STRUCTURAL DAMAGE DETECTION USING DICTIONARY LEARNING

by

Joseph Melville

A thesis submitted to the faculty of
The University of Utah
in partial fulfillment of the requirements for the degree of

Master of Science

Department of Mechanical Engineering

The University of Utah

May 2017

The University of Utah Graduate School

STATEMENT OF THESIS APPROVAL

The thesis of **Joseph Melville**
has been approved by the following supervisory committee members:

Joel Harley ,	Chair(s)	**8 Mar 2017** Date Approved
Daniel Adams ,	Member	**3 Mar 2017** Date Approved
Steven Naleway ,	Member	**7 Mar 2017** Date Approved

by **Tim Ameel** , Chair/Dean of
the Department/College/School of **Mechanical Engineering**
and by **David B. Kieda** , Dean of The Graduate School.

ABSTRACT

Nondestructive evaluation (NDE) is a means of assessing the reliability and integrity of a structural component and provides such information as the presence, location, extent, and type of damage in the component. Structural health monitoring (SHM) is a subfield of NDE, and focuses on a continuous monitoring of a structure while in use. SHM has been applied to structures such as bridges, buildings, pipelines, and airplanes with the goal of detecting the presence of damage as a means of determining whether a structure is in need of maintenance.

SHM can be posed as a modeling problem, where an accurate model allows for a more reliable prediction of structural behavior. More reliable predictions make it easier to determine if something is out of the ordinary with the structure. Structural models can be designed using analytical or empirical approaches. Most SHM applications use purely analytical models based on finite element analysis and fundamental wave propagation equations to construct behavioral predictions. Purely empirical models exist, but are less common. These often utilize pattern recognition algorithms to recognize features that indicate damage.

This thesis uses a method related to the k-means algorithm known as dictionary learning to train a wave propagation model from full wavefield data. These data are gathered from thin metal plates that exhibit complex wavefields dominated by multipath interference. We evaluate our model for its ability to detect damage in structures on which the model was not trained. These structures are similar to the training structure, but variable in material type and thickness. This evaluation will demonstrate how well learned dictionaries can both detect damage in a complex wavefield with multipath interference, and how well the learned model generalizes to structures with slight variations in properties. The damage detection and generalization results achieved using this empirical model are compared to similar results using both an analytical model and a support vector machine model.

For my wife and daughter, Ashley and Emma.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank Dr. Joel Harley for his patient instruction and guidance. He instructed me in some of the signal processing topics foundational to this work, and provided an ideal role model for me as a dedicated educator and researcher. I am extremely grateful for his help to keep me on track and headed in the right direction. He provided the praise, support, and constructive criticism that every student needs to flourish.

I am grateful for the advice and encouragement of Dr. Daniel Adams and Dr. Steven Naleway. They provided helpful feedback on thesis writing and presentation skills. I am grateful for their willingness to be involved in my work and help me along the way.

I thank Chris Deemer for the mentoring role that he played in this work. He helped me to see the long term potential of this research. I am grateful for the guidance he provided in all aspects of my research.

I thank Jeff Kessler for his support in putting together the data acquisition system. He provided most of the equipment needed for the construction of the system, offered guidance in learning how to operate it, and was available to help troubleshoot problems. He also provided the space where the system was run.

I thank all the other students that supported me, both inside and outside of the Wave Integrated Signal Processing (WISP) group. I especially recognize PhD candidate Supreet Alguri, with whom I worked beside in all of my work in dictionary learning and damage detection. He has helped me to understand and implement dictionary learning, and provided immeasurable support in developing experiments and writing about and presenting the results.

I thank the Department of Defense for funding my education and work through the Science, Mathematics and Research for Transformation (SMART) scholarship.

Finally, I am grateful for the tremendous support of my wife Ashley, and the motivation of my daughter Emma. They've been by my side though every high and every low throughout my education.

CHAPTER 1

INTRODUCTION

1.1 Motivation

Nondestructive evaluation (NDE) is a means of assessing the reliability and integrity of a structural component and provides information such as the presence, location, extent, and type of damage in the component [1]. Structural health monitoring (SHM) is a subfield of NDE, and focuses on a *continuous* monitoring of a structure *while in use*. SHM has been applied to structures such as bridges [2], buildings [3], and airplanes [4] with the goal of detecting the presence of damage as a means of determining whether a structure is in need of maintenance [5].

SHM can be posed as a modeling problem, where an accurate model allows for a more reliable prediction of structural behavior. More reliable predictions make it easier to determine if something is out of the ordinary with the structure. Structural models can be designed using analytical or empirical approaches. Most SHM applications use purely analytical models based on finite element analysis and fundamental wave propagation equations to construct behavioral predictions [6]. Other methods, such as sparse wavenumber analysis [7], combine analytical models with empirical data in a hybrid approach. Sparse wavenumber analysis uses analytically derived wave propagation equations together with empirical waveform signals to find the model constants that best recreate the true signal. Purely empirical models exist, but are less common. These often utilize pattern recognition algorithms to recognize features that indicate damage [8].

Analytical models have the advantage of using fundamental principles. Though an analytical model can be fundamentally correct, it often represents an oversimplification of the actual phenomenon. For this reason, analytical models of complex structures are unreliable in realistic conditions, and are typically only used to analyze simple geometries in controlled conditions [7].

Empirical models are constructed from data rather than fundamental principles. Ob-

servations of the phenomenon are recorded and used to generate a predictive model. If this is done correctly, these models can be generalized to other phenomena. An advantage of using an empirical model is its ability to model more complex structures than is possible with analytical models, including geometric complexity and multipath interference. However, there are much fewer guarantees that the resulting model will be correct or that it will be able to predict results that were not observed in the training of the model.

Techniques for training empirical models exist in the fields of machine learning and artificial intelligence and have been widely applied in many fields, such as data science [9], speech recognition [10], image processing and computer vision [11], and robotic manipulation and motion planning [12]. A few of these techniques include support vector machines [13], artificial neural networks [6], and genetic algorithms [14], and unsupervised algorithms such as k-means and nearest neighbor [15].

This thesis uses a method related to the k-means algorithm known as dictionary learning to train a wave propagation model from full wavefield data. These data are gathered from thin metal plates that exhibit complex wavefields dominated by multipath interference. We evaluate our model for its ability to detect damage in structures on which the model was not trained. These structures are similar to the training structure, but variable in material type and thickness. This evaluation will demonstrate how well learned dictionaries can both detect damage in a complex wavefield with multipath interference, and how well the learned model generalizes to structures with slight variations in properties. The damage detection and generalization results achieved using this empirical model are compared to similar results using both an analytical model and a support vector machine model.

The following sections provide a brief review of SHM and some of its uses in history, followed by a review of common damage detection methods. An abundance of analytical modeling techniques is noted. Empirical modeling of acoustic signals is then reviewed through the discussion of three machine learning algorithms: support vector machines, artificial neural networks, and genetic algorithms. A background for compressed sensing is also addressed, which will serve as a foundation for our empirical modeling method based on dictionary learning [16].

1.2 Background

Structural Health Monitoring (SHM) is used to continuously assess the damage state of a structure with the intent of answering the following questions [17]:

1. Is there damage present in the structure?
2. Where is the damage located?
3. What type of damage is present?
4. How extensive is the damage?
5. How much useful life remains in the structure?

SHM is related to the fields of condition monitoring [18], nondestructive evaluation (NDE) [1], statistical process control [19], and damage prognosis [20]. Some of these describe continuous monitoring, while others require a removal of the structure from service, which describes an NDE paradigm. NDE methods are more intrusive, but often provide much more detailed information about the damage state of the structure.

SHM methods are divided into two categories in this background: methods that use acoustic signals and methods that use ultrasonic signals. Regardless SHM methods find features that model the properties and dynamic response of a material and watch to see if they change. This is done almost exclusively through baseline subtraction or through the use of predictive analytical models that are used to recover baseline data. If a change is detected, damage is indicated [21]. If the change can be detected locally, then the damage has been located [22]. The way in which a feature changes can be an indicator of the type of damage present, and the extent of the change can indicate the size of the damage [23]. If the changes in features can be mapped to particular material properties, such as flexibility, stress, and/or strain, then a prognosis can be made to determine the remaining useful life of the structure [20].

A key challenge to this paradigm is that structural parameters are affected by environmental and operational variation, which are not indicators of damage. This can lead to false positives in damage detection and is prevalent in both acoustic and ultrasonic signal propagation [21, 24]. Signal interpretation and feature extraction are also challenges,

especially with ultrasonic signals, which can be very complex due to their high-frequency nature. This complexity often makes analytical models inadequate for predictive baseline subtraction. These challenges of environmental sensitivity and signal complexity will be addressed at the end of this section.

The following subsection reviews a brief history of SHM applications, followed by a description of some of the parameters and techniques used for both acoustic and ultrasonic damage detection. Finally, the challenges of environmental sensitivity and signal complexity are addressed. Note that a more detailed history of SHM applications can be found in the works of Doebling et al. [5], Sohn et al. [25], and Farrar et al. [8].

1.2.1 Brief History of SHM Applications

Damage detection in one form or another has been practiced for thousands of years. For example, a person with a trained ear, who is familiar with the normal response of the structure (train wheels, etc.), can do a tap test to listen for structural abnormalities [26]. While similar damage detection methods have been used for a very long time, advances in electrical measurement instruments have made it possible in the last 70 years to better quantify damage detection efforts. This has allowed for these methods to be more automatic, consistent, and sensitive. In recent years, the ever accelerating advances in microchip electronics and sensor technology have led to continued improvement in these areas.

Early attempts at quantifiable damage detection were made in the 50s with rotating machinery such as mills, lathes, engines, generators, and so forth. [27]. This falls under the the related field of condition monitoring. The types of damage that can typically be detected include damaged or loose bearings [28], misaligned shafts [29], and chipped gear teeth [30]. These attempts have been generally successful for a number of reasons, including: minimal operational and environmental variations, well defined damage type and location, large databases that include data from damage systems, and clear and quantifiable economic benefits.

Towards the 1970s and 80s, many attempts were made to identify damage on offshore platforms, such as pile drivers and oil rigs [31]. Shifts in the structure's natural frequency, modal shape, curvature, and flexibility were used to detect the existence and location of

damage. These methods are theoretically sensitive to changes in mass and stiffness of the structure. In practice however, these detection methods proved to be too insensitive to damage, and too sensitive to changes caused by aquatic growth and on-board tank levels [31]. These factors cause variations in natural frequencies and modal shapes that are of the same magnitude as detectable damage, often making it impossible to distinguish between the two [32].

Around this same time period, the aerospace community began to study the use of vibration-based damage detection in conjunction with the development of the first space shuttle [33]. The shuttle modal inspection system (SMIS) [34] was developed to identify fatigue damage in components such as control surfaces, fuselage panels, and lifting surfaces. These systems are covered by thermal protection and are hard to access using other methods. SMIS proved to be a successful method of fatigue damage detection.

High frequency ultrasonic signals (>20 kHz) have been used throughout the history of damage detection, and for many of the attempts mentioned above. More recently, however, increasing sensitivity and speed of measurement devices have made it possible to detect higher level modes of vibration, which contain even more data about the damage state of a structure [35]. Piezoelectric ceramic sensors are commonly used for the measurement of ultrasonic signals. Reliable manufacturing of these sensors has been around since the 1950s [36]. These have been used heavily for ultrasonic damage detection in the past two decades [37] by using the sensors in a sparse array arrangement, similar to those used in radiology [38] and seismology [39].

Another common measurement device is called a laser doppler vibrometer (LDV). One of the first uses of an LDV for damage detection was in 1990 by Sriram et al. [40], who used the device to measure the modal properties of a structure. Previously, accelerometers and piezoelectrics were used widely for these measurements. LDVs have also been used to measure full grids of evenly spaced ultrasonic measurements over the surface of structures, also known as full wavefield measurements [41]. This type of data provides a wealth of information about the properties and dynamic responses of a structure. However, the use of full wavefield data for damage detection is still an area of active research, limited to simple 2D structures.

1.2.2 Acoustics

Acoustic signals are fairly easy to capture, and provide much information about the general state of the structure as a whole, such as the natural frequencies, mass, and flexibility. These features can be used for damage detection by comparing the signal features to an empirical baseline of the feature, or a baseline prediction using an analytical model. Both approaches have been attempted. Changes in these properties, however, are not only indicative of damage, but also environmental and operational variations [42]. For this reason, acoustic damage detection is often most successful in controlled environments where these variations either do not exist, or can be measured and compensated for.

There are a number of damage sensitive features that can be extracted from an acoustic signal, and each provides specific insight into the damage state of a structure. These features are discussed in the following sections.

1.2.2.1 Modal Frequency Shift

Overall, a frequency shift due to damage in a structure is very small. Extremely sensitive sensors need to be used or a large amount of damage must be present to be able to adequately detect damage. Also, other variations in structural parameters could cause frequency shifts that are of the same magnitude as that of actual damage, such as environmental conditions. Therefore, damage detection using this method only realistically works in a controlled environment, such as with manufacturing quality control. Also, because natural frequency is a global property, it is hard to locate the damage after detecting it [5].

Vandiver et al. [21] describe a technique that can be used to detect subsurface structural failures by detecting changes in natural frequencies of the structure. The technique involves the use of accelerometers to pick up vibrations excited by the wind and waves of the ocean. The natural frequencies of the structure can then be extracted from the accelerometer data. Shifts in the natural frequency over time indicate changes in mass and stiffness of the structure. This technique has been used before for damage detection in buildings and rotating machinery such as generators and jet engines. By comparing frequency changes between modes one can determine whether or not the damage is in a location that causes stiffness in a predominantly X or Y direction. This method works best with large amounts of structural damage and is limited in its ability to detect small

amounts of damage.

Farrar et al. [2] performed vibration tests in the fracture-critical I-40 bridge over the Rio Grande. Cuts were made in the bridge grinders to simulate fatigue cracks in the bridge, and both passive and active excitations were used to acquire the resonant frequencies of the bridge. Resonant frequency shifts were seen for both methods, and both indicated damage in the bridge.

1.2.2.2 Modal Shapes

Modal shapes alone do not typically give a good indication of damage. They can, however, provide spatial information about damage location, and can be effective when coupled with frequency shift features. The effectiveness of this method depends highly on the selection of which modes are used.

Kam and Lee [23] used modal shapes and natural frequencies to identify the location and size of a crack in a structure. The cracked structure was discretized into a number of elements. The location of the crack was identified through the determination of the sectional parameters of the elements using a Taylor series estimated eigencouple (modal shape and natural frequency). The crack size was estimated by analyzing the strain energies (calculated from the modal shapes) of the structure with and without damage.

Lam et al. [22] defined a mode shape normalized by the change in natural frequency of another mode as a damage signature. The signature is a function of damage location but not of damage extent. An analytical model was then used to derive the signatures for damage in different locations. The measured signatures were matched to the damage state by selecting which analytical signature gave the best match using modal assurance criterion (MAC). MAC is one of many ways to determine the level of correlation between modes.

1.2.2.3 Modal Curvature and Mode Shape Derivatives

Modal curvature also provides spatial information about the damage. Curvature needs to be calculated through a measurement of the strain. It can also be calculated directly from the mode shapes, but provides less accurate results.

Salawu and William [43] used a modal shape curvature measure for finding damage location and extent. They compared this result to a mode shape relative difference method.

This was done without extensive computation or a theoretical damage model. Simulated data were used to determine the effectiveness of the method without measurement error. They demonstrated that the curvature change does not typically give a good indication of damage using experimental data.

Chance et al. [44] used a simple linear analytical model to produce modal shapes from both damaged and undamaged structures from which modal curvatures are derived. To validate the linear model, a nonlinear forced response analysis was carried out using a bilinear stiffness element.They found that numerically calculating curvature from mode shapes did not produce accurate results. Measured strains were instead used to calculate the modal curvature and the results improved.

1.2.2.4 Dynamic Flexibility

A flexibility matrix relates static applied force to a resulting displacement [5]. Approximations of this flexibility matrix are calculated using mass-normalized mode shapes and frequencies extracted from empirical data. Typically, flexibility matrices are compared to a baseline flexibility matrix to detect damage, but anomalies can be noticed without a baseline.

Aktan et al. [45] used modal flexibility obtained by postprocessing the frequencies and mass normalized modal vectors as structural damage indicators. The reliability of modal flexibility was verified by comparing bridge deflections derived from modal flexibility to actual static deflection of the bridge. Analytical models, tuned by experimental data from the bridges, were used as a basis for condition assessment. Continued work in this area can be found in research done by Toksoy and Aktan [46] and Pandey and Biswas [47].

1.2.2.5 Empirical Modeling of Acoustic Signals

By processing large amounts of data from both damaged and undamaged structures, a notion of damage can be empirically modeled using statistical pattern recognition (SPR) techniques. This model can then be used to detect the presence of damage in new signals. In contrast to the simple single or double featured damage detection techniques normally used for acoustic signals, this SPR method can find more complex relationships between the presence of damage and a combination of multiple features. This often allows even further insight into the damage state of the structure. The following presents three of these

techniques and a few of their applications.

1.2.2.5.1 Artificial Neural Networks

Artificial neural networks (ANN) work well for damage detection due to their ability to model arbitrarily complex functions with features. However, large amounts of labeled data (i.e., damaged and undamaged) are required for training a network, as well as many training iterations. A guide for developing neural networks for pattern recognition can be found in [48].

Instead of using an analytical model, the self organizing properties of neural networks can use structural measurements to develop an empirical model that represents all of the material behavior within one unified environment. Ghaboussi et al. [49] developed an empirical model of the behavior of concrete in a state of plane stress caused by different types of loading. This was done using a backpropagation neural network.

Kudva et al. [6] used finite-element analysis to create a data set of strain measurements taken at discrete locations. These data were then used to train a neural network to deduce damage size and location. The results for localization were very good, but damage size deduction proved to be more difficult.

1.2.2.5.2 Support Vector Machines

Like ANNs, support vector machines (SVM) also need labeled undamaged and damaged data to be able to distinguish which features indicate damage most clearly, but the largest margin approach used in this method requires less data than ANN. Kernels can also be used to train an SVM to model nonlinear relationships. Burgess [50] provided a tutorial for the development of SVM for pattern recognition.

Worden and Lane [13] used an SVM to do fault classification of ball bearings and damage localization on a framework structure. Worden also compared the SVM approach to other established means of pattern recognition such as neural networks, nearest neighbor, and kernel discriminant analysis. Kernel discriminant analysis [51] was able to consistently outperform the SVM approach, though SVM algorithms generally produce very good results.

Widodo et al. [52] provided a review of methods used for machine condition monitor-

ing, including neural networks, fuzzy expert systems, condition-based reasoning, random forests, and so forth. The SVM method resulted in very generalized results.

1.2.2.5.3 Genetic Algorithms

Genetic algorithms converge on a modeling function using a Darwinistic approach which favors functions that demonstrate best fitness, or best performance.

Rao et al. [14] used a genetic algorithm approach to find the relationship between residual forces and damage location and extent. The resulting model was validated using an analytical model, which had excellent agreement.

1.2.3 Ultrasonics

Ultrasonic signals are those that are above 20 kHz in frequency and can be transmitted through a structure as guided waves. Guided waves are high-frequency signals with wavelengths that are much smaller than the physical dimensions of the propagating structure. They often have to be actively excited in the structure, either by a separate sensor or an impact, and usually give information about the part of the structure between excitation and measurement. Guided waves attenuate little over large areas of the structure and are therefore useful for SHM. Guided waves in a plate are known as Lamb waves [53]. Other types of guided waves are Rayleigh and Love waves, commonly used in seismology [54]; Perekis waves, which are used in shallow water acoustics [55]; and electromagnetic waves transmitted through wires, used in electronics and power systems [56].

In this section we discuss the propagation of ultrasonic signals through structures as guided waves, and their use in damage detection. This is most often done using Lamb waves. First, some of the challenges of using ultrasonic signals are discussed, namely complexity and variability. Then a review of common damage detection techniques is addressed, including the use of three different types of data set: sparse array data, full wavefield data, and partial wavefield data.

1.2.3.1 Complexity

Ultrasonic signals can give far more insight into the properties and dynamic responses of a structure than lower frequency acoustics often can, but they introduce the challenge of complexity due to the multimodal nature of higher frequency signals. These modes are a

result of interference patterns created by the interaction of the guided waves with the top and bottom of the guide plate [53]. Each mode travels at a different velocity as a function of frequency [57] and distorts the waves as they propagate through the plate. Furthermore, as frequency increases, more modes are present in the signal. The more modes that are excited, the more information that can be extracted about the material state, though higher modes are increasingly difficult to detect and measure. At frequencies under 500 kHz, two modes are almost always present: the zeroth antisymmetric mode and the zeroth symmetric mode.

To better understand and visualize complex guided waves, the velocity of each mode can be plotted as function of frequency. These are called the dispersion curves of the wave. These dispersion curves are often used to tune analytical modeling parameters. The tuned parameters can then be used to predict signal responses to compare with true signals and to facilitate damage detection.

There are many methods that have been attempted to extract dispersion curves from guide waves. The two-dimensional discrete Fourier transform (2D-DFT) has been used to identify and measure the amplitude of individual Lamb waves [57], and to find the thickness, bulk velocities, and elastic constants of waves [58]. There are also many time-frequency analysis methods that use the reassigned spectrogram [59], pseudo-WignerVille distribution [60], or wavelet transform[61]. Time-domain matching pursuit approaches [62, 63] and other model-based strategies [64, 65] also exist.

These methods are good at analyzing guided waves but are not accurate enough to build an effective model for damage detection. Also, most of these methods do not work outside of ideal circumstances, when the plate is not of infinite length and width and multipath inference caused by reflections is present.

1.2.3.2 Variability

Another significant challenge to overcome when working with ultrasonic signals is very similar to that of acoustics signals: Environmental and operational variations can result in a false positive damage detection. This is because environmental parameters such as temperature [66] and applied stress [67] have a direct effect on material properties. Temperature produces the largest effect [68] and has therefore been addressed most often

in the literature. Temperature can affect the velocity of the waves in a plate and change the dispersion curve. Operational variations such as slight shifts in the structure during data acquisition and varying bond integrity of sensors often have a very similar effect.

