Analyzing the Effects of Force Variations on Expanded Residual Damage Detection using Load Dependent Ritz Vectors

A Thesis

Presented to

the Faculty of the Department of Mechanical Engineering

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

In Mechanical Engineering

by

George C. Khoury

May 2014

Analyzing the Effects of Force Variations on Expanded Residual Damage Detection using Load Dependent Ritz Vectors

George C. Khoury

Approved:

Chairman of the Committee
Dr. Matthew A. Franchek, Professor,
Mechanical Engineering

Committee Members:

Dr. Karolos Grigoriadis, Professor,
Mechanical Engineering

Dr. Robert Provence, Aerospace Engineer,
NASA Johnson Space Center

Dr. Philippe Masson, Assistant Professor,
Mechanical Engineering

Dr. Suresh K. Khator, Associate Dean,
Cullen College of Engineering

Dr. Pradeep Sharma, Professor and Chair,
Mechanical Engineering

Acknowledgements

I would like to thank Dr. Zimmerman for his support and guidance during the beginning of Master's Degree. Dr. Zimmerman inspired me with the passion he displayed for his research and the betterment of his students. I hope that this study serves some advancement to the work that he pioneered. I would next like to thank Dr. Franchek for taking me under his wing after Dr. Zimmerman's untimely passing. Dr. Franchek accepted to learn my research from scratch and was integral in improving my understanding of the concepts and discovering the modifications that helped improve the methods at work. Finally, I would like to thank my family and friends for their moral support and encouragement as I pursued my degree.

Analyzing the Effects of Force Variations on Expanded Residual Damage Detection using Load Dependent Ritz Vectors

An Abstract

of a

Thesis

Presented to

the Faculty of the Department of Mechanical Engineering

University of Houston

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

In Mechanical Engineering

by

George C. Khoury

May 2014

Abstract

Many non-destructive damage detection problems are presented with the challenge where limited sensor sets allow the measurement of fewer degrees of freedom than the analytical model. This study utilized damage residual expansion methods to determine the overall structural damage locations with reduced sensor sets. The new methods presented were the Modified Method of Expanded Dynamic Residuals and the Dot Product Damage Residual Expansion Method. Ritz vectors were also used with various sensor placement techniques to modify the sensor sets and focus the damage detection capabilities towards elements of interest. The results showed that the modified damage residual expansion methods were able to locate the damaged elements, while also simplifying the selection process, by employing the dot product method and the elemental disassembly. Additionally, the sensor set variations effectively focused the damage detection results towards elements of interest by modifying the Ritz vector loadings.

Table of Contents

List of Figures

List of Tables

1 Introduction

Structural damage detection is a valuable tool for studying the integrity of structures that undergo regular fatigue such as buildings, bridges, vehicles, and aircrafts. Non-destructive damage detection methods have received attention for their simplicity and autonomous monitoring capabilities. Detecting damage would allow repairs or replacements to be made when required rather than relying on time-based maintenance schedules.

Most damage detection methods approach structural damage detection as a finite element model (FEM) updating problem, by either updating a structural stiffness matrix or by physical parameter updating. Without *a priori* knowledge of the damage location, physical parameter updates require manual interaction and reliance on engineering experience. Damage detection of large scale structures utilizing techniques that examine changes in the dynamic properties or response of structures have been areas of extensive research and reviews [1]-[3].

Frequency based methods rely on the fact that changes in the structural properties cause variations in the vibration frequencies, this observation became the motivation for using modal methods for damage identification and health monitoring. The earliest known journal article to propose damage detection via vibration measurements was by Lifshitz and Rotem, where the change in dynamic moduli was examined for damage in particle filled elastomers [4]. The limitation of frequency based approaches is the fact that natural frequencies are not very sensitive to damage, requiring either very precise measurements or large levels of damage, which was a point demonstrated by Farrar et al. through tests conducted on the I-40 bridge [5].

Measuring mode shape changes is another widely accepted approach. One of the earliest and most influential studies utilizing mode shapes to locate structural damage without prior finite element analysis used the modal assurance criteria (MAC) to correlate modes from an undamaged Orbiter space shuttle body flap with the modes after it had been exposed to acoustic loading [6]. The study was also important since it was likely the first to locate damage without the use of a prior FEM. The change in the MAC across the modal partitions was used to localize the structural damage.

A number of other techniques have been developed to locate structural damage as discussed in various literature reviews [1]-[3]. Mode shape curvature methods involve using the derivatives of mode shapes. The dynamically measured flexibility matrix, defined as the inverse of the stiffness matrix, has been used to estimate changes in the static behavior of structures. Nonlinear methods attempt to account for damage that can cause nonlinearities in beam structures, such as cracks, which may not be detected using more traditional methods.

Another suggested damage identification approach involves the modification of structural model matrices such as mass, stiffness, and damping. In these methods, updated matrices are compared to the original matrices in order to quantify the location and extent of damage. The updated matrices are solved by forming a constrained optimization problem based on the structural equations of motion, the nominal model, and the measured data. Optimal matrix update methods use closed form solutions to compute the damaged matrices. One suggested approach to the optimal matrix update problem involved the minimization of the Forbenius norm of the parameter matrix

perturbations [7]-[11]. These approaches were initially developed for model correlation where small changes are desired through a broad range of structural members.

Using the observation that damage tends to be of large extent and localized in a few structural members, an approach based on the minimization of the rank of the perturbation matrix was established. The basic Minimum Rank Perturbation Theory (MRPT) algorithm was developed by Kaouk & Zimmerman [12] [13], and has been extensively improved and published by the authors. The update to each property matrix is of minimum rank, and is equal to the number of basis vectors that the modified model is to match. The minimum rank constraint is consistent with the matrix changes in a FEM needed to represent many situations of structural damage. The advantage of the MRPT is that it is a computationally attractive algorithm that decouples the damage location and extent problems. Although the MRPT was originally developed using mode shapes, Cao and Zimmerman extended the theory for use with Ritz vectors [14].

Load-dependent Ritz vectors were proposed by Wilson [15] as an alternative set of basis vectors to describe the dynamic response of a structure in place of mode shapes. Ritz vectors have since been used in areas of transient response prediction [16], model reduction [17], and component mode synthesis [18]. Sohn and Law used Ritz vectors for damage detection in a bridge structure [19], and also applied Ritz vectors to a Bayesian probabilistic damage detection method [20]. The first Ritz vector is the static deformation of the structure due to a particular applied load, and the remaining orthogonal Ritz vectors are calculated using inverse iteration and the Gram-Schmidt orthogonalization.

Ritz vectors have a number of advantages over traditional mode shapes, (1) Ritz vectors automatically include the static correction term. (2) Ritz vectors are

computationally less expensive than eigenvectors. (3) All of the Ritz vectors generated by a load will be excited by that load. (4) Fewer Ritz vectors than mode shapes are typically required to achieve the same level of accuracy in transient response and model reduction. (5) Ritz vectors may be less sensitive to noise since the Ritz vector extraction process depends on the solution of linear algebraic equations as opposed to an eigensolution for mode shapes. Ritz vectors were also found to have better damage diagnosis than measured mode shapes by Sohn and Law [19].

The extraction of Ritz vectors from dynamic testing data by Cao and Zimmerman [21] was an important step in the development of Ritz vector application. The extraction algorithm requires the identification of a state-space system realization using the Eigensystem Realization Algorithm (ERA) developed by Juang and Pappa [22]. Sohn and Law later presented an alternative method of extraction using a complete flexibility matrix constructed from measured vibration test data [23]. Boxoen and Zimmerman [24] expanded on [21] by introducing a method to extract test data from damped structures. Taylor and Zimmerman [25] also presented an improved Ritz vector extraction procedure that was able to extract the non-orthogonalized vectors required for some damage detection methods, such as MRPT.

Most damage detection problems, including the MRPT, assume a full set of measurements to accurately locate damage in a system. A common issue that affects all damage detection methods when a full measurement set is unavailable is the incomplete measurement problem, which consists of measuring: (i) fewer basis vectors than that of the analytical model, and (ii) fewer degrees-of-freedom (DOFs) than that of the analytical model. The first part is due to the inability to excite the structure above a certain

frequency and the corruption of global modes by local modes of vibration, while the second part occurs due to the finite number of sensors that can be used.

The approaches used to address the second part of the problem are either to reduce the analytical model to the test DOFs or to expand the measured modal data to all DOFs included in the analytical model. Both techniques were evaluated by Zimmerman et al. in an attempt to enhance structural health monitoring capability [26]. A problem observed with model reduction is that it reduces accuracy when locating damage, causing the localized changes in the full model to appear "smeared" throughout the reduced model. On the other hand, mode shape expansion will lead to false positive damage indications due to errors in the expansion process.

The Method of Expanded Dynamic Residuals (MEDR) was devised by Zimmerman et al. to circumvent the second part of the incomplete measurement problem by using structural connectivity information to determine full model damage [27]. MEDR extends the theory of MRPT damage residuals with the concept of stiffness matrix disassembly [28] to arrive at an expanded damage residual vector using a limited number of measurement points. The expanded damage residual is used to determine the structural location of damage, which significantly simplifies the task of determining the damage extent. Taylor and Zimmerman extended MEDR to include the application of Ritz vectors and concluded that Ritz vectors find more consistently correct results compared to mode shapes [29].

When faced with the second part of the incomplete measurement problem in a testing environment, a limited number of sensors and actuators must be placed in a configuration to obtain the most important dynamic information. Since sensor and

actuator locations strongly influence structural testing, it is important that the selected locations be optimized. Pre-test planning methods typically utilize information obtained from a baseline FEM to determine optimal sensor placement. Comparative studies on a number of the methods have been performed to evaluate the effectiveness of each technique [30]-[32]. Yap and Zimmerman put sensor placement methods into two categories, first the more commonly used methods that rely on sensor placement metrics (SPMs), and second the more elaborate methods referred to as sensor elimination algorithms (SEAs) [32].

Two SPMs of interest include Eigenvector Product (EVP) and Kinetic Energy (KE) Product presented by Larson et al. [31]. The EVP uses modal products from the reduced FEM eigenvectors to identify potential sensor and actuator locations. The eigenvector components over the desired mode range are multiplied and candidate locations are chosen based on the idea that maximum values contain high displacement while zero values are excluded as they represent node points DOFs. The KE product utilizes the modal kinetic energy that is obtained from the FEM mass matrix and target modes to place sensors at points of maximum kinetic energy. Nodal points are also precluded from this method, as there will be no kinetic energy at those locations.

Another commonly used method is the QR decomposition, which generates the most linearly independent set of sensor DOFs using the same number of sensors as the number of modes [33] [34]. The limitation with QR decomposition is that the number of sensors placed cannot be greater than the number of target basis vectors, which is why it was not considered in this study. In many cases, QR decomposition was used in

conjunction with SPMs to compromise between observability and linear independence [32].

SEAs are defined as a class of more elaborate sensor placement techniques that reduce the sensors from an initial set of candidate DOFs to the final sensor set within a number of iterations. One such technique is known as Effective Independence (EI), which Kammer presented as a method used to select sensor locations from a large candidate set [30]. The EI method attempts to maximize the trace and determinant of the Fisher information matrix. The reduced sensor set is obtained by removing the sensors that contribute the least to the linear independence of the target modes.

Khoury and Zimmerman extended the EVP, KE, and EI methods for use with Ritz vectors [35]. The study showed that the methods could be used to locate sensors for Ritz vectors. The key difference is that sensor sets vary when the loading on the structure is changed due to the modified Ritz vectors. A number of sensor sets were also found to locate both mode shapes and Ritz vectors simultaneously with reasonable accuracy. Since the sensor sets change with different load dependent Ritz vectors, the force direction and location could be used to optimize the sensor set that would best monitor specific locations of a structure.

The present research examined methods that would mitigate the 2nd part of the incomplete measurement problem. Two dynamic residual expansion algorithms were developed as modifications to the MEDR. The first is the Modified Method of Expanded Dynamic Residuals (MMEDR) where the matrix disassembly was replaced by an elemental disassembly in an attempt to combine and reduce the number of columns required during the residual selection process. The second method is the Dot Product

Damage Residual Expansion Method (DPDREM) which again uses the elemental disassembly method and simplifies the selection process by replacing the MEDR minimization problem with a maximization problem that utilizes the dot product method.

Utilizing the fact that varying actuator directions and locations can potentially modify the Ritz vectors identified and in turn vary the sensor sets determined using EVP, KE, and EI. The MRPT was used to determine the reduced dynamic residual for the system with the sensor sets defined. The different sensor sets were also used with the two dynamic residual expansion techniques identified above to locate the full system damage and also focus the damage detection to selected elements of interest.

This thesis contains three additional chapters. Chapter 2 introduces the theoretical background and presents the sensor placement and damage detection methods applied. Chapter 3 contains the analytical examples based on the concepts from Chapter 2. Chapter 4 summarizes the results and presents the recommendations for future work.

2 Theoretical Background

2.1 Modal Analysis

2.1.1 Analytical Modal Analysis

The dynamic equilibrium equation for a discrete *n*-degree-of-freedom (DOF) structure can be expressed as

$$\mathbf{M}\,\underline{\ddot{x}}(t)+\mathbf{C}\,\underline{\dot{x}}(t)+\mathbf{K}\,\underline{x}(t)=\mathbf{B}\,\underline{u}(t)\,, \tag{1}$$

where **M**, **C**, and **K** are the $(n\times n)$ mass, damping and stiffness matrices, respectively, $\underline{x}(t)$ is the $(n\times 1)$ position vector, **B** is the $(n\times m)$ force influence matrix, and $\underline{u}(t)$ is the $(m\times 1)$ force input vector. The over-dots represent differentiation with respect to time.

The standard solution to Equation (1) is found by assuming that

$$\underline{x}(t)=\underline{\Psi}e^{\lambda t}\,, \tag{2}$$

where $\underline{\Psi}$ is a $(n\times 1)$ complex vector and λ is a complex constant for a general damped system. Substituting Equation (2) into (1) yields the eigenvalue problem given by

$$\left(\lambda^2\mathbf{M}+\lambda\mathbf{C}+\mathbf{K}\right)\underline{\Psi}e^{\lambda t}=\underline{0}. \tag{3}$$

Equation (3) can be written in state-space form as

$$\begin{bmatrix}\mathbf{0} & \mathbf{I}\\ -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{C}\end{bmatrix}\begin{bmatrix}\underline{\Psi}\\ \lambda\underline{\Psi}\end{bmatrix}=\lambda\begin{bmatrix}\underline{\Psi}\\ \lambda\underline{\Psi}\end{bmatrix}, \tag{4}$$

where **I** is the $(n\times n)$ identity matrix. Equation (4) allows the eigenvalue problem to be solved for 2*n* eigenvalues and eigenvectors, consisting of *n* complex conjugate pairs. The eigenvectors can be solved as

$$\lambda_i=\omega_i\left(-\zeta_i\pm j\sqrt{1-\zeta_i^2}\right), \tag{5}$$

where ω_i is the natural frequency, ζ_i is the damping ratio of the i^{th} mode shape, and $j = \sqrt{-1}$. The complex conjugate eigenvectors can be written as

$$\underline{\Psi}_i = \underline{\Psi}_{iR} + j\,\underline{\Psi}_{iI}, \tag{6}$$

where $\underline{\Psi}_{iR}$ and $\underline{\Psi}_{iI}$ are the real and complex parts of the i^{th} mode shape, respectively.

2.1.2 Modal Assurance Criteria

The Modal Assurance Criteria (MAC) is a common tool used to measure the co-linearity between two vectors of equal length. The MAC was originally developed to measure the correlation of mode shapes, but can provide a measure of the least square deviation of any two vectors. The MAC is computed as

$$\mathrm{MAC} = \frac{\left(\underline{\Psi}_a^T \underline{\Psi}_b\right)^2}{\left(\underline{\Psi}_a^T \underline{\Psi}_a\right)\left(\underline{\Psi}_b^T \underline{\Psi}_b\right)}, \tag{7}$$

where $\underline{\Psi}_a$ and $\underline{\Psi}_b$ are the two vectors to be compared. Normally, the MAC is computed for all combinations of vectors between the two sets of basis vectors, resulting in a matrix of MAC values. For a good correlation, the MAC matrix values will be near unity along the diagonal, and near zero off the diagonal. An auto-MAC is a matrix of MAC values computed by comparing a vector set to itself, and is a useful tool when assessing the quality of experimental mode shapes.

2.1.3 Ritz Vectors

2.1.4 Analytical Ritz Vectors

The dynamic equilibrium equation for a discrete *n*-degree-of-freedom (DOF) structure can be expressed as

$$\mathbf{M}\,\ddot{\underline{x}}(t) + \mathbf{C}\,\dot{\underline{x}}(t) + \mathbf{K}\,\underline{x}(t) = \underline{B}\,u(t), \tag{8}$$

where **M**, **C**, and **K** are the $(n \times n)$ mass, damping and stiffness matrices, respectively, $\underline{x}(t)$ is the $(n \times 1)$ position vector, $\underline{B}$ is the $(n \times 1)$ force influence vector, and $u(t)$ is the scalar force input signal. The over-dots represent differentiation with respect to time.

The first Ritz vector, representing the deflection of the structure under a unit static load [15], is found from the solution of

$$\mathbf{K}\, \underline{\hat{v}}_1 = \underline{B}, \tag{9}$$

where $\underline{\hat{v}}_1$ is the non-normalized Ritz vector. Although Wilson [15] mass-normalized the Ritz vectors, it is more appropriate to unit-normalize the Ritz vectors since the mass matrix is unknown in experimental identifications [21] [24]. The first Ritz vector is unit-normalized as

$$\underline{v}_1 = \frac{\underline{\hat{v}}_1}{\sqrt{\left(\underline{\hat{v}}_1\right)^T \underline{\hat{v}}_1}}. \tag{10}$$

The subsequent Ritz vectors are found from the solution of

$$\mathbf{K}\, \underline{\hat{v}}_i = \mathbf{M}\, \underline{v}_{i-1}. \tag{11}$$

Each Ritz vector is orthogonalized using a Gram-Schmidt orthogonalization process

$$\underline{v}_i^* = \underline{\hat{v}}_i - \sum_{j=1}^{i-1} c_j \underline{v}_j, \tag{12}$$

where

$$c_j = \underline{v}_j^T \underline{\hat{v}}_i. \tag{13}$$

After each Ritz vector is orthogonalized, it is unit-normalized

$$\underline{v}_i = \frac{\underline{v}_i^*}{\sqrt{\left(\underline{v}_i^*\right)^T \left(\underline{v}_i^*\right)}}. \tag{14}$$

2.2 Incomplete Measurement Problem

The incomplete measurement problem is a well-known issue in damage detection problems. One aspect of the incomplete measurement problem occurs when there are fewer measured degrees-of-freedom (DOF) than that of the analytical model. Several methods have been proposed to correlate a model with an experimental test. The finite element model (FEM) may be reduced to the size of the test model, or the test model may be expanded to the size of the finite element model. Guyan reduction [36] is a common method used to reduce the FEM size, and is presented in Section 2.3.1. Three sensor placement techniques for the reduced systems are presented in the sections that follow; the Vector Product in Section 2.3.2, the Kinetic Energy Product in Section 2.3.3, and Effective Independence in Section 2.3.4.

2.2.1 Guyan Reduction

Given $(n \times n)$ mass and stiffness matrices, **M** and **K** respectively, a transformation matrix relating the active DOFs to the omitted DOFs can be used to find reduced mass and stiffness matrices as

$$\mathbf{K}_r = \mathbf{T}^T\mathbf{K}\mathbf{T} \quad \mathbf{M}_r = \mathbf{T}^T\mathbf{M}\mathbf{T}. \tag{15}$$

The matrix **T** is the transformation matrix that satisfies

$$\begin{bmatrix} \underline{x}_a \\ \underline{x}_o \end{bmatrix} = \begin{bmatrix} \mathbf{I} \\ \mathbf{G} \end{bmatrix} \underline{x}_a = \mathbf{T}\underline{x}_a, \tag{16}$$

where $\underline{x}_a$ and $\underline{x}_o$ refer to the active and omitted DOFs respectively. The stiffness matrix can be partitioned into the active and omitted DOFs so that the static solution can be written as

$$\begin{bmatrix} \mathbf{K}_{aa} & \mathbf{K}_{ao} \\ \mathbf{K}_{oa} & \mathbf{K}_{oo} \end{bmatrix} \begin{bmatrix} \underline{x}_a \\ \underline{x}_o \end{bmatrix} = \begin{bmatrix} \underline{0} \\ \underline{0} \end{bmatrix}. \tag{17}$$

Multiplying out the second row of Equation (17) gives

$$\mathbf{K}_{oa}\underline{x}_a + \mathbf{K}_{oo}\underline{x}_o = \underline{0}, \tag{18}$$

so that the omitted DOFs can be solved as

$$\underline{x}_o = -\mathbf{K}_{oo}^{-1}\mathbf{K}_{oa}\underline{x}_a. \tag{19}$$

The transformation matrix can be found by inserting the result from Equation (19) into (16) such that

$$\mathbf{T} = \begin{bmatrix} \mathbf{I} \\ -\mathbf{K}_{oo}^{-1}\mathbf{K}_{oa} \end{bmatrix}. \tag{20}$$

The transformation matrix is next rearranged such that the active and omitted DOFs are combined and arranged consecutively from 1 to *n*.

