}] \begin{bmatrix} x_n \\ x_{n+1} \end{bmatrix} \quad (5)$$

Which by using the linear integral representation in equation 6 we can convert it into a discrete time system, show in 7 and the discrete time left and right forces are shown in equations 8 and 9, respectively.

$$\begin{aligned} \ddot{x} &= A_n(t_i)x_n(t_i) + B_n(t_i) \\ \dot{x} &= D_n(t_i)x_n(t_i) + E_n(t_i) \end{aligned} \quad (6)$$

where  $A_n$  and  $D_n$  are typically constants related to the time step and  $B_n$  and  $E_n$  contain the previous time steps information.

$$M_{n,n}[A_n(t_i)x_n(t_i) + B_n(t_i)] = \tau_n^R(t_i) - \tau_n^L(t_i) + f_n \quad (7)$$

shown in equation 1,

$$M\ddot{X} + C\dot{X} + KX = F \quad (1)$$

where  $M$ ,  $C$ ,  $K$  are the mass, structural damping, stiffness matrices respectively, the sub-matrix of these general matrices can be seen in Appendix A. Two elements are the minimum set of elements that can be fully represented for this method. The minimum set of elements require three nodes, each of which represent a set of states. The minimum set of elements can be seen in equation 3.

$$M_{n,n}\ddot{x}_n(t_i) = \tau_n^R(t_i) - \tau_n^L(t_i) - f_n \quad (2)$$

The equation of state of the node of interest ( $n$ ) can be seen in equation 2. Where  $\tau_n^R(t_i)$  represents the contributions of the internal forces from the elements to the right at time  $t_i$ , which is the time at the  $i^{th}$  time step and  $\tau_n^L(t_i)$  represents the same for the left side.  $f_n$  is the external force contribution.

$$\tau_n^L = M_{n,n-1}[A_{n-1}(t_i)x_{n-1}(t_i) + B_{n-1}(t_i)] + [C_{n,n-1} \ C_{n,n}] \begin{bmatrix} D_{n-1}(t_i)x_{n-1}(t_i) + E_{n-1}(t_i) \\ D_n(t_i)x_n(t_i) + E_n(t_i) \end{bmatrix} + [K_{n,n-1} \ K_{n,n}] \begin{bmatrix} x_{n-1} \\ x_n \end{bmatrix} \quad (8)$$

$$\tau_n^R = M_{n,n+1}[A_{n+1}(t_i)x_{n+1}(t_i) + B_{n+1}(t_i)] + [C_{n,n} \ C_{n,n+1}] \begin{bmatrix} D_n(t_i)x_n(t_i) + E_n(t_i) \\ D_{n+1}(t_i)x_{n+1}(t_i) + E_{n+1}(t_i) \end{bmatrix} + [K_{n,n} \ K_{n,n+1}] \begin{bmatrix} x_n \\ x_{n+1} \end{bmatrix} \quad (9)$$

### 2.1 Left to Right

In order to determine the contributions of a node to the left of the node in interest the method of prorogation from the left to the right. The discrete equation of motion from equation 7 can be altered into a matrix form as in equation 10.

$$\begin{Bmatrix} x \\ \tau \\ 1 \end{Bmatrix}_n^R = \begin{bmatrix} 1 & 0 & 0 \\ M_n A_n & 1 & M_n B_n - f_n \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ \tau \\ 1 \end{Bmatrix}_n^L \quad (10)$$

For convenience we split the  $x_n$  state (vector) into left and right components. Given that it is important to remember that  $x_n^R = x_n^L$  and will be represented as  $x_n$ . As a short form the matrix equation from 10 can be represented as 21.

$$v_n^L = P_n v_{n-1}^R. \quad (11)$$

where,  $v_n^L = [x_n^L, \tau_n^L, 1]^T$  and  $v_{n-1}$  follows the same notation. We now have the capability of adding in external forces and quantifying some of contributions of a state to its self, to continue the left to right propagation the next set of states to the left must be related to the node. Equation 12 shows the means of propagating the previous states contributions to the left vector.

$$\begin{aligned} \tau_{n-1}^R = & M_{n-1,n}[A_n(t_i)x_n(t_i) + B_n(t_i)] + [C_{n-1,n-1} \ C_{n-1,n}] \\ & \left[ \begin{array}{c} D_{n-1}(t_i)x_{n-1}(t_i) + E_{n-1}(t_i) \\ D_n(t_i)x_n(t_i) + E_n(t_i) \end{array} \right] + [K_{n-1,n-1} \ K_{n-1,n}] \begin{bmatrix} x_{n-1} \\ x_n \end{bmatrix} \end{aligned} \quad (12)$$

Using equation 12 we can solve for the position states  $x_n$  as a combination of  $x_{n-1}$ ,  $\tau_{n-1}$ , and previous state knowledge.

$$x_n = F_{c-1}(F_c^{n-1}x_{n-1} - \tau_{n-1}^R + F_B^n) \quad (13)$$

where,

$$F_{c-1}^n = (-C_{n-1,n}D_n - K_{n-1,n} - M_{n-1,n}A_n)^{-1} \quad (14)$$

$$F_c^n = C_{n,n}D_n + K_{n,n} \quad (15)$$

and,

$$F_B^n = C_{n-1,n-1}E_{n-1} + C_{n-1,n}E_n + M_{n-1,n}B_n \quad (16)$$

which can then be used to determine the force states  $\tau_n^L$ .