The following are some of the methods used to account for these variations. Temperature variation will be used as an example for each method. One of the simplest methods used to account for temperature variation is to measure the temperature at time of data acquisition and account for it in the material model [69]. Another method is to take a baseline of the data at a number of different temperatures. Then when detecting damage, the baseline that most resembles the new measurement will be used to compare against [24]. These first two methods work well, but require additional baseline data.

Methods that do not need additional baseline data include local peak coherence [70–72] and optimal signal stretch (OSS) [73]. Both work by approximating the temperature change as a stretching of the signal. Local peak coherence is quick, but is sensitive to other variations that cannot be modeled as a stretching of the data. OSS correlates each signal with a library of stretched replicas of a single baseline, and is more robust, but not as efficient.

An additional method uses an analytical wave propagation model together with compressed sensing [74] to recover a version of the input signal that represents a measurement from the undamaged version of the structure in question, but with all of the same variations as the input signal. A baseline subtraction then cancels out the effects of those variations, and differences only indicate damage in the original signal [7].

1.2.3.3 Ultrasonic Methods of Damage Detection

Structural damage detection with ultrasonic signals can be approached in a number of different ways: using spare sensor arrays, full wavefield data, or partial wavefield data.

1.2.3.3.1 Sparse Array

Sparse sensor arrays use a small number of sensors, often around 3-5, distributed over the area of interest to detect damage. Damage is detected most successfully when it is encompassed by the sensors. High localization accuracy and resolution constitute a successful damage localization. Dense sensor arrays are also used in SHM, but usually

only on large structures when spatial distribution is difficult to achieve [75]. Both sparse and dense sensor arrays are often used in radiology [38], seismology [39], and underwater acoustics [76]. The use of sparse sensor arrays in SHM is a more recent application, and usually consists of piezoelectric sensors being permanently bonded to the surface of a structure.

Most methods used for detecting damage with spares sensor arrays again depend on comparing signals to a baseline recorded from the undamaged structure. A simple method uses the peak amplitude or energy of the waveform difference as an indicator of damage. This approach is useful when the signal variations are caused by damage alone.

Because other variables, such as temperature, can cause signals to vary, additional time and frequency domain features can be used to account for those effects. Gao et al. [37] introduced a signal difference coefficient that is not a function of amplitude changes. Only signal shape changes are detected. Michaels et al. [77] used a differential feature-based classifier for the same effect. Lu and Michaels [78] attempted to normalize for the effect of wetting and physical contact on a plate by selecting many differential signal features and using a voting method to determine if the structure is damaged.

Damage can also be located by finding the sensor that is nearest to the damage. Mal et al. [79] used a damage correlation index that depends on the differences found between the dynamic responses of the two signals. This is done using the frequency response function. Zhao et al. [80] expanded on this method with RAPID (reconstruction algorithm for probabilistic inspection of defects) by spatially distributing and summing signal difference coefficients in elliptical patterns for all transducer pairs.

Another method for damage visualization, called delay-and-sum, delays received signals and sums them according to appropriate spatial rules for each point on the image. This is often used for phased array imaging techniques. Wang [81] developed a digital imaging method based on this delay-and-sum algorithm. Michaels and Michaels [82] applied this imaging method using both raw and envelope-detected signal differences to create localized images at multiple frequencies. A combination of these images increased damage localization accuracy. Fromme [83] created delay-and-sum images using both finite element and experimental data to locate damage on a steel plate.

1.2.3.3.2 Full Wavefield

Full wavefield data consists of evenly spaced point scans done in a grid pattern over the interrogated surface of a structure using a laser doppler vibrometer (LDV). Instead of a sparse representation of the wave propagation over the structure, a full image is now available without the need for any additional processing. Full wavefield images contain a wealth of information about the structure, much of which can be used for damage detection, but it takes a long time to actually produce the scan.

One of the first uses of an LDV for structural health monitoring was done by Staszewski et al. [41]. The results of this work proved that Lamb waves can be measured accurately by an LDV. This was done by comparing a full wavefield scan to a theoretical model and similar measurements taken using piezoelectric sensors. The full wavefield scan produced using the LDV was then used to identify the location and extent of damage on a metal plate using a simple amplitude comparison method [84].

There have been many works since that have used LDVs to facilitate damage identification. Some of these include the use of an LDV to detect fatigue cracks [85], the use of a 3D LDV for fatigue crack detection [86], and the use of a laser as an actuator as well as a sensor for a complete noncontact approach to damage detection [87].

Other methods for damage detection and localization with a full wavefield include work done by Ruzzene [88], who used a frequency-wavenumber domain filter to remove incident waves from the wavefield image, isolate reflections caused by material flaws, and improve damage visualization. Sohn et al. [89] used an accumulated mass-normalized kinetic energy to visualize a delamination in a composite plate. He then used a Laplacian image filter to highlight the damaged area. In this same work, Sohn used local wavenumber analysis to visualize this same damage type.

1.2.3.3.3 Partial Wavefield

The full wavefield is not always necessary when detecting damage. A small percentage of the wavefield, coupled with a model of the wave propagation through the plate, can be used to recovery a full wavefield. The key to success with partial wavefields is the accuracy of the propagation model. With this model, compressed sensing [74] can be used to recover model parameters from the partial wavefield. Then the model and parameters

can be combined to recover the entire wavefield.

Harley and Moura [7] demonstrated a method of parameter recovery using an analytical model and compressed sensing. Alguri and Harley [90] and Esfandabadi et al. [91] have used these same parameter recovery techniques using an empirical model created through dictionary learning. These topics will be discussed in more detail as it is the main topic of this thesis.

1.3 Recent Work

The purpose of this section is to provide a review of more recent literature used in the development of the current work. First, the theory of compressed sensing and its use in SHM are discussed. Second, the theory of dictionary learning is discussed along with its uses for SHM. The limitations of the current methods will then be considered and the methodology of this thesis proposed.

1.3.1 Compressed Sensing

An LDV can measure enormous amounts of vibration data with relative ease, but the time it takes to acquire a full wave field measurement is far too long. A detailed scan of 10,000 points and 100 averaging signals can take around seven hours. If even more resolution is required, or a larger scan area, scans can take much longer. This extended acquisition time makes the process practically unusable for regular inspection and damage detection, especially for SHM, which demands a continuously monitored structure.

A technique called compressed sensing can be used to recover a full wave field scan using only a fraction of the wavefield. A 7 hour scan can be recovered over 100X faster using a partial scan of 3 minutes or less. With such fast scanning speeds, it is possible to achieve semicontinuous damage detection which can be used in the time frame of SHM. However, it is important to note that compressed sensing does require a model of the wave propagation to be able to recover the wavefield. The recovery will have the same spatial structure as the model while having the same frequency and amplitude structures of the partial scan data.

The abstract framework of compressed sensing was presented by Donoho [74]. A basis pursuit method was presented for the compression of signal data sets. For compressed

sensing to be effective, the data set needs to be sparse in a known basis. Sparsity is a measure of how much redundancy can be found in a data set, and refers to the smallest number of common bases the data set can be broken into. If few bases are needed to represent the whole data set, then the data set is sparse. Compressed sensing is applicable to full wavefield data because much of the data measured in a wavefield is redundant in certain domains and can be compressed into a smaller set of base signals. In other words, a small linear combination of bases can be used to recover a much larger set of signals with high accuracy. Compressed sensing has been used for both dispersion curve recovery [92] and full wave field recovery [90], using analytical models composed of different basis domains.

Ianni et al. [93] tested different bases (Fourier, curvelets, and wave atom domains) for use with compressed sensing to examine which one produces the best recovery of the full wave field. They were able to recover a field with reasonable accuracy using only 34% of the original data. The Fourier domain proved to provide the best recovery. Success in the Fourier domain was ascribed to 1) the fast computation of the Fourier transform, 2) the sparse representation of Lamb wave signals in the Fourier domain, and 3) the high incoherence of Fourier exponentials with the sampling schemes used (jittered subsampling and furthest-point sampling).

Harley and Moura [7] demonstrated that compressed sensing can be used to recover dispersion curves using orthogonal matching pursuit (OMP). This method is called sparse wavenumber analysis. The result was possible because the wavenumber domain is sparse in nature. A comparison between basis pursuit denoising and OMP was discussed, along with the construction of the wave propagation model used in sparse wavenumber analysis [92].

This same method of sparse wavenumber analysis was used again by Harley [94] to recover full data sets for standing waves on a string, Lamb waves in an isotropic plate, and guided waves in a unidirectional, anisotropic plate. He was able to accurately predict 149 765 experimental time-domain measurements from 36 local measurements.

1.3.2 Dictionary Learning

Compressed sensing allows for quick recovery of full wave field data, but this recovery is highly dependent on the wave propagation model used for the recovery. The use of an analytical model does not allow for the full exploitation of the rich information that can be found in full wavefield scans. However, there is a way to create an empirical model using the full wavefield scans themselves. This process is called dictionary learning [16].

If constructed correctly, a learned dictionary empirical model will facilitate in the recovery of wavefields that include more detailed features than those recovered using analytical models, such as multipath interactions. This empirical model should also be able to generalize for some material and environmental variations and can be used for damage detection with structures of similar geometry.

Dictionary learning is a process through which a basis (or dictionary) is learned given a signal training set. It has two steps: a sparse coding step (compressed sensing), and a dictionary update step. The sparse coding step can be accomplished using a sparse approximation algorithm such as orthogonal matching pursuit (OMP) [95]. The dictionary update step can be solved using a gradient decent procedure such as method of optimal directions (MOD) [96].

Tosic and Frossard [16] have provided a detailed overview of dictionary learning theory and methodology. Many examples for dictionary learning application were also presented and discussed.

An extremely flexible and efficient algorithm for dictionary learning was developed by Aharon et al.[15] called the K-SVD algorithm. This approach solves for the dictionary iteratively by updating one basis column (dictionary atom) at a time while finding a sparse recovery in between each atom's update. K-SVD can use any dictionary update algorithm or sparse coding algorithm desired because the two stages are separate in the pursuit algorithm that K-SVD is derived from.

Alguri and Harley [90] use the K-SVD algorithm to build an empirical wave propagation model to produce more accurate recoveries than with the analytical model. From just 24 measurements, they show that they can predict and extrapolate guided wave behavior with accuracies greater than 92%.

Bianco and Gerstoft [97] used the K-SVD algorithm to improve sound speed profile

(SSP) resolution. SSPs were modeled using empirical orthogonal functions, resulting in a very low working resolution. The resolution was improved by generating a dictionary of shape functions for sparse processing (e.g., compressive sensing) that optimally compress SSPs.

1.4 Conclusion

In this section, we address the limitations to the current methods of damage detection using analytical and hybrid modeling techniques, and the scope of the the thesis defined. A description of the methodology and an outline contents of the remaining chapters are also provided.

1.4.1 Limitations of Current Methods and Scope of Thesis

Each of the damage detection methods discussed depends heavily on the use of either baseline data from the undamaged version of the structure or a predictive model used to recover the baseline data. These predictive models are often created using fundamental equations and finite element analysis. Acoustic signals can be used to detect damage by comparing the values of features (such as natural frequency, modal shape and curvature, and flexibility matrices) to baseline values. Machine learning algorithms such as SVMs, ANNs, or genetic algorithms can also be used to create a model that can directly predict the damage state of a structure, but these also require large amounts of baseline data from the undamaged structure, as well as data from the damaged version of the structure. The latter is not often available.

The use of ultrasonics for damage detection is very similar. For all three types of data sets, (sparse arrays, full wavefield, and partial wavefield) each compares measurements to an acquired baseline, or modeled baseline, of the undamaged form of the structure. Ultrasonics have the important advantage of being more descriptive than acoustic signals because of their increased complexity, but this also makes them harder to predict and analyze. A correct model of the wave propagation is therefore even more important for the use of ultrasonic signals for damage detection.

The problem with using baseline data is that many different undamaged baseline states exist for a given structure throughout its life. These varying states are due to many differ-

ent factors, such as temperature, applied stress, and differing methods of data acquisition, all of which are not indicative of actual damage. To avoid the need to take multiple sets of baseline data for each of these healthy states or eliminate the need for baseline acquisition entirely, predictive models can be used to recover baseline data based on the current environmental state of the structure. While predictive models work well for modeling simple signals and structures, they can model only a limited amount of information from ultrasonic wavefields, and cannot accurately model wavefields of more geometrically complex structures, such as those that exist in real life applications.

This thesis addresses this problem by using an empirical model constructed using true experimental measurements. This method has been shown to more accurately model metal plates with more complex geometric shapes and more complex signal propagation caused by multipath interference. This method has also been shown to accurately recover wavefields from structures with different material characteristics than that of the structure from which the model was derived. This empirical model has also been used in a simple damage detection method to detect simulated damage on metal plates in the presence of environmental and operational variations.

While wavefield methods of damage detection have previously been limited to lengthy NDE applications, the use of compressed sensing improves full wavefield sampling speeds by a factor of over 100. The resulting sample time of 3 minutes puts this method in the realm of semicontinuous monitoring that can be used for SHM and can be used as a reliable means of screening for structural damage before more extensive NDE methods are performed.

As opposed to the work done by Alguri and Harley [90], which used simulated data to train a dictionary and evaluated the recovery data for accuracy, the current work goes a step further and trains dictionaries on empirical data and uses the dictionaries for damage detection. The data used for this work are full wavefield scans of small metal plates, acquired using an LDV data acquisition system constructed for this purpose.

1.4.2 Methodology and Outline

The following outlines the remaining chapters and briefly describes the methodology of each.

1.4.2.1 Chapter 2

Data acquisition of the full wavefield data used for empirical model construction and damage detection is described using a system built by the author. This system was built using a 2-axis stepper motor platform, a stationary mounted laser doppler vibrometer, and National Instruments software/hardware together with third party motor drivers. A validation of the system, and the specifications for the design, including hardware and software descriptions, are included in the following chapter.

1.4.2.2 Chapter 3

The empirical model described in this thesis is created using a method that combines compressed sensing and dictionary learning to converge on an accurate wave propagation model. The exact algorithm used for this purpose is the K-SVD algorithm, designed by Aharon [15]. A full wavefield data set in the low ultrasonic frequency range is used as the input to create the model. This empirical model and an analytical model are each evaluated for model accuracy using a recovery correlation. The empirical model recovers the wavefield data at above 98% accuracy, which is a significant improvement over the 20% accuracy achieved using an analytical model.

Both the empirical and analytical models are evaluated for how well they generalize to different types of material and operational variations. Both of these models, together with a third support vector machine model, are used to detect simulated damage in the metal plates. The empirical model predicted damage with 100% accuracy, while the analytical model and SVM model predicted damage with 57% and 64% accuracies respectively.

1.4.2.3 Chapter 4

Chapter 4 summarizes and discusses the above results, the advantages and disadvantages of the empirical model for use in damage detection are discussed, and future work is proposed.

CHAPTER 2

EXPERIMENTAL SETUP FOR DATA ACQUISITION

2.1 Introduction

The following chapter discusses the experimental setup for full wavefield data acquisition (DAQ) of acoustic Lamb waves. The constructed DAQ system uses hardware and software from National Instruments (NI), a 3D laser doppler vibrometer with single axis movement for focusing the laser, and a third party X-Y stepper motor stage. The system is capable of 150 mm by 150 mm scan areas.

The intent of this chapter is to provide the information necessary to allow the reader to build and operate a similar DAQ system. An appendix to this thesis is also provided to address the common system problems for troubleshooting, and additional options for expansion and modification of the system for customized use.

Included in this chapter are three sections. Section 2 contains a review of the hardware used in the system, and lists relevant specifications of the hardware. Section 3 describes the system software and the custom user interface used for control of the DAQ system. Section 4 then presents a validation of the system, and an analysis of the system performance, using the metrics of signal to noise ratio and repeatability.

The LabVIEW code and and select hardware data sheets are available by request.

2.2 Hardware Specifications

The following section reviews the hardware components necessary for the construction of the DAQ system. Manufacturers and model numbers will be provided, along with hardware specifications that are relevant to the characterization of the system. A more detailed list of specifications can be found in the individual data sheets and manuals for each component. These are typically found online on manufacturer websites and should be provided with any hardware purchase. Select data sheets are also available by request

to the author.

The system includes eight main components. An overview of the components and how they are connected is provided in Table 2.1. A System diagram can be found in Figure 2.1.

This section is divided into three subsections. Section 1 reviews the NI hardware included in the system. Section 2 lists the hardware used for data acquisition. Section 3 addresses motion control hardware. Each subsection will include a subsystem overview,

Table 2.1. Hardware components included in the data acquisition system, with listed input and output connections.

System Components and Connections

Component	Input	Output
PXI 1000 Chassis	Outlet power	(holds cards)
7344 motion card	–	to UMI
6052E DAQ card	–	to BNC
8360 interface card	–	to laptop
DC power supply	Outlet power	to UMI
UMI 7774	from power supply/motion card	to drivers
Motor drivers	from UMI	to stage
Stepper motor stage	from drivers	–
BNC 2120	from DAQ card/LDV	to piezo actuator
Piezo actuator	from BNC	–
Vibrometer (LDV)	–	to BNC

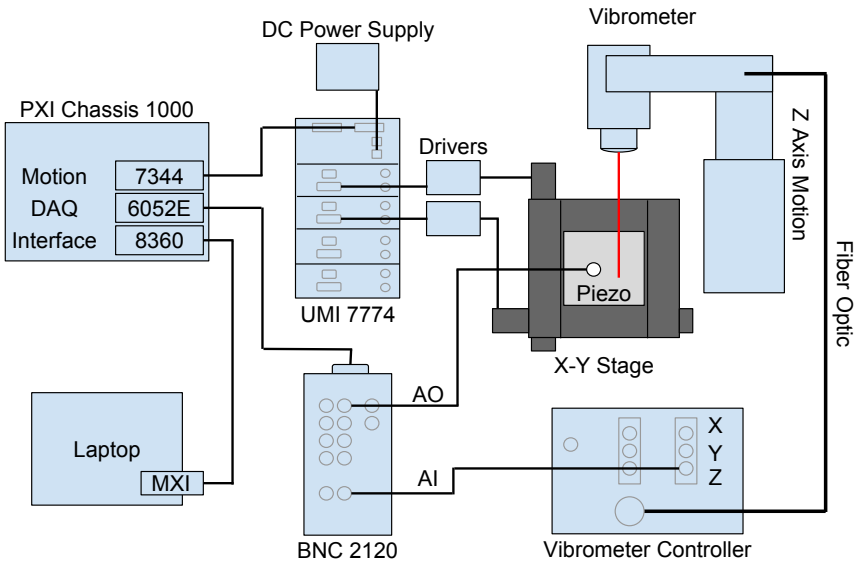

Figure 2.1. System diagram of the data acquisition system, including all of the major components and connections.

and subheadings for each component in the subsystem.

2.2.1 National Instruments Hardware

There are four components necessary for the implementation of a National Instruments (NI) system used for the current work: PXI chassis, application cards that are inserted into the chassis, interface boards that allow input/output access to the application cards, and a connection from the PXI chassis to a computer that runs LabVIEW software. The connection between the chassis and computer usually includes an interfacing application card for the chassis and some kind of express card for the computer. Laptop express cards are often inserted into an MXI card slot.

The following is a review of the hardware used in the current system.

2.2.1.1 PXI 1000 Chassis

The PXI 1000 chassis includes a system controller slot and seven peripheral module slots. One of the peripheral module slots is a star trigger slot that can provide individual triggers to all other peripherals. Star triggering is not used in the current setup. The system controller slot is not used either. There is an optional backup battery pack and mounting capabilities that are also not used. The fan speed is adjustable, but only the default speed was used. There is also a trigger bus that allows for triggering between modules. The chassis is powered through a standard AC connection, and provides and output of 150 W to the peripheral slots.

Inserted in the chassis were three application cards. In place of a built in system controller, a PXI interface card was used to interface the chassis with an external laptop. The second card was a PXI 7344 4-axis motion controller card, and the third card was a PXI 6052E legacy DAQ card. These cards will be discussed in more detail below.

A system reference clock of 10 MHz is supplied individually to each module. An external clock can be sourced, which would override the system clock. There is an independent buffer for impedance matching, with a delay of 1 ns, that drives the clock signal to the peripheral slots.

A picture of the PXI chassis can be found in Figure 2.2. A summary of system specifications can be found in Table 2.2 and system connections in Table 2.3.

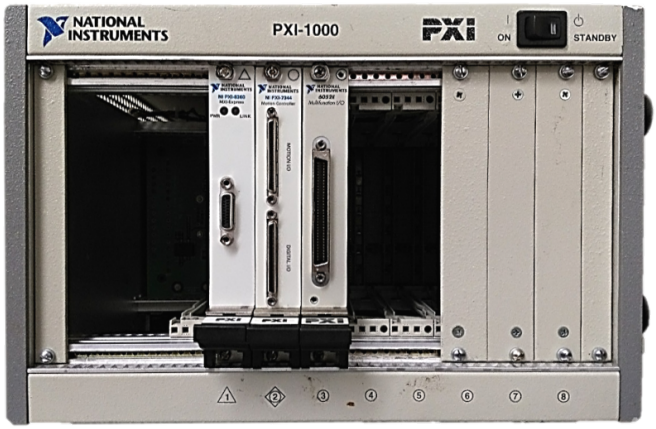

Figure 2.2. PXI 1000 chassis.

Table 2.2. Specifications for the PXI 1000 chassis.

Specifications	
Peripheral Slots	7
System Clock Speed	10 Mhz
System Clock delay	1 ns

Table 2.3. Connections for the PXI 1000 chassis.

Connections	
Power Supply	Standard 120 V AC
Star Peripheral Slot	PXI 8360 (Interface)
Peripheral Slot	PXI 7344 (Motion)
Peripheral Slot	PXI 6052E (DAQ)

2.2.1.2 Motion Controller Card: PXI 7344

The 7344 is a powerful tool for motion control. It has 4 axes of motion, and is capable of closed loop servo control, and open and closed loop control of stepper motors. This is accomplished through Pulse Width Modulation (PWM) signals of $\pm 10V$. Closed loop control can be done using quadrature encoders or analog inputs and triggers. All stepper axes support full, half, and micro stepping applications.