2.2.2 Basis Vector Product

This technique is generally referred to as the Eigenvector Product (EVP) technique [31]. Since EVP can be applied with basis spaces other than eigenvectors, a more general representation, Basis Vector Product (BVP), can be used. The basis vectors found from the finite FEM can be used to identify possible sensor or actuator locations. The shapes of interest are chosen as shown below

$$\Phi = \begin{bmatrix} \varphi_{11} & \varphi_{12} & & \varphi_{1m} \\ \vdots & \vdots & \cdots & \vdots \\ \varphi_{n1} & \varphi_{n2} & & \varphi_{nm} \end{bmatrix}. \tag{21}$$

Φ is the $(n \times m)$ set of mode shapes, Ritz vectors or a combination of both, where *n* is the number of DOFs and *m* is the number of basis vectors. The BVP is calculated by multiplying the Φ components, such that

$$BVP_i = \varphi_{i1}\varphi_{i2}\cdots\varphi_{im}, \tag{22}$$

where subscript i corresponds to the DOF of interest. The total set of vector products, BVP, is sorted so that the maximum values are found. A high value of BVP_i corresponds to nodes with high displacement and thus a candidate location. Vector product will exclude node point DOFs of any vector as BVP_i will give a value of zero.

2.2.3 Kinetic Energy Product

The KE product is used for sensor placement identification using the assumption that maximum observability will occur by placing sensors at locations with maximum kinetic energy [30] [31]. Using the FEM basis vectors, the kinetic energy is found as follows

$$ke_{im} = \varphi_{im}\sum_{j=1}^{n}\mathbf{M}_{ij}\varphi_{jm}, \tag{23}$$

where i corresponds to the DOF of interest, m is the target basis vector, and n corresponds to the total number of DOFs in the system. $\mathbf{M}_{ij}$, are the corresponding mass matrix components. The total kinetic energy matrix is as follows

$$\mathbf{ke} = \begin{bmatrix} ke_{11} & ke_{12} & \cdots & ke_{1m} \\ \vdots & \vdots & \cdots & \vdots \\ ke_{n1} & ke_{n2} & & ke_{nm} \end{bmatrix}. \tag{24}$$

Finally, the KE product is found by multiplying the **ke** components

$$KE_i = ke_{i1}ke_{i2}\cdots ke_{im}, \tag{25}$$

so that the total KE product, KE, can be sorted from highest to lowest. As with BVP, a high value of KE_i corresponds to a candidate location. The KE product will also negate node points associated with a single shape since the product will be zero at that DOF. The mass weighting inherent to the KE product causes the sensor and actuator placement to

be dependent on the finite element discretization, such that there is a bias against areas with fine mesh sizes and thus small masses.

2.2.4 Effective Independence

Effective independence was developed as a technique to select sensor locations for large space structures [30]. The candidate sensor sets are ranked according to their contribution to the linear independence of the target basis vectors. The first step is to find the Fisher information matrix

$$\mathbf{A}_o = \sum_{i=1}^{n} \varphi^{i\,T} \varphi^{i}, \tag{26}$$

where φ^{i} is the i^{th} row of the target basis vector and n is the total number candidate sensor locations in the system. Equation (26) demonstrates that information can be added to or subtracted from the Fisher information matrix with the inclusion or exclusion of DOFs. The number of DOFs in the sensor set can be reduced by eliminating locations that do not contribute to the independence of the target basis vectors.

The analysis starts by solving the eigenvalue equation for $\mathbf{A}_o$.

$$\left[\mathbf{A}_o - \Lambda_j \mathbf{I}\right] \underline{\Psi}_j = 0, \tag{27}$$

for $j = 1, 2, \ldots, m$, where m is the total number of basis vectors in the system, and $\underline{\Psi}_j$ are orthonormal vectors resulting in the relations

$$\underline{\Psi}_j^{T} \mathbf{A}_o \underline{\Psi}_j = \Lambda_j \text{ and } \underline{\Psi}_j^{T} \underline{\Psi}_j = 1. \tag{28}$$

The EI coefficients of the candidate sensors are computed as

$$\mathbf{G} = \left[\Phi\Psi\right] \otimes \left[\Phi\Psi\right], \tag{29}$$

where the symbol $\otimes$ represents term-by-term matrix multiplication, and $\mathbf{\Psi} = \left[\underline{\Psi}_1, \underline{\Psi}_2, \ldots, \underline{\Psi}_m\right]$. Next, Equation (29) is post-multiplied by the inverse of the matrix of eigenvalues,

$$\mathbf{F_E} = \mathbf{G}\mathbf{\Lambda}^{-1}, \tag{30}$$

where $\mathbf{F_E}$ represents the fractional eigenvalue distribution, and $\mathbf{\Lambda} = \text{diag}\left(\left[\Lambda_1, \Lambda_2, \ldots, \Lambda_m\right]\right)$. Finally, the terms within each row of $\mathbf{F_E}$ are added to obtain

$$\underline{\text{EI}} = \begin{bmatrix} \sum \mathbf{F}_{\mathbf{E}1} \\ \sum \mathbf{F}_{\mathbf{E}2} \\ \vdots \\ \sum \mathbf{F}_{\mathbf{E}n} \end{bmatrix}. \tag{31}$$

Alternatively, EI can be found using the following formulation

$$\underline{\text{EI}} = \text{diag}\left(\Phi\left[\Phi^{\text{T}}\Phi\right]^{-1}\Phi^{\text{T}}\right). \tag{32}$$

The values of EI will range from 0 to 1, with a value of 0 indicating that the DOF does not contribute to the observability of the system. The smallest value of the EI vector is removed and the above process is repeated iteratively until the desired number of sensors is found.

Li et al. developed a more computationally efficient method to compute the EI utilizing QR decomposition [37]. The method was shown to obtain the same results with fewer computational flops.

2.3 Damage Residual Matrices

When elements of a structure are affected by damage, the damage residual matrix can be found using the basis vectors for the damaged structure and the original finite element model matrices. For a full DOF set of basis vectors, the damage residual matrix

will show the exact damaged DOF locations. If the damage basis vectors contain fewer DOFs than the model, then the damage residual matrix becomes 'smeared' such that the damaged DOFs cannot be distinguished.

2.3.1 Modal Dynamic Residual

The modal dynamic residual is found using the eigenvalue problem presented in Section 2.1.1 and has been used to determine the extent of damage on structures. The eigenvalue problem can be written without damping as

$$\left(\mathbf{K}+\lambda^2\mathbf{M}\right)\underline{\Psi}=\underline{0}. \tag{33}$$

When damage is present in the system, the measured mode shapes will deviate and the relationship will need to be modified such that

$$\left(\left(\mathbf{K}-\Delta\mathbf{K}\right)+\lambda_i^2\mathbf{M}\right)\underline{\Psi}_i=\underline{0}, \tag{34}$$

where $\underline{\Psi}_i$ denotes each measured mode shape and $\Delta\mathbf{K}$ represents the effect of damage on the stiffness matrix **K**. By rearranging Equation (34), the i^{th} modal dynamic residual vector, $\underline{d}_{m,i}$, is defined as

$$\underline{d}_{m,i}=\left(\mathbf{K}+\lambda_i^2\mathbf{M}\right)\underline{\Psi}_i=\Delta\mathbf{K}\underline{\Psi}_i. \tag{35}$$

For the noise-free cases, the dynamic residual vector will only have nonzero terms at the DOFs affected by damage. If only *p* mode shapes are measured, Equation (35) can be formulated as a $(n\times p)$ matrix equation

$$\begin{aligned}&\mathbf{B}_{d,m}=\left(\mathbf{K\Psi}+\mathbf{MD\Psi}\right)=\Delta\mathbf{K\Psi}\\&\mathbf{D}=diag\left(\lambda_1^2,\lambda_2^2,\ldots\lambda_p^2\right)\end{aligned}, \tag{36}$$

where $\mathbf{B}_{d,m}$ is the modal dynamic residual matrix, $\mathbf{\Psi}$ is the $(n\times p)$ matrix of mode shapes, and **D** is the diagonal matrix of eigenvalues.

2.3.2 Ritz Damage Residual

Ritz vector based damage residual vectors can also be used to determine damage location [29]. Equation (11) can be rearranged as follows

$$\mathbf{K}\,\underline{\hat{v}}_i - \mathbf{M}\,\underline{v}_{i-1} = \underline{0}\,. \tag{37}$$

If the structure is damaged, the Ritz vectors will change and the relationship in Equation (37) will be modified to

$$\left(\mathbf{K} - \Delta\mathbf{K}\right)\underline{\hat{v}}_i - \mathbf{M}\,\underline{v}_{i-1} = \underline{0}\,, \tag{38}$$

where $\Delta\mathbf{K}$ represents the effect of damage on the stiffness matrix $\mathbf{K}$. The elements of the Ritz damage residual vector $\underline{d}_r$ can be defined as

$$\underline{d}_{r,i} = \mathbf{K}\underline{\hat{v}}_i - \mathbf{M}\,\underline{v}_{i-1} = \Delta\mathbf{K}\underline{\hat{v}}_i\,. \tag{39}$$

In the noise-free cases, the DOFs affected by damage appear as nonzero elements of the dynamic residual vector. If subset p Ritz vectors are measured, Equation (39) can be formulated as a $(n \times p-1)$ matrix equation

$$\begin{aligned} \mathbf{B}_{d,r} &= \left(\mathbf{K}\hat{\mathbf{V}} - \mathbf{M}\mathbf{V}\right) = \Delta\mathbf{K}\hat{\mathbf{V}} \\ \mathbf{V} &= diag\left(v_1, v_2, \ldots v_{p-1}\right) \end{aligned}\,, \tag{40}$$

where $\mathbf{B}_{d,r}$ is the Ritz damage residual matrix.

2.4 Minimum Rank Perturbation Theory

The Minimum Rank Perturbation Theory (MRPT) is a model updating technique used for model refinement and damage detection [12] [13]. The MRPT can use either the modal dynamic residual matrix or Ritz damage residual matrix to determine damage extent by solving for the stiffness perturbation matrix, $\Delta\mathbf{K}$

$$\Delta\mathbf{K} = \mathbf{B}_d\left(\mathbf{B}_d^T\mathbf{Z}\right)^{-1}\mathbf{B}_d^T\,, \tag{41}$$

where $\mathbf{Z}$ is the basis vector used. For modal analysis, $\mathbf{Z}$ will be equivalent to the $(n \times p)$ matrix of mode shapes, $\mathbf{\Psi}$. For Ritz vector analysis, $\mathbf{Z}$ will be equivalent to the $(n \times p-1)$ non-normalized Ritz vector, $\hat{\mathbf{V}}$.

The maximum rank of the stiffness perturbation matrix obtained in Equation (41) is limited to the number of mode shapes measured. If the effects of damage change the stiffness by a rank greater than the number of mode shapes measured, the stiffness perturbation matrix obtained will not be exact even when the data is noise free. It has been observed that when the number of modes are less than the true rank of the stiffness perturbation matrix, the largest elements of ΔK_d are typically identified.

If the matrix to be inverted in Equation (41) is not well conditioned, or if the number of measured basis vectors exceeds the number required to determine the extent of damage, then the subspace selection algorithm can be used to extract the maximum information from all available residual vectors. Singular value decomposition (SVD) provides a method to extract the consistent information from each column of the damage residual, which will be

$$\mathbf{B}_{\mathrm{d}} = \left[\mathbf{U}_1 \mid \mathbf{U}_2\right] \begin{bmatrix} \mathbf{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \left[\mathbf{V}_1 \mid \mathbf{V}_2\right]^{\mathrm{T}}, \tag{42}$$

where $\mathbf{U}$ and $\mathbf{V}$ are the left and right singular vectors, respectively, and $\mathbf{\Sigma}$ is a $(p \times p)$ diagonal matrix of singular values. When $\mathbf{B}_{\mathrm{d}}$ is rank deficient, its range is spanned by the p columns of $\mathbf{U}_1$. The matrix $\mathbf{U} = \left[\mathbf{U}_1 \mid \mathbf{U}_2\right]$ is a unitary matrix, so a well-conditioned problem is obtained if

$$\mathbf{B}_{\mathrm{d}} \mathbf{Y} = \mathbf{U}_1, \tag{43}$$

where matrix $\mathbf{Y}$ can be approximately calculated using the pseudoinverse of $\mathbf{B}_d$ as

$$\mathbf{Y} = \mathbf{B}_d^+ \mathbf{U}_1 = \mathbf{V}_1 \boldsymbol{\Sigma}^{-1} \mathbf{U}_1^T \mathbf{U}_1 = \mathbf{V}_1 \boldsymbol{\Sigma}^{-1}. \tag{44}$$

The extent of damage can be calculated from Equation (41) as

$$\Delta \mathbf{K} = \mathbf{B}_d \mathbf{Y} \left(\mathbf{Y}^T \mathbf{B}_d^T \mathbf{Z} \mathbf{Y} \right)^{-1} \mathbf{Y}^T \mathbf{B}_d^T. \tag{45}$$

2.5 Dynamic Residual Expansion Techniques

Dynamic residual expansion techniques extend the damage residuals found using the MRPT coupled with matrix disassembly to determine an expanded dynamic residual and determine the structural damage location. This section presents three dynamic residual expansion techniques, which are based on the concepts devised by Zimmerman et al. in [27].

2.5.1 Method of Expanded Dynamic Residuals

The Method of Expanded Dynamic Residuals (MEDR) is used to circumvent the problems associated with basis vector expansion. The damage residual vector indicates the exact damage location when using the full DOF measurements; however, when only reduced measurements may be used, the damage location may become "smeared" throughout the model. The MEDR uses the physical connectivity of the structure to create a database of "smear" patterns by projecting each of the structure's connectivity vectors onto the space of reduced measurement points. The reduced connectivity vectors are compared to the measured damage residual vectors. The connectivity vector with a projection onto the measurement space that is most collinear with the measured damage residual vector indicates the damaged DOFs. If the damage case has a rank greater than one, the process may be repeated until the required connectivity vectors are found to span the space needed to reproduce the measured damage residual vector.

2.5.1.1 Matrix Disassembly

Matrix disassembly is a process that decomposes a structural matrix into a matrix representation of the connectivity between DOFs and a matrix containing the magnitude information [28]. Advanced stiffness matrix disassembly techniques use a method that disassembles each elemental stiffness matrix. However, the method used in this work disassembles the entire structure into a set of equivalent springs. As a result, the disassembly is exact for truss structures, but inexact for more complicated systems. The advantage of this general technique is that it can be applied to any model without detailed knowledge of the actual elements used in the assembly. The formulation for the stiffness matrix is as follows

$$\mathbf{K} = \mathbf{C}\mathbf{P}\mathbf{C}^{\mathrm{T}}, \tag{46}$$

where **K** is the stiffness matrix, **C** is a sparse matrix containing all the connectivity information, and **P** is a diagonal matrix containing all magnitude information. Matrix **C** has $(n \times m)$ elements, where n is the matrix dimensions of **K**, and m is equal to the total number of unique entries in **K**, which consists of the non-zero entries in the upper triangular portion for symmetric stiffness matrices. Therefore, if the stiffness matrix is symmetric, m amounts to the number of nonzero entries in the upper triangular portion of **K**. The diagonal matrix **P** is calculated as

$$\begin{aligned} \mathbf{P}(\mathrm{i},\mathrm{i}) &= \sum_{\mathrm{j}=1}^{n} \mathbf{K}(\mathrm{i},\mathrm{j}) \quad \mathrm{i} = 1:n \\ \mathbf{P}(\mathrm{i},\mathrm{i}) &= -\mathbf{K}(\mathrm{j},\mathrm{k}) \quad \mathrm{i} = n+1:m \end{aligned} . \tag{47}$$

The equations above indicate that the first n diagonal elements of **P** are the row sums of the stiffness matrix, while the remaining diagonal elements of **P** are the opposites of the

nonzero off-diagonal elements of **K**. The nonzero off-diagonal elements of **K** play an important role in generating the connectivity matrix, **C**.

The connectivity matrix, **C**, can be written as [$\mathbf{C}_1$ $\mathbf{C}_2$], where $\mathbf{C}_1$ is the $(n \times n)$ identity matrix. $\mathbf{C}_2$ is a $(n \times m-n)$ matrix with the entries defined according to the element locations of the unique nonzero off-diagonal entries in matrix **K**. For each element $\mathbf{K}(\mathrm{j},\mathrm{k})$ used to define the element $\mathbf{P}(\mathrm{i},\mathrm{i})$ for the index $\mathrm{i} = n+1:m$, the i$^{\text{th}}$ column of **C** is given as

$$\begin{aligned} \mathbf{C}(\mathrm{j},\mathrm{i}) &= 1.0 \quad \mathrm{i} = n+1:m \\ \mathbf{C}(\mathrm{k},\mathrm{i}) &= -1.0 \quad \mathrm{i} = n+1:m \end{aligned}. \tag{48}$$

The expanded dynamic residual only requires the formulation of **C**.

2.5.1.2 MEDR Development

Using the model reduction transformation matrix relating the reduced system DOFs to the full system DOFs, **T**, found in Equation (20), the Ritz damage residual matrix in Equation (40) can be rewritten as

$$\mathbf{B}_{\mathrm{d}} = \mathbf{K}\mathbf{T}\left(\hat{\mathbf{V}}\right)_{\mathrm{r}} - \mathbf{M}\mathbf{T}\left(\mathbf{V}\right)_{\mathrm{r}}. \tag{49}$$

Guyan static reduction is utilized for this application of MEDR. Pre-multiplying by $\mathbf{T}^{\mathrm{T}}$, Equation (49) becomes

$$\mathbf{T}^{\mathrm{T}}\mathbf{B}_{\mathrm{d}} = \mathbf{T}^{\mathrm{T}}\mathbf{K}\mathbf{T}\left(\hat{\mathbf{V}}\right)_{\mathrm{r}} - \mathbf{T}^{\mathrm{T}}\mathbf{M}\mathbf{T}\left(\mathbf{V}\right)_{\mathrm{r}}. \tag{50}$$

The reduced damage residual is then defined as

$$\left(\mathbf{B}_{\mathrm{d}}\right)_{\mathrm{r}} = \mathbf{K}_{\mathrm{r}}\left(\hat{\mathbf{V}}\right)_{\mathrm{r}} - \mathbf{M}_{\mathrm{r}}\left(\mathbf{V}\right)_{\mathrm{r}}, \tag{51}$$

where $\mathbf{K}_{\mathrm{r}}$ and $\mathbf{M}_{\mathrm{r}}$ are the reduced stiffness and mass matrices defined in Equation (15). By combining Equations (50) and (51), the relationship between the reduced and full damage residual matrices is given as

$$\left(\mathbf{B}_{\mathrm{d}}\right)_{\mathrm{r}} = \mathbf{T}^{\mathrm{T}}\mathbf{B}_{\mathrm{d}}. \tag{52}$$

If the connectivity information is assumed to be unaffected by damage, the changes in the stiffness would be contained in the diagonal magnitude matrix **P**. Therefore, the connectivity matrix would be invariant and can be used as a linearly dependent set of basis vectors to approximate the target damage vector. The target vector, denoted $\underline{u}_{\mathrm{tar}}$, is chosen as the first left singular vector of the reduced damage residual matrix. It is assumed that the target vector is related to the unknown "full target" vector by Equation (52),

$$\underline{u}_{\mathrm{tar}} = \mathbf{T}^{\mathrm{T}}\underline{u}_{\mathrm{full}}. \tag{53}$$

The unknown "full target" vector is approximated as a linear combination of the columns of the connectivity matrix **C**

$$\underline{u}_{\mathrm{full}} = \mathbf{C}\underline{\alpha}, \tag{54}$$

where $\underline{\alpha}$ is a vector of constants defining the contribution of each individual column of **C**. The number of columns of **C** will exceed the rank of matrix **C**, therefore a subset of the columns must be selected for use in Equation (54), resulting in

$$\underline{u}_{\mathrm{full}} = \mathbf{C}_{\mathrm{sel}}\underline{\alpha}. \tag{55}$$

The reduced target vector can be approximated using Equations (53) and (55)

$$\underline{u}_{\mathrm{tar}} = \mathbf{T}^{\mathrm{T}}\mathbf{C}_{\mathrm{sel}}\underline{\alpha}. \tag{56}$$

The reason for selecting the "best" columns of **C** is known as the problem of subset selection, but the current problem differs from the standard problem. In the standard subset selection problem, the concept is to choose the columns of **C** that are most linearly independent, whereas in MEDR the columns that best represent the reduced

target vector are chosen. The combinatorial optimization problem that arises is not feasible for even modestly sized models, so a reasonable sub-optimal solution strategy must be implemented. The "best subspace" algorithm [34] poses the following minimization problem for each column of **C**

$$\min_{\substack{\text{wrt} \\ \beta}} \varepsilon_i = \left(\underline{u}_{tar} - \beta \mathbf{T}^T \mathbf{c}_i\right)^T \left(\underline{u}_{tar} - \beta \mathbf{T}^T \mathbf{c}_i\right), \tag{57}$$

yielding the solution

$$\beta^* = \frac{\underline{u}_{tar} \mathbf{T}^T \mathbf{c}_i}{\left(\mathbf{T}^T \mathbf{c}_i\right)^T \left(\mathbf{T}^T \mathbf{c}_i\right)}. \tag{58}$$

The next step of the "best subspace" algorithm is to define a new target vector as

$$\underline{u}_{tar}^{i} = \underline{u}_{tar}^{i-1} - \beta \mathbf{T}^T \mathbf{c}_i^*, \tag{59}$$

where $\mathbf{c}_i^*$ represents the first column selected and the superscript i is equal to two. Essentially, the new target vector has no component in the first selected "reduced" column direction. The next step is to determine that column of **C** that minimizes Equation (57) using the target vector defined in Equation (59). The process can be repeated a predetermined number of times or until the error defined in Equation (57) meets a user-specified criteria. All identified columns of **C** can be grouped in a matrix denoted $\mathbf{C}_{sel}$, these columns are used to indicate which DOFs of the full model have been affected by damage. Additionally, the full damage vector can be estimated by solving

$$\underline{u}_{full} \approx \mathbf{C}_{sel} \underline{\alpha}^*, \tag{60}$$

where $\underline{\alpha}^*$ represents the least squares solution to Equation (56).

