2016

Temperature And Pressure Effects On Elastic Properties Of Relaxor Ferroelectrics And Thermoelectrics: A Resonant Ultrasound Spectroscopy Study

Sumudu Tennakoon
University of Mississippi

TEMPERATURE AND PRESSURE EFFECTS ON ELASTIC PROPERTIES OF RELAXOR FERROELECTRICS AND THERMOELECTRICS : A RESONANT ULTRASOUND SPECTROSCOPY STUDY

A Dissertation
presented in partial fulfillment of requirements
for the degree of Degree of Doctor of Philosophy
in the Department of Physics and Astronomy
The University of Mississippi
University, MS 38677

by

SUMUDU P. TENNAKOON

August 2016

ABSTRACT

Relaxor ferroelectric lead magnesium niobate-lead titanate (PMN-PT) material exhibits exceptional electromechanical properties. The material undergoes a series of structural phase transitions with changes in temperature and the chemical composition. The work covered in this dissertation seek to gain insight into the phase diagram of PMN-PT using temperature and pressure dependence of the elastic properties. Single crystal PMN-PT with a composition near morphotropic phase boundary (MPB) was investigated using a resonant ultrasound spectroscopy (RUS) methodologies in the temperature range of 293 K – 800 K and the pressure range from near vacuum to 3.4 MPa. At atmospheric pressure, significantly high acoustic attenuation of PMN-PT is observed at temperatures below 400 K. A strong stiffening is observed in the temperature range of 400 K – 673 K, followed by a gradual softening at higher temperatures. With varying pressure, an increased pressure sensitivity of the elastic properties of PMN-PT is observed at the temperatures in the stiffening phase. Elastic behavior at elevated temperatures and pressures were studied for correlations with the ferroelectric domains at temperatures below the Curie temperature (T_C), the locally polarized nano-regions, and an existence of pseudo-cubic crystalline at higher temperatures between (T_C and T_B). Thermoelectric lanthanum tellurides and skutterudites are being investigated by NASA's Jet Propulsion Laboratory for advanced thermoelectric generates (TEGs). Effects of nickel (Ni) doping on elastic properties of lanthanum tellurides at elevated temperatures were investigated in the temperature range of 293 K – 800 K. A linear stiffening was observed with increased Ni content in the material. Elastic properties of p-type and n-type bismuth-based skutterudites were investigated in the temperature range of 293 K – 723 K. Elastic properties of rare-earth doped strontium titanate were also investigated in the temperature range of 293 K – 750 K.

DEDICATION

To my parents who inspired me to explore the world of science in very early stage of my childhood, and all my teachers who shared their knowledge with me, and guided me through thins academic journey.

ACKNOWLEDGEMENTS

This dissertation would not be possible without the mentoring, support, kindness and love of the following people I very gratefully mention below. My first acknowledgment goes to Dr. Joseph Gladden for giving me the opportunity to explore the elastic properties of materials using the RUS methodologies, and for being an excellent mentor, being very supportive, sharing his wisdom, and for guiding me to move forward as a research professional. I would also gratefully thank members of my dissertation committee Dr. Kevin Beach, Dr.Igor Ostrovskii, Dr. Raju Mantena for their generous time commitment in the dissertation procedure.

I am very grateful to Dr. Cremaldi, Dr. Luca Bombelli, Dr. Robert Kroger, and Faculty and Staff of the Physics Department for the support and encouragement to go through the physics Ph.D. program since submission of my application. Dr. Claire Mischker and her staff at the Graduate writing center for being kind and helpful in reading my manuscript to correct the language. The university of Mississippi Libraries and library staff for their support in finding me research resources. University of Mississippi Graduate School for the financial support through assistantships and fellowships, and for giving me an opportunity to develop my skills as a capable scholar. Faculty and Staff of the National Center for Physical Acoustics for making a welcoming and supportive environment for graduate student research.

My experience at the University of Mississippi is truly amazing, and I am very grateful to the different departments and officers, faculty and staff members and fellow students who supported me to shape up my professional and social skills as well as creating a memorable time at this wonderful university. There, I would specially mention Office of International Programs, Department of Campus Recreation, Deans of Students Office, and Graduate Student Council.

Support in the lab was very valuable to successfully conduct the experimental work. I would sincerely thank Dr. Rasheed Adebisi, Dr. Qin Zhang, Dr. Guangyan Li for their time in sharing their knowledge and experience with me and for the encouragement at the initial stage of my research with the material science group at NCPA. Ashoka Karunarathna who took a work share in some of the ongoing lab projects in the last few months. Financial and support and opportunities of studying advanced materials were an invaluable part of this dissertation work. There, I would sincerely thank NASA/JPL research grant for the thermoelectric studies, Diapak Inc., Delphi Inc., Cisco Inc., Crosslink Inc., and Dr. Amit Pandey and Materials Reliability Research team at the LG Fuel Cell Systems for supporting me through various projects. Dr. Sriparna Bhattacharya and the thermoelectric material research group at the Clemson University for being very the supportive in their collaborative research.

None of the progress I made so far would not be possible without a good foundation and care from many people in my life. I am very grateful to my beloved parents, Sunil and Anula Tennakoon for raising me a good human being and more importantly for enhancing my curiosity about nature and science since my early childhood, and for being supportive and inspiring me to explore the world through science, and also for proofreading my dissertation. My beloved wife Maheesha for sharing lots of hardships in the years of this journey from my high school time. All my family members who are always happy to see my success. Also, all my teachers who shared their wisdom and guided me to success as a scholar. I am very blessed to have good friends and colleagues who supported me in many different ways. There, I especially thank my friend Jehan Senevirathne for kindly spending his time proofreading my dissertation, and many of my Sri Lankan friends in Oxford, MS for their different levels of support while working on my dissertation. Thank you all !

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Materials for Solid State Electrical Energy Harvesting Devices

During the most recent chapters of human history, we have consumed different forms of energy for our survival and comfort. Naturally existing energy sources: heat, light, mechanical and chemical energy are considered as primary energy sources we used for a long time where the mechanisms to convert one energy form to another were discovered with the progression of the human knowledge. With the development of science and technology during the last couple of centuries, electricity emerged as a common form of energy that can be easily converted to all other forms, and vice versa. In the present day life, electricity has become the most essential form of energy to keep our lives running, and a significant amount of science and technology research are focused on enhancing transduction mechanisms that can be used for the conversion between other energy sources and electricity.

Figure 1.1 illustrates some examples of the energy conversion applications of solid state devices using properties of energy transduction materials. Photovoltaic materials that convert light into electricity are widely used in solar panels. Thermoelectric materials that convert thermal gradients into electric potential are used in thermoelectric generators (TEG). Piezoelectric materials that convert mechanical vibration into electricity and vice versa are commonly used in a variety of applications, including vibration energy harvesting devices and sensor applications. Due to the absence of moving parts and operating fluids, these solid state devices are well suited to power a variety of devices, scaling from miniature electronic equipment to propulsion vehicles, where less maintenance is required. These devices can also be considered as ideal energy sources to power isolated and remote systems.

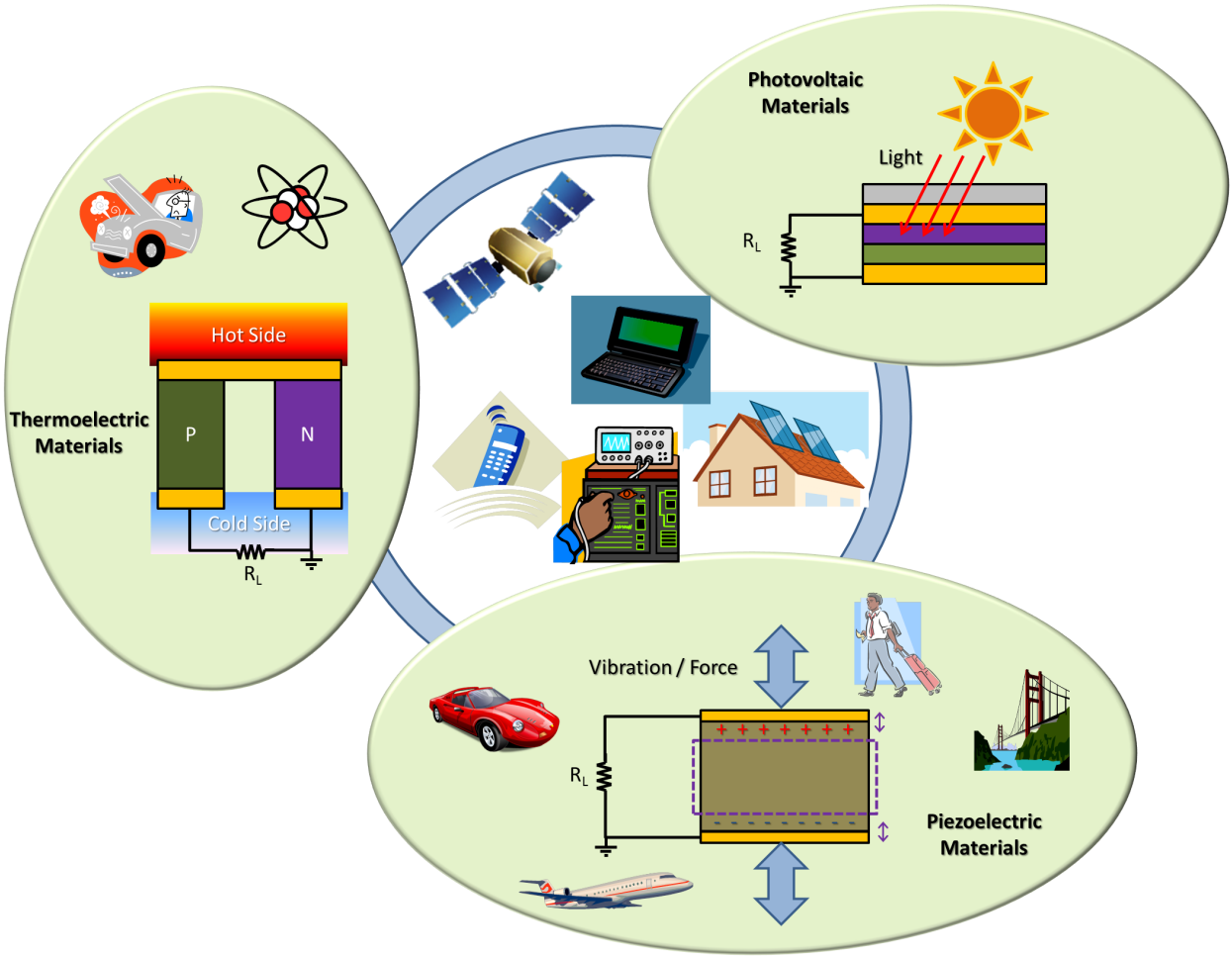

Figure 1.1. Solid state devices to harvest electric energy from underutilized sources.

Acoustic energy is considered a viable candidate for powering isolated and miniature systems with low power requirements. Examples of such systems are structural health monitoring systems and display devices that need a supply of long term uninterrupted power. Acoustic energy harvesting units can be embedded or attached to these systems where energy can be harvested from different vibration sources to which the system could be exposed. As an example, our human body has vibration sources including heartbeat, movements and interactions with external objects, etc. Moving or stationary structures such as aerospace or ground vehicles, machines and bridges have vibration sources with much higher energy levels. The reverse effect of converting an electric signal into mechanical response is used in actuators, acoustic transducers in medical systems, non destructive testing (NDT) and buried object detection.

Piezoelectric materials that convert mechanical strains into a voltage response are commonly used as transduction elements for the conversion between acoustic and electric energies. However, there exist some limitations associated with piezoelectric materials, such as energy conversion efficiency, robustness, resistance to the designated usage environment (temperature, pressure, chemical activities, etc.), frequency response, and sensitivity. Improving properties of current materials and developing new materials are a higher interest in present day research and development in order to address these limitations. During these material development processes, it is important to have a thorough understanding about the physical properties of piezoelectric materials in different temperature and pressure environments and their relationship to the energy conversion mechanisms.

Lead magnesium niobate lead titanate (PMN-PT) is a relaxor ferroelectric material exhibiting higher electromechanical properties than conventional piezoelectric materials such as PZT. PMN-PT material properties are shown in Table 4.1. These remarkable properties attract higher attention to PMN-PT in both research and industrial fields. It is reported in the literature that PMN-PT undergoes certain structural phase transitions with changes in temperature and chemical composition (PT content) [40]. However, the effects of hydrostatic

pressure on the material properties and the phase transition of this material are not very well understood. The main focus of this study is to investigate temperature and pressure effects on phase transitions of relaxor ferroelectric materials and other high temperature energy related materials by applying resonant ultrasound spectroscopy (RUS) methodologies. Another goal is to gain more insight into the phase diagram of the PMN-PT material using the elastic properties obtained under variable temperature and pressure conditions.

1.2 Acoustic methods of finding elastic constants of solid materials

A solid material changes its shape when an external load is applied. If it returns to its original shape after the force is removed, the material is said to be in its elastic limit. All materials are elastic to some degree, and elasticity is considered as a fundamental property of materials. For small deformations, strains (ϵ_{ij}) and stresses (σ_{ij}) are related by the elastic constants (C_{ij}), expressed in the form of generalized Hook's Law in the linear elasticity theory of elasticity $\sigma_{ij} = C_{ijkl}\epsilon_{kl}$ [60].

Elastic constants are related to different macroscopic and microscopic properties of a material. Elastic constants measured over a range of temperatures and pressures can be used to probe phase transitions. The same procedure can be used to determine properties of a material as a function of temperature and pressure. Such properties are specific heat capacity at constant volume (C_v), coefficient of thermal expansion (CTE), isothermal compressibility, and Debye temperature (Θ_D). Figure 1.2 shows a plot of potential energy versus the inter-atomic spacing following a simple inter-atomic potential model [37, 53]. The actual inter-atomic potentials in crystals can be much more complicated but closely follow a quadratic curve near the minimum. Elastic constants are the second derivative (curvature) of the interactomic potential energy curve. The sharper the curvature, the larger the elastic constants.

One such method to determine elastic constants is measuring the speed of sound in a material. In an isotropic material, sound speed of longitudinal and transverse modes can

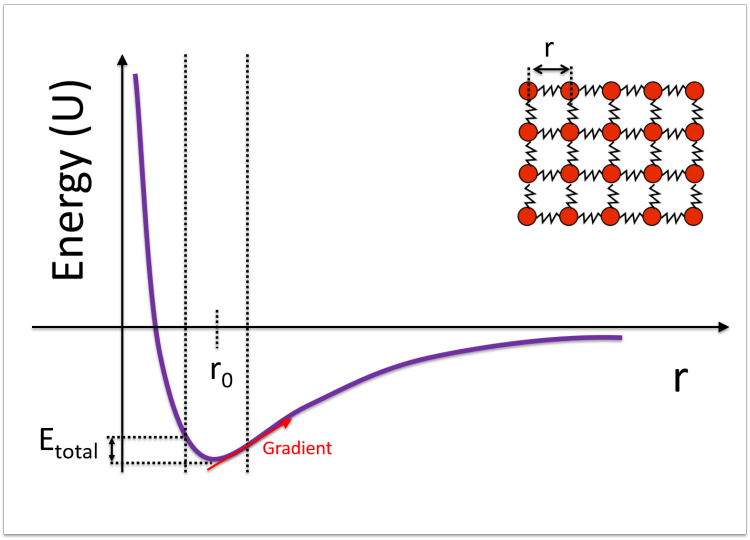

Figure 1.2. Inter-atomic potential curve. The equilibrium spacing (r_o) and the curvature at r_o determines the elastic constants. The gradient of potential at r is proportional to the restoring force toward the equilibrium spacing.

be used to determine longitudinal and shear moduli that can be used to calculate other elastic constants. This can be easily achieved by employing the pulse-echo technique. In this technique, wave speeds are determined by time tracking an acoustic pulse propagating in a material. This method is also used as a nondestructive technique to test for defects and structural damages. Applying this technique is straightforward in isotropic materials, but measurements are much more difficult for anisotropic materials because wave propagation is direction dependent, resulting in the elastic tensor with more than two independent elastic constants. The pulse-echo technique requires relatively larger samples and it is adequate for materials with higher symmetry. The standard errors of this technique usually depend on measuring wave propagation distance and the travel time. Applying pulse-echo technique is problematic in the presence of voids in the material because extra echoes and scattering cause ambiguity in determining the actual travel time of the pulse.

The idea of RUS came into materials research with the efforts of Shereiber et al. and Demarest to develop computational methods to determine elastic constants of solid mineral samples [22]. With the improvement in computational power during the 1990s, and with

the hard work by Migliori and Visscher and other profounders of the technique [4, 73, 76], RUS emerged as an elegant method of investigating elastic constants. RUS came with an advantage of using small (millimeter) scale material specimens that are highly suitable for novel material research applications due to higher cost of material and production efforts.

RUS came with the advantage of determination of a complete set of elastic constants (elastic tensor) from a single measurement. That eliminates the need of having separate measurements to determine the different elastic constants, which requires mounting and remounting the sample for each measurement. This makes RUS very convenient in taking the measurements under varying temperature and pressure conditions. Acoustic attenuation of the material is inversely proportional to the quality factor, which can also be determined from the same experiment. Irregular changes in elastic constants tracked in a temperature or pressure range are good indications of phase transitions of a material. Therefore, performing RUS study on a material over a range of temperatures and pressures is highly applicable in critical phenomena studies such as probing phase transitions.

Knowledge about the temperature and pressure effects on the elastic properties of materials is very important in fundamental physics studies for theory confirmation and to get an insight into physical behavior of materials such as phase transitions, melting mechanisms, etc. Elastic properties also play an important part in the development of new materials and application design where the materials need to be verified for usability and reliability under actual working conditions.

1.3 Outline of Dissertation

This dissertation primarily focuses on understanding temperature and pressure effects on elastic properties of selected energy related materials: relaxor-ferroelectrics, thermoelectrics and solid oxide fuel cell ceramics.

The next chapter (Chapter 2), gives the reader introductory information on the theoretical and experimental background of the RUS methodology, which is the core technique

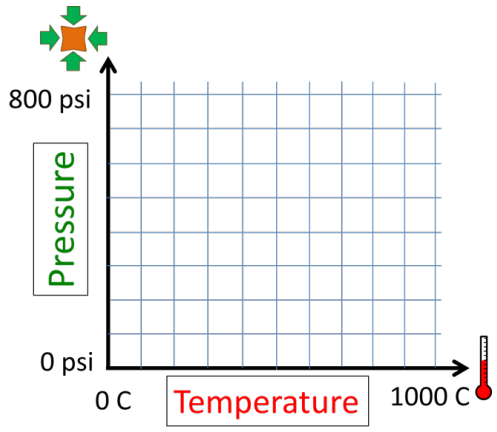

Figure 1.3. Temperature pressure exploration map for the elastic constant measurements.

used in studying elastic properties of materials at elevated temperature and pressure conditions. The discussion starts with the linear elastic theory and interpretation of elastic moduli in physical and engineering environments. At the end of this chapter, the reader will find more information on the experimental aspects of RUS, including sample preparation, instrumentation, validation of measurements and results, and data analysis procedures. This chapter ends with a brief introduction to the classical pulse-echo technique that is considered as an alternative method to characterize mechanical properties of materials.

Chapter 3 is dedicated to temperature and pressure effects on elastic properties of solids are also discussed at the end of the chapter. Implementation of RUS in studying high temperature and high pressure properties of materials will also be discussed in this chapter.

Chapter 4 is focused on on the relaxor ferroelectric material lead magnesium niobate tead titanate (PMN-PT). In the beginning of this chapter, the reader will get background information, significant material properties, current advancement, and open questions associated with the PMN-PT material. Chapter 4 also discuss experimental work on investigating temperature and pressure effects on elastic properties of the PMN-PT material. The discussion will take the reader through the results of high temperature and pressure RUS experiments intended to explore the temperature-pressure region shown in Figure 1.3.

Chapter 5 will provide the reader with some background information on several high

temperature thermoelectric materials and the experimental details of their elastic constant measurements at elevated temperatures. The chapter starts with a general introduction to thermoelectric materials and devices, and the following three sections are dedicated to the three thermoelectric materials: lead tellurides, lanthanide filled skutterudites and rear earth doped strontium titanate.

A pulse-echo study of understanding thermal aging effects on temperature-dependent elastic properties of SnAgCu lead-free solder alloy is briefly discussed in Appendix A. This short chapter supplements the dissertation by discussing other ultrasonic technique to measure the mechanical properties of materials.

Appendix B lists the elastic tensor of different crystal classes and symmetries in Voigt notation.

Appendices C and D contain data acquisition and analysis software codes developed as a part of the dissertation project, and other software used with courtesy of their authors.

CHAPTER 2

RESONANT ULTRASOUND SPECTROSCOPY METHODOLOGIES

Our civilization is currently facing a number of challenges including energy crisis, natural disasters, and critical issues in health and safety. The expected contribution of materials research in this situation is to keep progressing current scientific capabilities to discover new methods and develop materials to address those issues to help our survival as well as to uplift the living conditions of everyone. Understanding the physical properties of materials under different environmental conditions will play an important role in this process. This chapter discuss Resonance Ultrasound Spectroscopy methodologies as an elegant material characterization technique for cutting edge research and development.

2.1 Theory of Elasticity and Elastic Properties of Materials

2.1.1 Stress Strain Relation

A solid material is subjected to a deformation when an external load is applied. The scale of this deformation depends on the configuration of the load, geometry of the object and elastic properties of the material. This deformation also creates an internally distributed stress field, resulting in a change in potential energy of the solid material. Materials within their elastic limits with the applied load can return to their original shape when the load is removed. Elastic constants quantify the response of a material to elastic deformation when a stress load is applied.

Let $u = \psi_x$ and $v = \psi_y$ be the displacement of the 2D object shown in Figure 2.2. The total displacement can be expressed as $\psi = \psi_x + \psi_y$, and the deformation can be expressed as δu and δv. For an elastic solid, particle displacement can be expressed as $\psi = \psi_x + \psi_y + \psi_z$.

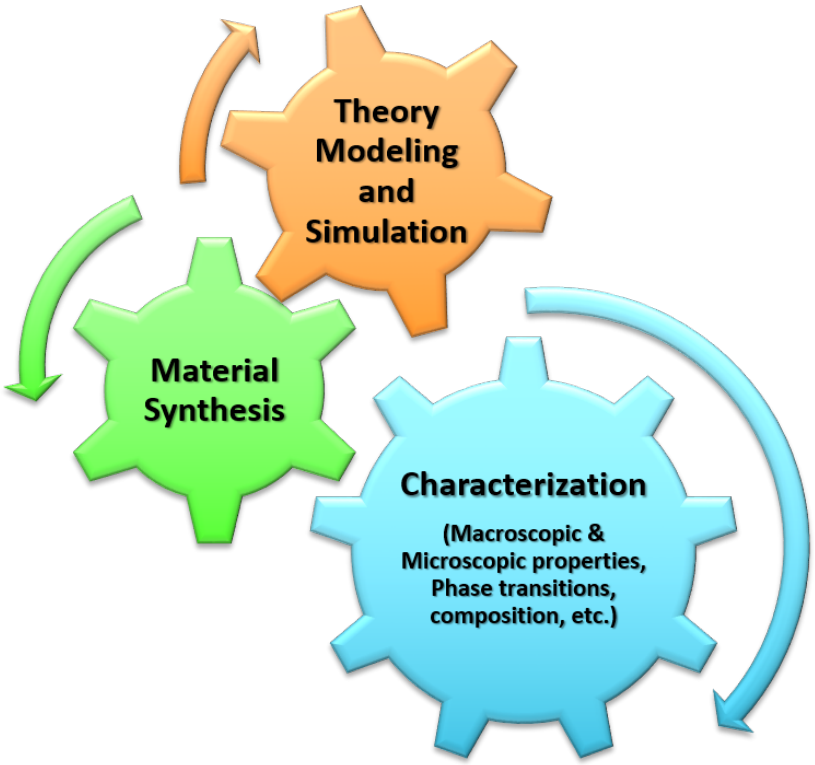

Figure 2.1. Role of RUS in the materials research as a characterization technique.

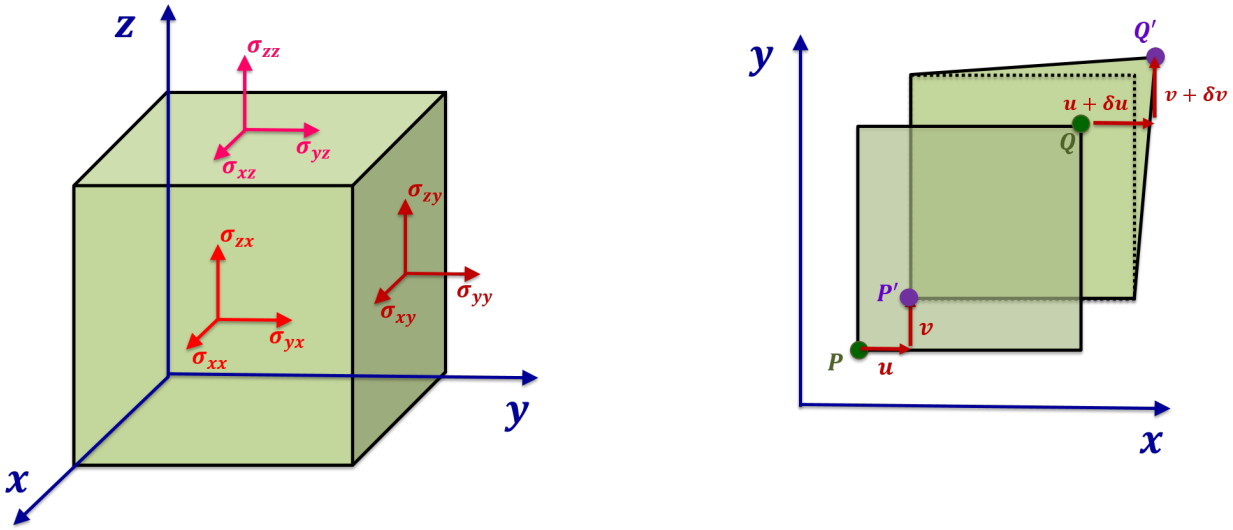

Figure 2.2. Stress components acting on surfaces of a cube and 2D illustration of strain ($\epsilon_{xx} = \frac{\partial u}{\partial x}, \epsilon_{yy} = \frac{\partial v}{\partial y}, \ldots$ [103].)

For small deformations, the strain tensor is defined as an extension in the direction i by a force in the direction j :

$$\epsilon_{ij} = \frac{1}{2}(\frac{\partial \psi_i}{\partial x_j} + \frac{\partial \psi_j}{\partial x_i}) \tag{2.1}$$

where $i, j = \{x, y, z\}$ in Cartesian coordinates. The strain tensor is symmetric such that $\epsilon_{ij} = \epsilon_{ji}$. For a small deformation in a solid, stresses (σ) can be expressed as a linear combination of strains (ϵ) coupled with the elastic constants (C_{ijkl}) given in the form of generalized Hooke's law in the linear elasticity theory of elasticity [60]

$$\sigma_{ij} = \sum_{k=1,l=1}^{3} C_{ijkl}\epsilon_{kl}. \tag{2.2}$$

The elastic constants C_{ijkl} form the elastic tensor. The indices $ijkl$ are reduced into a two-index form of ij using the Voight convention ($pq \rightarrow r : p\delta_{pq} + (9-i-j)(1-\delta_{pq})$), such that C_{ijkl} becomes C_{ij} in the reduced index form and the $3\times3\times3\times3$ elastic tensor simplifies to a 6×6 matrix. The reduction of indices can be listed as : $11 \rightarrow 1,\ 22 \rightarrow 2,\ 33 \rightarrow 3,\ 23 \rightarrow 4,\ 31 \rightarrow 5,\ 12 \rightarrow 6$). For example, $C_{1122} \rightarrow C_{12}$, and $C_{2323} \rightarrow C_{44}$.

Hooke's law can be expressed in the reduced index form $\sigma_i = C_{ij}\epsilon_j$ where i,j = {1,...,6}. That reduces the number of elements in the elastic tensor to 36, further reduced to 21 unique elements (independent elastic constants) by applying the symmetry relations where $C_{ij} = C_{ji}$. The matrix form of the Hooks law is expressed below in Voigt notation, where $\sigma_1, \sigma_2, \sigma_3$ are longitudinal stresses, $\sigma_4, \sigma_5, \sigma_6$ are transverse stresses, $\epsilon_1, \epsilon_2, \epsilon_3$ are longitudinal strains, and $\epsilon_4, \epsilon_5, \epsilon_6$ are transverse strains.

$$\begin{pmatrix} \sigma_1 \\ \sigma_2 \\ \sigma_3 \\ \sigma_4 \\ \sigma_5 \\ \sigma_6 \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ C_{21} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ C_{31} & C_{32} & C_{33} & C_{34} & C_{35} & C_{36} \\ C_{41} & C_{42} & C_{43} & C_{44} & C_{45} & C_{46} \\ C_{51} & C_{52} & C_{53} & C_{54} & C_{55} & C_{56} \\ C_{61} & C_{62} & C_{63} & C_{64} & C_{65} & C_{66} \end{pmatrix} \begin{pmatrix} \epsilon_1 \\ \epsilon_2 \\ \epsilon_3 \\ \epsilon_4 \\ \epsilon_5 \\ \epsilon_6 \end{pmatrix} \tag{2.3}$$

In material physics, elastic constants describes the interatomic force constants that are the second derivatives of the energy with respect to atomic positions. Therefore, elastic constants are a sensitive probe into the atomic environment and lattice dynamics. Elastic constants are often associated with many important physical quantities and material-related behaviors, such as Debye temperature [5], elastic moduli, Poisson's ratio, sound speeds, elastic anisotropy ratio [50, 100], melting temperature [26, 55], hardness [17], creep rate [41], ductility and brittleness [45, 96]. Temperature and pressure dependency of the elastic properties is very useful in studies of critical phenomena, and can be used to probe phase transitions and to confirm theoretical models [2, 48].

2.2 Elastic Anisotropy and Crystal Symmetries

Materials can be elastically characterized as isotropic or anisotropic. In an isotropic material, elastic properties are independent of direction, and the elastic tensor contains only two independent elastic constants, often reported as Youngs modulus ($E = \frac{(C_{11}-C_{12})(2C_{12}+C_{11})}{(C_{11}+C_{12})}$) and shear modulus ($G = \frac{(C_{11}-C_{12})}{2} = C_{44}$). Bulk modulus for isotropic materials can be expressed in terms of E and G as $K = \frac{EG}{3(3G-E)}$ (see Figure 2.3).

Materials exhibiting directional dependence of elastic properties are called elastically anisotropic materials. Crystalline materials, certain polycrystalline materials, composite and structured materials are some examples of anisotropic materials. Elastic properties of layered materials and some crystalline materials belonging the hexagonal crystal class are

symmetric about an axis that is normal to a plane of isotropy. These materials are classified as transversely isotropic materials, where the elastic properties are not direction dependent within the place of isotropy. Transversely isotropic materials are a subset of orthotropic materials, where the elastic properties vary in different planes of orthotropic materials or along three mutually orthogonal axes of rotational symmetry.

Acoustic wave velocities in a material are related to elastic constants $v_{mode}^{direction} = \sqrt{\frac{f(C_{ij})}{\rho}}$. In an isotropic material, the elastic moduli E and G can be determined by measurements of longitudinal wave speed (v_l) and shear wave speed (v_s). For anisotropic materials, measurements of wave speeds can be used to determine elastic constants (see Sections 2.7 and 2.3 for more details).

2.2.1 Anisotropy Ratio

The Zener anisotropic ratio (A_Z) is a dimensionless number that is used to quantify the elastic anisotropy of cubic crystals and is given by

$$A_Z = \frac{2C_{44}}{C_{11} - C12} \tag{2.4}$$

Conceptually, the value of A_Z indicates the deviation of a material from being perfectly isotropic, in which case $A_Z = 1$ [64, 91, 114]. Ledbetter et al. introduced a generic equation to quantify the anisotropy of all seven crystal classes.

$$A_L = \frac{v_2^2}{v_1^2} \tag{2.5}$$

where v_1 and v_2 are minimum and maximum shear sound wave velocities, as measured in scattering or ultrasound experiments over all propagation and polarization directions [61, 64]. It is also proved that $A_L = A_Z$ for cubic crystals. Another empirical measure of anisotropy

in cubic crystals was introduced by Chung et al. [91] :

$$A_C = \frac{G_V - G_R}{G_V + G_R}, \tag{2.6}$$

where G_V and G_R are the Voigt and Reuss bounds [39] on the shear modulus. By addressing various issues in other methods, a "Universal Elastic Anisotropy Index", A_U was introduced by Ranganathan et al., which is simply related to definitions of aggregate moduli:

$$A_U = 5\frac{G_V}{G_R} + \frac{K_V}{K_R} - 6 \tag{2.7}$$

where G and K are shear and bulk moduli, and the superscripts V and R denote Voigt and Reuss bounds on each modulus. Unlike the previous methods, $A_U = 0$ for isotropic materials [64, 91].

The Voigt approximations of aggregate elastic moduli are given by

$$G_V = \frac{1}{15}(C_{11} + C_{22} + C_{33} - (C_{12} + C_{13} + C_{23}) + 3(C_{44} + C_{55} + C_{66})) \tag{2.8}$$

$$K_V = \frac{1}{9}(C_{11} + C_{22} + C_{33} + 2(C_{12} + C_{13} + C_{23})). \tag{2.9}$$

The compliance matrix S_{ij} can be calculated from the elastic constant matrix C_{ij} using the relationship $[S_{ij}] = [C_{ij}]^{-1}$. Then the Reuss approximations of the aggregate elastic moduli is given by

$$\frac{1}{G_R} = \frac{1}{15}(4(S_{11} + S_{22} + S_{33}) + 3(S_{44} + S_{55} + S_{66}) - 4(S_{12} + S_{13} + S_{23})), \tag{2.10}$$

$$\frac{1}{K_R} = (S_{11} + S_{22} + S_{33}) + 2(S_{12} + S_{13} + S_{23}). \tag{2.11}$$

The Reuss approximations set a lower bound and the Voigt approximations set a upper bound to the aggregate elastic moduli. By taking the simple average between these

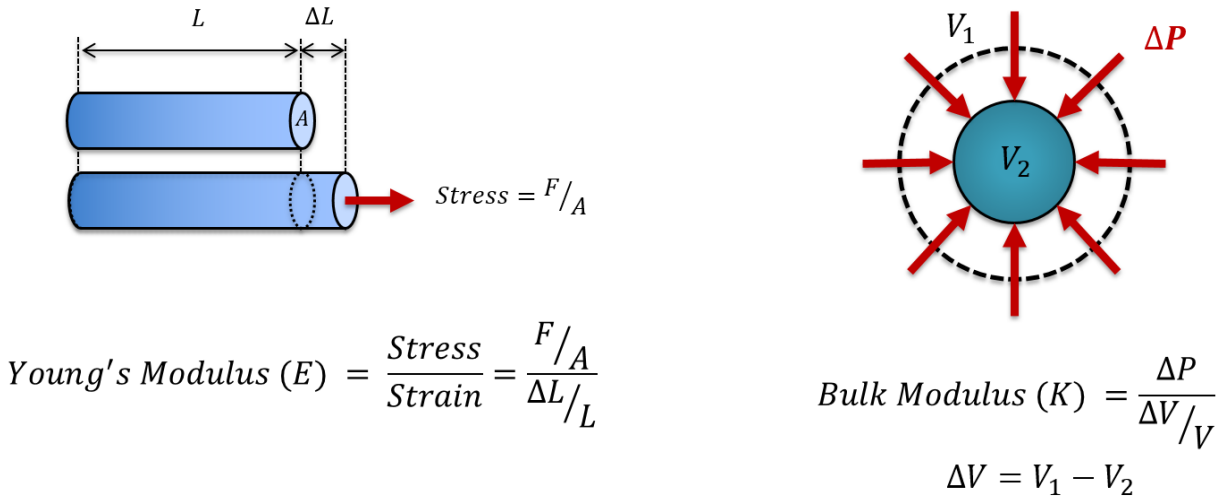

Figure 2.3. Young's Modulus and Bulk Modulus.

two bounds, Hill's approximation gives the best theoretical polycrystalline elastic modulus [18].

$$G_H = \frac{G_V + G_R}{2} \tag{2.12}$$

$$K_H = \frac{K_V + K_R}{2} \tag{2.13}$$

2.2.2 Crystal Classes

Single crystal materials are classified into seven different crystal classes, and their 32 point groups, which are listed in Table 2.1. The elastic tensor of each of these crystal classes contains different numbers of independent elastic constants based on their crystal symmetry described in Appendix B and also summarized in Table 2.1. In ultrasonic measurements, polycrystalline materials can be treated as isotropic when their grain size is smaller than the acoustic wavelength of the frequency range being examined. The other factor to consider in isotropy is that "having more than 1000 grains in the sample is a reasonable approximation for a particular polycrystalline sample" [54].

Table 2.1. List of independent elastic constants in the elastic tensor for seven crystal systems and their 32 point groups in three dimensions [66, 67, 103]. See appendix B for more details.

