

Sparsity-based approaches for damage detection in plates

Debarshi Sen^a, Amirali Aghazadeh^b, Ali Mousavi^b, Satish Nagarajaiah^{a,c,*}, Richard Baraniuk^b

^a*Department of Civil and Environmental Engineering, Rice University, Houston, TX 77005, USA*

^b*Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005, USA*

^c*Department of Mechanical Engineering, Rice University, Houston, TX 77005, USA*

Abstract

The data deluge in Structural Health Monitoring (SHM) and the need for automated online damage detection systems necessitates a move away from traditional model-based approaches. To that end, we propose sparsity-based algorithms for damage detection in plates. Instead of high-fidelity models, our proposed algorithms use *ictionaries*, consisting of response signals acquired directly from the system of interest, as the key feature to both detect and localize damages. We address the damage detection problem both when the damage is located *on* or *off* a grid of possible damage coordinates defined by the dictionary. This gives rise to two classes of problems, namely, *on the grid* and *off the grid* problems. In our sparsity-based on the grid damage detection (SDD-ON) platform, we solve a LASSO regression problem, where, the unknown vector is a pointer for existence of damage at the various locations defined on the grid used for dictionary construction. In our proposed off the grid damage detection (SDD-OFF) platform, we use a penalized regression algorithm to extend the dictionary of measured damage signals to points off-the-grid by linear interpolation. We evaluate the performance of both SDD frameworks, in detecting damages on plates, using finite element simulations as well as laboratory experiments involving a pitch-catch setup using a single actuator-sensor pair. Our results suggest that the proposed algorithms perform damage detection in plates efficiently. We obtain area under receiver operating characteristic (ROC) curves of 0.997 and 0.8314 for SDD-ON and SDD-OFF, respectively.

Keywords:

Structural Health Monitoring, Damage Detection, Wave Propagation, Sparsity

1. Introduction

Structural Health Monitoring (SHM) is an essential component of maintenance for engineering systems. The research community has contributed vastly to this field over the past three decades, aiming to improve the efficiency and accuracy of detection and localization of damages in engineered systems [1, 2]. In the past few years, significant strides have been made in data transmission and acquisition systems [3] for developing efficient SHM systems [4, 5]. These improvements have led to a massive increase in the volume of data collected from engineered systems. This further opens the possibility of using more data, instead of detailed models, for damage detection [6].

Traditionally, a class of SHM approaches involve the development of detailed structural models for studying the behavior of structural systems in both pristine and damaged conditions [7]. One performs damage detection using features, extracted from the models, that best define the changes in the behavior of systems from a pristine to damaged condition. However, modeling complicated systems is challenging. Capturing the physics of a system accurately in terms of behavior of all degrees of freedom involved and modeling of specialized boundary conditions pose a significant obstacle to a designer. Although, creating such high fidelity models is possible using advanced simulation softwares, analyzing them becomes computationally prohibitive. For example, in this study, the Guided Ultrasonic Waves (GUW)-based damage detection is used. The advantages of GUW-based damage detection [8–10] over traditional vibration-based techniques [11] is well-addressed in the literature [12], with the key advantage

*Corresponding author

Email addresses: debarshi.sen@rice.edu (Debarshi Sen), amirali@rice.edu (Amirali Aghazadeh), ali.mousavi@rice.edu (Ali Mousavi), satish.nagarajaiah@rice.edu (Satish Nagarajaiah), richb@rice.edu (Richard Baraniuk)

being its ability to detect, localize and quantify minute damages. In spite of these advantages, from an experimental standpoint, it requires a heavy computational effort for simulating the waves propagating in a structural system. Gopalakrishnan [13] discusses the various computational issues involved in modeling wave propagation problems using the finite element method.

Recently, the applications of statistical learning algorithms for damage detection, has received significant attention [6, 14–16]. In such approaches, statistical learning algorithms are used to construct surrogate models [17] that overcome the need for developing high-fidelity models of systems. Statistical learning algorithms are well equipped for handling large data sets of signals generated by a dense array of sensors deployed on a system. Support Vector Machines (SVM) [18] and Artificial Neural Networks (ANN) [19] are the most widely used data-intensive approaches for damage detection using guided waves. A review of such methods can be found in Raghavan and Cesnik [10]. However, these techniques require extensive training for damage classification and localization. In addition, the amount of training data vastly increases for achieving high resolution in damage localization. In this paper, we propose platforms, that harness the inherent sparsity in the damage detection and localization problem and achieve this by minimal a priori data acquisition.

We propose sparsity-based detection algorithms, that avoid modeling of the real system at hand. Here, we construct a damage location indicator vector \mathbf{x} . Under the assumption that the number of possible damage locations, at a certain instant of time in a system, is small, the vector \mathbf{x} will have very few non-zero elements. A k -sparse vector \mathbf{x} is defined as a vector of length n with k nonzero entries, where $k \ll n$. We associate a matrix, known as the *dictionary*, with the damage location indicator vector \mathbf{x} and call it the Damage Characterization Matrix (DCM). As shown in figure 1, the DCM consists of signals acquired from the identical system with changing damage locations. Since, DCM captures the behavior of the system for different locations of damage, it alleviates the need for high fidelity models representing the system. Existing sparse representation-based methods still use a physics-based wave propagation model for dictionary construction [20, 21]. However, we circumvent the need of such models by directly using acquired data for constructing the dictionary. The data reflects the dynamics of the system as well as possible changes due to presence of a damage. From a system identification point of view, the inherent physics involved in the elements of the DCM ensure that the proposed framework is not a black-box method, but closer to a gray-box approach.

Clearly the efficacy of the proposed dictionary-based approaches depend on the efficiency of the DCM. Two possibilities arise; the first is when a grid of locations are selected such that they represent the most probable locations of damage in the system. The second is when the number of possible damage locations is large enough rendering the construction of the DCM impossible with all the locations. Although refining the DCM would seem to be an alternative, the effort necessary for construction of such a detailed DCM would be prohibitive. This leads to a branching of the sparsity based damage detection problem into two sub-classes, namely, *on the grid* and *off the grid* problems.

For certain systems, *a priori* knowledge about the system can help in predicting the most probable locations of damage. Hence, for such systems, a few damage locations constituting a grid is sufficient for capturing the damaged behavior. The DCM is then constructed using damage responses from these most probable damage locations only. Hence, when a damaged signal with unknown location is acquired for damage detection purposes, the assumption enforces that the unknown damage location belongs to the set of these most probable locations for the given system. We define this class of damage detection problems as an on-the-grid problem. We propose SDD-ON for solving this class of problems.

In most SHM applications, however, the resolution of the grid is not typically high enough to cover all possible damage scenarios owing to the space of possible damage locations being continuous. In these scenarios, detecting damages located off the grid becomes extremely important, since in real field damage detection scenarios it is highly probable that the damage occurs in between the grid points. The off-the-grid damage profiles may differ drastically from the profiles of damages located on on-the-grid coordinates due to inherent complexity of the system at hand. We define this class of damage detection problems as an off-the-grid problem. We propose SDD-OFF for solving this class of problems.

We use both Finite Element Method (FEM)-based simulations and experiments for demonstrating the performance of the proposed platforms in dealing with on-the-grid and off-the-grid problems in damage detection, wherein, we use circular holes as a form of damage. For experimental verification, we use a mass scatterer to simulate the damage in the system [22].