2.5.1.3 Identical Signatures

Since the MEDR process involves a projection from one vector space onto one of its subspaces, the possibility exists for a particular measured damage residual vector to be equally expressible by multiple columns of the connectivity matrix. If two linearly independent columns of the connectivity matrix have collinear projections on the subspace of measured DOFs, the corresponding full DOF damage residual vectors will have identical damage signatures. These vectors will be indicated by identical values of ε_i in Equation (57). When multiple columns of the connectivity matrix produce the same minimum value of ε_i the actual damage location might be indicated by any of those columns. In practice, identical damage signatures are found in structures with repetitive substructures, such as trusses, conventional buildings, and bridges.

2.5.2 Modified Method of Expanded Dynamic Residuals

The MEDR method above has been determined to be a good tool for detecting damage when only reduced measurements are available. The Modified Method of Expanded Dynamic Residuals (MMEDR) utilizes the same concepts as those discussed above with some modifications to the inputs used. While the MEDR uses the Matrix Disassembly method presented above to determine the DOF connectivity, the aim of the MMEDR is to exploit the structural information when it is available to determine the actual elemental connectivity.

As discussed in Section 2.6.1.1, the Matrix Disassembly method divides a structural matrix into a set of equivalent springs. A more advanced and robust method would be to disassemble the stiffness matrix into individual elements. The main advantage of the Matrix Disassembly method is that it requires only the stiffness matrix

of the structure; however, if the elemental connectivity information is available, then a more accurate disassembly is possible.

2.5.2.1 Elemental Disassembly

The sparse matrix containing the elemental information, $\mathbf{U}_{da,}$ will be a $(n \times p)$ matrix where *n* corresponds to the matrix dimensions of the stiffness matrix **K**, and *p* equals the number of elements in the structure. The connectivity can be found by simulating damage to each element of the full FEM and determining the singular vector for each damage case such that

$$\Delta \mathbf{K}_i = \mathbf{K} - \mathbf{K}_{di}, \tag{61}$$

where $\mathbf{K}_{di}$ is the simulated damage stiffness matrix for each element, and $\Delta \mathbf{K}_i$ indicates the simulated effects of damage on the stiffness matrix. Next, the SVD process is used to determine the simulated singular vectors and singular values

$$\Delta \mathbf{K}_i = [\mathbf{u}_{di}][\mathbf{s}_{di}][\mathbf{v}_{di}]^T. \tag{62}$$

In the above equation, $\mathbf{u}_{di}$ are the left singular vectors, $\mathbf{s}_{di}$ are the singular values, and $\mathbf{v}_{di}$ the right singular vectors. If the damage is simulated on a single element basis, then the damage will be of rank one and only the first left singular vector is required to indicate the damaged DOFs. The matrix consisting of the elemental information is next constructed as follows.

$$\mathbf{U} = \left[\mathbf{u}_{d1} \mid \mathbf{u}_{d2} \mid \ldots \mid \mathbf{u}_{dp}\right]. \tag{63}$$

The **U** matrix inherently contains a number of advantages over the **C** matrix presented in Section 2.6.1.1. First, the **U** matrix will have fewer columns than **C** when there are diagonal elements in the structure. This is due to the fact that diagonal elements in the **C** matrix are represented by two or three columns while the **U** matrix will have

them combined in a single column. This will help in avoiding any false positives that may occur when the columns are reduced to the measured DOFs. Secondly, since the columns of the **U** matrix correspond to the structural elements, it is easier to remove any elements that are known not to be damaged and therefore not required.

2.5.2.2 MMEDR Development

The MEDR development shown in Equations (49) through (53) will also apply to the MMEDR. However, the "full target" vector in Equation (54) is modified to replace **C** with **U** such that

$$\underline{u}_{full} = \mathbf{U}\underline{\alpha 1}, \tag{64}$$

where $\underline{\alpha 1}$ is a vector of constants defining the contribution of each individual column of **U**. Since the number of columns of **U** exceed the rank of the **U** matrix, a subset of the columns must be selected for use in Equation (64), resulting in

$$\underline{u}_{full} = \mathbf{U}_{sel}\underline{\alpha 1}. \tag{65}$$

The reduced target vector can be approximated using Equations (53) and (65)

$$\underline{u}_{tar} = \mathbf{T}^T\mathbf{U}_{sel}\underline{\alpha 1}. \tag{66}$$

As in Section 2.6.1.2, the "best" columns of **U** are selected that represent the reduced target vector best. The "best subspace" algorithm presents the following minimization problem for each column of **U**

$$\min_{\substack{wrt \\ \beta 1}} \varepsilon 1_i = \left(\underline{u}_{tar} - \beta 1\mathbf{T}^T u_{di}\right)^T\left(\underline{u}_{tar} - \beta 1\mathbf{T}^T u_{di}\right), \tag{67}$$

yielding the solution

$$\beta 1^* = \frac{\underline{u}_{tar}\mathbf{T}^T u_{di}}{\left(\mathbf{T}^T u_{di}\right)^T\left(\mathbf{T}^T u_{di}\right)}. \tag{68}$$

The next step of the "best subspace" algorithm is to define a new target vector as

$$\underline{u}_{tar}^{i} = \underline{u}_{tar}^{i-1} - \beta 1 \mathbf{T}^{T} u_{di}^{*}, \tag{69}$$

where u_{di}^{*} represents the first column selected and the superscript i is equal to two. Again, the new target vector has no component in the first selected "reduced" column direction. The next step is to determine that column of **U** that minimizes Equation (67) using the target vector defined in Equation (69). The process can be repeated a predetermined number of times or until the error defined in Equation (67) meets a user-specified criteria. All identified columns of **U** can be grouped in a matrix denoted $\mathbf{U}_{sel}$, these columns are used to indicate which DOFs of the full model have been affected by damage. Additionally, the full damage vector can be estimated by solving

$$\underline{u}_{full} \approx \mathbf{U}_{sel} \underline{\alpha 1}^{*}, \tag{70}$$

where $\underline{\alpha 1}^{*}$ represents the least squares solution to Equation (66).

2.5.3 Dot Product Damage Residual Expansion Method

The Dot Product Damage Residual Expansion Method (DPDREM) uses the same principles as the MEDR and MMEDR presented above, where the full DOF damage vector cases are reduced to the target damage vector DOFs and compared to determine the full DOF vector that would represent the damage location. Unlike the two methods above, the DPDREM uses the dot product method in an attempt to simplify the subset selection method.

To begin, the DPDREM uses the same disassembly method presented for the MMEDR in Section 2.6.2.1. After determining the elemental information matrix, **U**, the target vector is noted to be related to the unknown "full target" vector by Equation (52)

$$\underline{u}_{full} = \mathbf{U}_{sel}\underline{\alpha 2}\,. \tag{71}$$

The reduced target vector can be approximated using Equations (53) and (71)

$$\underline{u}_{tar} = \mathbf{T}^{T}\mathbf{U}_{sel}\underline{\alpha 2}\,. \tag{72}$$

Additionally, Equation (72) can be modified as follows

$$\left|\underline{u}_{tar}\right| = \left|\mathbf{T}^{T}\mathbf{U}_{sel}\underline{\alpha 2}\right|, \tag{73}$$

where the bars represent absolute values for each side of the equation. This helps avoid sign mismatches when determining the "best" columns of **U** to represent the reduced target vector. Next, the following maximization problem is presented for each column of **U**

$$\max \varepsilon 2_{i} = \left|\underline{u}_{tar}\right| \bullet \left|\mathbf{T}^{T} u_{di}\right|. \tag{74}$$

The value of $\varepsilon 2_{i}$ closest to one will determine the selected column of **U**. For damage affecting multiple elements, a new target vector can be defined

$$\underline{u}_{tar}^{i} = \underline{u}_{tar}^{i-1} - \left(\max \varepsilon 2_{i}\right)\left(\mathbf{T}^{T} u_{di}^{*}\right), \tag{75}$$

where u_{di}^{*} represents the first column selected and the superscript i is equal to two. The new target vector has no component in the first selected "reduced" column direction. The next step is to determine that column of **U** that maximizes Equation (74) using the target vector defined in Equation (75). The process can be repeated a predetermined number of times or until the error defined in Equation (74) meets a user-specified criteria, keeping in mind that the process will not require repetition for single elemental damage as each column of **U** denotes the selected element. For multiple elemental damage, all identified columns of **U** can be grouped in a matrix denoted $\mathbf{U}_{sel}$, these columns are used to indicate

which DOFs of the full model have been affected by damage. The full damage vector can be estimated by solving

$$\underline{u}_{full} \approx \mathbf{U}_{sel} \underline{\alpha 2}^{*}, \tag{76}$$

where $\underline{\alpha 2}^{*}$ represents the least squares solution to Equation (72). The advantage of the DPDREM method lies in the simplicity of the dot product as compared to the previous methods.

2.6 Health Monitoring for Critical Structural Areas

Considering that Ritz vectors are constructed using a recursive process based on the applied load, a different set of orthogonal Ritz vectors will be generated when the loading location or angle is changed. The varying Ritz vector sets will produce different sensor sets using the sensor placement techniques discussed in Section 2.3. With different sensor sets the damage residual matrix may vary, which in turn can help improve the damage vector selection process.

Therefore, changing the loading case gives the flexibility to optimize the sensor set when attempting to monitor certain areas of the structure. This would be most advantageous in complex structures where relatively few sensors can be placed compared to the full model DOFs.

3 Analytical Examples

This chapter presents examples of the methods introduced in the previous chapter. Section 3.1 introduces the basic concept of singular vector projections and dot products using a simple spring-mass system for both mode shapes and Ritz vectors. Section 3.2 is used to show the Modified Method of Expanded Dynamic Residual (MMEDR) and Dot Product Damage Residual Expansion Method (DPDREM) results for a simple 7-degree-of-freedom (DOF) truss system with reduced sensor sets. The NASA eight-bay truss, discussed in Section 3.3, is used to demonstrate the methods for a more complex system.

3.1 Example 1: Spring-Mass System

A three-DOF spring-mass system was used as a simple example and to verify the basic concepts employed. The section presents the use of the damage residual singular vectors and values for damage detection in a visual manner. The system consists of three springs placed in-line with equal masses attached at each end. One end of the 1st spring is fixed. Figure 1 shows a depiction of the system. The 1st and 3rd spring constants were $1x10^8$ N/m, the 2nd spring constant was $2x10^8$ N/m and the three masses were 100 kg each.

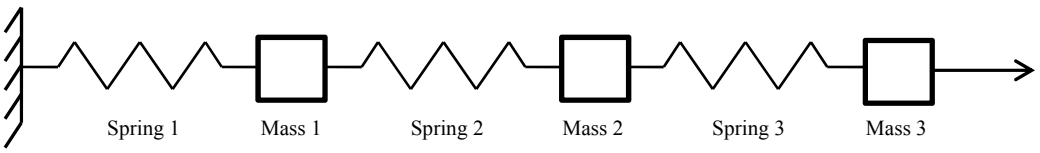

Figure 1: Three degree of freedom spring-mass system

The healthy and damaged stiffness matrices were obtained using simple FEM. Damage was simulated by reducing the selected spring stiffness constant by 50%. Noise was also simulated by reducing the selected stiffness constant by 10%. The noise-free damage cases, dK_{NF}, were found by subtracting the healthy and damaged stiffness

matrices. The actual noise induced damage cases, dK_N, were found by subtracting the healthy stiffness matrix from the noise induced damage stiffness matrices. The three DOFs were measured when using the Ritz vectors and mode shapes to simulate the damaged test structure. Additionally, the Ritz vectors were found by forcing the three masses with 10 N in the direction of the arrow shown in Figure 1.

The first case includes damage to Spring 1 and noise in Spring 3. The actual absolute values of the noise-free and noise induced damage matrices (dK) are shown in Figure 2.

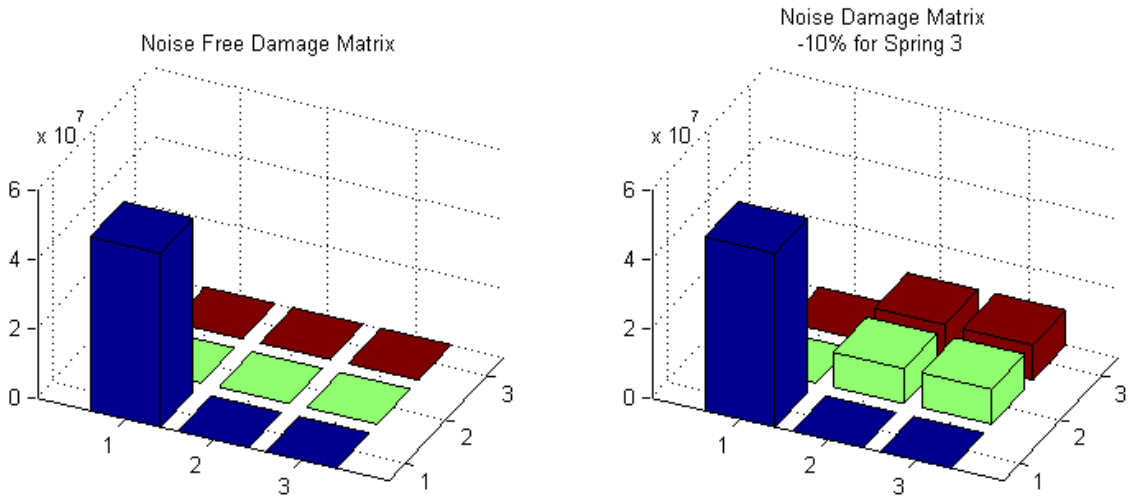

Figure 2: Spring 1 Damage Matrix

As can be seen in the figure above, all the DOFs affected by noise were separate from the damaged DOF. The Minimum Rank Perturbation Theory (MRPT) was applied to the noise induced damaged Ritz vectors and mode shapes to find the analytical damage results. Figure 3 shows the normalized damage residual singular values obtained to select the damage rank in MRPT.

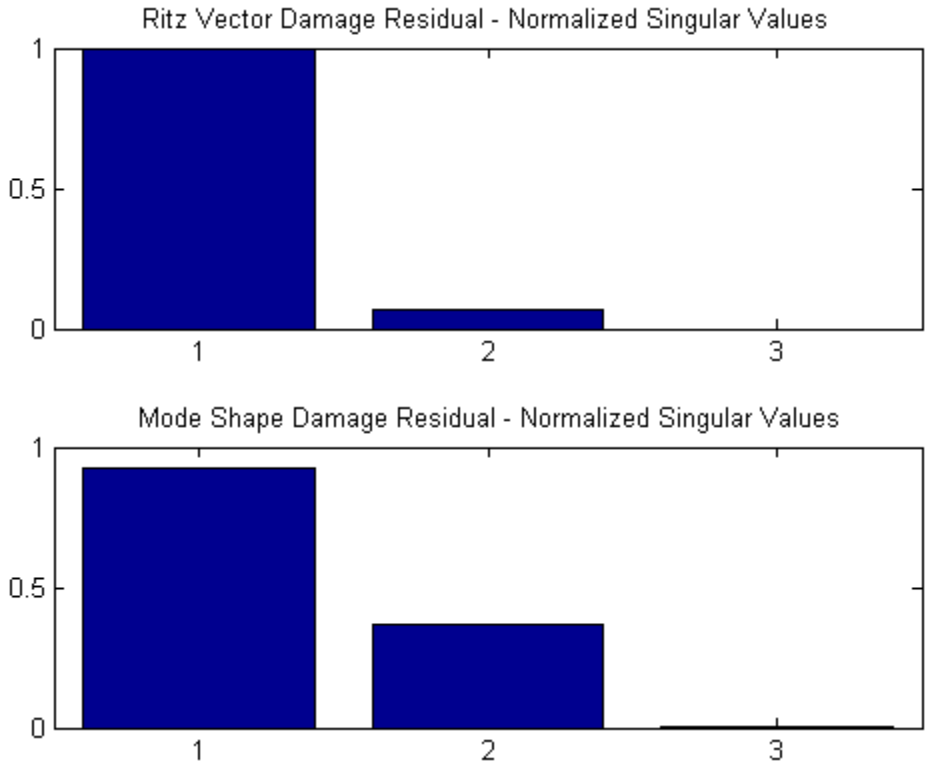

Figure 3: Spring 1 Damage Residual Singular Values

The second normalized singular value for the mode shape damage residual was found to be 40% the magnitude of the first singular value, which is significant and should be considered in the MRPT calculation. On the other hand, the second damage residual singular value obtained using the Ritz vectors was 7% of the first singular value.

The first damage residual singular vectors for the mode shapes (u_M) and Ritz vectors (u_R) were next inspected and compared to the first noise free singular vector taken from dK_{NF} in Figure 1 (u_{NF}). Figure 4 shows the singular vectors plotted as 3-dimensional projections where each axis represents a degree of freedom. The dot product between u_M and u_{NF} was used to find the angle between the two vectors ($Angle_M$), similarly ($Angle_R$) was found with the dot product between u_R and u_{NF}. This was the basis for the sensitivity analysis performed to determine the effect of noise on each method.

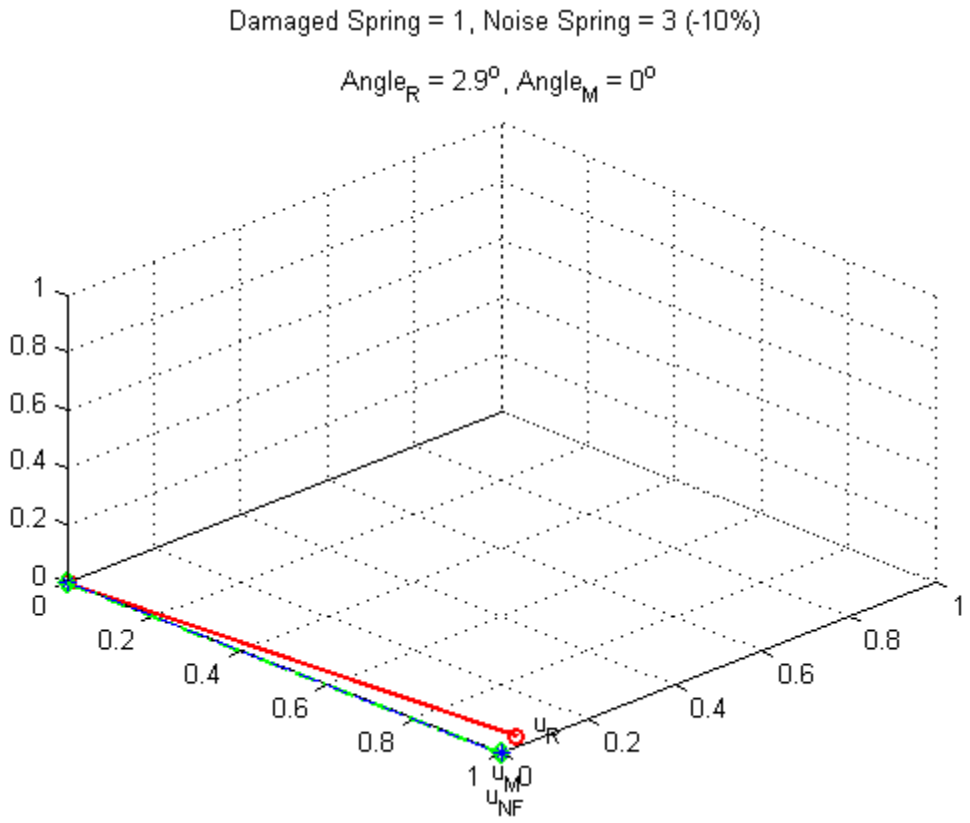

Figure 4: Damaged Residual Singular Vector Projections

The figure above shows that for the first singular vector, both u_M and u_R accurately replicate u_{NF}. Although the Ritz vectors did display a slight angle of 2.9º, which would again be attributed to the coupling inherent in the Ritz vector calculation, it is important to note that the Ritz vectors still obtained better overall results as the rank of damage was correctly found to be one.