Crystal class	Point Group(s)	Number of C_{ij}s	List of C_{ij}s
Triclinic	$\frac{1}{1}$	21	All 21 constants $(C_{11}, ..., C_{66})$
Monoclinic	$2, m, \frac{2}{m}$	13	$C_{11}, C_{12}, C_{13}, C_{15}, C_{22}, C_{23}, C_{25}, C_{33}, C_{35}, C_{44}, C_{46}, C_{55}, C_{66}$
Orthorhombic	$222, mm2, \frac{2}{m}, \frac{2}{m}, \frac{2}{m}$	9	$C_{11}, C_{12}, C_{13}, C_{22}, C_{23}, C_{33}, C_{44}, C_{55}, C_{66}$
Trigonal	$3, \bar{3}, 32,$ $3m, \bar{3}\frac{2}{m}$	7 6	$C_{11}, C_{12}, C_{13}, C_{14}, C_{25}, C_{33}, C_{44}$ $C_{11}, C_{12}, C_{13}, C_{14}, C_{33}, C_{44}$
Tetragonal	$4, \bar{4}, \frac{4}{m}, 422,$ $4mm, \bar{4}2m, \frac{4}{m}\frac{2}{m}\frac{2}{m}$	7 6	$C_{11}, C_{12}, C_{13}, C_{16}, C_{33}, C_{44}, C_{66}$ $C_{11}, C_{12}, C_{13}, C_{33}, C_{44}, C_{66}$
Hexagonal	$6, \bar{6}, \frac{6}{m}, 622, 6mm, \bar{6}m2, \frac{6}{m}\frac{2}{m}\frac{2}{m}$	5	$C_{11}, C_{12}, C_{14}, C_{33}, C_{44}$
Cubic	$23, \frac{2}{m}\bar{3}, 432, \bar{4}3m, \frac{4}{m}\bar{3}\frac{2}{m}$	3	C_{11}, C_{12}, C_{44}
Isotropic		2	C_{11}, C_{12} or C_{44}

2.3 Elastic Constants and Speed of Sound

Speed of sound is a material property that is associated with elastic constants and mass distribution of particles in the material. In anisotropic materials, speed of sound is also a direction dependent property, and also different wave modes (e.g. longitudinal and shear) propagates in different speeds. Formulation of the speed of sound relationships begins with the wave equation for plane waves ($\psi = \psi_0 \exp i(\vec{k} \cdot \vec{r} - \omega t)$), is given as :

$$f_i = \rho \frac{\partial^2 \psi_i}{\partial t^2} - C_{ijkl} \frac{\partial^2 \psi_k}{\partial x_j \partial x_l} \tag{2.14}$$

The Christoffel equation for plane waves is given as

$$[\Gamma_{ik} - \rho v^2 \delta_{ik}]\psi_k = 0, \tag{2.15}$$

where

$$\Gamma_{ik} = C_{ijkl} n_j n_k,$$

$\vec{k} = (k_1, k_2, k_3)$, $\vec{n} = \vec{k}/|\vec{k}| = (n_1, n_2, n_3), n^2 = n_1^2 + n_2^2 + n_3^2 = 1$, v is velocity, and $\vec{\psi_k} = (\psi_{k1}, \psi_{k2}, \psi_{k3})$ is polarization. This is an eigenproblem having $\vec{\psi_k}$ as the eigenvector and ρv^2 as the eigenvalues.

$$\begin{vmatrix} \Gamma_{11} - \rho v^2 & \Gamma_{12} & \Gamma_{13} \\ \Gamma_{21} & \Gamma_{22} - \rho v^2 & \Gamma_{23} \\ \Gamma_{31} & \Gamma_{32} & \Gamma_{33} - \rho v^2 \end{vmatrix} = 0 \tag{2.16}$$

Consider the two cases of wave propagation in the z direction in materials with cubic (case 1) and rhomboidal (case 2) symmetries. For this setup $\vec{n} = (0, 0, 1)$ and $\Gamma_{ik} = C_{i3k3}$

$$\begin{vmatrix} C_{1313} - \rho v^2 & C_{1323} & C_{1333} \\ C_{2313} & C_{2323} - \rho v^2 & C_{2333} \\ C_{3313} & C_{3323} & C_{3333} - \rho v^2 \end{vmatrix} = \begin{vmatrix} C_{55} - \rho v^2 & C_{54} & C_{53} \\ C_{45} & C_{44} - \rho v^2 & C_{43} \\ C_{35} & C_{34} & C_{33} - \rho v^2 \end{vmatrix} = 0 \tag{2.17}$$

Case 1 (Cubic): Since $C_{34} = C_{43} = 0, C_{45} = C_{54} = 0, C_{35} = C_{53} = 0$, and $C_{55} = C_{44}$ for the rhomboidal 3m space group (see Appendix B),

$$\begin{vmatrix} C_{44} - \rho v^2 & 0 & 0 \\ 0 & C_{44} - \rho v^2 & 0 \\ 0 & 0 & C_{11} - \rho v^2 \end{vmatrix} = 0 \ . \tag{2.18}$$

The relationship between longitudinal and shear velocities and elastic constants is $v_3^l = \sqrt{\frac{C_{11}}{\rho}}$ and $v_3^s = \sqrt{\frac{C_{44}}{\rho}}$

Case 2 (Rhomboidal): Since $C_{34} = C_{43} = 0, C_{45} = C_{54} = 0, C_{35} = C_{53} = 0$, and $C_{55} = C_{44}$ for the rhomboidal 3m space group (see Appendix B),

$$\begin{vmatrix} C_{44}-\rho v^2 & 0 & 0 \\ 0 & C_{44}-\rho v^2 & 0 \\ 0 & 0 & C_{33}-\rho v^2 \end{vmatrix} = 0 \ . \tag{2.19}$$

The relationship between longitudinal and shear velocities and elastic constants is $v_3^l = \sqrt{\frac{C_{33}}{\rho}}$ and $v_3^s = \sqrt{\frac{C_{44}}{\rho}}$

The calculations following the Eq. 2.15 equation can be applied to measure elastic properties of single crystal materials using the pulse-echo technique, where the speed of sound (longitudinal and shear) along different directions can be easily obtained [51, 74, 75, 103]. If the material is isotropic, the speed of sound values follow simple relationships, as shown in Figure 2.10.

2.4 Mean speed of sound and Debye temperature

The Debye temperature (Θ_D) defines a border between quantum-mechanical and classical behavior of phonons. Θ_D is defined as

$$\Theta_D = \frac{h\nu_D}{k} \tag{2.20}$$

where $\nu_D = (\frac{3N}{4\pi V})^{\frac{1}{3}} v_m$ is the Debye frequency, h is Planck's constant, k is Boltzmann's constant, v_m is mean speed of sound, and $\frac{N}{V}$ is number of atoms per unit volume. The Debye temperature of an material is an important physical parameter [51]. The constant volume heat capacity of a solid can be written in terms of Θ_D :

$$C_v = 9nk_B(\frac{T}{\Theta_D})^3 \int_0^{\frac{\Theta_D}{T}} \frac{x^4 \exp(4) dx}{(\exp(4)-1)^2}. \tag{2.21}$$

Θ_D depends on the dispersion law of the acoustic phonon and has a linear relationship with the averaged sound velocity (v_m). A method to calculate Θ_D that uses the elastic

constants and some other material properties was derived by O.L. Anderson in 1963 [5]. In his method, longitudinal ($v_l = \sqrt{\frac{C_{11}}{\rho}}$) and transverse ($v_s = \sqrt{\frac{C_{44}}{\rho}}$) sound velocities are derived from measured elastic moduli C_{11} and C_{44} and are used to estimate the mean acoustic velocity: $v_m = [\frac{1}{3}(\frac{2}{v_s^3} + \frac{1}{v_l^3})]^{-\frac{1}{3}}$. Then, $\Theta_D = \frac{h}{k_B}(\frac{3nN_A\rho}{4\pi M})^{\frac{1}{3}} v_m$, where h is Plank's constant, N_A is Avogadro's number, k_B is the Boltzmann Constant, ρ is the density, n is number of atoms in the molecule, and M is the molecular weight [5].

2.5 Theoretical Background of RUS

The Rayleigh-Ritz approach predicts the normal mode frequencies of a vibrating elastic solid. For a vibrating elastic solid, potential energy density (U) is given as

$$U = \frac{1}{2} C_{ijkl} \epsilon_{ij} \epsilon_{kl} = \frac{1}{2} \sum_{ijkl} C_{ijkl} \frac{\partial \psi_i}{\partial x_j} \frac{\partial \psi_k}{\partial x_l}, \tag{2.22}$$

and the kinetic energy density for a normal mode is

$$T = \frac{1}{2} \rho \omega^2 \vec{\psi} \cdot \vec{\psi}. \tag{2.23}$$

Applying the energy conservation analogy of a simple oscillator to a vibrating elastic solid, the maximum kinetic energy (zero strain and maximum particle velocity) and maximum potential energy (maximum strain and zero particle velocity) are equal. Rayleigh's principle for an elastic solid [93] allows the estimation of normal modes of a vibrating elastic solid. Normal mode frequencies are computed using the equation below; where $\vec{\psi}$ describes the displacement field, x_i are the dimensions, ρ is the density, and C_{ijkl} are the components of the elastic stiffness matrix.

$$\omega^2 = \frac{\iiint C_{ijkl} \frac{\partial \psi_i}{\partial x_j} \frac{\partial \psi_k}{\partial x_l} dV}{\iiint \rho \psi \cdot \psi dV} \tag{2.24}$$

Solving for the normal modes of a solid object is a straightforward calculation when

the mass density, sample dimensions and the elastic constants are given, but the inverse approach of finding elastic constants from given normal modes is a theoretically challenging approach. However, solving the inverse problem can be achieved indirectly by solving the forward problem to calculate frequency eigenvalues from initially guessed elastic constants processed through a minimization algorithm that compares calculated normal mode frequencies to experimentally measured resonance frequencies of a vibrating body.

The recipe to find the solution to this problem assumes free boundary conditions and involves minimizing the Lagrangian listed below. The boundary value problem has eigenvalues $\omega_n = 2\pi f_n$ yielding resonance frequencies of the solid body.

Newton's Law applied to a volume element of vibrating solid is given as $C_{ijkl}\frac{\partial^2\psi_k}{\partial x_j \partial x_l} = \rho\frac{\partial^2\psi_i}{\partial t^2}$. Assuming the $\exp(i\omega t)$ time dependence the equation of motion becomes

$$\rho\omega^2\psi_i + C_{ijkl}\frac{\partial^2\psi_k}{\partial x_j \partial x_l} = 0. \tag{2.25}$$

Assuming stress free boundary conditions, $\sigma_{ij}n_j = C_{ijkl}\frac{\partial\psi_k}{\partial x_l}n_j$, the Lagrangian equation ($L = \iiint(T-U)dV$) for the system can be formulated using the kinetic energy and potential energy densities:

$$L = \iiint(\rho\omega^2\psi_i\psi_i - C_{ijkl}\frac{\partial\psi_i}{\partial x_j}\frac{\partial\psi_k}{\partial x_l}dV). \tag{2.26}$$

To apply the Rayleigh-Ritz method, a complete set of basis functions is used to approximate the displacement field $\psi_i = \sum_{p=1}^{N} a_{pi}\phi_p$. Various orthogonal functions, such as Legendre polynomials, trigonometric functions and many other mathematical functions, have been exploited as basis functions by Holland [43], Demarest [22], and Ohno [84]. In 1991, Visscher et al. introduced a basis function based on simple powers of Cartesian coordinates that enables an efficient computational solution and is applicable to many different shapes

[4, 38, 66, 102]. The new basis set is

$$\phi_p = \left(\frac{x}{a}\right)^l + \left(\frac{y}{b}\right)^m + \left(\frac{z}{c}\right)^n. \tag{2.27}$$

where, $l + m + n \leq N$, and a, b, c are half the edge lengths of the sample. Larger N, makes the higher agreement between the actual displacement field to the approximation by the basis function. However, the computational scale of the solution algorithm has to be taken into account when choosing a value for N because the larger the N, more computational power is required, and some machine precision issues may also arise [38]. $N = 12$ has been considered in order to be the optimum N after a comparison between approximate solution and the analytical (exact) solution with less than 0.0005 %. [38, 102].

In the new basis, the potential energy density and the kinetic energy density becomes $E_{piqj} = \delta_{ij} \iiint \rho\phi_p\phi_q dV$ and $\Gamma_{piqj} = \iiint C_{ijkl}\frac{\partial\phi_p}{\partial x_k}\frac{\partial\phi_q}{\partial x_l}dV$, and the Lagrangian is

$$L = (\omega^2 E_{piqj} - \Gamma_{piqj})a_{pi}a_{qj}. \tag{2.28}$$

The integrals can now be evaluated analytically with fixed limits (-1, 1) by normalization [102, 110]. Applying the scheme of reduced indices, such that $pi \to \mu$ and $qj \to \nu$, minimizing the Lagrangian with respect to a_μ yields the $3N \times 3N$ matrix eigenvalue problem which typically becomes $\approx 1000 \times 1000$ in RUS [38, 66].

$$\Gamma_{\mu\nu} - \omega^2 E_{\mu\nu}a_\nu = 0. \tag{2.29}$$

This approach allows one to obtain a set of natural frequencies (ω^2) and the corresponding displacement fields (eigenvectors) of an elastic solid by solving a standard eigenproblem.

Volume integrals of the chosen basis have the form of $F(p, q, r) = \iiint (x^p y^q x^r) dxdydz$.

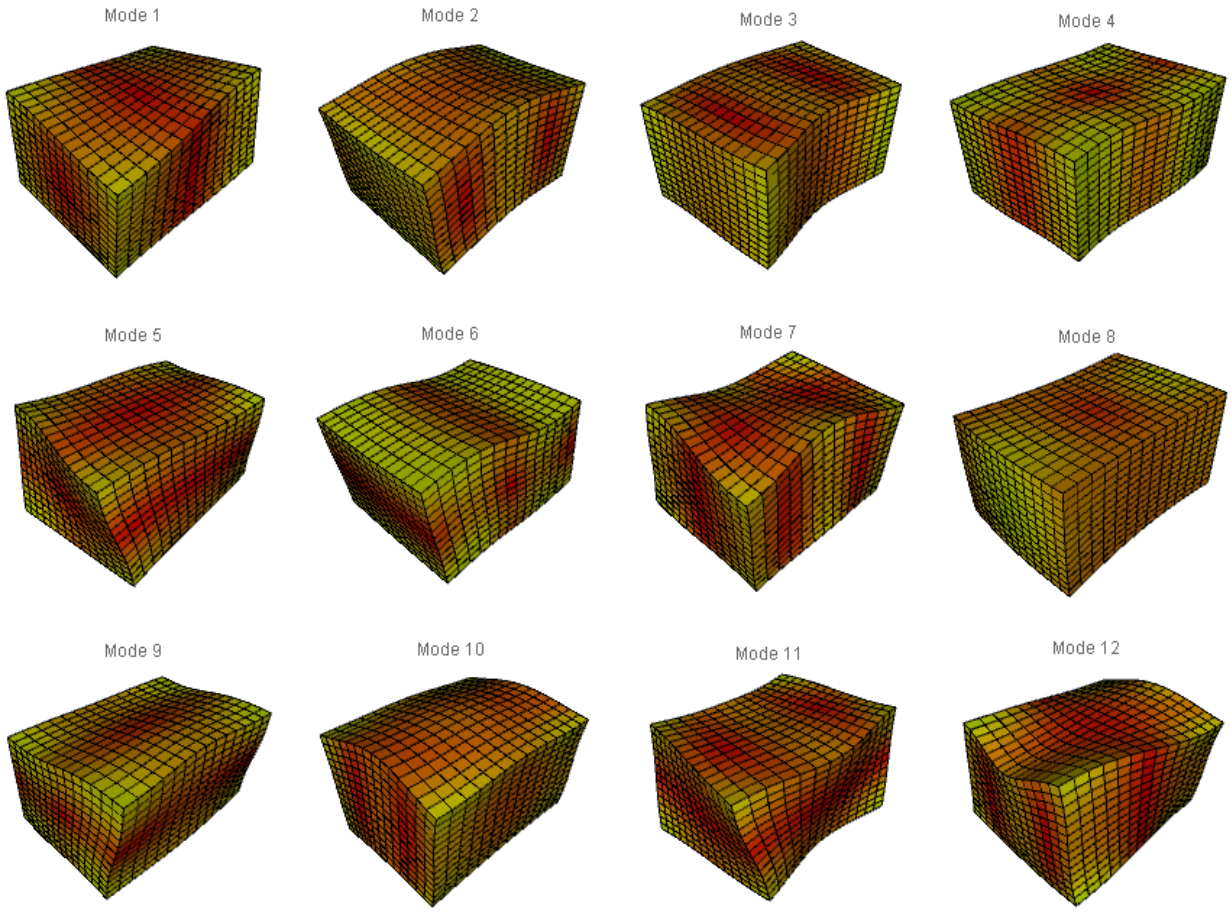

Figure 2.4. Visual plot of vibration modes of RPP solid object

For a rectangular parallelepiped (RPP) centered at the origin, the integral becomes,

$$F(p,q,r) = \begin{cases} \frac{8}{(p+1)(q+1)(r+1)} & p,\ q,\ r\ are\ even \\ 0 & otherwise \end{cases} \tag{2.30}$$

2.5.1 Sample Preparation

After obtaining the material sample for an RUS study, the first task is to prepare sample specimens with a specific geometrical shape with precise measurement of its dimensions and density (see Figure 2.6). RUS theory has been developed to analyze standard geometries such as spheres, cylinders and cubes. However, the most common form to use is

parallelepipeds. It is very important to have sample preparation instruments with precision maneuverability. A high precision diamond wheel saw, and a lapping and polishing machine is to be utilized in this process.

For single crystal materials with lower symmetries, crystal class and orientation also have to be considered in the analysis. Therefore, alignments of crystallographic directions with the sample axes need to be determined during preparation of test specimens. Non-destructive methods such as X-ray diffraction (XRD) can be used to determine crystallographic direction. Any offset with crystallography directions to the sample alignment (Euler angles) can be input as a parameter in the computation.

Prepared samples then need to be cleaned to remove any surface residue from materials exposed during sample preparation and dried well to evaporate any solvent that might absorb into the specimen. An ultrasonic cleaner and annealing in a vacuum is used as necessary in the cleaning process. A micrometer and high precision mass scale are then used to measure sample dimensions and mass to calculate the density. The calculated density may compared with the previously reported density values or with other samples in the study to identify the existence of voids or other internal defects in the sample specimen. Handling the sample specimen after preparation step should be delicately done to avoid surface and corner damages, which could result in errors in the resonance measurements.

2.6 RUS as an Experimental Technique

2.6.1 Instrumentation

In the RUS experimental setup, a sample is placed between two acoustic transducers, one driven with a swept sinusoid and the other connected to a lock-in amplifier that monitors the vibrational response of the sample (Figure 2.7). When the drive frequency matches a natural frequency of the sample, the vibrations are amplified approximately by the quality factor of the resonance ($Q \simeq \frac{f_0}{\Delta f}$), resulting in a peak in the mechanical resonance spectrum (MRS), as shown in the Figure 2.9. In a typical RUS measurement the first $20 - 30$ reso-

nances are recorded, and a Lorentzian line shape fit is used to determine the center frequency (f_0) and quality factor (Q) of each peak. An ordered list of these center frequencies then constitutes the list of natural frequencies of the sample. Then the elastic moduli will be estimated from that least-square fitting algorithm in which the measured spectrum is compared to a computed spectrum adjusting the elastic moduli as a variable parameter until a best fit is obtained. Formulation of these algorithms are discussed in detail in the latter sections of this chapter. More details on construction, parts and materials of high temperature and high pressure RUS systems can be found in Li [66, 67], and Adebisi [2].

Prepared samples then will be moved to room temperature resonance measurement setup. The problem of which frequency range to be scanned needs to be addressed as the next step. With a thorough literature survey about the specific material or its close variations, the elastic constants can be estimated for the material specimen being studied. Solving the forward problem using a computational algorithm that takes estimated elastic constants and measured sample dimensions and density will produce a list of frequencies from fundamental to several higher order modes. A highly accurate elastic constant can be obtained from RUS study with consideration of 20–25 resonance modes. A higher number of resonance modes needs to be considered for materials with lower symmetries where a higher number of independent elastic constants are present in the elastic tensor. The scan frequency range may include 20 – 25 modes from the calculated list of frequencies, or more if necessary. However, this needs to be verified and may need to follow a back-and-forth process until frequency range is determined with confidence. In the case of novel materials, elastic constants may not be available in the literature, and forward calculation results may not be very reliable. In this case, we may need to start with an educated guess of elastic constants followed by trial scans over a broad frequency range to determine the scan frequency range. Frequency range is not the only parameter known to get optimal results out of the instrumentation. Frequency resolution (step size) and amplitude resolution (sensitivity) also need to be optimized based on observed peaks during initial scans.

After fixing the frequency range to be scanned, a room temperature RUS experiment will be performed to obtain a list of resonant modes. Elastic constants will be derived by a computational algorithm specific to the crystal symmetry of the material, solving the backward problem by taking the list of center frequencies and sample dimensions and density as the input parameters. This nonlinear fitting computer algorithm also needs to be provided with initial guess values for elastic constants, which is similar to solving the forward calculation problem, but this non-linear fitting algorithm will optimize elastic constant values based on experimentally measured resonant frequencies specific to the material specimen being studied. These values can again be compared with reported values for the same or similar materials to validate results.

After obtaining reasonable room temperature results for the material specimen, the investigation can be carried out by varying other parameters such as temperature, pressure, and electromagnetic fields, which can affect the physical state of the material specimen. Natural frequencies, elastic constants, and acoustic attenuation can be interpreted as functions of physical parameters such as temperature and pressure. An extensive study of RUS can lead to an understanding of fundamental thermodynamic relations and probe phase transitions. These results are very important in both fundamental research and industrial applications where properties of novel materials are crucial facts to be known.

Varying temperature as a parameter in RUS is an established technique in the material science community in order to probe temperature dependent properties of materials. Many different authors have reported their work using the RUS technique. Necessary heating and cooling apparatus were developed in respective studies to perform RUS in a wide range of temperatures, both above and below room temperatures with respect to room temperature. Thermal expansion of the material must be considered in fitting the elastic constants since a change of dimensions of the material specimen results in a change in the density. Mode frequencies, acoustic attenuation, and elastic constants can be interpreted and analyzed as a function of temperature. These results can then be used to probe temperature induced phase

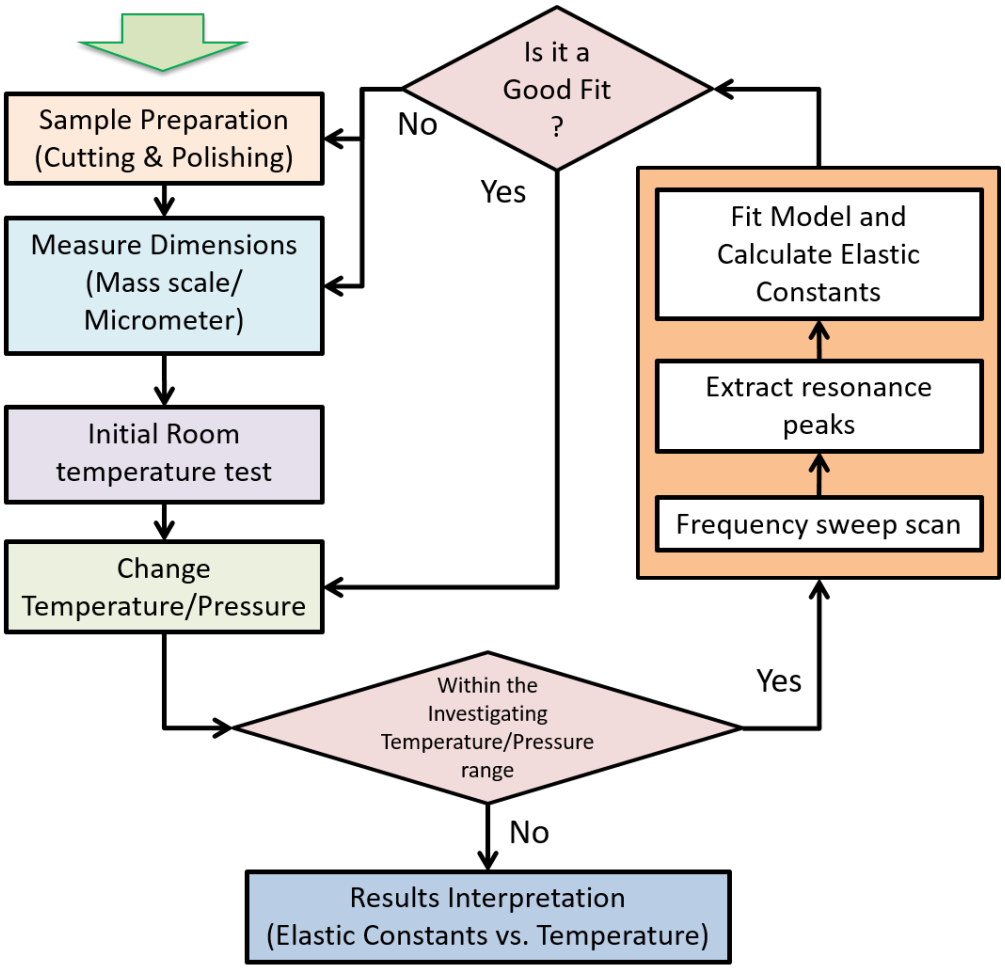

Figure 2.5. RUS experimental procedure

transitions and to construct phase diagrams. Including both temperature and pressure as variable parameters, one can explore temperature pressure phase diagrams using the RUS technique [106].

2.6.2 Sample Mounting Schemes and Loading Effects

The RUS technique has the advantage of eliminating the hard bonding between sample and the transducer that is required in most other acoustic methods. Also the transducers do not need to cover large areas of the sample surface in both excitation and response detection. Two major sample mounting schemes are used: corner-mounting, which is the traditional and most effective mounting scheme, and face-mounting, when the corner mounting

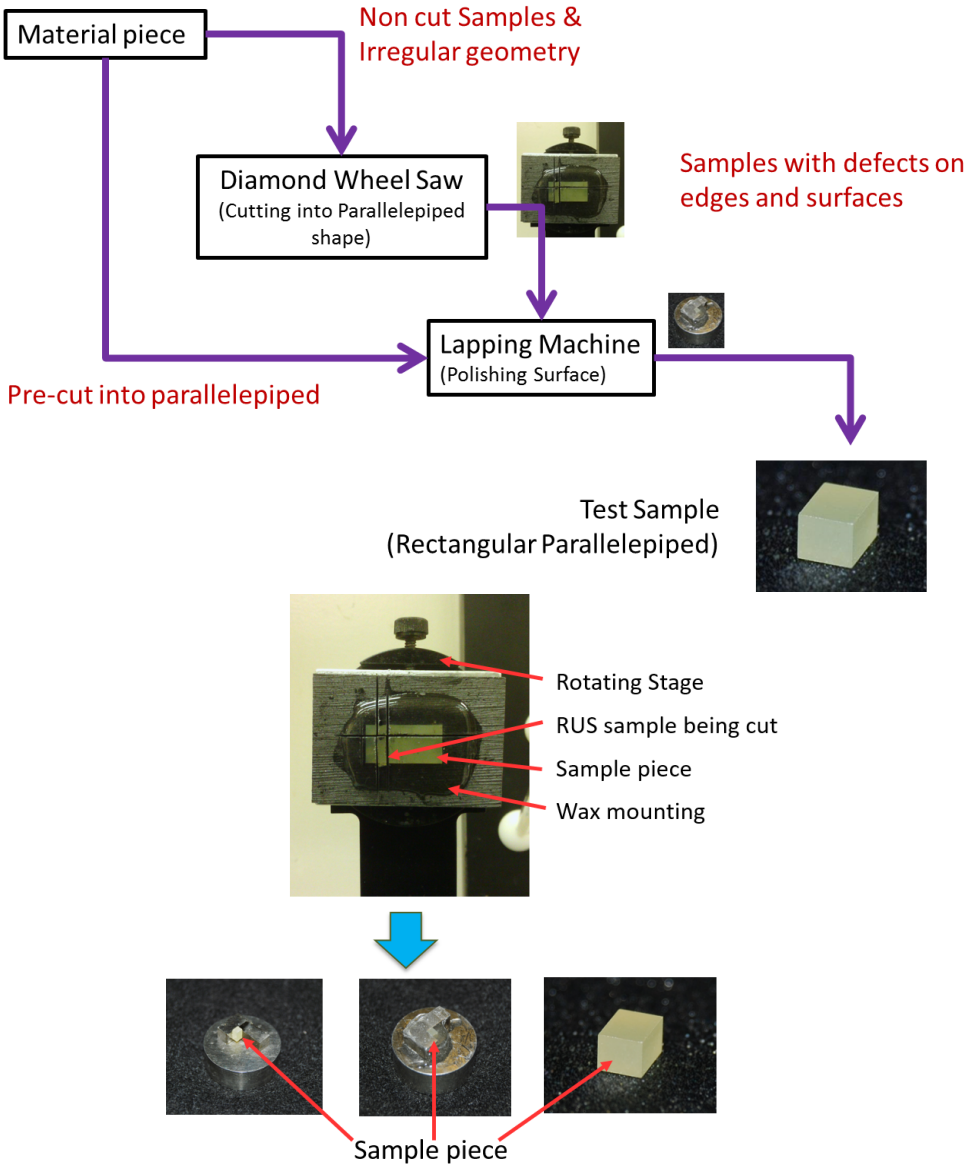

Figure 2.6. Sample preparation for RUS experiment

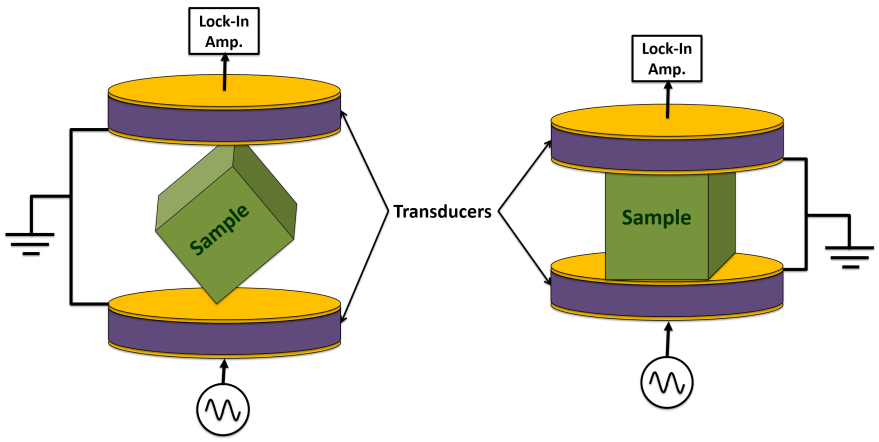

Figure 2.7. Sample mounting configurations: corner-mounting and face-mounting

is not preferred with the setup (see Figure 2.7). The visual plot of the surface displacement of a vibrating solid shows that the corners are the most active regions at resonances. That makes the corner-mounting the most effective method capturing resonance modes. However, the face-mounting scheme works very well in creating less cautions about the sample being displaced or dropped from the transducers during a high temperature and high pressure measurements.

2.6.3 Computer Interfacing and Data Analysis

High temperature and high pressure RUS experimental procedure is shown in Figure 2.5. The procedure involves a number of computational tasks in both data acquisition and analysis. A computer, installed with instrument communication interface (e.g. USB, GPIB) compatible with the data acquisition and frequency generator unit(s) is required in the data acquisition step. The software requirement for this computer system will be standard Windows or Linux operating system. The Python programming language is free and open source, and also has a good knowledge base and a good reputation within the scientific computing community. Python supports many cutting edge laboratory instruments through their custom application programming interface (API) or Virtual Instrument Software Architecture (VISA). Computer algorithms allow us to automate some steps through instrument control and data acquisition to obtain mechanical resonance spectrum (MRS) of the sample. After

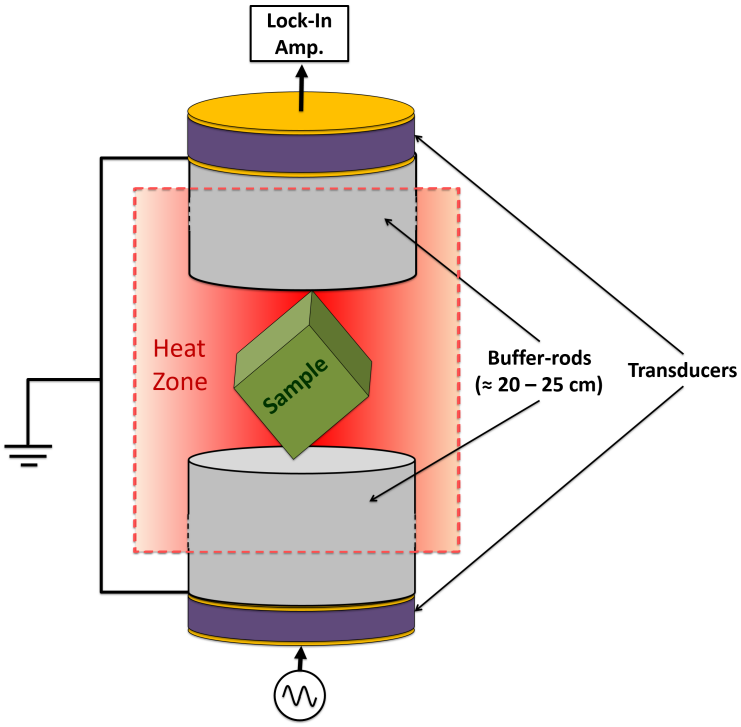

Figure 2.8. Buffer-rod transducer arrangement keeps the transducers away from the heating zone.

recording the MRS of the sample, the next step would be to analyze the spectrum to extract frequencies and the quality factor of each resonance. In a typical RUS experiment, 21 such resonances are considered, and for materials with lower symmetries, an increased number of resonances needs to be considered to increase the reliability of the study. This step is not computationally intensive and a regular PC can be utilized in this task.

The final step of determining the elastic constants involves a minimization algorithm that compares experimentally measured resonance frequencies of the normal modes calculated by Rayleigh-Ritz method described earlier in this chapter. Running this minimization algorithm can become computationally intensive depending on the crystal symmetry with the increased number of variable parameters included in the algorithm. This step requires a PC with higher processing power and memory capacity. All the data acquisition and analysis tasks in an RUS experimentation can be integrated into a single PC with optimal computing power for data analysis.

2.6.4 Extracting Data from Resonances

The Lorentzian function is expressed in the form

$$A(f) = A_0 \left(\frac{\frac{f}{f_0}\cos(\phi) + (1 - (\frac{f}{f_0})^2)Q\sin(\phi)}{(\frac{f}{f_0})^2 + (1 - \frac{f}{f_0}^2)^2 Q^2} \right), \tag{2.31}$$

where f_0 is the center frequency (resonance frequency), Q is the Quality Factor, ϕ is the phase angle A_0, and is the peak amplitude. However, in an actual measurement environment resonance peaks are associated with background noise coming from electronics, electrical connections, vibration responses of system components detected by transducers. Assuming the total background noise can be modeled to a higher order polynomial, the fitting function becomes

$$A(f) = A_0 \left(\frac{\frac{f}{f_0}\cos(\phi) + (1 - (\frac{f}{f_0})^2)Q\sin(\phi)}{(\frac{f}{f_0})^2 + (1 - \frac{f}{f_0}^2)^2 Q^2} \right) + (A_1 + A_2 f + A_3 {}_f^2 + A_4 f^3) \tag{2.32}$$

The peaks can be analyzed as adjacent groups as shown in Figure 2.9 or as isolated peaks.

Energy in a decaying wave can be given as $E(t) = E_0 \exp -\alpha\prime t$ where, $E(t)$ is the energy at time t, E_0 is the initial energy, and α' is the attenuation per unit time. Attenuation per unit length is given as $\alpha = \frac{\alpha'}{v}$ [103]. The quality factor (Q) represents the rate of energy loss in resonator vibrating at frequency f_0 is defined as $Q = \frac{f_0}{f_2 - f_1}$, where f_1 and f_2 are the two frequencies bounding full width at half maximum (FWHM) of the resonance curve. This portion of the curve also represents one half of the average power of the resonance [103]. Q has an inverse relationship with the acoustic attenuation (α), which is also an important material property we can extract from the resonance measurements in an RUS measurement. For individual resonances, attenuation can be given in terms of resonance frequency f_0, speed of sound v, and Q

$$\alpha = \frac{2\pi f_0}{Qv}. \tag{2.33}$$

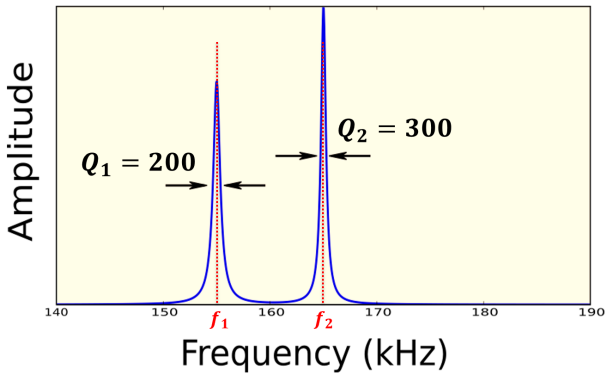

Figure 2.9. Extracting data from resonances.

Investigating inverse quality factor ($\sim Q^{-1}$) over a temperature and pressure range can be used to study loss mechanisms in solids. Leisure et al. used RUS to study the internal friction due to hydrogen motion in hydrogen-metal materials by analyzing the quality factor of resonances [62]. Migliori in his book on RUS [4] discussed the differences of Q measurements between RUS and pulse-echo: an acoustic measurement technique which will be discussed later in this chapter. In RUS, Q is a measure of vibrational energy loss from the sample per cycle, where scattering and geometry also cause the energy loss in the pulse-echo measurements [4]. Mounting configuration and loading on the sample from transducers as well as from the atmospheric pressure also affects on the Q value of resonances in RUS measurements.

2.6.5 Estimation of Elastic Moduli

The problem of solving the eigenproblem, given sample geometry, density, and following the Rayleigh-Ritz method in order to find natural frequencies of a solid object was covered in section 2.5. We call this method the forward problem. However, an analytical method to solve the inverse of that forward problem is not yet available due to mathematical difficulties of attaining a solution [38, 66]. The experimental method of estimating the elastic moduli using the list of experimentally measured natural frequencies involves solving the forward problem coupled to a minimization algorithm. The minimization algorithm varies the

parameters of the forward problem to minimize the difference between experimentally measured resonance frequencies with the computed natural frequencies. The problem is treated as a nonlinear least-squares problem, and a version of Levenberg-Marquart algorithm is used in the computational solver [4]. The analysis process requires initial values for elastic moduli as input parameters to start the algorithm. These initial parameters should come from a scientific judgment based on a good literature review or from a theoretical calculation. The agreement of measured frequencies (g_i) and computed frequencies ($f_i = \frac{\omega_i}{2\pi}$) is evaluated by a measure of fractional division also considered for the goodness of the fit or figure of merit in the elastic moduli estimation. $F = \sum_{i=1}^{N} w_i(f_i - g_i)^2$. An RMS value of the error between $0.1\% - 0.3\%$ is considered a "good fit" in typical RUS measurements. However, this error can be increased with the temperature due to broadening in peaks that increase the uncertainty of determining the peak as well as the thermal expansion effects on the parameters such as geometry and density. More details of this algorithm can be found in Migliori [4, 76] and Spoor [102].

2.6.6 Sources of Errors in RUS measurements

Errors in an RUS experiment can be decomposed into errors of individual measurements: center frequencies of resonances, sample dimension, sample mass/density, and temperature and pressure values recorded at each spectrum.