2. SDD-ON: Sparse Regression for *on the grid* Damage Detection

In this section we propose the SDD-ON platform for damage detection. As discussed earlier, SDD-ON is a sparsity-based algorithm. Figure 1 shows the basic idea behind formulating damage detection as a sparsity-based problem. The matrix \mathbf{A} is the dictionary, or damage characterization matrix (DCM), for the given system. The columns of matrix \mathbf{A} are signals retrieved from the experiments for different damage locations in the system when subjected to a specific actuation signal. We perform a baseline subtraction (we subtract an undamaged signal from all the acquired signals) on all these signals before they are added to the DCM. Assuming that each signal is of length n , and that there are p possible damage locations, the matrix $\mathbf{A} \in \mathbb{R}^{n \times p}$. The vector \mathbf{y} is a test vector, that is a background subtracted response retrieved from the system for an unknown damage location(s) when subjected to the same actuation signal used for construction of the DCM, hence $\mathbf{y} \in \mathbb{R}^{n \times 1}$. $\mathbf{x} \in \mathbb{R}^{p \times 1}$ is a damage location pointer vector. Assuming that the DCM contains signals from all possible damage locations and the number of damages is low, the vector \mathbf{x} will be sparse, *i.e* only a few non zero elements. As discussed earlier, mathematically, we define a k -sparse vector of length n as one with k non-zero elements such that $k \ll n$. To be more precise, if the test vector was obtained from a single damage scenario, only one element in \mathbf{x} will be nonzero. The location of the nonzero element will correspond to the column of the DCM that is similar to the damage signal of the test vector \mathbf{y} . When \mathbf{y} is from an undamaged signal, post background subtraction $\mathbf{y} = 0$, that would render $\mathbf{x} = 0$. For a multiple damage case, there will be multiple non zero elements in \mathbf{x} . This assumption holds only when the damages are small and the effects of multiple damages can be linearly combined. In addition, the number of damages must be limited such that the vector \mathbf{x} is still sparse.

The readers should also note that, in the present formulation we look at different damage locations as elements of the DCM. In reality, damages in a system maybe of various sizes and orientations. These may affect the acquired signals in very different ways. Hence, in this study, we limit ourselves to the task of damage detection and localization. Another issue that needs a discussion is the case for multiple damages. For example, if the test signal is a signal acquired from a scenario where we have damage at locations 1 and 2. In that case we assume $\mathbf{y} = \mathbf{a}_1 + \mathbf{a}_2$, where \mathbf{a}_1 and \mathbf{a}_2 are the first and second columns of the DCM \mathbf{A} . This assumption holds only for small damages [21]. This works because the scattered waves from small damages are typically an order of magnitude lower than the incident signals. These scattered waves, when scattered by other damages produce signals that are another order of magnitude lower than the incident signal. Such low amplitude signals are typically lost in noise and may be safely neglected. For larger damages the amplitudes may not be insignificant and the assumption of linearity will not hold.

Given this framework, one needs to solve for \mathbf{x} from the linear system of equations $\mathbf{y} = \mathbf{Ax}$, under the assumption that \mathbf{x} is sparse. This problem is formulated as an optimization problem as follows [23]

$$\underset{\mathbf{x}}{\text{minimize}} \|\mathbf{x}\|_0 \text{ such that } \mathbf{y} = \mathbf{Ax}, \quad (1)$$

where $\|\cdot\|_0$ is the ℓ_0 -norm that counts the number of nonzero elements in a vector. The problem defined in equation (1) is however both numerically unstable and NP complete in addition to being non-convex. Solving it entails enumeration of all $\binom{p}{k}$ possible locations of nonzero entries of \mathbf{x} , if \mathbf{x} is k -sparse [24]. To overcome these issues, equation (1) can be reformulated in an ℓ_1 -norm minimization framework (also referred to as the relaxation of ℓ_0 norm to ℓ_1 norm), generally referred to as a *basis pursuit* problem [23]

$$\underset{\mathbf{x}}{\text{minimize}} \|\mathbf{x}\|_1 \text{ such that } \mathbf{y} = \mathbf{Ax}, \quad (2)$$

where $\|\cdot\|_1$ is the ℓ_1 -norm defined as $\|\mathbf{v}\|_1 = \sum_{i=1}^n |v_i|$ for any vector $\mathbf{v} \in \mathbb{R}^{n \times 1}$, where $|\cdot|$ computes the absolute value of its argument. The computational complexity of the above problem is $\mathcal{O}(p^3)$. Equation 2 can be further modified to involve the effects of noise as follows

$$\underset{\mathbf{x}}{\text{minimize}} \|\mathbf{x}\|_1 \text{ such that } \|\mathbf{y} - \mathbf{Ax}\|_2 \leq \epsilon, \quad (3)$$

where ϵ is a measure of the noise. This framework has been used earlier for various classes of problems over a wide spectrum of engineering fields like face recognition [25] and microbial diagnostics [26]. The ℓ_1 minimization framework has also been used for vibration based damage detection purposes earlier [27–29]. We extend the application of

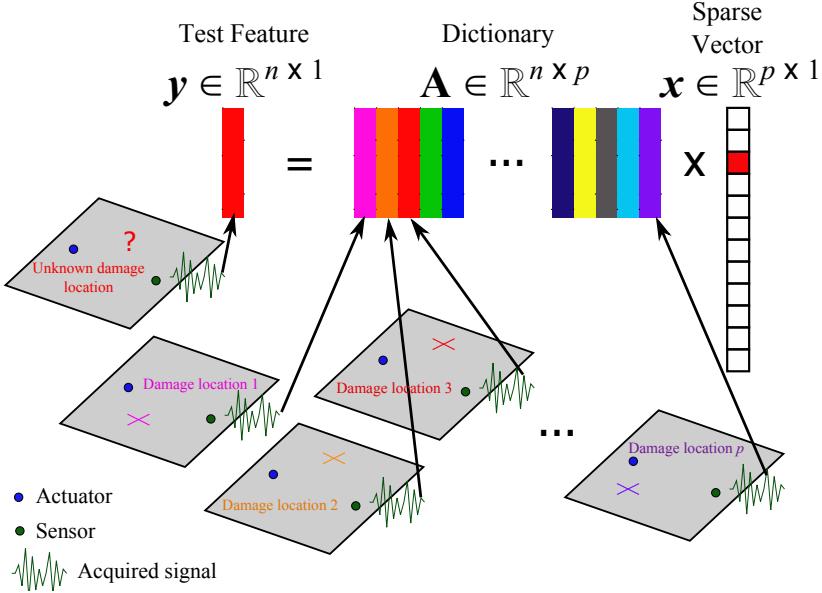


Figure 1: Illustration of SDD-ON for damage detection. Each element of the dictionary is constructed using processed acquired signals from different damage locations as shown. In this case the unknown damage location, from the test vector \mathbf{y} , is location 3. Hence, the third element of the sparse vector \mathbf{x} , evaluated by sparse regression, points to location 3 in the dictionary \mathbf{A} (DCM).

this framework to high frequency ultrasonic wave based damage detection. Pursuit algorithms are used for obtaining solutions to equations (2) and (3). The alternative is to reformulate the problem as a sparse regression problem solved by the LASSO (a regularized linear regression technique used for sparse regression) [30] technique as follows

$$\underset{\mathbf{x}}{\text{minimize}} \|\mathbf{y} - \mathbf{Ax}\|_2^2 + \lambda \|\mathbf{x}\|_1, \quad \lambda > 0, \quad (4)$$

where $\|\cdot\|_2$ is the ℓ_2 -norm and λ is the regularization parameter governing how sparse vector \mathbf{x} should be. This can be looked at as a Lagrange multiplier form of equation (3). A detailed description of the LASSO can be found in [31]. In this paper, we use the `l1ls` solver developed by [32] for LASSO implementation.