The 7344 controllers are capable of performing arbitrary and complex motion trajectories. These motion profiles are controlled with enhanced PID/PIVff servo updates at 62 μs per axis. Each axis has motion input/output (I/O) ports, end-of-travel and home limit switch inputs, and encoder feedback rates up to 20 MHz. There are also four analog inputs

which have a multiplexer scans rate of 50 μs, which can be used for $\pm 10V$ signals, feedback inputs, joystick inputs, monitoring of analog sensors, and 32 bits of nondedicated digital I/O.

The 7344 uses a real-time 32 bit CPU, and can perform up to 10 simultaneous motion programs. real-time system integration (RTSI) bus allows for high speed connectivity and synchronization of image and data acquisition devices.

The 7344 motion card is inserted into a peripheral slot of the PXI 1000, and is connected to a motion interface board through two 68-pin digital I/O connectors. A summary of system specifications can be found in Table 2.4.

2.2.1.3 Universal Motion Interface (UMI): UMI 7774

The UMI is an interface board used to connect the 7344 controller to the motion system. It is simply used to provide access to the I/O capabilities of the controller card. The board is powered to allow switches and digital I/O to toggle high and low, to power LED indicators, and to provide power to the connected motor drivers. There is an option for optically isolated power used in the optical isolation output circuitry for the drive enable outputs and digital output signals. The specifications for the board are the same as the 7344 controller because the UMI just relays information.

There are power, disable, enable, and limit switch LED indicators to indicate the state of corresponding ports on the board. The 4 axes each have both a feedback and control connector. There are also toggle switches that can be used to set the fault, enable, and limit LEDs active high or low. There is a global stop connector, as well as trigger/break point and digital I/O connectors.

Table 2.4. Specifications for the PXI 7344 motion card.

Specifications	
Number of Control Axes	4
Axis Update Rate	62 μs per axis
Max Quadrature Frequency	20 MHz
CPU	Real-time 32 bit
Analog inputs	4
Analog scan rate	50 μs per enabled ADC channel
Digital I/O	32 bits

For the current system, only the open loop stepper control was used. Encoders were therefore not wired, and neither were limit switches, though the stepper motor stage had both of these components. Details are discussed more below in the section on the stepper motor stage.

The UMI requires a 24 VDC power supply, which will be described later in this section. A picture of the UMI can be found in Figure 2.3.

2.2.1.4 Data Acquisition (DAQ) Card: PXI 6052E

The 6052E is fairly capable DAQ device. It has a midrange sampling rate and numerous analog and digit I/O ports and a very high resolution. Digital and analog triggering are available on this card. However, this card is a legacy model and is no longer currently produced. The PXI 6251 could be a good replacement that is currently available and has a higher sampling rate.

For the current system, only one analog output and one analog input were used. This is all that was necessary to excite the piezoelectric actuator and receive a reading from the vibrometer.

The 6052E DAQ card is inserted into a peripheral slot in the PXI 1000 chassis, and is connected to a DAQ I/O board using a 68-pin conductor cable (NI part SH68-68-D1). A summary of system specifications can be found in Table 2.5.

2.2.1.5 Data Acquisition Interface: BNC 2120

The BNC 2120 is a terminal block for the the chosen DAQ card and simply provides an easy to use interface for the inputs and outputs of the card. There is no power terminal to

Figure 2.3. Universal motion interface (UMI 7774).

Table 2.5. Specifications for the PXI 6052E DAQ card.

Specifications	
Max Sampling Rate	333 kS/s
Output Rate	333 kS/s
Input Resolution	16 bits
Output Resolution	16 bits
Input Range	$\pm 10V$
Output Range	$\pm 10V$
Analog Outputs	2
Analog Inputs	16 SE/8 DI
Digital I/O	8
Counter/Timers	2, 24 bit

the board. There are a few indicator LEDs that indicate the state of some of the terminals, options for toggling analog channels to either floating or ground ground sources, and controls for a function generator section. The connectors in the board correspond to the inputs and outputs of the the chosen DAQ card.

In the current system, the 68 conductor cable (NI part SH68-68-D1) used above connects this device to the 6052E. A picture of the BNC 2120 can be found in Figure 2.4.

2.2.1.6 PXI Interface Module: PXI 8360

The 8360 refers to the PXI interface module application card that is inserted into the PXI, and the express card that is inserted into the computer. This express card can either be an MXI interface, which is inserted into an MXI slot on a laptop, or a card that can be inserted into a PCI slot on a desktop mother board. These all allow for a connection between the PXI

Figure 2.4. Data acquisitions interface (BNC 2120).

chassis and a computer with LabVIEW. This connection eliminates the need for a system controller in the PXI. In the current system a laptop MXI slot was used to interface with the PXI. These interface modules are capable of 110 MB/s sustained throughput.

2.2.2 Data Acquisition Hardware

There are three main components of data acquisition hardware included in the current system: the laser doppler vibrometer (LDV), piezoelectric actuators, and a standard vibration isolation table. The actuator is used to excite the dynamics of the structure, the LDV to measure the propagating Lamb waves along the surface of the structure, and the isolation table to damp outside vibrations from interfering.

Piezoelectrics can be used as simple, consistent, and inexpensive actuators that can transmit acoustic signals such as swept sinusoidal waves. They are useful for the current application, because a consistent excitation is needed for successful SHM applications. Variations in the excitation of the structure might lead to an operational variation that results in a false positive detection of damage.

LDVs can come in many forms, but generally consist of three components: a controller, which generates the laser, processes the reflected laser, and measures the frequency shift; a sensor head that focuses the laser and receives the reflected light from the specimen; and a heavy duty fiber optic cable that tethers the two. Most of the system specifications for the LDV system, such as the frequency range of operation and sensitivity, are determined by the controller chassis. Polytec controller modules can be selected for different types of demodulation, input and output connectors, internal laser units, and include integrator modules and even remote control modules. These are each chosen to match specific measurement needs and inserted into the controller chassis. The sensor head can then be attached to the controller, and the controller to analog input on your DAQ device. Sensor head specifications typically include standoff distance and spot diameter.

The isolation table is good practice, and can be used to ensure that outside frequencies within a certain bandwidth do not interfere with measurements. The isolation mounts used with the current system are pneumatic isolation mounts of type XL-B from Newport Cooperation.

2.2.2.1 Laser Doppler Vibrometer (LDV)

The current system uses a Polytec 3D laser vibrometer system with a CLV 3000 controller chassis, and a CLV 3D sensor head. Three CLV-M200 input modules with three CLV-M030.B decoder modules are included. The geometry calculation and output module CLV-M000.G1 is used. A monitor output module CLV-M000.M is also used to allow access to the individual laser beam velocity signals before calculating the X, Y, and Z axis velocities.

Though this is a 3D vibrometer, the vibrometer was only used in a single-point configuration for Z axis vibration measurements. The Polytec 3D laser vibrometer system promises high sensitivity even without a reflective surface.

It should also be noted that the LDV was mounted to the vertical test stand, which is sold optionally with the 3D LDV from Polytec, and this test stand was bolted to the breadboard table top of the isolation table to keep the setup from moving.

The Z axis signal output on the LDV controller is connected to an analog input on the BNC 2120 using a BNC cable. A picture of the vibrometer can be found in Figure 2.5, and the LDV controller in Figure 2.6. A summary of system specifications can be found in Table 2.6.

Figure 2.5. Polytec 3D laser vibrometer.

Figure 2.6. Polytec laser vibrometer controller.

Table 2.6. Specifications for the Polytec 3D laser vibrometer.

Specifications	
Controller	
Frequency Range	0.5 Hz to 250 kHz
Sensitivity Range	0.1, 0.5, 2.5 $\frac{m}{s}$
Scaling factor	5, 25, 125 $\frac{mm/s}{V}$
Filter	10 Hz high pass
Analog Velocity Output	$\pm$10 V
Sensor Head	
Laser Type	Helium-Neon (HeNe), 633 nm (red)
Laser Output	$<$1 mW per beam
Laser Beam Incline	12°
Standoff Distance	140 mm
Spot Size	65 μm

2.2.2.2 Piezoelectric Actuators

Piezoelectrics can come in the form of small, round, layered ceramic wafers with positive and negative leads. These can then be glued to the structure and connected to any analog voltage to create a vibration. They can also be embedded into a housing, such as those sold by Vallen Systeme: www.vallen.de/. These can be coupled to the structure using coupling fluid and removed and reapplied easily.

The sensors used in the current system are permanently attached wafers purchased from Steminc-Piezo, part number SMD07T02R412WL. The glue used to adhere the sensors was a combination of medium 5-15 second flexible INSTA-FLEX+^{TM} cyanoacrylate from Metra Electronics Corporation and an INSTA-SET^{TM} accelerator form Bob Smith Industries. The INSTA-FLEX^{TM} can be applied to the structure, and the INSTA-SET^{TM} to the

sensor. When they come in connect, the glue hardens quickly to adhere the sensor to the structure. The only connection necessary for the piezoelectric sensor is to the DAQ analog output from the NI BNC 2120. This is done using a BNC to banana clip cable to allow for quick switching between sensors being used to actuate.

A picture of the piezoelectric actuator can be found in Figure 2.7. A summary of system specifications can be found in Table 2.7.

2.2.3 Motion Control Hardware

The motion control hardware includes three parts, an X-Y stage, motor drivers for the stage motors, and a DC power supply to power the drivers. The stage that was chosen for the current system uses stepper motors for precise and repeatable movement, even without encoder feedback. There are two drivers, one for each motor, which are controlled by the NI UMI motion interfacing board. These drivers drive each stepper motor separately for X-Y movement of the stage. The DC power supply used in the system is also used to power the the UMI board.

2.2.3.1 Stepper Motor Stage

The stepper motor stage used in the current system is built by Ludl Electronic Products Ltd. The stage has 2-axes with an optional third rotation axis in the middle, which is not

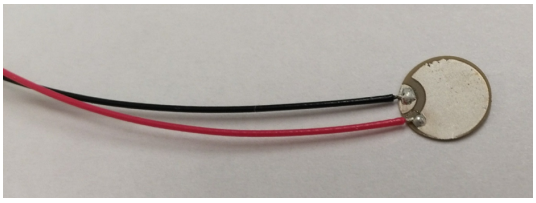

Figure 2.7. Steminc-Piezo piezoelectric actuator (SMD07T02R412WL).

Table 2.7. Specifications for the Steminc-Piezo piezoelectric actuator (SMD07T02R412WL).

Specifications	
Piezo Material	SM412
Dimensions	7mm dia. x 0.2mm thickness
Resonant frequency	300 KHz$\pm$10 KHz
Electromechanical coupling coefficient Kp	$\geq$55%
Resonant impedance Zm	$\leq$10.0
Static capacitance Cs	3000pF$\pm$15%@1kHz

included on this model. There are forward and backward limit switches on both the X and Y directions, and rotary encoders on both of the motors. There are also "SmartStage" memory chips for each axis that are not currently used.

Only the motors are wired in the current system. The limit switches and encoders are not wired for current use. As is, the stepper motors provide very accurate and repeatable control. There has been no noticeable skipping of steps (drifting of the motors) on the stage after numerous 10000 points scans.

The stage was aligned with the LDV by bolting the stage to the breadboard table top of the isolation table. Both motors on the stage are connected to stepper motor driver described below, providing 2 phase control of the motors.

A picture of the stage can be found in Figure 2.8. A summary of system specifications can be found in Table 2.8.

2.2.3.2 Motor Drivers

The motor driver chosen was a TB6600, which has six selectable resolutions ($\frac{\text{pulses}}{\text{revolution}}$) and 8 selectable currents. It can be powered by a large range of voltages and deliver high current to the motors or limit the current when necessary. There is also an enable port, high speed photoelectric isolation of the input signal, casing to protect the circuitry, and built in heat sink. It also has built in thermal and voltage protection circuits. These qualities allow the driver to work in a variety of different situations and drive many different types

Figure 2.8. Ludl Electronic Products Ltd. stepper motor stage.

Table 2.8. Specifications for the Ludl Electronic Products Ltd. stepper motor stage.

Specifications	
Stage	
Travel Range	8 inches by 8 inches
Lead Screw	4mm pitch
Resolution	0.4 micron
Accuracy	$\pm$5 microns/200mm
Repeatability	$\pm$1.5 micron
Stepper Motors	
Type	Permanent Magnet, 4-phase bipolar
Step Count	200 steps/revolution
Current draw	1A per phase, 4A
Encoders	
Type	Relative Rotary
Counts per Revolution	2500, quadrature to 10000

of motors. For our system, a current of 0.5 amperes was used and a step resolution of 200 pulse/rev.

A picture of a driver can be found in Figure 2.9. A summary of system specifications can be found in Table 2.9 and system connections in Table 2.10.

2.2.3.3 DC Power Supply

The power supply used was built by Condor D.C. Power Supplies Inc., model HB24-1.2-A+. The supply can be powered by 100 to 240 VAC at standard frequencies. The output voltage is 24V, and output current is 1.2 A. The output current proved to be sufficient for

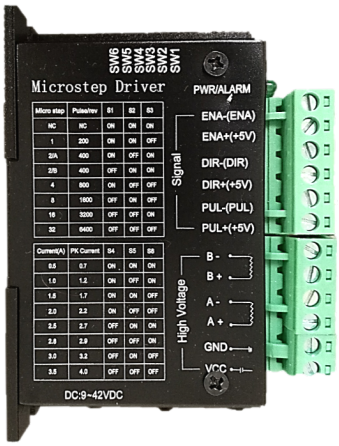

Figure 2.9. Stepper motor driver (TB6600).

Table 2.9. Specifications for the motor driver (TB6600).

Specifications	
Max voltage	50 V
Current Range	0.5, 1, 1.5, 2, 2.5, 2.8, 3, 3.5 A
Resolution Range	200, 400, 800, 1600, 3200, 6400 pulse/rev

Table 2.10. Connections for the motor driver (TB6600).

Connections	
Power	9V-40V DC
To UMI	Pulses (PUL)
To UMI	Direction (DIR)
To UMI	Enable (ENA)
To motor	A+, A-, B+, B- coil wires, each driver

the application, but the ideal power supply would be able to supply the total 4 A for each motor and 1.5 A for the UMI board. This kind of current draw is not likely necessary for normal use of the system.

2.3 Software Specifications

A review of the software specifications of the DAQ system are discussed in the following chapter. Nation Instruments (NI) Laboratory Virtual Instrument Engineering Workbench (LabVIEW) was used to interface the NI hardware to the motion stage and laser doppler vibrometer (LDV).

Section 1 will include a description of the DAQ program and its various components. The code created for this system can be provided at request to the author. Section 2 describes the graphical user interface used to control the system.

2.3.1 Data Acquisition Program

The following is an outline of the pieces used to build the LabVIEW program for the system. A bare bones event capture program was created to capture bottom clicks on the front control panel. These buttons are **Initialize**, **Global Stop**, **Step**, **Home**, and **Get Data**. **Initialize** disables the limit switches and clears the power status of the motion card so that it is ready to use. **Global Stop** terminates the entire program, but it can only be used between event captures. If a hard stop is required while an event is still being

processed (this most often happens during the **Get Data** event), the "Abort Execution" button provided next to the LabVIEW "Run" button can be used. Use of "Abort Execution" should be avoided especially during used of the **Get Data—** function as to not lose track of the current stage position, but is available if needed.

Step uses the example VI One_Axis_Move to move the X or Y axis either relative to the current position (relative movement), or relative to the home position (absolute movement). The **Step** command can be used to position the stage in the a new home position and is used in conjunction with **Home**. All movements in this program are commanded in millimeters, and converted to stepper step counts within the program. **Home** will set the X and Y motion axes to zeros, which is the start position of the X-Y inspection grid movement.

Get Data is by far the most complex event. It prompts a "Get User Input" dialog box where the parameter values for the scanning grid and data acquisition are entered, such as grid size, step size, DAQ frequency, and excitation signal type. These parameters are then used to create and travel along an X-Y grid, and are also fed into the DAQ sub-VI that is executed at each step in the grid. The grid motion control and DAQ VI will be described in more detail below.

The user interface for the data acquisition program can be seen in Figure 2.10 and the block diagram view of part of the code for the **Get Data** function can be found in Figure 2.11.

2.3.1.1 Motion Control Virtual Instrument

The example X-Y Grid Inspection (Template) was used as a foundation for the motion control VI. This can be found by going to Help-Find Examples, then searching for the name. This example VI takes inputs of grid size, steps size, ramp up/down velocity, and ramp up/down acceleration, and outputs corresponding PWM signals to the X-Y motors to move in a grid fashion. The entire example VI was copied and pasted into the event capture program described above. A conditional box is provided in the example where a measurement can be taken at each position before moving to the next step. The DAQ VI described below was inserted here in sequence with a 100ms wait command to always allow for enough time for the stage to stop moving and the data acquisition to

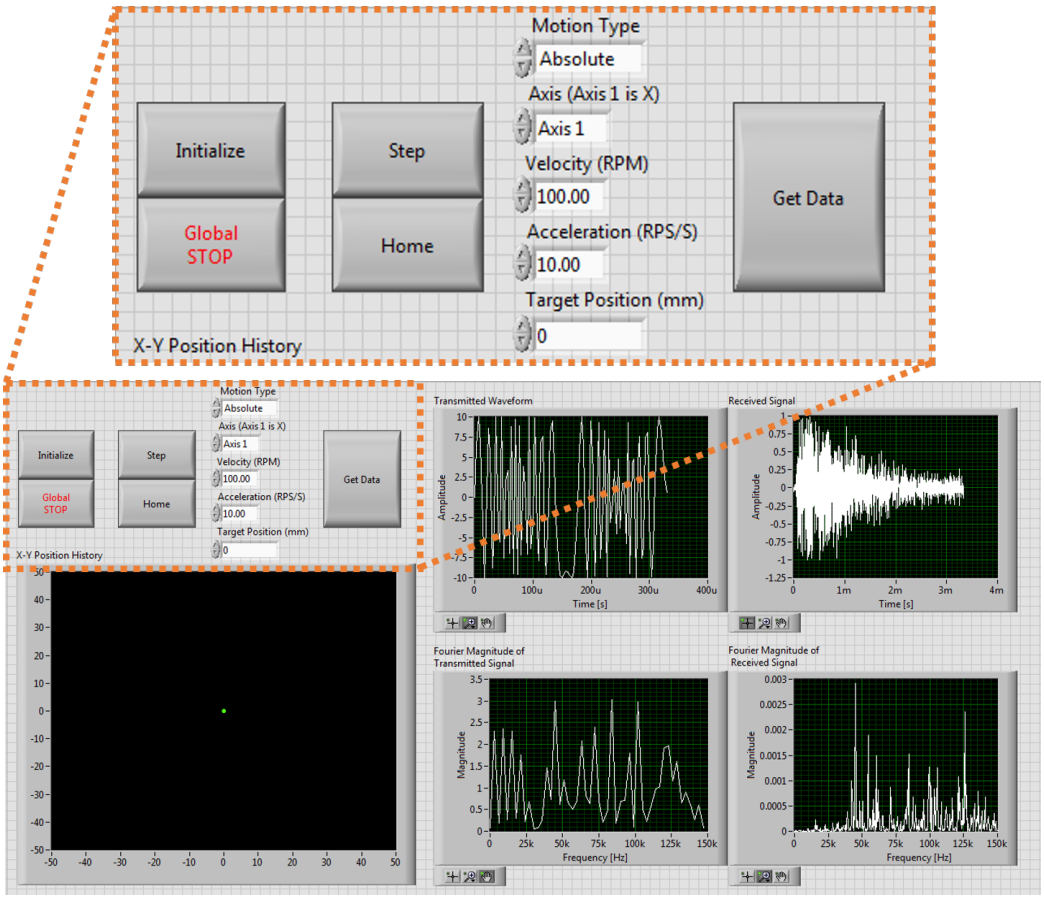

Figure 2.10. Graphical user interface for data acquisition system.

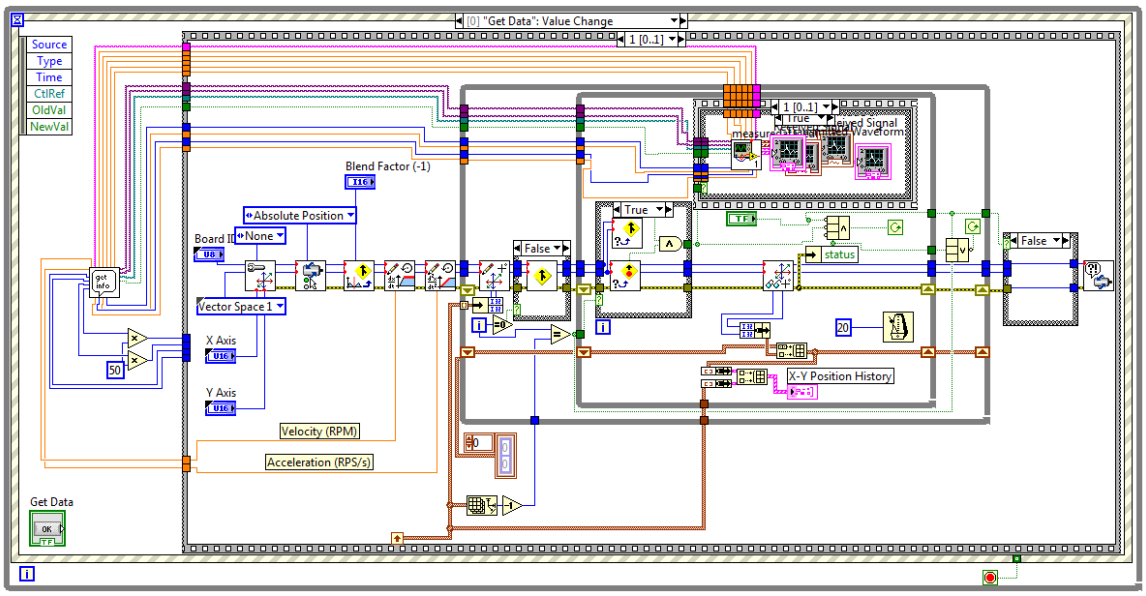

Figure 2.11. Box diagram for a portion of the data acquisition code that is launched when the **Get Data** button is pressed.

take place before the next step. This wait command may need to be increased if a longer data acquisition time is needed. A longer acquisition time would be required if longer excitation signals or more averaging are used.