Temperature is a very important measurement in each RUS experiment, because the temperature value at each data point must be taken into account to calculate thermal expansion of the material sample as well as to interpret results (elastic properties, attenuation, etc.). A conventional thermocouple is used in the high temperature RUS setup and a Resistance Temperature Detector (RTD) is used in the high temperature and high pressure setup. Errors in the temperature readings can be identified as systematic errors in thermocouple/RTD as well as the stability issues of the temperature controller where the temperature fluctuates between the set-point during a single spectrum scan. The standard

error range of the thermocouple/RTD is ±0.01 C and the thermocouple errors can also be caused by temperature changes in the reference temperature (0 C).

Errors in sample preparation can also be a cause of error in measurements. Presence of internal voids and cracks, absorption of solvents in the sample preparation, residues of foreign substances on the sample surface, or imperfect sample geometry can lead to higher errors, especially for sub-millimeter scale samples. However, such errors can be easily detected in the very first step of the measurement by the absence of a "good" elastic constant fit. In that situation, going back and forth in the sample preparation and measurement steps are required until a "good fit" is obtained.

Errors in frequency values can be associated with significantly low signal to noise ratio, peaks broadening in lossy materials and the presence of extra resonance peaks in the spectrum caused by parts of the measurement setup. These issues are high in the buffer-rod system (See Figure 2.8) which may include resonances of buffer rods in the spectrum.

In the formulation of RUS theory, we assumed free boundary conditions. However, in the experimental setup, transducers will add some load to the sample surfaces, especially in face-mounting configuration. With the construction of the high temperature RUS system, Gladden et al. investigated the loading effects on the sample resonances. In that study, the first twelve resonances of a steel sample were analyzed under variable load from 5 g (transducer weight) up to 50 g, and it was found that the loading effects on the resonance frequencies are less than 0.3% at the entire loading range of 5 – 50 g.

2.7 Measuring Elastic Properties of Materials using Pulse-Echo Technique

The ultrasonic pulse-echo (PE) technique is one of the widely used methods to characterize mechanical properties of materials. In this technique, time trace of a short acoustic pulse propagating a known length through a material sample is used to calculate the sound speed of the material. A pulse emitting/receiving transducer is often directly coupled to the sample, but in some cases a buffer-rod is used to guide both sending and receiving pulses.

The experimental setup of the PE consists of pulse emitting/receiving transducer, pulse generator, and oscilloscope to visualize and record the pulse-echo signal (Figure 2.11). Material samples have to be prepared with two parallel surfaces to get an unambiguous echo signal. The propagating pulse entered to the sample through one of these surfaces, and is reflected back from the opposite surface. The reflected pulse is called the echo. The presence of voids, cracks, or any other structural defect in the material as well as the two surfaces of interest can be problematic because of possible scattering and the addition of extra echoes to the signal. Energy of the initial pulse is attenuated by the material while it is propagating and the amplitude of each echo decays exponentially with travel distance. The speed of sound is a material property that depends on the density of the material and elastic constants coupled with the mode of the propagating pulse. A longitudinal pulse can be used to determine a longitudinal modulus (C_{11}), and a shear pulse can be used to determine the shear modulus (C_{44}). The PE method is a non-destructive material characterization technique that uses the same principle as the industrial ultrasonic scanners used to locate defects and cracks in structures and pipes. More details on the technique can be found in Pantena et al. [85], McSkimin [74, 75] and the tech report from Brown [14].

Longitudinal and shear wave speeds (C_l and C_s) can be obtained by measuring travel time of a pulse between two parallel surfaces of the material specimen and the total distance of the pulse traveled (twice as the separation of parallel surfaces). Longitudinal modulus (M) and shear modulus (G) can be determined using C_l and C_s, and mass density (ρ) of the material. In an isotropic material, Young's modulus (E), Bulk modulus (K), and Poisson's ratio (ν) can be derived from M and G; Figure 2.10 shows this analysis scheme. The temperature and pressure dependence of elastic moduli can be studied by carrying out pulse echo measurements under variable temperature and pressure conditions.

Compared to RUS, PE is a straightforward technique that requires less effort in the data analysis. However, for high precision PE measurements, it is necessary to consider many details of the experimental setup and the electronics. PE transducers are attached

to the sample using a bonding agent (couplant) which makes a proper interface to ease the pulse transmission between transducer samples. This layer has to be very thin in order to add no significant travel time to the propagating pulse. Permanent couplants are often used, but temporary gel type couplants are also available. Choosing the couplant needs to be done carefully based on the application as well as the mode of the pulse generated by the transducer. Ultrasonic Gel, vacuum grease, super glue and often water is used as couplant for longitudinal pulse transmission. For shear pulse transmission, a special Shear Gel is available, and other common high viscous materials can also be used. Multiple measurements have to be taken in order to have accurate sound speeds. Different transducers have to be used to measure the speed of sound in different modes (e.g. longitudinal and shear). Transducers may have to go through mount and re-mount cycles in these measurements. Pulse energy, attenuation and damping settings of the pulse generator also have to be tuned to get a decent pulse-echo signal.

Sample thickness (the distance that the pulse travels), bandwidth and center frequency of the pulse are also important factors in making accurate and precise measurements. A material specimen is referred to as a bulk specimen when the thickness is greater than one-fifth of the wave length. When this criteria is met, more confident longitudinal velocity measurements can be obtained from the pulse-echo technique [14]. Considering longitudinal and shear waves with the same frequency, shear speed is approximately one half of the longitudinal speed and the shear wavelength one-half of the longitudinal wavelength. That leads one to choose the frequency of the shear wave PE transducer to be one half of the longitudinal transducer frequency [14]. For samples with smaller dimensions, pulse bandwidth has to be considered to have sufficient time delays between pulse and echo signals. A 10 MHz longitudinal PE transducer and a 5 MHz shear PE transducer have been used for the work presented in this dissertation.

Sources of the errors in PE technique are mainly from the measurement of transit time, material density and the pulse propagation distance.

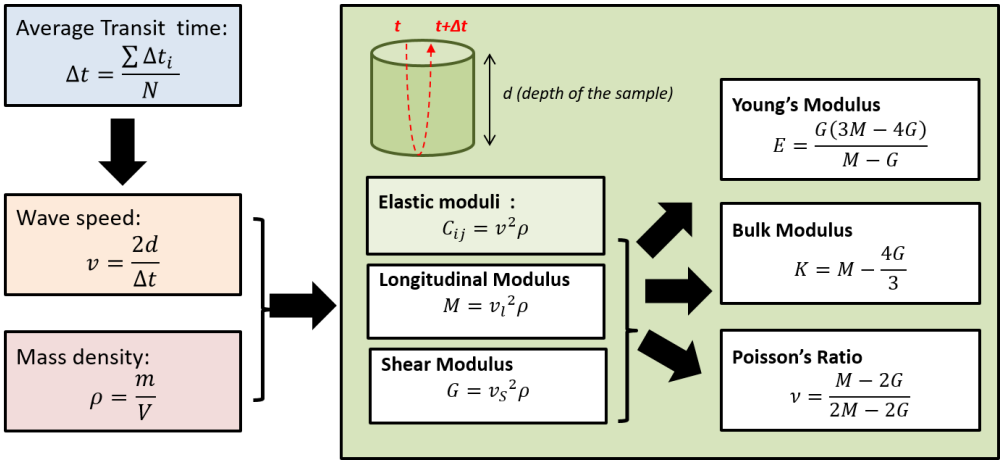

Figure 2.10. Pulse echo data analysis procedure.

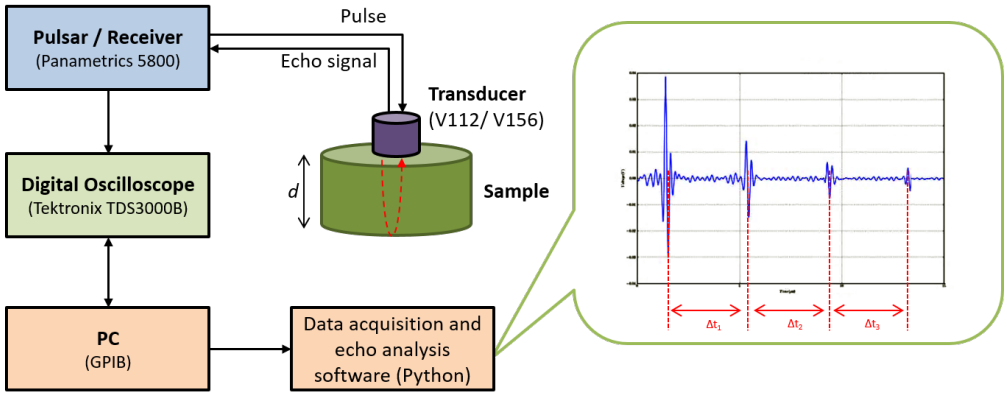

Figure 2.11. Pulse echo experimental setup.

CHAPTER 3

TEMPERATURE AND PRESSURE EFFECTS ON ELASTIC PROPERTIES OF SOLIDS

3.1 Temperature Effects on Elastic Properties of Solids

The theory of the temperature dependence of elastic properties was developed by Born et al. [12, 109] and contributed to by many others [69]. Varshni in 1970 proposed that the relationship between elastic constants and temperature can be described by semi-empirical equations after an investigation of a number of different substances [109]. The proposed equations are

$$C_{ij} = C_{ij}^{0} - \frac{s}{\exp\left(t/T\right) - 1} \tag{3.1}$$

and

$$C_{ij} = a - \frac{bT^2}{T + c} \tag{3.2}$$

where C_{ij}^{0} is the value of elastic constant at 0 K and s, t, a, b, and c are empirical constants. In the formulation of this theory, taking the Einstein model for the solid into the analysis, Varshni points out that the equation 3.1 represents the total temperature dependence of elastic constants, including the electronic contribution. Elastic constants of "normal" solids follow this trend upon heating.

3.2 Pressure Effects on Elastic Properties of Solids

The connection between elastic constants to other physical properties of materials and the importance of measuring elastic constants at elevated temperature and pressure environments were discussed in Chapter 2. These measurements are highly applicable in advanced

materials development characterization as well as to understand the material behavior at extreme environments such as deep earth.

For an example, a material placed at the center of the earth is assumed to be experiencing a pressure around 350 GPa, which is close to 3.5 million times atmospheric pressure [88]. In addition, extremely high temperature will also play a major role in the existence of the material in that environment. Most geological measurements are based on measurements of seismic velocities that are highly coupled with the elastic constants of deep earth materials [65, 88]. Elastic constants of materials under extreme environments are also crucial in understanding different physical phenomenon in deep earth through simulations and models. There are a number of ongoing research in this area from first principles studies [78], single-crystal and powder X-ray diffraction [86, 89], Mossbauer spectroscopy [89], nuclear inelastic scattering [89], and ultrasonic techniques [65, 79]. The resonance measurements can become slightly difficult under higher hydrostatic pressures due to damping and other instrumentation issues. However, the measurements systems, such as laser based non-contact RUS system [99] can be utilized to test materials at moderately high pressures.

The high temperature and high pressure RUS instrumentation at the NCPA is capable of measuring elastic properties of materials up to about 550 C (823 K) and pressures up to about 1000 psi (6.9 MPa). Measurements from the current version of high-pressure instrumentation are not fully applicable to pressure studies in earth materials research because of the lower pressure limits. However, this instrumentation is an elegant experiment setup that can be utilized for preliminarily studies on pressure effects on elastic properties of materials. The setup can also be used to conduct basic physics research on understanding temperature and pressure effects on elastic properties of materials. For example, this work is focused on mapping the T–P phase diagram of PMN-PT based on elastic constant measurements carried out by that system.

3.3 Phase Transitions

A thermodynamic system changes from one state to another as a result of a change in its free energy exchanged mainly in the form of heat, called a phase transition. The most common example is the phase transitions that water goes through as it changes from gas to liquid to solid under different temperature and pressure conditions. Phase transitions can be probed by tracking macroscopic and microscopic properties when the system is exposed to external influences such as temperature, and pressure, or variation in the chemical composition. Some examples of phase transitions that can be seen in solids are the ferromagnetic-paramagnetic phase transition in magnetic materials at the Curie point, the emergence of superconductivity in superconducting materials when cooled below a critical temperature, and structural phase transitions in crystalline materials when a material changes its crystallographic structure [20].

Phase transitions are associated with a measure of order (order parameter) that characterizes the amount of change taking place in the transition [94]. The experimental investigations on phase transitions are focused on this order parameter determined by direct measurements in the atomic scale or effects on other measurable quantities such as material properties. Measurements of dielectric constants, refractive index, specific heat, thermal conductivity, and elastic constants are widely used material properties in probing phase transitions.

Applying RUS methodologies in studying phase transitions is very effective since RUS is a technique of making precise measurements of elastic constants developed with instrumentation capability to conduct experiment in broad temperature and pressure range [63, 67, 76].

In Landau theory, phase transitions can be described in terms of the free energy of the system [60, 63, 94]

$$F = F_e + F_c + F_L + F_0 \tag{3.3}$$

The elastic energy (F_e) is given as

$$F_e = \frac{1}{2}\sum_{i,j} C_{ij}\epsilon_i\epsilon_j \ , \tag{3.4}$$

where elastic strains and elastic constants (ϵ_i and C_{ij}) are in Voigt notation [63]. The concept of elastic constants being the second derivative of the potential energy with respect to strain is briefly discussed in Chapter 1 with the interatomic potential curve. The elastic constants can be expressed as the second derivatives of F with respect to ϵ_i

$$C_{ij} = \frac{\partial^2 F}{\partial\epsilon_i\partial\epsilon_j}. \tag{3.5}$$

The coupling between strain and order parameters is represented by F_c where this coupling leads to the effects on elastic constants near the phase transition [63].

The elastic constants are usually measured under the condition that $\frac{\partial F}{\partial\eta_i} = 0$ under externally applied strain. However, the elastic constants associated with free energy from purely elastic (C^0_{ij}) are determined under the condition of $\eta_i = 0$, which excludes the effects of the phase transitions [63, 94]. The difference between these elastic constants can be given as

$$C_{ij} = C^0_{ij} + \frac{\partial^2 F_c}{\partial\epsilon_i\partial\epsilon_j} - \sum_{k,l}[\frac{\partial^2 F_c}{\partial\epsilon_i\partial\eta_k}][\frac{\partial^2 F}{\partial\eta_k\partial\eta_l}]^{-1}[\frac{\partial^2 F_c}{\partial\epsilon_j\partial\eta_l}]. \tag{3.6}$$

The Landau free energy F_L is represented by a polynomial in powers of the order parameters (η_i) associated with the transition. F_0 accounts for the portion of free energy in which order parameters or strains are not involved. The equilibrium state of a system can be determined by minimizing F with respect to ϵ and η. A more detailed discussion on the relationship between phase transitions and elastic consonants, with the focus on ultrasonic and resonance measurements, can be found in Leisure [63] and Rehwald [94].

3.4 Application of RUS Methodologies in High Temperature and Pressure Materials Study

The sample cell with two transducers are the core parts of the RUS instrumentation. By modifying the sample cell to be placed in a controllable furnace, one can investigate materials at higher temperatures. Depending on the requirement of the experiment, this system may be further modified to have a sample cell with the capability of controlling physical environments such as pressures, electric and magnetic fields, etc. Figure 3.1 shows a schematic of a high-temperature, high-pressure RUS sample cell.

The current RUS instrumentation at the University of Mississippi National Center of Physical Acoustics is capable of investigating material properties in the temperature range of 233 K – 1272 K and the pressure range of near vacuum to 6.9 MPa ($\approx$ 68 atm). However, it is possible to push these temperature and pressure limits of the technique to a higher level by tweaking the system. The major limitation in the standard RUS measurement to reach much higher temperatures is the operating performance of transducer materials at higher temperatures. Li and Gladden considered lithium niobate as the best candidate to push this limit to about 850 K [66, 67] . To push the upper temperature limit close to 1273 K, they have used the conventional buffer-rod setup, which guides the acoustic signal through either alumina or fused quartz from the transducer to sample and vise versa. That will keep the transducers in the cold zone, even while the sample approaches much higher temperatures. However, this buffer-rod system still has a few issues, such as the weight of the buffer-rod creating a slight pressure on the sample. Buffer-rod resonances appearing as extra peaks in the MRS, which need to be eliminated in the data analysis. The sample response signal in the buffer-rod system may become significantly weaker compared to standard direct contact system, resulting in a low signal to noise ratio (SNR).

Adebisi and Gladden designed and developed instrumentation to perform high temperature and high pressure RUS to investigate pressure effects on resonance modes and elastic constants of single crystal palladium hydride material near a tri-critical point. Hydrostatic pressure is achieved by pressurized hydrogen gas, also served as the medium inside

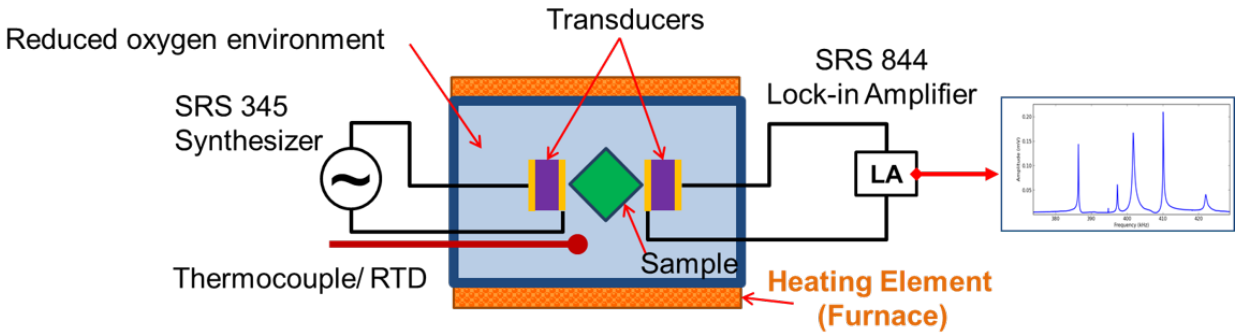

Figure 3.1. Schematic of high temperature and pressure RUS instrumentation.

the sample testing chamber [2]. That work extended the capability of the high temperature RUS instrumentation designed and developed by Li and Gladden to investigate materials in the temperature range of room temperature to 1000 C.

For the need of investigating materials in much higher temperatures while eliminating the temperature effects on transducer material, a non-contact RUS system is more appropriate. One such method is laser-based non-contact Resonant Ultrasound Spectroscopy which separates the detention transducer by a laser based vibration detection system. Sedlak et al. [98, 99] published their work on modifications to the RUS system by replacing the excitation transducer with a pulse laser unit and the detection transducers by Laser Doppler Vibrometer (LDV) systems.

Constructing and maintaining a high temperature and high pressure RUS instrumentation with the capability of varying different environment parameters will take a significant time and effort but that will open many doors to study interesting physics of materials as well as the measurements capabilities of such instrumentation will benefit the application oriented research and development on advanced materials.

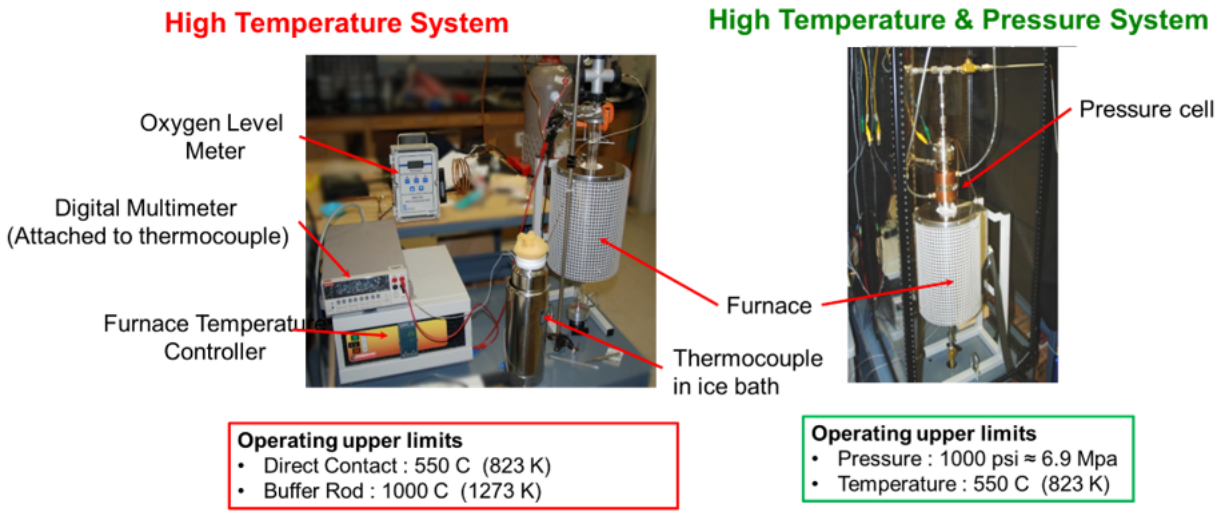

Figure 3.2. RUS instrumentation at the National Center for Physical Acoustics, University of Mississippi.

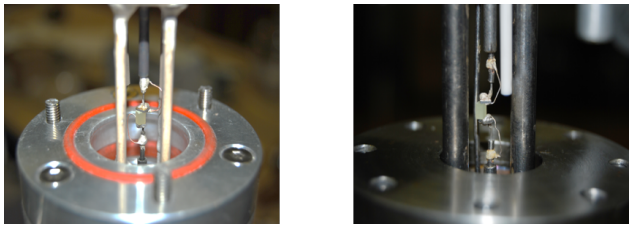

Figure 3.3. PMN-PT sample mounted in the direct contact RUS cell.

CHAPTER 4

ELASTIC PROPERTIES OF RELAXOR FERROELECTRIC PMN-PT

4.1 Relaxor Ferroelectric Materials

Ferroelectricity is the phenomenon of the existence of spontaneous electric polarization in certain materials where the polarization can be reversed by the application of an external electric field. The name ferroelectricity comes from the analogues phenomenon of ferromagnetism, in which some iron (Fe) based materials exhibit a permanent magnetic moment. Even though the term "ferro" is used, unlike the ferromagnetic materials most ferroelectric materials do not contain iron. The ferroelectric effect was first observed by Valasek in 1921, in Rochelle salt ($KNaC_4H_4O_6 \cdot 4H_2O$) [31, 108]. Since then many different inorganic materials in their crystalline forms have been classified as ferroelectric materials, and several new organic ferroelectric materials are also being discovered [13, 44, 57].

Ferroelectric materials are utilized in a broad range of optical, acoustical and electronic applications. They are used in memory modules (RAM) and other electronic components in microelectronic devices, electro-optical-acoustic modulators, electro-mechanical actuators and transducers, and ultrasound sensors [57]. Ferroelectric materials going through a geometrical deformation under the application of an electric field are classified as relaxor ferroelectrics and are widely used in acoustic energy transduction devices as piezoelectric elements (see Figure 4.1). "The complex lead-based perovskites having the general formula of $Pb(A_{1-x}B_x)O_3$ and exhibiting broad and frequency dispersive phase transitions" where A and B sites are filled by Ti, In, Mg, Nb are commonly used for this purpose [101]. Lead zirconate titanate (PZT): $Pb(Zr_{1-x}Ti_x)O_3$ has been a top choice as a piezoelectric material used in various piezoelectric transducers for a long period of time.

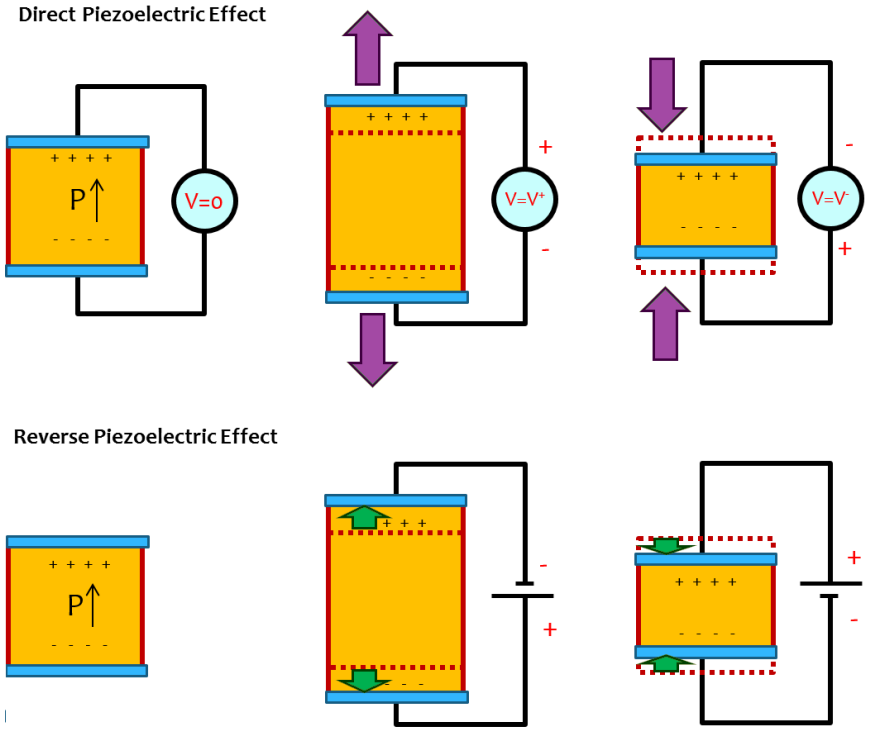

Figure 4.1. Direct Piezoelectric effect (internal generation of electrical charge resulting from an applied mechanical stress) and Reverse Piezoelectric Effect (internal generation of a mechanical strain resulting from an applied electrical field).

Understanding the complete set of elastic, dielectric, and piezoelectric constants and their relationships are crucial in the piezoelectric materials research. Details on 21 elastic constants were discussed in Chapter 2. In general, there are 18 piezoelectric constants, and 6 dielectric constants for an anisotropic material. However, the number of these constants in specific materials is reduced based on their spacial symmetry. A good discussion on piezoelectric constitutive equations can be found in Sun et al. [104].

4.2 Lead magnesium niobate-lead titanate (PMN-PT)

PMN-PT is a relaxor ferroelectric material that exhibits a higher piezoelectric coefficient, large electric-mechanical coupling coefficient, high dielectric constants and low dielectric losses. Some electromechanical properties of two solid solutions of PMN-PT and PZT are listed in Table 4.1. The dielectric constant K is the relative permittivity of a dielec-

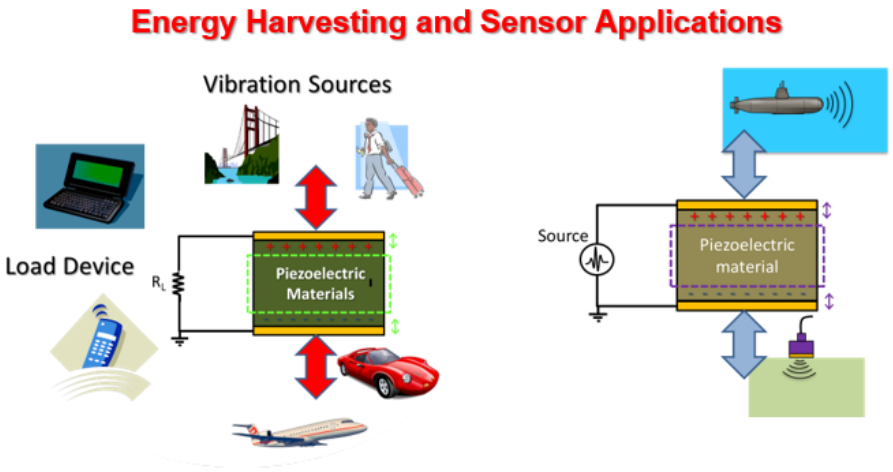

Figure 4.2. Applications of piezoelectric materials.

tric material. Piezoelectric constant d_{33} is the induced polarization in direction 3 per unit stress applied in direction 3, or induced strain in direction 3 per unit electric field applied in direction 3. The higher the Piezoelectric Constant, the easier it is to create large displacements. Electromechanical coupling factor (k_{33}) shows up-conversion efficiency between electric energy and mechanical (acoustic) energy for electrodes placed in the direction 3, and mechanical stress is applied along the same direction.

PMN-PT has ABO_3 perovskite crystal structure as illustrated in the Figure 4.3 [34, 87]. The molecular formula is $(1-x)\mathrm{Pb}(\mathrm{Mg}_{1/3}\mathrm{Nb}_{2/3})\mathrm{O}_3$-$x\mathrm{PbTiO}_3$. It is reported in the literature that PMN-PT single crystals, in the rhomboidal phase near the morphotropic phase boundary, exhibit large piezoelectric, dielectric, and electromechanical coupling coefficients. These promising indicators in PMN-PT have attracted great attention because of the ample room for enhancing the material properties to be utilized as transducer elements with higher sensitivity and broadband characteristics. It can also be used in vibration energy harvesting units and other electromechanical applications such as piezoelectric actuators, which can yield more mechanical output than use of conventional piezoelectric elements (Figure 4.2).

Table 4.1. Comparison of electromechanical properties PMN-PT and PZT [16].

Material Property	PMN-PT (PT $\approx$ 0.27 – 0.30)	PMN-PT (PT $\approx$ 0.30 – 0.33)	PZT
K_{33}^T	$\approx$ 4500 – 5500	$\approx$ 5500 – 6500	$\approx$1000 – 3400
d_{33}^T (pC/N)	$\approx$ 1400 – 2000	$\approx$ 2000 – 3500	$\approx$ 200 – 600
k_{33}	$\approx$ 0.87 – 0.90	$\approx$ 0.90 – 0.94	$\approx$ 0.65 – 0.75

4.3 Phase Diagram(s) of PMN-PT

Phase diagrams are important in understanding and mapping the existence of a material under different physical and chemical conditions. Temperature and Pressure (T-P) phase diagrams are commonly used for this purpose. Morphotropic phase diagrams illustrate phase transitions of a material due to changes in the chemical composition and are often used to map the phases of a material in varying temperature and chemical compositions.

Figure 4.4 shows the morphotropic phase diagram of PMN-PT in temperature and chemical composition. Current studies on PMN-PT report three major phases and transitions as a result of varying temperature and chemical composition. Several authors defined the morphotropic phase boundary (MPB) as the phase transition between the rhomboidal (R) and tetragonal (T) phases as a result of varying the composition. The relaxor phase R belongs to the $4mm$ space group, and the relaxor phase R belongs to the $3m$ space group [35] and the ferroelectric phase T belongs to the space group $P4mm$ [35]. Near the MPB, the crystal structure changes abruptly and piezoelectric properties of the material become maximum, and the existence of monoclinic (M_c) phases around the MPB are also reported [24, 34, 70]. This region of the morphotropic phase diagram was extensively studied by many other researchers. First principles calculations [90], X-ray diffraction studies [56, 82], neutron scattering experiments and dielectric constant measurements [40, 70, 101] were carried out in mapping the meromorphic phase diagram. Experimental evidence on the effects of hydrostatic pressure on dielectric response and the phase transitions of PMN-PT reported by Xu et al. [113] and Hilczer et al. [42] conducted their research based on dielectric

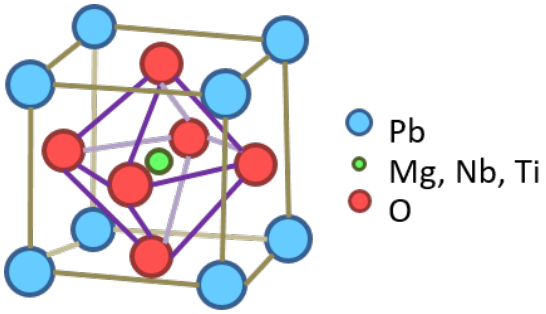

Figure 4.3. ABO_3 Perovskite structure of PMNPT.

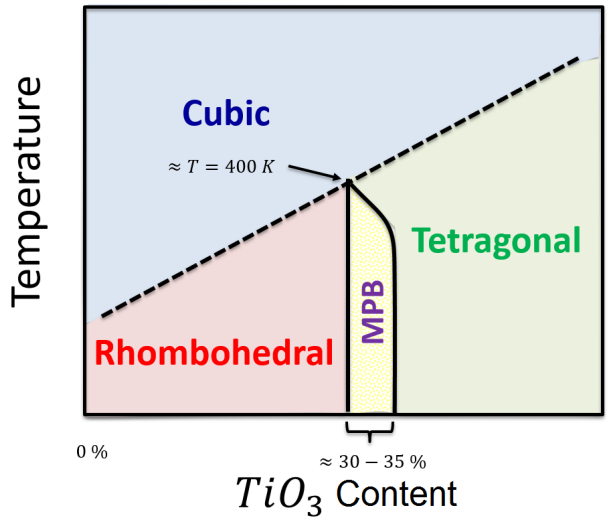

Figure 4.4. Temperature versus PT content phase diagram of PMN-PT. Note the morphotropic phase boundary region.

measurements up to 0.6 GPa hydrostatic pressure.

4.4 Recent Advancements of PMN-PT Material Development and Characterization

In the last couple of decades, PMN-PT and similar ferroelectrics were extensively studied as a vibration energy transduction material due to their promising electromechanical properties. Finkel et al. [27] and Dong et al. [24] and many other researchers have experimentally shown a significantly large electromechanical energy conversion utilizing phase transition in the material. These properties made PMN-PT an excellent candidate as an energy transduction material to be used in a variety of acoustic transducers, actuators, and sonar applications. Understanding the mechanisms behind the high electromechanical

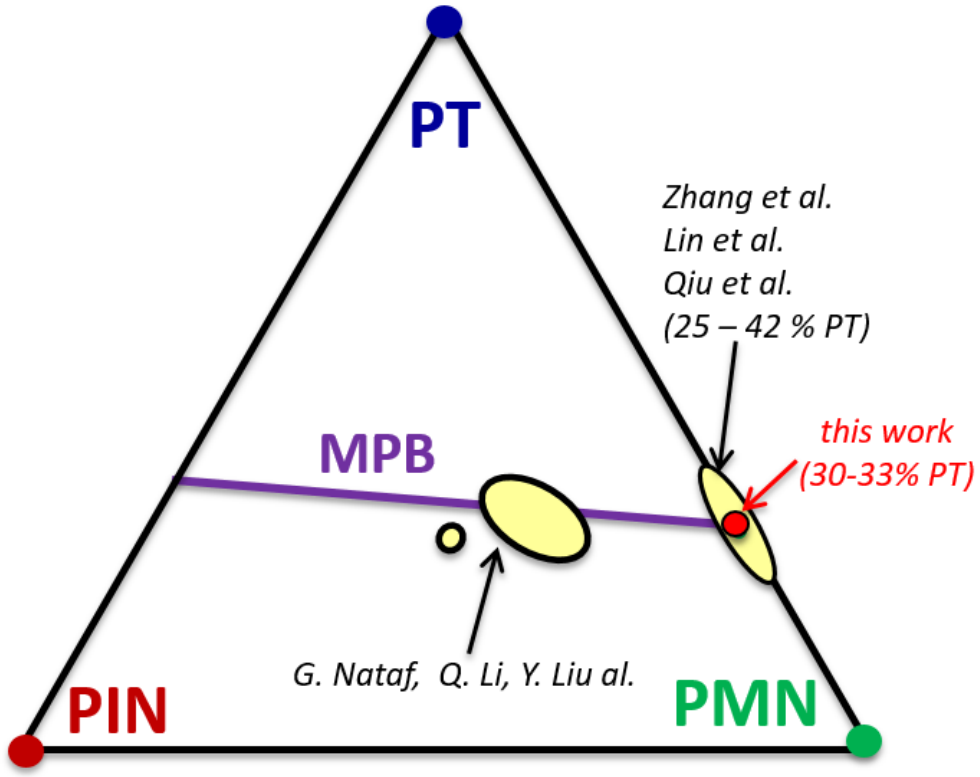

Figure 4.5. Phase diagram for PMN-PIN-PT ternary system [80].

properties of the PMN-PT and similar ferroelectrics is an important area to explore in the materials research. Some of the work related to morphotropic phase diagram were mentioned in the previous section. There are lots of other studies that were conducted on PMN-PT and similar lead titanate relaxor ferroelectric materials targeting on enhancing the material properties and characterization properties. A detailed discussion on structure and high piezoelectricity in lead titanate relaxor ferroelectrics is available in Noheda [81]. Ahart et al. reported their findings on the origin of MPB in ferroelectrics from a combination of theoretical and experimental studies on pure lead titanate in high temperatures and pressures [3]. Frantti et al. reported their findings of the first principles computation on pressure induced phase transitions related to the polarization rotation theory [32].

Even though the PMN-PT is an extensively studied material, the experimental data on mechanical properties of PMN-PT in high temperatures and pressures are minimally investigated. Bao et al. have reported their findings on internal friction and elastic moduli

of three different chemical compositions of PMN-PT [8]. Nataf et al. [80] worked on the PIN-PMN-PT material, which is the ternary relaxor ferroelectric material being studied. Their study has reported significant acoustic loss temperature range of $\approx 290\ K - 380\ K$. A complete set of material constants for both PIN-PMN-PT and PMN-PT measured from ultrasonic and resonance methods is available in Liu et al. [68] and Cao et al. [15]. With the lesser amount of available information related to temperature and pressure effects on mechanical properties of PMN-PT, many questions of physics are remain open to explore. This work is primarily focused on addressing this gap of knowledge in the temperature and pressure effects on mechanical properties of PMN-PT.

4.5 PMN-PT Samples

Unpoled PMN-PT single crystal material grown using the Bridgman method [58] was obtained from the H.C. Materials Corporation. Chemical compositions of the specimens, were reported to be near the morphotropic phase boundary with $30 - 33\%$ PT content. The specimen was oriented along the [001] crystal axis. The sample preparation method described in Chapter 2 was followed to prepare RPP samples from the specimens. These were cut and polished to approximate dimensions of 2 mm $\times$ 2.5 mm $\times$ 3 mm. Information about the samples that are investigated in the RUS study are shown in Table 4.2. There is a notable variation in the density measurements in the samples. Sample masses and dimensions were re-checked to verify the accuracy of the density values. The measured density of the bulk material sample (Figure 4.6) received from HC materials was 8.09 g·cm^{-3}. Additional samples prepared from the bulk material specimen also showed variations in the density between 8.02 g·cm^{-3} to 8.17 g·cm^{-3} (2% of measured density of bulk specimen).