Based on the above discussion, it becomes clear that, for an effective implementation of the algorithm described in figure 1, an inherent assumption about the system is made: the damage locations used for dictionary construction are sufficient for effectively detecting and localizing damage of the system. As discussed earlier, when the test vector \mathbf{y} is from a damage location already in the DCM we will use the nomenclature on-the-grid problem, which essentially points that the test vectors are already on the DCM grid. SDD-ON addresses this problem. As stated earlier, an extension of the on-the-grid problem is the off-the-grid problem. In that case, it is not necessary for test signals to be obtained from dictionary grid points. The performance of SDD-ON both on- and off-the-grid have been evaluated.

3. SDD-OFF: a statistical learning approach for *off the grid* damage detection

We propose a novel statistical learning based platform to solve the off-the-grid problem. Our SDD-OFF platform consists of a training and a test phase. In the training phase, SDD-OFF collects a finite number of damage profiles by placing damages in different locations on the structure. We then form an ℓ_1 -penalized least squares optimization problem that learns a linear model that predicts the location of the damage given a query signal. In the test phase, SDD-OFF maps the measured query signal to the location of the damages using the linear model obtained from the previous stage.

In the training phase, we measure p damage profiles \mathbf{b}_i for $i = 1, 2, 3, \dots, p$ at p different damage locations on the surface of the structure, for example (but not necessarily) on a predefined grid, and store the time domain signals \mathbf{b}_i

of length n in an $p \times n$ matrix \mathbf{B} . We also store the d -dimensional locations of the damages \mathbf{l}_i for $i = 1, 2, 3, \dots, p$ in a $p \times d$ matrix \mathbf{L} , where d determines the dimension of the geometric structure, i.e., $d = 1$ for a beam, $d = 2$ for a plate, and $d = 3$ for a pipe or any 3-d structure. In this case, since we are dealing with plates, $d = 2$. Given a damage response profile b measured from a damaged structure, we are interested to recover (estimate) the damage location vector y using the set of training samples.

In the SDD-OFF algorithm, we assume that the damage profiles \mathbf{b}_i linearly map to coordinates of the damage locations \mathbf{l}_i on the d -dimensional structure, i.e., we assume the damage profiles are linearly linked to their location coordinates through a regression parameter matrix β of size $n \times d$ such that $\mathbf{L} = \mathbf{B}\beta$. Our goal is to use the training data and learn the regression matrix β . Following the classical regression problems in statistics, one can estimate β by minimizing the reconstruction error via the following least squares optimization problem:

$$\underset{\beta}{\text{minimize}} \|\mathbf{L} - \mathbf{B}\beta\|_F^2.$$

The Frobenius norm $\|\cdot\|_F^2$ averages the reconstruction error of the signal profiles over the d coordinates of the structure. We use the Frobenius norm instead of the popular Euclidean norm because, the argument $\mathbf{L} - \mathbf{B}\beta$ is a matrix, and the Euclidean norm is only defined for a vector. In fact, the problem is completely separable for each coordinate and can be solved separately. However, the main challenge in learning β is that this optimization problem is highly underdetermined. The number of unknowns, n , for each dimension of the structure corresponds to the length of the time signal that is typically large (couple of thousands) in SHM applications [33] and hence is much higher than the number of training data points. To robustly estimate β in such underdetermined regime we use an ℓ_1 -regularizer to reduce the variance of our estimate and prevent overfit [34]. SDD-OFF solves the following ℓ_1 -regularized least squares problem to learn β :

$$\underset{\beta}{\text{minimize}} \|\mathbf{L} - \mathbf{B}\beta\|_F^2 + \lambda \|\beta\|_{1,1},$$

where $\|\beta\|_{1,1} = \sum_{i,j} |\beta_{ij}|$ and λ is a scalar variable. The ℓ_1 -norm regularizer naturally promotes a sparse solution for β . The non-zero coordinates in β correspond to the measurements at time instances that are mostly correlated with the coordinates of the damage location. To solve the optimization problem, we use the FISTA [35] algorithm.

4. Results and discussions

4.1. SDD-ON

4.1.1. Finite Element Method based Simulations

This work studies active SHM in systems using high frequency GUWs as a means of damage detection. The word *active* points to the fact that the system at hand is subjected to an excitation using a piezo-electric actuator. The response of the system, subjected to the excitation is then sensed for the purposes of damage detection. Guided waves are generally multi-modal in nature, resulting in complicated waveforms traversing through systems. It becomes necessary to limit the frequency content of the signal to a narrow band, below a cut-off frequency (to minimize number of participating modes), to ensure effective damage detection. To achieve this, a 5 cycle tone burst signal with an appropriate central frequency f_0 is used. Dispersion curves, associated with each material, govern the choice of the central frequency based on the number of modes that will be allowed to propagate. Each wave mode is generally associated with a cut-off frequency and is not triggered until that frequency is reached. Figure 2 shows a typical five cycle tone burst signal with $f_0 = 50$ kHz.

We use a FEM software, developed by Doyle [36] and named QED, for simulating response signals in plates. The FEM model constitutes a square aluminum plate of side 12 inches and 0.1 inch thick with fixed boundary condition at all the edges. We use this alternate boundary condition for the purpose of demonstration of efficacy of the proposed algorithms. This is due to the inability of modeling the exact boundary conditions of the experimental setup using the FEM. In addition, the size of the simulated plate is much smaller than the experimental plate discussed below. This is to reduce the computational effort involved in the simulations. For guided wave response simulation the element size used in FEM has to be of the order of the wavelength of propagating elastic wave, to ensure numerical accuracy

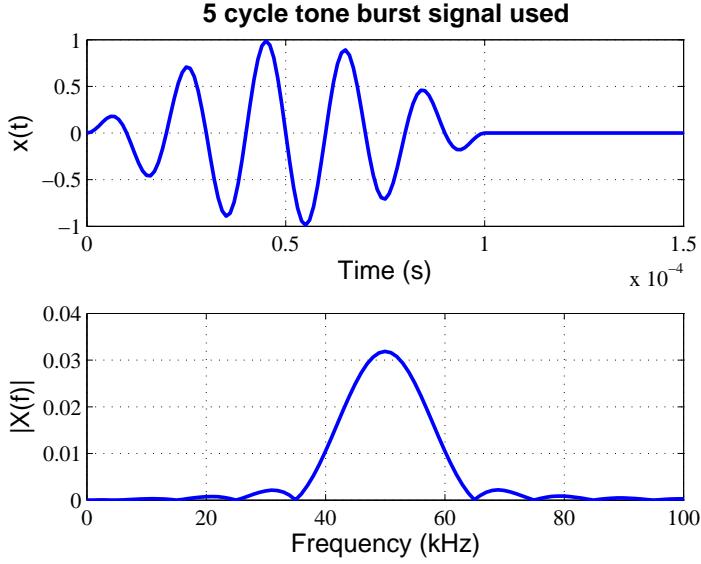


Figure 2: A typical 5 cycle tone burst signal used in this paper. The narrow band in the frequency domain ensures minimal dispersion of propagating waves. The central frequency, f_0 for this signal is 50 kHz. The dispersion curves of the system governs the choice of the central frequency.

[13]. Typically a very fine mesh is used to realize this, which leads to heavy computational effort. We use a circular through-hole of diameter 0.5 inch for modeling damage. A 5 cycle tone burst signal with a central frequency of 100 kHz is used for excitation. This ensures that only the S0 and A0 Lamb modes propagate. We model the excitation as a point load applied on the plate. Responses are measured at a single sensor location. Figure 3 shows the FEM setup of the system. The simulations are based on a single damage. As discussed earlier, the algorithms can easily be extended to multiple damages, under the assumption that the size of the damage is small compared to the actual size of the system.