The second case includes damage to the 3rd spring and noise in the 2nd spring. Figure 5 shows the dK_{NF} and dK_N matrices. Note that the noise in this case also affected a damaged DOF. The damage residual singular values of the noise induced Ritz vectors and mode shapes are shown in Figure 6.

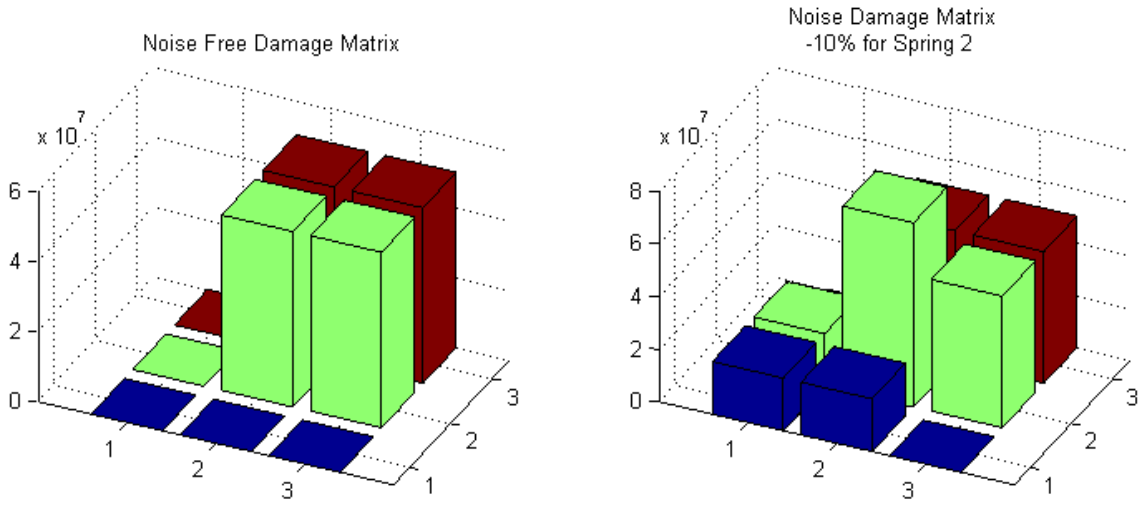

Figure 5: Spring 3 Damage Matrix

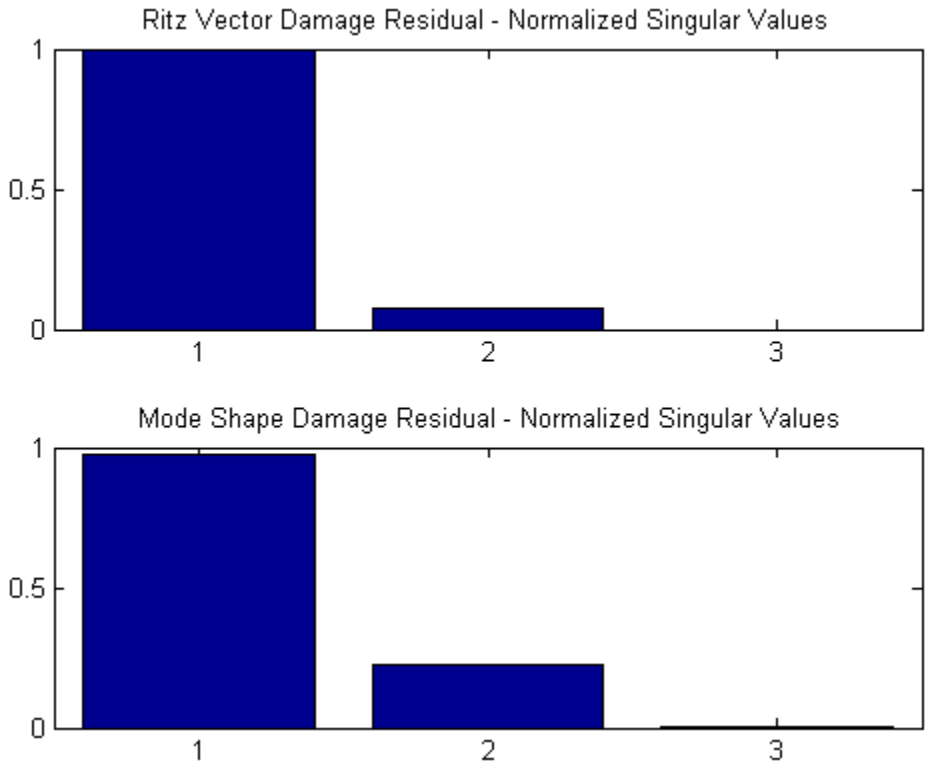

Figure 6: Spring 3 Damage Residual Singular Values

The second normalized singular value of the mode shape damage residual was 23% of the first singular value, while second normalized singular value of the Ritz vector

damage residual was 8% of the first singular value. In this case, it is clearer that the damage is of single rank for both basis vectors.

The first singular vector for each of the mode shapes (u_M) and Ritz vectors (u_R) were again inspected and compared to the first noise free singular vector taken from dK_{NF} (u_{NF}). $Angle_M$ and $Angle_R$ were found using the dot product as in the previous example. The results are shown in Figure 7.

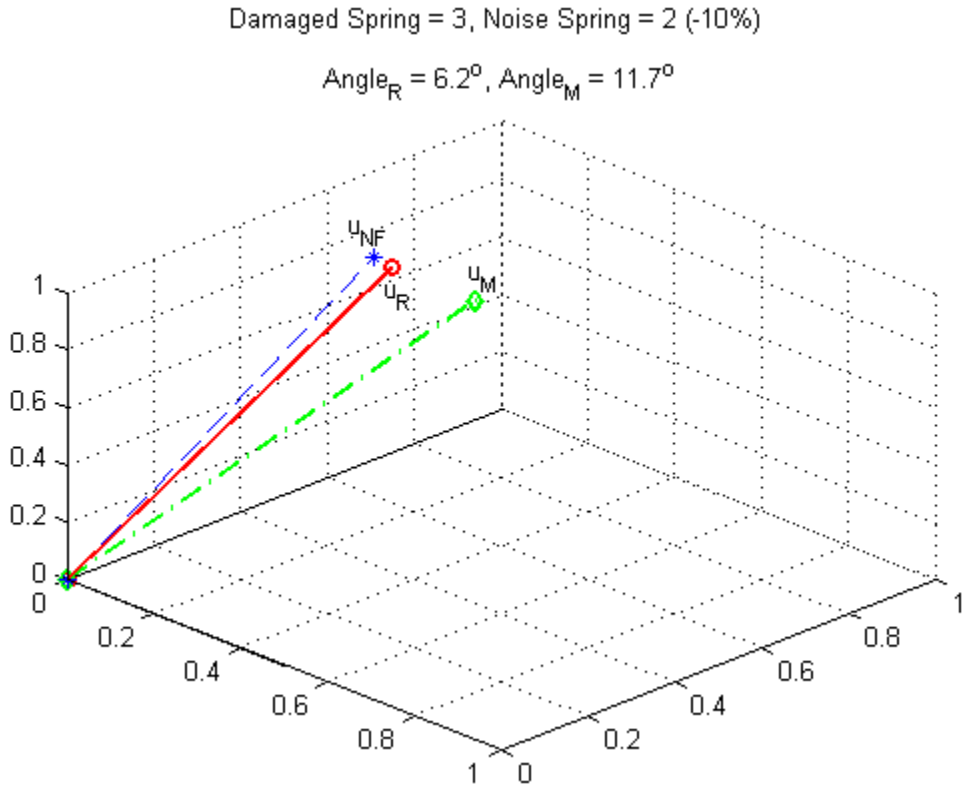

Figure 7: Damaged Residual Singular Vector Projections

The singular vector projections in Figure 7 clearly show the Ritz vector results to be closer to the noise free damage singular vector, with an angle of 6.2º compared to the 11.7º angle for the mode shape singular vector.

The results above aid in confirming the higher accuracy of Ritz vectors compared to mode shapes. Figures 3 and 6 indicate that Ritz vector damage residuals are less

susceptible to falsely identifying higher ranks of damage due to noise. This is due to the fact that each Ritz vector describes behavior across a range of frequencies whereas each mode correlates to a single frequency. The inherent coupling in the Ritz vectors aid in analyzing multiple frequency responses for a damage case when calculating the damage residuals.

Figures 4 and 7 show that both the Ritz vectors and mode shapes identify damage projections that closely match the actual noise-free damage using the singular vectors. For the second case, the Ritz vector singular vector projection showed a smaller angle when compared to the actual damage. For the first case, even though the Ritz vector singular projection angle was higher, the damage residual showed a rank one damage scenario more clearly than the mode shape damage residual. Overall, the Ritz vector results were slightly better compared to the mode shapes for this example.

3.2 Example 2: Simple Truss System

The second example consists of a simple truss system used to illustrate the Modified Method of Expanded Dynamic Residual (MMEDR) and Dot Product Damage Residual Expansion Method (DPDREM) presented in Sections 2.6.2 and 2.6.3 respectively. The effect of varying actuator locations on the damage detection results is also demonstrated in this example using the Basis Vector Product (BVP), Kinetic Energy Product (KE), and Effective Independence (EI) shown in Sections 2.3.2 through 2.3.4. The 7-DOF truss system shown in Figure 8 consists of seven elements with cross sectional areas of 1 cm^2, an elastic modulus of 210 GPa, and a density of 7850 kg/m^3. Damage was simulated by reducing the elastic modulus of the element of interest by 50%. The range of noise on the system was ±10% of the elastic modulus.

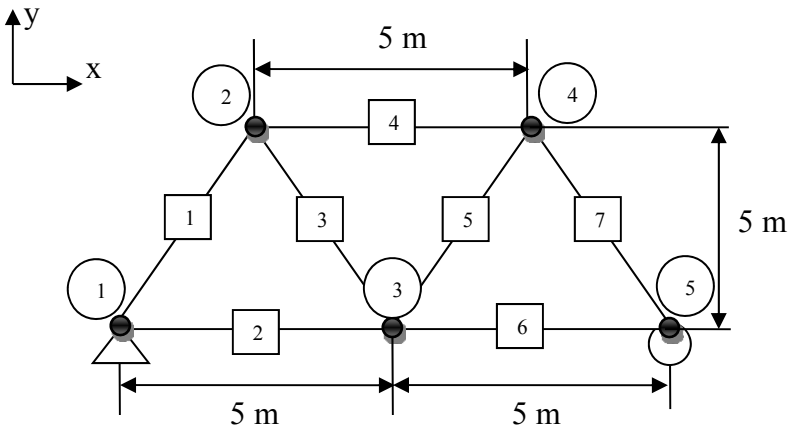

Figure 8: Simple Truss System

3.2.1 Case 1: EI sensor set with force at node 4

For the first case, the EI method was used with a 141 N force placed at node 4 with a 45º angle. The EI method was utilized to determine the four most linearly independent DOFs for the first three full undamaged Ritz vectors. The sensor set included DOFs 1, 4, 5, and 6 as shown in Figure 9.

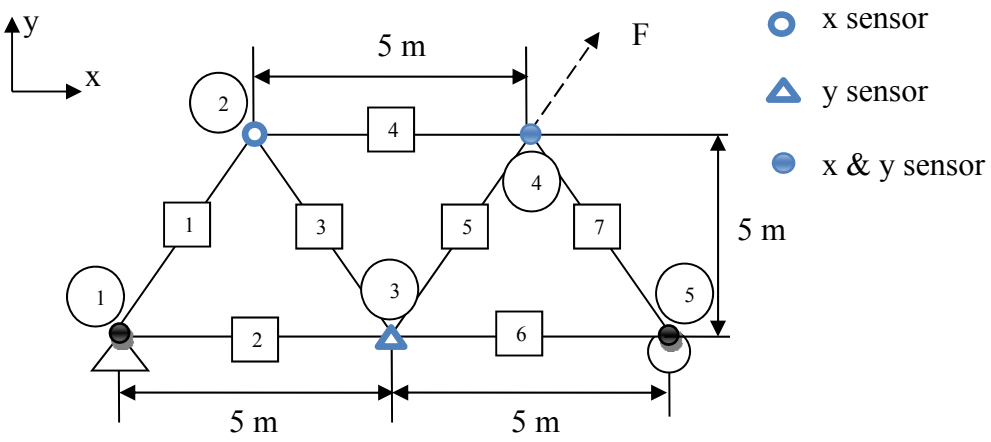

Figure 9: Case 1 Actuator and Sensor Set

The DPDREM and MMEDR were next used to determine the damaged elements for all single damaged element cases with the sensor set found above. Tables 1 and 2 show the DPDREM and MMEDR results respectively for all seven damage cases.

Table 1: Case 1 Damaged Simple Truss DPDREM results

		$T'*u_{fa}$						
	Element	1	2	3	4	5	6	7
u_{dr}	1	**0.99**	0.69	**1.00**	0.57	0.56	0.15	0.10
	2	0.65	**1.00**	0.66	0.88	0.57	0.54	0.41
	3	**0.99**	0.69	**1.00**	0.57	0.56	0.15	0.10
	4	0.45	0.87	0.53	**1.00**	0.19	0.40	0.26
	5	0.60	0.57	0.55	0.27	**1.00**	0.77	0.77
	6	0.12	0.39	0.12	0.30	0.78	**0.99**	**1.00**
	7	0.12	0.39	0.12	0.30	0.78	**0.99**	**1.00**

Table 2: Case 1 Damaged Simple Truss MMEDR results

		$T'*u_{fa}$						
	Element	1	2	3	4	5	6	7
u_{dr}	1	**0.01**	0.99	**0.00**	0.77	0.68	1.00	1.00
	2	0.98	**0.00**	0.99	0.95	0.72	0.90	0.99
	3	**0.01**	0.99	**0.00**	0.77	0.68	1.00	1.00
	4	0.79	0.95	0.72	**0.00**	0.96	0.93	0.94
	5	0.64	0.72	0.70	0.97	**0.00**	0.79	0.64
	6	1.00	0.98	1.00	0.91	0.61	**0.05**	**0.00**
	7	1.00	0.98	1.00	0.91	0.61	**0.05**	**0.00**

Tables 1 and 2 show the actual damage cases along the diagonal of the table and the selected damage cases highlighted in gray. Note that while the DPDREM locates the damage by finding the maximum dot product, the MMEDR determines the damage locations by finding the minimum 'error' value. In this case, the two methods identified the exact same DOFs. The difference is that the DPDREM uses the simpler method for comparison whereas the more complex subset selection algorithm is used in the MMEDR. The tables show that all seven damage cases were correctly identified using

either method, however reduced cases 1 and 3 and cases 6 and 7 were indistinguishable with the defined sensor set.

The selected full DOF damage residual singular vectors using DPDREM and MMEDR were next plotted as shown in Figure 10. The darker color bars indicate the actual damaged residual singular vectors while the lighter color bars indicate the selected singular vector results. In the cases where two singular vectors could not be distinguished, the vectors were combined and normalized for visual simplicity. As can be seen in Figure 10, all the damage locations were identified correctly.

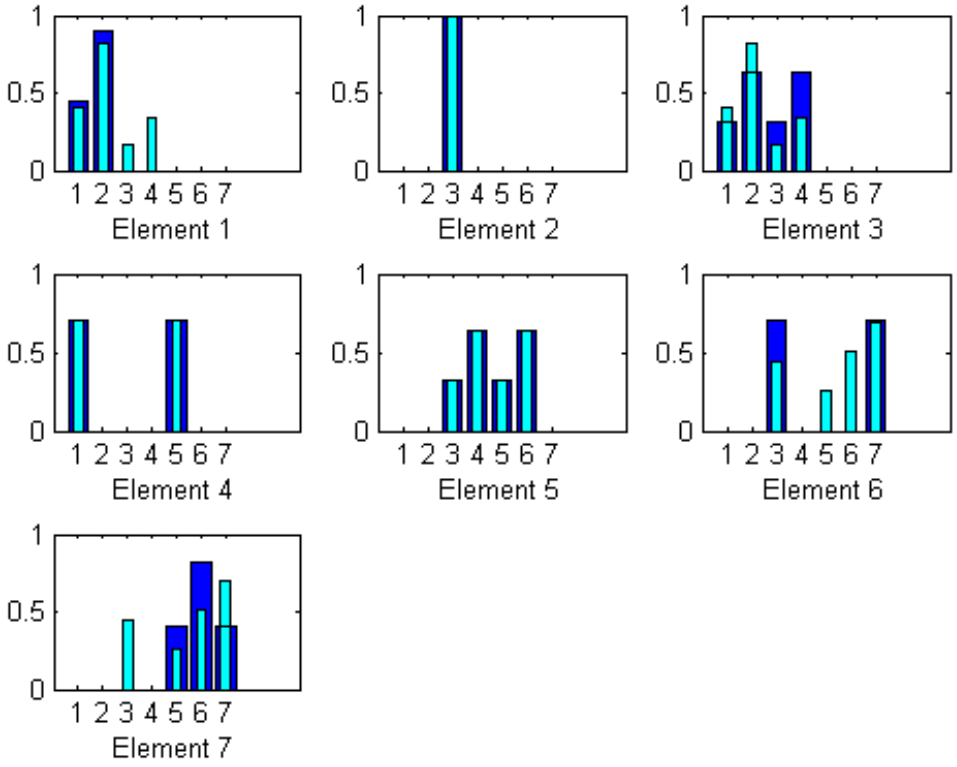

Figure 10: Case 1 Damaged Simple Truss DPDREM and MMEDR Singular Vectors

3.2.2 Case 2: BVP and KE sensor sets with force at node 4

The second case is used to illustrate how the results may differ using the BVP or KE methods from the EI method presented in case 1 with the same force input of 141 N placed at node 4 with a 45º angle. As mentioned previously, the BVP and KE methods

identify DOFs that display the highest displacement and kinetic energy, respectively. For this system, the two methods obtain very similar sensor set results since the mass matrix is nearly uniform. In Case 2, the sensor set determined included DOFs 1, 2, 4, and 6 as shown in Figure 11. The DPDREM was again used for all seven single element damage cases with the new sensor set. The results are shown in Table 3.

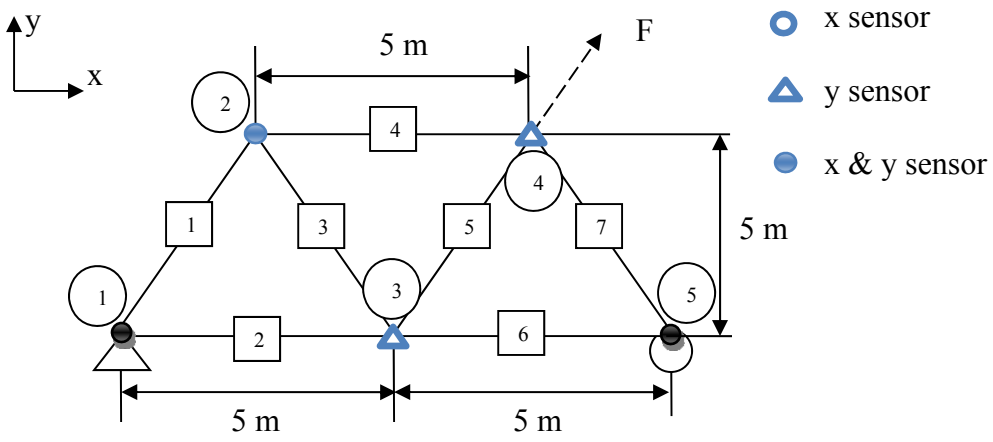

Figure 11: Case 2 Actuator and Sensor Set

Table 3: Case 2 Damaged Simple Truss DPDREM results

		$T'*u_{fa}$						
	Element	1	2	3	4	5	6	7
u_{dr}	1	**1.00**	0.81	0.68	0.46	0.25	0.49	0.25
	2	0.91	**0.98**	0.76	0.77	0.52	**0.81**	0.43
	3	0.72	0.70	**0.99**	0.71	0.55	0.53	0.26
	4	0.43	0.72	0.90	**0.93**	0.77	0.78	0.41
	5	0.18	0.38	0.60	0.60	**0.96**	0.53	0.83
	6	0.20	0.31	0.23	0.32	0.84	0.44	**0.99**
	7	0.20	0.31	0.23	0.32	0.84	0.44	**0.99**

As can be seen in Table 3, the case 2 sensor set improved in some instances and deteriorated in others. Damaged elements 1 and 3 were clearly identified using the second sensor set, whereas element 6 was incorrectly identified as element 2, and once again

element 7 could not be distinguished from element 6. The reason for this change in the damage identification result is due to the sensor set variation with the replacement of DOF 5 with DOF 2. The sensor set variation basically shifted the focus towards the left of the structure improving the detectability of elements 1 and 2, and diminishing the detectability of element 6. Figure 12 shows the selected DPDREM damage residual singular vectors compared to the actual singular vectors.