Uncertainty in the density measurement of these millimeter-scale small samples is associated with uncertainties in the sample preparation and the measurement steps. During the sample preparation, human and instrument errors can be introduced to the geometry of the sample. One such error is the parallelism in cutting and polishing samples. These

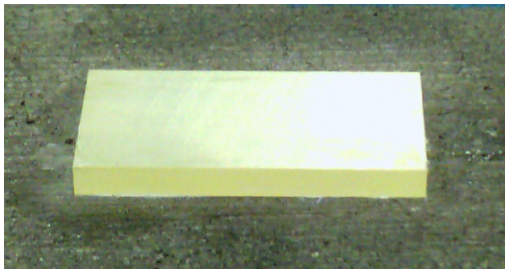

Figure 4.6. PMN-PT material slab (20 mm × 10 mm × 2 mm) received from HC materials.

Table 4.2. PMN-PT sample names and dimensions

Measurement	Sample 1	Sample 2	Sample 3
d_x (cm)	0.313	0.354	0.355
d_y (cm)	0.236	0.258	0.243
d_z (cm)	0.200	0.199	0. 193
mass (g)	0.120	0.146	0.135
Density (gcm^{-3})	8.123	8.033	8.107

errors are relatively more important when the samples are smaller in size. Careful microscopic observations and high precision cutting and polishing tools were used to minimize these errors. Precision in the geometry and mass measurements also contribute to the uncertainty in density measurements. Considering all these factors, the uncertainty of density measurements is determined as ± 0.06 g·cm^{-3}. Variations in the density measurements of PMN-PT samples are well within these uncertainty limits. It is also found in the literature that Cao et al. [15] reported a density variation of PMN-PT related to the change of PT content. However, the cause for the density variation within samples is determined to be the measurement uncertainty because the variation of PT content cannot be expected in the range reported in Cao et al. [15] to match with the density variation in the samples of this study.

4.6 High Temperature RUS Results

4.6.1 Temperature Cycles

Several high temperature RUS measurement cycles were performed on three different PMN-PT samples. During the first phase of this study, sample 2 was exposed to different temperature cycles in high temperature resonant ultrasound spectroscopy experiments. Cycle 1 was done with 50 K temperature increments up to 523 K, and Cycle 2 was done with 10 K temperature resolution around the temperature range of 373 K (100 C) - 423 K (150 C), where a significant change in the spectrum was observed in the previous run (Figure 4.7).

During cycle 2, the temperature range 373 K – 423 K was investigated using higher temperature resolution. The signal to noise ratio (SNR) becomes lower for temperatures below 400 K. Therefore, resonance peaks cannot be detected with confidence at lower temperatures. However, the peaks emerged from the noise at temperatures above 400 K resulting a higher SNR, where the spectrum becomes clear and peaks were prominent. This observation leads us to narrow down the transition range of 393 - 408 K (120 – 135 C) where the spectrum showed an abrupt change in the SNR. Temperature cycle 3 was done in the motivation of verifying the observations of cycle 2 in the temperature range of 293 K (20 C) – 473 K with a broader frequency range to extract resonance modes for a complete RUS study. Temperature cycle 4 was focused on temperatures below 400 K and above 573 K, the region not covered in the previous temperature sweep. Temperature cycle 4 was carried out until 873 K and cooled down to room temperature while recording the resonant spectrum at each temperature point.

Sample 1 was used for several high temperature and high pressure RUS experiments. The first high temperature RUS experiment on the sample 1 was done in the temperature range of 393 K – 473 K, investigating the frequency spectrum of the first 21 modes for their temperature dependence. The sample was later used in the study of understanding equilibrium dynamics in varying temperature and pressure. A high pressure RUS experiment

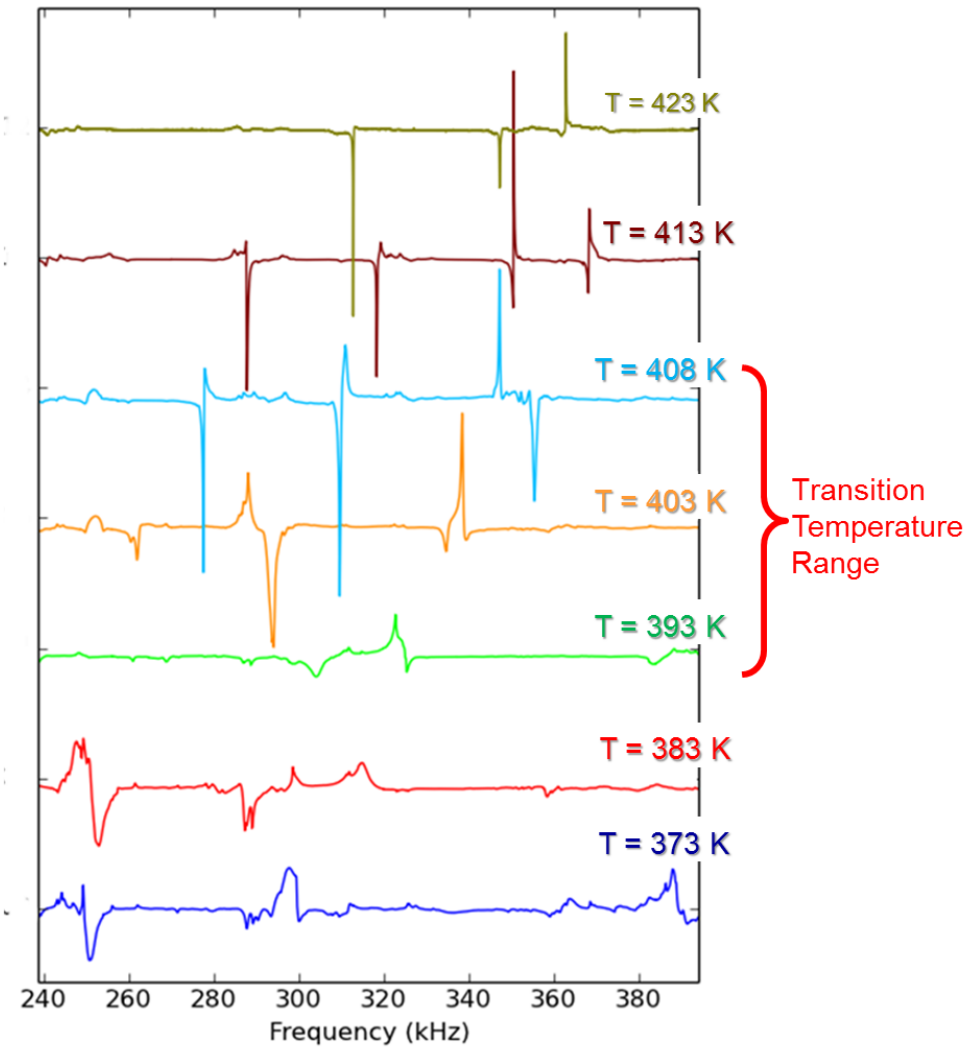

Figure 4.7. Resonant spectrum in different temperatures around the transition point.

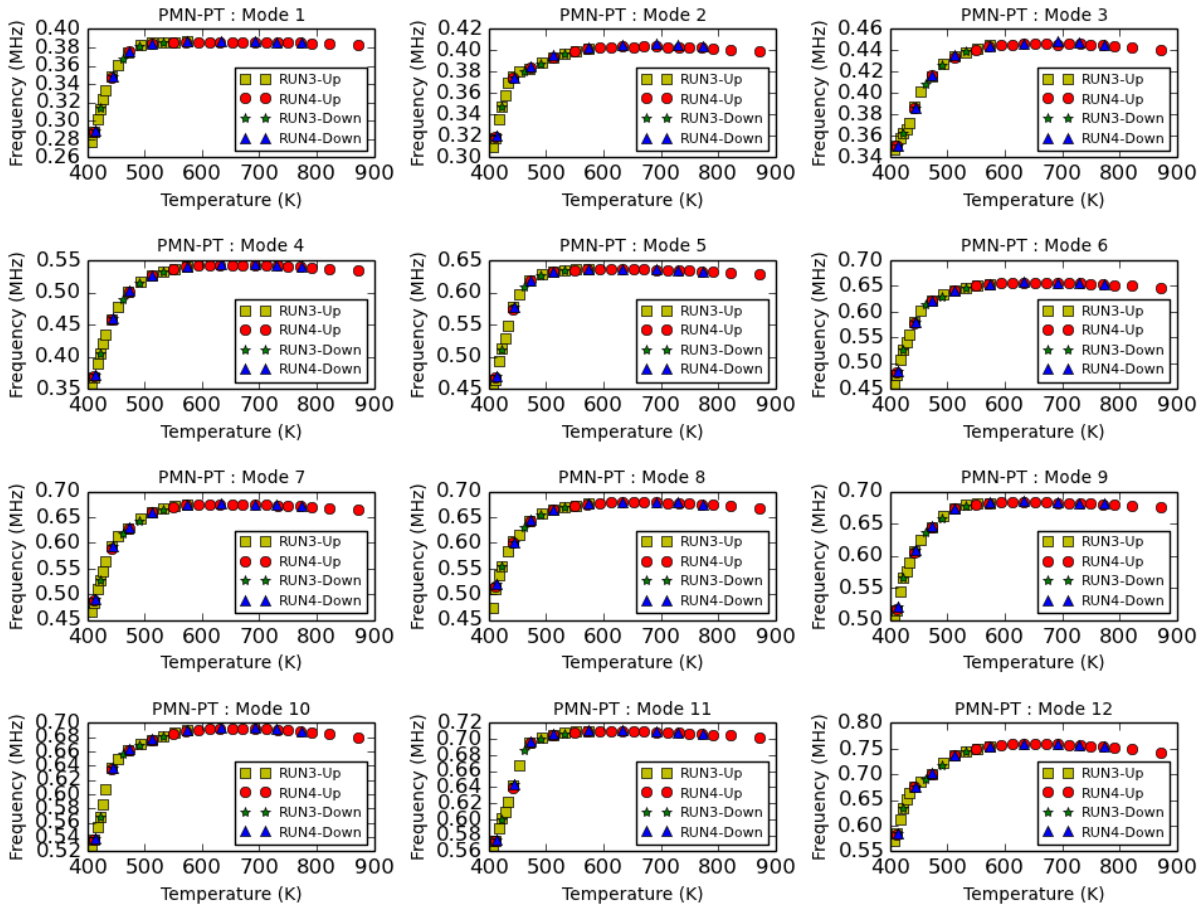

Figure 4.8. Temperature trends of mode frequencies of the first 12 modes for two different temperate cycles on PMN-PT sample 2 show higher temperature stability of the material.

was performed along the 414 K (141 C) isothermal line for the sample 1 in a pressure varying environment from 2 psi – 300 psi using 99.9% argon gas. A high temperature RUS experiment was performed on sample 1 in reduced atmospheric pressure (1 – 3psi). Finally, a high pressure RUS experiment was performed on sample 1 along isotherms 419 K (146 C), 529 K (256 C) and 653 K (380 C).

4.6.2 Equilibrium of resonance modes under changes in the temperature

During the measurements, we have to be certain that the material reaches equilibrium after environment parameters such as temperature and pressure are changed. If the material reaches equilibrium with a large time constant, resonance measurements will not reflect material properties in the desired temperature and pressure, but a value during the path of reaching equilibrium. The RUS apparatus also has its own equilibration characteristics. Therefore, it is important to understand equilibrium dynamics of both the RUS system and the material specimen with the changes in environment parameters. This can be achieved by monitoring the selected peak over time range with varying environment parameters (temperature and pressure). Peak frequency can be recorded from the time before the changes were made until a few minutes after reaching equilibrium. Temperature, pressure, and the peak frequency may reach their equilibrium with different time constants, which means a full range RUS scan can be started only after the system is in this equilibrium state to ensure accuracy of the measurement.

Sample 1 was used to investigate equilibration of the resonance modes with respect to temperature. A temperature and pressure equilibrium study revealed that resonance modes of the material showed a fast response to changes in the temperature where as the equilibrium time constant was negligible. System dynamics on varying temperatures and pressures were also investigated.

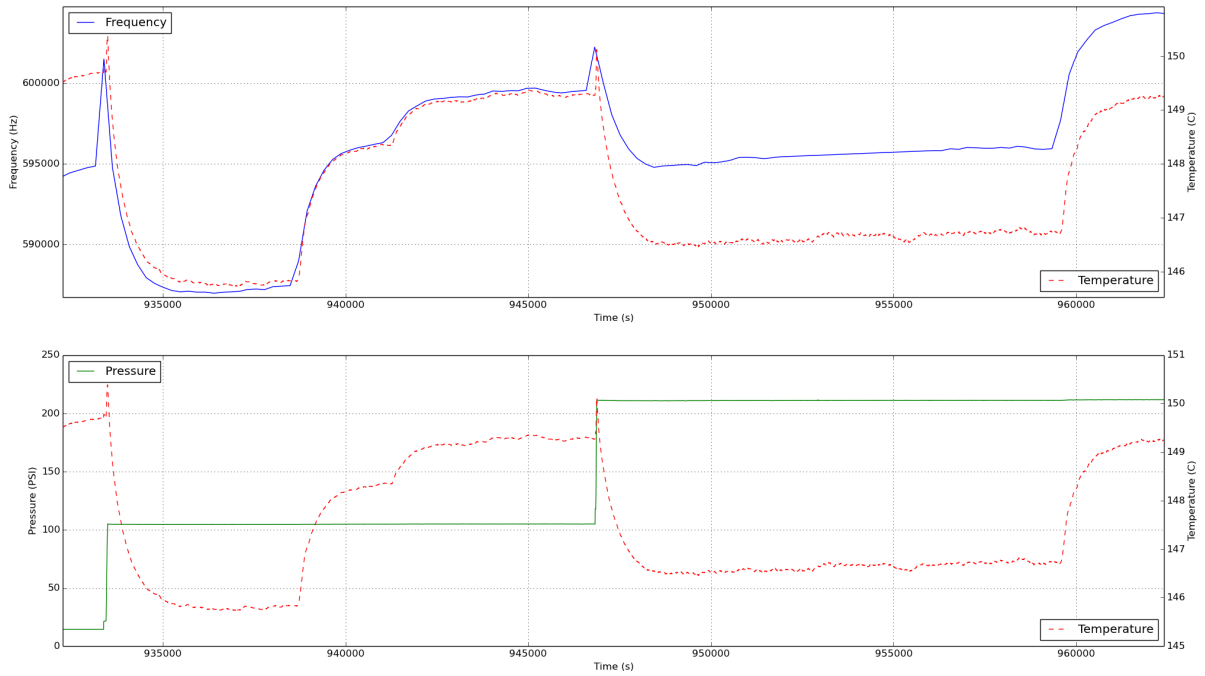

Figure 4.9. Investigating thermodynamic characteristics of the system by varying temperature and pressure.

4.6.3 High Temperature Results

At temperatures above 400 K, spectra for both samples were investigated, and the center frequencies and average quality factor of the resonance peaks were plotted as functions of temperature. Mode frequencies showed a monotonically increasing relationship with the temperature, followed by a gradual decrease, which can be interpreted as stiffening and softening behaviors, respectively in that temperature range.

The first 21 resonant modes of the PMN-PT sample were investigated in the temperature range of room temperature up to 850 K. Trends of natural frequencies in both heating and cooling did not show signs of hysteresis, which was also consistent in different heating-cooling cycles. Acoustic attenuation derived from the quality factor of resonance modes showed an inverse relationship with the temperature, which indicates higher attenuation in lower temperatures. These results can be compared with the high temperature elastic constants obtained for the strontium titanate sample ($SrTiO_3$), which is a perovskite titanate system that will be later discussed in Chapter 5. Despite the pronounced peak around 470 K, the relative attenuation curve of $SrTiO_3$ shows an increasing trend with the increase of temperature, which is observed in most materials.

Elastic constants were then derived from the frequency data using the computational algorithm for cubic symmetry where the PMN-PT material is reported to be in the cubic phase in the temperature range above 400 K. Similar to frequency trends, elastic constants showed drastic increase in the elastic constants C_{11} and C_{44} in the temperature range of 400 K – 600 K. However, the increasing trend of C_{44} slowed down at lower temperature than C_{11}. These results can be compared with the high temperature elastic constants obtained for the strontium titanate sample ($SrTiO_3$). As seen with most ordinary materials, gradual softening behavior is observed in strontium titanate when increasing the temperature, indicated by a negative slope in the elastic constant trends.

Mode interactions were detected between modes 2 and 3 around 430 K, and modes 3 and 4 around 405 K, modes 8 and 9 around 430 K, and modes 13 and 14 around 435

Table 4.3. Peak temperature locations of $C_{11},\ C_{44},\ A_Z\ and\ Q^{-1}$ based on RUS measurement of two different samples. See Figures 4.10 and 4.11 for plots.

C_{11}	$\approx$ 650 K – 700 K
C_{44}	$\approx$ 560 K – 610 K
A_Z	$\approx$ 623 K – 673 K
Q^{-1}	$\approx$ 500 K – 550 K
$ResoanceFrequencies(M1 - M16)$	$\approx$ 593 K – 693 K

K during a coarse temperature sweep (Figure 4.15). A fine temperature sweep on the first four modes revealed a repulsive interaction between modes 2 and 3 at 430 K (Figure 4.15). These mode interactions can be explained as repulsion of non orthogonal modes near phase transition region [4].

Temperature trends of mode frequencies peaked at the temperature range of 573 K – 673 K, and followed by a gradual decrease (Figures 4.13 and 4.14). This indicates the material is undergoing a softening process at temperatures above the peak temperatures. This observation is reflected in the temperature trends of the longitudinal (C_{11}) and shear (C_{44}) moduli, which were peaked in the temperature ranges of 650 K – 700 K and 550 K – 600 K respectively (Figure 4.12).

Temperature dependence of the elastic anisotropy of the material has been analyzed using the Zener Anisotropy ratio (A_Z) discussed in Chapter 2. Within the temperature range of 400 K – 823 K, A_Z exponentially dropped approximately by 50 % and gradually increased after a minima ($\approx$ 1.56) in the temperature range 650 K – 700 K.

4.7 Turning the Pressure Knob: High Temperature and High Pressure Study

4.7.1 Procedure

A high temperature RUS experiment was performed under low hydrostatic pressure (1 - 3psi) to compare with the previously obtained results in the atmospheric pressure. High temperature results in low hydrostatic pressure are shown in Figure 4.11. The slight softening observed in low pressure in the same temperature regime lead to the next phase of this study,

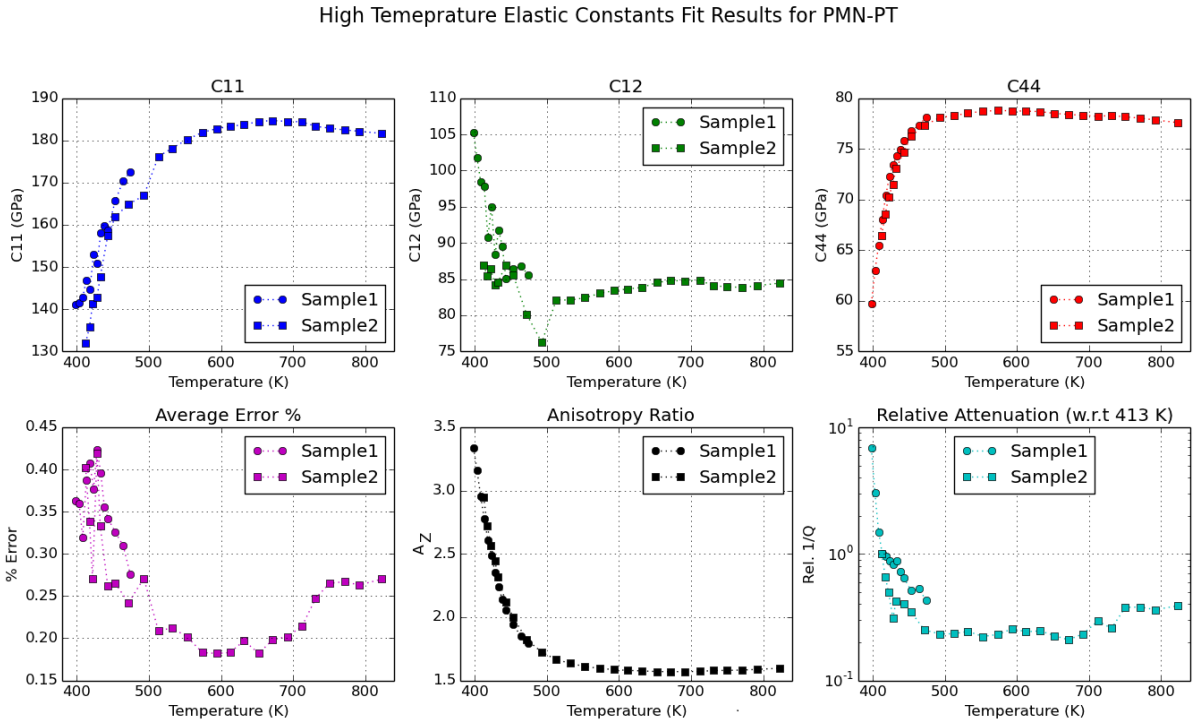

Figure 4.10. Elastic constants, quality factor and relative attenuation versus Temperature for PMN-PT sample1 and sample2.

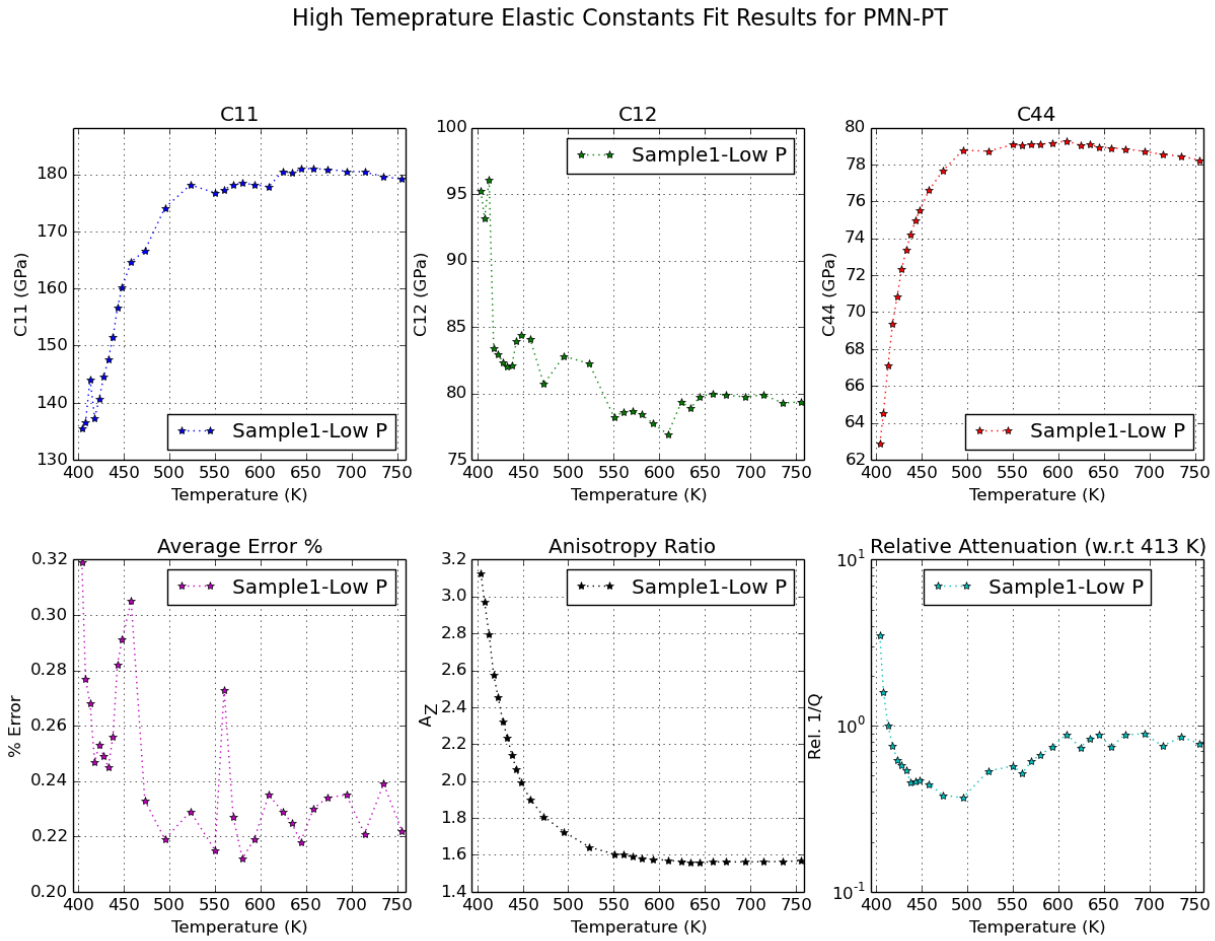

Figure 4.11. Elastic Constants and Quality Factor and Relative Attenuation versus Temperature for PMN-PT sample 1 at reduced atmospheric pressure (1 – 3psi).

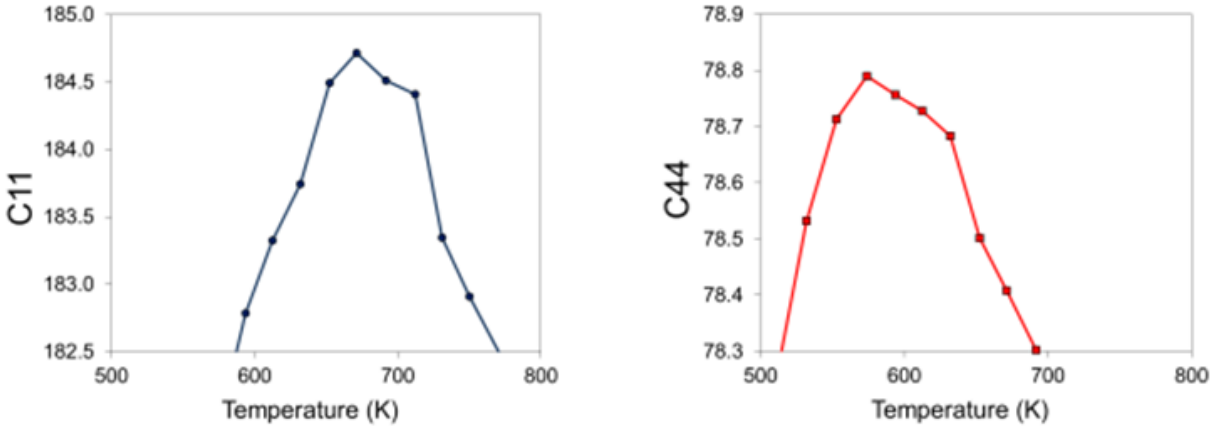

Figure 4.12. Peaks at the curves of elastic constants C_{11}, C_{44} versus temperature.

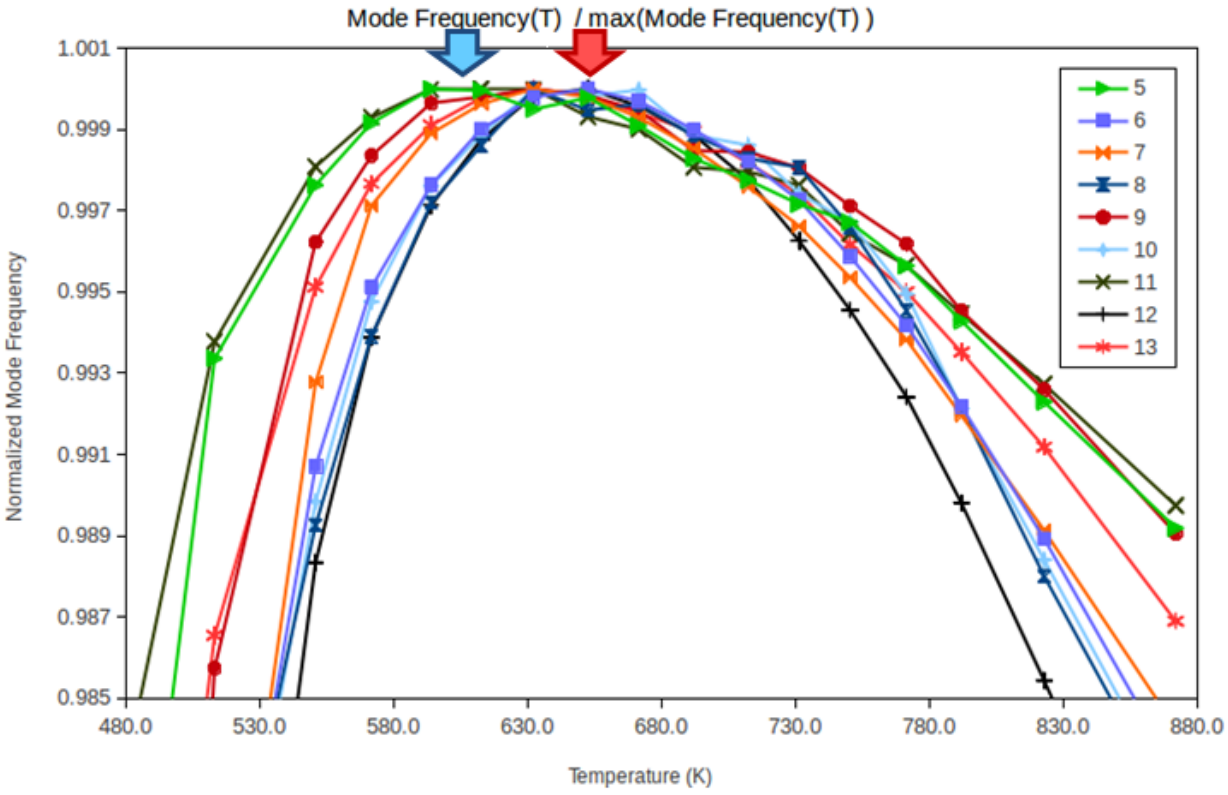

Figure 4.13. Peaks at the curves of natural frequencies of selected modes versus temperature. Amplitude of natural frequency was scaled to the maximum of each curve.

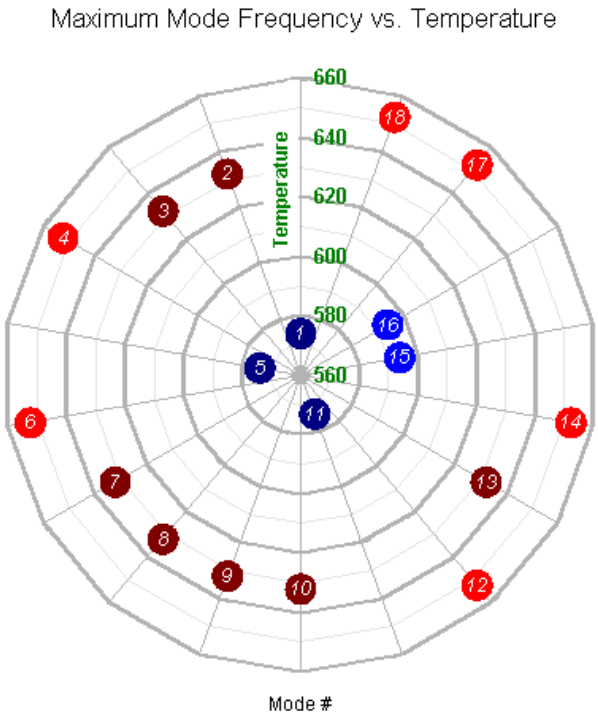

Figure 4.14. Grouping individual modes according to the temperature at the temperature versus elastic moduli maxima.

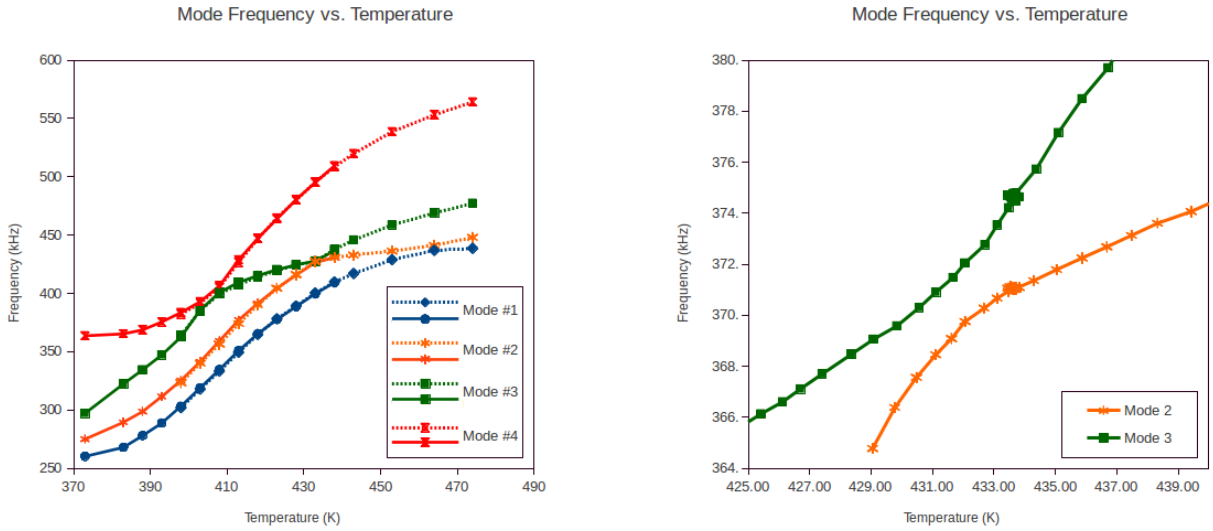

Figure 4.15. Mode interaction modes 1 and 4 shows a repulsive interaction.

which involved varying pressure as an environment parameter.

4.7.2 Exploring Temperature and Pressure Map

Previously described high temperature results from RUS experiments under atmospheric pressure ($\approx$ 14.2 psi) , and at reduced atmospheric pressure (1–3 psi). Based on initial observations, the next steps of this study was focused on exploring the temperature and pressure map (Figure 1.3) to investigate elastic properties of the material at different temperatures and pressures in order to get insight into the phase transitions.

In the work presented in this dissertation, four isotherms have been studied to investigate pressure effects on the elastic properties. In the isotherms 419 K and 473 K, the material is in the transition region where elastic properties change rapidly. In the isotherm 529 K, where the material is undergoing a certain stabilization process and elastic properties change at a slower rate with the increase of the temperature. The previous high temperature study of the material revealed a peak in elastic constant curves in the temperature range of 573 K – 673 K. Last isotherm 653 K was picked to be in this region. Mode frequency shifts from the measurements in vacuum and at various pressure for different isotherms are shown in Figure 4.16.

Relative attenuation (scaled to attenuation at 0 psi) was calculated using the quality factor averaged over several modes. The 419 K isotherm showed rapid increase of the attenuation with the increase of pressure. However, the attenuation increased at a lower rate in the isotherms 529 K and 653 K. This is an indication of an increased loss mechanism in the 419 K isotherm where the material is at the rising edge of the phase transition.

The pressure dependence of acoustic attenuation of PMN-PT was compared with the results of a similar experiment on an aluminum sample measured by Adebisi [2] in the process of testing the high pressure RUS system. The aluminum sample showed minimal response to the pressure change in the range of 0 – 540 psi at 373 K isotherm compared to the PMN-PT sample in all three isotherms measured (419 K, 523 K, 673 K), which showed substantial

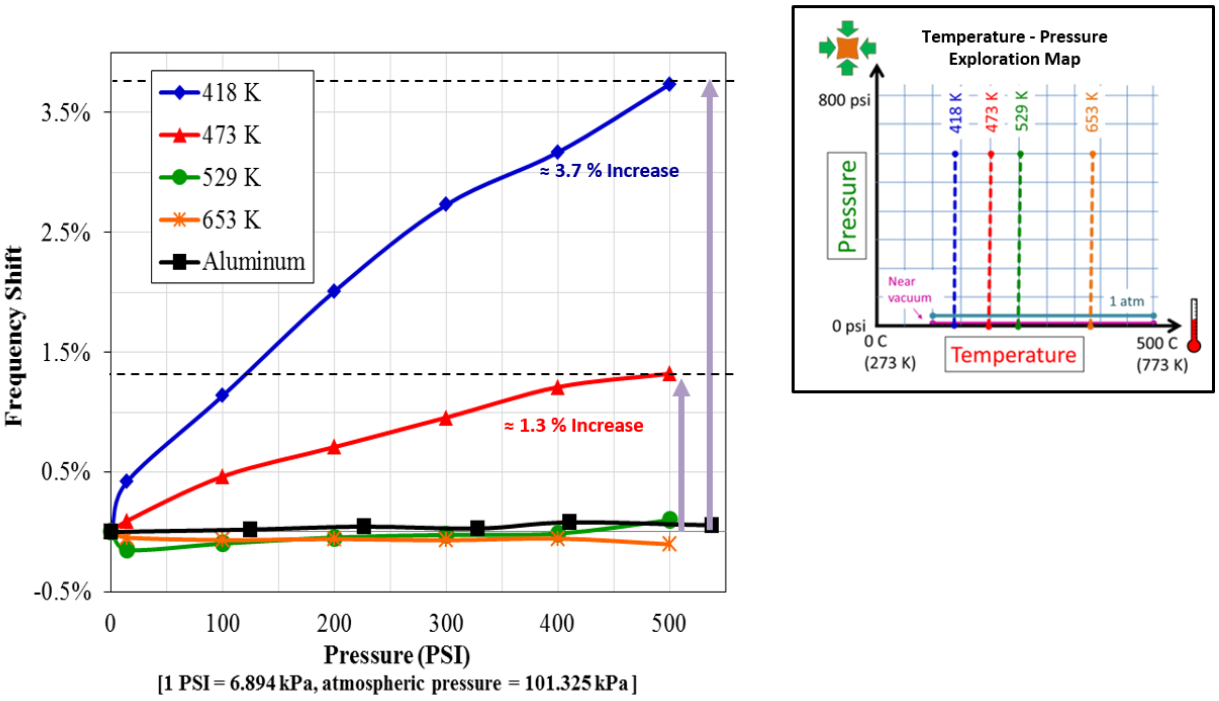

Figure 4.16. Average percentage mode frequency shifts versus pressure for different isotherms of PMN-PT and for aluminum at room temperature (293 K).