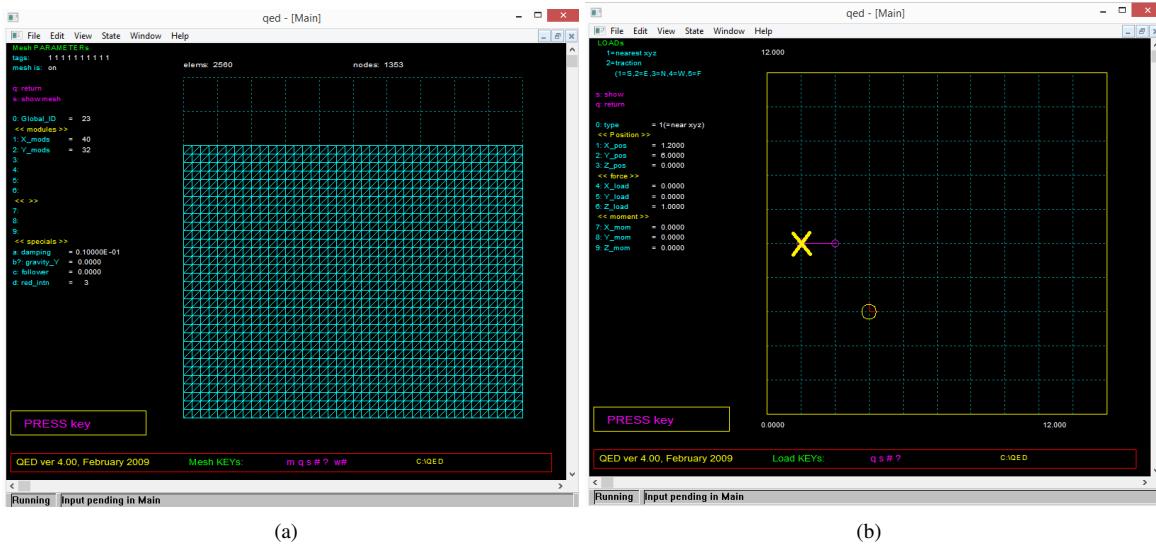


Figure 3: The FEM setup of the plate. (a) The plate used for simulation, with the mesh of triangular elements used for modeling the plate. (b) The model of damage that is used in the simulations. The through damage is depicted by the big circle on the plate. The cross shows the point of application of the pulse load. As discussed earlier, the actuator is modeled as a point load.

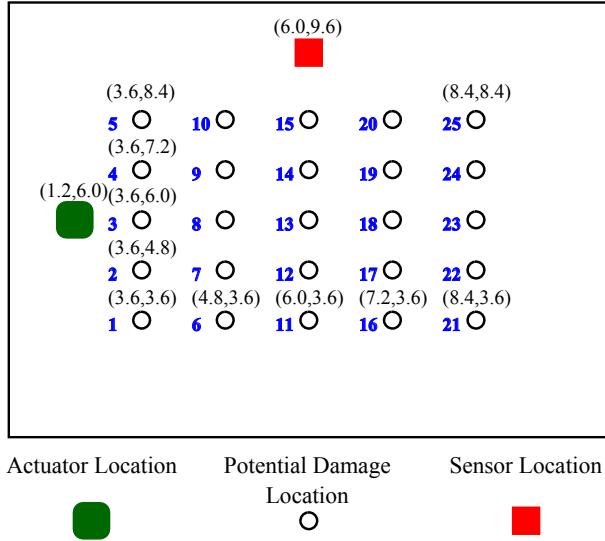


Figure 4: Simulation setup with 25 potential damage locations. The filled squircle is the actuator and the filled square is the sensor. The locations of each damage is shown as a hollow disc. The coordinates of the actuator, sensor and the potential damage locations are shown. The potential damage locations are labeled with blue digits. All the coordinates are in inches

We observe velocity response signals from nodes of the finite element mesh, at the sensor location. We then bandpass filter the signal between 50 kHz and 100 kHz. Figure 4 shows the location of the actuator, sensor and twenty five possible locations of damage. For each simulation, one of the possible locations is damaged and a response signal is observed at the sensor. As described in Figure 1 this is repeated for all the twenty five different locations producing a set of response signals. The coordinates of the possible damage locations constituting the dictionary, actuator and sensor are shown in figure 4. Locations 1, 3, 5, 11, 13, 15, 21, 23 and 25 are selected for constituting the grid. The rest of the locations are used to test the prowess of this algorithm for off-the-grid cases. The length of each response signal acquired is 8192, with a sampling frequency of 1 MHz. A typical signal acquired from the simulation is shown in Figure 5.

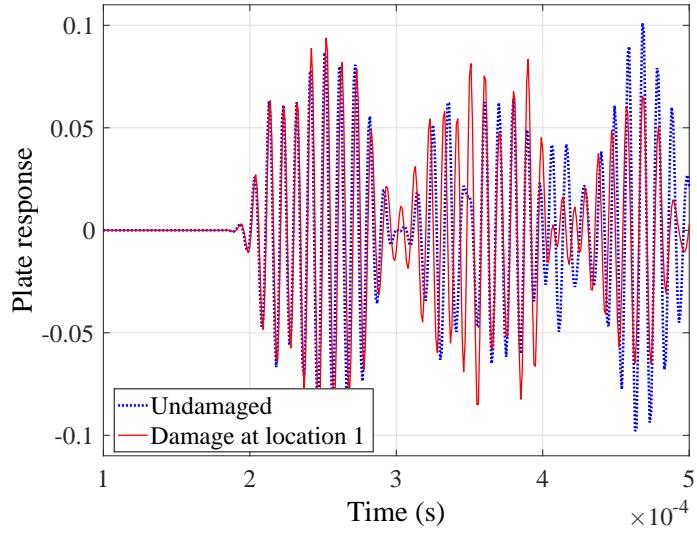


Figure 5: A typical signal acquired from the FEM simulation.

4.1.2. Validation of SDD-ON using FEM simulation results

SDD-ON uses the signals sensed in FEM simulation as inputs. Since, each signal is of length of 8192, the dictionary matrix or DCM, $\mathbf{A} \in \mathbb{R}^{8192 \times 9}$ as only nine of the damage locations constitute the grid for DCM construction. To test SDD-ON, we use signals acquired from damage locations 5 and 18 as test signals. Figure 6 shows the results from SDD-ON. It shows the values of each element of the sparse vector \mathbf{x} . The labels of each element are such that they represent the corresponding damage location. The element with the maximum value points to the specific column in the DCM. For figure 6 (a), location 5 is returned as the damage location for the test vector \mathbf{y} . Location 18 is a damage location that is off-the-grid. In such cases, SDD-ON predicts damage locations on-the-grid close to these off-the-grid locations. In this case, since locations 13 and 23 have the highest magnitudes, it is clear that the damage is located somewhere in between these two on-the-grid damage locations.