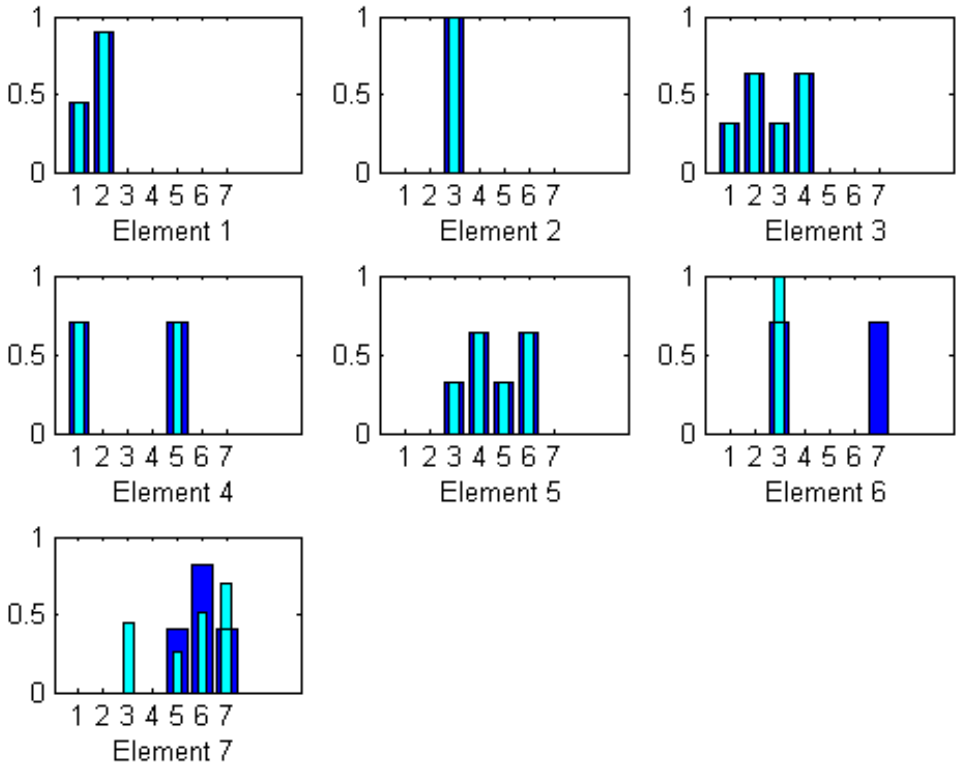

Figure 12: Case 2 Damaged Simple Truss DPDREM Singular Vectors

Figure 12 reiterates the discussion above. It should be noted that while element 6 was not determined exactly, the reduced sensor set was still capable of finding one of the two DOFs affected by the damage. When examining the reduced singular vectors for element 6, it should be noted that none of the cases obtain high dot product results due to the fact that the vectors don't compare well. This should prompt the experimenter to further examine the sensor set used for that element.

Next, the MMEDR 'error' values were determined and shown in Table 4. The selected MMEDR damage residual singular vector plots are shown in Figure 13.

Table 4: Case 2 Damaged Simple Truss MMEDR results

		$T'*u_{fa}$						
	Element	1	2	3	4	5	6	7
u_{dr}	1	**0.00**	1.00	0.88	0.94	0.94	0.88	0.94
	2	0.95	**0.04**	0.42	**0.42**	0.92	**0.38**	0.96
	3	0.82	0.52	**0.02**	0.49	0.91	0.72	0.98
	4	0.87	0.98	0.73	0.98	0.41	0.97	1.00
	5	0.98	1.00	0.79	0.91	**0.09**	0.97	0.56
	6	0.96	0.98	0.98	0.91	0.63	0.82	**0.02**
	7	0.96	0.98	0.98	0.91	0.63	0.82	**0.02**

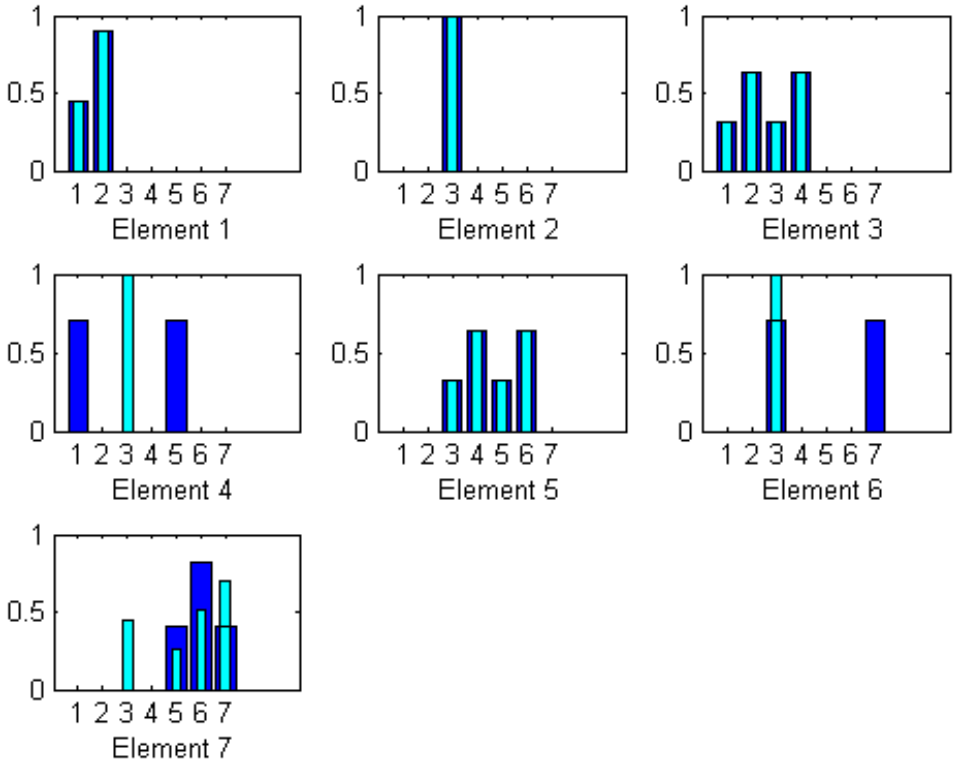

Figure 13: Case 2 Damaged Simple Truss MMEDR Singular Vectors

The MMEDR results mostly correlate with the DPDREM results. The difference was in element 4 where the MMEDR determined the damage to be at element 2 instead. The reason for this incorrect identification is due to the fact that the MMEDR takes into

account the sign differences, while the DPDREM circumvents such issues by taking the absolute values of the singular vectors. When comparing the MMEDR results in Figure 13 to the DPDREM results in Figure 12, it can be seen that the DPDREM more accurately identified the damage for element 4 while all other cases were identified identically. Once again the relatively high 'error' values for elements 4 and 6 indicate that the singular vectors do not match well and need further examination.

3.2.3 Case 3: EI sensor set with force at node 2

The third case returns to the EI method with the 141 N force placed at node 2 with a 45º angle as shown in Figure 14. The resulting sensor set is at DOFs 1, 2, 4, and 5. The DPDREM and MMEDR results are shown in Tables 5 and 6, respectively.

The sensor set obtained improved the DPDREM and MMEDR damage detection results for elements 1 and 2 using the EI sensor set but deteriorated detection in elements 5, 6, and 7. This is due to the fact that the sensors were shifted towards left side of the structure, which in turn diminished the detectability for the elements on the right-hand side.

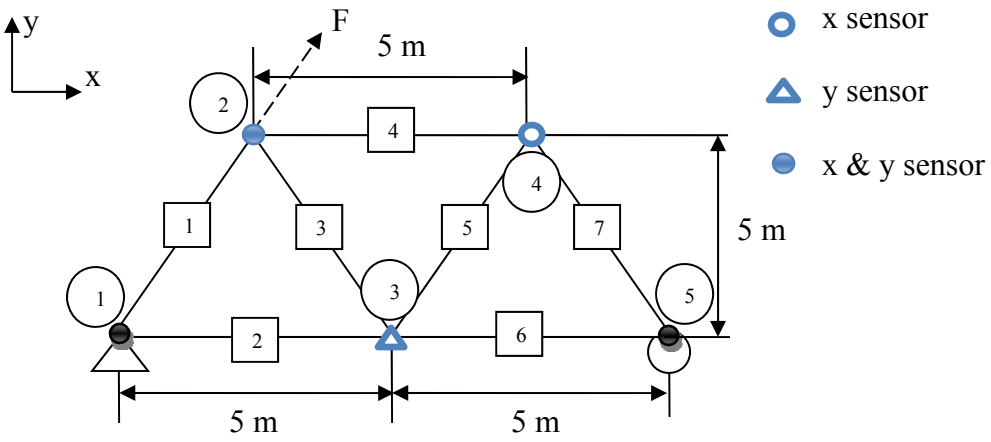

Figure 14: Case 3 Actuator and Sensor Set

Table 5: Case 3 Damaged Simple Truss DPDREM results

		$T'*u_{fa}$						
	Element	1	2	3	4	5	6	7
u_{dr}	1	**1.00**	0.78	0.65	0.35	0.18	0.08	0.18
	2	0.79	**1.00**	0.61	0.71	0.55	0.42	0.49
	3	0.73	0.65	**1.00**	0.35	0.74	0.71	0.77
	4	0.32	0.69	0.28	**1.00**	0.42	0.39	0.38
	5	0.24	0.58	0.78	0.53	**1.00**	**0.98**	**0.99**
	6	0.24	0.58	0.78	0.53	**1.00**	**0.98**	**0.99**
	7	0.24	0.58	0.78	0.53	**1.00**	**0.98**	**0.99**

Table 6: Case 3 Damaged Simple Truss MMEDR results

		$T'*u_{fa}$						
	Element	1	2	3	4	5	6	7
u_{dr}	1	**0.00**	0.79	0.85	0.91	0.98	1.00	0.98
	2	0.76	**0.00**	0.77	0.97	0.97	0.90	0.99
	3	0.83	0.70	**0.00**	0.94	0.46	0.64	0.40
	4	0.90	0.97	0.92	**0.01**	0.82	0.91	0.86
	5	0.99	0.97	0.40	0.82	**0.00**	**0.10**	**0.02**
	6	0.99	0.97	0.40	0.82	**0.00**	**0.10**	**0.02**
	7	0.99	0.97	0.40	0.82	**0.00**	**0.10**	**0.02**

Figure 15 shows the resulting damage residual singular vectors for both the DPDREM and MMEDR compared to the actual damage cases. The results in Figure 15 agree with the findings in Tables 5 and 6. This case shows the force locations can affect the sensor sets obtained using the EI sensor placement method.

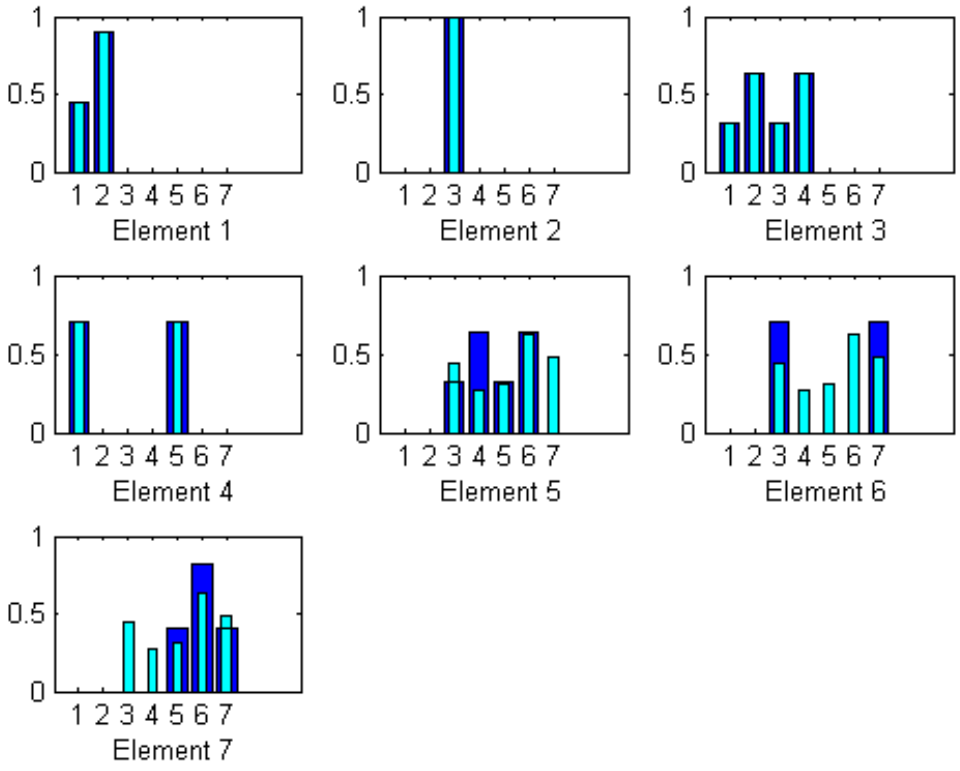

Figure 15: Case 3 Damaged Simple Truss DPDREM and MMEDR Singular Vectors

3.2.4 Case 4: BVP, KE, and EI sensor sets with force at nodes 4 and 5

The final case is used to show the improvement in detecting damage on the right side of the structure with force F1 located at node 4 (141 N with a 45º angle) and force F2 at node 5 (100 N with a 0º angle). The resulting sensor set using BVP, KE, and EI is at DOFs 1, 4, 6, and 7. The forces and sensor sets are shown in Figure 16. The DPDREM and MMEDR results are shown in Tables 7 and 8 respectively.

In this example case, elements 6 and 7 are clearly identified whereas elements 1 and 2 are indistinguishable from one another using both the DPDREM and MMEDR methods. Note that element 4 shows a fairly high MMEDR 'error' value even though the damaged element is properly identified. This can once again be mostly attributed to the sign differences in the damage residual singular vector sign differences. Figure 17 shows the damage residual singular vectors for both method results.

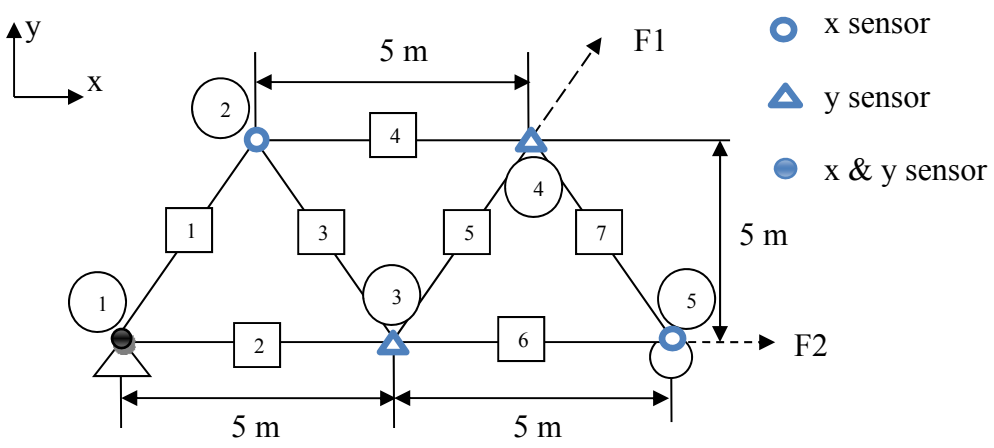

Figure 16: Case 4 Actuator and Sensor Set

Table 7: Case 4 Damaged Simple Truss DPDREM results

		$T'*u_{fa}$						
	Element	1	2	3	4	5	6	7
u_{dr}	1	**1.00**	0.43	**1.00**	0.81	0.69	0.38	0.41
	2	0.40	**1.00**	0.41	0.57	0.45	0.99	0.68
	3	**1.00**	0.43	**1.00**	0.81	0.69	0.38	0.41
	4	0.94	0.68	0.93	**0.86**	0.64	0.65	0.48
	5	0.68	0.48	0.67	0.67	**1.00**	0.31	0.80
	6	0.37	1.00	0.37	0.57	0.40	**0.99**	0.64
	7	0.38	0.72	0.41	0.34	0.82	0.60	**1.00**

Table 8: Case 4 Damaged Simple Truss MMEDR results

		$T'*u_{fa}$						
	Element	1	2	3	4	5	6	7
u_{dr}	1	**0.00**	0.99	**0.01**	0.57	0.90	0.88	0.83
	2	1.00	**0.00**	1.00	0.88	0.91	0.53	0.77
	3	**0.00**	0.99	**0.01**	0.57	0.90	0.88	0.83
	4	1.00	0.96	0.99	**0.53**	0.68	0.60	0.87
	5	0.88	0.91	0.91	0.64	**0.00**	0.90	0.69
	6	0.96	0.56	0.96	0.75	0.84	**0.02**	0.98
	7	0.86	0.70	0.83	1.00	0.66	0.93	**0.00**

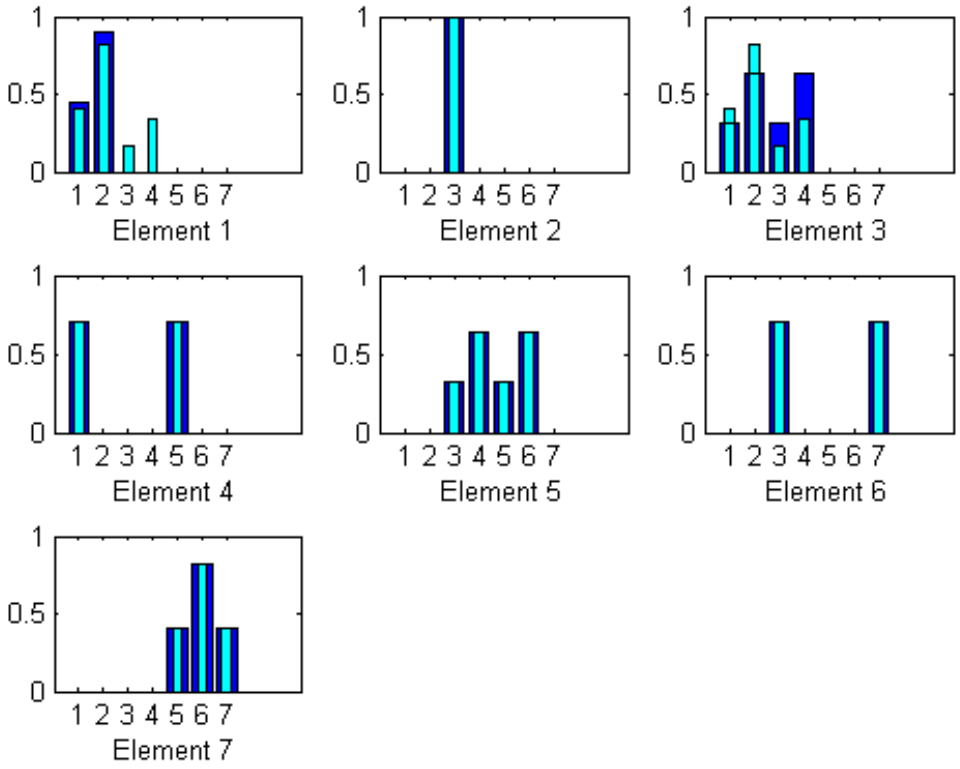

Figure 17: Case 4 Damaged Simple Truss DPDREM and MMEDR Singular Vectors

3.2.5 Discussion

The four cases above show the capability of the DPDREM and MMEDR methods in locating the full DOF damaged elements with reduced sensor sets. Both methods use the same inputs and obtained similar results. It could be argued that the DPDREM results are slightly better due to the simplicity of the method. The DPDREM method simplifies the selection process by comparing the absolute values and utilizing the dot product function rather than the "best subspace" algorithm. The DPDREM also seems less prone to errors that may occur with sign differences in the damage residual singular vectors since the absolute value is taken for those vectors.

The second item to note from the results is that the BVP, KE, and EI sensor sets can be tailored to focus on certain elements by shifting the actuator locations for the load dependent Ritz vectors. It is important to note that this does not take place with mode

shapes as they are not altered by the actuator location as with the Ritz vectors. The Ritz vectors can be adjusted to focus on selected elements of interest using an optimization method. This would be useful for complex structures where certain elements need to be monitored with a limited sensor set.