$$\tau_n^L = C_{n,n}E_n + C_{n,n-1}E_{n-1} + M_{n,n-1}B_{n-1} + F_c^n F_{c-1}^{n-1} \tau_{n-1}^R \\ (F_c^n F_{c-1}^{n-1} F_c^{n-1} + F_A^{n-1})x_{n-1} + F_c F_{c-1} F_B^n \quad (17)$$

where,

$$F_A^{n-1} = K_{n,n-1} + C_{n,n-1}D_{n-1} + M_{n,n-1}A_{n-1} \quad (18)$$

equations 13 and 17 can be rearranged to create the matrix relation between the previous right states and the node of interests

left states.

$$\begin{Bmatrix} x \\ \tau \\ 1 \end{Bmatrix}_n^L = \begin{bmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ \tau \\ 1 \end{Bmatrix}_{n-1}^R \quad (19)$$

where,

$$\begin{aligned} F_{11} &= F_{c-1}^n F_c^{n-1} \\ F_{12} &= F_{c-1}^n \\ F_{13} &= F_{c-1}^n F_B^n \\ F_{21} &= ((C_{n,n}D_n + K_{n,n} + M_{n,n-1}A_{n-1})F_{c-1}^n F_c^{n-1} + F_A^{n-1}) \\ F_{22} &= F_c^{n-1} F_{c-1}^n \\ F_{23} &= F_B^{n-1} + F_A^{n-1} F_{c-1}^n F_B^n \end{aligned} \quad (20)$$

equation 19 can be represented as equation 21.

$$v_n^L = F_n v_{n-1}^R. \quad (21)$$

## 2.2 Right to Left

While only one directionality is necessary for DT-TMM for simulation in order to create the control centric model propagation from both directions is necessary. Once again using 7 a matrix form can be created.

$$\begin{Bmatrix} x \\ \tau \\ 1 \end{Bmatrix}_n^L = \begin{bmatrix} 1 & 0 & 0 \\ -M_n A_n & 1 & -M_n B_n + f_n \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ \tau \\ 1 \end{Bmatrix}_n^R \quad (22)$$

In this case the right to left matrix will be represented as  $J_n$ .

$$v_n^L = J_n v_n^R. \quad (23)$$

The right to left propagation requires the the nodes to the right of the node of interest must be related. This is done by using the left force component of the next node.

$$\begin{aligned} \tau_{n+1}^L = & M_{n+1,n}[A_n(t_i)x_n(t_i) + B_n(t_i)] + [C_{n+1,n} \ C_{n+1,n+1}] \\ & \left[ \begin{array}{c} D_n(t_i)x_n(t_i) + E_n(t_i) \\ D_{n+1}(t_i)x_{n+1}(t_i) + E_{n+1}(t_i) \end{array} \right] + [K_{n+1,n} \ K_{n+1,n+1}] \begin{bmatrix} x_n \\ x_{n+1} \end{bmatrix} \end{aligned} \quad (24)$$

from there the current nodes right positions states can be determined.

$$x_n = H_{c-1}(F_c^{n+1}x_{n+1} - \tau_{n+1}^L + H_B^n) \quad (25)$$

where,

$$H_{c^{-1}} = (-C_{n+1,n}D_n - K_{n+1,n} - M_{n+1,n}A_n)^{-1} \quad (26)$$

and,

$$H_B^n = C_{n+1,n+1}E_{n+1} + C_{n+1,n}E_n + M_{n+1,n}B_n \quad (27)$$

Which can then be used to determine the right force states.

$$\tau_n^R = F_c^n H_{c^{-1}} \tau_{n+1}^L + F_c^n H_{c^{-1}} H_B^n + M_{n,n+1}B_{n+1} + C_{n,n}E_n + C_{n,n+1}E_{n+1} (F_c^n H_{c^{-1}} F_c^{n+1} + H_A^{n+1}) x_{n+1} \quad (28)$$

where,

$$H_A^{n+1} = C_{n,n+1}D_{n+1} + K_{n,n+1} + M_{n,n+1}A_{n+1} \quad (29)$$

The two above equations can then be adapted to matrix form to give equation 30.

$$\begin{Bmatrix} x \\ \tau \\ 1 \end{Bmatrix}_n^L = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{Bmatrix} x \\ \tau \\ 1 \end{Bmatrix}_{n+1}^R \quad (30)$$

where,

$$\begin{aligned} H_{11} &= H_{c^{-1}} F_c^{n+1} \\ H_{12} &= H_{c^{-1}} \\ H_{13} &= H_{c^{-1}} H_B^n \\ H_{21} &= F_c^n H_{c^{-1}} F_c^{n+1} + H_A^{n+1} \\ H_{22} &= F_c^n H_{c^{-1}} \\ H_{23} &= F_c^n H_{c^{-1}} H_B^n + M_{n,n+1}B_{n+1} + C_{n,n}E_n + C_{n,n+1}E_{n+1} \end{aligned} \quad (31)$$

The right to left propagation will be represented as,

$$v_n^R = H_n v_{n+1}^L \quad (32)$$

## 2.3 Relating the Left Boundary Conditions to current node $n$

In this section, we establish the notation for the matrix relation between the left most boundary conditions and nodal states of  $n$ . From equations 11 and 21

$$v_n^R = P_n F_n v_{n-1}^R, \quad (33)$$

and we can continue this process until we reach subsystem  $n$ ,

$$v_n^L = F_n Q_n v_0^R, \quad (34)$$

where

$$Q_n = \prod_{i=0}^n P_i F_i \quad (35)$$

and  $Q_n$  denotes the transfer function relating the left boundary condition to node  $n$ . Note that  $v_0^R$  represents boundary conditions at the left edge.