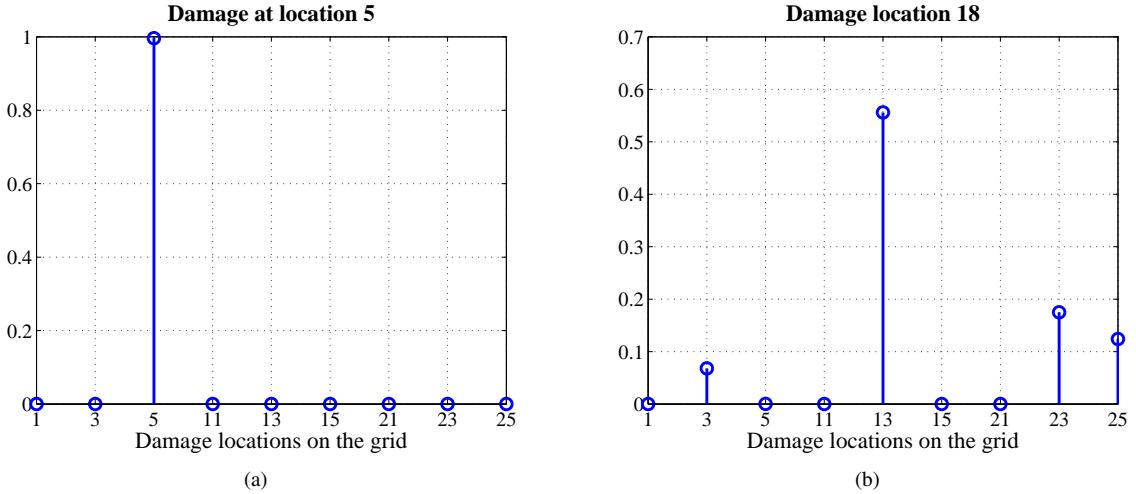


Figure 6: Results of sparse regression for simulation results. This shows the absolute value of the elements of the vector \mathbf{x} obtained from SDD-ON for two different test cases. Damage location 5 is a on-the-grid location, whereas location 18 is not. So SDD-ON outputs the exact location if the damage location is on-the-grid. If not it returns a damage location on-the-grid in the vicinity of the actual damage location. In this case damage location 13 is close to damage location 18.

The above demonstration was for a deterministic case. In real field monitoring, noise in the sensors will corrupt the acquired signals and may affect the performance of SDD-ON. To observe the effect of noise, SDD-ON was applied for each on grid location for multiple times with varying levels of noise corrupting the DCM elements and test vectors independently.

$$\begin{aligned}\tilde{\mathbf{a}}_i &= \mathbf{a}_i + \boldsymbol{\varepsilon}_a \quad \boldsymbol{\varepsilon}_a \sim \mathcal{N}(0, \sigma^2), \\ \tilde{\mathbf{y}} &= \mathbf{y} + \boldsymbol{\varepsilon}_y \quad \boldsymbol{\varepsilon}_y \sim \mathcal{N}(0, \sigma^2),\end{aligned}\tag{5}$$

where \mathbf{a}_i and \mathbf{y} are the noise free signals constituting the DCM and test vector respectively. $\tilde{\mathbf{a}}_i$ and $\tilde{\mathbf{y}}$ are the noisy signals constituting the DCM and the noisy test vector, respectively. $\boldsymbol{\varepsilon}_a$ and $\boldsymbol{\varepsilon}_y$ are the zero mean Gaussian white noise with a variance of σ^2 added to the DCM elements and test vector respectively. The additive noise is measured using signal to noise ratio (SNR) in dB:

$$SNR_{dB} = 10 \log_{10} \left(\frac{X_{signal}}{X_{noise}} \right)^2 \tag{6}$$

where X_{signal} and X_{noise} are the root mean square amplitudes of the noise-free signal and the noise. To quantify the accuracy, the number of incorrect on grid predictions is used as a measure (maximum being 9). Figure 7 shows the results for the sensitivity to noise study. Clearly, SDD-ON is shown to be robust to effects of noise. This is because, only negative values of SNR adversely affect the performance of the platform.

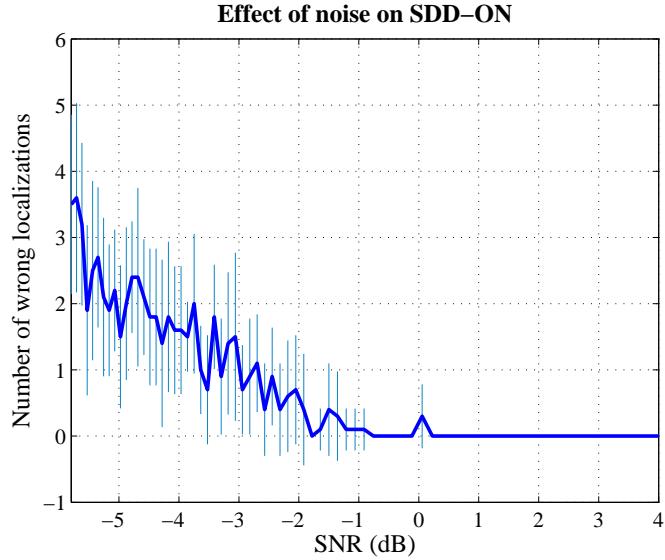


Figure 7: The effect of noise on SDD-ON, where noise is quantified using signal to noise ratio. The number of wrong localizations is an average of the number of wrong localizations over all possible damage locations. The maximum number of wrong localizations possible is 9. The above figure reports both the mean and standard deviations for various noise levels.

4.1.3. Experimental validation of SDD-ON

For experimental verification of SDD-ON, we use a square aluminum plate of 90 cm edge length and 3.51 mm thickness as shown in figure 8. In order to isolate the plate from the bench on which it is placed, we support the plate at four corners using rubber supports. The actuator and the sensor are circular piezoelectric sensors made of lead zirconium titanate (PZT) manufactured by Steminc Piezo. The circular disks are 10 mm in diameter and 3 mm thick with a dominant R mode vibration at a resonant frequency of 215 kHz. They are attached to the plate using 3M double sided tape. The plate is subjected to a five cycle tone burst signal of 10 V peak-to-peak with a central frequency of 100 kHz using a Tektronix AFG 3022B function generator. The signals are acquired using a Tektronix TDS2024C oscilloscope. We acquire sixty four signals for each case at a sampling rate of 5 MHz and then average them for noise reduction. This is followed by bandpass filtering of the average signal between 50 kHz and 150 kHz. As discussed earlier, we place masses at the prospective locations for simulating damage (see figure 8). Figure 9 shows the location of the actuator, sensor and the damage locations used to construct the dictionary. The filtered signal from the nine different on grid locations constitute the DCM. Following the construction of the DCM, we acquire test signals from the system.

Figure 11 shows the results for SDD-ON for a typical on-the-grid problem. The test vector is a signal acquired when the damage is at location 3. Again the elements of the sparse vector \mathbf{x} is shown. The third element has the maximum value again, demonstrating that SDD-ON is an effective tool for solving *on the grid* damage detection problems. It should be noted that due to presence of noise, all the elements of the vector \mathbf{x} is non-zero. To quantify the performance of SDD-ON platform for *on-the-grid* points, we use a receiver operating characteristic (ROC) curve. A ROC curve compares the true positive and false positive rates for binary classification problems and helps in quantifying the efficiency of the classifier. This is typically achieved by measuring the area under the curve (AUC) of the ROC curve.

The parameter for the ROC curve is the threshold value for magnitude of each element of \mathbf{x} such that a location is classified as damaged if the value of the location pointer vector at that coordinate is above the specified threshold. Figure 10 shows the ROC curve. The AUC associated with this ROC curve is 0.997. This shows that SDD-ON performs efficiently for on-the-grid locations. Figure 12 shows the performance of SDD-ON for off-the-grid points. We expect that for the off-grid points, the on-grid-point close to the actual damage location should carry the maximum weight, as evaluated from SDD-ON. The title of each subplot in the above figure contains two numbers. The first number is the actual location of the damage and the second signifies the predicted damage location. It can be seen that

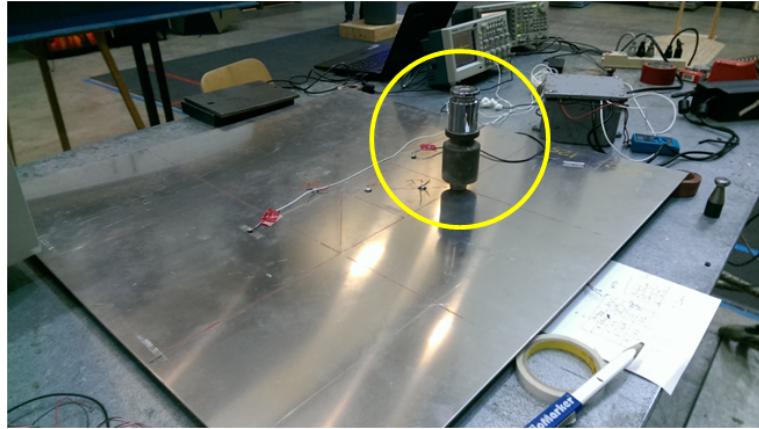


Figure 8: Experimental setup. The yellow circle shows the mass used for simulation of damage.