2.3.1.2 DAQ Virtual Instrument

The DAQ VI used is a VI that has been used by Professor Joel Harley's research group for other applications, and was provided by him. This program allows for options of waveform type, length of transmitted signal, number of measurements, length of received signal, sampling rate, transmitter max voltage, receiver max voltage (used to control resolution), number of averaging measurements, and measurement file save location. The default used for the current application is a swept sinusoidal chirp from 1 to 150 kHz with a signal length of 1000 samples and excited at $\pm$10 V. The signal was sampled at 300 kHz for 3000 samples, using a max of $\pm$5 V for higher resolution measurements, and 100 averaging measurements at each location. This VI was also capable of filtering, but filtering was not use in our application.

2.3.2 User Interface

The user interface button functions have been described somewhat already, but there are a few more things to mention concerning the user interface (see Figure 2.10). The input boxes, "Motion Type," "Axis," "Velocity," "Acceleration," and "Target Position" are all used to determine what kind of motion will occur when using the **Step** button. "Motion Type" allows the user to set an absolute or relative step. "Axis" determines whether the step will be along the X or Y axis. "Velocity" determines how quickly the stage will move. "Acceleration" determines how quickly the stage gets up to speed, and back to zero. "Target Position" is how much the **Step** command will move the stage along the current axis, relative to the current position (relative motion type), or the home position (absolute motion type). Both positive and negative values can be entered for "Target Position" to move up and down an axis.

A slow ramp up velocity (acceleration) is important to keep the stepper motors from over/under stepping do to momentum. However, an acceleration that is too slow significantly increases the scan time, because it must be repeated as many times as there are points to scan. Default values of velocity = 100 RPM and acceleration = 10 RPS/S are used.

These values allow for smooth, quick, and repeatable movements that do not cause the motors to drift or miss step.

There are five indicators on the control panel. The first is an X-Y position history which shows the grid being used for a grid inspection during a scan. The scan progress is tracked using a white line drawn from point to point. A transmitted signal waveform and its Fourier transform are displayed, as well as the received signal and its Fourier transform.

When the **Get Data** button is pressed, a "Get User Input" window opens. This dialog box can be seen in Figure 2.12; the "Data" and "Material" boxes will be used to name two files, one with the data (date_material.csv) and the other a description of the data, containing all of the data input into this window (date_material_desc.csv). The description file can be used to later process the data file correctly.

Inputs for the grid inspection are also available. "Grid Dim. X" and "Grid Dim. Y" determine the number of division in the grid in the X and Y directions, and "Step Size X" and "Step Size Y" determine the distance between scan points. If both grid dimensions are set to 99 by 99 divisions and step sizes to 1 mm each, the resulting grid will be 99 by 99 mm, and 10,000 points at 100 points by 100 points.

Parameters for the excitation signal (output), and acquisition of the signal (input) are also entered here, and the physical channels used for each are selected. Physical parameters of the scan can be entered here, such as material dimensions, material thickness, excitation sensor location, and start scan location on the material. These will each be stored in the the description file for the scan. There is also an input box for the file location path, and a save data toggle. Pressing the OK button starts the scan.

2.4 System Validation and Performance

It is important to validate the DAQ system and ensure that it is working correctly before use. If the system is not exciting the correct signal and measuring the responses that we would expect, then the system can't be used for damage detection. It is also important to understand the limitations of the data acquired through the system. These issues are addressed in this section. Included in this section is a validation of the DAQ system, and a performance analysis which considers both the signal-to-noise ratio and repeatability of the measurements. Future work on the system is also proposed.

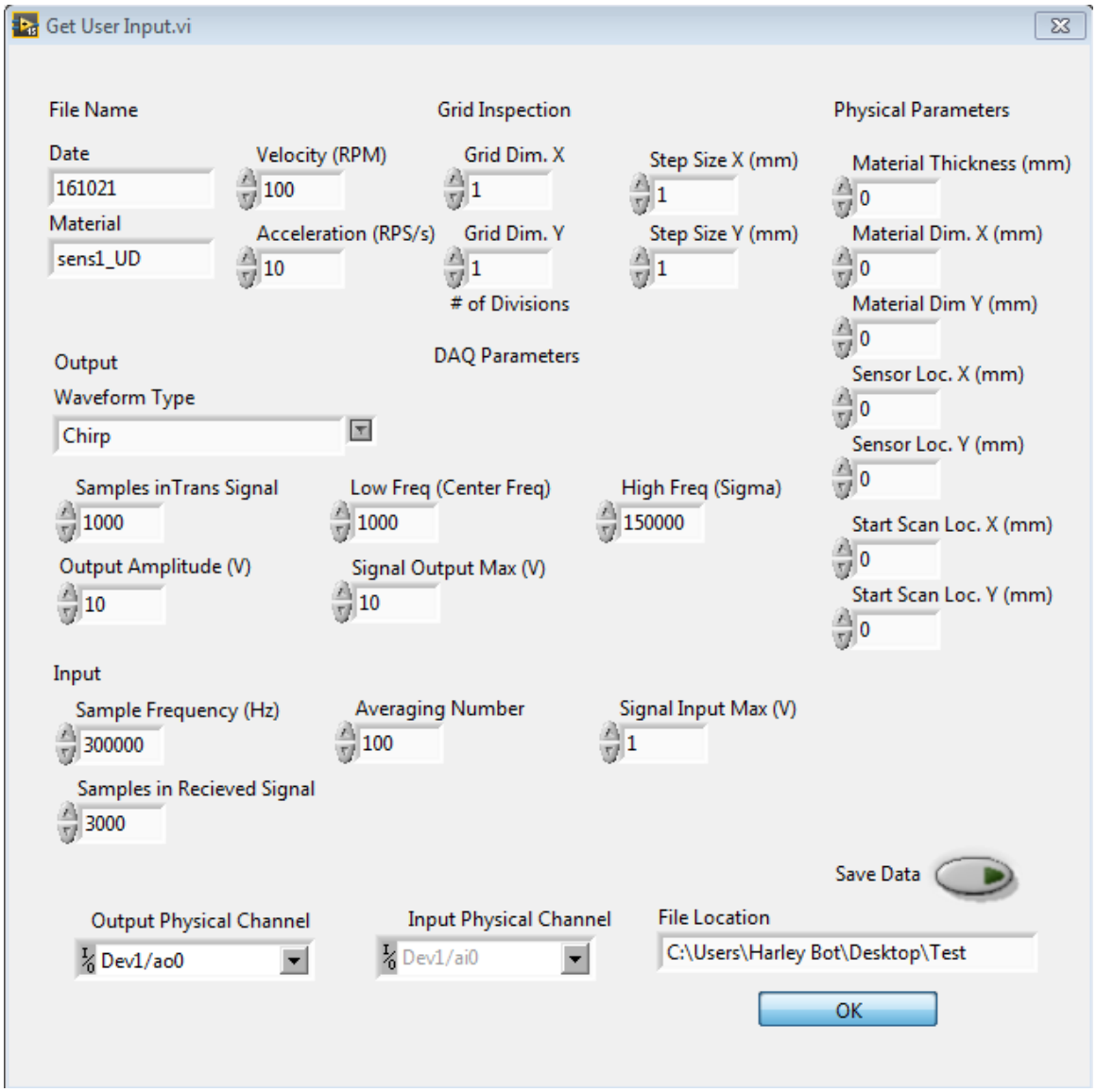

Figure 2.12. "Get User Data" dialog box.

2.4.1 Validation

The DAQ system is validated by comparing the wave velocities measured by the system to theoretical wave velocities for a signal of the same frequency traveling through the same material. To accomplish this a 100 kHz frequency signal was excited in an aluminium plate, using a piezoelectric actuator, and measured 60 mm away. The walls of the plate were sufficiently far away that the zeroth symmetric mode (S0) mode and zeroth antisymmetric (A0) mode, the dominant modes in a signal of this frequency, could both travel to the point of measurement before the reflected waveforms could interfere. The group velocity of the S0 mode and A0 mode excited at 100 kHz in a 1.6 mm aluminium plate are roughly 5.5 $\frac{mm}{\mu s}$ and 2 $\frac{mm}{\mu s}$ respectively [60]. Traveling at those velocities, the S0 mode should begin to be seen at 11 μs, and the A0 at 30 μs.

The waveform measured using the DAQ system is shown in Figure 2.13. It can be seen that the A0 mode lines up very well with the theoretical arrival time of the mode. The S0 mode, however, is too faint to be seen at such a low frequency. The alignment of the A0 mode suggests that the system is in fact exciting the correct signal in the material and measuring the true waveform.

Another validation can be performed using a full wavefield image acquired using the system. In Figure 2.14, three wavefield snapshots can be seen, taken at three different times: 23.33, 30, and 36.66 μs. A wavefield peak was tracked in each of these snapshots

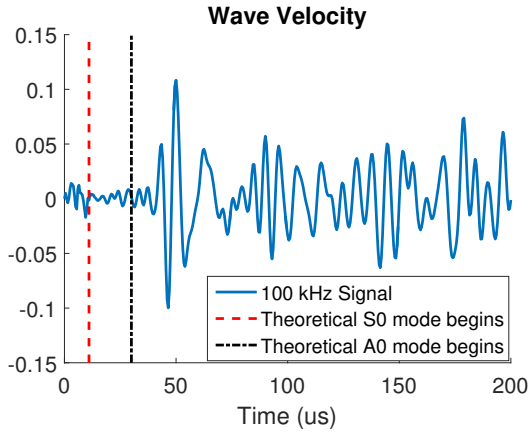

Figure 2.13. A 100 kHz waveform sampled at 2 MHz. The zeroth antisymmetric (A0) is clearly visible, but the zeroth symmetric mode (S0) is too faint to be seen. A0 mode appears to arrive close to its theoretical arrival time.

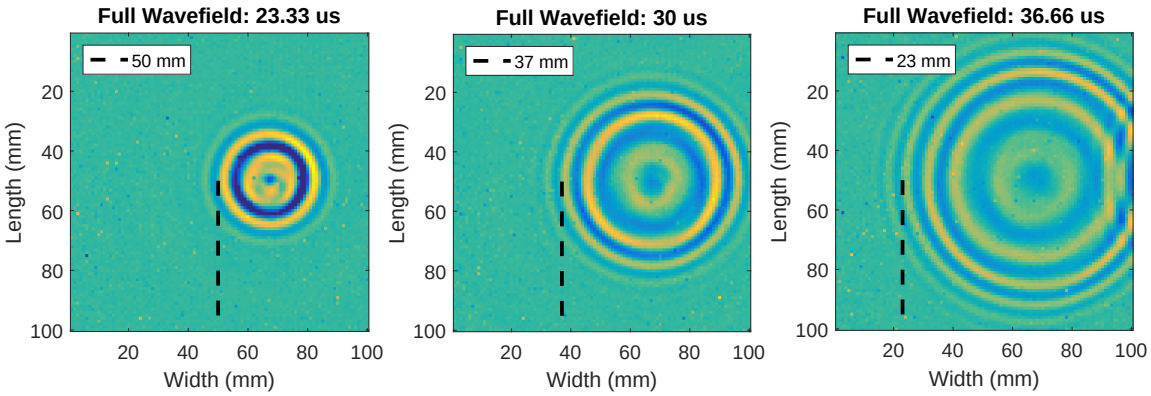

Figure 2.14. Snapshots of a full wavefield scan at 23.33, 30, and 36.66 μs, with a peak tracked through each one. The peak is moving at 2.03 $\frac{mm}{\mu s}$, well within normal ranges for a chirp of 1 kHz to 150 kHz. The color maps range from -0.3 V (dark) to 0.3 V (light).

and found at 50, 37.5, and 23 mm respectively. The wave traveled 13.5 mm in 6.66 μs with a velocity of 2.03 $\frac{mm}{\mu s}$. This is within a reasonable range because the wavefield is excited by a swept sinusoid from 1 kHz to 150 kHz, and A0 modes could travel anywhere from < 1 $\frac{mm}{\mu s}$ to just over 2.5 $\frac{mm}{\mu s}$. The complex combination of wave propagation modes makes it hard to determine exactly how fast the peak should be moving, but 2.03 $\frac{mm}{\mu s}$ is reasonable. Also note in Figure 2.14 that the wave propagation is clear and well defined.

The excitation signal (swept sinusoidal signal, or chirp) used in Figure 2.14 is the same used for all of the scans examined in Chapter 3 of this thesis. Figure 2.15 illustrates what a swept sinusoid (1-150 kHz) excitation and a corresponding measured signal look like. The excitation signal shown here changes amplitude towards higher frequencies because the sampling rate used to excite the signal was barely double the highest frequency present in the signal. The frequencies are preserved in the signal, but the amplitudes vary somewhat.

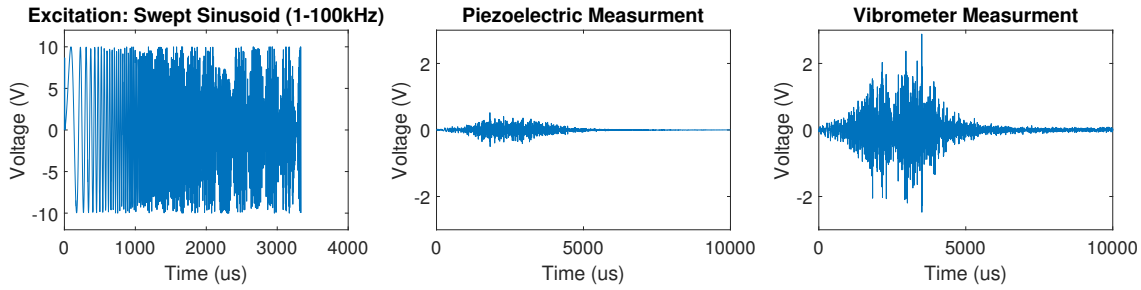

Figure 2.15. Swept sinusoidal signal and the corresponding measured waveforms using both a piezoelectric sensor and a vibrometer.

The swept sinusoid is excited at $\pm$10 V and is captured at a lower voltage due to the sensitivity of the sensor. A piezoelectric sensor often outputs measurements in the range of $\pm$0.5 V, and a vibrometer $\pm$3 V. The vibrometer is more sensitive to the excited signal, but it will be shown later that the piezoelectric sensor has a better signal-to-noise ratio and better repeatability than the vibrometer. However, measuring a full wavefield with piezoelectric sensors would be very hard, time consuming, and inflexible. A Pearson correlation between the piezoelectric measurement and the vibrometer signal shows only a 5% correlation. The captured signals are obviously different and should not be used interchangeably.

2.4.2 Performance

To quantify the performance of the DAQ system, signal-to-noise ratio (SNR) and repeatability were calculated using single point measurements. After single point measurements are considered, the SNRs of full wavefields are calculated. These wavefields were taken from different metals with varying thicknesses. The repeatability of a full wavefield scan is also addressed. The SNR and repeatability of a full wavefield is found to be very similar to that of a single point measurement.

To create the data set used to calculate SNR and repeatability, a single point was chosen on an aluminium metal plate of 1.6 mm thickness. A piezoelectric sensor was adhered to the bottom of the plate at that point and the the laser of the vibrometer focused on top. The excitation was measured five times by each sensor at 10 V, and an additional five times at 5 V each, totalling 20 measurements. The plate was then removed and remounted with mounting tape and placement guides. The 10 V measurements were then repeated (10 measurements), then the plate was reset again. The plate was reset a total of four times for five different sets of 10 V measurements, totalling 50 measurements plus the ten 5 V measurements. The excitation signal used was a swept sinusoid (1-150 kHz). This performance data set is further described in Table 2.11.

2.4.2.1 Signal-to-Noise Ratio

A noise signal set was created by taking the measurement from both the piezoelectric sensor and vibrometer without an excitation, repeated five times (10 measurements). The SNR was calculated in MATLAB using these noise signals, and signals from the perfor-

Table 2.11. Signal set used for evaluating the signal-to-noise ratio and repeatability of the DAQ system measurements. This data set consists of single point signals, all of which were acquired on a 1.6 mm thick aluminium plate using a swept sinusoidal (1-150 kHz) excitation.

Performance Data Set

Indices	Sensor	Voltage
1-5	Piezo	10 V
6-10	Piezo	5 V
11-15	Vibrometer	10 V
16-20	Vibrometer	5 V
	...reset plate...	
21-25	Piezo	10 V
26-30	Vibrometer	10 V
	...reset plate...	
31-35	Piezo	10 V
36-40	Vibrometer	10 V
	...reset plate...	
41-45	Piezo	10 V
46-50	Vibrometer	10 V
	...reset plate...	
51-55	Piezo	10 V
56-60	Vibrometer	10 V

mance data set described above.

An SNR is calculated for each signal with each of the five corresponding noise signals, resulting in five SNR values per signal. The average of all of these values for all of the signals and their standard deviation (STD) are listed in Table 2.12. The SNR is computed using the ratio of the summed squared magnitude of the signal to that of the noise. The noise was acquired by taking a measurement without exciting a vibration in the plate.

Table 2.13 lists SNR values for full wavefield scans taken on both aluminium and steel, with thicknesses of 1.6 and 6.35 mm. This is done by finding five SNR values for each

Table 2.12. Average signal-to-noise ratio (SNR) and standard deviation (STD) for a large set of signals taken using the DAQ system.

Single point SNR

Sensor	Voltage	SNR	STD
Piezo	10 V	28.96 dB	0.31 dB
Piezo	5 V	23.88 dB	0.31 dB
Vibrometer	10 V	21.36 dB	1.96 dB
Vibrometer	5 V	15.16 dB	1.96 dB

Table 2.13. Average signal-to-noise ratio (SNR) and standard deviation (STD) for all of the signals found in a full wavefield data set. A 10 V swept sinusoidal excitation signal (1-150 kHz) was used, and the measurement was taken using a vibrometer.

Full Wavefield SNR

Material	Thickness	SNR	STD
Aluminium	1.6 mm	24.42 dB	2.01 dB
Aluminium	1.6 mm	26.78 dB	2.01 dB
Aluminium	6.35 mm	17.12 dB	2.14 dB
Steel	1.6 mm	23.00 dB	2.17 dB
Steel	6.35 m	3.18 dB	2.06 dB

signal using each of the five noise signals, resulting in 50000 SNR values. The average of all of these, and their STD, are reported. The SNR values for the 1.6 mm thick plates are of a similar value to those found for single point signals using the 10 V vibrometer measurements, which were also measured on a 1.6 mm thick plate.

2.4.2.2 Repeatability

The repeatability of the system is evaluated by taking Pearson correlations between signals measured both with and without resetting of the plate. The plate is reset by removing it from the stage and then remounting it using mounting tape and placement guides. The process of resetting the plate is a significant source of operational variability that is present in the system, and it is important to note the effect that it has on the repeatability of the signal measurements.

From the performance data set described above, only the 10 V signals are used here. Each group of ten signals is separated by a resetting of the plate, five for one sensor and five for the other. To evaluate the repeatability of the measurements without resetting the plate, each group of five is correlated with themselves in every possible pair. To evaluate the repeatability of the measurements with resetting of the plate, every unique combination of signals from different groups is correlated. The averages and STDs of these two sets of correlations are found in Table 2.14.

The piezoelectric sensor has an almost perfect repeatability with a 99.58% average correlation, which is 5% higher than the vibrometer. This is likely because the piezoelectric sensor is glued to the plate, and has no real chance of moving. The vibrometer also contains slightly more noise than the piezoelectric, which could add to the difference. When the

Table 2.14. Average correlation between repeated signals with and without resetting the plate in between scans, and the corresponding standard deviation (STD).

Single Signal Repeatability

Sensor	Reset?	Correlation	STD
Piezo	No	99.58%	0.41%
Piezo	Yes	95.01%	2.52%
Vibrometer	No	96.02%	0.30%
Vibrometer	Yes	89.24%	4.53%

plate is reset, the average correlation for each sensor drops 5-7%, a somewhat uniform amount. This is likely caused by the change in how the plate is adhered to the stage. The vibrometer likely drops a few percent extra, 7% instead of 5%, because resetting the plate causes the laser to be focused on a slightly different location than it was before, while the piezoelectric sensor never moves.

The repeatability of a full wavefield scan was evaluated by correlating every signal in the first wavefield with every signal in a second wavefield in a single correlation. Both wavefield were taken using the exact same 1.6 mm aluminium plate, piezoelectric excitation sensor, and swept sinusoid signal (1-150 kHz). The plate was reset in between scans. These two wavefields had a correlation of 86.38%, 3% lower than the single signal correlation of the same type, but still within one standard deviation.

2.5 Conclusion

This chapter has described in detail the hardware and software components used in the construction of a data acquisition system for full Lamb wavefield data. To accomplish this, we have used a National Instruments motion control system and DAQ card, a laser doppler vibrometer, and a 2-axis stepper motor stage. The system was validated using wave velocity measurements, and the performance was measured using signal-to-noise and scan repeatability metrics.

In Chapter 3 of this thesis, the wavefields that have been acquired using this system are used to explore the dictionary damaged detection technique introduced in Chapter 1.

CHAPTER 3

DAMAGE DETECTION USING DICTIONARY LEARNING

3.1 Introduction

Sparse wavenumber analysis (SWA) has demonstrated the use of an analytical dictionary for the recovery of Lamb wave dispersion curves [7]. This method successfully separates different modes of vibration and measures their amplitude and velocity. This result is achieved using sparse coding [74], which is facilitated by the sparse nature of Lamb waves in the frequency-wavenumber domain. SWA assumes that the propagation medium has infinite boundaries and that only incident waves are present. This is a powerful result that can be used to detect damage on simple structures, but Lamb waves on real structures have added complexity due to multipath interference. This complexity makes it very difficult to use SWA to detect damage on real world structures.