## 2.4 Relating the Right Boundary Condition to current node $n$

Following the general approach described in section 2.3,  $v_n^R$  can be computed from equations 23 and 32:

$$v_n^R = H_{n+1} J_{n+1} v_{n+1}^R, \quad (36)$$

This recursive methodology can be used to compute:

$$v_n^R = T_n v_m^R, \quad (37)$$

where  $v_m^R$  is the boundary conditions at right edge, and

$$T_n = \prod_{i=m}^n H_i J_i, \quad (38)$$

and  $T_n$  represents the transfer function from node  $m$  to node  $n$ .

## 3 Decentralized Control Problem Formulation

The previous sections developed the underlying theory needed to create a DT-FE-TMM control centric model. In this section a single node decentralized control will be formulated as will a sub block version that will contain the direct neighboring nodes as well but focus on controlling the center node.

### 3.1 Single Node Controller Formulation

Equation 39 represents the relation of the left most boundary condition to the node in interest. Equation 40 shows the same process but propagating from the right boundary condition instead of the left. Equation 41 is the final component of the system of equation that relates the left side determined from equation 39 to the right side determined by equation 40.

$$\begin{bmatrix} Q_{11} & Q_{12} & Q_{13} \\ Q_{21} & Q_{22} & Q_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ \tau_0 \\ 1 \end{bmatrix} = \begin{bmatrix} H_{11}^n & H_{12}^n & H_{13c}^n (M_{n,n-1}B_{n-1} + C_{n,n-1}E_{n-1} + C_{n,n}E_n) \\ H_{21}^n & H_{22}^n & H_{23c}^n M_{n-1,n}B_n + M_{n,n-1}B_{n-1} + (H_{23c}^n C_{n,n-1} + C_{n-1,n-1})E_{n-1} + (H_{23c}^n C_{n,n} + C_{n-1,n})E_n \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_n \\ \tau_n^L \\ 1 \end{bmatrix} \quad (39)$$

$$\begin{bmatrix} T_{11} & T_{12} & T_{13} \\ T_{21} & T_{22} & T_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_m \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} F_{11}^{n+1} & F_{12}^{n+1} & F_{13c}^{n+1}(C_{n,n}E_n + C_{n,n+1}E_{n+1} + M_{n,n+1}B_{n+1}) \\ F_{21}^{n+1} & F_{22}^{n+1} & F_{23c}^{n+1}M_{n+1,n}B_n + M_{n,n+1}B_{n+1} + (F_{23c}^{n+1}C_{n,n+1} + C_{n+1,n+1})E_{n+1} + (F_{23c}^n C_{n,n} + C_{n+1,n})E_n \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_n \\ \tau_n^R \\ 1 \end{bmatrix} \quad (40)$$

$$M_{n,n}(A_n x_n + B_n) = \tau_n^R - \tau_n^L + f_n \quad (41)$$

Equations 39, 40, and 41 can be reduced into equation 42.

$$\begin{aligned} C_{x_n} x_n = & C_{E_{n-1}} E_{n-1} + C_{E_n} E_n + C_{E_{n+1}} E_{n+1} + C_R + C_L \\ & + C_{B_{n-1}} B_{n-1} + C_{B_{n+1}} + C_{B_n} B_n + f_n \end{aligned} \quad (42)$$

The representation of the constants from equation 42 can be found in Appendix B. For the purpose of this paper we selected the integration technique to be the third order Houbolt integration scheme, represented in equation 43. The Houbolt integration scheme was selected for its combination of simplicity and robustness.

$$\begin{aligned} A_n(t_i) &= \frac{2}{\Delta T^2} \\ B_n(t_i) &= -\frac{1}{\Delta T^2} [5x(t_{i-1}) - 4x(t_{i-2}) + x(t_{i-3})] \\ D_n(t_i) &= \frac{11}{6\Delta T} \\ E_n(t_i) &= -\frac{1}{6\Delta T} [18x(t_{i-1}) - 9x(t_{i-2}) + 2x(t_{i-3})] \end{aligned} \quad (43)$$

Substituting the Houbolt integration scheme into equation 42 allows it to be rearranged the previous states to be related to the current state in a controllable form in 44.

$$X(t_i) = AX(t_{i-1}) + B(\alpha + f_n), \quad (44)$$

where the system matrices  $(A, B)$  and the exogenous input  $\alpha$  are given by

$$A = \begin{bmatrix} \frac{1}{C_{x_n}} \left( \frac{-3C_{E_n}}{\Delta T} + \frac{5C_{B_n}}{\Delta T^2} \right) & \frac{1}{C_{x_n}} \left( \frac{3C_{E_n}}{2\Delta T} + \frac{-4C_{B_n}}{\Delta T^2} \right) & \frac{1}{C_{x_n}} \left( \frac{C_{E_n}}{3\Delta T} + \frac{C_{B_n}}{\Delta T^2} \right) \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (45)$$

$$B = \begin{bmatrix} \frac{1}{C_{x_n}} \\ 0 \\ 0 \end{bmatrix} \quad (46)$$

$$\alpha = C_{E_{n-1}} E_{n-1} + C_{E_{n+1}} E_{n+1} + C_{B_{n-1}} B_{n-1} + C_{B_{n+1}} B_{n+1} + C_R + C_L \quad (47)$$

equation 44 is the equation of motion for the node  $n$ , where  $\alpha$  consists of the contribution of forces from neighboring nodes and