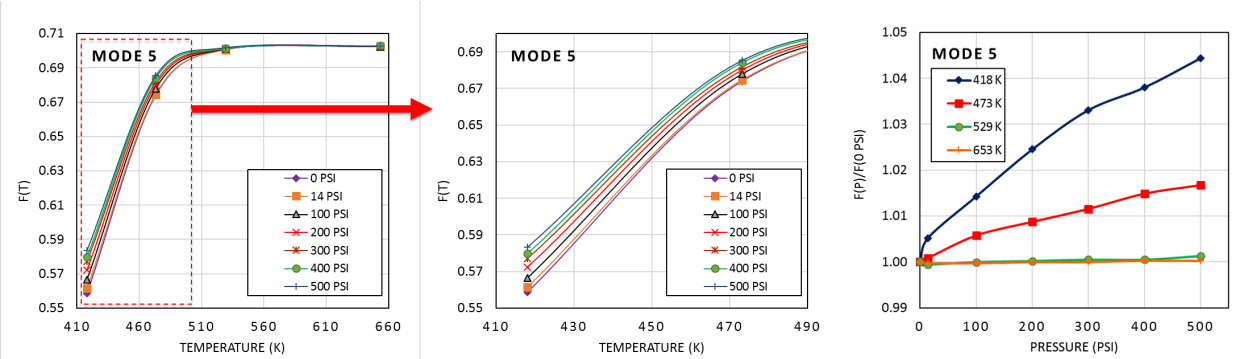

Figure 4.17. Temperature and pressure response of mode 5.

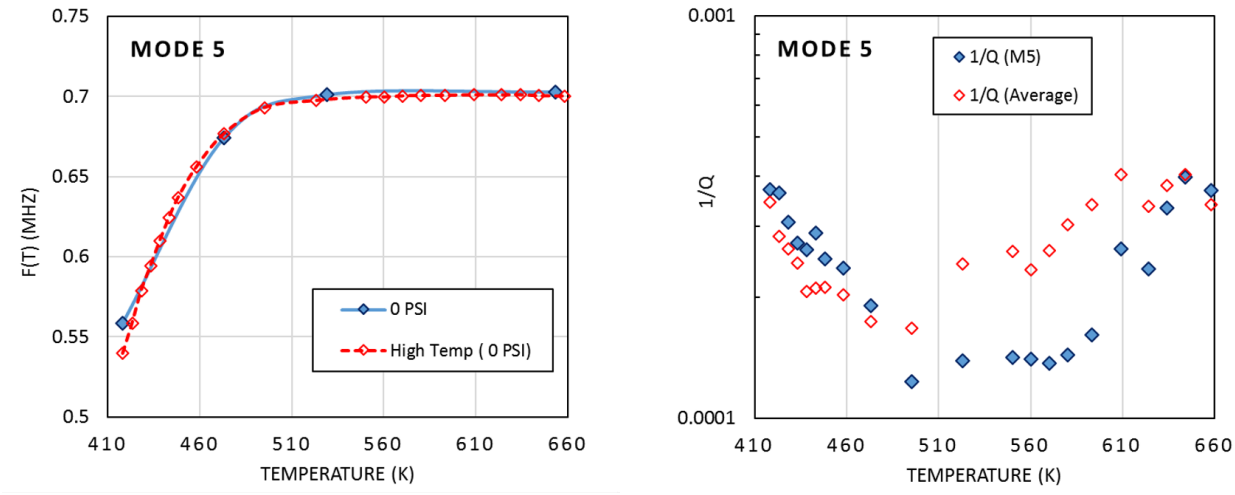

Figure 4.18. (a) Resonance Mode 5 frequency trends of the high pressure RUS compared to high resolution data taken at 0 PSI, (b) Temperature trends of 1/Q averaged over 16 modes to 1/Q of Mode 5.

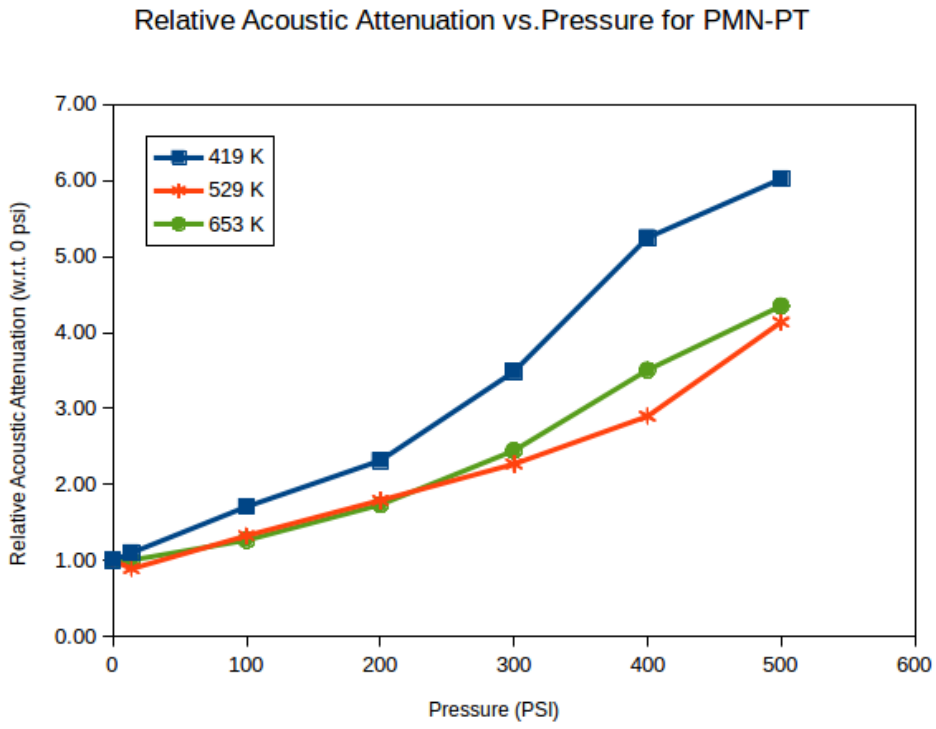

Figure 4.19. Pressure dependence of the attenuation (1/Q) averaged over frequency modes.

response to the pressure change in the same range along the 419 K isotherm. In the pressure range of 0 - 500 psi, the increase of relative attenuation for PMN-PT at 419 K isotherm was nearly 500%, and for aluminum at 373 K isotherm it was only 170%. This observation indicates a significant pressure effect on the acoustic attenuation of the PMN-PT material in the phase transition region. Since the characteristics of the pressure sensitivity of resonance frequencies in the isotherms 529 K and 653 K of the PMN-PT sample are comparable with the pressure sensitivity of the aluminum sample within the pressure range being investigated. This is a typical behavior materials in response to increasing hydrostatic pressure at lower pressures. This indicates the material is behaving as a "normal" material in the isotherms 529 K and 653 K, where the material is passing the temperature range. These isotherms are within the temperature range, which includes the peaks of elastic moduli as functions of temperature, minima of attenuation and anisotropy ratio as a function of temperature, and the Burns temperature of PMN-PT reported in the literature (See Section4.10).

4.8 Measurements Below the Room Temperature

PIN-PMN-PT is a relaxor ferroelectric perovskite in the same ferroelectric family of PMN-PT, but a ternary material having indium shares the sites of Ti, Mg, Nb in the crystal. In a recently published research article based on an RUS study, Nataf et al. reported significant acoustic loss at temperatures below the transition temperature of PIN-PMN-PT [80], which is similar to what we have observed in the PMN-PT material. The mechanism behind the higher acoustic loss at lower temperatures of these materials is an open question. A low temperature RUS experiment was attempted to understand the resonance characteristics of the PMN-PT material below the room temperature. This part of the study is intended to locate any peaks emerging from the spectrum while cooling down the material, as Nataf et al. reported with the PIN-PMN-PT material that can allow to identify the temperature range of the higher acoustic loss. A laboratory freezer, which is capable of maintaining temperatures between -40 C and 10 C with fine adjustments, was used to set measurement temperature,

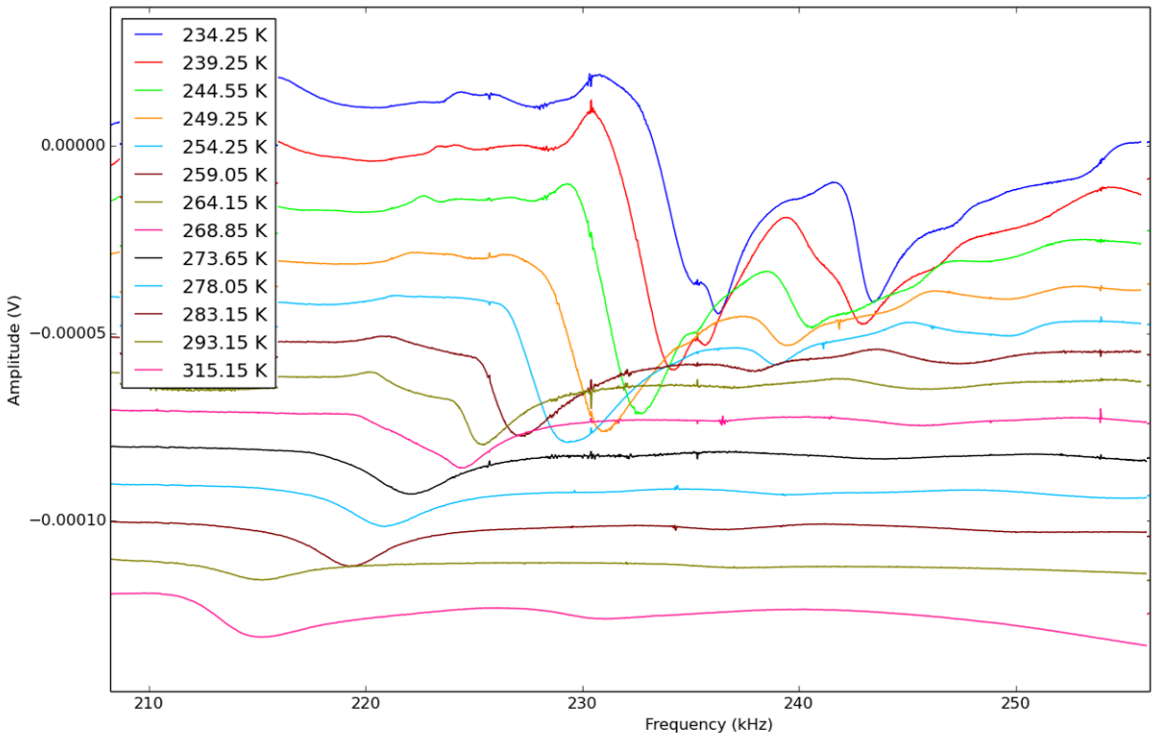

Figure 4.20. Low temperature RUS spectrum obtained at different frequencies.

and a custom made RUS apparatus was placed inside the freezer unit (Figure 4.22). Couple of peaks were tracked and in the temperature range of room temperature to 234 K (−40 C) as shown in Figure 4.20 and the square frequency versus temperature is shown in Figure 4.21.

4.9 Pulse-Echo Measurements

A good set of resonance data cannot be obtained from the PMN-PT samples at room temperatures because of the loss mechanisms described in the previous section. The pulse-echo method has been applied to measure longitudinal wave velocity of PMN-PT at the room temperature. The bulk material sample with 2 mm thickness has been used in this work. An Olympus V112 10 MHz longitudinal pulse transducer was coupled to the sample using

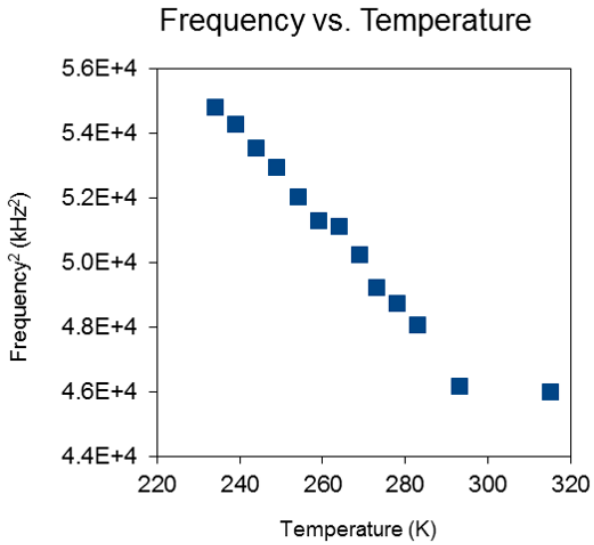

Figure 4.21. Tracking frequency with temperature for low temperature RUS measurement on PMN-PT sample.

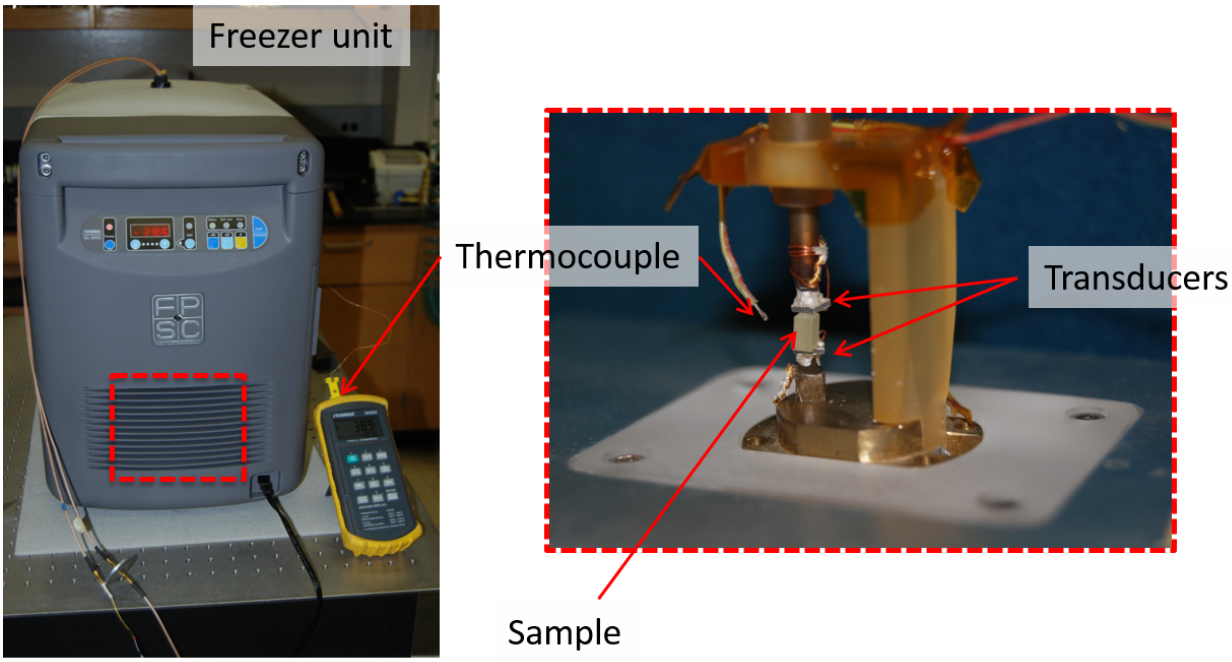

Figure 4.22. Low temperature RUS measurement setup.

Table 4.4. Room temperature pulse-echo results for the pulse measurement along z direction.

Height (mm)	20.033
Width (mm)	10.033
Thickness (mm)	1.997
Mass (g)	3.247
ρ (g cm^{-3})	8.089
V_3^l (ms^{-1})	3915.7
V_3^s (ms^{-1})	2780.3
C_{33} (GPa)	124.0
C_{44} (GPa)	62.5
$C_{11} = C_{33}$ (GPa) at 400 K (RUS)	$\approx 144 - 147$
C_{44} (GPa) at 400 K (RUS)	$\approx 67 - 68$

Ultrasonic Gel. Both direct contact and buffer-rod configurations were experimented (Figure 4.23), where the buffer-rod configuration is considered for high temperature measurements to prevent the transducer being exposed to high heat. In the direct contact method an echo train of the pulse reflected from the back surface could be obtained and the pulse transit time was determined by measuring the average time gap (Δt) between the first three echoes. This measurement was repeated for three different positions of the sample specimen and also reversing the reflection surface to make a total of six measurements. The speed of sound for both longitudinal and shear waves was calculated using the transit time and sample thickness. Sound speeds and associated elastic moduli are listed in Table 4.4. The method of calculating elastic constants from speed of sound measurements is described in Chapter 2. Sound speeds and elastic constant values are within the range of reported values by Cao et al. [15] and Sun et al. [105].

4.10 Discussion

PMN-PT undergoes a structural phase transition above a critical temperature (T_C). This work provides high-temperature elastic properties in the vicinity of the phase transition. Many studies have identified the crystal structure above this temperature in cubic symmetry, and the structure below T_C is not fully understood for chemical compositions near the MPB.

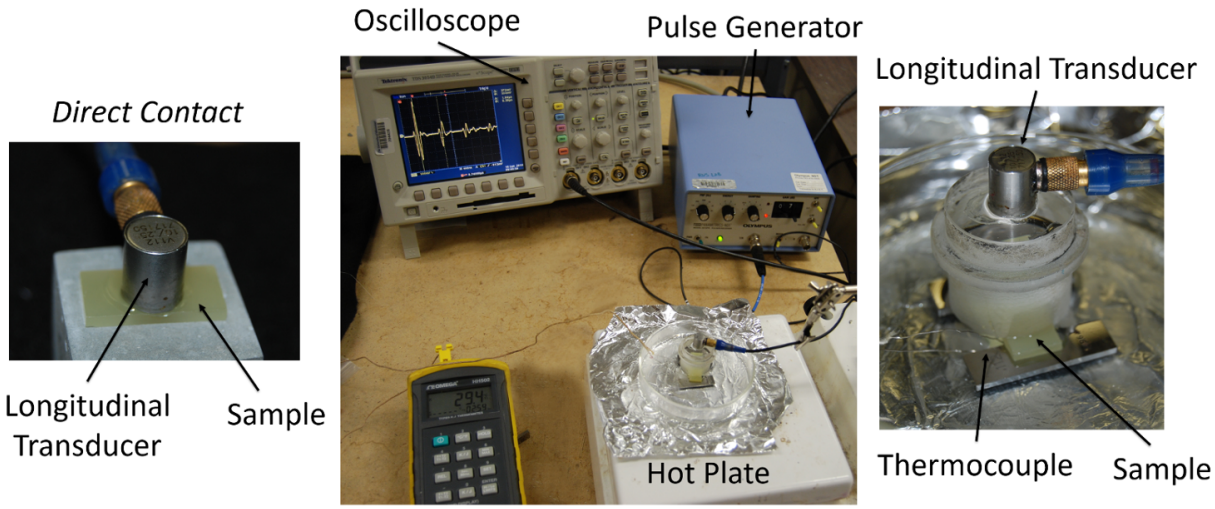

Figure 4.23. Pulse-echo measurements on PMN-PT bulk sample.

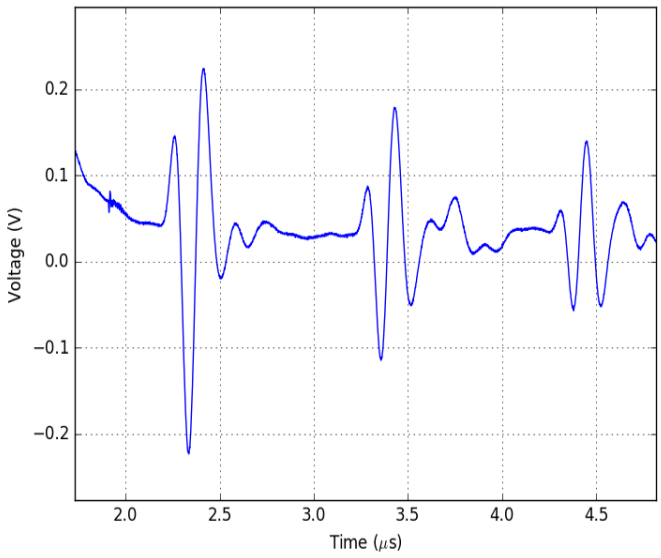

Figure 4.24. First three echoes from the longitudinal pulse (10 MHz) reflected from the back side of 2 mm thick of the PMN-PT slab (direct contact).

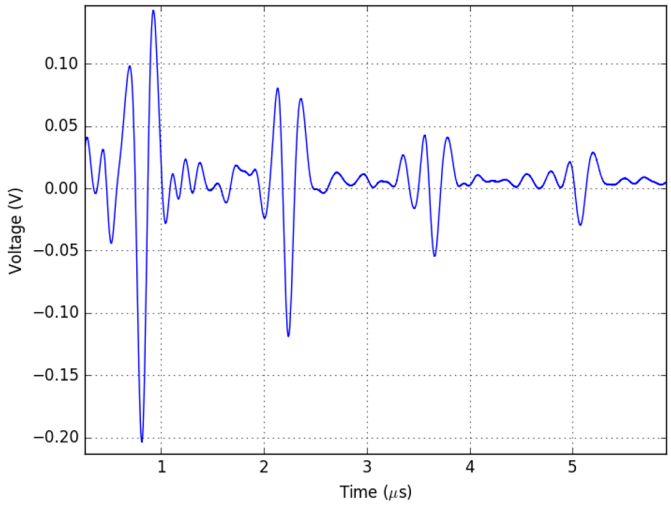

Figure 4.25. First three echoes from the shear pulse (5 MHz) reflected from the back side of 2 mm thick of the PMN-PT slab (direct contact).

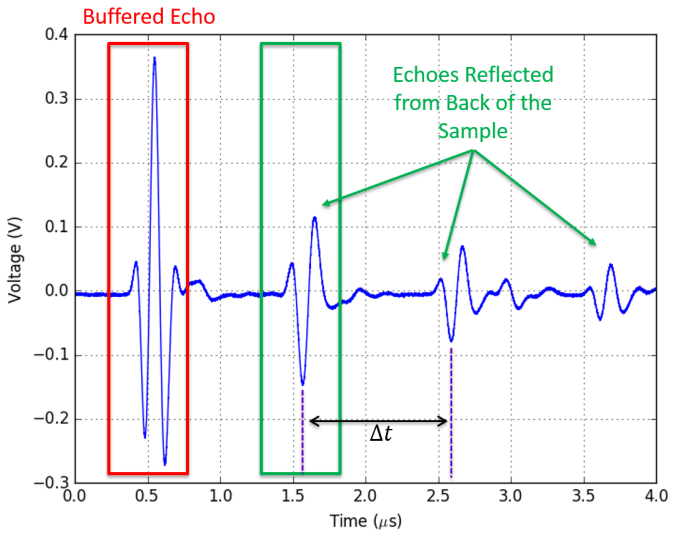

Figure 4.26. Buffer-rod echo and sample echoes.

A material near MPB is investigated in this study. A significant transition of the resonance spectrum is detected in the temperature range of 390 K – 410 K where the resonance peaks showed a decrease in attenuation (obtained from the quality factor: Q^{-1}) in the temperature above that range. The quality factors of resonances were significantly low close to this transition region and further decreased at lower temperatures indicating higher attenuation. The exact reason for this higher attenuation is not fully understood, but we will speculate on some possible mechanisms based on our experimental observation.

When the chemical composition is close to MPB, the PMN-PT is in a mixed phase state where tetragonal, monoclinic and rhombohedral phases co-exist. The polarization of the PMN-PT crystal is generated by the displacement of the Ti, Mg, Nb atoms at the center of oxygen octahedron in the B sites of the perovskite structure. Single crystal PMN-PT materials are grown at much higher temperatures ($\gtrsim$ 1000 K) than the Burns temperature (T_B) and cooled down to lower temperatures, passing through the critical temperatures T_B and T_C. During this process, the crystal structure and the electric polarization properties change with the other material properties. Upon cooling through T_C the crystal structure changes from the higher symmetry cubic phase to lower symmetries depending on the PT content. It is evident in the literature that T_C for PMN-PT depends on the PT content. The morphotropic phase diagram (Figure 4.4) of PMN-PT shows a linear relationship between T_C and PT content, and for $T_C \approx 400$ K near MPB on the rhombohedral side of the morphotropic phase diagram [82, 104]. Many authors have reported that more than one structural phase transition can occur in the material near MPB. Araujo et al. [6] and Noheda et al. [82] have reported a coexistence of monoclinic and tetragonal phases in the MPB of PMN-PT below T_C, and a phase transition sequence of monoclinic $\rightarrow$ tetragonal $\rightarrow$ cubic with increasing the temperature. Noheda et al. report a phase transition sequence of rhombohedral $\rightarrow$ tetragonal $\rightarrow$ cubic for the PMN-PT solid solutions near MPB.

Polarization in unpoled or depoled PMN-PT crystal in cubic and tetragonal phases can be aligned parallel to one of the crystal axes, and in the rhombohedral and monoclinic

phases it can be rotated with the inclinations of the crystal structure. When the crystals are externally poled along definite directions, these possible polarization orientations become limited. Polarization rotation is being studied as the cause of the exceptional electromechanical properties. Some studies reported that creating deformations in a PMN-PT crystal can lead to changes in the crystal structure, and to rotations of the polarization [34, 71].

Spontaneous polarization in ferroelectrics is a result of electric dipoles uniformly aligned across domains. Walls separating these domains coincide with abrupt change in the dipole orientations. Depending on the domain size, domain structures in a ferroelectric material can be identified by microscopic observations. When an electric field is applied, these domains can reorient, resulting in a change in the net polarization of the material. Existence of domains and their interactions upon dynamically applied strains can be accounted as a prominent mechanism of higher acoustic loss observed in the temperature below and close to T_c. When a ferroelectric material undergoes ferroelectric $\rightarrow$ paraelectrcic transition, these domains are depleted, and randomly oriented polar regions come into existence. At temperatures above T_c, PMN-PT transitions into a para-electric phase with cubic crystal symmetry, and randomly oriented polar nano-regions (PNR) exist in the material[17, 80]. PNRs substantially affect the crystal properties of the material [6, 11]. These PNRs are depleted above the Burns temperature (T_B) which turns into a completely paraelectrcic material. T_B of PMN-PT appeared far above the ferroelectric phase transition temperature (T_C). Wongmaneerung et al. reports that $T_B \approx 623K$ for PMN [111], and Ko et al. suggested that the addition of PT does not have a substantial effect on T_B [59].

Peak temperature locations of $C_{11},\ C_{44},\ A_Z$ and Q^{-1} based on the RUS measurement of two different samples are shown in Table 4.3. Considering the temperature ranges in which these peaks occur, it is suggested that these peaks are related to the paraelectric transition at T_B. It can also be argued that the drastic stiffening upon heating the material from $\approx$ 400 K up to 700 K is a combined result of the presence of PNRs in the material above T_C that are depleting with the temperature and the emergence of the pseudo-cubic crystallines [9, 82, 95]

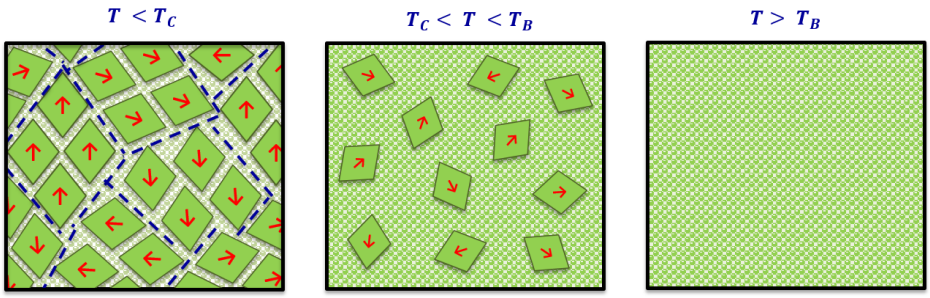

Figure 4.27. (a) Ferrorelectic domains below T_C (b) Existence of PNRs below Burns temperature (c) paraelectrcic transition above T_B [1, 33].

that are in the process of transitioning to cubic symmetry at higher temperatures. The fact that elastic anisotropy A_Z exponentially decayed in the temperature range starting from ≈ 3.2 reaching the minimum of ≈ 1.55 ($A_Z = 1$ for perfectly isotropic material) also support the above argument on pseudo-cubic crystallines. Taking our experimental observations on temperature dependent elastic constants and other measurements reported on PMN-PT, we can reason the occurrences in peaks of elastic constant measurements do not evident a structural phase transition, but a significant change in material at this temperature.

Upon heat cycling, a higher temperature stability with no temperature hysteresis (Figure 4.8) effects on elastic properties of unpoled PMN-PT are observed in this study, suggesting the physical process the material underwent with heating is a reversible process.

The analysis on pressure effects on resonances in the range of near vacuum to 500 psi (≈ 3.45 MPa) showed that a higher pressure sensitivity on resonances when the material is at the temperature range creates the rising edge of the transition ($\approx$ 400 K - 500 K). This pressure sensitivity is increased towards the lower temperatures in this range. Figures 4.16 and 4.17 show the temperature and pressure effects on resonances in terms of the frequency shifts and changes in the quality factor. A definite conclusion on this effect is not made by this study, but it is suggested that the applied hydrostatic pressure pushes the material to a stiffer state in this transition temperature region by forcing the pseudo-cubic crystallines to become cubic. It is evident that this process is reversible under pressure loading and

unloading on the material. Figure 4.18(a) shows a comparison resonance frequency of mode 5 under vacuum, taken at initial temperature, to the resonance frequency of mode 5 during the pressure cycles in each isotherm. The mode frequencies showed no hysteresis in multiple loading and unloading cycles in four different isotherms.

CHAPTER 5

ELASTIC PROPERTIES OF THERMOELECTRIC MATERIALS

This chapter will give a general introduction to thermoelectric materials, their applications and experimental findings on high temperature elastic properties of three different thermoelectric materials: lead tellurides, lanthanide filled skutterudites, and rare-earth doped strontium titanate.

5.1 Thermoelectric effect

The thermoelectric effect is the phenomenon of direct conversion of temperature gradient to an electric potential and vice versa. The materials exhibiting a strong thermoelectric effect are classified as thermoelectric materials. When a temperature difference is applied across a thermoelectric material device, a potential is created. This phenomenon was first discovered by the German physicist Thomas Johann Seebeck in 1821. In 1834, the French physicist Jean Charles Athanase Peltier discovered the inverse phenomena of generating a temperature difference by applying an electric potential across a thermoelectric material device. These two connected phenomenon are known as the Peltier-Seebeck effect.

The Seebeck coefficient is a measure of the magnitude of an induced thermoelectric voltage in response to a temperature difference across that material. It is defined as $S = \lim_{\Delta T \to 0} \frac{\Delta V}{\Delta T} = \frac{dV}{dT}$. The Seebeck coefficient gives an indication of the expected thermoelectric performance of a material. The Peltier coefficient is defined as $\Pi = ST$. More details on the thermoelectric materials can be found in the review article by C. Wood [112], in the CRC handbook of thermoelectrics [97], and in Nolas et al. [83].

The thermoelectric figure of merit is $zT = \frac{\sigma S^2 T}{\kappa_T}$, where κ_T is the total thermal conductivity and σ is electrical conductivity. The total thermal conductivity is a combination

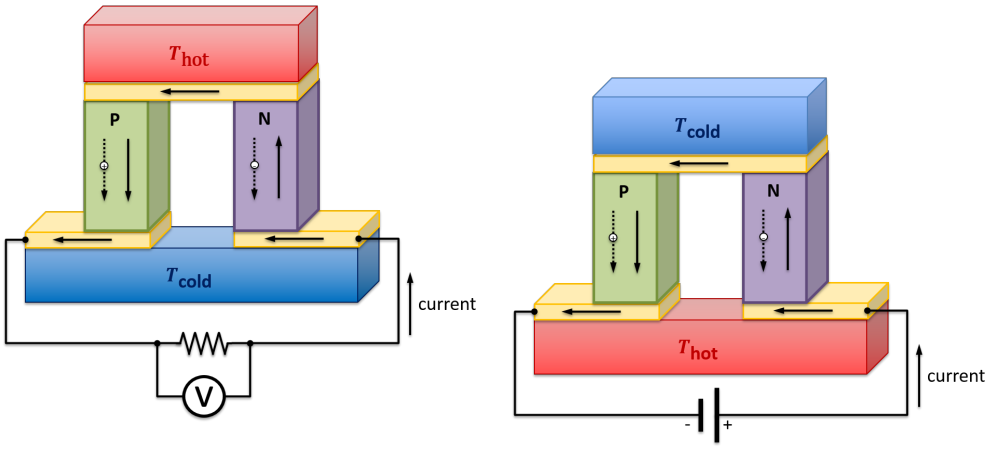

Figure 5.1. Seebeck Effect and Peltier Effect.

of electronic thermal conductivity and lattice (phononic) thermal conductivity, which can be expressed as $\kappa_T = \kappa_{electroctic} + \kappa_{phononic}$. The thermoelectric figure of merit can be increased by either increasing σ or decreasing κ_T. Electrical conductivity increases with the presence of more electrons in the material but, that will also increase electronic thermal conductivity which suppresses on the thermoelectric figure of merit. One challenge in developing novel thermoelectric materials with increased efficiency is understanding how to balance the contribution from $\kappa_{electroctic}$ and $\kappa_{phononic}$ on zT. This involves a lot of studies on lattice dynamics to gain insight into the material. A thorough knowledge in elastic constants is crucial for the research and development process. The thermoelectric materials are also subjected to elevated temperatures, and most of the cutting edge thermoelectric materials are classified for high temperature usage. Therefore understanding high temperature elastic properties of these materials will facilitate the path to discovering advanced thermoelectric materials.

5.2 Thermoelectric Materials and Devices

Thermoelectric materials are deployed in solid-state thermoelectric generators (TEGs), in which thermal gradients produce electric potentials following the Seebeck effect. TEGs are highly suited for waste heat recovery applications as well as to provide uninterrupted electric power to isolated and mission critical devices. TEGs are currently being considered

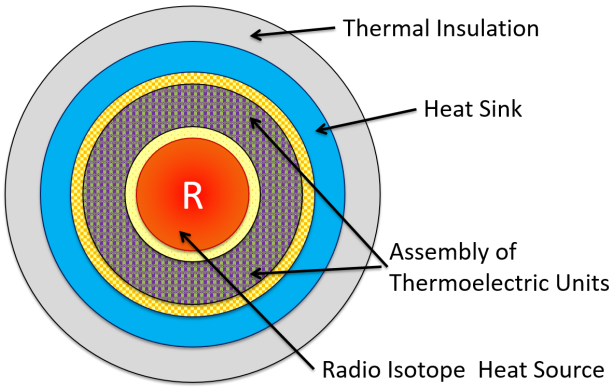

Figure 5.2. General schematic of the Radioisotope Thermoelectric Generator mostly used in deep space probes.

for many different terrestrial and extraterrestrial applications because of their higher durability and their longer life span as power sources . The Jet Propulsion Laboratory (JPL) is particularly interested in Radioisotope Thermoelectric Generators (RTGs) as power sources in deep-space probes. Thermoelectric silicon germanium (SiGe) and lead telluride (PbTe) alloys have been used in RTGs to power several US spacecrafts in the past few decades [92, 107, 112]. Another application of thermoelectric materials is thermoelectric cooling devices, which can be effective as cooling systems for isolated electronic devices, such as laser diodes and computer processors. Thermoelectric cooling is also applicable in some ordinary applications such as refrigeration and air conditioning.

As explained in the previous section, one of the crucial problems to solve in thermoelectric materials is increasing the thermoelectric efficiency, as quantified by the figure of merit (zT). This measure of efficiency is a temperature dependent quantity, making certain alloys better suited to high temperature ($>$ 1000 K) heat sources. Lanthanum telluride ($La_{3-\delta}Te_4$), along with silicon germanium and Zintl phase alloys, has shown peak efficiencies in this temperature range [72, 112]. In addition to the figure of merit and associated thermoelectric properties, temperature dependent mechanical properties are also critical in two ways in the thermoelectric materials research: (1) understanding the lattice dynamics for material enhancements; (2) Evaluating the reliability in application design to ensure the

stress and vibration levels remain safely below fracture loads and to predict response to thermal stress in the operation (e.g. to evaluate the reliability of TEG components deployed in spacecrafts during launch and cruise). The JPL has pioneered efforts to determine the thermo-mechanical properties of advanced TE materials [92, 107].

5.3 Lanthanum Telluride

5.3.1 Material Information

$La_{3-\delta}Te_4$ crystallizes in the cubic Th_3P_4 structure [72], accepting vacancies on the La position over the range from $0 \leq \delta \leq 1/3$. For $\delta = 0$, the compound is metallic, while for $\delta = 1/3$ the system is a charge-balanced semiconductor. The introduction of lanthanum vacancies and the corresponding reduction in free carrier concentration is essential for optimal thermoelectric efficiency, and also leads to a stiffening of the phonon spectrum [21]. In this study, we focus on compositions in the region of large thermoelectric performance ($zT > 1$ above 1100 K), as well as the vacancy-free composition. This work focuses on investigating the temperature dependence of the mechanical properties of bulk $La_{3-\delta}Te_4 + x\%$Ni materials, elastic moduli were measured between room temperature ($\approx$ 293 K) and 1273 K using the resonant ultrasound spectroscopy (RUS) technique.

5.3.2 Sample Preparation

All the lanthanum telluride materials used in this study were fabricated at the Jet Propulsion Laboratory using the spark plasma sintering (SPS) method. Details of the synthesis $La_{3-\delta}Te_4$ materials can be found in May et al. [72].

RUS samples were prepared and measured in the National Center for Physical Acoustics (NCPA) at the University of Mississippi. Precise rectangular parallelepiped samples were cut from the bulk materials using a diamond wheel saw, and the faces were polished using a lapping machine. $La_{3-\delta}Te_4$ samples are highly reacting with the atmospheric oxygen and water. This was identified in the early stage of this study, and the necessary steps were taken

to prevent exposure of the sample to water and to minimize heating in open air. However samples were exposed to oil based coolant in the cutter, and acetone and methanol in the cleaning. Sample dimensions were measured using a micrometer ($\pm 10^{-4}$ cm) and, masses ($\pm 10^{-4}$ g) were recorded for density calculations. Sample name, nominal composition, and the measured density of the samples are summarized in Table 5.1. Before proceeding with the high temperature measurements, samples were thoroughly cleaned in a methanol bath placed in a low power ultrasonic cleaner.

5.3.3 Elastic moduli versus Temperature

In this work, elastic moduli of three different lanthanum telluride alloys were determined in the temperature range of 293 K – 823 K. Figure 5.3 shows plots of the elastic moduli as a function of temperature. In general, the trends are quite linear as is typical for moduli at temperatures well above the Debye temperature (Θ_D), where May et al. [72] reported that the Θ_D of $La_{2.81}Te_4$ is 208 K and estimated Θ_D for the materials used in this work are within the range of 198 K – 216 K. Similarly, a softening of the phonon density of states with temperature has been previously observed and described by a quasi-harmonic model [21]. The solid lines are linear fits to the elastic moduli [Young's modulus (E) and Shear Modulus (G)] data for each sample. The coefficients of determination (R^2) for all E and G fits were larger than 0.99. Table 5.2 shows the slopes of these fits. None of these samples exhibited any hysteresis in the cooling cycling.