3.3 Example 3: NASA Eight-Bay Truss

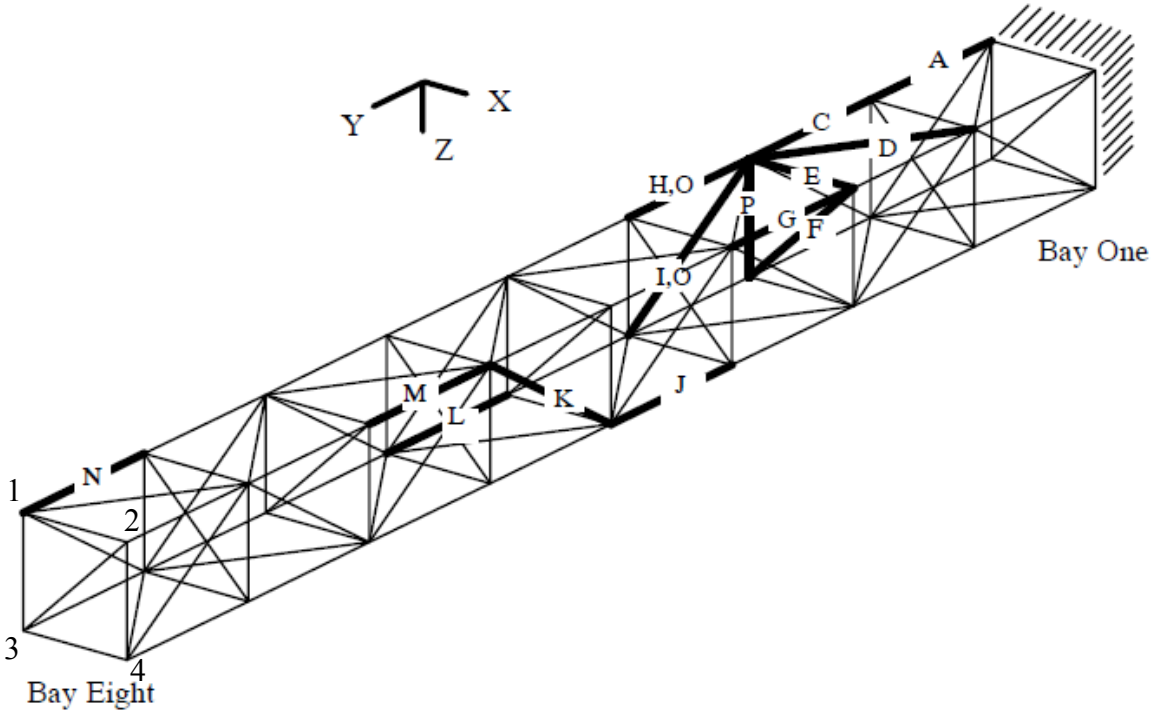

Figure 18: NASA Eight-Bay Truss with damage cases

The third example investigates the eight-bay hybrid-scaled truss structure designed for research in a dynamic scale model ground testing of large space structures at the NASA Langley Research Center [38]. The truss was modeled as an un-damped cantilevered 96-DOF structure. Each truss strut was modeled as a rod element. Concentrated masses were attached at each node to represent joint and instrumentation mass properties. The structure has been used in a complete analytical and experimental analysis to generate a realistic test-bed for structural damage location and extent

algorithms [39]. A schematic of the truss with the first four node numbers and different damage cases highlighted is shown in Figure 18.

Damage cases A through N, as shown in Figure 18, consist of the full removal of one member from the truss. Damage case O consists of the full removal of two members and damage case P consists of a buckled member to illustrate a partial damage scenario.

The examples presented in this work are based on the analytical models rather than the experimental results. This is due to the fact that only modal information was available from the test-bed. The analytical stiffness matrices were utilized to determine the Ritz vectors with selected force locations. Since the DPDREM and MMEDR results were determined to be very similar, only the DPDREM was used for these examples. The sensor locations were determined using the EI method.

3.3.1 Damage Case L Detection with DPDREM using EI Sensor Set

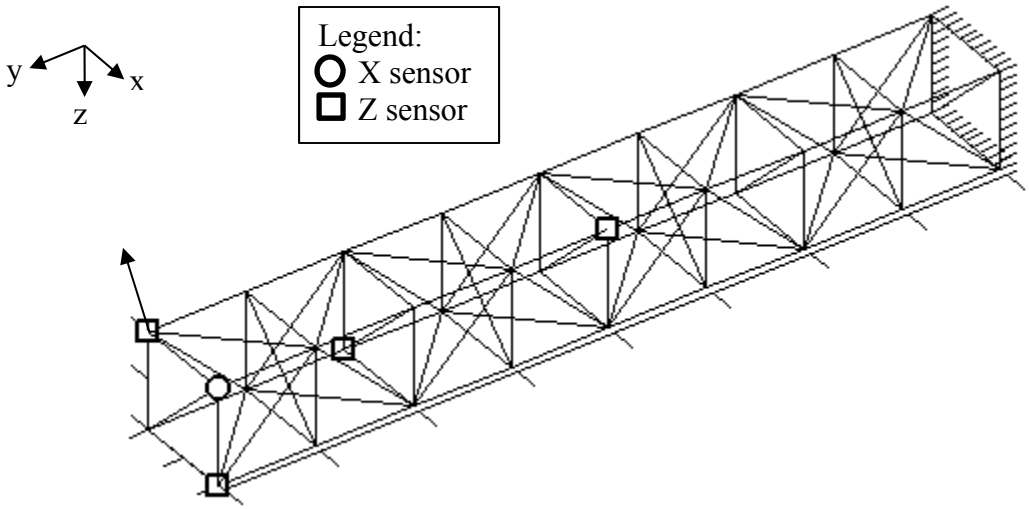

Figure 19: EI sensor set with force at node 1

Damage case L includes the full removal of element 21 at DOFs 44 and 56. The Ritz vectors were found by placing a force on node 1 with an 80º angle in the x-z plane,

shown as an arrow in Figure 19. The EI method located five sensors at DOFs 3, 4, 12, 33, and 54.

Using the reduced sensor set, the DPDREM results determined two potential damage locations at element 22 between DOFs 44 and 56 as well as element 31 between DOFs 56 and 68. As can be seen in Figure 20 below, the damage was correctly identified, however further investigation would be required to differentiate the identical signatures of elements 22 and 33.

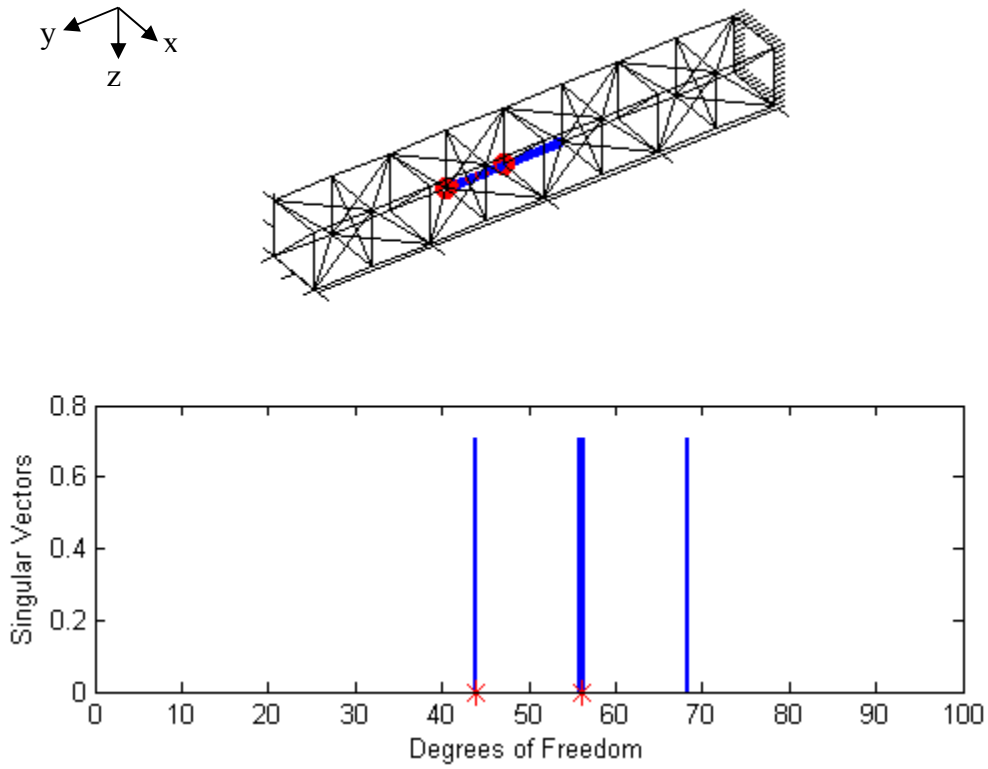

Figure 20: Damage Case L DPDREM using EI sensor set with force at node 1

In the top portion of Figure 20, the damaged element is shown with circles at the associated DOFs, and the elements found with DPDREM are shown in thick lines. In the bottom portion, the damaged DOFs are identified by stars, and the selected damage

residual singular vectors found are shown as bars. To further investigate the two selected elements the reduced singular vectors were plotted and are shown in Figure 21.

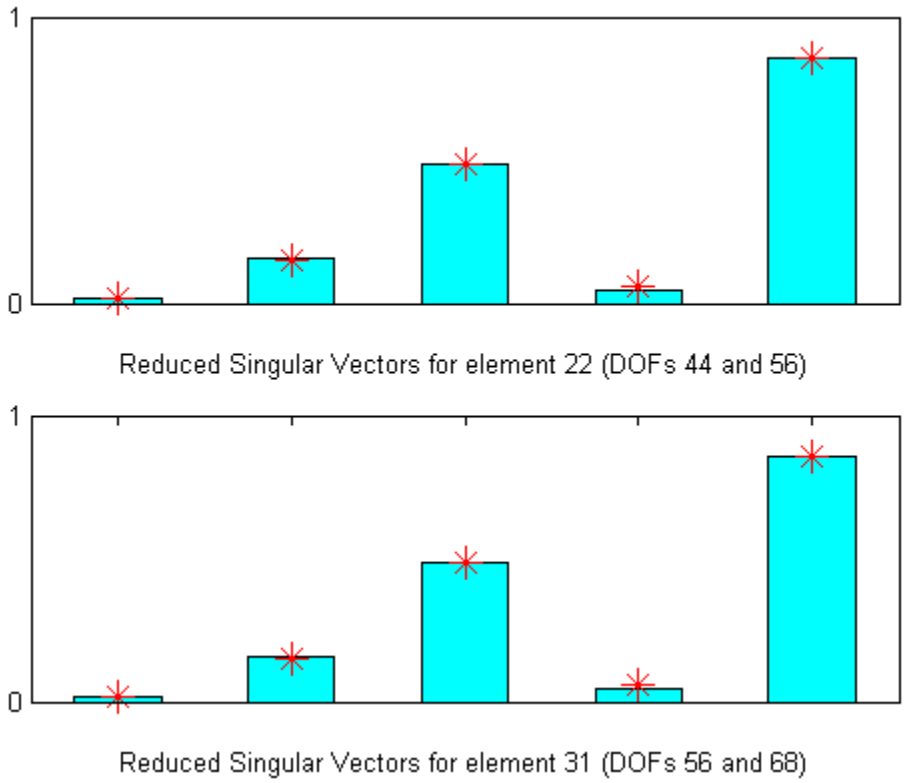

Figure 21: Reduced singular vectors for elements 22, 31 at DOFs 3, 4, 12, 33, 54

Figure 21 above shows the actual damage residual singular vectors as stars and the reduced singular vectors from elements 22 and 31 as bars. As expected, both reduced singular vectors from the two elements are nearly identical to the reduced damage residual singular vectors. Most of the contribution to the damage detection comes from the bar on the far right, which corresponds to element 54, while the least contribution comes from the first bar on the far left corresponding to DOF 3. Since both elements contain DOF 56, it would make a good sensor location for the damage case. In this case, it would be best to replace the sensor at DOF 3 with one at DOF 56. The reduced singular value results with the revised sensor set are shown in Figure 22.

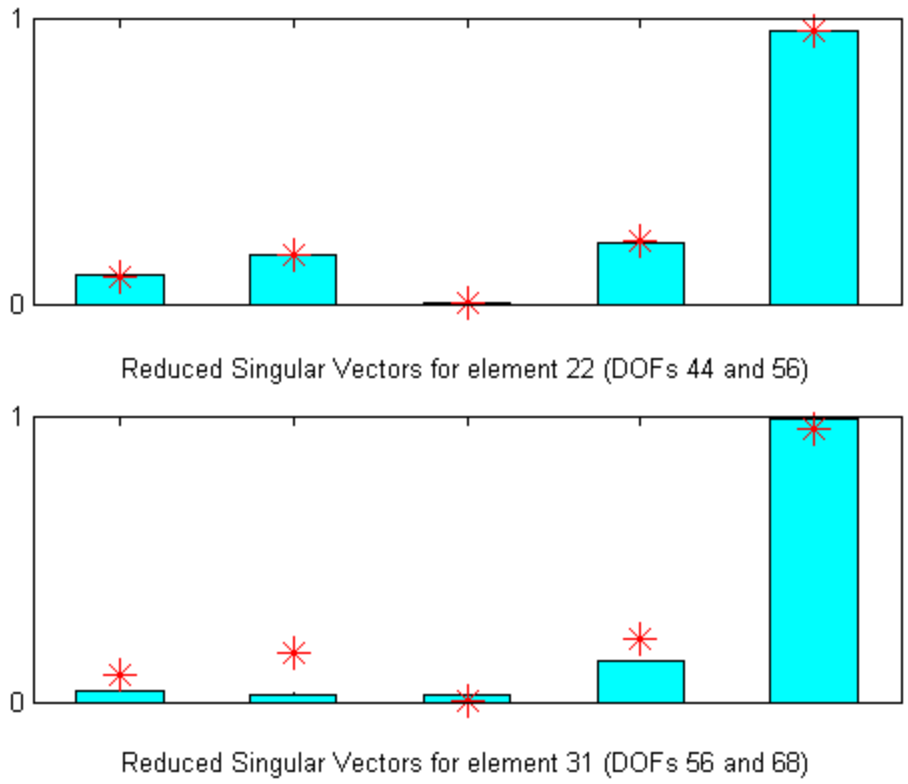

Figure 22: Reduced singular vectors for elements 22, 31 at DOFs 4, 12, 33, 54, 56

As expected, the singular vector bar on the far right of Figure 22, corresponding to DOF 56, was found to be much higher than the rest. Additionally, the new sensor set clearly distinguished the reduced damage residual singular vector results between the two elements with the bars for element 22 found to be nearly identical to the stars representing the actual singular vectors. On the other hand, the element 31 damage residual singular vectors deviated from the actual damage. Figure 23 shows the DPDREM results for damage case L with the new sensor set.

The results in Figure 23 show the damage detection correctly identified the damaged element for case L with the revised sensor set. This confirms the singular value results from Figure 22.

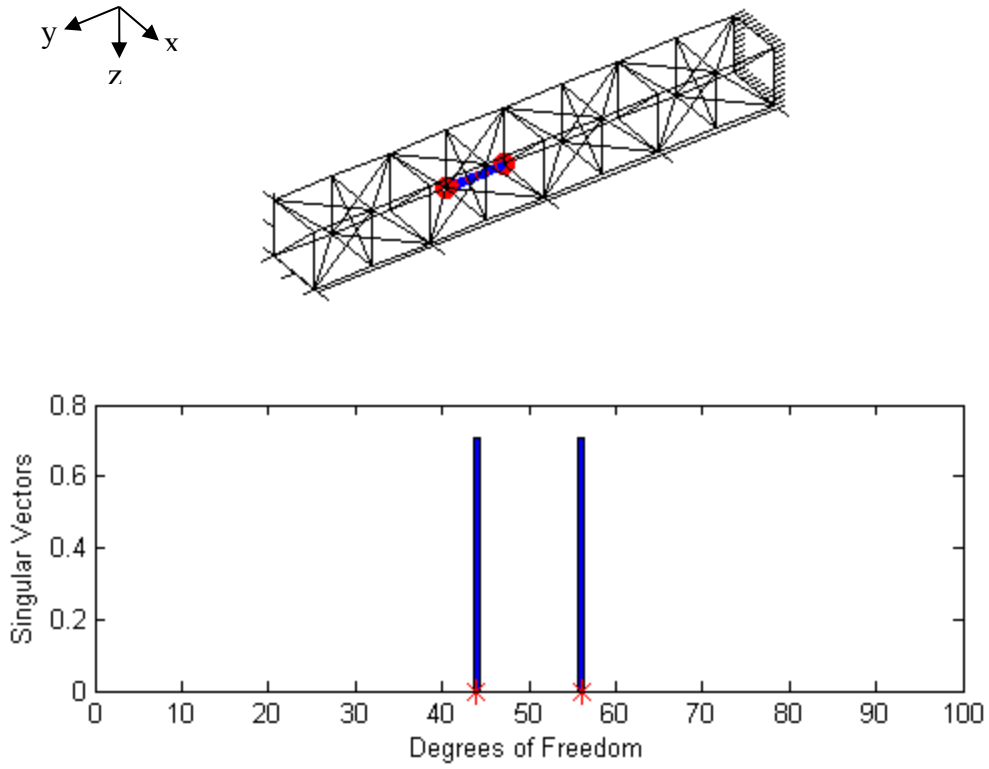

Figure 23: Damage Case L with DPDREM using the revised sensor set

3.3.2 Damage Case I Detection with DPDREM using EI Sensor Set

Damage case I consists of the full removal of element 99 at DOFs 68, 69, 74, and 75. The damage case is used to illustrate the difference in results when the actuator location is changed. Initially, the Ritz vectors were found by placing a force at node 1 with an 80º angle in the x-z plane as done in the case L example. The EI method located the five sensors at DOFs 3, 4, 12, 33, and 54 as shown in Figure 19. The DPDREM damage detection results are shown in Figure 24, which shows the DPDREM results using the EI sensor set determined using a force at node 1 does not correctly locate the damaged element.

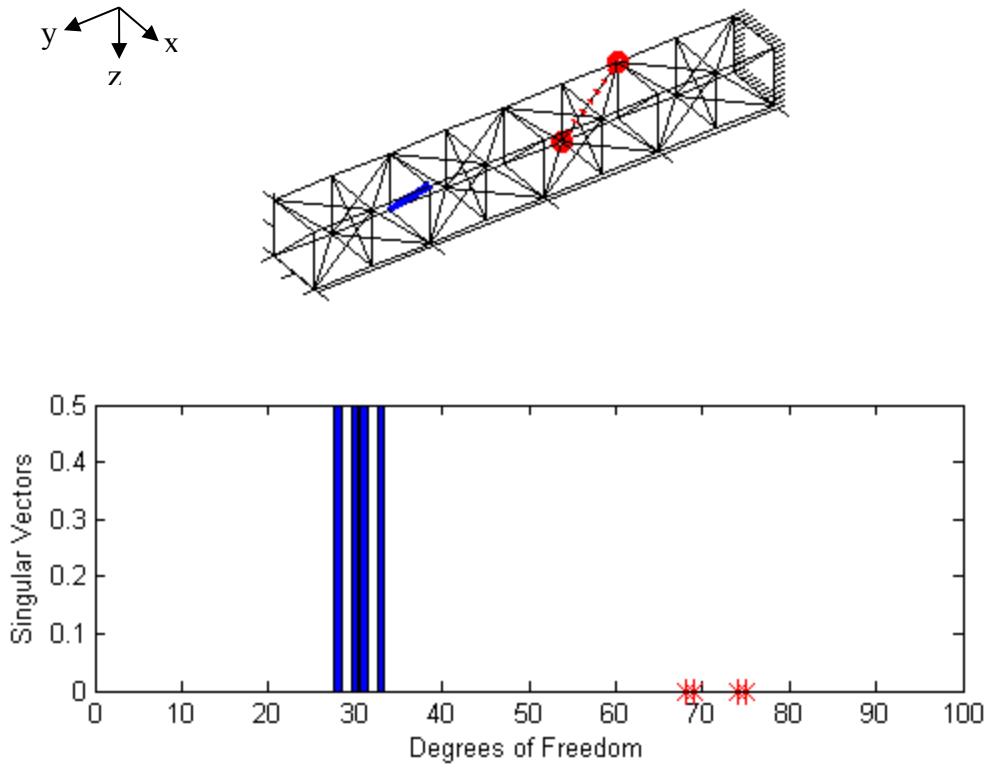

Figure 24: Damage Case I with DPDREM using EI sensor set with force at node 1

In an attempt to improve the results, the force location was moved to node 23 with an 80º angle in the x-z plane. With the new force location, the EI method placed the sensors at DOFs 6, 7, 9, 48, and 69 as shown in Figure 25. Note that the sensor at DOF 69 falls directly on one of the damaged element DOFs. The DPDREM results are shown in Figure 26.