In this chapter, we model this multipath interference by learning a dictionary using empirical Lamb wave data. This learned dictionary is constructed using the K-SVD algorithm that will be addressed later in this chapter [15]. The result of sparse coding using this learned dictionary is a sparse representation of the signal set that is similar to the dispersion curves recovered using SWA. However, this sparse representation no longer represents the velocities and amplitudes of the different modes directly, but a linear combination of base patterns found in the signal set. Though this new frequency-"pattern" domain is only loosely connected to the frequency-wavenumber domain, Lamb waves are expected to be sparse in this new domain as well. This is because the frequency-pattern domain represents a combination of incident and reflection waves which are both individually sparse in this framework.

Dictionary learning produces a dictionary that can model the complex relationships between interfering modes from multiple primary and reflective sources. This dictionary

can only be used for the geometric structure that it was trained on, but is no longer constrained by unrealistic assumptions that make damage detection difficult in real world circumstances. This learned dictionary is capable of modeling complex wavefields with full wavefield recovery correlations of 98% as opposed to the 20% achieved using an analytical dictionary. It has also been shown that a learned dictionary trained on one structure can distinguish between undamaged and damaged states on a structure that is identical in geometry, but variable in material and thickness. This detection has been accomplished with 100% accuracy using the learned dictionary, as opposed to 50-60% using an analytical dictionary. These damage detection results are also compared to a second empirical model approach that uses a support vector machine, which achieved at most 83% accuracy when trained on one structure and used to predict damage on another.

Chapter 3 is outlined as follows. Section 2 provides an overview of the K-SVD dictionary learning algorithm and the equations used to construct the analytical dictionary used for SWA. The support vector machine framework is also reviewed briefly. Section 3 describes the data set used to for dictionary learning and damage detection, then details the process for detecting damage using dictionaries and support vector machines. Finally, Section 4 contains the results of damage detection using learned dictionaries, analytical dictionaries, and support vector machines. Damage is detected on structures other than those that the models were trained with to evaluate the generalizability of the models.

3.2 Model Construction and Damage Detection

This section discusses the details of constructing a dictionary using dictionary learning, and a method that uses these dictionaries to detect structural damaged. This section concludes with a description of parameter choices for bot learned dictionaries and analytical dictionaries, which will be described in the next section.

3.2.1 Dictionary Learning

Dictionary learning is essentially a method of matrix decomposition that enforces sparse representations of the training signals. Figure 3.1 illustrates a detailed view of this decomposition. A dictionary consists of columns (called dictionary atoms) of signal bases. The number of dictionary atoms is a set parameter and is usually much smaller than

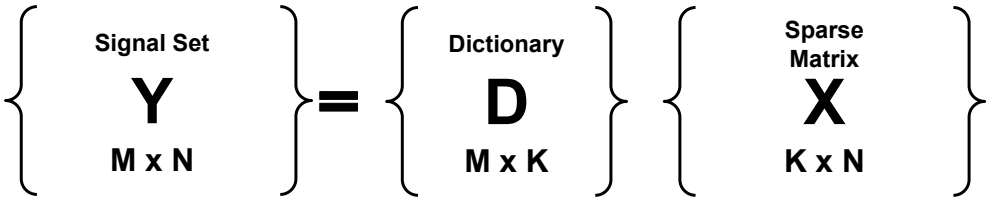

Figure 3.1. Description of the matrices involved in a dictionary decomposition.

the number of signals used to train the dictionary. Limiting the number of atoms in the dictionary forces it to include only the most general bases that can be found in the highest number of training signals.

Linear combinations of the atoms can approximately reconstruct the signals in the training set. The coefficients of the linear combination are encoded in a sparse matrix during the dictionary learning process. The dictionary $\mathbf{D}$ is multiplied by this sparse matrix $\mathbf{X}$ to recover the training set $\mathbf{Y}$. The sparse matrix is said to be sparse because the number of atoms that can be used to reconstruct a signal, and therefore the number of nonzero entries in each column of the matrix, are limited to a number much smaller than the number of atoms in the dictionary. This sparsity is also a set parameter.

Dictionary learning requires two inputs: an initial dictionary and a training signal set represented in a sparse domain. The sparsity of a signal set is a measure of how much redundancy can be found in the set and refers to the smallest number of common bases the set can be broken into. If few bases are needed to recover the data set with high accuracy, then the set is sparse in that particular domain. To train our dictionary, we input a random dictionary of complex numbers and wavefield snapshots in time that have been transformed into the Fourier domain which serve as our training signals. Dictionary learning is effective with these snapshots because spatial wavefield data has been shown to be sparse in the Fourier domain [7].

The dictionary learning objective function is

$$\min_{\mathbf{X},\mathbf{D}} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 \text{ s.t. } \|\mathbf{x}_i\|_0 \leq T \; \forall i, \tag{3.1}$$

where $\mathbf{Y}$ is the signal training set, $\mathbf{D}$ is the dictionary, and $\mathbf{X}$ is the sparse coefficient matrix. To enforce sparsity of the coefficient matrix $\mathbf{X}$, each column is limited to $\mathbf{T}$ nonzero entries. The optimization is approached iteratively and is done in two steps: a sparse coding step

and a dictionary update step. The sparse coding step fixes the dictionary $\mathbf{D}$ and updates the sparse matrix $\mathbf{X}$. The dictionary update step fixes $\mathbf{X}$ and updates one column in $\mathbf{D}$.

To train our dictionary, we use the K-SVD algorithm developed by Aharon et al. [15]. This algorithm updates the dictionary one atom at a time between sparse coding steps instead of all at once. This method provides a superior solution in a fewer number of iterations. The following describes the sparse coding step and dictionary update step used in the K-SVD algorithm in further detail.

3.2.1.1 Sparse Coding

During the sparse coding step of K-SVD, $\mathbf{D}$ is held constant and the sparse matrix $\mathbf{X}$ is updated. The goal is a minimize $\|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2$ subject to $\mathbf{X}$ being sparse. This objective function can be rewritten as

$$\|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 = \sum_{i=1}^{N} \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2. \tag{3.2}$$

The overall objective function (with the sparsity constraint) can be written as N separate problems of the form

$$\min_{\mathbf{X}} \|\mathbf{y}_i - \mathbf{D}\mathbf{x}_i\|_2^2 \text{ s.t. } \|\mathbf{x}_i\|_0 \leq T \ \forall i. \tag{3.3}$$

where N is the number signals input into the algorithm and the number of columns in $\mathbf{X}$. This problem can be solved by using any pursuit algorithm. We use orthogonal matching pursuit (OMP) for reasons of speed and efficiency [95, 98, 99].

3.2.1.2 Dictionary Update

In the dictionary update step, we fix $\mathbf{X}$, and instead of updating the entire dictionary $\mathbf{D}$, we only update one column at a time. The kth dictionary column will be indicated by $\mathbf{d}_k$, and the corresponding kth row in $\mathbf{X}$ by $\mathbf{x}_k^T$. The objective function can again be rewritten as

$$\begin{aligned} \|\mathbf{Y} - \mathbf{D}\mathbf{X}\|_F^2 &= \left\| \mathbf{Y} - \sum_{j=1}^{K} \mathbf{d}_j \mathbf{x}_j^T \right\|_F^2 \\ &= \left\| \left(\mathbf{Y} - \sum_{j \neq k} \mathbf{d}_j \mathbf{x}_j^T \right) - \mathbf{d}_k \mathbf{x}_k^T \right\|_2^2 \\ &= \left\| \mathbf{E}_k - \mathbf{d}_k \mathbf{x}_k^T \right\|_F^2. \end{aligned} \tag{3.4}$$

$\mathbf{E}_k$ contains the portions of the objective function that are not changing during this update. It is also important that only the portions of $\mathbf{E}_k$ that are directly related to $\mathbf{d}_k$ are used for

the optimization. To satisfy this, we employ a matrix $\mathbf{\Omega}_k$ such that $\mathbf{x}_k^T \mathbf{\Omega}_k = \mathbf{x}_k^R$, where the row vector $\mathbf{x}_k^R$ contains only the non-zero values in $\mathbf{x}_k^T$. For example, if $\mathbf{x}_k^T = [2, 0, 0, 6, 0]$, then $\mathbf{x}_k^R = [2, 6]$. Likewise, $\mathbf{E}_k \mathbf{\Omega}_k = \mathbf{E}_k^R$, where $\mathbf{E}_k^R$ is an error matrix that contains only the signals that are currently using the $\mathbf{d}_k$ atom. Equation 3.4 is multiplied by $\mathbf{\Omega}_k$ to create the new objective function:

$$\left\| \mathbf{E}_k \mathbf{\Omega}_k - \mathbf{d}_k \mathbf{x}_T^k \mathbf{\Omega}_k \right\|_2^2 = \left\| \mathbf{E}_k^R - \mathbf{d}_k \mathbf{x}_R^k \right\|_2^2. \tag{3.5}$$

The minimization can now be accomplished using singular value decomposition (SVD) [100], which results in $\mathbf{E}_k^R = \mathbf{U} \Delta \mathbf{V}^T$. The updated dictionary atom $\tilde{\mathbf{d}}_\mathbf{k}$ is the first column of $\mathbf{U}$.

The sparse coding step and dictionary update step will continue to iterate until a stopping criterion is reached. We use a set number of iterations for our training. The result is a dictionary that, if all the parameters have been chosen correctly, models the geometric signal structures found in the training set and a sparse matrix that can be used to recover the training set with high accuracy. For recovering other sets of signals with the same geometric structure as the training set, we apply a single sparse coding step to a small subset of the new signals and the trained dictionary.

3.2.2 Dictionary Recovery Correlation

After a dictionary is created, it can be used to detect damage by calculating a dictionary recovery correlation. Figure 3.2 illustrates the process of calculating the recovery correlation. Given a dictionary and a partial signal set, the recovery correlation can be calculated in three steps: sparse coding of the partial signal set $\tilde{\mathbf{Y}}$ given the dictionary $\mathbf{D}$ to calculate the sparse matrix $\mathbf{X}$; multiplication of the sparse matrix $\mathbf{X}$ and the full dictionary $\mathbf{D}$ to recover the full signal set $\widehat{\mathbf{Y}}$; and finally, correlation between the input signal set $\mathbf{Y}$ and the recovery $\widehat{\mathbf{Y}}$ to get the recovery correlation C. Each of these three steps will now be addressed in more detail.

Only a small spatial subset of the wavefield is actually needed to calculate the recovery correlation - about 400 of the full 10,000 points. This 400 point scan can be performed in under 4 minutes. The reason for the choice of 400 samples is discussed in the following subsection. Since the full dictionary $\mathbf{D}$ is trained using a full wavefield $\mathbf{Y}$, only the rows in the dictionary that correspond to the exact spatial points in the partial sample set are used

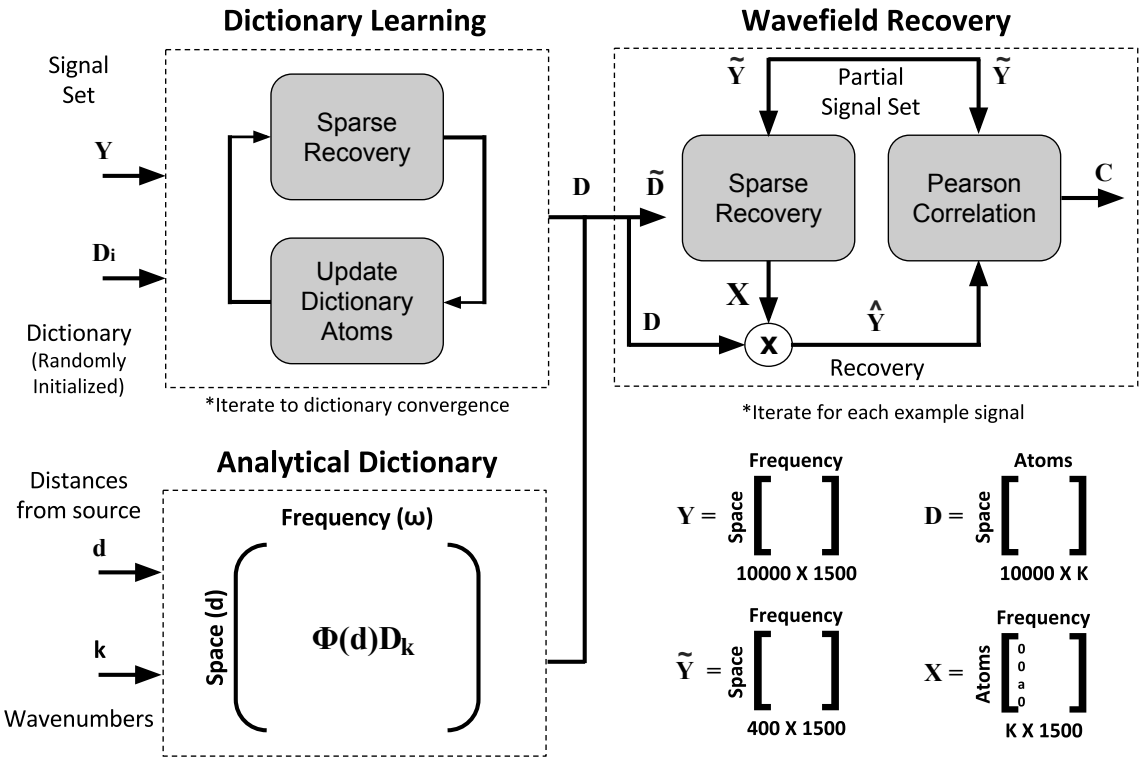

Figure 3.2. The entire dictionary damage detection process from dictionary construction, to dictionary recovery correlation.

when performing sparse coding. A sparse coding technique called orthogonal matching pursuit [95] takes the partial dictionary $\tilde{\mathbf{D}}$ with and the partial signal set $\tilde{\mathbf{Y}}$ and calculates the sparse matrix $\mathbf{X}$.

The sparse matrix $\mathbf{X}$ can then be multiplied by the full dictionary $\mathbf{D}$ to recover an estimate of the full data set $\widehat{\mathbf{Y}}$. The resulting $\widehat{\mathbf{Y}}$ is a full wavefield recovered from a partial sampling. If there is damage present in the measured structure, the recovery $\widehat{\mathbf{Y}}$ will not reconstruct the damage because it was recovered from a dictionary representing a pristine structure. The partial signal set $\tilde{\mathbf{Y}}$ correlated with the corresponding rows in the recovered signal set $\widehat{\mathbf{Y}}$ gives the recovery correlation C. A high correlation indicates a healthy, undamaged structure. A correlation below a certain threshold indicates damage.

3.2.3 Dictionary Hyperparameters

There are a number of hyper-parameters that can be adjusted during both the dictionary learning and analytical dictionary construction process. Each has been given a

variable name for easier referencing. These hyper-parameters are shown in Table 3.1. Dictionary learning uses all of these parameters. Constructing an analytical dictionary uses all of these parameters except the number of iterations i, but the number of atoms in the dictionary (a) and the number of sparse elements in the sparse matrix (v) are very different. Both methods share the same number of spatial samples s and time samples t used in recovery.

For dictionary learning, a and v do not actually effect the results of damage detect beyond small changes in the margin of classification. Larger dictionaries (larger a) with more sparse elements (larger v) tend to have slightly larger classification margins, but the difference is not significant. The number of iterations (i) has little effect as long as the training root mean square error has begun to converge to 2 decimal places. The time sample size (t) was chosen to include a vast majority of the reflections in the plate, and ends when the signal starts to diminish and be less meaningful. By $t = 800$ the A0 mode has reflected about 40 times.

For analytical dictionaries, a and v also did not have a very large effect on the results. The values chosen reflect best practice. The wavenumbers chosen to populate the atoms (a) in the dictionaries where spaced 0.6 m^{-1} apart in a range of 1 to 630 m^{-1}. The number of sparse elements (v) in the matrix where chosen according to the literature [7], and to be small to encourage sparsity. Increasing the number of sparse elements (v) increased the recovery correlations, but does not increase damage detection accuracy, which seemed to fluctuate based only on chance.

The hyper-parameter that has the greatest effect in damage classification accuracy is the

Table 3.1. The dictionary construction hyper-parameters are each assignment a variable for easier referencing, and the values chosen for each are listed. "LD" stands for learned dictionary, and "AD" for analytical dictionary.

Dictionary Hyper-parameters

Variable	LD Value	AD Values	Description
a	100	1049	Number of atoms in the dictionary
v	1	3	Number of sparse elements in sparse matrix
i	50	NA	Number of training iterations
s	400	400	Spatial sample size used in recovery
t	800	800	Time sample size used in recovery correlation

number of spatial samples s that are used in recovery. If a sample size that is too small is used, the classification margins decrease for plates with the lower signal to noise ratio, such as for thicker plates. For all the plate, A smaller sample size s gives worse classification, and larger s gives better classification. The sample size $s = 400$ is chosen to ensure correct classification for a learned dictionary with as small a sample size as possible. A smaller sample size ensures that measurements can be taken as quickly as possible. Note that for analytical dictionaries, the number of spatial samples s used only affected the variability of the recovery correlations, not the classification accuracy.

3.3 Comparative Methods

Two alternative damage detection methods will be used in comparison to the learned dictionary results. These include the use of an analytical dictionary, which will be used to detect damage using the same method as the learned dictionary, and a support vector machine method. The following is a description of both of these methods.

3.3.1 Analytical Dictionary

An analytical dictionary is compared to the learned empirical dictionary. The analytical dictionary that we use comes from a method for dispersion curve recovery called sparse wavenumber analysis (SWA) [7]. The full process of recovering wavefield data with the analytical dictionary can be split into two steps: a dictionary construction step and a sparse coding step. SWA uses wave propagation theory to derive a dictionary, and then uses a pursuit algorithm such as basis pursuit [7] or orthogonal matching pursuit [92] to calculate the dispersion curves, represented by a sparse matrix, found in a measured wavefield. Note that the construction of the dictionary is completely separate from the second sparse coding step.

It is also important to note that while the sparse matrix calculated from a well-structured analytical dictionary defines the dispersion curves of the wavefield, the sparse matrix of our learned dictionaries does not. The following overviews the steps used for the construction of the analytical dictionary.

The wave propagation model used in the derivation is

$$Y(d,\omega) = \sum_m \frac{1}{\sqrt{k_m(\omega)d}} S(\omega) G_m(\omega) e^{-jk_m(\omega)d}, \tag{3.6}$$

which represents a Lamb wave signal between any two transducers on a plate, given a particular frequency ω and a distance d from the source. The received wave $Y(d,\omega)$ is a sum of each of the guided wave modes $m = 1, 2, \ldots$. Each mode has a frequency-dependant amplitude $G_m(\omega)$ and frequency-dependant dispersion relation $k_m(\omega)$ such that the mode velocity varies as a function of frequency. The signal $S(\omega)$ is the original source excitation.

A set of M different distance measurements at one frequency ω_n is represented as an $M \times 1$ vector

$$\mathbf{y}_n = \mathbf{\Phi}(\mathbf{d})\mathbf{D_k}\mathbf{x}_n = [Y(d_1,\omega_n),\ \ldots\ ,\ Y(d_M,\omega_n)]^T. \tag{3.7}$$

The vector $\mathbf{x}_n$ is an $K \times 1$ matrix that represents the amplitudes of K discretized wave numbers $k_1,\ k_2,\ \ldots\ ,\ k_K$ for a single frequency ω_n. $\mathbf{d} = [d_1,\ \ldots\ ,\ d_M]$ describes the travel distances corresponding to $\mathbf{y}_n$. The relationship between the sparse vector $\mathbf{x}_n$ and the signal $\mathbf{y}_n$ is the analytical dictionary that we are interested in. The analytical dictionary $\mathbf{D} = \mathbf{\Phi}(\mathbf{d})\mathbf{D_k}$, where

$$\mathbf{D_k} = \text{diag}[k_1^{-\frac{1}{2}}, k_2^{-\frac{1}{2}}\ ,\ \ldots\ ,\ k_N^{-\frac{1}{2}}], \tag{3.8}$$

$$\mathbf{\Phi}(\mathbf{d}) = [d_i^{-\frac{1}{2}} e^{-jk_j d_i}]_{ij}. \tag{3.9}$$

3.3.2 Support Vector Machine (SVM)

The support vector machine (SVM) is a common machine learning technique used for classification. We will use it for binary classification between undamaged and damaged structures, and compare it with our dictionary learning results. The following is brief description of the SVM model used in this chapter. A more detailed description of the SVM algorithm can be found in [50].

For SVM, we define $\mathbf{x}$ to be a high dimension vector of features acquired from the structure. The boolean valued y represents the true damaged state, where $y = 1$ denotes damaged and $y = 0$ denotes undamaged. We assume $y = f(\mathbf{x})$, where f is an unknown function that maps a feature vector to the damage state. The goal of an SVM is to learn a model function such that prediction $\hat{y} = \hat{f}(\mathbf{x})$, where $\hat{f}$ is a learned approximation of f represented by a linear weighted vector $\mathbf{w}$. This vector $\mathbf{w}$ can be described as a high-dimensional plane that will separate the feature vector points $\mathbf{x}$ that are labeled damaged from those that are labeled undamaged.

During the training phase, a set of feature vectors $\mathbf{x}_i$, where $1 \leq i \leq M$, are input into the algorithm. Each $\mathbf{x}_i$ is coupled with a true label y_i. The weighted vector $\mathbf{w}$ is randomly initialized with the same number of elements as a feature vector $\mathbf{x}_i$. A prediction is made by computing $\mathbf{w}_i^T\mathbf{x}_i$ where $\hat{y}_i = 1$ if $\mathbf{w}_i^T\mathbf{x}_i \geq 0$ and $\hat{y}_i = 0$ if $\mathbf{w}_i^T\mathbf{x}_i \leq 0$. The method by which the vector $\mathbf{w}$ will update is based on whether or not that prediction $\hat{y}_i$ is correct. The update step of the SVM is the heart of the SVM algorithm, and will be explored in more detail.

3.3.2.1 Update Rule

The objective function for a soft SVM

$$\mathbf{J}^t = \min_{\mathbf{w}} \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum \max(0, 1 - y_i\mathbf{w}^T\mathbf{x}_i), \tag{3.10}$$

where the term $C\sum \max(0, 1 - y_i\mathbf{w}^T\mathbf{x}_i)$ is the loss function, in this case a hinge loss, and $\frac{1}{2}\mathbf{w}^T\mathbf{w}$ is the regularizer. The minimization of the loss function biases the equation towards making correct predictions. The minimization of the regularizer biases the solution towards smaller classifiers, which has the effect of producing the largest margin separator between the two classifications. C is a generalization parameter that can be adjust to ensure that the learned vector $\mathbf{w}$ (model) generalizes to feature vectors not seen during training.