Temperature trends of the elastic moduli of the $La_{3-\delta}Te_4$+%xNi samples showed a notable increase in the stiffness with the nickel content, and it is consistent over the temperature range being investigated. Furthermore, this increase in the stiffness fitted in to a linear trend which is shown in the Figure 5.4. Table 5.4 shows the linear fit parameters of the elastic moduli as a function of percentage nickel content at room temperature and 800 K.

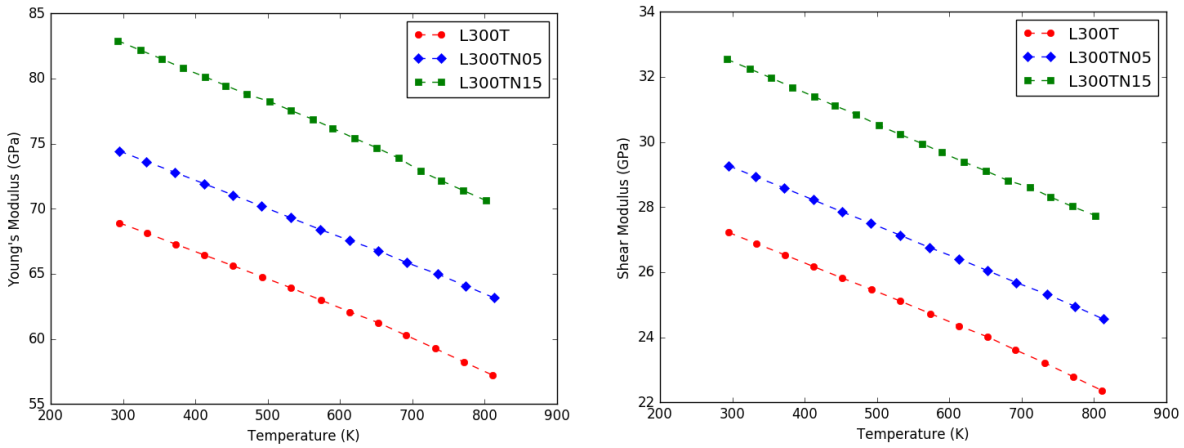

Figure 5.3. Youngs modulus (E) and shear modulus (G) of $La_{3-\delta}Te_4$+%xNi samples in the temperature range of room temperature ($\approx$ 295 K) to $\approx$ 700 K.

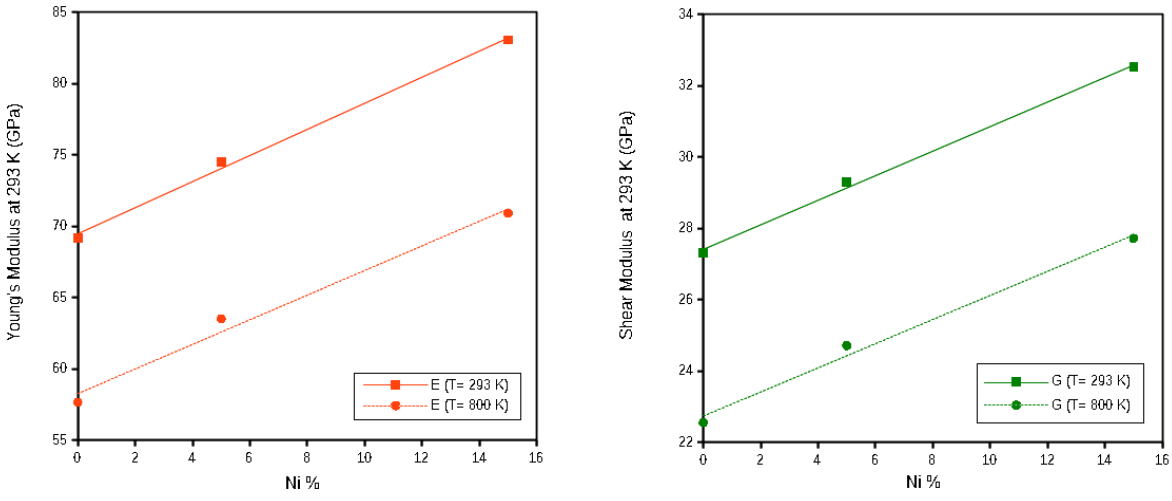

Figure 5.4. Elastic moduli of $La_{3-\delta}Te_4$+%xNi samples for different x values at 293 K and 800 K. See Table 5.4 for slopes of the linear fits to elastic moduli vs. Ni %

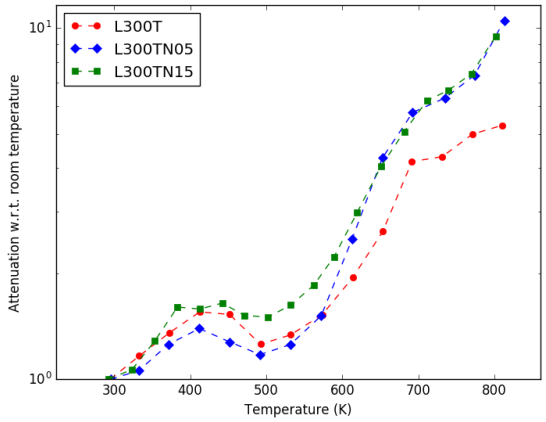

Figure 5.5. Temperature dependence of the acoustic attenuation of the $La_{3-\delta}Te_4$+%xNi with respect to the attenuation at room temperature. Note the log scale.

5.3.4 Acoustic Attenuation

The resonance peaks are clear and sharp with quality factor (Q) values in the 10,000 range at moderate temperatures. The agreement between computed and measured center peak frequencies was excellent ($\approx 0.15\%$ averaged over all modes). The attenuation in these $La_{3-\delta}Te_4$+%xNi samples exhibits rather interesting features. Trends of acoustic attenuation ($1/Q$) averaged over all modes for each sample scaled to the attenuation at the room temperature versus temperature for all the $La_{3-\delta}Te_4$+%xNi samples are shown in Figure 5.5. There is a consistent peak in the attenuation vs. temperature plots. The magnitude and the peak temperature are sample dependent but, within a narrow range of temperature. This feature does repeat upon thermal cycling. Repeatable peaks in attenuation are typically associated with thermally activated loss mechanisms such as an elastic dipole, which could have two different orientations or point defects hopping between vacant sites in the lattice. At temperatures above the peak, attenuation for all samples exhibit a typical rise with temperature increase. These attenuation curves are shown in Figure 5.5. The source of this loss mechanism is an open question.

Table 5.3 shows the Θ_D for each sample that was estimated following Anderson's method (see Chapter 2) using the room temperature elastic constants obtained from an

Table 5.1. Sample names, nominal compositions, of LaTe samples.

Sample	Composition	Measured Density (g/cm^3)
L300-SP	La_3Te_4	6.53
L300TN05-SP	$La_3Te_4 + 5\%Ni$	6.70
L300TN15-SP	$La_3Te_4 + 15\%Ni$	6.94

Table 5.2. Temperature derivatives of each modulus obtained from the linear fits to the direct contact data (293 – 800 K) and buffer-rod data (293 – 1273 K) in Figure 5.3 .

Sample	dE/dT (10^{-2} GPa/K)	E_0 (GPa)	dG/dT (10^{-3} GPa/K)	G_0 (GPa)
L300-SPS	−2.28	75.86	−9.38	30.05
L300N05-SPS	−2.16	80.83	−9.05	31.95
L300N15-SPS	−2.40	90.08	−9.48	35.30

RUS study.

5.4 Elastic constants of thermoelectric skutterudite materials (SKD) at elevated temperatures

5.4.1 Materials and Experimental Procedure

Skutterudite materials gained prominence in thermometric materials research as promising transduction materials for moderate temperature applications [49, 77]. Skutterudites have the general formula of MN_3, which resembles the structure of $CoAs_3$, a naturally occurring mineral found in Stuttered, Norway. The N sites in the general formula is filled

Table 5.3. Temperature derivatives of mean acoustic speed (v_m) and intercepts were obtained from linear fits to the v_m vs. temperature data points within the temperature range of 293 – 800 K in Figure 5.3.

Sample	dv_m/dT (m/s K)	v_{m0} (m/s)	$v_m(T = 293K)$ (m/s)	$v_m(T = 800K)$ (m/s)	Θ_D (K)
L300-SPS	−0.3603	2382	2276	2094	199
L300N05-SPS	−0.3196	2422	2328	2166	206
L300N15-SPS	−0.3138	2504	2412	2253	216

Table 5.4. Slopes of elastic moduli (M) to the nickel percentage (Ni %) obtained for the direct contact data at 293 K and 800 K in Figure 5.3. The coefficient of determination (R^2) of all fits were greater than 0.98

Elastic Moduli	$\frac{d[M(T=293K)]}{d[Ni\%]}$	M(T=293 K, Ni=0 %)	$\frac{d[M(T=800K)]}{d[Ni\%]}$	M(T=800 K, Ni=0 %)
E	0.9149	69.48	0.8631	58.27
G	0.3443	27.41	0.3385	22.74

by pnictogens such as Sb, Te, and Sn and the M sites are filled by transition metals such as Pd, Co, Fe, and Ni [49, 77]. Skutterudite materials have relatively low coefficient of thermal expansion, which makes the materials are geometrically stable in high temperature applications [83].

Two types of skutterudite materials each p and n, which were synthesized at the the JPL facility were investigated in this work. See Table 5.5 for chemical compositions. Samples received from the JPL were precut to a millimeter-scale rectangular parallelepiped; however samples were polished at the NCPA RUS lab to break the dimensional symmetry. Dry sandpaper on a lapping machine operated under slow speeds was used during the polishing process. Samples were not exposed to water and heating in open air was minimized because of the material's high reactivity with water and oxygen. Acetone and methanol solutions were used to clean samples after polishing. Samples were kept in airtight containers at all times. Reduced oxygen atmosphere ($<$ 4 ppm) was maintained in the RUS measurement chamber by having slow and constant flow of UHP argon gas.

Sample SKD1 was tested using the buffer rod system. Quality factors of the resonance peaks rapidly decreased for all the samples after about 500 K. The signal to noise ratio (SNR) of the resonance spectrum from the buffer rod system became lower with increasing temperatures because of the broadening of peaks from various loss mechanisms in the system. Samples SKD 2–4 were measured using the direct contact system to get a better SNR at the complete temperature range of interest. Masses of the samples were measured before and after the test to check for any mass changes due to possible oxidization or sublimation. No

Table 5.5. Chemical composition of the SKD material samples being studied

Sample ID	JPL Material ID	Composition (Percentage weight)
SKD-1 SKD-4	SKD-011-N SKD-018-N	Sb (75–85%) Fe(5–15%) Ce(1–10%) Ru(1–10%)
SKD-2 SKD-3	SKD-017-P SKD-015-P	Sb (75–85%) Te(20–30%) Co(10–15%) Ce(0.1–5%) Pd(0.1–5%)

significant mass change ($< 1\%$) was detected. However, the surfaces of all samples came out darkened after the thermal cycling. Thermal expansion coefficients for p-type SKD and n-type SKD, 13.3 ppm/K and 10.65 ppm/K (source: JPL) were used in the fitting algorithms to estimate the elastic moduli at elevated temperatures.

5.4.2 Analysis of Results

The thermal dependence of the SKD elastic properties exhibit typical softening behavior in the temperature regime. The agreement between computed and measured center peak frequencies was excellent (≈0.2% averaged over all modes). The absolute stiffness of the p-type was higher than the n-type over the temperature range measured. Elastic moduli (M) data were fitted with a quadratic equation $M(T) = aT^2 + bT + M_0$.

The resonance peaks are clear and sharp with Q values in the range of 6,000 – 10,000 at room temperature and were drastically decreased with increasing the temperature. The Q values lower than 100 could not be used to determine resonance frequencies with confidence. Therefore elastic constants were not available for temperatures higher than 673 K except SKD2, where the maximum obtained temperature data was 721 K. Relatively modest elastic softening ($\approx 10\%$) coupled with strong increases in attenuation ($\approx 30\%$) was also observed in the investigated temperature range.

Table 5.6. SKD Sample Dimensions

Sample	Length (cm)	Width (cm)	Depth (cm)	Mass (g)	Density (g/cm^3)
SKD-1	0.359	0.326	0.272	0.237	7.445
SKD-4	0.349	0.333	0.263	0.227	7.427
SKD-2	0.319	0.348	0.255	0.220	7.772
SKD-3	0.360	0.348	0.276	0.268	7.751

Table 5.7. Room Temperature Elastic modulus for SKD Samples

Sample	E (GPa)	G (GPa)	Poissons ratio (ν)	Fit error (%)
SKD-1 (N)	133.3	54.5	0.223	0.12
SKD-4 (N)	132.9	54.2	0.226	0.12
SKD-2 (P)	136.9	55.9	0.225	0.10
SKD-3 (P)	137.1	55.8	0.227	0.14

Table 5.8. Elastic moduli were interpreted as functions of temperature. E is Youngs modulus G is shear modulus. The coefficient of determination (R^2) of all fits were greater than 0.999.

	Sample	$a(10^{-5}$ GPa/K$^2)$	$b(10^{-2}$ GPa/K)	E_0 (GPa)
$E(T)$	SKD-1 (N)	-2.84	- 0.0374	136.0
	SKD-4 (N)	-3.03	0.141	135.2
	SKD-2 (P)	-0.398	- 2.62	145.1
	SKD-3 (P)	-1.42	- 1.62	142.9
$G(T)$	SKD-1 (N)	-1.26	- 0.0148	55.6
	SKD-4 (N)	-1.27	0.0504	55.2
	SKD-2 (P)	-0.170	- 1.15	59.4
	SKD-3 (P)	-0.338	- 0.961	58.7

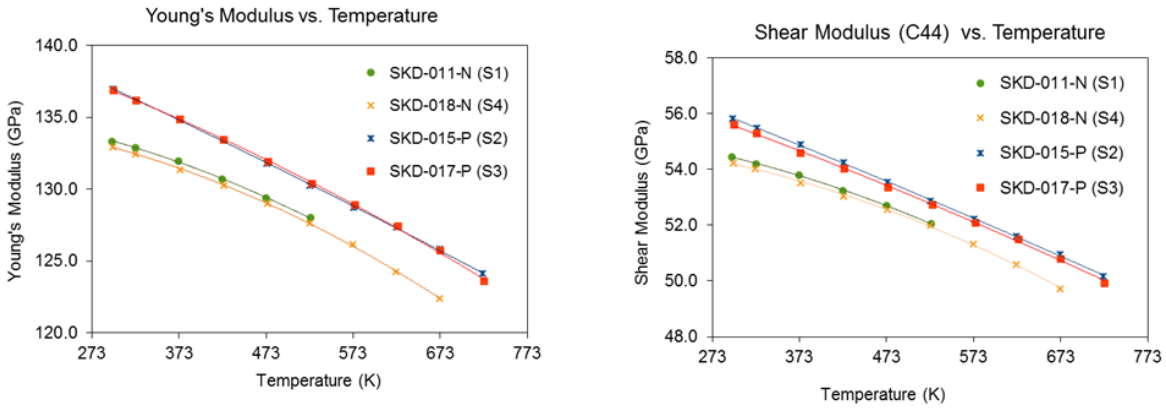

Figure 5.6. Youngs modulus (E) and shear modulus (G) in the temperature range of room temperature (≈295 K) to ≈700 K for samples SKD 1–4. Solid lines are quadratic fits.

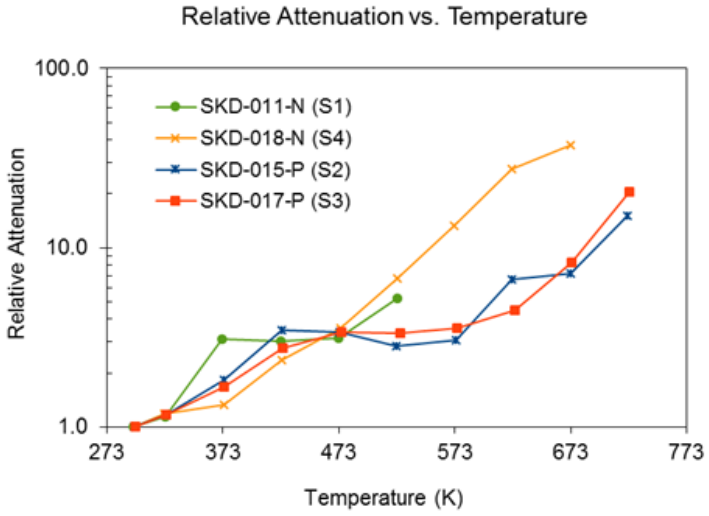

Figure 5.7. Average factor and relative attenuation (normalized to room temperature values) in the temperature range of room temperature (≈295 K) to ≈700 K for samples SKD 1–4. Note the log scale.

Table 5.9. Room Temperature Elastic modulus for STO Samples

Sample	Density (g cm^{-3})	E (GPa)	G (GPa)	Fit error (%)
STO	5.00	268.7	109.0	0.1
STO-Y1	5.01	245.9	97.8	0.19
STO-Y2	5.00	253.6	102.2	0.77
STO-Pr1	5.13	222.2	85.8	0.70
STO-Pr2	5.08	202.1	82.9	0.52

5.5 High Temperature Elastic Constants of Rare-earth Doped Strontium Titanate

Strontium titanate is a stable material at higher temperatures. "The crystal structure of pure $SrTiO_3$ exhibits a cubic lattice ($a \approx 3.905$) $Pm\bar{3}m$ space group at room temperature, with a unit cell consisting of a titanium-oxygen (TiO_6) octahedron enclosed in a cube of Sr atoms." [10]. Strontium titanate exhibits n-type semiconducting behavior when it is donor-doped. It also exhibits high electronic and ionic conductivities upon reduction or doping. Depending on the doping conditions, these materials can be used as a thermoelectric material, solid oxide fuel cell (SOFC) anode or oxygen extraction membrane. As a thermoelectric material, the thermoelectric figure of merit can be increased by decreasing the lattice component of thermal conductivity under different doping conditions.

It is important to understand elastic properties of these materials at elevated temperatures to assist the studies of enhancing material properties as well as design and development engineering applications where the material will be exposed to higher temperatures during the device operation. The temperature dependence of the elastic constants of spark plasma sintered polycrystalline rare-earth doped strontium titanate (STO) [$Sr_{0.9}X_{0.1}TiO_{3-\delta}$, (X=Pr, Y)] was investigated in the temperature range of 300 K – 750 K using resonant ultrasound spectroscopy methodologies. Specimens used in this study were millimeter scale rectangular parallelepiped samples from each material precisely cut and polished. Room temperature elastic moduli of STO samples are shown in Table 5.9.

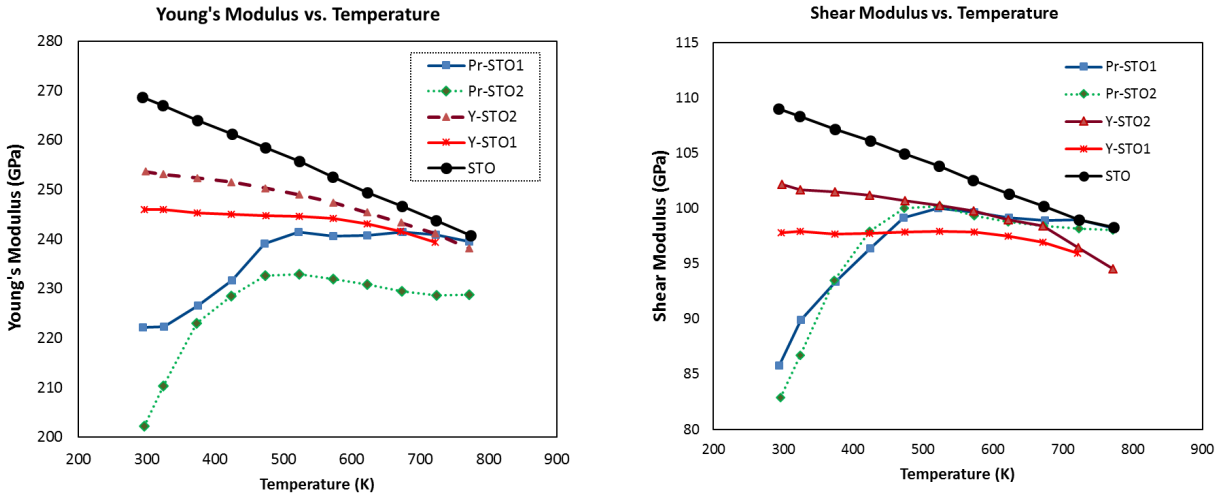

Figure 5.8. Elastic moduli of $Sr_{0.9}X_{0.1}TiO_{3-}$, (X=Pr, Y) in the temperature range of room temperature ($\approx$ 295 K) to $\approx$ 750 K

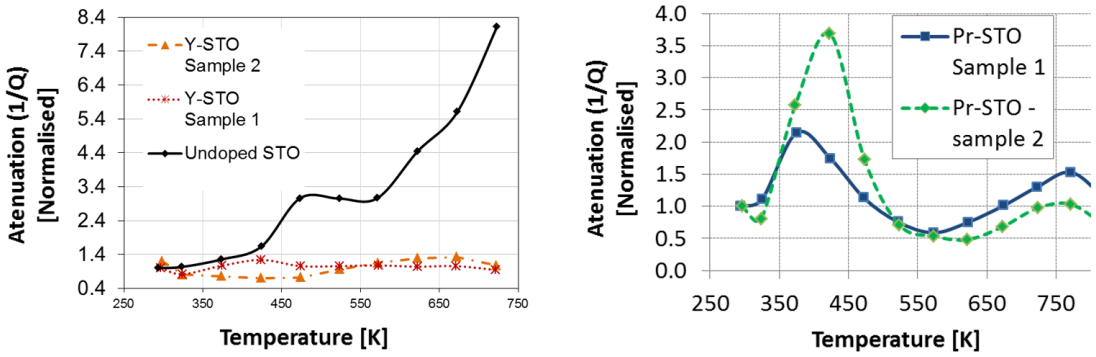

Figure 5.9. Relative attenuation (normalized to room temperature values) in the temperature range of room temperature ($\approx$ 295 K) to $\approx$ 750 K for $Sr_{0.9}X_{0.1}TiO_{3-}$ (X=Pr, Y) and undoped STO

5.5.1 Analysis of Results

Elastic constants of pure STO decrease linearly, indicating typical softening with increased temperature. Yttrium (Y) doped STO also exhibits a monotonic softening, except with a pronounced curvature in this high temperature regime. Trends of elastic constants of the praseodymium (Pr) doped STO show a non-monotonic stiffing from room temperature up to 475 K, followed by a gradual softening. Changes in attenuation were quantified by the inverse quality factor ($1/Q$) averaged over measured resonances. Undoped STO showed a monotonic gradual increase of attenuation with increasing temperature while yttrium doped STO showed little variation. In contrast, attenuation of Pr doped STO exhibited a peak around 425 K. The stiffening behavior and the attenuation peak around 400 K of the Pr doped STO is an open question.

CHAPTER 6

CONCLUSION

The primary goal of the work presented in this dissertation was to investigate temperature and pressure dependence of the elastic properties of relaxor ferroelectric PMN-PT material to gain insight into phase transitions of the material. This dissertation also address investigating temperature dependent elastic properties of selected novel thermoelectric materials and a lead-free solder alloys. High-Temperature and High-Pressure Resonant Ultrasound Spectroscopy (HTP-RUS) methodologies were successfully applied for investigating relaxor ferroelectric and thermoelectrics. Ultrasonic pulse echo technique was used in investigating the lead-free solder material. A brief introduction to the theory of elasticity, implementation of RUS methodologies, pulse-echo technique are included in Chapter 2 to give some background information to the work covered in the subsequent chapters.

The work on PMN-PT provides a detailed study of high temperature elastic constants and acoustic attenuation of a material that can be used as a foundation for future work on studying the family of lead titanate based relaxor ferroelectrics. Our findings of this study lead us to a number of studies in the immediate future. The argument on local polarization effects on that stiffening characteristics to be verified by conducting RUS study, placing the sample under a variable electric field. This can be achieved by adding an electric field as a control parameter in the RUS cell as shown in Figure 6.1. The argument on interactions of ferroelectric domains serve as the acoustic energy loss mechanism at the temperatures below T_C is to be verified by conducting PE measurements to determine the temperature dependent acoustic attenuation of PMN-PT samples cut in different crystal orientations and poled in different directions. Measured sound speeds at room temperature for the PMN-PT material

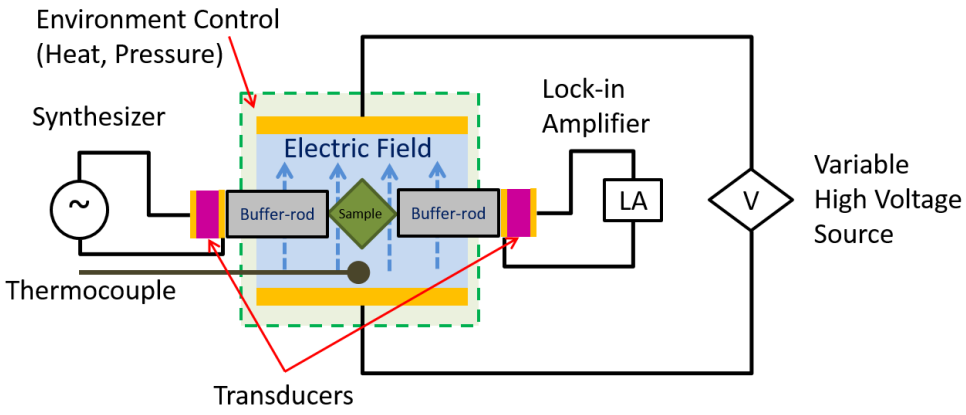

Figure 6.1. Addition of electric field as a control parameter in the RUS cell to investigate polarization effects on elastic properties of ferroelectric materials.

sample along the [001] direction used to determine the two elastic constants C_{33} and C_{44}. Conducting the measurements on PMN-PT samples cut in different crystal orientations, as well as measuring sound speed along different directions in each sample can lead to finding the complete set of elastic constants at lower temperatures below T_C, where we were unable to apply RUS methodologies in that temperature range due to elevated acoustic attenuation. Theoretical and experimental details on PE technique is discussed under Sections 2.3 and 2.7.

The scope of the study can be broadened by investigating properties of PMN-PT with different PT contents. that will lead to the original goal of this dissertation of mapping out the temperature and pressure phase diagram of PMN-PT. The pressure range (0 psi – 500 psi) explored in this study was high enough to induce a phase transition. However, an unusual stiffening mechanism is observed at the temperatures close to the phase transition temperature. A definite conclusion on this observation can be drawn by conducting more experiments and comparing results with theoretical models.

RUS methodologies have been applied in measuring novel elastic properties of thermoelectric materials at elevated temperatures. The results of RUS measurements on lanthanum tellurides and skutterudites covered in this work will be of use in the development of advanced thermoelectric generates (TEGs) by NASA's Jet Propulsion Laboratory. High temperature

elastic constant measurements on rare earth doped strontium titanates were used to determine effect of yttrium (Y) doping on the elastic moduli of $Sr_{1-x}Y_xTiO_{3-\delta}$ ($x = 0,\ 0.1$) included in the publication "Role of phonon scattering by elastic strain field in thermoelectric $Sr_{1-x}Y_xTiO_{3-\delta}$" in the year of 2014 [10]. Results for high temperature elastic constants of the praseodymium (Pr) doped material are to be published in the future. These studies also show the importance and power of RUS methodologies in novel material research because of the need of significantly small portion of material.

When RUS methodologies are not applicable, e.g. on materials with a higher acoustic loss, polycrystalline materials or metals with substantially large grain structure, the pulse-echo (PE) technique can be used to determine elastic moduli from measured sound speeds. This was applied in measuring sound speeds of PMN-PT at the room temperature, where resonance measurements were not possible due to high acoustic attenuation. Appendix A covers an application of PE in measuring of the temperature dependence of elastic moduli of SnAgCu solder alloy under different aging conditions. This work provided material information for understanding the reliability issues in lead-free solders.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Ferroelectrics: Nanoregions team together. *NPG Asia Materials*, feb 2010. ISSN 1884-4049. doi: 10.1038/asiamat.2010.17. URL `http://www.nature.com/doifinder/10.1038/asiamat.2010.17`.

[2] R. Adebisi. *High temperature and pressure resonant ultrasound spectroscopy study of metal hydride systems*. PhD thesis, The University of Mississippi, 2011.

[3] M. Ahart, M. Somayazulu, R. E. Cohen, P. Ganesh, P. Dera, H.-k. Mao, R. J. Hemley, Y. Ren, P. Liermann, and Z. Wu. Origin of morphotropic phase boundaries in ferroelectrics. *Nature*, 451(7178):545–8, jan 2008. ISSN 1476-4687. doi: 10.1038/nature06459.

[4] J. L. S. Albert Migliori. *Resonant Ultrasound Spectroscopy: Applications to Physics, Materials Measurements, and Nondestructive Evaluation*. Wiley-VCH, 1997. ISBN 9780471123606.

[5] O. L. Anderson. A simplified method for calculating the debye temperature from elastic constants. *Journal of Physics and Chemistry of Solids*, 24(7):909–917, jul 1963. ISSN 00223697. doi: 10.1016/0022-3697(63)90067-2.

[6] E. Araújo. Recent advances in processing, structural and dielectric properties of pmn-pt ferroelectric ceramics at compositions around the mpb. 2011.

[7] J. Axelson. *Serial Port Complete : COM Ports, USB Virtual COM Ports, and Ports for Embedded Systems*. Lakeview Research., 2007.

[8] P. Bao, F. Yan, W. Li, Y. R. Dai, H. M. Shen, J. S. Zhu, Y. N. Wang, H. L. W. Chan, and C. L. Choy. Mechanical properties related to the relaxor-ferroelectric phase transition of titanium-doped lead magnesium niobate. *Applied Physics Letters*, 81(11): 2059–2061, 2002. ISSN 00036951. doi: 10.1063/1.1498498.

[9] L. Bellaiche. Piezoelectricity of ferroelectric perovskites from first principles. *Current Opinion in Solid State and Materials Science*, 6(1):19–25, 2002. ISSN 13590286. doi: 10.1016/S1359-0286(02)00017-7.

[10] S. Bhattacharya, A. M. Dehkordi, S. Tennakoon, R. Adebisi, J. R. Gladden, T. Darroudi, and T. M. Tritt. Role of phonon scattering by elastic strain field in thermoelectric Sr1xYxTiO3δ. *Journal of Applied Physics*, 115:1–23, 2014. doi: 10.1063/1.4882377.

[11] a. a. Bokov and Z.-G. Ye. Recent progress in relaxor ferroelectrics with perovskite structure. *Journal of Materials Science*, 41(1):31–52, jan 2006. ISSN 0022-2461. doi: 10.1007/s10853-005-5915-7.

[12] M. Born and K. Huang. *Dynamical theory of crystal lattices.* The International series of monographs on physics. Oxford : Clarendon Press, 1954., 1954.

[13] J. J. Brondijk, K. Asadi, P. W. M. Blom, and D. M. De Leeuw. Physics of organic ferroelectric field-effect transistors. *Journal of Polymer Science, Part B: Polymer Physics*, 50(1):47–54, 2012. ISSN 08876266. doi: 10.1002/polb.22363.

[14] A. E. Brown. Rationale and summary of methods for determining ultrasonic properties of materials at lawrence livermore national laboratory. Technical report, 1997.

[15] H. Cao, V. H. Schmidt, R. Zhang, W. Cao, and H. Luo. Elastic, piezoelectric, and dielectric properties of 0.58pb(mg1/3nb2/3)o3-0.42pbtio3 single crystal. *Journal of Applied Physics*, 96(1):549, 2004. ISSN 00218979. doi: 10.1063/1.1712020.

[16] J. L. B. Charles H. Sherman. *Transducers and Arrays for Underwater Sound.* The Underwater Acoustics Series. Springer-Verlag New York, 2007.

[17] X.-Q. Chen, H. Niu, D. Li, and Y. Li. Modeling hardness of polycrystalline materials and bulk metallic glasses. *Intermetallics*, 19(9):1275–1281, sep 2011. ISSN 09669795. doi: 10.1016/j.intermet.2011.03.026.

[18] D. Connétable and O. Thomas. First-principles study of the structural, electronic, vibrational, and elastic properties of orthorhombic nisi. *Physical Review B*, 79(9): 094101, 2009. ISSN 1098-0121. doi: 10.1103/PhysRevB.79.094101.

[19] N. I. Corporation. National instruments visa @Online, 2014. URL `http://www.ni.com/visa/`.

[20] R. a. Cowley. Structural phase transitions i. landau theory. *Advances in Physics*, 29 (1):1, 1980. ISSN 0001-8732. doi: 10.1080/00018738000101346.

[21] O. Delaire, A. F. May, M. A. McGuire, W. D. Porter, M. S. Lucas, M. B. Stone, D. L. Abernathy, V. A. Ravi, S. A. Firdosy, and G. J. Snyder. Phonon density of states and heat capacity of la3xte4. *Physical Review B*, 80(18):184302, nov 2009. ISSN 1098-0121. doi: 10.1103/PhysRevB.80.184302.

[22] H. Demarest. Cube resonance method to determine the elastic constants of solids. *The Journal of the Acoustical Society of America*, 49(3), 1971.

[23] E. I. A. E. Department. *Interface between Data Terminal Equipment and Data Communication Equipment Employing Serial Binary Data Interchange.* Electronic Industries Association, Engineering Dept., 1969.

[24] W. D. Dong, P. Finkel, A. Amin, and C. S. Lynch. Giant electro-mechanical energy conversion in [011] cut ferroelectric single crystals. *Applied Physics Letters*, 100(4): 1–4, 2012. ISSN 00036951. doi: 10.1063/1.3679644.

[25] Fan He. Usb port and power delivery: An overview of usb port interoperabiliy. In *2015 IEEE Symposium on Product Compliance Engineering (ISPCE)*, pages 1–5. IEEE, may 2015. ISBN 978-1-4799-5930-3. doi: 10.1109/ISPCE.2015.7138710.

[26] M. Fine, L. Brown, and H. Marcus. Elastic constants versus melting temperature in metals. *Scripta Metallurgica*, 18(9):951 – 956, 1984. ISSN 0036-9748. doi: http://dx.doi.org/10.1016/0036-9748(84)90267-9.

[27] P. Finkel, K. Benjamin, and A. Amin. Large strain transduction utilizing phase transition in relaxor-ferroelectric pb(in1/2nb1/2)o3pb(mg1/3nb2/3)o3pbtio3 pb(in1/2nb1/2)o3pb(mg1/3nb2/3)o3pbtio3 single crystals. *Applied Physics Letters*, 98(19):192902, 2011. ISSN 00036951. doi: 10.1063/1.3585088.

[28] I. Foundation. Ivi specifications @Online, 2016. URL `http://www.ivifoundation.org/specifications/default.aspx`.

[29] P. S. Foundation. Pyvisa 1.8 : Python visa bindings for gpib, rs232, and usb instruments @Online, 2015. URL `https://pypi.python.org/pypi/PyVISA`.

[30] P. S. Foundation. Python @Online, 2015. URL `https://www.python.org/`.

[31] J. Fousek. Joseph valasek and the discovery of ferroelectricity. In *Proceedings of 1994 IEEE International Symposium on Applications of Ferroelectrics*, pages 1–5. IEEE, 1994. ISBN 0-7803-1847-1. doi: 10.1109/ISAF.1994.522283.

[32] J. Frantti, Y. Fujioka, and R. M. Nieminen. Pressure-induced phase transitions in pbtio3 : A query for the polarization rotation theory. *The Journal of Physical Chemistry B*, 111(17):4287–4290, may 2007. ISSN 1520-6106. doi: 10.1021/jp0713209.

[33] D. Fu, H. Taniguchi, M. Itoh, S. Y. Koshihara, N. Yamamoto, and S. Mori. Relaxor pb(mg1/3nb2/3)o3: A ferroelectric with multiple inhomogeneities. *Physical Review Letters*, 103(20), 2009. ISSN 00319007. doi: 10.1103/PhysRevLett.103.207601.

[34] H. Fu and R. Cohen. Polarization rotation mechanism for ultrahigh electromechanical response in single-crystal piezoelectrics. *Nature*, 403(6767):281–3, jan 2000. ISSN 1476-4687. doi: 10.1038/35002022.

[35] M. Ghasemifard, S. M. Hosseini, and G. H. Khorrami. Synthesis and structure of pmn-pt ceramic nanopowder free from pyrochlore phase. *Ceramics International*, 35 (7):2899–2905, 2009. ISSN 02728842. doi: 10.1016/j.ceramint.2009.03.036.

[36] R. Gilbert. The general-purpose interface bus. *IEEE Micro*, 2(1):41–51, feb 1982. ISSN 0272-1732. doi: 10.1109/MM.1982.290948.

[37] L. A. Girifalco and V. G. Weizer. Application of the morse potential function to cubic metals. *Physical Review*, 114(3):687–690, 1959. ISSN 0031899X. doi: 10.1103/PhysRev.114.687.

[38] J. R. Gladden, III. *Characterization of the thin films and novel materials using resonant ultrasound spectroscopy*. PhD thesis, The Pennsylvania State University, 2003.

[39] R. Golesorkhtabar, P. Pavone, J. Spitaler, P. Puschnig, and C. Draxl. Elastic: A tool for calculating second-order elastic constants from first principles. *Computer Physics Communications*, 184(8):1861–1873, aug 2013. ISSN 00104655. doi: 10.1016/j.cpc.2013.03.010.

[40] Y. Guo, H. Luo, D. Ling, H. Xu, T. He, and Z. Yin. The phase transition sequence and the location of the morphotropic phase boundary region in (1 x)[pb (mg 1/3 nb 2/3)o 3] xpbtio 3 single crystal. *Journal of Physics: Condensed Matter*, 15(2):L77–L82, jan 2003. ISSN 0953-8984. doi: 10.1088/0953-8984/15/2/110.

[41] Z. Guo, A. P. Miodownik, N. Saunders, and J. P. Schillé. Influence of stacking-fault energy on high temperature creep of alpha titanium alloys. *Scripta Materialia*, 54(12): 2175–2178, 2006. ISSN 13596462. doi: 10.1016/j.scriptamat.2006.02.036.

[42] A. Hilczer, M. Szafrański, A. Bokov, and Z.-G. Ye. Effect of Hydrostatic Pressure on the Dielectric Properties of PMN-0.31PT Single Crystal. *Ferroelectrics*, 339(1):75–84, sep 2006. ISSN 0015-0193. doi: 10.1080/00150190600738030.