**TABLE 1.** Parameters for simulated finite element beam model

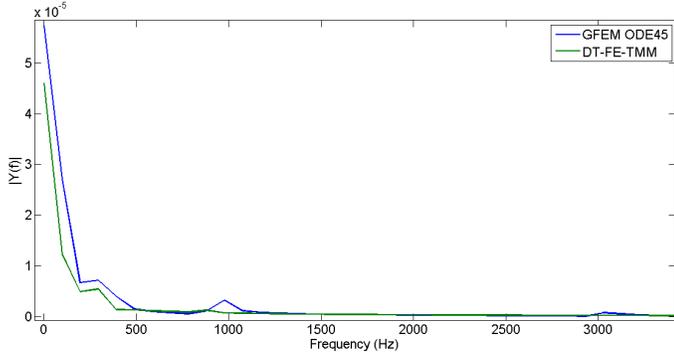
Parameter	Value
Number of Elements	20
Density	946 $\frac{kg}{m^3}$
Modulus of Elasticity	1.5GPa
Viscosity	0.0809Pas
Cross Section Area	3.26e <sup>-3</sup>
Area Moment of Inertia	7.1e <sup>-3</sup>
Element Length	50mm
Time Step	10 $\mu$ s

$f_n$  is the control input applied to the node  $n$ . To design a stabilizing decentralized controller, we consider  $\alpha$  to be a "weak" coupling between node  $n$  and its neighbors, and therefore we can treat node  $n$  as a local subsystem. [20] This is unfortunately is a much more difficult criteria to meet in a finite element method than it was using the lumped mass-spring-damper system in previous works. [14] Like the previous work the time step is crucial to being able to create an effective control though there is not a direct scalar relationship between desired model parameters and the time step as there was in the work on lumped mass-spring-damper systems.

It is important to note that  $Q_{i,i}$ , and,  $T_{i,i}$  contain information on the configuration of the elements that were present between subsystem  $n$  and the boundary conditions. The configuration of  $C_{x_n^R}$  and  $C_{E_n}$  also change depending on the boundary conditions.

### 3.2 Sub-Block Controller Formulation

Much of the work done in this field has been focused on the use of sub-block or substructure analysis for control. [21–23] This has largely been done to take advantage of the work done by Ikeda et. al. on decentralized control of overlapping subsystems. [24] We will be taking a similar approach in developing the sub-block systems so that we will be able to eventually compare to other methods. The block equation of state can be represented by equation 48.



**FIGURE 2.** Fast Fourier Transforms of the tip of the beam for both GFEM and DT-FE-TMM.

$$\begin{bmatrix} X_{n-1} \\ X_n \\ X_{n+1} \end{bmatrix}_{t_i} = A \begin{bmatrix} X_{n-1} \\ X_n \\ X_{n+1} \end{bmatrix}_{t_{i-1}} + \begin{bmatrix} 0 \\ B_n \\ 0 \end{bmatrix} (\alpha + f_n) \quad (48)$$

where,

$$A = \begin{bmatrix} A_{n-1,n-1} & A_{n-1,n} & 0 \\ A_{n,n-1} & A_{n,n} & A_{n,n+1} \\ 0 & A_{n+1,n} & A_{n+1,n+1} \end{bmatrix} \quad (49)$$

where the  $A_{ii}$  are represented by equation 45 and  $B_n$  is represented by equation 46. The off diagonal components  $A_{ij}$  are represented by equation 50.

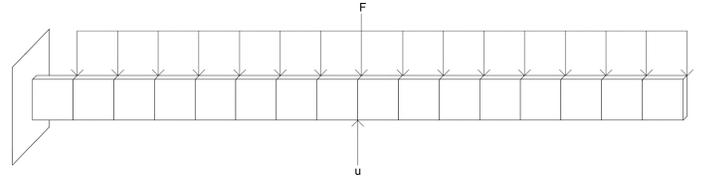
$$A_{ij} = \begin{bmatrix} \frac{1}{C_{x_i}} \left( \frac{-3C_{E_j^i}}{\Delta T} + \frac{5C_{B_j^i}}{\Delta T^2} \right) & \frac{1}{C_{x_i}} \left( \frac{3C_{E_j^i}}{2\Delta T} + \frac{-4C_{B_j^i}}{\Delta T^2} \right) & \frac{1}{C_{x_n}} \left( \frac{C_{E_j^i}}{3\Delta T} + \frac{C_{B_j^i}}{\Delta T^2} \right) \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (50)$$

where  $C_{x_i}$  is the state of interest  $i$ 's constant,  $C_{E_j^i}$  is the relation of the previous velocity components of the  $j^{th}$  states to the  $i^{th}$  states, and  $C_{B_j^i}$  is the relation of the previous acceleration components of the  $j^{th}$  states to the  $i^{th}$  states.

Using the same assumption presented earlier where  $\alpha$  is the “weak” coupling of the  $n^{th}$  state to the other non-explicitly established states. In this case due to the fact that the neighboring states contributions are encapsulated in the off diagonal components the assumption of “weak” coupling is a more attainable assumption over a larger range of time steps.