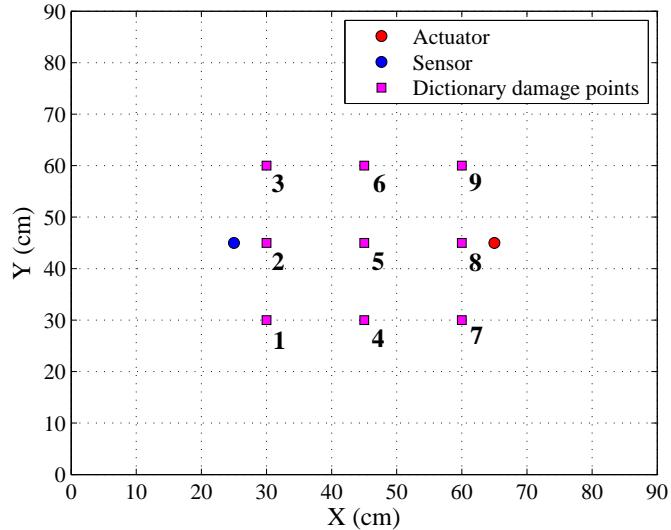


Figure 9: Locations of actuator, sensor and possible damage locations along with labels for the plate experiment

only for damage locations 7, 8, 9, 10 and 20 that SDD-ON makes inaccurate predictions. However, for all the other cases an on grid location in the neighborhood of the actual damage is predicted by the algorithm. SDD-ON is clearly effective for on-the-grid problems.

4.1.4. Detection and localization of damage on an experimental plate

In section 4.1.3, we demonstrate the efficacy of SDD-ON using test signals acquired by placing masses at various locations to simulate damage. In this section, we use the same dictionary from section 4.1.3, but use a test signal we acquire by drilling a partial hole at location 8, shown in figure 9. Although an added mass and a partial hole are very different from a physics point of view, the sparsity of the damage detection problem is still preserved. This implies, for a small enough damage, the dictionary constructed using adding masses, can still detect and localize a damage that was induced by drilling a partial hole. For a more comprehensive understanding of correlation between the adding mass and partial damages, one requires to perform a calibration study that relates crack size and added mass.

We induce damage to the plate at location 8 by drilling a 1 mm deep circular hole with a diameter of 4.5 mm. We

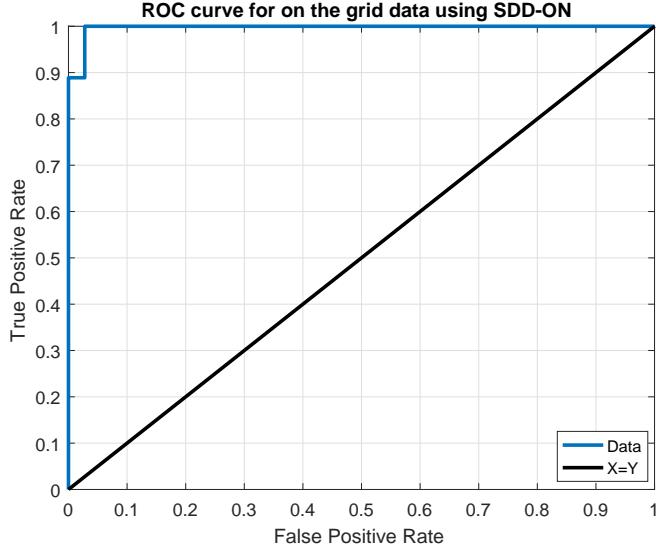


Figure 10: ROC curve for SDD-ON.

show the damage in figure 13 (a). Figure 13 (b) shows the results after applying SDD-ON. Clearly, SDD-ON localizes damage at location 8.

4.2. SDD-OFF

In this section, we present both simulation and experimental results to show the performance of SDD-OFF for the damage localization problem. We further investigate the case in which several damages are present in the structure simultaneously to assess SDD-OFF in localizing multiple damages. Finally, we show the robustness of the SDD-OFF method to different possible sources of noise and perturbations.

4.2.1. Validation of SDD-OFF using FEM simulation results

We begin by describing the simulation setup. We consider a setting in which we have a 2-dimensional plate. This plate has 25 designated points where on each of them there could be potentially a damage. The plate has one actuator and one sensor. Each time we put a damage on one of the designated points on the plate and measure the damage profile. Therefore, we obtain 25 different damage profiles which consist our training and test sets. Figure 4 shows the simulation setup. The sensor on the plate records the data at 1 MHz. We take 8192 samples for each damaged profile. However, in order to work with a stabilized signal and since there is a distance between the actuator and the sensor, we only consider the 659 middle samples of the whole damaged profile. Therefore, in our setting $N = 659$. Since we are working on a plate structure, $d = 2$. Finally, we divide the damaged profiles into training and test sets such that we do training on 24 damage profiles and do testing on 1 damage profile. Therefore, $M = 24$ during the training and $M = 1$ during the testing. The damage profile on which we test the performance of SDD-OFF could be any of the 25 damage profiles. Figure 15 shows the performance of SDD-OFF in localizing any of the damage profiles when they are in the test set.

In the next step, we are interested to find out about the performance of SDD-OFF in detecting more than one damage. In other words, we would like to see how SDD-OFF performs when it receives more than one damage profile during the testing phase. In this new experiment, we learn the β based on 20 damage profiles and test the performance of it on the remaining 5 damage profiles. Figure 14 shows the recovered locations corresponding to these 5 damage profiles. In Figure 14, red circles show the recovered locations of 5 damage profiles. We call these recovered locations P_1, P_2, \dots, P_5 . As we can see in the figure, some of the recovered locations are closer to their real locations than the others.

In order to quantify the performance of SDD-OFF we use the receiver operating characteristic (ROC). We consider 5 different circles centered at P_1, P_2, \dots, P_5 all of radius $0 \leq \varepsilon \leq \varepsilon_{\max}$ where ε_{\max} is chosen such that the area of each

ℓ_1 minimization based sparse representation for damage at location 3

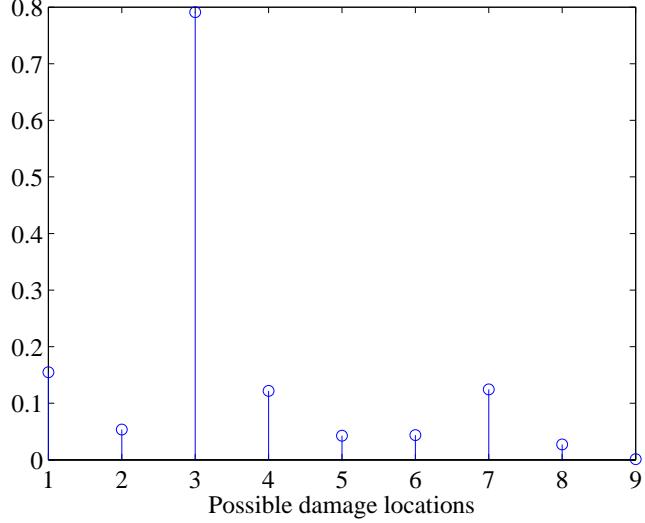


Figure 11: Values of the elements of the sparse vector \mathbf{x} obtained from ℓ_1 minimization from plate experiment for damage location 3.