[43] R. Holland. Resonant properties of piezoelectric ceramic rectangular parallelepipeds. *The Journal of the Acoustical Society of America*, 43(5), 1968.

[44] S. Horiuchi and Y. Tokura. Organic ferroelectrics. *Nature Materials*, 7(5):357–366, may 2008. ISSN 1476-1122. doi: 10.1038/nmat2137.

[45] Y.-J. Hu, S.-L. Shang, Y. Wang, K. A. Darling, B. G. Butler, L. J. Kecskes, and Z.-K. Liu. Effects of alloying elements and temperature on the elastic properties of w-based alloys by first-principles calculations. *Journal of Alloys and Compounds*, 671 (5):267–275, jun 2016. ISSN 09258388. doi: 10.1016/j.jallcom.2016.02.018.

[46] IEEE. Ieee 802.3: Ethernet @Online, 2012. URL `http://standards.ieee.org/about/get/802/802.3.html`.

[47] U. I. F. Inc. Universal serial bus @Online, 2016. URL `http://www.usb.org/home`.

[48] D. G. Isaak. Elasticity of fused silica spheres under pressure using resonant ultrasound spectroscopy. *The Journal of the Acoustical Society of America*, 104(4):2200, oct 1998. ISSN 00014966. doi: 10.1121/1.423733.

[49] A. B. J. P. Fleurial, T. Caillat. Skutterudites: an update. In *Thermoelectrics, 1997. Proceedings ICT '97. XVI International Conference on*, pages 1–11, Aug 1997. doi: 10.1109/ICT.1997.666968.

[50] M. Jamal, S. Jalali Asadabadi, I. Ahmad, and H. Rahnamaye Aliabad. Elastic constants of cubic crystals. *Computational Materials Science*, 95:592–599, dec 2014. ISSN 09270256. doi: 10.1016/j.commatsci.2014.08.027.

[51] C. Jasiukiewicz and V. Karpus. Debye temperature of cubic crystals. *Solid State Communications*, 128(5):167–169, 2003. ISSN 00381098. doi: 10.1016/j.ssc.2003.08.008.

[52] J. Jia, J. Kuang, Z. He, and J. Fang. Design of automated test system based on gpib. In *2009 9th International Conference on Electronic Measurement & Instruments*, pages 1–943–1–948. IEEE, aug 2009. ISBN 978-1-4244-3863-1. doi: 10.1109/ICEMI.2009.5274384.

[53] J. E. Jones. On the determination of molecular fields. ii. from the equation of state of a gas. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 106(738):463–477, oct 1924. ISSN 1364-5021. doi: 10.1098/rspa.1924.0082.

[54] G. Kaplan, T. W. Darling, and K. R. McCall. Resonant ultrasound spectroscopy and homogeneity in polycrystals. *Ultrasonics*, 49(1):139–142, 2009. ISSN 0041624X. doi: 10.1016/j.ultras.2008.08.001.

[55] D. Kim, S.-L. Shang, and Z.-K. Liu. Effects of alloying elements on elastic properties of Ni by first-principles calculations. *Computational Materials Science*, 47(1):254–260, nov 2009. ISSN 09270256. doi: 10.1016/j.commatsci.2009.07.014.

[56] H. King, S. Ferguson, D. Waechter, and S. Prasad. An x-ray diffraction study of pmnpt ceramics near the morphotropic phase boundary. In *Proceedings of ICONS 2002. International Conference on Sonar Sensors and Systems.*

[57] N. Kirova and S. Brazovskii. Electronic ferroelectricity in carbon based materials. *Synthetic Metals*, pages 1–18. ISSN 03796779. doi: 10.1016/j.synthmet.2015.10.015.

[58] K. Kliemt and C. Krellner. Crystal growth by Bridgman and Czochralski method of the ferromagnetic quantum critical material YbNi4P2. *Journal of Crystal Growth*, 449: 129–133, 2016. ISSN 00220248. doi: 10.1016/j.jcrysgro.2016.05.042.

[59] J.-H. Ko, D. Kim, S. Tsukada, S. Kojima, A. Bokov, and Z.-G. Ye. Crossover in the mechanism of ferroelectric phase transition of pb[(mg_{1/3}nb_{2/3})_{1x}ti_{x}]o_{3} single crystals studied by brillouin light scattering. *Physical Review B*, 82(10):1–7, 2010. ISSN 1098-0121. doi: 10.1103/PhysRevB.82.104110.

[60] L. D. Landau and E. M. Lifshits. *Theory of elasticity (Translated from the Russian by J.B. Sykes and W.H. Reid).* Addison-Wesley physics books. London, Pergamon Press; Reading, Mass., Addison-Wesley Pub. Co., 1959., 1959.

[61] H. Ledbetter and A. Migliori. A general elastic-anisotropy measure. *Journal of Applied Physics*, 100(6):1–6, 2006. ISSN 00218979. doi: 10.1063/1.2338835.

[62] R. Leisure, K. Foster, J. Hightower, and D. Agosta. Internal friction studies by resonant ultrasound spectroscopy. *Materials Science and Engineering: A*, 370(1-2):34–40, apr 2004. ISSN 09215093. doi: 10.1016/j.msea.2003.08.070.

[63] R. G. Leisure and F. A. Willis. Resonant ultrasound spectroscopy. *Journal of Physics: Condensed Matter*, 9(28):6001–6029, jul 1997. ISSN 0953-8984. doi: 10.1088/0953-8984/9/28/002.

[64] Z. A. D. Lethbridge, R. I. Walton, A. S. H. Marmier, C. W. Smith, and K. E. Evans. Elastic anisotropy and extreme poisson's ratios in single crystals. *Acta Materialia*, 58 (19):6444–6451, 2010. ISSN 13596454. doi: 10.1016/j.actamat.2010.08.006.

[65] B. Li, J. Kung, W. Liu, and R. C. Liebermann. Phase transition and elasticity of enstatite under pressure from experiments and first-principles studies. *Physics of the Earth and Planetary Interiors*, 228:63–74, 2014. ISSN 00319201. doi: 10.1016/j.pepi.2013.11.009.

[66] G. Li. *High temperature resonant ultrasound spectroscopy studies of thermoelectrics and other novel materials*. PhD thesis, The University of Mississippi, 2010.

[67] G. Li and J. R. Gladden. High Temperature Resonant Ultrasound Spectroscopy: A Review. *International Journal of Spectroscopy*, 2010:1–13, 2010. ISSN 1687-9449. doi: 10.1155/2010/206362.

[68] X. Liu, S. Zhang, and J. Luo. Complete set of material constants of Pb (In 1/2 Nb 1/2) O 3Pb (Mg 1/3 Nb 2/3) O 3PbTiO 3 single crystal with morphotropic phase boundary composition. *Journal of applied ...*, 106(7):74112, oct 2009. ISSN 0021-8979. doi: 10.1063/1.3243169.

[69] P. Lloyd. The temperature dependence of the elastic constants of a lattice. *Australian Journal of Physics*, 17(4):524, 1964. ISSN 0004-9506. doi: 10.1071/PH640524.

[70] Y. Lu, D. Y. Jeong, Z. Y. Cheng, Q. M. Zhang, H. S. Luo, Z. W. Yin, and D. Viehland. Phase transitional behavior and piezoelectric properties of the orthorhombic phase of pb(mg1/3nb2/3)o3-pbtio3 single crystals. *Applied Physics Letters*, 78(20):3109–3111, 2001. ISSN 00036951. doi: 10.1063/1.1372360.

[71] W. Ma and A. Hao. Polarization rotation and piezoelectricity of electric field-induced monoclinic and triclinic structures in strained pbtio3. *Journal of Applied Physics*, 116 (21):1–7, 2014. ISSN 10897550. doi: 10.1063/1.4903523.

[72] A. May, J.-P. Fleurial, and G. Snyder. Thermoelectric performance of lanthanum telluride produced via mechanical alloying. *Physical Review B*, 78(12):125205, sep 2008. ISSN 1098-0121. doi: 10.1103/PhysRevB.78.125205.

[73] J. Maynard. Resonant Ultrasound Spectroscopy. *Physics Today*, 49(1):26, 1996. ISSN 00319228. doi: 10.1063/1.881483.

[74] H. J. McSkimin. Ultrasonic measurement techniques applicable to small solid specimens. *The Journal of the Acoustical Society of America*, 22(4):413, 1950. ISSN 00014966. doi: 10.1121/1.1906618.

[75] H. J. McSkimin. Pulse Superposition Method for Measuring Ultrasonic Wave Velocities in Solids. *The Journal of the Acoustical Society of America*, 33(1):12, 1961. ISSN 00014966. doi: 10.1121/1.1908386.

[76] A. Migliori, J. Sarrao, W. M. Visscher, T. Bell, M. Lei, Z. Fisk, and R. Leisure. Resonant ultrasound spectroscopic techniques for measurement of the elastic moduli of solids. *Physica B: Condensed Matter*, 183(1-2):1–24, jan 1993. ISSN 09214526. doi: 10.1016/0921-4526(93)90048-B.

[77] L. H. N. V. N. M. S. T. Mohsen Y. Tafti, Mohsin Saleemi. On the chemical synthesis route to bulk-scale skutterudite materials. *Ceramics International*, 42(4):5312 – 5318, 2016. ISSN 0272-8842. doi: http://dx.doi.org/10.1016/j.ceramint.2015.12.061.

[78] M. Mookherjee and J. Tsuchiya. Elasticity of superhydrous phase, b, mg10si3o14(oh)4. *Physics of the Earth and Planetary Interiors*, 238(2):42–50, jan 2015. ISSN 00319201. doi: 10.1016/j.pepi.2014.10.010.

[79] H. J. Mueller. Measuring the elastic properties of natural rocks and mineral assemblages under earth's deep crustal and mantle conditions. *Journal of Geodynamics*, 71: 25–42, nov 2013. ISSN 02643707. doi: 10.1016/j.jog.2012.11.001.

[80] G. Nataf, Q. Li, Y. Liu, and R. Withers. Ferroelastic aspects of relaxor ferroelectric behaviour in pb (in1/2nb1/2) o3-pb (mg1/3nb2/3) o3-pbtio3 perovskite. *Journal of Applied ...*, 124102:1–15, 2013.

[81] B. Noheda. Structure and high-piezoelectricity in lead oxide solid solutions. *Current Opinion in Solid State and Materials Science*, 6(1):27–34, 2002. ISSN 13590286. doi: 10.1016/S1359-0286(02)00015-3.

[82] B. Noheda, D. E. Cox, G. Shirane, J. Gao, and Z.-G. Ye. Phase diagram of the ferroelectric relaxor $(1-x)\mathrm{pbmg}_{1/3}\mathrm{nb}_{2/3}\mathrm{o}_3 - x\mathrm{pbtio}_3$. *Phys. Rev. B*, 66:054104, Aug 2002. doi: 10.1103/PhysRevB.66.054104.

[83] G. S. Nolas, J. Sharp, and H. J. Goldsmid. *Thermoelectrics : basic principles and new materials developments.* Springer series in materials science: v. 45. Berlin ; New York : Springer, 2001., 2001. ISBN 354041245X.

[84] I. Ohno. Free vibration of a rectangular parallelepiped crystal and its application to determination of elastic constants of orthorhombic crystals. *Journal of Physics of the Earth*, 24(4):355–379, 1976. ISSN 1884-2305. doi: 10.4294/jpe1952.24.355.

[85] C. Pantea, D. G. Rickel, A. Migliori, R. G. Leisure, J. Zhang, Y. Zhao, S. El-Khatib, and B. Li. Digital ultrasonic pulse-echo overlap system and algorithm for unambiguous determination of pulse transit time. *Review of Scientific Instruments*, 76(11):114902, 2005. ISSN 00346748. doi: 10.1063/1.2130715.

[86] J. B. Parise, D. J. Weidner, J. Chen, R. C. Liebermann, and G. Chen. In situ studies of the properties of materials under high-pressure and temperature conditions using multi-anvil apparatus and synchrotron x-rays. *Annual Review of Materials Science*, 28 (1):349–374, aug 1998. ISSN 0084-6600. doi: 10.1146/annurev.matsci.28.1.349.

[87] S.-e. Park and T. R. Shrout. Ultrahigh strain and piezoelectric behavior in relaxor based ferroelectric single crystals. *Journal of Applied Physics*, 82(4):1804, 1997. ISSN 00218979. doi: 10.1063/1.365983.

[88] W. Paul and D. Warschauer. *Solids Under Pressure*. McGraw-Hill series in materials science and engineering. McGraw-Hill, 1963.

[89] C. Prescher, L. Dubrovinsky, E. Bykova, I. Kupenko, K. Glazyrin, A. Kantor, C. McCammon, M. Mookherjee, Y. Nakajima, N. Miyajima, R. Sinmyo, V. Cerantola, N. Dubrovinskaia, V. Prakapenka, R. Rueffer, A. Chumakov, and M. Hanfland. High poisson's ratio of earth's inner core explained by carbon alloying. *Nature Geoscience*, 8(3):220 – 223, 2015. ISSN 17520894.

[90] J. H. Qiu, J. N. Ding, N. Y. Yuan, and X. Q. Wang. Phase diagram of (1x)pbmg1/3nb2/3 o3xpbtio3 single crystals. *Journal of Applied Physics*, 117(7): 074101, feb 2015. ISSN 0021-8979. doi: 10.1063/1.4908112.

[91] S. I. Ranganathan and M. Ostoja-Starzewski. Universal elastic anisotropy index. *Physical Review Letters*, 101(5), 2008. ISSN 00319007. doi: 10.1103/PhysRevLett.101.055504.

[92] V. Ravi, S. Firdosy, T. Caillat, E. Brandon, K. Walde, L. Maricic, and A. Sayir. Thermal Expansion Studies of Selected High-Temperature Thermoelectric Materials. *Journal of Electronic Materials*, 38(7):1433–1442, mar 2009. ISSN 0361-5235. doi: 10.1007/s11664-009-0734-2.

[93] B. Rayleigh, John William Strutt and R. B. Lindsay. *The theory of sound.* New York, Dover Publications, 1945., 1945.

[94] W. Rehwald. The study of structural phase transitions by means of ultrasonic experiments. *Advances in Physics*, 22(6):721–755, nov 1973. ISSN 0001-8732. doi: 10.1080/00018737300101379.

[95] W. Ren, L. Han, and R. Wicks. Electric-field-induced phase transitions of< 001>-oriented pb (mg1/3nb2/3) o3-pbtio3 single crystals. *Smart ...*, pages 1–7, 2005.

[96] J. R. Rice and R. Thomson. Ductile versus brittle behaviour of crystals. *Philosophical Magazine*, 29(1):73–97, 1974. doi: 10.1080/14786437408213555.

[97] D. M. Rowe. *CRC handbook of thermoelectrics.* Boca Raton, FL : CRC Press, 1995., 1995. ISBN 0849301467.

[98] P. Sedlák, M. Landa, H. Seiner, L. Bicanova, and L. Heller. Non-contact resonant ultrasound spectroscopy for elastic constants measurement. In *1st International Symposium on Laser Ultrasonics: Science, Technology and Applications, Montreal, Canada*, 2008.

[99] P. Sedlák, H. Seiner, J. Zídek, M. Janovská, and M. Landa. Determination of all 21 independent elastic coefficients of generally anisotropic solids by resonant ultrasound spectroscopy: Benchmark examples. *Experimental Mechanics*, 54(6):1073–1085, 2014. ISSN 17412765. doi: 10.1007/s11340-014-9862-6.

[100] S. Shang, A. Saengdeejing, Z. Mei, D. Kim, H. Zhang, S. Ganeshan, Y. Wang, and Z. Liu. First-principles calculations of pure elements: Equations of state and elastic stiffness constants. *Computational Materials Science*, 48(4):813–826, jun 2010. ISSN 09270256. doi: 10.1016/j.commatsci.2010.03.041.

[101] T. Shrout and J. Fielding. Relaxor ferroelectric materials. In *IEEE Symposium on Ultrasonics*, pages 711–720. IEEE, 1990. doi: 10.1109/ULTSYM.1990.171456.

[102] P. Spoor. *Elastic Properties of Novel Materials Using PVDF Film and Resonance Ultrasound Spectroscopy.* PhD thesis, The Pennsylvania State University, 1997.

[103] R. R. Stern, H. E. Bass, and M. Levy. *Modern Acoustical Techniques for the Measurement of Mechanical Properties.* Number v. 39 in Experimental Methods in the Physical Sciences. Academic Press, 2001. ISBN 9780124759862.

[104] E. Sun and W. Cao. Relaxor-based ferroelectric single crystals: Growth, domain engineering, characterization and applications. *Progress in Materials Science*, 65:124–210, aug 2014. ISSN 00796425. doi: 10.1016/j.pmatsci.2014.03.006.

[105] E. Sun, S. Zhang, J. Luo, T. R. Shrout, and W. Cao. Elastic, dielectric, and piezoelectric constants of pb(in[sub 1/2]nb[sub 1/2])o[sub 3]pb(mg[sub 1/3]nb[sub 2/3])o[sub 3]pbtio[sub 3] single crystal poled along [011][sub c]. *Applied Physics Letters*, 97(3): 032902, jul 2010. ISSN 00036951. doi: 10.1063/1.3466906.

[106] R. Tarumi, Y. Yamaguchi, and Y. Shibutani. Theoretical foundations of resonant ultrasound spectroscopy at high pressure. *Proceedings of the Royal Society A*, 470 (20140448), 2014. ISSN 14712946. doi: 10.1098/rspa.2014.0448.

[107] R. Turner, P. Fuierer, R. Newnham, and T. Shrout. Special issue on transducers materials for high temperature acoustic and vibration sensors: A review. *Applied Acoustics*, 41(4):299 – 324, 1994. doi: http://dx.doi.org/10.1016/0003-682X(94)90091-4.

[108] J. Valasek. Piezo-electric and allied phenomena in rochelle salt. *Phys. Rev.*, 17:475–481, Apr 1921. doi: 10.1103/PhysRev.17.475. URL `http://link.aps.org/doi/10.1103/PhysRev.17.475`.

[109] Y. P. Varshni. Temperature dependence of the elastic constants. *Physical Review B*, 2(10):3952–3958, nov 1970. ISSN 0556-2805. doi: 10.1103/PhysRevB.2.3952.

[110] W. M. Visscher, A. Migliori, T. M. Bell, and R. A. Reinert. On the normal modes of free vibration of inhomogeneous and anisotropic elastic objects. *The Journal of the Acoustical Society of America*, 90(4):2154, 1991. ISSN 00014966. doi: 10.1121/1.401643.

[111] R. Wongmaneerung, R. Guo, A. Bhalla, R. Yimnirun, and S. Ananta. Thermal expansion properties of pmn-pt ceramics. *Journal of Alloys and Compounds*, 461(1-2): 565–569, 2008. ISSN 09258388. doi: 10.1016/j.jallcom.2007.07.086.

[112] C. Wood. Materials for thermoelectric energy conversion. *Reports on Progress in Physics*, 51(4):459–539, apr 1988. ISSN 0034-4885. doi: 10.1088/0034-4885/51/4/001.

[113] Z. Xu, Z. Xi, F. Chen, Z. Li, L. Cao, Y. Feng, and X. Yao. Effects of hydrostatic pressure on the dielectric response and phase transition of pmn-pt68/32 single crystal. *Ceramics international*, 30(7):1699–1701, jan 2004. ISSN 02728842. doi: 10.1016/j.ceramint.2003.12.150.

[114] C. Zener. Contributions to the theory of beta-phase alloys. *Physical Review*, 71(12): 846–851, 1947. ISSN 0031899X. doi: 10.1103/PhysRev.71.846.

APPENDICES

APPENDIX A

THERMAL AGING EFFECTS ON ELASTIC PROPERTIES OF LEAD FREE SOLDER ALLOYS : A PULSE-ECHO STUDY

APPENDIX A

THERMAL AGING EFFECTS ON ELASTIC PROPERTIES OF LEAD FREE SOLDER ALLOYS : A PULSE-ECHO STUDY

A.1 Lead Free Solder Alloys and Reliability Issues

Investigating lead free solders was a highly demanding research topic in late 1990 with the higher concerns of environmental and health issues related to conventional lead based solder alloys. Rules and regulations become more strict on manufacturing consumer electronic equipment and waste disposal where lead is categorized to be a hazardous material. On July 1, 2006, the European Union Waste Electrical and Electronic Equipment Directive (WEEE) and Restriction of Hazardous Substances Directive (RoHS) came into effect, prohibiting the intentional addition of lead to most consumer electronics produced in the EU. Manufacturers in the U.S. may receive tax benefits by reducing the use of lead-based solder. That made the industries seek alternative solder materials, but making reliable and defect free solder joints the same as conventional solder joints became a challenge. Lead free solders in use today contain tin, copper, silver, bismuth, indium, zinc, antimony, and traces of other metals. Some of the applications showed failures of solder joints and increased creep rate with aging under different temperature conditions. Temperature dependence of elastic properties of solder materials under different temperature aging conditions could give insight to the process of understanding reliability of those materials in their usage environment.

A.2 Experimental Procedure

This appendix supplements the dissertation as an example of a limitation in the resonance techniques. In this work, two types of SnAgCu solder alloys were investigated.

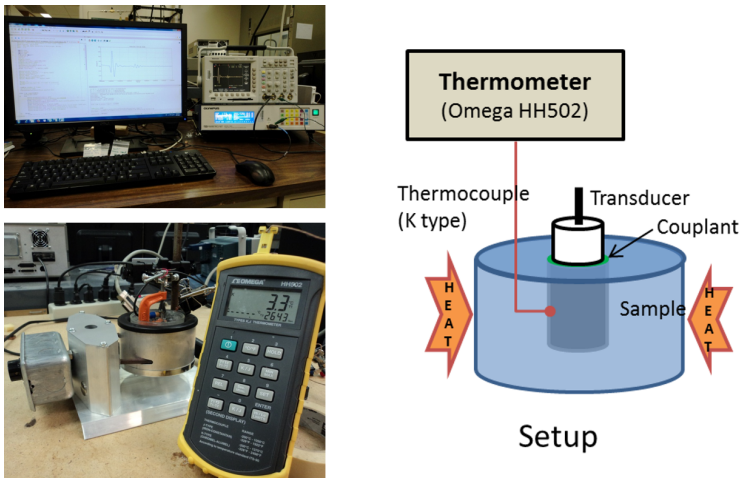

Figure A.1. Pulse echo measurement setup for 0 C – 100 C measurements.

Sample materials were provided by Cisco Inc. Samples for RUS measurements were prepared by Li and resonance spectrum has also been obtained for the samples in the initial stage of this study. However, elastic moduli fits could not be obtained with the resonance data. These alloys are isotropic and homogeneity is also expected where the RUS methodologies can be applied for the elastic constant measurements. The issue with this material was the large grain size, which was later investigated by following chemical etching and microscopic techniques (Figure A.4). Therefore, ultrasonic pulse-echo technique (see Chapter 2 for more details on the technique) was used to measure longitudinal wave speed (C_l) and shear wave speed (C_s). Based on wave speed measurements, elastic properties of the alloy were calculated using the method discussed in Chapter 2. The same experimental procedure room temperature elastic moduli of pure Sn and Cu were measured to validate the results. Multiple samples of each material were measured to increase the accuracy of the results. Room temperature measurements elastic moduli of SnAg alloys were also taken as a reference measurement. Room temperature elastic moduli and sound speeds for the materials being studied are shown in Table A.3. Note that the SnAgCu alloy types A and B have different chemical compositions.

To study the aging effects on the elastic moduli, a total of nine SnAgCu (A) samples were made to cylindrical pellets having the dimension of two parallel surfaces (2 mm – 4 mm).

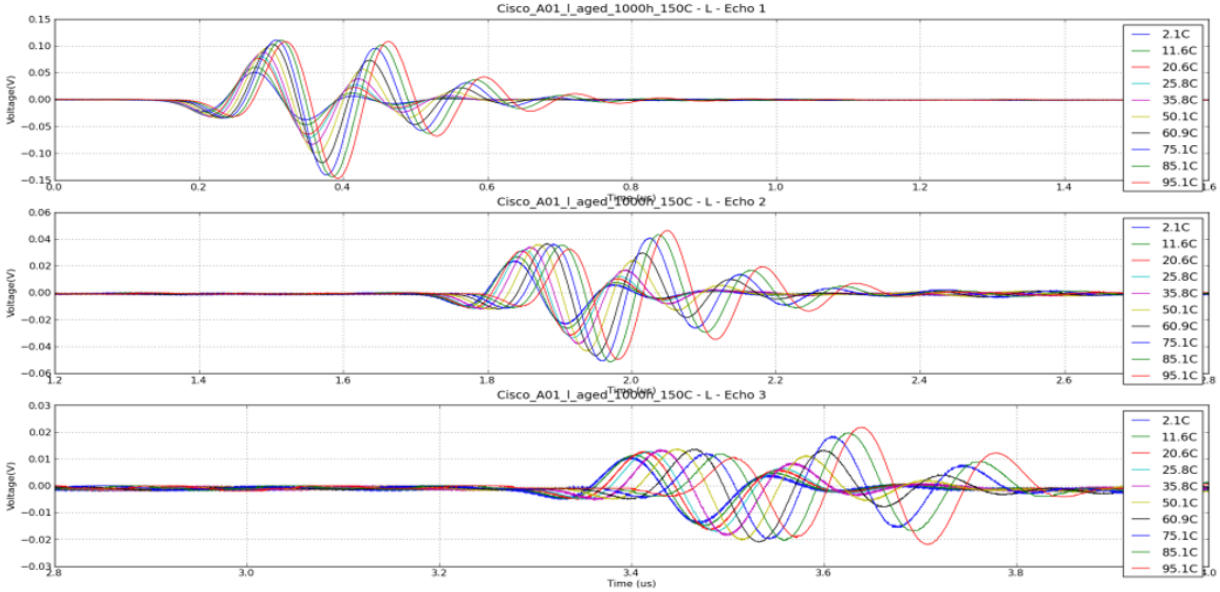

Figure A.2. Time shift of the longitudinal echo with increasing temperature.

Table A.1. Elastic moduli and sound speeds measurements of of SnAgCu Alloys and other reference materials at room temperature.

	Al	Cu	Sn	SnAg (C)	SnAgCu (A)	SnAgCu (B)
M (GPa)	111	189	70	79 ±1%	83 ±2%	88 ±2%
G (GPa)	28	44	17	23 ±4%	23 ±4%	23 ±5%
E (GPa)	74	119	61	59±3%	60 ±3%	60 ±4%
K (GPa)	74	130	59	49 ±3%	52 ±5%	58 ±6%
ν	0.33	0.35	0.33	0.30 ±4%	0.31 ±5%	0.33 ±5%
v_l (ms^{-1})	6400	4660	3336	3293 ±1%	3384 ±1%	3494 ±1%
v_s(ms^{-1})	3204	2251	1656	1764 ±2%	1785 ±2%	1775 ±2%

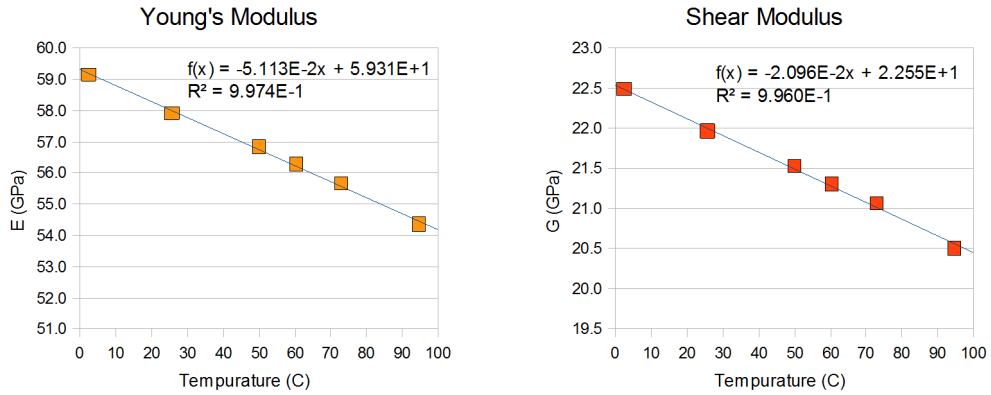

Figure A.3. Temperature dependence of the Young's and shear moduli of SnAgCu sample

Table A.2. Sample Aging Matrix.

Aging Temperature (C)	Duration	
	500 h	1000 h
75	3 samples	3 samples
100	3 samples	3 samples
150	3 samples	3 samples

All samples were aged according to the criteria shown in Table A.2. Sound wave velocity measurements were conducted in the temperature range of 0 C – 100 C. Regular gel type super-glue was used as couplant to avoid a loss in wave transmission from transducer to sample at temperatures higher than 50 C.

Table A.3. Young's modulus and shear modulus as functions of temperature (0 C – 100 C) of SnAgCu(A) samples under different aging conditions (except for the Tin (Sn) sample, function parameter values were averaged over three samples from each aging condition).

Aging Temperature (C)	Duration (hours)	Young's Modulus (E) dE/dT ($\times 10^{-2}$ GPa/C)	Young's Modulus (E) E0 (GPa)	Shear Modulus (G) dG/dT ($\times 10^{-2}$ GPa/C)	Shear Modulus (G) G0 (GPa)
75	500	−6.625	63.88	−2.809	24.46
75	1000	−5.455	60.56	−2.235	22.92
100	500	−6.388	59.20	−2.730	22.78
100	1000	−5.631	58.21	−2.337	22.39
150	500	−6.084	62.83	−2.614	24.13
150	1000	−6.470	60.53	−2.790	23.29
Non-aged		−5.195	62.22	−2.348	23.69
Tin		−7.504	61.18	−3.323	23.54

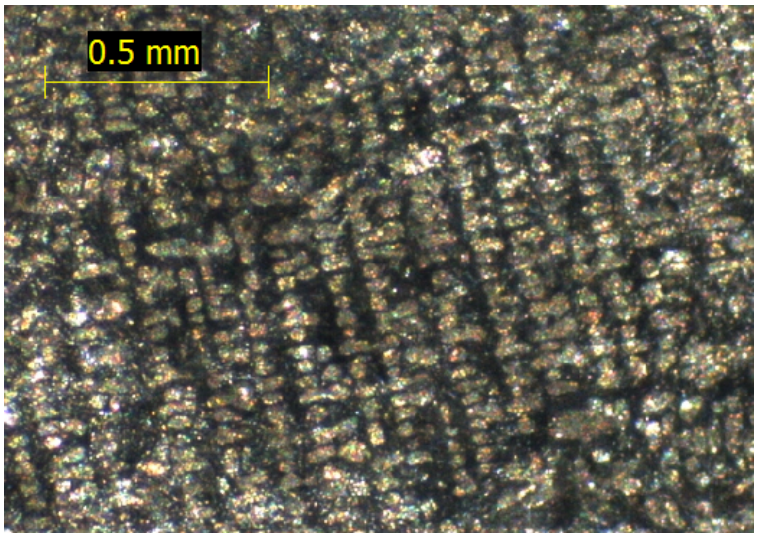

Figure A.4. Micro-structures on etched surface of a SnAgCu sample. Nital (Methanol + HNO_3 (2% by weight)) solution was used as the etching solution.

APPENDIX B

ELASTIC TENSOR FOR DIFFERENT CRYSTAL SYMMETRIES

APPENDIX B

ELASTIC TENSOR FOR DIFFERENT CRYSTAL SYMMETRIES

B.1 Elastic Tensor For Different Crystal Symmetries

The following content is extracted from the references [4, 102, 103]

B.1.1 Triclinic

The triclinic system has 21 independent elastic constants.

$$\begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} & C_{15} & C_{16} \\ C_{12} & C_{22} & C_{23} & C_{24} & C_{25} & C_{26} \\ C_{13} & C_{23} & C_{33} & C_{34} & C_{35} & C_{36} \\ C_{14} & C_{24} & C_{34} & C_{44} & C_{45} & C_{46} \\ C_{15} & C_{25} & C_{35} & C_{45} & C_{55} & C_{56} \\ C_{16} & C_{26} & C_{36} & C_{46} & C_{56} & C_{66} \end{pmatrix}$$

B.1.2 Monoclinic

The monoclinic system has 13 independent elastic constants.

$$\begin{pmatrix} C_{11} & C_{12} & C_{13} & 0 & C_{15} & 0 \\ C_{12} & C_{22} & C_{23} & 0 & C_{25} & 0 \\ C_{13} & C_{23} & C_{33} & 0 & C_{35} & 0 \\ 0 & 0 & 0 & C_{44} & 0 & C_{46} \\ C_{15} & C_{25} & C_{35} & 0 & C_{55} & 0 \\ 0 & 0 & 0 & C_{46} & 0 & C_{66} \end{pmatrix}$$

B.1.3 Orthorombic

The orthorombic system has 9 independent elastic constants.

$$\begin{pmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{22} & C_{23} & 0 & 0 & 0 \\ C_{13} & C_{23} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{55} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{66} \end{pmatrix}.$$

B.1.4 Trigonal

The trigonal system has seven independent elastic constants for the point groups 3 and $\bar{3}$ is

$$\begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} & -C_{25} & 0 \\ C_{12} & C_{11} & C_{13} & -C_{14} & C_{25} & 0 \\ C_{13} & C_{13} & C_{33} & 0 & 0 & 0 \\ C_{14} & -C_{14} & 0 & C_{44} & 0 & C_{25} \\ -C_{25} & C_{25} & 0 & 0 & C_{44} & C_{14} \\ 0 & 0 & 0 & C_{25} & C_{14} & \frac{1}{2}(C_{11} - C_{12}) \end{pmatrix}.$$

and for the the point groups $3m, 32$ and $\bar{3}\frac{2}{m}$ the reduced form of the elastic tensor becomes

$$\begin{pmatrix} C_{11} & C_{12} & C_{13} & C_{14} & 0 & 0 \\ C_{12} & C_{11} & C_{13} & -C_{14} & 0 & 0 \\ C_{13} & C_{13} & C_{33} & 0 & 0 & 0 \\ C_{14} & -C_{14} & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{44} & C_{14} \\ 0 & 0 & 0 & 0 & C_{14} & \frac{1}{2}(C_{11} - C_{12}) \end{pmatrix}$$

For the point groups $4, \bar{4}$, and $\frac{4}{m}$ of the tetragonal system has seven independent constants.

$$\begin{pmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & C_{16} \\ C_{12} & C_{11} & C_{13} & 0 & 0 & -C_{16} \\ C_{13} & C_{13} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{44} & 0 \\ C_{16} & -C_{16} & 0 & 0 & 0 & C_{66} \end{pmatrix}$$

For the point groups $m, 422$ and $\frac{4}{m}\frac{2}{m}\frac{2}{m}$ of the tetragonal system has six independent constants.

$$\begin{pmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{11} & C_{13} & 0 & 0 & 0 \\ C_{13} & C_{13} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{66} \end{pmatrix}$$

B.1.5 Hexagonal

Hexagonal system has five constants

$$\begin{pmatrix} C_{11} & C_{12} & C_{13} & 0 & 0 & 0 \\ C_{12} & C_{11} & C_{13} & 0 & 0 & 0 \\ C_{13} & C_{13} & C_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2}(C_{11} - C_{12}) \end{pmatrix}$$

B.1.6 Cubic

Cubic system has three constants

$$\begin{pmatrix} C_{11} & C_{12} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{11} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{12} & C_{11} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{44} \end{pmatrix}$$

B.1.7 Isotropic

Isotropic system has two constants

$$\begin{pmatrix} C_{11} & C_{12} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{11} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{12} & C_{11} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2}(C_{11} - C_{12}) & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2}(C_{11} - C_{12}) & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2}(C_{11} - C_{12}) \end{pmatrix}$$

APPENDIX C

RUS DATA ACQUISITION AND ANALYSIS COMPUTER PROGRAMS

APPENDIX C

RUS DATA ACQUISITION AND ANALYSIS COMPUTER PROGRAMS

Computer backed data acquisition and analysis is an essential part of the modern day experimental research in science. Most of the modern lab equipment comes with the capability of interfacing with a computer system that allows the scientist to control the equipment via computer softwares and to electronically record data. That also enables the automation of some experimental procedures to save time and efforts for the scientist and ease complexity of the experiential procedures. However, the human factor is still an essential component in both data acquisition and analysis regarding decision making, and scientific thinking as well as the safety concerns.

The common ways to interface with the laboratory equipment are, General Purpose Interface Bus (GPIB) also known as IEEE-488 [36, 52], RS-232 [7, 23], Ethernet [46], and Universal Serial Bus (USB) [7, 25, 47].

This appendix contains the RUS data acquisition computer program codes in Python programing language, and data analysis computer programing codes were written in both python and Fortran.

The RUS data acquisition system (RUS-DAQ) Python [30] program code was recreated in this work from its original version. Object oriented concepts were applied to increase the re-usability and ease of maintenance. A new graphical user interface was also added to the RUS-DAQ system. RUS-DAQ uses Virtual Instrument Software Architecture(VISA) [28] to interface with the equipment via GPIB ports. VISA is a widely used input/output Application programming interface (API) with the test and measurement equipment and, PyVISA [29] coupled with NI-VISA [19] works as the API layer between the RUS-DAQ program code and the equipment hardware.

A simple python code (RUS-PeakFit) was created to analyze resonance peaks, following the algorithm created by Gladden et al. Peaks are analyzed as adjacent groups or as isolated peaks by slicing the dataset into smaller windows where each window contains one or more peaks. A modified Lorentzian peak was fitted to the quadrature response of the spectrum data in order to extract the center frequency and quality factor of each resonance peak. These center frequency values are used as the list of resonance infrequences of the sample. In high temperature and pressure experiments, this computer code was slightly modified to track the peaks with the change in temperature to avoid peak misidentification.

The computer algorithm (RUS-Fit) solves the forward of finding the natural modes when given the elastic constants, density and geometry. The basic form of the RUS-Fit algorithm is the isotropic fitting routine and, the algorithm also handles the fitting routines for the single crystal materials with different crystal classes. Crystal orientation also taken as an input parameter in this calculation. The RUS-Fit algorithm(s), written in the Fortran programing language and are created by Gladden et al. [2, 38, 66], was used for data analysis in this work.

The high temperature elastic constant fit (RUS-TempRun) algorithm was re-created to fit frequency data at each temperature point to estimate elastic constants. RUS-TempRun algorithm accepts the sample dimensions, density at room temperature and pressure, and thermal expansion data for the temperature range being investigated. The algorithm computes the sample dimensions and density at each temperature points using the thermal expansion data, and calls the RUS-Fit Fortran algorithm within the python environment to estimate the elastic moduli for the data set at each temperature. This process continues until all the temperature data has been analyzed, and outputs the results tables and plots of elastic moduli and fitting errors verses the temperature.