As shown in Figure 26, the damaged element was located exactly using the new sensor set determined with the force at node 23. The findings confirm the sensor set variations obtained by modifying the force locations can attain different results, which in this case helped improve the detectability of case I.

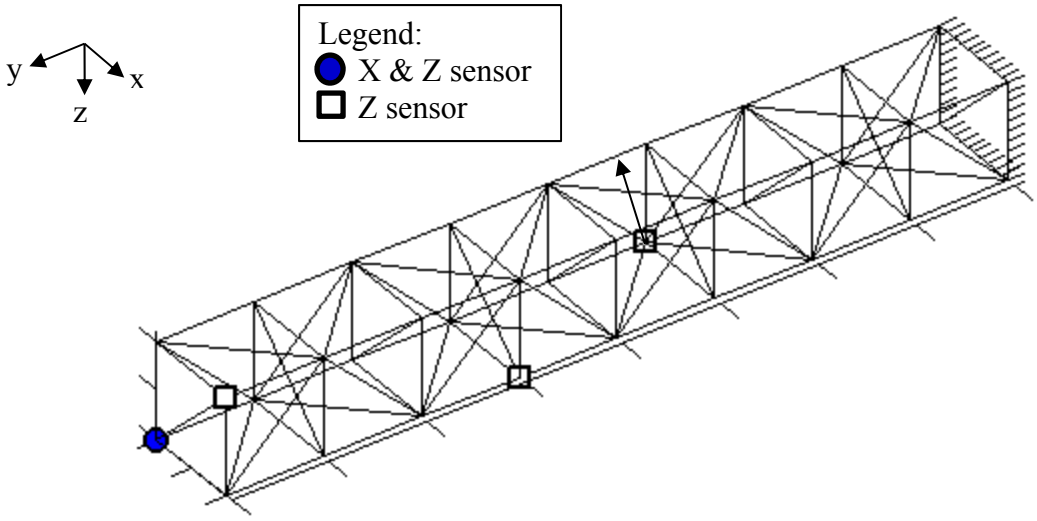

Figure 25: EI sensor set with force at node 23

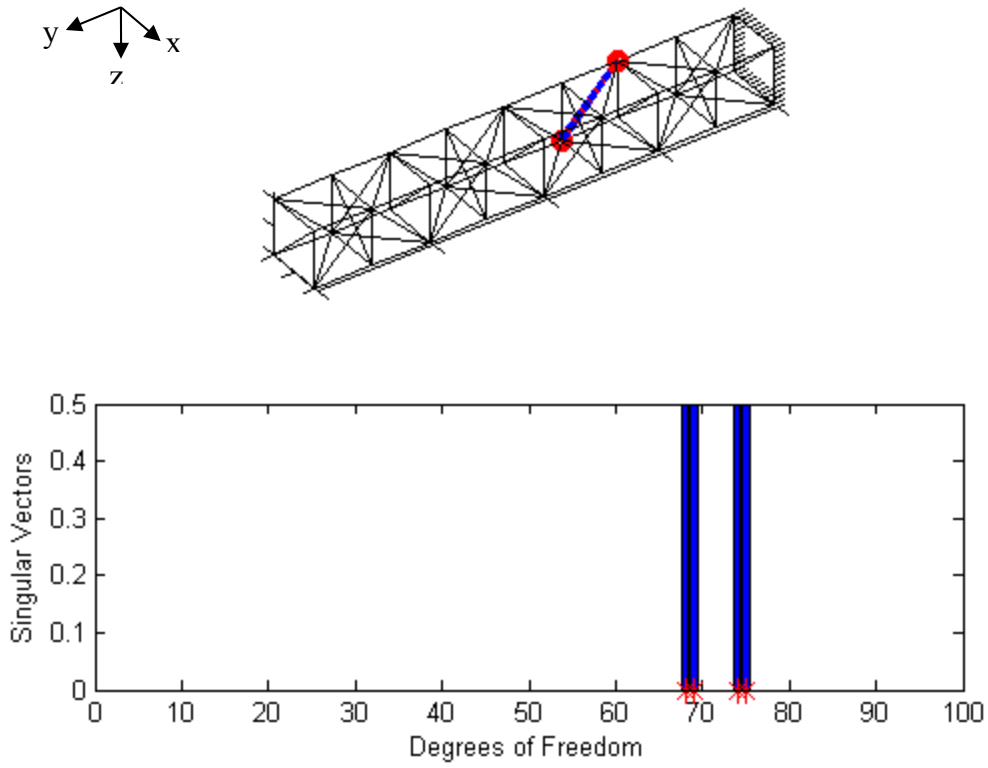

Figure 26: Damage Case I with DPDREM using EI sensor set and force at node 23

3.3.3 Discussion

The results above show that the DPDREM is capable of determining the damage locations for complex and repetitive structures, like the NASA eight-bay truss, with measurements from a relatively smaller number of sensors than the total present in the finite element model. The method was capable of locating the damage for both horizontal elements with two DOFs affected and diagonal elements where four DOFs were affected. Most damage cases presented in Figure 18 of this example can be determined using the same techniques as shown in Sections 3.3.1 and 3.3.2.

As shown in the Section 3.2 simple truss example, the selected actuator location plays a major role in finding different Ritz vector sensor sets, which can help improve damage detection results. Additionally, the example for damage case L shows that, by examining the reduced singular vectors, the sensor sets can be manually modified to better locate damaged elements and avoid identical signatures. When examining the reduced damage residual singular vectors, the DOFs with the lowest magnitudes corresponded to sensors with the least contribution. Replacing the low contributing sensors with DOFs found from the initial DPDREM run improved the detection results. Most cases for the eight-bay truss had identical signatures for two elements that were generally connected by a DOF. The exact damaged element can be found by replacing a low contributing sensor with the connected DOF and re-performing the DPDREM.

The cases that could not be determined as presented above were E, F, P, and O. For cases E, F, and P, the reduced basis vectors were not excited as a result of their removal. Of course, since only five sensors were used in these examples, additional sensors would undoubtedly improve the results. It should be noted that case F could not

be found using the MEDR or MMEDR even with a full sensor set because the singular vectors of the target and damaged vectors cancel each other out in equations (58) and (68) reducing to a near-negligible value. The DPDREM is capable of determining the damage since absolute values are taken in equation (73).

Damage case O includes two affected elements, which is essentially an equal combination of cases H and I as shown in Figure 18. Since the two elements are connected by DOF 68, the damage detection becomes more complex because the damaged singular vectors are combined and normalized. The first diagonal element at DOFs 68, 69, 74, and 75 was found with a 5 DOF sensor set. However, once the new target vector was defined, the reduced sensor set could not detect a contribution from DOF 62, which in turn led to an inability to find the second element correctly. The second element could be found when additional sensors were used.

The damage detection results could be further improved by using optimization methods that would determine the best Ritz vector forcing functions and subsequent sensor locations. In that case, it may be possible to locate the damaged elements with fewer sensors. This would be especially useful for the more difficult cases such as E, F, P, and O.

4 Conclusions and Suggestions for Future Work

This study investigates pre-test planning when faced with the second part of the incomplete measurement problem where fewer degrees-of-freedom (DOFs) are measured than that of the analytical model. Sensor and actuator placement techniques were used with two variations of the Method of Expanded Dynamic Residuals (MEDR) designated as the Modified Method of Expanded Dynamic Residuals (MMEDR) and the Dot Product Damage Residual Expansion Method (DPDREM).

The damage residual singular vectors obtained using mode shapes and load dependent Ritz vectors were first compared using a simple spring mass system model. While both basis vectors obtained good representations of the actual damage residual projection, the Ritz vector damage residuals were found to be slightly less susceptible to noise that would falsely identify damage compared to the mode shape damage residuals.

The simple truss structure example demonstrated that the different sensor placement techniques could be used to vary the sensor sets. Additionally, the Ritz vectors could be modified by changing the load locations, angles, or both. This in turn altered the subsequent sensor sets and reduced damage residuals in a manner that could focus the damage detection results for different elements of interest.

The MMEDR and DPDREM were determined to be effective methods to locate the structural damage using reduced sensor sets. The elemental disassembly improved the damage detection results by using a set of vectors that better represent each structural element. The dot product method helped simplify the damaged element selection process from the elemental information matrix. Both the DPDREM and MMEDR obtained similar results, but it could be argued that the DPDREM was less susceptible to incorrect

damage identification since the absolute values of the damage residual singular vectors are utilized. The study also showed that identical signatures can be mitigated by either moving sensors with low contributions or varying the system forcing to find better Ritz vector sensor sets.

Optimization techniques, such as Genetic Algorithms (GAs), can be used in future studies to improve the forcing direction and magnitude determination, which will aid in focusing the MMEDR and DPDREM results to selected elements of interest. GAs are a form of directed random search first introduced by Holland [40]. GAs combine the survival of the fittest with a structured yet randomized search. GAs efficiently exploit historical information to speculate on new search points. Each new generation will introduce a set of 'offspring' using information from the fittest 'parents', while an occasional new part is also introduced to the generations for good measure. Goldberg later presented GAs as search algorithms based on the mathematics of natural selection and natural genetics in his work [41].

GAs have been used in numerous applications and studies. A study by Zimmerman investigated the use of GAs to optimize the placement of actuators by developing an algorithm that would cycle through candidate configurations [42]. The method was capable of incorporating the effects of actuator mass in the system.

Future studies could also focus on the investigation of multiple elemental damage cases. The current methods presented in this work determine a new target vector after finding the first damaged element. When reducing the actual damage residual singular vectors to the sensor set DOFs, it can be difficult to replicate the experimental damage residual singular vectors, especially when the damaged elements are connected. One

suggestion would be to keep the target vector unchanged and combine the selected damage residual singular vectors as they are selected in a way to add up to the target vector. These two improvements would further advance and enhance pre-test planning methodologies using Ritz vectors.

References

[1] Doebling, S.W., Farrar, C.R., Prime, M.B., and Shevitz, D.W., 1996, "Damage Identification and Health Monitoring of Structural and Mechanical Systems from Changes in Their Vibration Characteristics: A Literature Review", *Los Alamos National Laboratory Report LA-13070-MS*.

[2] Hoon, S., Farrar, C.R., Hemez, F.M., Shunk, D.W., Stinemates. D.W., Nadler, B.R., and Czarnecki, J.J., 2004, "A Review of Structural Health Monitoring Literature: 1996–2001", *Los Alamos National Laboratory Report LA-13976-MS*

[3] Fan, W. and Qiao, P., 2011, "Vibration-based Damage Identification Methods: A Review and Comparative Study", *Structural Health Monitoring, 10 (1), pp. 83-111.*

[4] Lifshitz, J.M. and Rotem, A., 1969, "Determination of Reinforcement Unbonding of Composites by a Vibration Technique," *Journal of Composite Materials*, 3, pp. 412–423.

[5] Farrar, C.R., Baker, W.E., Bell, T.M., Cone, K.M., Darling, T.W., Duffey, T.A., Eklund, A., and Migliori, A., 1994, "Dynamic Characterization and Damage Detection in the I-40 Bridge Over the Rio Grande", *Los Alamos National Laboratory Report LA-12767-MS*.

[6] West, W.M., 1984, "Illustration of the Use of Modal Assurance Criterion to Detect Structural Changes in an Orbiter Test Specimen", *Proceedings of Air Force Conference on Aircraft Structural Integrity*, pp. 1–6.

[7] Baruch, M. and Bar Itzhack I.Y., 1978, "Optimum Weighted Orthogonalization of Measured Modes," *AIAA Journal*, 16 (4), pp. 346–351.

[8] Berman, A. and Nagy, E.J., 1983, "Improvement of Large Analytical Model Using Test Data," *AIAA Journal*, 21 (8) pp. 1168–1173.

[9] Kabe, A.M., 1985, "Stiffness Matrix Adjustment Using Mode Data," *AIAA Journal*, 23 (9), 1431–1436.

[10] Chen, J.C. and Garba, J.A., 1988, "On-Orbit Damage Assessment for Large Space Structures," *AIAA Journal*, 26 (9), 1119–1126.

[11] Kammer, D.C., 1988, "Optimal Approximation for Residual Stiffness in Linear System Identification," *AIAA Journal*, 26 (1), 104–112.

[12] Kaouk, M. and Zimmerman, D.C., 1994, "Structural Damage Detection Using a Minimum Rank Update Theory," *ASME Journal of Vibration and Acoustics*, April 1994, Vol. 116, No. 2, pp. 222-231.

[13] Kaouk, M. and Zimmerman, D.C., 1994, "Structural Damage Assessment Using a Generalized Minimum Rank Perturbation Theory," *AIAA Journal*, 32 (4), pp. 836–842.

[14] Cao, T. and Zimmerman, D.C., 1997, "Application of Load Dependent Ritz Vectors in Structural Damage Detection," *Proceedings of SPIE, the International Society for Optical Engineering*, 3089 (2), pp. 1319-1324.

[15] Wilson, E.L., Yuan, M.W. and Dicken, J.M., 1982, "Dynamic Analysis by Direct Superposition of Ritz Vectors," *Earthquake Engineering and Structural Dynamics*, 10, pp. 813-821.

[16] Nour-Omid, B. and Clough, R.W., 1984, "Dynamic Analysis of Structures Using Lanczos Coordinates," *Earthquake Engineering and Structural Dynamics*, 12, pp. 565-577.

[17] Burton, T. D., Farrar, C. R., and Doebling, S. W., 1998, "Two Methods for Model Updating Using Damage Ritz Vectors," *Proceedings of the 16th International Modal Analysis Conference*, Santa Barbara, pp. 973-979.

[18] Craig, R.R, Jr., Kim, H.M., and Su, T.J, 1988, "Some Applications of Lanczos Vectors in Structural Dynamics," *Proceedings of the 6th International Modal Analysis Conference*, pp. 501-506.

[19] Sohn, H. and Law, K.H., 2001, "Damage Diagnosis Using Experimental Ritz Vectors," *Journal of Engineering Mechanics*, 127 (11), pp. 1184-1193.

[20] Sohn, H. and Law, K.H., 2000, "Application of Load-Dependent Ritz Vectors to Bayesian Probabilistic Damage Detection," Probab. Eng. Mech., 15 (2), pp. 139–153.

[21] Cao, T. and Zimmerman, D.C., 1999, "Procedure to Extract Ritz Vectors From Dynamic Testing Data," *ASCE Journal of Structural Engineering*, 125 (12), pp. 1393–1400.

[22] Juang, J.N. and Pappa, R.S., 1985, "An Eigensystem Realization Algorithm for modal parameter identification and model reduction," *Journal of Guidance Control and Dynamics* (1985) Volume: 8, Issue: 5, pp. 620-627.

[23] Sohn, H. and Law, K.H., 2001, "Extraction of Ritz Vectors from Vibration Test Data," *Mechanical Systems and Signal Processing*, 15 (1), pp. 213-226.

[24] Boxoen, T. and Zimmerman, D.C., 2003, "Advances in Experimental Ritz Vector Identification," *ASCE Journal of Structural Engineering*, Volume 129 Issue 8, pp. 1131-1140.

[25] Taylor, S.G. and Zimmerman, D.C., 2010, "Improved Experimental Ritz Vector Extraction with Application to Damage Detection," *Journal of Vibration and Acoustics*, Volume 132, Issue: 1, 011012 (10 pages).

[26] Zimmerman, D.C., Smith, S.W., Kim, H.M., and Bartkowicz, T.J., 1996, ''An Experimental Study Of Structural Damage Detection Using Incomplete Measurements,'' *Journal of Vibration and Acoustics*, Volume 118, Issue: 4, Oct., pp. 543–550.

[27] Zimmerman, D.C., Kim H.M., Bartkowicz, T. J., and Kaouk, M., 2001, "Damage Detection Using Expanded Dynamic Residuals," *Journal of Dynamic Systems, Measurement, and Control,* **123**(4), pp. 699-705.

[28] Doebling, S.W., Peterson, L.D., and Alvin, K.F., 1998, "Experimental Determination of Local Structural Stiffness by Disassembly of Measured Flexibility Matrices," Journal of Vibration and Acoustics, Volume 120, Issue: 4, Oct., pp. 949–957.

[29] Taylor, S.G. and Zimmerman, D.C., 2007, "Structural Health Monitoring Using Ritz Vectors and Expanded Residuals," *Proceedings of the SEM International Modal Analysis Conference*, February, Paper #106.

[30] Kammer, D.C., 1991, "Sensor Placement for On-Orbit Modal Identification and Correlation of Large Space Structures," *Journal of Guidance, Control, and Dynamics*, Volume 15, Issue 2, pp.251-259.

[31] Larson, C.B., Zimmerman, D.C., and Marek, E.L, 1994, "A Comparison of Modal Test Planning Techniques: Excitation and Sensor Placement Using the NASA 8-

Bay Truss," *Proceedings of the 12th International Modal Analysis Conference*, pp. 205-211.

[32] Yap, K.C. and Zimmerman, D.C., 2000, "Optimal Sensor Placement for Dynamic Model Correlation," *Proceedings of the 18th International Modal Analysis Conference*, Volume 4062 Issue 2, pp. 607-612.

[33] Schedlinski, C. and Link, M., 1996, "An Approach to Optimal Pick-Up and Exciter Placement," *Proceedings of the 14th International Modal Analysis Conference*, Dearborn, MI, pp. 376-382.

[34] Golub, G.H. and Van Loan, C.F., 1996, Matrix Computations. 3rd ed. The Johns Hopkins University Press, Baltimore, MD.

[35] Khoury, G.C. and Zimmerman, D.C., 2011, "Development of Pretest Planning Methodologies for Load Dependent Ritz Vectors," *Proceedings of the SEM International Modal Analysis Conference*, February, Paper #11.

[36] Guyan, R. J., 1965, "Reduction of Stiffness and Mass Matrices," *AIAA Journal*, **3**.

[37] Li, D.S., Li, H.N, and Fritzen, C.P., 2009, "A note on fast computation of effective independence through QR downdating for sensor placement," *Mechanical Systems and Signal Processing*, Volume 23, Issue: 4, pp. 1160-1168.

[38] McGowan, P.E., Edighoffer, H.H., and Wallace J.W., 1989, "Development of an Experimental Space Station Model for Structural Dynamics Research," *Proceedings of the 60th Shock and Vibration Symposium*, Virginia Beach, VA.

[39] Kashangaki, T.A.L., 1992, "Ground Vibration Tests of a High Fidelity Truss for Verification of On Orbit Damage Location Techniques," *NASA-LaRC, Technical Memorandum 107626*.

[40] Holland, J.H., 1975, Adaption in Natural and Artificial Systems. The University of Michigan Press, Ann Arbor, MI.

[41] Goldberg, E.G., 1989, Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley, Reading, MA.

[42] Zimmerman, D.C., 1993, "A Darwinian Approach to the Actuator Number and Placement Problem with Non-negligible Actuator Mass," *Mechanical Systems and Signal Processing*, Volume 7, Issue: 4, pp. 363-374.