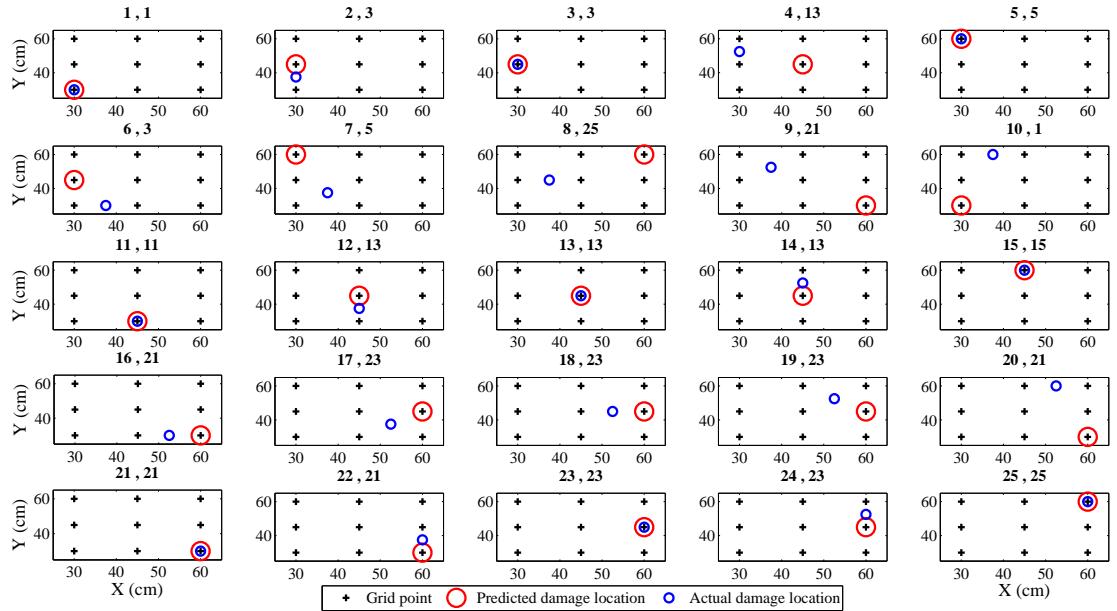


Figure 12: Performance of SDD-ON from plate experiment for all damage locations. The numbers on top of each figure denote the actual location of damage and the predicted damage location, shown by the small and big circles respectively. The black crosses denote the damage locations used as grid points for DCM construction.

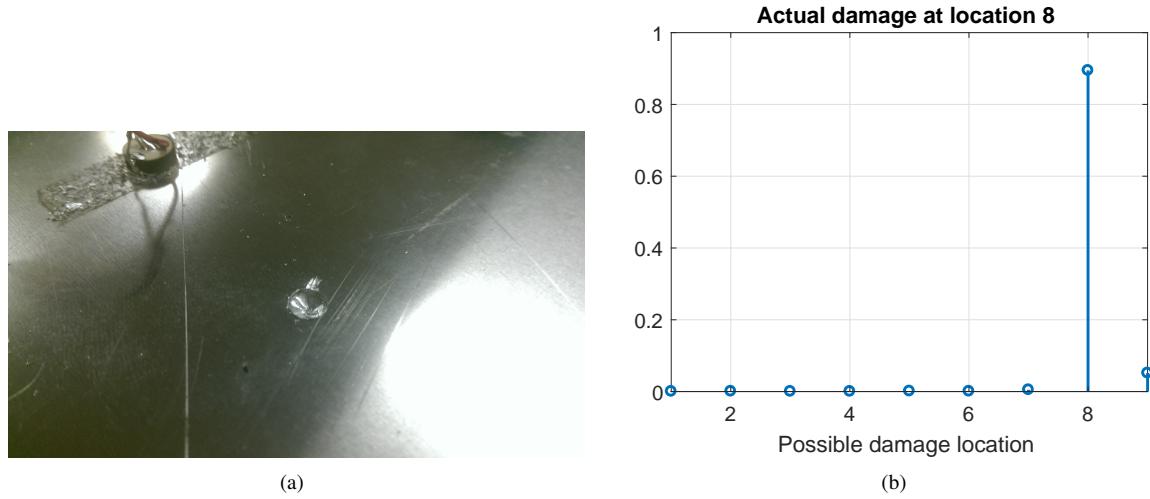


Figure 13: (a) The circular damage we induce at location 8, (b) Results of sparse regression for actual damage scenario. The algorithm correctly localizes damage at location 8.

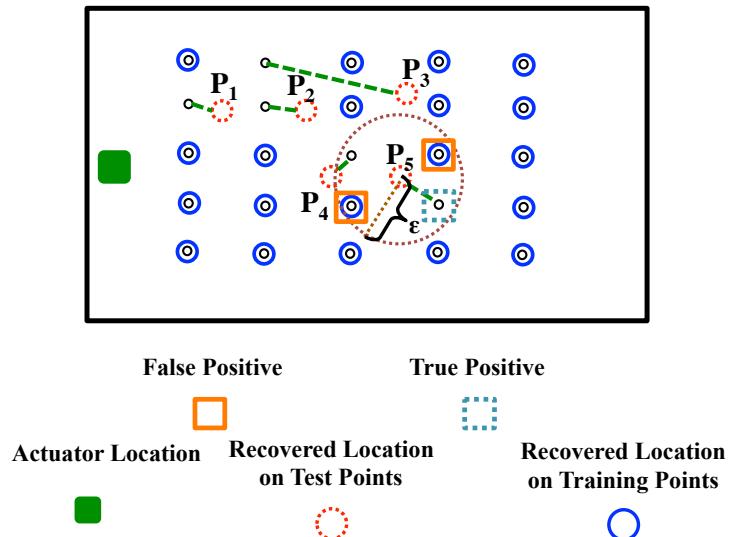


Figure 14: Setup of the experiment for testing the performance of SDD-OFF algorithm in detecting one damage on a plate. In this figure red dashed hollow circles show the recovered locations of 5 damage profiles in the test set. Dashed lines are used to connect the recovered locations to their actual locations.

circle of radius ϵ_{\max} centered at P_1, P_2, \dots, P_5 covers all the actual locations of 5 damage profiles in the test set (see Figure 14). We call these circles $C(P_1, \epsilon), \dots, C(P_5, \epsilon)$. For 100 different equi-spaced values of ϵ in the range of $0 \leq \epsilon \leq \epsilon_{\max}$, we calculate the ratio of true positives and false positives, i.e., we count the number of actual damage locations located in the circle centered at their corresponding recovered locations (True Positives) and we count the number of actual damage locations not located in the circle centered at their corresponding recovered locations (False Positives). Having calculated the true positives and false positives for different values of ϵ we can draw the ROC curve.

Finally, we are interested in seeing the effect of noise on the overall performance of our algorithm. Previously, we did not assume any noise on the system. However, sensor measurements are always mixed with some noise which

could be modeled as a Gaussian noise. Since we have put our sensor measurements in \mathbf{X}_{tr} , in order to incorporate noise in the model we assume that instead of \mathbf{X}_{tr} , we are given $\tilde{\mathbf{X}}_{tr} = \mathbf{X}_{tr} + \mathbf{X}_N$, where $\mathbf{X}_N \sim N(0, \sigma^2 \mathbf{I}_{M \times N})$. Figure 15 shows the ROC curve for noisy cases with different noise values.