**TABLE 2.** Parameters for controlled finite element beam model

Parameter	Value
Number of Elements	16
Density	$15.43e^{-3} \frac{kg}{m^3}$
Modulus of Elasticity	11.81MPa
Viscosity	1μPas
Cross Section Area	7589.8mm <sup>2</sup>
Area Moment of Inertia	583000mm <sup>4</sup>
Element Length	36.21mm
Time Step	.25ms

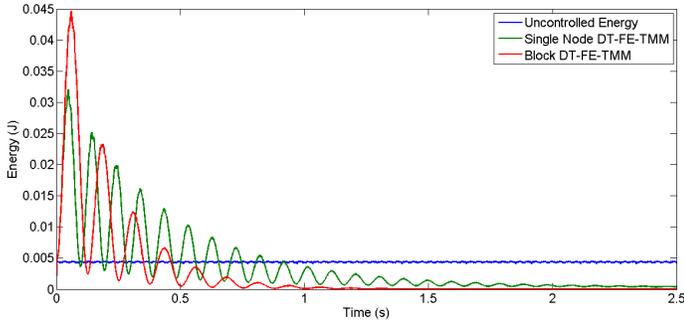


**FIGURE 3.** The configuration of the decentralized control beam, where an external impulse is applied to the tip and the controller acts on the third node.

## 4 Results and Discussion

To validate the accuracy of the proposed DT-FE-TMM method, we validate its prediction against an accepted GFEM method Table 1 shows the parameters used for both simulations. Both a GFEM and a DT-FE-TMM model were created and simulated. The GFEM was solved using a Runge-Kutta 4th/5th order and the DT-FE-TMM used a Houbolt integration technique as was described above. Figure 2 shows the Fast Fourier Transform of the tip of the beam subject to an external force applied at the tip for a duration of 10μs. The results indicate that in the low frequency part of the spectra the two methods predict similar amplitudes, but also a slight frequency shift. At high frequencies, the magnitudes of the predicted spectra are significantly different, but because the overall influence of the higher frequencies is minimal, the approach was considered accurate enough for a first approximation. It is worth noting at this point in time that the comparable third order Runge-Kutta method is not numerically stable and therefore was not able to be compared to the other integration methods.

To test the control aspects of the presented work the parameters from Table 2 were used. The system was disturbed by an impulse along the beam at every node of a magnitude of one Newton and one Newton-meter. Using the methodology presented in Section 3 we developed an localized model and then used it to design a single node and block localized LQR con-

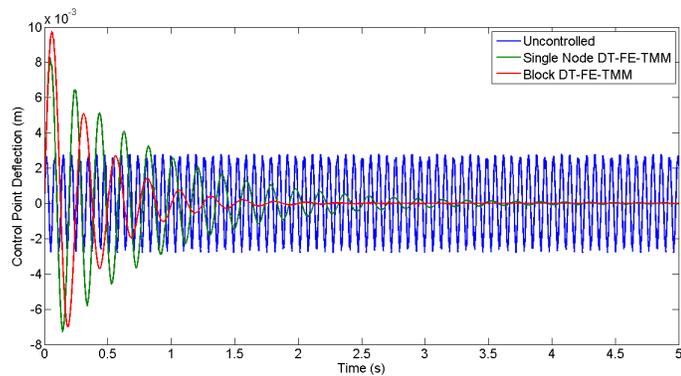


**FIGURE 4.** The total energy within the bending beam for the uncontrolled case, single node DT-FE-TMM, and block DT-FE-TMM.

**TABLE 3.** Total beam energy over the simulation

Type	Value (Js)
Free Vibrating	0.02186
Single DT-FE-TMM	0.009897
Block DT-FE-TMM	0.006389
Full State LQR	$2.4015e - 07$

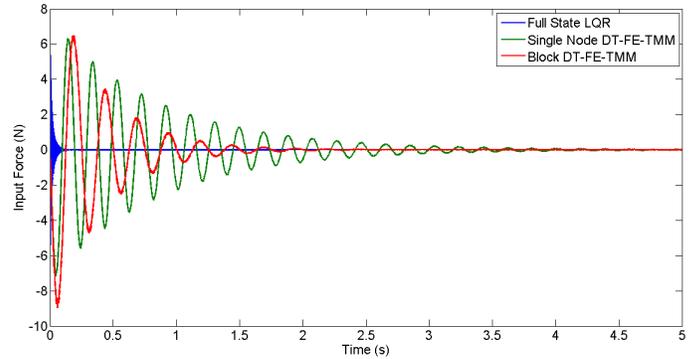
troller for the eighth node. The LQR parameters can be seen in Appendix C and were selected to have similar ranges of input forces. The configuration of the simulation can be seen in Figure 3. Figure 4 show the total energy of the simulated beam for the



**FIGURE 5.** The displacement for the eighth node for uncontrolled case, single node DT-FE-TMM, and block DT-FE-TMM.

uncontrolled beam, single node DT-FE-TMM, and block DT-FE-TMM. From this we can see that both DT-FE-TMM controllers have a temporary increase in the system energy and then it regulates the system energy below that of the uncontrolled beam.

For the block controller the final crossing point below the uncontrolled beam comes at 0.4625s and for the single controller the final crossing point is 0.925s. We also ran a simulation of a full state LQR controller but it is not pictured because it is not visible compared to the other controllers and the uncontrolled beam energy. Table 3 shows the total beam energy over the five second simulation. We can see that the full state LQR far exceeds the performance of the DT-FE-TMM controllers and provides a ceiling of performance.