Because the hinge loss function $C\sum \max(0, 1 - y_i\mathbf{w}^T\mathbf{x}_i)$ is not continuous, and therefore not differentiable, Equation 3.10 is solved using a stochastic subgradient descent. The stochastic update of the vector $\mathbf{w}$ is $\mathbf{w}_{i+1} = \mathbf{w}_i - r\nabla\mathbf{J}^t$, where r is a learning rate, and

$$\nabla\mathbf{J}^t = \begin{cases} \text{if } \hat{y}_i = y_i & \mathbf{w} \\ \text{if } \hat{y}_i \neq \hat{y}_i & \mathbf{w} - Cy_i\mathbf{x}_i. \end{cases} \tag{3.11}$$

Then the updated $\mathbf{w}_{i+1}$, which is calculated after every training example $\mathbf{x}_i$, is

$$\mathbf{w}_{i+1} = (1 - r)\mathbf{w}_i + \begin{cases} \text{if } \hat{y}_i = y_i & 0 \\ \text{if } \hat{y}_i \neq \hat{y}_i & rCy_i\mathbf{x}_i. \end{cases} \tag{3.12}$$

After training, the vector $\mathbf{w}$ is used to predict the damage state $\hat{y}$ of similar structures given a feature vector $\mathbf{x}$ from that structure.

To choose the hyper-parameters of learning rate r, generalization parameter C, and the number of times to iterate through the example set, a cross-validation was performed. This cross validation was performed by splitting the training set into five parts and validating for highest prediction accuracy.

3.3.2.2 Feature Extraction

The feature vectors $\mathbf{x}_i$ mentioned above are created using single point signals from the Lamb wavefields described in the experimental setup section of this chapter. To create the feature vector, the time domain signal was transformed into both the Fourier domain and the Mellin domain [101]. The following eight features were extracted from these two domains and the values were placed in a 16×1 feature vector.

Features 1 through 3 are the amplitude of the first three peaks. Features 4 and 5 are the mean and variance of the of peak amplitudes greater than 20% of the maximum peak amplitude. Feature 6 is the variance of the of peak amplitudes greater than 60% of the maximum peak amplitude. Feature 7 is kurtosis, which is a measure of the tailedness of the probability distribution of a real-valued random variable. Kurtosis is a descriptor of the shape of a probability distribution, and is calculated using

$$K[\mathbf{X}] = \frac{\mu_4}{\sigma^4} = \frac{E[(\mathbf{X} - \mu)^4]}{(E[(\mathbf{X} - \mu)^2])^2}. \tag{3.13}$$

Feature 8 is the curve length, calculated using

$$L = \sum_{i=2}^{N} |x(i) - x(i-1)|. \tag{3.14}$$

These features were selected to correspond with features used in the literature for damaged detection using ultrasonic guided waves, namely [101].

3.3.2.3 SVM Damage Detection

An SVM model is a high dimensional vector $\mathbf{w}$ whose dot product with an input signal will predict whether the signal came from an undamaged or damaged plate. This model vector must first be trained using labeled data (undamaged or damaged labels). Each spatial signal $\mathbf{y}_i$, where $1 \leq i \leq M$, from the wavefields is used for training. These training vectors are denoted as $\mathbf{x}_i$ in the SVM notation above. To train, the model vector predicts the damage state of the input signal using a dot product, and updates based on the outcome of the prediction. After a certain number of iterations, the model accuracy stops increasing and the SVM training terminates. More details of the training process can be found in the the SVM subsection above.

The benefit of using an SVM to detect damage is that only one signal from the wavefield is required to detect damage, instead of 400. However, the damaged data required to train

the model is not often available, and the accuracy of this damage detection technique, using Lamb waves, is very low. This is demonstrated later in this chapter.

3.4 Experimental Setup

The data acquisition system described in Chapter 2 was used to acquire wavefield data. This full wavefield data was acquired from four metal plates: a 1.6 mm aluminium plate, a second identical 1.6 mm aluminium plate, a 6.35 mm aluminium plate, and a 1.6 mm steel plate. These plates are named "A," "B," "C," and "D," respectively, for easier referencing later in the chapter. An undamaged and damaged scan were done for each of these four plates, resulting in eight wavefields. Each plate was reset in between undamaged and damaged scans for each. The damaged state of each plate was simulated using a scatterer. Our scatterer was a small steel washer attached to the bottom of the plate using coupling fluid, which had a wave scattering effect on the structure. The outer diameter of the washer was 10 mm, the inner diameter 4 mm, and the thickness 1 mm. The total weight of the washer was about 2 grams. Each wavefield scan is 100 points by 100 points, or 10,000 spatial points total. These scans each cover a 99 mm by 99 mm area in the center of 108 mm by 108 mm metal plates. An excitation sensor is adhered to the bottom of each plate, offset from the center by about 20 mm, and excited using a $\pm$10 V swept sinusoid of 1kHz to 150 kHz. The size of each wavefield matrix is 10,000 points in space by 3001 points in time. The signals were sampled at 300kHz. A detailed description of the raw data is found in Table 3.2.

Table 3.2. Raw data used for damage detection and dictionary learning. Four plates of different materials and thickness where scanned twice, producing an undamaged and damaged scan for each plate. The four plates are named "A," "B," "C," and "D".

Raw Data: Wavefields

Material	Index	Plate	Thickness	Damage?
Aluminium	1	A	1.6 mm	No
Aluminium	2	A	1.6 mm	Yes
Aluminium	3	B	1.6 mm	No
Aluminium	4	B	1.6 mm	Yes
Aluminium	5	C	6.35 mm	No
Aluminium	6	C	6.35 mm	Yes
Steel	7	D	1.6 mm	No
Steel	8	D	1.6 mm	Yes

3.5 Damage Detection Results

The following section details the damage detection results using the data set and methods described in Section 3. Results for the learned dictionaries are presented first, the analytical dictionary second, and the support vector machines last.

3.5.1 Learned Dictionary

Figure 3.3 contains the damage detection results for learned dictionaries. Each row of the figure contains results using a dictionary trained by the undamaged scan of one of the four plates and tested on the other three. Each dot in the plots represents a different random sampling of 400 spatial points taken from either the undamaged or damage version of the plate. We used 26 such samplings of 400 for each damage classification. Of the 26 samplings, 13 are from the undamaged plate and 13 are from the damaged plate.

The results show that undamaged and damaged samplings are separable with 100% accuracy for every plate with any of the four dictionaries. Note that 100% accuracy seems to be achievable for any dictionary trained using any number of atoms higher than 50 and with any number of sparse elements. Dictionary sizes of 50 to 300 atoms were attempted with 1 to 15 sparse elements, each resulting in 100% accuracy. The critical parameter in this result is the use of 400 samples per plate sampling. At 300 samples, the undamaged and damaged samplings for plate C begin to blend together.

Samplings from plate C are especially difficult to classify because the magnet used to simulate damage on the plate was the same weight as the one used on the thinner plates. Since plate C has 4 times more mass than the other plates, the simulated damage was relatively small. The larger the sample size, the clearer the results and the more sensitive the method is to small amounts of damage. Therefore a higher sample size is required to classify samplings from the thicker plate. If a larger mass was used for the plate C damage simulation, it is possible that all the plates would classify as 100% using a smaller and quicker to acquire sample size.

3.5.2 Analytical Dictionary

Figure 3.4 contains damage detection results using the analytical dictionary designed for SWA [7]. Only one dictionary is used, and it is tested on all four plates. The dictionary

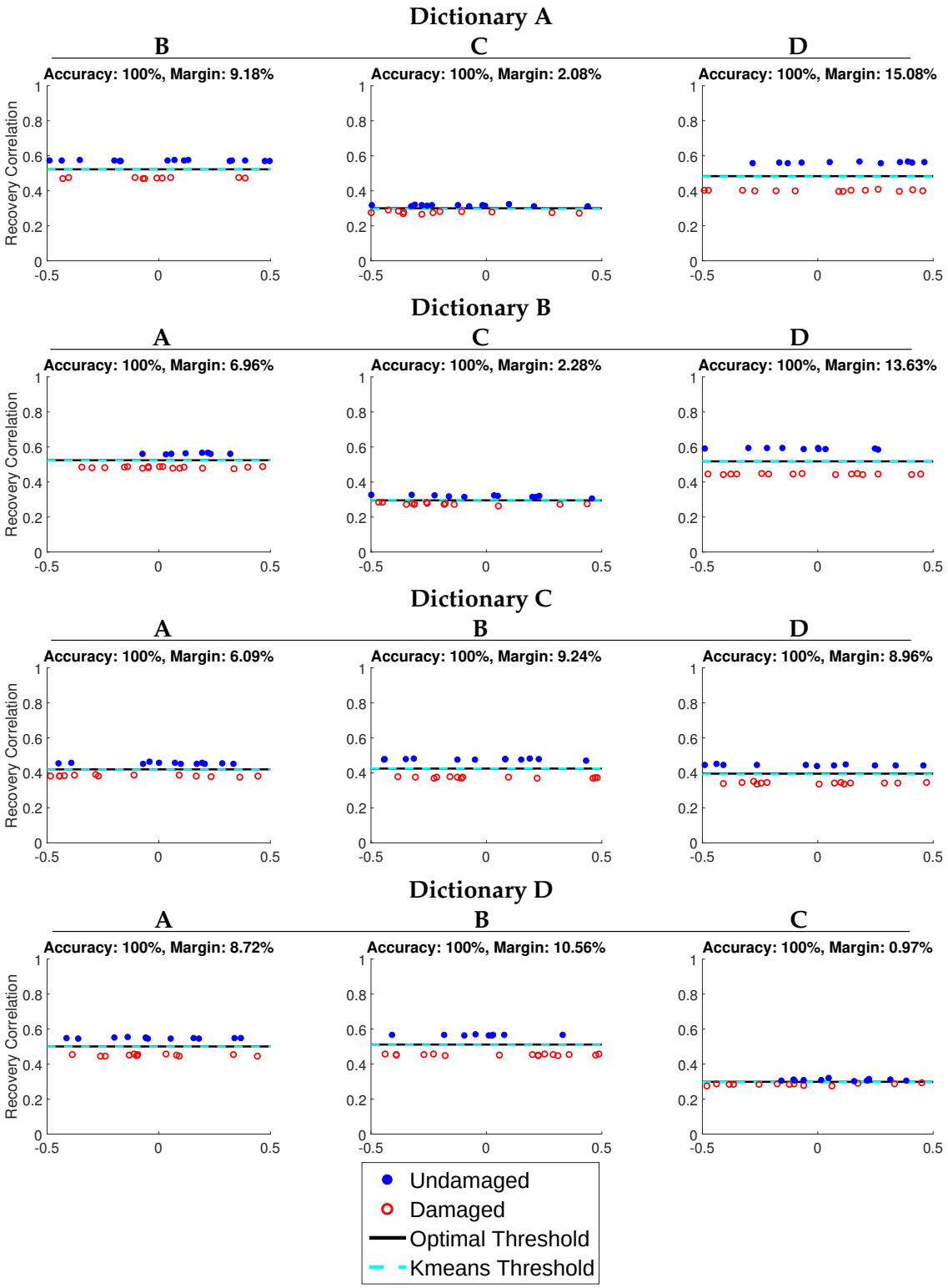

Figure 3.3. Results for damage detection using a learned dictionary. Each row shows results for a dictionary trained using one of the plates, then tested on the other three plates. The x axis is a random number, used for recovery correlation clarity.

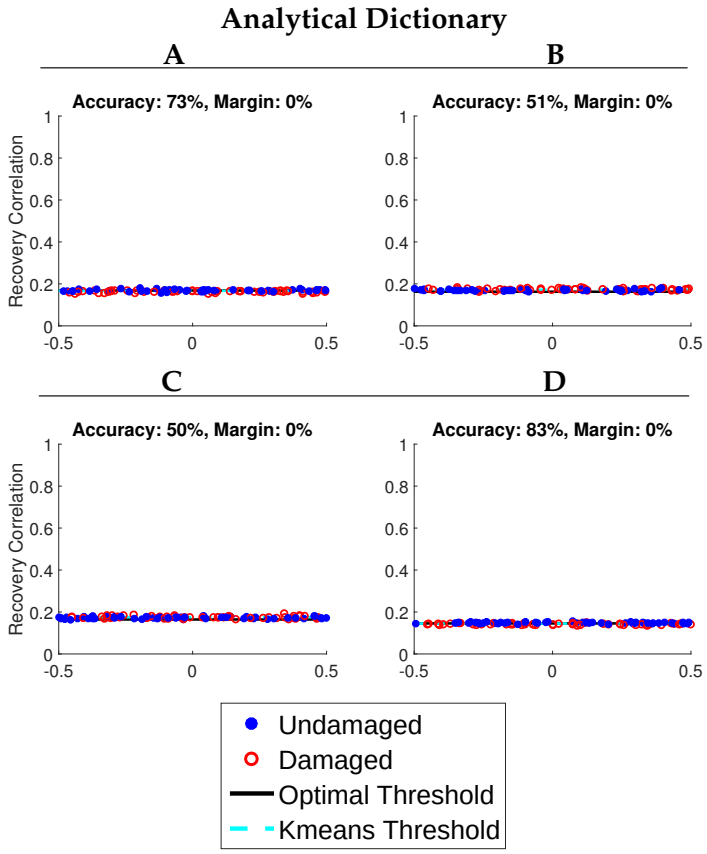

Figure 3.4. Results for damage detection using an analytical dictionary. The dictionary is tested on all four plates. The x axis is a random number, used for recovery correlation clarity.

parameters are chosen to reflect the best results. The data and spatial and time sample sizes used are the same as those used for the learned dictionary damage detection. This is done to provide a fair comparison between the two.

The results show that most of the recovery correlations sit around 15-20% with little to no separation between the undamaged and damaged samplings. Classification using the analytical dictionary averages around 64%. It is important to note that the results are are very inconsistent between plates and between trials. The range of recovery correlations is so small that one attempt will classify the points very well, and the next, with 26 different randomly selected points, will not. To eliminate as much of this uncertainty as possible, the results shown reflect 100 different samplings instead of 26, but the results still demonstrate a large accuracy variance.

3.5.3 Support Vector Machine

The results for damage detection using an SVM can be found in Table 3.3 and Table 3.4. Table 3.3 contains results when using all of the features, and Table 3.4 contains results for just the Fourier domain features. All of the spatial signals from one wavefield, both undamaged and damaged, are used to train four different predictors. These were each tested on the remaining three plates for damaged detection accuracy. Overall, SVM achieves an average classification accuracy of 52%. As expected, the best results are achieved when training with a 1.6 mm aluminum plate and testing with a 1.6 mm aluminum plate. Using only the Fourier domain features gave slightly better results at an average of 64%, instead of the 52% achieved using all of the features.

3.6 Conclusion

In this chapter we have demonstrated the effectiveness of a learned dictionary in modeling wavefields that contain complex multipath interference. The learned dictionary was able to recover full wavefields from as little 4% of the original wavefield with up to 98% accuracy, as opposed to the 20% recovery correlations achieved using analytical dictionaries.

Table 3.3. Damage detection accuracy using support vector machines: includes all features.

Damage Detection Accuracy: SVM

Trained by →	A	B	C	D
A	–	51%	52%	50%
B	50%	–	51%	54%
C	52%	53%	–	56%
D	53%	44%	54%	–

Table 3.4. Damage detection accuracy using support vector machines: includes only Fourier features.

Damage Detection Accuracy: SVM

Trained by →	A	B	C	D
A	–	61%	50%	51%
B	83%	–	77%	50%
C	79%	79%	–	53%
D	72%	68%	63%	–

Methods for structural damage detection using a learned dictionary model, an analytical dictionary model, and a support vector machine were presented. Using the dictionary damage detection method, we demonstrated the ability of a learned dictionary to detect damage on a structure with both velocity and amplitude variations in the wavefield due to changes in material and thickness. We achieved a 100% accuracy using learned dictionaries, as opposed to 57% and 64% accuracies achieved using an analytical dictionary and support vector machines.

CHAPTER 4

CONCLUSION

4.1 Summary

In this thesis, we have accomplished three major tasks: We built and validated a system for the purpose of acquiring full wavefield data of guided Lamb waves; we showed that dictionary learning can be used to recover wavefield data with multipath interference with 98% accuracy; and finally, we demonstrated that learned dictionaries can generalize to different materials and thicknesses for use in damage detection. This damage detection was performed with 100% accuracy for each wavefield, regardless of the source of the learned dictionary.

These results have significantly improved our ability to detect damage with full wavefield data containing large amounts of multipath interference. This is an important improvement because most real world structures have very complex wave propagation patterns due to reflections from boundaries and fasteners. Damage detection for more and more complex structures could be facilitated by learned dictionaries to enable more complex structures to benefit from non-destructive evaluation and structural health monitoring.

The ability of learned dictionaries to generalize makes it possible for one dictionary to detect damage in everything with a similar geometry. This would eliminate the need for multiple dictionary models, unless the geometry changes, which simplifies the damage detection process and reduces the amount of baseline data necessary.

4.2 Future Work

Dictionary propagation models have huge potential for future development. The following section lists possible future work to further develop these capabilities and adapt the data acquisition system to accompany these developments.

4.2.1 Damage Detection and Dictionary Learning

Currently we are able to detect damage with a single dictionary on variable structures, addressed separately. It would be better to be able to be able to address all structures with the same threshold. This would eliminate the need to calibrate a new threshold for each variation of the geometry for which the dictionary was learned. To accomplish this, the dictionary would need to generalize better. Two thoughts on how to achieve this include training a dictionary using input from multiple wavefields, or training a dictionary on only the first 10 or 20 reflections instead of the full 60 or 70.

In this thesis we have tested for dictionary model generalization for changes in material and thickness, but other variations could be explored. These variations could include variations in temperatures, large spatial shifts in the measured data, and variations in applied stress.

Learned dictionaries can be applied to visualize and locate damage. Sparse wavenumber analysis has been used to recovery undamaged versions of structures, then that recovery is subtracted from the original wavefield to better visualize portions of the wavefield that indicate damage. Further sparse array processing can be done after this portion of the signal is isolated, which will locate damage with a high resolution. This same approach can be used with learned dictionary to produce the same results but for wavefields with multipath interference.

Learned dictionaries have been shown to model and recover wavefield data with multimodal interference with high accuracy. In this thesis, a small metal plate was used to show this, but more work can be done the test how well this would work in increasingly complex wavefields. Learned dictionary models could be explored for materials with fasteners (rivets, bolts, etc.) or anisotropic material properties and could be applied to three-dimensional objects.

4.2.2 DAQ System

There are a few simple alterations that could be made to the platform to improve its performance. The scan area of the DAQ system could be increased by purchasing or building a larger stage. A new DAQ card could be purchased to increase the sampling rate, but a new laser doppler vibrometer would be needed to measure at higher frequencies.

Encoder feedback control and limit switches could easily be attached to ensure that the system performs consistently.

Another addition to the system that would improve usability and reduce operational variability would be the construction of a sensor embedded scanning guide. The scanning guide would be attached to the stage and metal plates would be clamped to the top of the guide. A guide insert could be used on top of the scanning guide to adapt to different geometries. The excitation signal would come from the embedded sensor which would be coupled with the metal plate using coupling fluid, not permanently glued. Such a scanning guide would allow for quick repeatable scans with reduced operational variability and reduced set up. In the right circumstances more sensors can be embedded in the scanning guide and used to measure partial wavefields for damage detection. This could possibly be done simultaneously to speed up data acquisition of the partial wavefield.

The scanning capabilities of the platform can be expanded to three-dimensional object wavefields. This would require a motion system that is capable of additional degrees of freedom to be able to scan objects from different angles. A sturdy design would be necessary as to limit excess movement in the system while the vibrometer is taking data and allow precise and repeatable measurements. Setting up the laser to be able to refocus for different scanning distances would simplify the motion system. Computer vision algorithms might be necessary to be able to model the surfaces of the object to allow the system to know how far away the surface of the object is, and what direction is perpendicular to the surface. Motion planning algorithms may also be needed to create a path that will scan each important point along the surface of the object and feedback control systems may be needed to ensure that the path is followed.

Improvements on the LabVIEW software used to control the system include error and exception capturing, stop capabilities in the middle of a movements and/or wavefield measurements, simpler user interface for saving the data in the correct location, a random sampling function that would take a quick partial wavefields given sample size over a given area, and a plot for RMS visualization of a wavefield during scanning.

APPENDIX

DAQ SYSTEM USERS GUIDE

This appendix provides additional information on the construction and operation of the DAQ system described in Chapter 2 of this thesis. Additional information on select hardware used to build the system will be discussed in Section A.1, including some suggestions for alternative hardware. Section A.2 gives a brief introduction to LabVIEW programming, avenues of pursuing help with LabVIEW, and a list of packages to include for LabVIEW installation when working with the DAQ system program described in Chapter 2. Section A.3 provides a setup procedure for the system, and Section A.4 gives solutions to common problems with the system.

A.1 Adapting the System

Some of the choices made while developing this system were largely arbitrary, or based on what hardware was readily available at the time. A similar performing system could have been developed using different hardware. The following sections provide a more details overview of some of the system componenets that could be developed different depending on what hardware is available to the reader.

Nation Instruments hardware will be addressed first, then laser doppler vibrometer hardware, and finally piezoelectric actuators.

A.1.1 Nation Instruments Hardware

There are four components necessary for the implementation of a National Instruments (NI) system used for the current work: PXI chassis, application cards that are inserted into the chassis, interface boards that allow input/output access to the application cards, and a connection from the PXI chassis to a computer that runs LabVIEW software. The connection between the chassis and computer usually includes an interfacing application card for the chassis and some kind of express card for the computer. Laptop express cards

are often inserted into an MXI card slot, though these are becoming increasingly hard to find on laptops, while desktop computer express cards can be inserted into the PCI or PCI-e slot of a motherboard. All of this will, of course, require the use of correct connector cables which will either be provided with the system, or described in the data sheet for the particular component.