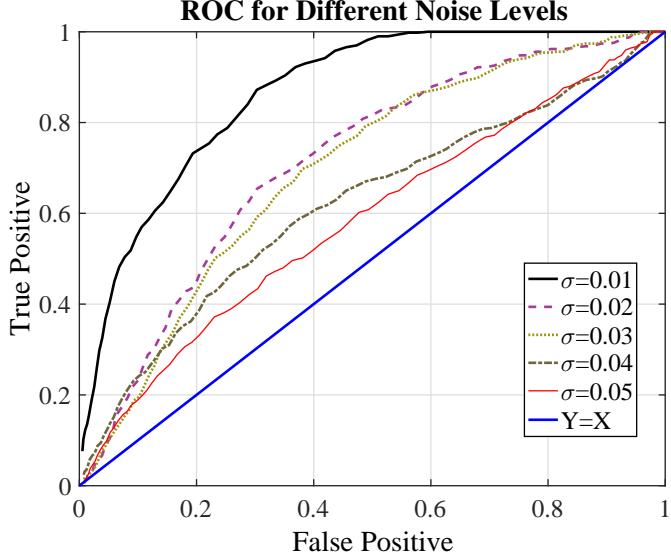


Figure 15: ROC curves illustrating the performance of the SDD-OFF algorithm in identifying damages on a numerically simulated plate in presence of noise with variance σ^2 .

4.2.2. Experimental validation of SDD-OFF

We experimentally validate the SDD-OFF platform in localizing the off-grid damages using the exact same setup that we used for the simulations in subsection 4.2.1. Figure 16 depicts the damage localization results on 25 test cases described in subsection 4.2.1 and Figure 17 plots SDD-OFF's damage detection performance. The ROC curve is generated using a similar procedure as described in section 4.1.2. In this real plate experiment SDD-OFF successfully detects the damages with AUC (Area Under the Curve) = 0.8314.

5. Conclusions

We have proposed two sparsity based damage detection algorithms for SHM of plates using GUWs. They are SDD-ON and SDD-OFF for on-the-grid and off-the-grid problems, respectively. The key idea behind these algorithms is the construction of a dictionary, or damage characterization matrix (DCM), representing the system at hand as it constitutes damaged response profiles of the system for different locations of damage. This aids in avoiding developing detailed models for damage detection purposes. Sparse damage pointer vectors are used for damage localization. Both the algorithms are able to perform damage localization on simulation as well as experimental data from plates with satisfactory accuracy. The possibility of extending SDD-ON and SDD-OFF for performing damage quantification, requires further research. However, the proposed algorithms are promising for extension to online health monitoring systems for complicated structures, as the development of high-fidelity models is circumvented.

6. Acknowledgment

The authors gratefully acknowledge the funding from Texas Instruments (TI), TI-168 G84040/G83198, for this project. The authors thank Dr. Dabak Anand at TI for the useful comments during the course of this work. The authors also thank Dr. Amardeep Sathyaranayana and Dr. Domingo Garcia from TI and Mr. Albert Daniel Neumann from Rice for their help in setting up of the hardware for the experiments.

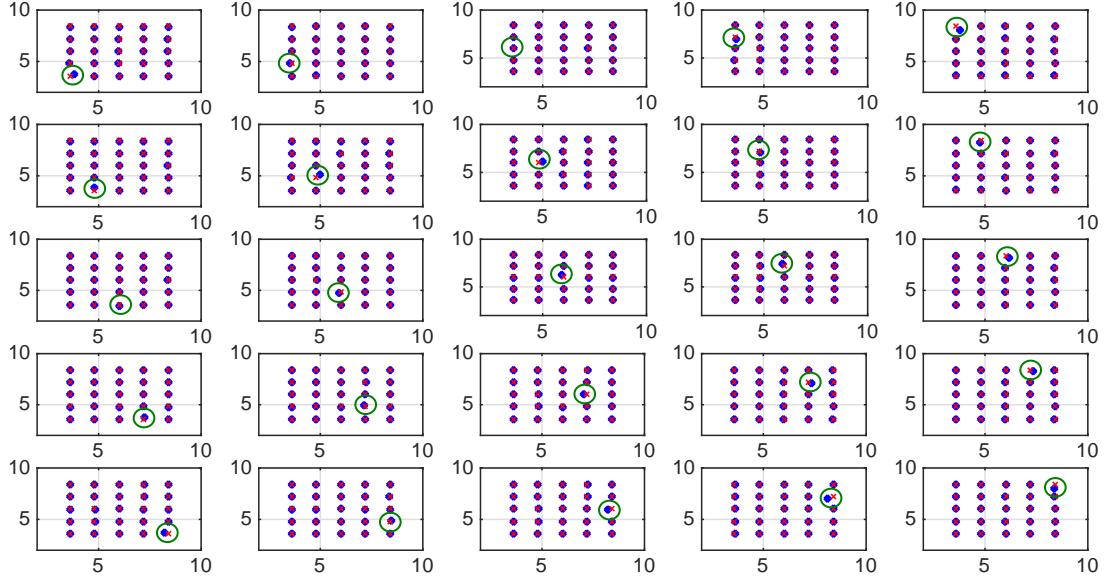


Figure 16: Damage localization results for 25 test cases from experimental data using SDD-OFF. Red dots indicates the correct damage locations and blue dots indicate the predicted damage location. The green circles show the coordinates used for the testing phase of SDD-OFF for each case. SDD-OFF finds the location of damage with very small error.

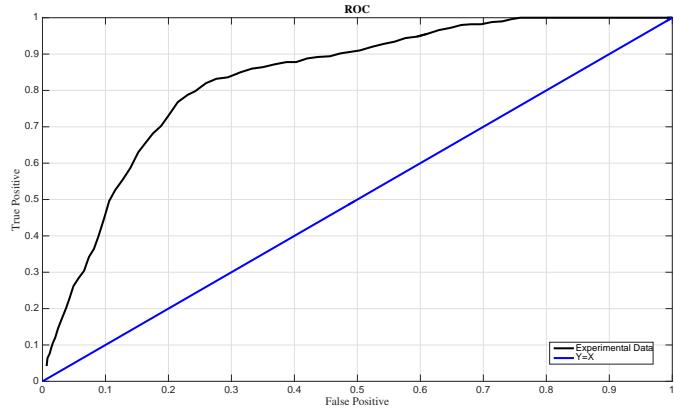


Figure 17: ROC curve for SDD-OFF in detecting damages in the plate experiment using real data.

References

- [1] C Farrar and K Worden. An introduction to structural health monitoring. *Philosophical Transactions of the Royal Society A*, 365:303–315, 2007.
- [2] H Sohn, CR Farrar, F Hemez, and J Czarnecki. A Review of Structural Health Monitoring Literature 1996 -2001. Technical report, Los Alamos National Laboratory, 2004.
- [3] J Ou and Hui Li. Structural Health Monitoring in mainland China: Review and Future Trends. *Structural Health Monitoring*, 9(3):219–241, 2010.
- [4] C Boller. Next generation structural health monitoring and its integration into aircraft design. *International Journal of Systems Science*, 31 (11):1333–1349, 2000.
- [5] YQ Ni, Y Xia, WY Liao, and JM Ko. Technology innovation in developing the structural health monitoring system for Guangzhou New TV Tower. *Structural Control and Health Monitoring*, 16(1):73–98, 2009.