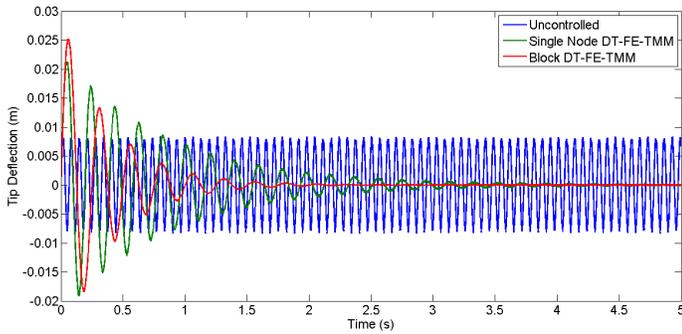
**FIGURE 6.** The force input for the eighth node for the full state LQR, single node DT-FE-TMM, and block DT-FE-TMM controllers.

Figure 5 shows the displacement of the eighth node where the control input is being applied. When combining the information viewed in Figure 5 with the control input forces shown in Figure 6 we can see that the controller is trying to counter the large swing of the control point for both controllers by applying an opposing force. This only makes the initial situation worse. We can stipulate that due to the lack of state knowledge that the local controller have they initially excite some primary frequency in response to higher order frequencies resulting in the initial over shoot but still manage to regulated the full system. Figure6 also shows that the maximum and minimum forces for all of the controllers are similar but that the full states controller is very active and operating at a higher base frequency than the DT-FE-TMM controllers.

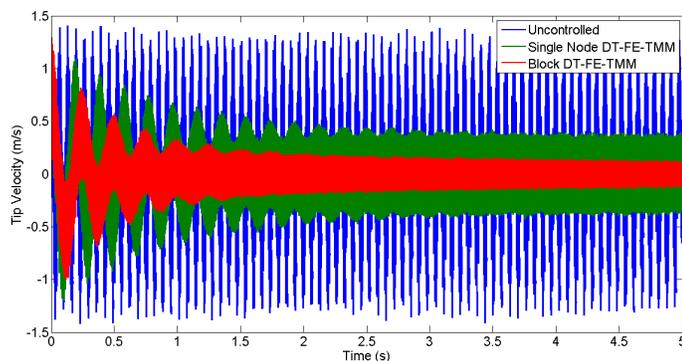
Figures 7 and 8 show the tip of the beams position and velocity respectively. We can see that the tip velocity for the DT-FE-TMM controlled simulations is always bounded by the uncontrolled. This correspondence with previous findings that DT-TMM based controllers do a particularly good job at regulating velocities of systems.

## 5 Conclusion

In conclusion we derived a model for DT-FE-TMM which showed comparable results to the continuous time GFEM simu-



**FIGURE 7.** The displacement for the tip for the uncontrolled case, single node DT-FE-TMM, and block DT-FE-TMM.



**FIGURE 8.** The velocity for the tip for the uncontrolled case, single node DT-FE-TMM, and block DT-FE-TMM.

lations methods. The proposed model was used to create a local decentralized model for use with control concepts. The operational range of the controller parameters from this model was smaller than hoped due to constraints on the selected time step. The results of the controllers that were able to be designed were promising even given this limitation. We also presented the sub-block DT-FE-TMM controller that will yield controllers that can be directly compared to other finite element decentralized control methods in future works. The advantages of the proposed methodology consist in that we only need to deal with a localized model of small number of states and need to feedback only local states, hence less sensor knowledge. Some of the additional advantages from using the discrete-time approach are easy migration to flight control software and explicit control with maximum bandwidth.

## REFERENCES

[1] Nguyen, N., and Urnes, J., 2012. “Aeroelastic modeling of elastically shaped aircraft concept via wing shaping control for drag reduction”. In *AIAA Atmospheric Flight Mechan-*

*ics Conference*, pp. 13–16.

- [2] Swei, S. S.-M., and Nguyen, N., 2014. “Aeroelastic wing shaping control subject to actuation constraints”.
- [3] Nguyen, N., 2010. “Elastically shaped future air vehicle concept”. *NASA Innovation Fund Award*.
- [4] Sofla, A., Meguid, S., Tan, K., and Yeo, W., 2010. “Shape morphing of aircraft wing: Status and challenges”. *Materials & Design*, **31**(3), pp. 1284–1292.
- [5] Majji, M., and Junkins, J. L., 2006. “Robust control of redundantly actuated dynamical systems”. PhD thesis, Texas A&M University.
- [6] Majji, M., Rediniotis, O. K., and Junkins, J. L., 2007. “Design of a morphing wing: modeling and experiments”. In *Proceedings of the AIAA Atmospheric Flight Mechanics Conference and Exhibit*, Hilton Head, South Carolina, USA, pp. 20–23.
- [7] Vos, R., Gurdal, Z., and Abdalla, M., 2010. “Mechanism for warp-controlled twist of a morphing wing”. *Journal of Aircraft*, **47**(2), pp. 450–457.
- [8] Cheung, K. C., Calisch, S. E., and Gershenfeld, N. A., 2014. Flexural digital material construction and transduction, Mar. 7. US Patent App. 14/201,781.
- [9] Xiong, J., Ma, L., Wu, L., Wang, B., and Vaziri, A., 2010. “Fabrication and crushing behavior of low density carbon fiber composite pyramidal truss structures”. *Composite Structures*, **92**(11), pp. 2695–2702.
- [10] Vasiliev, V., and Razin, A., 2006. “Anisogrid composite lattice structures for spacecraft and aircraft applications”. *Composite Structures*, **76**(1), pp. 182–189.
- [11] Cheung, K. C., and Gershenfeld, N., 2013. “Reversibly assembled cellular composite materials”. *science*, **341**(6151), pp. 1219–1221.
- [12] T.M. Tan, A. Yousuff, L. B., and Konstantinidis, M., 1990. “A modified finite element-transfer matrix for control design of space structures”. *Computers and Structures*, **36**(1), pp. 47 – 55.
- [13] Kumar, A. S., and Sankar, T., 1986. “A new transfer matrix method for response analysis of large dynamic systems”. *Computers & structures*, **23**(4), pp. 545–552.
- [14] Cramer, N., Swei, S. S.-M., Cheung, K., and Teodorescu, M. “Application of transfer matrix approach to modeling and decentralized control of lattice-based structures”.
- [15] Krauss, R., and Okasha, M., 2013. “Discrete-time transfer matrix modeling of flexible robots under feedback control”. In *American Control Conference (ACC), 2013, IEEE*, pp. 4104–4109.
- [16] Krauss, R. W., 2011. “Computationally efficient modeling of flexible robots using the transfer matrix method”. *Journal of Vibration and Control*, p. 1077546311408466.
- [17] Rong, B., Rui, X., Wang, G., and Yang, F., 2010. “Discrete time transfer matrix method for dynamics of multi-body system with real-time control”. *Journal of Sound and*