Appendix

Appendix A: Ritz Vector M-file

```
function [X,Xn] = ritz(M,K,f,n)
%
% function [X,Xn] = ritz(M,K,f,n)
%
% Generates orthogonal Ritz vectors [X]
%
% Updated 7/1/10 changed mass orthogonalization to unit
orthogonalization
%
% Input:
% M = Mass Matrix
% K = Stiffness Matrix
% f = load vector
% n = Number of Ritz Vectors to output
%
% Output:
% X  = Orthogonlized Ritz Vectors
% Xn = Non-Orthogonlized Ritz Vectors
%
% Reference: Dynamic Analysis by Direct Superposition of Ritz Vectors
%            Earthquake Engineering and Structural Dynamics, Vol 10,
%            813-821 (1982)
if n==[]
    n=length(K);
end

x(:,1) = K\f;                    %First Ritz Vector - Static response
to f.
xn(:,1)=x(:,1);

d = x(:,1)'*x(:,1);              %Unit Normalization
x(:,1) = (x(:,1)/sqrt(d));

for i = 2:n
    x(:,i) = K\(M*x(:,i-1));     %In experimentation, Ritz vectors are
found without the mass matrix.

    xn(:,i) = x(:,i);

    m=0;                         %Unit Orthogonalization
    for y = 1:i-1
    c(y) = (x(:,y)'*x(:,i))/(x(:,y)'*x(:,y));
    m = m + c(y)*x(:,y);
    end
    x(:,i) = x(:,i) - m;

    d = x(:,i)'*x(:,i);           %Unit Normalization
    x(:,i) = (x(:,i)/sqrt(d));

end
```

```
X = x;
Xn = xn;
```

Appendix B: Basis Vector Product

```
function [P,I] = evp(v,dim,num)
%
% function [P,I] = evp(v,dim,num)
%
% Generates the optimal sensor/actuator locations
% using the Vector Product
% Updated on 4/21/10 - see notes
%
% Input:
% v = Eigenvector or Ritz Vector
% dim = 2 for 2D, 3 for 3D
% num = select number or range of vectors to use. For all vectors use:
0.
%
% Output: In descending order from best to worst locations
% P = Basis Vector Products
% I = Degrees of freedom

% Select the Basis Vectors to use
if length(num)==1
if num == 0 % To use all basis vectors
     n = 1:length(v(1,:));
else n = 1:num; % To use a selected number of basis vectors
end
else n = num; % To use a range of basis vectors
end

% Multiply the Basis Vector components
P = (v(:,n(1)).*v(:,n(2)));
P = P/sqrt(P'*P);
for ii = n(3:end)
    P = (P.*v(:,ii));
    P = P/sqrt(P'*P);
end

% Sort the Basis Vector Products
[P,I] = sort(abs(P),1,'descend');
```

Appendix C: Kinetic Energy Product

```
function [ke,I] = ken(v,M,dim,N,num)
%
% function [ke,I] = ken(v,M,dim,N,num)
%
% Generates the optimal sensor/actuator locations
% using the Kenetic Energy Method
%
% Input:
% v   = Eigenvector or Ritz Vector
```

```
% M   = Mass Matrix
% dim = 2 for 2D
%       3 for 3D
% N   = 1 for Kinetic Energy
%       2 for Average Kinetic Energy
% num = Select number or range of vectors to use. For all vectors use:
0.
%
% Output: In descending order from best to worst locations
% ke  = Kenetic Energy Product
% I   = Degrees of freedom

% Select the Basis Vectors to use
if length(num)==1
if num == 0 % To use all basis vectors
     n = 1:length(v(1,:));
else n = 1:num; % To use a selected number of basis vectors
end
else n = num; % To use a range of basis vectors
end

k=1;
for l = n(1:end)% Determine the System Kinetic Energy
KE(:,k)=v(:,l).*(M*v(:,l));
k=k+1;
end

if N == 1 % To obtain the Kinetic Energy Product
ke = (KE(:,1).*KE(:,2));
ke = ke/sqrt(ke'*ke);
for ii = 3:length(n)
    ke = (ke.*KE(:,ii));
    ke = ke/sqrt(ke'*ke);
end
end

if N == 2 % To obtain the Average Kinetic Energy Product (not used)
 for ii = 1:length(n)
     ke(ii,:) = sum(KE(ii,:))/length(n);
 end
end

% Sort the Kinetic Energy Product
[ke,I] = sort(abs(ke),1,'descend');
```

Appendix D: Effective Independence Product

```
function [E,I] = efi2(v,dim,num,N)
%
% function [E,I] = efi2(v,dim,num,N)
%
% Generates the optimal sensor/actuator locations
% using the Effective Independence Method - Method C from Ref.
% Ref: A Note on Fast Computation of Effective
%      Independence through QR Downdating for Sensor Placement
```

```
%
% Input:
% v   = Eigenvector or Ritz Vector
% dim = 2 for 2D
%       3 for 3D
% num = select number or range of vectors to use. For all vectors use:
0.
% N   = select number of sensor locations to obtain.
%       Generally, N > num. Will be badly scaled if N < num. See QR
%       decomposition limitations.
%
% Output: In descending order from best to worst locations
% E  = Effective Independence Product
%      0 = the sensor can be deleted with no impact to the target
modes.
%      1 = the sensor is vital to the independence of the target modes.
% I   = Degrees of freedom

% Select the Basis Vectors to use
if length(num)==1
if num == 0 % To use all basis vectors
     n = 1:length(v(1,:));
else n = 1:num; % To use a selected number of basis vectors
end
else n = num; % To use a range of basis vectors
end

%Find the Fisher Information Matrix
[Q1,R1] = qr(v(:,n));
Q = Q1(:,n);

%Solve for the Effective Independence
E = diag(Q*Q');

[E,I1] = sort(abs(E),1,'descend');
j=1;
I = 1:length(v);I=I';
I1x=I1;

%Iterative loop to remove least useful sensor locations
while length(E) > N

    E3(1:length(E),j) = E(:,1);
    I3(1:length(I),j) = I1x(:,1);
    k(j) = I1(end);
    clear E I1

I(k(j))=[];
v(k(j),:) = [];
j = j+1;

%Find the Fisher Information Matrix
[Q1,R1] = qr(v(:,n));
Q = Q1(:,n);
```

```
%Solve for the Effective Independence
E = diag(Q*Q');
[E,I1] = sort(abs(E),1,'descend');
I1x = I(I1,:);

end
I = I(I1,:);
```

Appendix E: Minimum Rank Perturbation Theory

```
function [dk,B,u,s] = mrpt(K,M,z,e,rank,option)
%
% function [dk,B,u,s] = mrpt(K,M,z,e,rank,option)
%
% Minimum Rank Perturbation Theory
%
% Input:
% K      = Healthy Stiffness Matrix
% M      = Healthy Mass Matrix
% z      = Damaged Modeshapes / Ritz Vectors (i=1:n-1)
% e      = Damaged Natrual Frequencies (in rad/sec) diagonalized and
squared / Non-orthogonalized Ritz Vectors (i=2:n)
% rank   = Rank of the update
% option = 1 for Mode Shapes / 2 for Ritz Vectors
%
% Output:
% df     = Stiffness perturbation matrix
% B      = Damage residuals
% u      = Damage residual singular vectors
% s      = Singular values (used to find the rank input)

if option == 1
    B = K*z-M*z*e;
elseif option == 2
    z = z(:,1:end-1);
    e = e(:,2:end);
    B = K*e-M*z;
end

[u,s,c] = svd(B);
u = u(:,[1:rank]);
Y = pinv(B)*u;

if option == 1     % Using Mode Shapes
dk = B*Y*inv(Y'*B'*z*Y)*Y'*B';
elseif option == 2 % Using Ritz Vectors
dk = B*Y*inv(Y'*B'*e*Y)*Y'*B';
end
```

Appendix F: Modified Method of Expanded Dynamic Residuals (MMEDR)

```
function [uf,Usel,utar,e,I,ddof,su,npr] = mmedr(ut,ufa,T,mpass,plot)
%
% function [uf,Usel,utar,e,I,ddof,su,npr] = mmedr(ut,ufa,T,mpass,plot)
%
```

```
% Modified Method of Expanded Dynamic Residuals:
% Performs the final process involved in completing the MMEDR.
% Identifies important columns of ufa and estimates the full damage
vector.
%
% Input:
% ut   = Target vector
% ufa  = Actual Full damage vectors
% T    = Transformation matrix
% mpass= Maximum number of iterations allowed (i.e. max rank of damage)
% plot = select 1 to plot damage locations
%
% Output:
% uf   = Full damage vector for the required number of passes
% Usel = Full damage vectors for each pass
% utar = Reduced damage residual target vectors
% e    = Sorted Dot Product Values (Max to Min) for each pass
% I    = Index Matrix of sorted dot product values for each pass
% ddof = Selected Damaged location DOFs for each pass
% su   = Summary of results with a matrix for each pass. First columns
are
%        normalized e values, second columns are the e values,
remaining
%        columns are the corresponding DOFs.
% npr  = Number of passes required

%% Identify Important Columns of ufa
alpha = [];
ex = zeros(1,1);
utar=ut(:,1);
pass=1;lpass = 1;upass = 1;%Needed to determine the number of passes
required.
for iv=1:mpass
j=1;
tc = (T'*ufa); % Reduced actual damage residual vectors
for i = 1:length(ufa(1,:)) % "Best subspace" algorithm
    if abs(tc(:,i))<eps
        B(i)=0;              %Keep B from going to NAN
    else
        tc(:,i) = tc(:,i)/norm(tc(:,i));
        B(i) = (utar(:,iv)'*tc(:,i))/(tc(:,i)'*tc(:,i));
    end
er(iv,i) = (utar(:,iv)-B(i)*tc(:,i))'*(utar(:,iv)-B(i)*tc(:,i));
end
[en(iv,:),In(iv,:)] = sort(abs(er(iv,:))); % Sort the 'error' values
e(:,iv)=en(iv,:)';I(:,iv)=In(iv,:)'; % Switch from rows to columns

% Display the top columns of ufa
if length(ufa(:,1))>10
n=10; % Limits the number of columns to 10
else
n = min(length(ufa(:,1)),length(ufa(1,:)));
end
e1(:,iv) = e(:,iv)/min(e(:,iv)); % Sets the min error value at 1

% Display the top damage locations
```

```
ddof(:,:,iv) = zeros(6,n);
for iii=1:n
k1(iii,iv) = length(find(ufa(:,I(iii,iv))));
ddof(j:j-1+k1(iii,iv),iii,iv) = find(ufa(:,I(iii,iv)));
end
j=max(k1(:,iv))+1;
su(:,:,iv) = [e1(1:n,iv),e(1:n,iv),ddof(:,:,iv)'];

% Find important columns of ufa based on e. Loop will combine values of
e that are very similar.
clear Usel1
k(iv) = 0;
    while abs(e(1,iv)-e(k(iv)+1,iv))<1e-10;
        k(iv) = k(iv)+1;
        Usel1(:,k(iv)) = ufa(:,I(k(iv),iv));
    end
   Usel{iv}=Usel1;

% Show the 'error' for each selected column of ufa
ex(end+1:end+k(iv)) = e(1:k(iv),iv);
% Remove first value of ex (always equals 0)
if ex(1)==0
ex(1)=[];
end
ef(iv) = ex(end);

% Update and normalize the target vector based on the selected column
of ufa
utar(:,iv+1) = utar(:,1)-B(I(1,iv))*tc(:,I(1,iv));
utar(:,iv+1) = utar(:,iv+1)/norm(utar(:,iv+1));

%% Estimate Full Damage Vector
la = length(alpha);
for ii = 1:k(iv)
alpha(:,ii+la) = pinv(T'*Usel1(:,ii))*utar(:,iv);
uf(:,ii+la) = Usel1(:,ii)*alpha(:,ii+la);
end

% Finds the number of passes required based on the percent difference
from the consecutive error values obtained.
percent_req = 70;
if length(ef)>1
lpass = lpass+1;
for iiv=lpass:upass
per(iiv) = abs(ef(iiv)-ef(1))/max(ef(1),ef(iiv));
    if per(iiv) < percent_req/100;
       pass=pass+1;
    end
end
end
upass = upass+1;

% Keep only required values based on above loop
len = sum(k(1:pass));
uf = uf(:,1:len);uf = (abs(uf));% Full damage residual
em = ex(1:len);
```

```
% Loop combines all columns of uf and normalizes
if length(uf(1,:))>1
    for ik = 1:length(uf(1,:))
    temp(ik)=norm(uf(:,ik));
    uf(:,ik)=uf(:,ik)/temp(ik);
    end
else
    uf = uf/norm(uf);
end
end
if plot == 1 % Optional bar plot of uf
figure
bar(abs(uf))
end

utar(:,iv+1:end) = []; % Remove last column of reduced damage residual
target vector
npr = pass;

%Display only the damage residuals based on number of passes needed.
uf = uf(:,1:len);
```

Appendix G: Dot Product Damage Residual Expansion Method (DPDREM)

```
function [uf,Usel,utar,e,I,ddof,su,npr] = dpdrem(ut,ufa,T,mpass,plot)
%
% function [uf,Usel,utar,e,I,ddof,su,npr] = dpdrem(ut,ufa,T,mpass,plot)
%
% Dot Product Damage Residual Expansion Method (DPDREM)
% Determines the important columns of ufa and estimates the full target
% vector.
%
% Input:
% ut    = Target vector
% ufa   = Actual Full damage vectors
% T     = Transformation matrix
% mpass = Maximum number of iterations allowed (i.e. max rank of
damage)
% plot  = select 1 to plot damage locations
%
% Output:
% uf   = Full damage vector
% Usel = Full damage vectors for each pass
% utar = Reduced damage residual target vectors
% e    = Sorted Dot Product Values (Max to Min)
% I    = Index Matrix of sorted dot product values
% ddof = Selected Damaged location DOFs
% su   = Summary of results with a matrix for each pass. First columns
are
%        normalized e values, second columns are the e values,
remaining
%        columns are the corresponding DOFs.
% npr  = Number of passes required
```

```
alpha = [];
ex = zeros(1,1);
utar=ut(:,1);
pass=1;lpass = 1;upass = 1;%Needed to determine the number of passes
required.
for iv=1:mpass
j=1;
udar = T'*ufa; % Reduced actual damage residual vectors
for i = 1:length(ufa(1,:)) % Dot Product maximization problem loop
    if abs(udar(:,i))<eps
     udar(:,i)=0; % Keeps udar from going to NAN
     else
    udar(:,i)=udar(:,i)/norm(udar(:,i));
    end
    d(iv,i) = abs(dot(abs(utar(:,iv)),abs(udar(:,i))));
end
[en(iv,:),In(iv,:)]=sort(d(iv,:),'descend');
e(:,iv)=en(iv,:)';I(:,iv)=In(iv,:)'; % Switch from rows to columns

% Display the top columns of ufa
if length(ufa(:,1))>10
n=10; % Limits the number of columns to 10
else
n = min(length(ufa(:,1)),length(ufa(1,:)));
end
e1(:,iv) = e(:,iv)/max(e(:,iv)); % Sets the max dot product value at 1

% Display the top damage locations
ddof(:,:,iv) = zeros(6,n);
for iii=1:n
k1(iii,iv) = length(find(ufa(:,I(iii,iv))));
ddof(j:j-1+k1(iii,iv),iii,iv) = find(ufa(:,I(iii,iv)));
end
j=max(k1(:,iv))+1;
su(:,:,iv) = [e1(1:n,iv),e(1:n,iv),ddof(:,:,iv)'];

% Find important columns of ufa based on e. Loop will combine values of
e that are similar.
clear Usel1
k(iv) = 0;
    while abs(e(1,iv)-e(k(iv)+1,iv))<1e-10;
        k(iv) = k(iv)+1;
        Usel1(:,k(iv)) = ufa(:,I(k(iv),iv));
    end
     Usel{iv}=Usel1;

% Show the top dot product for each selected column of ufa
ex(end+1:end+k(iv)) = e(1:k(iv),iv);
% Remove first value of ex (always equals 0)
if ex(1)==0
ex(1)=[];
end
ef(iv) = ex(end);

% Update and normalize the target vector based on the selected columns
of ufa
```

```
utar(:,iv+1)=abs(utar(:,iv))-abs(e(1,iv)*udar(:,I(1,iv)));
utar(:,iv+1) = utar(:,iv+1)/norm(utar(:,iv+1));
% Estimate Full Damage Vector
la = length(alpha);
for ii = 1:k(iv)
alpha(:,ii+la) = pinv(T'*Usel1(:,ii))*utar(:,iv);
uf(:,ii+la) = Usel1(:,ii).*alpha(:,ii+la)';
end

% Finds the number of passes required based on the percent difference
from the consecutive dot product values obtained.
percent_req = 10;
if length(ef)>1
    lpass = lpass+1;
for iiv=lpass:upass
per(iiv) = abs(ef(iiv)-ef(1))/max(ef(1),ef(iiv));
    if per(iiv) < percent_req/100;
       pass=pass+1;
    end
end
end
upass = upass+1;

% Keep only required values based on above loop
len = sum(k(1:pass));
uf = uf(:,1:len);uf = (abs(uf));% Full damage residual
em = ex(1:len);

% Loop normalizes uf columns
if length(uf(1,:))>1
    for ik = 1:length(uf(1,:))
    temp(ik)=norm(uf(:,ik));
    uf(:,ik)=uf(:,ik)/temp(ik);
    end
else
    uf = uf/norm(uf);
end
end
if plot == 1 % Optional bar plot of uf
figure
bar(abs(uf))
end
utar(:,end) = []; % Remove last column of reduced damage residual
target vector
npr = pass;

%Display only the damage residuals based on number of passes needed.
uf = uf(:,1:len);
```

Appendix H: Example 2, Case 1, Element 3 Damage Scenario

```
% Example 2, Case 1, Element 3 Damage

clc
clear
```

```
close all

delem = '3'; % Damaged element number
set = 3; % Select 1 for VP, 2 for KE, 3 for EI.

ndof = 4; % Number of sensors used.
nritz= 3; % Number of ritz vectors used in VP, KE, or EI
rank = 1; % Rank of damage used
f = [0 0 0 0 100 100 0]'; % Forcing funciton

% Load stiffness and mass matrices obtained using dtruss3d

% Healthy stiffness matrix
kh = [5702637.68087986,0,-751318.840439929,1502637.68087986,...
-4200000,0,0;0,6010550.72351943,1502637.68087986,...
-3005275.36175972,0,0,0;-751318.840439929,1502637.68087986,...
9902637.68087986,0,-751318.840439929,-1502637.68087986,...
-4200000;1502637.68087986, -3005275.36175972,0,6010550.72351943,...
-1502637.68087986,-3005275.36175972, 0;-4200000,0,-751318.840439929,...
-1502637.68087986,5702637.68087986,0,-751318.840439929;0,0,...
-1502637.68087986,-3005275.36175972,0,6010550.72351943,...
1502637.68087986;0,0,-4200000,0,-751318.840439929,1502637.68087986,...
4951318.84043993;];

% Mass matrix
mh = [6.35078340584334,0,0,0,0,0,0;...
      0,6.35078340584334,0,0,0,0,0;...
      0,0,8.31328340584334,0,0,0,0;...
      0,0,0,8.31328340584334,0,0,0;...
      0,0,0,0,6.35078340584334,0,0;...
      0,0,0,0,0,6.35078340584334,0;...
      0,0,0,0,0,0,4.15664170292167;];

% Elemental Stiffness Information Matrix
uca = [-0.447213595499958,0,-0.316227766016838,-0.707106781186547,...
0,0,0;-0.894427190999916,0,0.632455532033676,0,0,0,0;...
0,1, 0.316227766016838,0,-0.316227766016838,-0.707106781186547,0;...
0,0,-0.632455532033676,0,-0.632455532033676,0,0;0,0,0,...
0.707106781186547,0.316227766016838,0,-0.408248290463863;0,0,0,0,...
0.632455532033676,0,0.816496580927725;0,0,0,0,0,0.707106781186547,...
0.408248290463864];

% Damaged stiffness matrix (with noise)
kdn = [5053116.15922250,728779.275226731,-349363.260804567,...
698726.521609134,-3990000,0,0;728779.275226731,4252464.63689000,...
698726.521609134,-1397453.04321827,0,0,0;-349363.260804567,...
698726.521609134,9485655.72443570,773858.405653127,...
-736292.463631131,-1472584.92726226,-3906000;698726.521609134,...
-1397453.04321827,773858.405653127,4342622.89774279,...
-1472584.92726226,-2945169.85452452,0;-3990000,0,-736292.463631131,...
-1472584.92726226,5515177.24609306,-105184.637661590,...
-788884.782461926;0,0,-1472584.92726226,-2945169.85452452,...
-105184.637661590,6100708.98437223,1577769.56492385;...
0,0,-3906000,0,-788884.782461926,1577769.56492385,4694884.78246193];
```

```
[rh,rhn] = ritz(mh,kh,f,nritz);
if set == 1 % Basis Vector Product
[Ps,Is] = evp(rh,2,nritz);
elseif set == 2% Kenetic Energy Product
[Ps,Is] = ken(rh,mh,2,1,nritz);
elseif set == 3% Effective Independence
[Ps,Is] = efi2(rh,2,nritz,ndof);
end
dofkeep = sort(Is(1:ndof))'; % Sensor set used

%Obtain the reduced damage residual signular vector.
[rd,rdn] = ritz(mh,kdn,f,length(mh)); % Full Damage Ritz vectors
rdr = rd(dofkeep',1:ndof);rdnr = rdn(dofkeep',1:ndof); % Reduced damage
Ritz vectors
[khr,mhr,Dh,dofrm]=guyanmr(kh,mh,dofkeep); % Reduce healthy mass and
stiffness matrix to use in mrpt
[dkr,Br] = mrpt(khr,mhr,rdr,rdnr,rank,2); % Obtain the reduced damage
residual
[ur,sr]=svd(Br); % Find the damage residual singular vectors and values

% Create Transformation Matrix
T=zeros(length(mh),length(dofkeep));
T(dofkeep,:)=eye(length(dofkeep)); % Set ones at DOFs where sensors are
located
T(dofrm,:)=Dh; % Set the transformed values where there are no sensors

[uf,Usel,utar,e,I,ddof,su] = dpdrem(abs(ur),uca,T,rank,0);
[ufm,Uselm,utarm,em,Im,ddofm,sum] = mmedr(ur,uca,T,rank,0);

% Re-sort e values to show in DOF order
for i = 1:7
e1(I(i))=e(i);
em1(Im(i))=em(i);
end
e1=e1';em1=em1';

% Plot Results
figure(1)
bar(abs(uca(:,str2num(delem))),'stacked','b')
hold on
bar(uf,.4)
colormap spring
hold off

figure(2)
bar(abs(uca(:,str2num(delem))),'stacked','b')
hold on
bar(ufm,.4)
colormap spring
hold off
su
sum
```