Many other types of systems are produced by NI that require slightly different setups. There exist different chassis other than PXI, with different sized cards slots. There also exist application cards that can be inserted directly into the PCI or PCI-e slots on the motherboard of a desktop computer. A separate chassis is not needed in this case. CompactDAQ devices also exist, which include removable module boxes for different uses. These often connect through a Universal Serial Bus (USB) but may not provide truly simultaneous data acquisition. CompactRIO devices have embedded field-programmable gate array (FPGA) chips that allow for real-time processing, and a rugged build. Many more devices exist with varying specifications. NI systems are customizable and different systems may have widely varying applications. Third party systems can also be coupled with NI systems for more customization. Multiple chassis can also be synchronized to work in tandem for bigger systems.

Different applications cards have different capabilities, including: truly simultaneous DAQ measurements; different clock and sampling rates; different sensitivity, resolution, or precision; different usability and connectibility with existing systems, and much more. Given the variability of NI systems, the reader is refereed to the NI website and specific component data sheets for further reading and guidance: www.ni.com/. Sales persons and technicians are also available for consulting and are best reaches by submitting a service request through a user account setup on the website, or by calling them directly through customer service or sales.

A.1.2 Laser Doppler Vibrometer (LDV)

Laser doppler vibrometers (LDV) can come in many forms, but generally consist of three components: a controller, which generates the laser and processes the reflected light to measures the frequency shift; a sensor head that focuses the laser and receives the reflected light from the specimen; and a heavy duty fiber optic cable that tethers the two.

Most of the system specifications for the LDV system, such as the frequency range of operation and sensitivity, are determined by the controller chassis. Polytec controller modules can be selected for different types of demodulation, input and output connectors, internal laser units, and include integrator modules and even remote control modules. These are each chosen to match specific measurement needs and inserted into the controller chassis. The sensor head can then be attached to the controller, and the controller to the analog input on your DAQ device. Sensor head specifications typically include standoff distance and spot diameter.

The choice of modules in the controller is often determined by the input laser used. This being the case, the input laser is typically the main system design decision, and the decoder and output modules often come recommended to fit the laser choice. Integrator modules then come recommended for the choice of decoder. A controller module can also be used, which allows for switching to different ranges of sensitivity, and provides a switch to turn filtering on and off.

Polytec seems to sell that largest variety of vibrometers: www.polytec.com/. There are many companies (OMETRON, OMS Corporation, Wavelength Opto-Electronic, Optomet, etc.) that have developed laser vibrometry solutions for specific applications, but their selections are specialized and limited.

While all LDVs operate in a similar fashion, utilizing the doppler effect to determine the velocity of the object that the laser hits, there are a number of different types which offer different advantages. The following are a few from the Polytec website. A single-point vibrometer, as one might expect, measures vibration at a single point in the plane incident to the the laser beam. This means that if the laser were directed at some angle other than 90 degrees from the structure, some combination of the out-of-plane and in-plane vibration velocities would be measured. A 3D vibrometer uses this effect with three different lasers to measure the vibrations of a structure in the X, Y, and Z directions; Z being incident to the laser beam. There also exists scanning vibrometers that use an X-Y mirror to interrogate large area of a structure quickly. The operation is similar to a single-point or 3D vibrometer, but the laser can be redirected quickly using the mirrors for quick, successive, multi-point scans. 3D scanning vibrometers exist, which couple the advantages of 3D measurements and quick area scans, but are very expensive.

These different types of vibrometers can be packaged in mobile devices for field measurements or in stationary bench top equipment. The frequency ranges available span from kilo-Hertz to mega-Hertz, 24 MHz being the highest found on the Polytec website. Prices also range from a few thousand dollars for a low frequency, single-point vibrometer, to tens of thousands or more for a high frequency, 3D or scanning vibrometer.

There are many other designations for vibrometers that offer a slightly different performance such as rotational, differential, multibeam, and self mixing [102] vibrometers. There also exists a continuous scan laser doppler vibrometer which continuously scans a surface to determine its motion at many point simultaneously [103]. These types are left to the reader to further explore.

The current system uses a stage for X-Y movement. This was done because scanning vibrometers are expensive and one was not available for this work. Using a scanning vibrometer could possibly decrease operational variation compared to measurements taken using a stage. However, the stage allows the use of the 3D vibrometer, which has its own particular benefits of being able to measure vibrations in any direction on the structure. Also, because of the high number of averaged measurements needed to decrease noise, the scanning vibrometer would not provide for a much faster scan, just the potential of a simpler setup.

To improve sensitivity of the LDV measurements, reflective tape such as 3M^{TM} Scotchlite^{TM} can be applied to the surface of the specimen to increase reflected light to the LDV.

A.1.3 Piezoelectric Actuators

Each piezoelectric sensor has a frequency spectrum of operation that needs to be matched with the frequency spectrum for the LDV being used to measure the excited vibrations. The most common frequency ranges are between 100 and 400 kHz, but can go as low as 20 kHz, and as high as 600-700 kHz. However, these devices often operate best at one or two peak frequencies, which can make it troublesome to excite a swept sinusoid. There also exist broad band sensors, but they are often between 100-900 kHz, which would require a faster LDV and DAQ device. Some nonhoused piezoelectric wafers can achieve higher frequencies (1-10 MHz), but are much more flimsy and usually need to be attached

permanently to the structure for use. These can be found on www.steminc.com/PZT/.

In place of piezoelectric actuators, an impact can be used to excite vibrations in a structure. This is often done by dropping a hammer or pin on the structure. Impact excitations are consistently and often require large setups. It is also hard to know exactly what kind of signal was excited in the structure. There is also the possibility of damaging the system with an impact, and damage caused by a nondestructive evaluation is undesirable. Piezoelectrics are simply a more versatile, easy to use, and repeatable method of generating ultrasonic frequencies.

A.2 LabVIEW: Brief Introduction

LabVIEW is a visual programming language developed by NI: www.ni.com/labview/. It can be used to program applications using graphical programming blocks (or Virtual Instruments (VI)), strung together using input and output wires that are used to propagate variables between the VIs. LabVIEW is often used for data acquisition, instrument control, and automation in industrial and research environments, and is available for use on various operating systems, including: MAC OS, Windows, and many versions of Unix and Linux.

When developing LabVIEW applications, there are two panels that can be used. The first is the block diagram panel (see Figure A.1), which can be used to arrange all of the VI blocks necessary for the application, and the second is a control panel (see Figure A.2). Each system variable, represented as input and output wires to VIs in the block diagram window, can be represented in the control panel view using controls or indicators. Indicators allow the user to see what the program is doing (the variable states) during execution, and controls allows for user input into the system.

In the block diagram view, many blocks for mathematical operations and transformations are available for use, as well as more complicated blocks that can be used for conditional and iterative programming, timing, data type conversion, data structuring, interfacing with hardware, ect. Many VI blocks are only available if the correct package is downloaded and installed. Packages can be installed either during the main program installation or at some later point in time.

LabVIEW add-on packages are categorized into six categories: Design, Deploy, Inter-

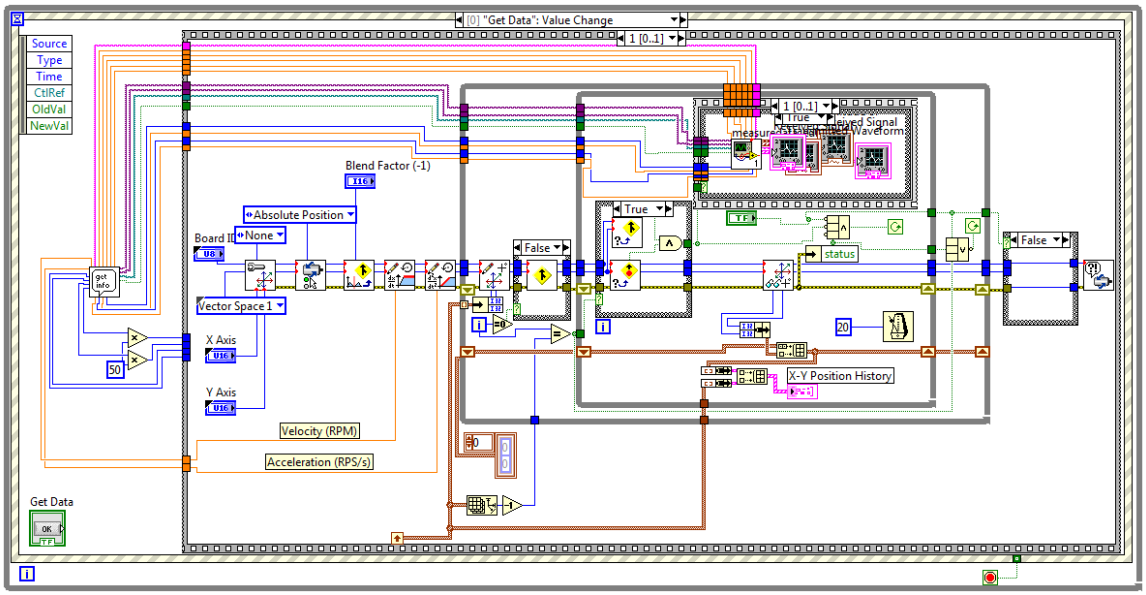

Figure A.1. Box diagram for a portion of the data acquisition code that is launched when the **Get Data** button is pressed.

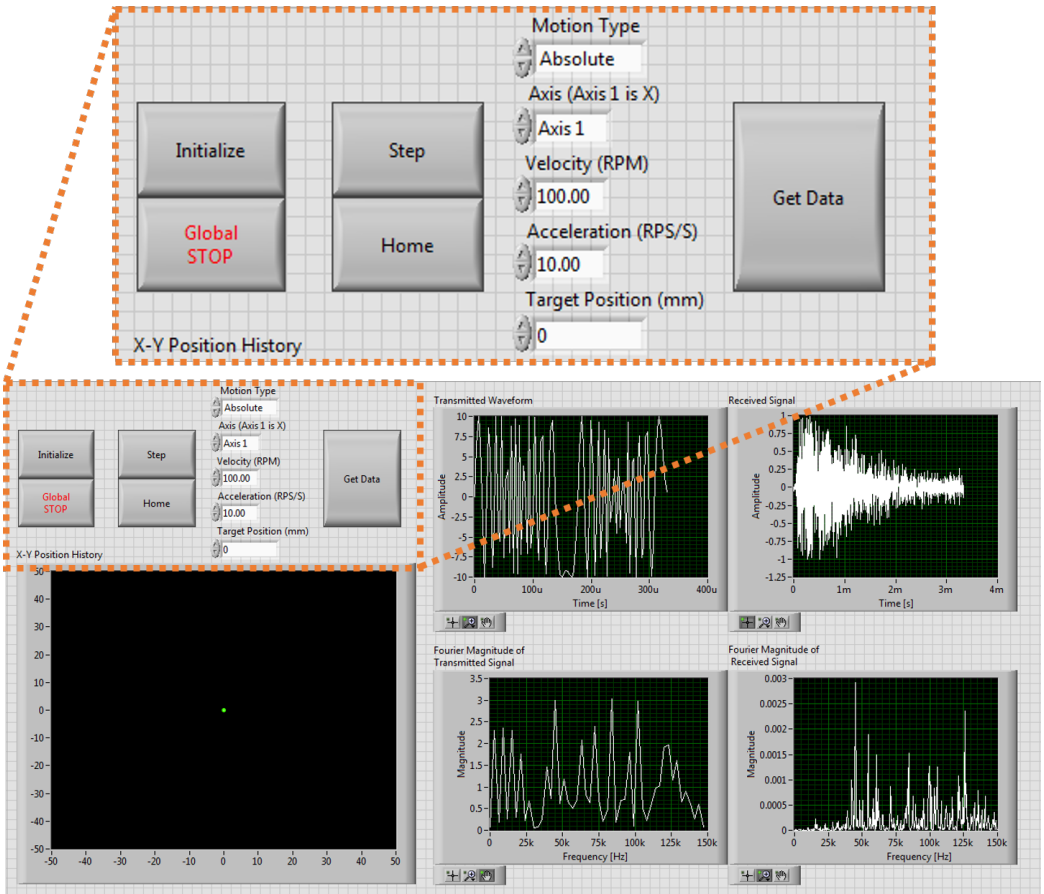

Figure A.2. Graphical user interface for data acquisition system.

face, Integrate, Analyze, and Validate. Packages included in each section are listed on the NI website. NI hardware components have recommended LabVIEW packages that can be used for hardware interfacing, also known as device drivers.

Within LabVIEW, there are abundant example VIs available in the help drop down menu that come prebuilt with the main installation and with each package. It is generally a good idea to search these examples before beginning the development of any application to see if the VI you need already exists, or if there is an example that can be easily modified to fit your application needs. These examples are also a good source to turn to for learning how to use a certain VI if the help dialog box isn't clear enough.

A.2.1 LabVIEW Helps

To get started, NI provides many tutorials on their website for a complete introduction to LabVIEW and it's capabilities. Comprehensive introductory books are also for sale. There also exist many forums managed by NI where more specific questions can be addressed. If even more specific help is necessary, NI support can be contacted by submitting a service request through your NI user account. NI support may request a copy of the code or a screen shot of the error, and may need a few days to explore the problem. Each support request opened in your user account will be assigned to a specific person that you can then contact by phone or email for help with that problem.

A.2.2 Installation and Registration

The current system runs the Windows version of LabVIEW 2015 64-bit and 32-bit, and uses the Motion, DAQmx, and SignalExpress packages. The 32-bit version of LabVIEW is required for LabVIEW use of NI motion control. The 64-bit environment was used to develop the DAQ application VI, and then the DAQ VI was inserted into the motion control VI, which was developed in the 32-bit environment.

Other possibly useful packages for future development of this system could include: Advanced Signal Processing Toolkit, Sound and Vibration Toolkit with included Sound and Vibration Assistant, NI Motion Assistant, and LabVIEW SoftMotion Module.

A.3 System Setup

System setup can be somewhat detailed, and, if not done correctly the system will not work as intended. It is important to power all of the components in the right sequence to allow the NI components to boot up before use. Please adhere to the following instructions for powering on the system, setting up the scanning program, and preparing the specimen for scanning. A diagram of the system componenets can been found in Figure A.3 for reference. Ensure all hardware componenets are connected correctly now. See also Figure A.2 to identify the correct buttons mentioned below.

A.3.1 Power on System

Before the system is powered, all of the components should be plugged into the same power strip, but the DC power supply that powers the UMI should be unplugged. The cords from the drivers should also be unplugged from the UMI to start. The order in which you should power the system is as follows:

1. Turn on the power strip.
2. Turn on the main power switch for the PXI chassis in the back, and then flip the standby power switch in front.

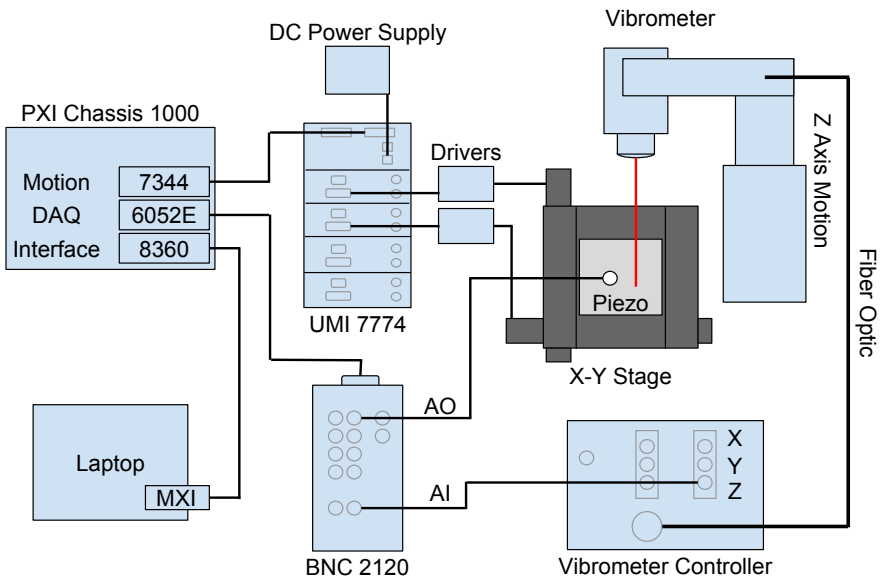

Figure A.3. System diagram of the data acquisitions system, including all of the major components and connections.

3. . . . wait 20 seconds . . .

4. Turn on the computer (the computer should only be powered on once the PXI chassis is turned on and running).

5. Plug in the DC power supply, powering the UMI.

6. Turn the key on the LDV controller to turn on the LDV laser.

7. Remove the cap from the LDV sensor head when ready to start scan.

A.3.2 Prepare Program Parameters

After the computer is booted up, the following steps will need to be taken to start using the system.

1. Initialize the UMI (see instructions below).

2. Plug the drivers into the UMI board into the correct axes.

3. Locate and open the program called "DAQ Motion Stage."

4. Once in the program, click initialize in the top left hand corner.

5. Position the stage at the home position using the **Step** function.

6. Press home to set the start position of the scan.

7. Press get data and enter the parameters of the scan.

The UMI is initialized by going in to the Measurement and Automation Explorer (MAX) that is installed with LabVIEW. In the far left pane go to My System - Devices and Interfaces - NI Motion Devices - PXI-7344(1), and then click Initialize in the pane just to the right. The "DISABLED" LED should now be off. The motor drivers can now be plugged into UMI 7774 (ensure the drivers are set to the correct current and step size). Next, find the LabVIEW program used to control the stage, called "DAQ Motion Stage". This program should be opened in the 2015 version of 32 bit LabVIEW. A control panel should open with the buttons described above, and some indicators showing a grid path and four waveforms. Click **Initialize** to clear the motion card and disable limit switches.

After you have prepared the specimen for scanning and attached it to the plate (see "Prepare Specimen for Scanning" section below), use the **Step** button to position the LDV where you would like to start the scan, and press **Home** to set the current location as the scan start position. Press **Get Data** and use the pop up "Get User Input" window to fill in the parameters desired for the scan. When you press "OK," the scan will start. Each point scan will be appended in a new row in a file that is saved to the location indicated in the "Get User Input" window. A scan of 10,000 points is usually about one half of a gigabyte, so be sure to have enough room.

A.3.3 Prepare Specimen for Scanning

The following procedure is the current best practice for preparing a plate to be scanned:

1. Glue a piezoelectric sensor the the bottom of the plate in the correct location.
2. Place three or four bits of mounting tape on the bottom of the plate.
3. Align the edges of the plate to the moving portion of the stage using two straight edges, and press down firmly.
4. Use the center of the plate to focus the laser using the Z axis motion laser mount. Move the sensor head up or down until the three laser dots converge on a single point.
5. Align the laser to the correct location and **Home** the position.
6. Use the **Get Data** button to enter the scanning parameters and start the scan.

The metal plate being scanned is positioned on the motion stage using two straight edges to be flush with the moving portion of the stage, and mounted using three or four $12 \times 12 \times 3$ mm pieces of mounting tape. The sensor, previously glued to the underside of the plate, is connected to the DAQ interface analog output using banana clips. This cord needs to be secured to the moving portion of the stage in such a way that it will not put stress on the sensor, interfere with the motion of the stage, or come loose throughout the entire grid inspection movement. A piece of 5 by 5 mm mounting tape is placed flush with the corner that is opposite from the corner where the scan is started. The scan is started

2 grid divisions outside the material from one corner in both the X and Y directions, and continued until at least 2 grid divisions outside the material on the far corner. This can best be done by aiming the laser directly at the corner of the material, then stepping back 2 times the grid division size in both axes and using the **Home** button to set the start position of the scan. The resulting scan will shows the entire wavefield of the plate, and also the orientation because of the tape that will also be clearly visible on the scan. It may also be helpful to use the **Step** function to move the entire scanning grid distance to ensure that there are no collisions or limits reached.

A.4 System Troubleshooting

This section addresses some of the common problems found using the DAQ system, and their solutions.

A.4.1 Motion Stage Will not Move and/or the UMI Is Disabled

If the motion stage doesn't move when commanded, this is often because the setup sequence was not done correctly. Usually the UMI was not initialized through MAX, the motor drivers were plugged in during power up and initialization, or sometimes the UMI is disabled after a long idle period for the system. These all can be fixed by closing all LabVIEW programs, ensuring that the drivers are disconnected from the UMI, re-initializing the UMI through MAX, plugging the drivers back in, and then opening the DAQ motion stage program and initializing the whole system with the initialize button. If this doesn't work, shut everything down and go through the system setup again with the correct sequence.

If a correct setup does not fix the problem, ensure that all the wires are connected correctly and try again.

A.4.2 Received Signal Does not Look Correct

If the received signal from the LDV does not look correct, there could a number of things causing this. First, make sure that the cap is removed from the optic lens of the LDV, and that the laser is focused on the surface of the material. This can be done by using the vertical mount controller to adjust the height of the laser head until the three laser dots of the LDV converge into one on the surface of the material. Also ensure that the piezoelectric

sensor is receiving the excitation signal. Usually there is an audible sound if it is, but this can also be checked by connecting the excitation signal to a separate oscilloscope, and verifying connections with a multimeter.

If the received signal still does not look correct, then the LDV controller might need to be plugged into the same power strip as the NI equipment to ensure that there are no ground loops. If all of the equipment is plugged into the same power strip already, a different power strip may be required. There is likely some grounding problem created by the power strip that is interfering with the functionality of the LDV.

If the signal weakens part-way through a scan, the leads connecting the piezoelectric sensor and the DAQ card interface likely came loose. The wires may need to be more effectively secured so as to not come loose or interfere with the motion of the stage. This can best be done by taping the wires to the same surface that the plate is mounted to, and then doing a trial run of the full motion of the stage to make sure that nothing will snag.

If there are random signals dispersed through the scan that do not look correct, but the rest do, it could be that the system was not entirely isolated during the scan and someone entered the room and/or tampered with the scans while in progress. Alternatively, the surface quality of the material might be affecting the scans. The material surface might need to be polished, or reflective tape applied to the surface, for better measurements.