*Vibration*, **329**(6), pp. 627–643.

- [18] Rui, X.-T., Kreuzer, E., Rong, B., and He, B., 2012. “Discrete time transfer matrix method for dynamics of multi-body system with flexible beams moving in space”. *Acta Mechanica Sinica*, **28**(2), pp. 490–504.
- [19] Hendy, H., Rui, X., Zhou, Q., and Khalil, M., 2014. “Controller parameters tuning based on transfer matrix method for multibody systems”. *Advances in Mechanical Engineering*, **2014**.
- [20] Siljak, D. D., 1978. *Large-scale dynamic systems: stability and structure*, Vol. 310. North-Holland New York.
- [21] Park, K., Kim, N.-I., and Reich, G. W., 2000. “Theory of localized vibration control via partitioned lqr synthesis”. In SPIE’s 7th Annual International Symposium on Smart Structures and Materials, International Society for Optics and Photonics, pp. 520–531.
- [22] Lim, K. B., 1996. “Robust control design framework for substructure models”. *Journal of guidance, control, and dynamics*, **19**(1), pp. 181–190.
- [23] Sunar, M., and Rao, S., 1997. “A substructures method for the active control design of large periodic structures”. *Computers & structures*, **65**(5), pp. 695–701.
- [24] Ikeda, M., Šiljak, D., and White, D., 1981. “Decentralized control with overlapping information sets”. *Journal of optimization theory and Applications*, **34**(2), pp. 279–310.

$$C_{n+1,n+1} = \frac{\mu I}{L^3} \begin{bmatrix} 12 & -6 \\ -6 & 4 \end{bmatrix} \quad (56)$$

$$C_{n-1,n} = C_{n,n+1} = \frac{\mu I}{L^3} \begin{bmatrix} -12 & 6 \\ -6 & 2 \end{bmatrix} \quad (57)$$

$$C_{n,n-1} = C_{n+1,n} = \frac{\mu I}{L^3} \begin{bmatrix} -12 & -6 \\ 6 & 4 \end{bmatrix} \quad (58)$$

$$M_{n-1,n-1} = \frac{\rho CA}{420 * L} \begin{bmatrix} 156 & 22 \\ 22 & 4 \end{bmatrix} \quad (59)$$

$$M_{n+1,n+1} = \frac{\rho CA}{420 * L} \begin{bmatrix} 156 & 22 \\ 22 & 4 \end{bmatrix} \quad (60)$$

$$M_{n-1,n} = M_{n,n+1} = \frac{\rho CA}{420 * L} \begin{bmatrix} 54 & -13 \\ 13 & -3 \end{bmatrix} \quad (61)$$

$$M_{n,n-1} = M_{n+1,n} = \frac{\rho CA}{420 * L} \begin{bmatrix} 54 & -22 \\ -22 & 4 \end{bmatrix} \quad (62)$$

$$M_{n,n} = M_{n-1,n-1} + M_{n+1,n+1} \quad (63)$$

## Appendix A: Galerkin Finite Element Matrices

$$K_{n-1,n-1} = \frac{EI}{L^3} \begin{bmatrix} 12 & 6 \\ 6 & 4 \end{bmatrix} \quad (51)$$

$$K_{n+1,n+1} = \frac{EI}{L^3} \begin{bmatrix} 12 & -6 \\ -6 & 4 \end{bmatrix} \quad (52)$$

$$K_{n-1,n} = K_{n,n+1} = \frac{EI}{L^3} \begin{bmatrix} -12 & 6 \\ -6 & 2 \end{bmatrix} \quad (53)$$

$$K_{n,n-1} = K_{n+1,n} = \frac{EI}{L^3} \begin{bmatrix} -12 & -6 \\ 6 & 4 \end{bmatrix} \quad (54)$$

$$C_{n-1,n-1} = \frac{\mu I}{L^3} \begin{bmatrix} 12 & 6 \\ 6 & 4 \end{bmatrix} \quad (55)$$