A.4.3 Computer Goes to Sleep in the Middle of a Scan

Computer settings can be adjusted to not allow the system to go to sleep. Instructions for this can be found online for Windows. It might also be important to make sure your internet connection and MXI and USB ports don't go to sleep or time out either. This would be done through the same settings.

A.4.4 New Scan Saved Over Previous Scan

If the name of the scan is not changed in-between scans, the data and description files from the previous scan may be either written over, or the new data may be appended to the old data. If the files were appended to, but not written over, then the data might be saved by opening the files and extracting the correct lines of data. However, sometimes the data are just lost, so be very careful to change the name of the destination file when starting a new scan.

REFERENCES

[1] P. J. Shull, *Nondestructive Evaluation: Theory, Techniques, and Applications*. Boca Raton, FL, USA: CRC press, 2002.

[2] C. R. Farrar *et al.*, "Dynamic characterization and damage detection in the I-40 bridge over the Rio Grande," Los Alamos National Lab., NM, USA, Tech. Rep. LA–12767-MS, 1994.

[3] H. Prion, C. Ventura, and M. Rezai, "Damage detection of steel frame by modal testing," in *Proc. 14th Int. Modal Analysis Conf.*, vol. 2768, 1996, p. 1430.

[4] N. Robinson, L. Peterson, G. James, and S. Doebling, "Damage detection in aircraft structures using dynamically measured static flexibility matrices," Los Alamos National Lab., NM, USA, Tech. Rep. LA-UR–95-4382, 1996.

[5] S. W. Doebling, C. R. Farrar, M. B. Prime, and D. W. Shevitz, "Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: a literature review," Los Alamos National Lab., NM, USA, Tech. Rep. LA-13070-MS, 1996.

[6] J. Kudva, N. Munir, and P. Tan, "Damage detection in smart structures using neural networks and finite-element analyses," *Smart Mater. Struct.*, vol. 1, no. 2, p. 108, Feb. 1992.

[7] J. B. Harley and J. M. Moura, "Sparse recovery of the multimodal and dispersive characteristics of lamb waves," *J. Acoust. Soc. Am.*, vol. 133, no. 5, pp. 2732–2745, Mar. 2013.

[8] C. R. Farrar and K. Worden, "An introduction to structural health monitoring," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 365, no. 1851, pp. 303–315, Feb. 2007.

[9] I. H. Witten, E. Frank, M. A. Hall, and C. J. Pal, *Data Mining: Practical Machine Learning Tools and Techniques*. Burlington, MA, USA: Morgan Kaufmann, 2016.

[10] G. Hinton *et al.*, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Oct. 2012.

[11] H. A. Rowley, S. Baluja, and T. Kanade, "Neural network-based face detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 1, pp. 23–38, Jan. 1998.

[12] S. M. LaValle, *Planning Algorithms*. Cambridge, UK: Cambridge University Press, 2006.

[13] K. Worden and A. Lane, "Damage identification using support vector machines," *Smart Mater. Struct.*, vol. 10, no. 3, p. 540, Sept. 2001.

[14] M. A. Rao, J. Srinivas, and B. Murthy, "Damage detection in vibrating bodies using genetic algorithms," *Comput. Struct.*, vol. 82, no. 11, pp. 963–968, May 2004.

[15] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on Signal Processing*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[16] I. Tosic and P. Frossard, "Dictionary learning," *IEEE Signal Processing Magazine*, vol. 28, no. 2, pp. 27–38, Feb. 2011.

[17] A. Rytter, "Vibrational based inspection of civil engineering structures," Ph.D. dissertation, Dept. of Bldg. Technol. Struct. Eng., Aalborg Univ., Aalborg, 1993.

[18] D. E. Bently *et al.*, "Fundamentals of rotating machinery diagnostics," *Mechanical Engineering-CIME*, vol. 125, no. 12, pp. 53–54, Apr. 2003.

[19] D. C. Montgomery, *Introduction to Statistical Quality Control*. Hoboken, NJ, USA: John Wiley & Sons, 2007.

[20] C. R. Farrar and N. A. Lieven, "Damage prognosis: The future of structural health monitoring," *Phil. Trans. R. Soc. A: Math Phys. Eng. Sci.*, vol. 365, no. 1851, pp. 623–632, Feb. 2007.

[21] J. K. Vandiver *et al.*, "Detection of structural failure on fixed platforms by measurement of dynamic response," *J. Pet. Technol.*, vol. 29, no. 03, pp. 305–310, May 1977.

[22] H. Lam, J. Ko, and C. Wong, "Detection of damage location based on sensitivity analysis," in *Proc. 13th Int. Modal Analysis Conf.*, vol. 2460, 1995, p. 1499.

[23] T. Kam and T. Lee, "Detection of cracks in structures using modal test data," *Eng. Fract. Mech.*, vol. 42, no. 2, pp. 381–387, May 1992.

[24] G. Konstantinidis, B. Drinkwater, and P. Wilcox, "The temperature stability of guided wave structural health monitoring systems," *Smart Mater. Struct.*, vol. 15, no. 4, p. 967, Jun. 2006.

[25] H. Sohn, C. R. Farrar, F. M. Hemez, D. D. Shunk, D. W. Stinemates, B. R. Nadler, and J. J. Czarnecki, "A review of structural health monitoring literature: 1996–2001," Los Alamos National Lab., NM, USA, Tech. Rep. LA-UR-02-2095, 2003.

[26] S. Dorey, "Acoustic and general methods of non-destructive testing," *Inst. Electrical Eng.*, vol. 84, no. 509, pp. 552–557, May 1939.

[27] R. B. Randall, "State of the art in monitoring rotating machinery-part 1," *Sound and vibration*, vol. 38, no. 3, pp. 14–21, Mar. 2004.

[28] P. McFadden and J. Smith, "Model for the vibration produced by a single point defect in a rolling element bearing," *Sound and Vibration*, vol. 96, no. 1, pp. 69–82, Sept. 1984.

[29] J. Antoni and R. B. Randall, "Differential diagnosis of gear and bearing faults," *J. Vib. Acoust*, vol. 124, no. 2, pp. 165–171, Mar. 2002.

[30] R. B. Randall, "A new method of modeling gear faults," *J. Mech. Des.*, vol. 104, no. 2, pp. 259–267, Apr. 1982.

[31] J. K. Vandiver *et al.*, "Detection of structural failure on fixed platforms by measurement of dynamic response," in *Proc. Offshore Tech. Conf.*, vol. 2267, 1975, pp. 243–252.

[32] R. Brincker, P. H. Kirkegaard, P. Andersen, and M. Martinez, "Damage detection in an offshore structure," Dept. Bldg. Technol. Struct. Eng., Aalborg Univ., Tech. Rep. 56, 1994.

[33] W. M. West Jr, "Single point random modal test technology application to failure detection," in *Proc. Shock and Vib. Inform. Center Bull.*, ser. 4, no. 52, 1982, pp. 25–31.

[34] D. L. Hunt, S. P. Weiss, W. M. West Jr, T. A. Dunlap, and S. Freesmeyer, "Development and implementation of a shuttle modal inspection system," in *Proc. 8th Int. Modal Analysis Conf.*, vol. 2, Kissimmee, FL, 1990, pp. 919–925.

[35] C. B. Scruby and L. E. Drain, *Laser Ultrasonics Techniques and Applications*. Boca Raton, FL, USA: CRC Press, 1990.

[36] B. Jaffe, *Piezoelectric Ceramics*. Amsterdam, Netherlands: Elsevier, 2012, vol. 3.

[37] H. Gao, Y. Shi, and J. Rose, "Guided wave tomography on an aircraft wing with leave in place sensors," in *Review of progress in Quantitative Nondestructive Evaluation*, vol. 760. AIP Publishing, 2005, pp. 1788–1794.

[38] P. Gerstoft, D. F. Gingras, L. T. Rogers, and W. S. Hodgkiss, "Estimation of radio refractivity structure using matched-field array processing," *IEEE Trans. Antennas Propag.*, vol. 48, no. 3, pp. 345–356, Mar. 2000.

[39] P. Gerstoft, "Inversion of seismoacoustic data using genetic algorithms and aposteriori probability distributions," *J. Acoust. Soc. Am.*, vol. 95, no. 2, pp. 770–782, Jun. 1994.

[40] P. Sriram, J. Craig, and S. Hanagud, "A scanning laser doppler vibrometer for modal testing," *Int. J. Anal. Exper. Modal Analysis*, vol. 5, Jul. 1990.

[41] W. J. Staszewski, B. Lee, L. Mallet, and F. Scarpa, "Structural health monitoring using scanning laser vibrometry: I. lamb wave sensing," *Smart Mater. Struct.*, vol. 13, no. 2, p. 251, Feb. 2004.

[42] H. Sohn, "Effects of environmental and operational variability on structural health monitoring," *Phil. Trans. R. Soc. A: Math Phys. Eng. Sci.*, vol. 365, no. 1851, pp. 539–560, Feb. 2007.

[43] O. Salawu and C. Williams, "Damage location using vibration mode shapes," in *Proc. 12th Inter. Modal Analysis Conf.*, vol. 2251, 1994, p. 933.

[44] J. Chance, G. Tomlinson, and K. Worden, "A simplified approach to the numerical and experimental modelling of the dynamics of a cracked beam," in *Proc. 12th Inter. Modal Analysis Conf.*, 1994, pp. 778–778.

[45] A. Aktan, K. Lee, C. Chuntavan, and T. Aksel, "Modal testing for structural identification and condition assessment of constructed facilities," in *Proc. 12th Inter. Modal Analysis Conf.*, 1994, pp. 462–462.

[46] T. Toksoy and A. Aktan, "Bridge-condition assessment by modal flexibility," *Experimental Mechanics*, vol. 34, no. 3, pp. 271–278, Apr. 1994.

[47] A. Pandey and M. Biswas, "Damage detection in structures using changes in flexibility," *J. of Sound and Vibration*, vol. 169, no. 1, pp. 3–17, Jan. 1994.

[48] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford, UK: Oxford University Press, 1995.

[49] J. Ghaboussi, J. Garrett Jr, and X. Wu, "Knowledge-based modeling of material behavior with neural networks," *J. Engineering Mechanics*, vol. 117, no. 1, pp. 132–153, Jan. 1991.

[50] C. J. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121–167, Jun. 1998.

[51] K. Worden and G. Manson, "Damage identification using multivariate statistics: Kernel discriminant analysis," *Inv. Problems Eng.*, vol. 8, no. 1, pp. 25–46, Oct. 2000.

[52] A. Widodo and B.-S. Yang, "Support vector machine in machine condition monitoring and fault diagnosis," *Mech. Sys. and Sig. Process.*, vol. 21, no. 6, pp. 2560–2574, Aug. 2007.

[53] K. F. Graff, *Wave Motion in Elastic Solids*. North Chelmsford, MA, USA: Courier Corporation, 1975.

[54] J. Achenbach, *Wave Propagation in Elastic Solids*. Amsterdam, Netherlands: Elsevier, 2012, vol. 16.

[55] F. B. Jensen, W. A. Kuperman, M. B. Porter, and H. Schmidt, *Computational Ocean Acoustics*. Berlin, Germany: Springer Science & Business Media, 2000.

[56] P. C. Magnusson, A. Weisshaar, V. K. Tripathi, and G. C. Alexander, *Transmission Lines and Wave Propagation*. Boca Raton, FL, USA: CRC Press, 2000.

[57] D. Alleyne and P. Cawley, "A two-dimensional fourier transform method for the measurement of propagating multimode signals," *J. Acoust. Soc. Am.*, vol. 89, no. 3, pp. 1159–1168, Sept. 1991.

[58] W. Gao, C. Glorieux, and J. Thoen, "Laser ultrasonic study of lamb waves: determination of the thickness and velocities of a thin plate," *Int. J. Eng. Sci.*, vol. 41, no. 2, pp. 219–228, Jan. 2003.

[59] M. Niethammer, L. J. Jacobs, J. Qu, and J. Jarzynski, "Time-frequency representation of lamb waves using the reassigned spectrogram," *J. Acoust. Soc. Am.*, vol. 107, no. 5, pp. L19–L24, Mar. 2000.

[60] W. Prosser, M. D. Seale, and B. T. Smith, "Time-frequency analysis of the dispersion of lamb modes," *J. Acoust. Soc. Am.*, vol. 105, no. 5, pp. 2669–2676, Jan. 1999.

[61] F. Li, G. Meng, L. Ye, Y. Lu, and K. Kageyama, "Dispersion analysis of lamb waves and damage detection for aluminum structures using ridge in the time-scale domain," *Meas. Sci. Technol.*, vol. 20, no. 9, p. 095704, Jul. 2009.

[62] A. Raghavan and C. E. Cesnik, "Guided-wave signal processing using chirplet matching pursuits and mode correlation for structural health monitoring," *Smart Mater. Struct.*, vol. 16, no. 2, p. 355, Jan. 2007.

[63] Y. Wang, Z. Longxiang, L. Fan, and S. Fengrui, "Guided waves modes identification in pipes detection by application of the matching pursuit method," in *Proc. 10th Int. Conf. Elec. Meas. & Instr.*, vol. 4. IEEE, 2011, pp. 50–53.

[64] J. S. Hall and J. E. Michaels, "A model-based approach to dispersion and parameter estimation for ultrasonic guided waves," *J. Acoust. Soc. Am.*, vol. 127, no. 2, pp. 920–930, Nov. 2010.

[65] ——, "Model-based parameter estimation for characterizing wave propagation in a homogeneous medium," *Inverse Problems*, vol. 27, no. 3, p. 035002, Feb. 2011.

[66] A. Raghavan and C. E. Cesnik, "Effects of elevated temperature on guided-wave structural health monitoring," *J. Intell. Mater. Syst. Struct.*, vol. 19, no. 12, pp. 1383–1398, May 2008.

[67] A. Degtyar, W. Huang, and S. Rokhlin, "Wave propagation in stressed composites," *J. Acoust. Soc. Am.*, vol. 104, no. 4, pp. 2192–2199, Oct. 1998.

[68] G. Konstantinidis, P. D. Wilcox, and B. W. Drinkwater, "An investigation into the temperature stability of a guided wave structural health monitoring system using permanently attached sensors," *IEEE Sensors J.*, vol. 7, no. 5, pp. 905–912, May 2007.

[69] A. Mazzeranghi and D. Vangi, "Methodology for minimizing effects of temperature in monitoring with the acousto-ultrasonic technique," *Exper. Mech.*, vol. 39, no. 2, pp. 86–91, Jun. 1999.

[70] Y. Lu and J. E. Michaels, "A methodology for structural health monitoring with diffuse ultrasonic waves in the presence of temperature variations," *Ultrasonics*, vol. 43, no. 9, pp. 717–731, Oct. 2005.

[71] J. E. Michaels and T. E. Michaels, "Detection of structural damage from the local temporal coherence of diffuse ultrasonic signals," *IEEE Trans. Ultrason., Ferroelect., Freq. Control*, vol. 52, no. 10, pp. 1769–1782, Dec. 2005.

[72] Y. Lu and J. E. Michaels, "Feature extraction and sensor fusion for ultrasonic structural health monitoring under changing environmental conditions," *IEEE Sensors J.*, vol. 9, no. 11, pp. 1462–1471, Sept. 2009.

[73] T. Clarke, F. Simonetti, and P. Cawley, "Guided wave health monitoring of complex structures by sparse array systems: Influence of temperature changes on performance," *J. Sound and Vibration*, vol. 329, no. 12, pp. 2306–2322, Jun. 2010.

[74] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, Apr. 2006.

[75] R. A. Swartz, D. Jung, J. P. Lynch, Y. Wang, D. Shi, and M. P. Flynn, "Design of a wireless sensor for scalable distributed in-network computation in a structural health monitoring system," in *Proc. 5th Int. Workshop Struct. Health Monitor.* Princeton, NJ, USA: Citeseer, 2005, pp. 12–14.

[76] A. Tolstoy, "Applications of matched-field processing to inverse problems in underwater acoustics," *Inv. Problems*, vol. 16, no. 6, p. 1655, Jun. 2000.

[77] T. E. Michaels and J. E. Michaels, "Sparse ultrasonic transducer array for structural health monitoring," in *Proc. AIP Conf.* Philadelphia, PA, USA: IOP Institute of Physics Publishing LTD, 2003, pp. 1468–1475.

[78] Y. Lu and J. E. Michaels, "Consideration of surface variations on ultrasonic structural health monitoring," in *Proc. 6th Int. Workshop Struct. Health Monitor.* Lancaster, PA, USA: DEStech Publications, Inc., 2007, pp. 1275–1282.

[79] A. Mal, F. Ricci, S. Banerjee, and F. Shih, "A conceptual structural health monitoring system based on vibration and wave propagation," *Struct. Health Monitor.*, vol. 4, no. 3, pp. 283–293, Sept. 2005.

[80] X. Zhao *et al.*, "Active health monitoring of an aircraft wing with embedded piezoelectric sensor/actuator network: I. defect detection, localization and growth monitoring," *Smart Mater. Struct.*, vol. 16, no. 4, p. 1208, Jun. 2007.

[81] C. H. Wang, J. T. Rose, and F.-K. Chang, "A synthetic time-reversal imaging method for structural health monitoring," *Smart Mater. Struct.*, vol. 13, no. 2, p. 415, Mar. 2004.

[82] J. E. Michaels and T. E. Michaels, "Enhanced differential methods for guided wave phased array imaging using spatially distributed piezoelectric transducers," in *Proc. AIP Conf.*, vol. 820. Philadelphia, PA, USA: IOP Institute of Physics Publishing LTD, 2005, pp. 837–844.

[83] P. Fromme, "Monitoring of plate structures using guided ultrasonic waves," in *Proc. 34th Ann. Rev. Prog. Quantitative Nondestructive Eval.*, vol. 975. Melville, NY, USA: AIP Publishing, 2008, pp. 78–85.

[84] L. Mallet, B. Lee, W. J. Staszewski, and F. Scarpa, "Structural health monitoring using scanning laser vibrometry: Ii. lamb waves for damage detection," *Smart Mater. Struct.*, vol. 13, no. 2, p. 261, Feb. 2004.

[85] W. Leong, W. J. Staszewski, B. Lee, and F. Scarpa, "Structural health monitoring using scanning laser vibrometry: Iii. lamb waves for fatigue crack detection," *Smart Mater. Struct.*, vol. 14, no. 6, p. 1387, Oct. 2005.

[86] W. J. Staszewski, B. Lee, and R. Traynor, "Fatigue crack detection in metallic structures with lamb waves and 3D laser vibrometry," *Meas. Sci. Technol.*, vol. 18, no. 3, p. 727, 2007.

[87] Y.-K. An, B. Park, and H. Sohn, "Complete noncontact laser ultrasonic imaging for automated crack visualization in a plate," *Smart Mater. Struct.*, vol. 22, no. 2, p. 025022, Jan. 2013.

[88] M. Ruzzene, "Frequency–wavenumber domain filtering for improved damage visualization," *Smart Mater. Struct.*, vol. 16, no. 6, p. 2116, Oct. 2007.

[89] H. Sohn *et al.*, "Delamination detection in composites through guided wave field image processing," *Compos. Sci. Technol.*, vol. 71, no. 9, pp. 1250–1256, Jun. 2011.

[90] K. S. Alguri and J. B. Harley, "Consolidating guided wave simulations and experimental data: A dictionary leaning approach," in *Proc. SPIE Health Monitor. Struct. Bio. Sys.*, vol. 9805. International Society for Optics and Photonics, 2016, pp. 1–10.

[91] Y. K. Esfandabadi, L. De Marchi, A. Marzaniz, and G. Masetti, "Damage imaging through compressed wavefield recovery in lamb wave inspections," in *Proc. IEEE Int. Ultrason. Symp.* Tours, France: IEEE, 2016, pp. 1–4.

[92] J. B. Harley and J. M. Moura, "Dispersion curve recovery with orthogonal matching pursuit," *J. Acoust. Soc. Am.*, vol. 137, no. 1, pp. EL1–EL7, Oct. 2015.

[93] T. Di Ianni, L. De Marchi, A. Perelli, and A. Marzani, "Compressive sensing of full wave field data for structural health monitoring applications," *IEEE Trans. Ultrason., Ferroelect., Freq. Control*, vol. 62, no. 7, pp. 1373–1383, Jul. 2015.

[94] J. B. Harley, "Predictive guided wave models through sparse modal representations," *Proc. IEEE*, vol. 104, no. 8, 2016.

[95] Y. C. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. 27th Asilomar Conf. on Signals, System, and Computers*. IEEE, 1993, pp. 40–44.

[96] K. Engan, S. O. Aase, and J. H. Husoy, "Method of optimal directions for frame design," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, vol. 5. IEEE, 1999, pp. 2443–2446.

[97] M. Bianco and P. Gerstoft, "Dictionary learning of sound speed profiles," *J. Acoust. Soc. Am.*, vol. 141, no. 3, pp. 1749–1758, Feb. 2017.

[98] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient implementation of the k-svd algorithm using batch orthogonal matching pursuit," *CS Technion*, vol. 40, no. 8, pp. 1–15, Apr. 2008.

[99] J. S. Turek, M. Elad, and I. Yavneh, "Clutter mitigation in echocardiography using sparse signal separation," *J. Biomedical Imaging*, vol. 2015, p. 9, Jan. 2015.

[100] S. Lesage, R. Gribonval, F. Bimbot, and L. Benaroya, "Learning unions of orthonormal bases with thresholded singular value decomposition," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, vol. 5, Mar. 2005, pp. 293–296.

[101] J. B. Harley *et al.*, "Application of mellin transform features for robust ultrasonic guided wave structural health monitoring," in *Proc. AIP Conf.*, vol. 1430, no. 1. AIP Publishing, 2012, pp. 1551–1558.

[102] G. Giuliani, M. Norgia, S. Donati, and T. Bosch, "Laser diode self-mixing technique for sensing applications," *J. Optics A: Pure and Applied Optics*, vol. 4, no. 6, p. S283, Nov. 2002.

[103] M. S. Allen and M. W. Sracic, "A new method for processing impact excited continuous-scan laser doppler vibrometer measurements," *Mech. Sys. Sig. Process.*, vol. 24, no. 3, pp. 721–735, Nov. 2010.