## Appendix B: Constants from Decentralized Control Construction

$$C_{y_n^R} = -(T_{21}T_{11}^{-1}F_{12}^{n+1} + F_{22}^{n+1})^{-1}(-T_{21}T_{11}^{-1}F_{11}^{n+1} + F_{21}^{n+1}) + (Q_{22}Q_{12}^{-1}H_{12}^n - H_{22}^n)^{-1}(H_{21}^n - Q_{22}Q_{12}^{-1}H_{11}^n) + M_{n,n}A_n \quad (64)$$

$$C_{E_n} = (T_{21}T_{11}^{-1}F_{12}^{n+1} + F_{22}^{n+1})^{-1}(F_{23c}^{n+1}C_{n,n} + C_{n+1,n} - T_{21}T_{11}^{-1}F_{13c}^{n+1}C_{n,n}) - (Q_{22}Q_{12}^{-1}H_{12}^n - H_{22}^n)^{-1}(H_{13c}^n C_{n,n} + C_{n-1,n} - Q_{22}Q_{12}^{-1}H_{13c}^n C_{n,n}) \quad (65)$$

## Appendix C: LQR Parameters

$$C_{E_{n+1}} = (T_{21}T_{11}^{-1}F_{12}^{n+1} + F_{22}^{n+1})^{-1}(F_{23c}^{n+1}C_{n,n+1} + C_{n+1,n+1} - T_{21}T_{11}^{-1}F_{13c}^{n+1}C_{n,n+1} - T_{21}T_{11}^{-1}M_{n+1,n+1}A_{n+1}H_{13c}^{n+2}C_{n+2,n+1} + H_{23c}^{n+2}C_{n+2,n+1} + C_{n+1,n+1}) \quad (66)$$

$$Q_{single} = \begin{bmatrix} 1e^5 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (73)$$

$$C_{E_{n-1}} = -(Q_{22}Q_{12}^{-1}H_{12}^n - H_{22}^n)^{-1}(H_{23c}^n C_{n,n-1} + C_{n-1,n-1} - Q_{22}Q_{12}^{-1}H_{13c}^n C_{n,n-1} - M_{n-1,n-1}A_{n-1}F_{13c}^{n-1}C_{n-2,n-1} - F_{23c}^{n-1}C_{n-2,n-1} - C_{n-1,n-1} + Q_{22}Q_{12}^{-1}F_{13c}^{n-1}C_{n-2,n-1}) \quad (67)$$

$$R_{single} = 7e^{-6}I \quad (74)$$

where  $I$  is a 2-by-2 identity matrix

$$C_{B_{n+1}} = (T_{21}T_{11}^{-1}F_{12}^{n+1} + F_{22}^{n+1})^{-1}(-T_{21}T_{11}^{-1}F_{13c}^{n+1}M_{n+1,n+1} + M_{n,n+1} + T_{21}T_{11}^{-1}H_{13c}^{n+2}M_{n+2,n+1} + M_{n+1,n+1}A_{n+1}H_{13c}^n M_{n+2,n+1} - M_{n+2,n+1} + M_{n+1,n+1}) \quad (68)$$

$$Q_{block} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 * Q_{single} & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (75)$$

where the 0's are 6-by-6 zero matrices

$$C_{B_n} = (T_{21}T_{11}^{-1}F_{12}^{n+1} + F_{22}^{n+1})^{-1}F_{23c}^{n+1}M_{n+1,n} - M_{n,n} + (Q_{22}Q_{12}^{-1}H_{12}^n - H_{22}^n)^{-1}H_{23c}^n M_{n-1,n} \quad (69)$$

$$R_{block} = R_{single} \quad (76)$$

$$Q_{FullState} = 1e^4 I \quad (77)$$

$$C_{B_{n-1}} = -(Q_{22}Q_{12}^{-1}H_{12}^n - H_{22}^n)^{-1}(-Q_{22}Q_{12}^{-1}H_{13c}^n M_{n,n-1} + M_{n,n-1} + Q_{22}Q_{12}^{-1}F_{13c}^n M_{n-2,n-1} - M_{n-1,n-1}A_{n-1}F_{13c}^{n-1}M_{n-2,n-1} - M_{n-2,n-1} - M_{n-1,n-1}) \quad (70)$$

where  $I$  is a 64-by-64 identity matrix

$$R_{FullState} = 2e^3 I \quad (78)$$

where  $I$  is a 2-by-2 identity matrix

$$C_L = -(Q_{22}Q_{12}^{-1}H_{12}^n - H_{22}^n)^{-1}(F_{21}^{n-1}\mu_{13} + F_{22}^{n-1}\mu_{23} - M_{n-1,n-1}A_{n-1}(F_{11}^{n-1}\mu_{13} + F_{12}^{n-1}\mu_{23}) + F_{23c}^{n-1}M_{n-1,n-2}B_{n-2} + (F_{23c}^{n-1}C_{n-2,n-2} + C_{n-1,n-2})E_{n-2} + Q_{22}Q_{12}^{-1}H_{13c}^n(F_{11}^{n-1}\mu_{13} + F_{12}^{n-1}\mu_{23})) \quad (71)$$

$$C_R = (T_{21}T_{11}^{-1}F_{12}^{n+1} + F_{22}^{n+1})^{-1}((H_{23c}^{n+2}C_{n+2,n+2} + C_{n+1,n+2})E_{n+2} + T_{21}T_{11}^{-1}(H_{11}^{n+2}\varepsilon_{13} + H_{12}^{n+2}\varepsilon_{23} + H_{13c}^{n+2}C_{n+2,n+2}E_{n+2}) + M_{n+1,n+1}A_{n+1}(H_{11}^{n+2}\varepsilon_{13} + H_{12}^{n+2}\varepsilon_{23} + H_{13c}^{n+2}C_{n+2,n+2}E_{n+2}) + H_{21}^{n+2}\varepsilon_{13} + H_{22}^{n+2}\varepsilon_{23} + H_{23c}^{n+2}M_{n+1,n+2}B_{n+2}) \quad (72)$$