C.1.1 RUS-DAQ Main

```
import rt2010.EXPERIMENT as EXPR
import rt2010.RUSGUI as GUI

print '''
This is a Python based RUS data aquasition and analysis system
    version %s.
Required modules are: PyVISA, Pylab (matplotlib)
-------------------------------------------------------------
Supplied modules are:
 * INSTRUMENT (Genaralized instrument class)
 * SYNT_DS345 (Synthesizer)
 * LAMP_SR844 (Lock-in Amp)
 * SCAN (RUS scan tools)
 * PLOTSCAN (ploting tools)
 * EXPERIMENT (work environment)
-------------------------------------------------------------
For single can
 * rus.single_scan(start_frequency,stop_frequency,freq_points,
    amplitude)
For Batch Scan
 * rus.batch_scan()
 * The file "rusin.txt" should be in the folder
-------------------------------------------------------------
''' %__version__

rus=EXPR.EXPERIMENT()
rus.setFRange("LF")

def runGUI():
    try:
        GUI.RUSGUI()
    except:
        print "GUI Error : Cannot Implement GUI, use text commands
            "


def single_scan(start, stop, numPoints, fName="tmp_dump.dat", ampl
    =6.0, sens=8, dfSleep=0.05, numAve=1, dsSleep=0.01):
    rus.single_scan(start, stop, numPoints, fName="tmp_dump.dat",
        ampl=6.0, sens=8, dfSleep=0.05, numAve=1, dsSleep=0.01)
```

```
def batch_scan():
    rus.batch_scan()

def batch_scan_temp():
    rus.batch_scan_temp()

def change_lamp(fRange):# "HF" or "LF"
    rus.setFRange(fRange)
    print "%s Lock-in Amp selected"%rus.getFRange()
#rus.single_scan(0.4,0.5,100)
#rus.batch_scan()

runGUI()
```

C.1.2 Instrument Control Classes

C.1.2.1 INSTRUMENT

```
if __name__ == "__main__":
    print "Instrument Class";
#Error Messages
Err_NoPyVISA="No PyVISA module found !!! \n Install PyVSA first (http
    ://pypi.python.org/pypi/PyVISA , http://pyvisa.sourceforge.net/)"

#import required modules
try: import visa.visa as visa
except: print Err_NoPyVISA


class INSTR:
    def __init__(self,name="Instrument", model="model",year="2000",gpib
        ="01"):
          self.name=name
          self.model=model
          self.year=year
          self.gpib=gpib
          self.inst=""

    def setGPIB(self,gpib):
          self.gpib=gpib

    def getGPIB(self):
          return self.gpib

    def connectInst(self):
          try:
```

```
            self.inst=visa.instrument("GPIB::%i"%self.gpib)
        except: print "Couldn't communicate with Instrument : "+self.
            name

    def query(self):
        try:
            self.inst.write("*IDN?")
            qr=self.inst.read()
        except:
            qr="NONE"
        return qr

    def getModel(self):
        return self.model

    def getName(self):
        return self.name

    def getYear(self):
        return self.year

    def info(self):
        return "Name   : "+self.getName()+ "\nModel : "+self.getModel()+
            "\nYear   : "+self.getYear()+"\nGPIB   : "+str(self.getGPIB())

def test():
    myInst=INSTR()
    myInst.info()

test()
```

C.1.2.2 SR830

```
if __name__ == "__main__":
    print "SR830 Lock-In Amplifier Class";

from INSTRUMENT import *

class LAMPSR830(INSTR):
    def __init__(self,name="Lo-F Lock-In Amp", model="SR830",year="2000
        ",gpib=8):
        self.name=name
        self.model=model
        self.year=year
        self.gpib=gpib
        self.inst=0

    def setSensitivity(self,sens):
```

```
        self.sens=sens
        self.inst.write("SENS %i"% self.sens)

    def setAutoPhaseFn(self):
        self.inst.write("APHS")

    def getOutputXY(self):
        X,Y=self.inst.ask_for_values("SNAP?1,2")
        return X,Y

    def reset(self):
        self.inst.write("RSET 40")
        return "SR830 was reset..."

    def getSens(self):
        sens=["2 nV/fA", "5 nV/fA", "10 nV/fA", "20 nV/fA", "50 nV/fA",
              "100 nV/fA", "200 nV/fA", "500 nV/fA", "1 uV/pA", "2 uV/pA"
             , " 5 uV/pA", " 10 uV/pA", " 20 uV/pA","50 uV/pA", "100 uV/
             pA", "200 uV/pA", "500 uV/pA", "1 mV/nA", "2 mV/nA", "5 mV/
             nA", "10 mV/nA", "20 mV/nA", "50 mV/nA", "100 mV/nA", "200 
             mV/nA", "500 mV/"]
        s=int(self.inst.ask_for_values("SENS?")[0])
        return "%d : %s"%(s,sens[s])
```

C.1.2.3 SR844

```
if __name__ == "__main__":
    print "SR844 Lock-In Amplifier Class";

from INSTRUMENT import *

class LAMPSR844(INSTR):
    def __init__(self,name="Hi-F Lock-In Amp", model="SR844",year="2000
        ",gpib=8):
        self.name=name
        self.model=model
        self.year=year
        self.gpib=gpib
        self.inst=0

    def setSensitivity(self,sens):
        self.sens=sens
        self.inst.write("SENS %i"% self.sens)

    def setAutoPhaseFn(self):
        self.inst.write("APHS")

    def getOutputXY(self):
```

```
        X,Y=self.inst.ask_for_values("SNAP?1,2")
        return X,Y

    def reset(self):
        self.inst.write("RSET 40")
        return "SR844 was reset..."

    def getSens(self):
        sens=["100 nVrms", "300 nVrms", "1 uVrms", "3 uVrms", "10 uVrms
            ", "30 uVrms", "100 uVrms", "300 uVrms", " 1 mVrms", " 3
            mVrms", "10 mVrms", "30 mVrms", "100 mVrms", "300 mVrms", "1
            Vrms"]
        s=int(self.inst.ask_for_values("SENS?")[0])
        return "%d : %s"%(s,sens[s])
```

C.1.2.4 DS345

```
if __name__ == "__main__":
    print "SR844 Lock-In Amplifier Class";

from INSTRUMENT import *
from pylab import *
import time as t

class SYNTDS345(INSTR):
    def __init__(self, name="Synthesizer", model="DS345", year="2000",
        gpib=19):
        self.name = name
        self.model = model
        self.year = year
        self.gpib = gpib
        self.ampl = 5
        self.freq=1.0
        self.inst = 0
    def reset(self):
        self.inst.write("FUNC %i"% "*RST")
        return "DS345 was reset..."

    def setAmplitude(self, ampl):
        self.ampl=ampl
        self.inst.write("AMPL %.2fVP"% ampl)

    def getAmplitude(self):
        ampl=self.inst.ask_for_values("AMPL?")[0]
        return ampl

    def setFrequency(self,freq):
        #self.freq=freq
```

```
        freq=freq*(1e6) # Convert value in MHz to Hz
        self.inst.write("FREQ %f"% freq)

    def getFrequency(self):
        freq=self.inst.ask_for_values("FREQ?")[0]/(1e6)
        return freq # Value in MHz

    def setFunction(self,func=0):
        #0 SINE | 1 SQUARE | 2 TRIANGLE | 3 RAMP | 4 NOISE | 5
           ARBITRARY
        self.inst.write("FUNC %i"% func)

    def setModWavFrm(self,swmod):
        # 0 SINGLE SWEEP | 1 RAMP | 2 TRIANGLE | 3 SINE | 4 SQUARE | 5
           ARB | 6 NONE
        self.inst.write("MDWF %i"% swmod)

    def setDCOffset(self,dcOff):
        #Offset + Peak amplitude should be less than 5V
        if(dcOff+self.ampl/2)<= 5:
            self.inst.write("OFFS %f"% dcOff)

    def freqSweep(self,ampl,start,stop,numPoints, response,dfSleep
       =0.05,numAve=1,dsSleep=0.01):
        self.setAmplitude(ampl)
        freqStep=(stop-start)/numPoints
        f=start
        data=[]
        print "-"*50
        print "Amplitude        : %.2f V" % ampl
        print "Start Frequency  : %.2f MHz" % start
        print "Stop Frequency   : %.2f MHz" % stop
        print "Step             : %.1f Hz" % (freqStep*(1e6))
        print "Averages         : %d" % numAve
        print "-"*50
        print "Scan Started..."
        for i in range(numPoints+1):
            t.sleep(dfSleep)
            self.setFrequency(f)
            x=0
            y=0
            for j in range(numAve):
                tempx,tempy=response()
                x+=tempx
                y+=tempy
            x=x/numAve
            y=y/numAve
```

```
            data.append([f,x,y])
            f=f+freqStep
            done=(float(i)/float(numPoints)*100)
            if (done%10==0):
                print "%.1f" % done +"% done"
                savetxt("dump.dat",data,fmt='%1.6e',delimiter='\t')
        print "-"*50
        return array(data)
```

C.1.2.5 K2000

```
if __name__ == "__main__":
    print "Keithley 2000 DMM";

from INSTRUMENT import *
from pylab import *
import time as t

class DMMK2000(INSTR):
    def __init__(self, name="Keithley 2000 DMM", model="2000", year="
        2000", gpib=16):
        self.name = name
        self.model = model
        self.year = year
        self.gpib = gpib
        self.ampl = 5
        self.freq=1.0
        self.inst = 0

    def reset(self):
        self.inst.write("")
        return "DS345 was reset..."

    def getVoltage(self):
        v=self.inst.ask_for_values(":MEAS:VOLT:DC?")
        return float(v[0])*1e6

    def getVoltageAvg(self,nAvg):
        vsum=0
        for i in range(nAvg):
            vsum+=self.getVoltage()
        v=vsum/nAvg
```

C.1.2.6 Thermocouple (TC)

```
#from INSTRUMENT import *
from pylab import *
import time as t
```

```
class TCP():
    def __init__(self, DMM, name="Thermocouple", type="N"):
        self.name = name
        self.type = type
        self.DMM=DMM

    def calibCurve(self,x):
        if self.type=="N": ## For 0 - 1300C range (+/- 0.06C)(N-type)
            a0=0.000000000
            a1=3.8783277E-2
            a2=-1.1612344E-6
            a3=6.9525655E-11
            a4=-3.0090077E-15
            a5=8.8311584E-20
            a6=-1.6213839E-24
            a7=1.6693362E-29
            a8=-7.3117540E-35
            temp=a0+a1*x+a2*x**2+a3*x**3+a4*x**4+a5*x**5+a6*x**6+a7*x
                **7+a8*x**8
        elif self.type=="J":
            temp=0
            print "J type thermocouple not supported !"
        elif self.type=="K":
            temp=0
            print "K type thermocouple not supported !"

        return temp


    def getTemp(self,nAvg=2):
        v=self.DMM.getVoltage()
        return self.calibCurve(v)
```

C.1.3 Experiment Control Classes

C.1.3.1 EXPERIMENT

```
import SCAN as SC
import RUSSETUP as RUS
from pylab import *


class EXPERIMENT(): #Interface to users familiar with old rus software
    def __init__(self,measT=False,fRange="HF"):
        self.measT=measT
        self.fRange=fRange
        self.rusSetup=RUS.RUSSETUP(measT,fRange)
```

```
        self.folder="" #Folder to store data files (Not implemented yet
            !
        self.rusSetup.printRUSSetup()

    def setMeasureT(self,measT=True): #Also can set it to False when
        calling
        self.measT=measT

    def isMeasureT(self):
        return self.measT

    def getFRange(self):
        return self.fRange

    def setFRange(self,fRange): # "HF" or "LF"
        self.fRange=fRange
        self.rusSetup.setLAmp(fRange)

    def setFolder(self,fPath):
        self.folder=fPath

    def getFolder(self):
        return fPath

    def single_scan(self, start, stop, numPoints, fName="tmp_dump.dat",
        ampl=6.0, sens=8, dfSleep=0.05, numAve=1, dsSleep=0.01):
        scan=SC.RUSSCAN(self.rusSetup,self.measT)
        scan.singleScan(ampl, start, stop, numPoints, fName, sens,
            dfSleep, numAve, dsSleep)
        scan.plotScan()
        show()
        self.printMesssage()

    def batch_scan(self):
        batch = SC.RUSBATCH(self.rusSetup)
        batch.batchScan()
        self.printMesssage()

    def batch_scan_temp(self):
        batch = SC.RUSBATCH(self.rusSetup)
        batch.batchScan(measT=True)
        self.printMesssage()

    def printMesssage(self):
        print '''
Python based RUS data aquasition and processing system version %s.
```

```
For single can
 * rus.single_scan(start_frequency,stop_frequency,freq_points,
    amplitude)

For Batch Scan
 * rus.batch_scan()
 * The file "rusin.txt" should be in the folder
-------------------------------------------------------------
            ''' %__version__

#def main():
#    myExp=EXPERIMENT()
#    #myExp.batch_scan()
#    myExp.single_scan(0.4,0.5,100)
#    show()
#
##programs starts here
#main()
```

C.1.3.2 RUSSETUP

```
import LAMP_SR844 as HFLAMP
import LAMP_SR830 as LFLAMP
import SYNT_DS345 as SYNT
import DMM_K2000 as DMM
import TC as TC

class RUSSETUP():
    def __init__(self, measT=False,fRange="HF"):
# Currently measT and fRange values hold by the EXPERIMENT
#         self.measT=measT
#         self.fRange=fRange
        self.mySynth = SYNT.SYNTDS345()#Synthesizer
        self.mySynth.connectInst()
#         self.setFreq=self.mySynth.setFreqency(freq)

        if(fRange=="HF"):
            self.myLAmp = HFLAMP.LAMPSR844()#HF Lock-in Amplifier
        elif(fRange=="LF"):
            self.myLAmp = LFLAMP.LAMPSR830() #LF Lock-in Amplifier
        self.myLAmp.connectInst()
        self.response = self.myLAmp.getOutputXY #Symbolize function
           which can get response signal

        self.myDMM=DMM.DMMK2000()#DMM
        if (measT==True):
            self.myDMM.connectInst()
            self.myTC=TC.TCP(self.myDMM)#Thermocouple
```

```
#             self.getTCVoltage=self.myDMM.getVoltage() #Thermocouple
    voltage

#     def setMeasureT(self,measT=True): #Also can set it to False when
    calling
#         self.measT=measT
#
#     def isMeasureT(self):
#         return self.measT
#
#     def getFRange(self):
#         return self.fRange
#
#     def setFRange(fRange): # "HF" or "LF"
#         self.fRange=fRange
    def setLAmp(self,fRange):
        if(fRange=="HF"):
            self.myLAmp = HFLAMP.LAMPSR844()#HF Lock-in Amplifier
        elif(fRange=="LF"):
            self.myLAmp = LFLAMP.LAMPSR830() #LF Lock-in Amplifier
        self.myLAmp.connectInst()
        self.response = self.myLAmp.getOutputXY #Symbolize function
           which can get response signal

    def printRUSSetup(self):
        print "Synthesizer : " + self.mySynth.query() #
        print "Lock-In Amp : " + self.myLAmp.query()#
        print "DMM         : " + self.myDMM.query()
```

C.1.3.3 SCAN

```
from pylab import *
import time as t

class RUSBATCH():
    def __init__(self, rusSetup ,batchFile="rusin.txt"):
        self.rusSetup = rusSetup
        self.batchFile = batchFile
        self.myScans = []
        self.ampl = 0
        self.start=[]
        self.stop=[]
        self.numPoints=[]
        self.fname=[]
        self.sens=[]
        self.numAve=1
        self.dfSleep=0.05
        self.dsSleep=0.01
```

```
    def batchScan(self,measT=False):
        self.readScanFile()
        for i in range(len(self.start)):
            print "=" * 50
            print "Scan %i" % (i + 1) #Scan numebr
            print "-" * 50
            self.myScans.append(RUSSCAN(self.rusSetup,measT))
            self.myScans[i].singleScan(self.ampl, self.start[i], self.
               stop[i],self.numPoints[i], self.fname[i], self.sens[i],
               self.dfSleep, self.numAve, self.dsSleep)

        for j in range(len(self.myScans)):
            figure(j)
            self.myScans[j].plotScan()
        show()

    def readScanFile(self):
        scanFile = open(self.batchFile, 'r')
        scans = scanFile.readlines()
        scanFile.close()
        #print scans
        self.ampl = float(scans[0].split(',')[0])
        self.dfSleep = float(scans[0].split(',')[1])
        self.dsSleep = float(scans[0].split(',')[2])
        self.numAve = int(scans[0].split(',')[3])

        for scan in range(1, len(scans)):
            params = scans[scan].split(',')
            self.start.append(float(params[0]))
            self.stop.append(float(params[1]))
            self.numPoints.append(int(params[2]))
            self.fname.append(params[3].strip())
            self.sens.append(int(params[4]))

class RUSSCAN():
    def __init__(self, rusSetup,measT):
        self.mySynth = rusSetup.mySynth
        self.myLAmp = rusSetup.myLAmp
        self.response = rusSetup.response
        self.measT=measT
        if self.measT==True:
            self.myTC=rusSetup.myTC
        self.fName = ""
        self.data = []

    def singleScan(self, ampl, start, stop, numPoints, fName, sens,
```

```
    dfSleep, numAve, dsSleep):
        self.myLAmp.setAutoPhaseFn()
        self.myLAmp.setSensitivity(sens)
        self.fName = fName
        sensitivity=self.myLAmp.getSens()
        self.data = self.freqSweep(ampl, start, stop, numPoints, self.
            response,sensitivity, dfSleep, numAve, dsSleep,self.measT)
        self.saveScanData()


def freqSweep(self,ampl,start,stop,numPoints, response,sens,dfSleep
    =0.05,numAve=1,dsSleep=0.01,measT=False):
        self.mySynth.setAmplitude(ampl)
        freqStep=(stop-start)/numPoints
        f=start
        data=[]
        print "-"*50
        print "Amplitude        : %.2f V" % ampl
        print "Start Frequency  : %.2f MHz" % start
        print "Stop Frequency   : %.2f MHz" % stop
        print "Step             : %.1f Hz" % (freqStep*(1e6))
        print "Averages         : %d" % numAve
        print "Sensitivity      : %s" % sens
        print "-"*50
        print "Scan Started..."
        startT=t.time()

        if self.measT==True:
            tdata=[]
            for i in range(numPoints+1):
                t.sleep(dfSleep)
                x,y=scanPoint(f,response,numAve)
                data.append([f,x,y])
                f=f+freqStep
                done=(float(i)/float(numPoints)*100)
                if (done%10==0):
                    nowT=t.time()
                    elapsedT=nowT-startT
                    temp=self.myTC.getTemp()
                    tdata.append([elapsedT,temp])
                    print "%.1f" % done +"% done" + "   Temperature : "
                        %.2f C" %temp
                    savetxt("dump.dat",data,fmt='%1.6e',delimiter='\t')
            self.tempStatus(tdata)

        else:
            for i in range(numPoints+1):
```

```
            t.sleep(dfSleep)
            x,y=self.scanPoint(f,response,numAve)
            data.append([f,x,y])
            f=f+freqStep
            done=(float(i)/float(numPoints)*100)
            if (done%10==0):
                print "%.1f" % done +"% done"
                savetxt("dump.dat",data,fmt='%1.6e',delimiter='\t')

    print "-"*50
    return array(data)

def scanPoint(self,f,response,numAve): # Scans one frequency point
    and returns output
    self.mySynth.setFreqency(f)
    x=0
    y=0
    for j in range(numAve):
        tx,ty=response() #Temporary variables
        x+=tx
        y+=ty
    x=x/numAve
    y=y/numAve
    return x,y



def tempStatus(self,tdata):
    tempfile="temps.dat"
    atdata=array(tdata)
    temps=atdata[:,1]
    times=atdata[:,0]
    maxtemp=max(temps)
    mintemp=min(temps)
    avgtemp=average(temps)
    deltemp=std(temps)
    savetxt(tempfile,tdata,fmt='%4.2f',delimiter='\t')
    print  "-"*50
    print "Batch scan complete.  \n Time required was %3.2f minutes
        \n Final Temperature = %.2f" % (tdata[-1][0],tdata[-1][1])
    print  "="*50+"\n"
    tempStatusFileString= '''
    Temperature Stats:
    Maximum temp = %3.2f C
    Minimum Temp = %3.2f C
    Average Temp = %3.2f +/- %2.2f C
    Std.Dev.     = %3.2f C
```

```
        ==========
        '''%(maxtemp,mintemp,avgtemp,(maxtemp-mintemp)/2.,deltemp)
        print tempStatusFileString
        tempStatusFile=open('temp_stats.dat','w')
        tempStatusFile.write(tempStatusFileString)
        tempStatusFile.close()


    def saveScanData(self):
        savetxt(self.fName, self.data, fmt='%1.6e', delimiter='\t')
        try:
            #Need to avoid overwrite on existing file !
            savetxt(self.fName, self.data, fmt='%1.6e', delimiter='\t')
            print "NOTE: Data succesfully saved in %s" % self.fName
            self.data = [] #to release memory from large amout of data.
        except:
            print "Error: Couldn't save data!"

    def loadScanData(self):
        '''
        To reload data saved in the file
        '''
        try:
            self.data = loadtxt(self.fName, delimiter='\t')
        except:
            print "Error: Couldn't load data!"

    def plotScan(self):
        self.loadScanData()
        F = self.data[:, 0]
        X = self.data[:, 1]
        Y = self.data[:, 2]
        title("RUS Scan - Spectrum")
        xlabel("Frequency (MHz)")
        ylabel("Voltage (V)")
        plot(F, Y)
```

C.2 Data Visualization and Peak Finding Algorithm

C.2.0.4 RUSGUI

```
# To change this template, choose Tools | Templates
# and open the template in the editor.

__author__ = "Sumudu Tennakoon"
__date__ = "$Oct 26, 2010 11:18:16 PM$"
```

```
from pylab import *
import wx as wx
import wx.lib.plot as plot
import EXPERIMENT as EXPR

ID_ABOUT = 101
ID_EXIT  = 102

class MyFrame(wx.Frame):
    def __init__(self, parent, id, title):
        wx.Frame.__init__(self, parent, id, title, size=(250, 450))
        self.Bind(wx.EVT_CLOSE, self.closeWindow)

        #Status Bar
        self.CreateStatusBar()
        self.SetStatusText("RUS Lab - NCPA")

        #Panels
        self.pnlCtrl = wx.Panel(self)
        #self.pnlPlot=wx.Panel(self)

        #Grid
        posCol1 = 20 #Left margin of 1st colum (x axis)
        posCol2 = 270 #Left margin of 1st colum (x axis)
        posCol3 = 520 #Left margin of 1st colum (x axis)

        lblWidth = 75
        txtWidth = 80
        unitWidth = 50
        btnWidth = 65
        btnHeight = 25
        mtxtHeight = 100

        #Column1
        posVoltage = 20
        posSFreq = 50
        posEFreq = 80
        posDResol = 110
        posSens = 140
        posSIntv = 170
#         posScans = 200
        posAveg = 200
        posFile=230
        posFRange=260
        posMTemp=290
        posBtn=320
```

```
#Column2
posWDir = 20
posCFile = 50
posSFile = 80
posConf = 110
posConfBtn = posConf + mtxtHeight

#Column3
posRunBtn = 20
posTemp = 50
posFreq = 80
posPrgss = 110

## Scan Info##
#Voltage Control
self.lblVoltage = wx.StaticText(self.pnlCtrl, -1, "Voltage",
    pos=(posCol1, posVoltage), size=(lblWidth, -1))
self.txtVoltage = wx.TextCtrl(self.pnlCtrl, -1, "6", pos=(
    posCol1 + lblWidth, posVoltage), size=(txtWidth, -1))
self.txtVoltage.SetInsertionPoint(0)
self.lblVoltageUnit = wx.StaticText(self.pnlCtrl, -1, " V", pos
    =(posCol1 + lblWidth + txtWidth, posVoltage), size=(
    unitWidth, -1))

#Start Frequency
self.lblSFreq = wx.StaticText(self.pnlCtrl, -1, "S Freq", pos=(
    posCol1, posSFreq), size=(lblWidth, -1))
self.txtSFreq = wx.TextCtrl(self.pnlCtrl, -1, "0.4", pos=(
    posCol1 + lblWidth, posSFreq), size=(txtWidth, -1))
self.txtSFreq.SetInsertionPoint(0)
self.lblSFreqUnit = wx.StaticText(self.pnlCtrl, -1, " MHz", pos
    =(posCol1 + lblWidth + txtWidth, posSFreq), size=(unitWidth,
     -1))

#End Frequency
self.lblEFreq = wx.StaticText(self.pnlCtrl, -1, "E Freq", pos=(
    posCol1, posEFreq), size=(lblWidth, -1))
self.txtEFreq = wx.TextCtrl(self.pnlCtrl, -1, "0.5", pos=(
    posCol1 + lblWidth, posEFreq), size=(txtWidth, -1))
self.txtEFreq.SetInsertionPoint(0)
self.lblEFreqUnit = wx.StaticText(self.pnlCtrl, -1, " MHz", pos
    =(posCol1 + lblWidth + txtWidth, posEFreq), size=(unitWidth,
     -1))

#Data Points
self.lblPoints = wx.StaticText(self.pnlCtrl, -1, "DPoints", pos
    =(posCol1, posDResol), size=(lblWidth, -1))
```

```
        self.txtPoints = wx.TextCtrl(self.pnlCtrl, -1, "100", pos=(
            posCol1 + lblWidth, posDResol), size=(txtWidth, -1))
        self.txtPoints.SetInsertionPoint(0)
        self.lblPointsUnit = wx.StaticText(self.pnlCtrl, -1, "␣Pt", pos
            =(posCol1 + lblWidth + txtWidth, posDResol), size=(unitWidth
            , -1))

        #Sensitivity
        self.lblSens = wx.StaticText(self.pnlCtrl, -1, "Sens", pos=(
            posCol1, posSens), size=(lblWidth, -1))
        self.txtSens = wx.TextCtrl(self.pnlCtrl, -1, "8", pos=(posCol1
            + lblWidth, posSens), size=(txtWidth, -1))
        self.txtSens.SetInsertionPoint(0)

        #Scan Interval
        self.lblSIntv = wx.StaticText(self.pnlCtrl, -1, "SIntv", pos=(
            posCol1, posSIntv), size=(lblWidth, -1))
        self.txtSIntv = wx.TextCtrl(self.pnlCtrl, -1, "0.03", pos=(
            posCol1 + lblWidth, posSIntv), size=(txtWidth, -1))
        self.txtSIntv.SetInsertionPoint(0)
        self.lblSIntvUnit = wx.StaticText(self.pnlCtrl, -1, "␣s", pos=(
            posCol1 + lblWidth + txtWidth, posSIntv), size=(unitWidth,
            -1))

        #Lockin Amp Range
        choice=["HF","LF"]
        self.lblFRange = wx.StaticText(self.pnlCtrl, -1, "LAmp", pos=(
            posCol1, posFRange), size=(lblWidth, -1))
        self.cmbFRange = wx.ComboBox(self.pnlCtrl, -1, value=choice[0],
             pos=(posCol1+lblWidth, posFRange), size=(txtWidth, -1),
            choices=choice)

        #Monitor Temp
        self.lblMTemp = wx.StaticText(self.pnlCtrl, -1, "Temp␣Monitor",
             pos=(posCol1, posMTemp), size=(lblWidth, -1))
        self.chkMTemp = wx.CheckBox(self.pnlCtrl, -1 ,'', pos=(posCol1+
            lblWidth, posMTemp))

#         #Scans
#         self.lblScans = wx.StaticText(self.pnlCtrl, -1, "Scans", pos=(
    posCol1, posScans), size=(lblWidth, -1))
#         self.txtScans = wx.TextCtrl(self.pnlCtrl, -1, "3", pos=(
    posCol1 + lblWidth, posScans), size=(txtWidth, -1))
#         self.txtScans.SetInsertionPoint(0)

        #Averages
        self.lblAveg = wx.StaticText(self.pnlCtrl, -1, "NumAveg", pos=(
```

```
            posCol1, posAveg), size=(lblWidth, -1))
        self.txtAveg = wx.TextCtrl(self.pnlCtrl, -1, "1", pos=(posCol1
            + lblWidth, posAveg), size=(txtWidth, -1))
        self.txtAveg.SetInsertionPoint(0)

        #Save File Name
        self.lblSFile = wx.StaticText(self.pnlCtrl, -1, "Save File",
            pos=(posCol1, posFile), size=(lblWidth, -1))
        self.txtSFile = wx.TextCtrl(self.pnlCtrl, -1, "test1.dat", pos
            =(posCol1 + lblWidth, posFile), size=(txtWidth, -1))
        self.txtSFile.SetInsertionPoint(0)

        #Start Button
        self.btnStart = wx.Button(self.pnlCtrl, label="Start", pos=(
            posCol1 + lblWidth, posBtn), size=(btnWidth, btnHeight))
        self.Bind(wx.EVT_BUTTON, self.startScan, self.btnStart)
#
#        #Pause Button
#        self.btnPause = wx.Button(self.pnlCtrl, label="Pause", pos=(
    posCol3 + btnWidth, posRunBtn), size=(btnWidth, btnHeight))
#        self.Bind(wx.EVT_BUTTON, self.pauseScan, self.btnPause)
#
#        #Stop Button
#        self.btnStop = wx.Button(self.pnlCtrl, label="Stop", pos=(
    posCol1 + btnWidth * 2, posRunBtn), size=(btnWidth, btnHeight))
#        self.Bind(wx.EVT_BUTTON, self.stopScan, self.btnStop)
#
#
#        ## Configuration Info##
#        #Working Directory
#        self.lblWDir = wx.StaticText(self.pnlCtrl, -1, "PWD", pos=(
    posCol2, posWDir), size=(lblWidth, -1))
#        self.txtWDir = wx.TextCtrl(self.pnlCtrl, -1, "C:\\public\\data
    \\delphi", pos=(posCol2 + lblWidth, posWDir), size=(txtWidth * 2,
    -1))
#        self.txtWDir.SetInsertionPoint(0)
#
#        #rusin.txt File Name
#        self.lblCFile = wx.StaticText(self.pnlCtrl, -1, "Conf. File",
    pos=(posCol2, posCFile), size=(lblWidth, -1))
#        self.txtCFile = wx.TextCtrl(self.pnlCtrl, -1, "rusin.txt", pos
    =(posCol2 + lblWidth, posCFile), size=(txtWidth, -1))
#        self.txtCFile.SetInsertionPoint(0)
#
#        #Save File Name
#        self.lblSFile = wx.StaticText(self.pnlCtrl, -1, "Save File",
    pos=(posCol2, posSFile), size=(lblWidth, -1))
```

```
#         self.txtSFile = wx.TextCtrl(self.pnlCtrl, -1, "test1.dat", pos
    =(posCol2 + lblWidth, posSFile), size=(txtWidth, -1))
#         self.txtSFile.SetInsertionPoint(0)
#
#         #rusin.txt File
#         self.lblConf = wx.StaticText(self.pnlCtrl, -1, "Conf.", pos=(
    posCol2, posConf))
#         self.txtConf = wx.TextCtrl(self.pnlCtrl, -1, "rusin.txt", pos
    =(posCol2 + lblWidth, posConf), size=(-1, mtxtHeight), style=wx.
    TE_MULTILINE)
#         self.txtConf.SetInsertionPoint(0)
#
#         #Edit Button
#         self.btnEdtConfF = wx.Button(self.pnlCtrl, label="Edit", pos=(
    posCol2 + lblWidth, posConfBtn), size=(btnWidth, btnHeight))
#         self.Bind(wx.EVT_BUTTON, self.editConf, self.btnEdtConfF)
#
#         #Update Button
#         self.btnUpdConfF = wx.Button(self.pnlCtrl, label="Update", pos
    =(posCol2 + lblWidth + btnWidth, posConfBtn), size=(btnWidth,
    btnHeight))
#         self.Bind(wx.EVT_BUTTON, self.updateConf, self.btnUpdConfF)
#
#         ## Progress Info##
#         #Current Temp
#         self.lblTemp = wx.StaticText(self.pnlCtrl, -1, "Curr. Temp",
    pos=(posCol3, posTemp), size=(lblWidth, -1))
#         self.txtTemp = wx.TextCtrl(self.pnlCtrl, -1, "22", pos=(
    posCol3 + lblWidth, posTemp), size=(txtWidth, -1))
#         self.txtTemp.SetInsertionPoint(0)
#         self.lblTempUnit = wx.StaticText(self.pnlCtrl, -1, " C", pos=(
    posCol3 + lblWidth + txtWidth, posTemp), size=(unitWidth, -1))
#
#
#         #Current Freq
#         self.lblFreq = wx.StaticText(self.pnlCtrl, -1, "Curr. Freq",
    pos=(posCol3, posFreq), size=(lblWidth, -1))
#         self.txtFreq = wx.TextCtrl(self.pnlCtrl, -1, "0.6", pos=(
    posCol3 + lblWidth, posFreq), size=(txtWidth, -1))
#         self.txtFreq.SetInsertionPoint(0)
#         self.lblFreqUnit = wx.StaticText(self.pnlCtrl, -1, " MHz", pos
    =(posCol3 + lblWidth + txtWidth, posFreq), size=(unitWidth, -1))
#
#         #Progress
#         self.lblPrgss = wx.StaticText(self.pnlCtrl, -1, "Progress",
    pos=(posCol3, posPrgss))
#         self.txtPrgss = wx.TextCtrl(self.pnlCtrl, -1, "Progress ...",
```

```
    pos=(posCol3 + lblWidth, posPrgss), size=(-1, mtxtHeight), style=wx.
    TE_MULTILINE)
#         self.txtPrgss.SetInsertionPoint(0)
#
#         #Start Button
#         self.btnStart = wx.Button(self.pnlCtrl, label="Start", pos=(
    posCol3, posRunBtn), size=(btnWidth, btnHeight))
#         self.Bind(wx.EVT_BUTTON, self.startScan, self.btnStart)
#
#         #Pause Button
#         self.btnPause = wx.Button(self.pnlCtrl, label="Pause", pos=(
    posCol3 + btnWidth, posRunBtn), size=(btnWidth, btnHeight))
#         self.Bind(wx.EVT_BUTTON, self.pauseScan, self.btnPause)
#
#         #Stop Button
#         self.btnStop = wx.Button(self.pnlCtrl, label="Stop", pos=(
    posCol3 + btnWidth * 2, posRunBtn), size=(btnWidth, btnHeight))
#         self.Bind(wx.EVT_BUTTON, self.stopScan, self.btnStop)
#
        #Menu Bar
        self.menuBar = wx.MenuBar()
        self.fileMenu = wx.Menu()
        self.editMenu = wx.Menu()
        self.helpMenu = wx.Menu()
        self.helpMenu.Append(wx.ID_ABOUT, "&About", "More information
            about this program")
        self.fileMenu.Append(wx.ID_EXIT, "E&xit", "Terminate the
            program")
        self.menuBar.Append(self.fileMenu, "&File");
        self.menuBar.Append(self.editMenu, "&Edit");
        self.menuBar.Append(self.helpMenu, "&Help");
        self.SetMenuBar(self.menuBar)

        #Events
        wx.EVT_MENU(self, wx.ID_ABOUT, self.OnAbout)
        wx.EVT_MENU(self, wx.ID_EXIT, self.closeApp)

    def startScan(self, event):
        start=float(self.txtSFreq.GetValue())
        stop=float(self.txtEFreq.GetValue())
        numPoints=int(self.txtPoints.GetValue())
        fName=self.txtSFile.GetValue()
        sens=float(self.txtSens.GetValue())
        dfSleep=float(self.txtSIntv.GetValue())
        numAve=int(self.txtAveg.GetValue())
        dsSleep=float(self.txtSIntv.GetValue())
        ampl=float(self.txtVoltage.GetValue())
```

```
        measT=self.chkMTemp.GetValue()
        fRange=self.cmbFRange.GetValue()

        rus=EXPR.EXPERIMENT(measT,fRange) #Creates experiment instance
        #rus.batch_scan()
        rus.single_scan(start,stop,numPoints,fName,ampl, sens, dfSleep,
            numAve, dsSleep)
        #Call scan start function

    def pauseScan(self, event):
        print "Scan Paused"
        #Call scan pause function

    def stopScan(self, event):
        print "Scan Stopped"
        #Call scan stop function

    def editConf(self, event):
        #Set txtConf active for edit
        print "Editing enabled"

    def updateConf(self, event):
        #Warning message
        #if(ok):
            #save current rusin.txt as rusin.temp
            #Update rusin.txt file with text in txtConf
        #else:
            #Do nothing
        print "Updated"


    def closeApp(self, event):
        dlg = wx.MessageDialog(self, "Do you want to Exit?", "Warning!"
            , wx.YES_NO)
        if dlg.ShowModal() == wx.ID_YES:
            dlg.Destroy()
            self.Close(True)
        else:
            dlg.Destroy()

    def closeWindow(self, event):
        self.Destroy()

    def OnAbout(self, event):
        print "About"
        dlg = wx.MessageDialog(self, "Resonant Ultrasound Spectroscopy\
            n"
```

```
                                    "Data Acquisition Program\n"
                                "RUS Lab - NCPA",
                                "University of Mississippi , USA", wx.OK)
        print dlg.ShowModal()
        dlg.Destroy()

class RUSGUI():
    def __init__(self):
        try:
            app = wx.PySimpleApp()
            frame = MyFrame(parent=None, id=-1, title="RUS-NCPA")
            frame.Show()
            app.MainLoop()
        except:
            print "GUI Error : Cannot Implement GUI"

gui=RUSGUI()
```

C.2.0.5 PLOTSCAN

```
from pylab import *
import wx as wx

class PLOTSCAN(wx.Frame):
    def __init__(self, parent, id, title):
        self.data=[]
        wx.Frame.__init__(self, parent, id, title, size=(100, 100))
#         self.Bind(wx.EVT_CLOSE, self.closeWindow)

        #Status Bar
        self.CreateStatusBar()
        self.SetStatusText("RUS Lab - NCPA")
        btnWidth = 65
        btnHeight = 25
        #Panels
        self.pnlCtrl = wx.Panel(self)
        #self.pnlPlot=wx.Panel(self)

        self.btnRefresh = wx.Button(self.pnlCtrl, label="Refresh Plot",
            pos=(10,10), size=(btnWidth, btnHeight))
        self.Bind(wx.EVT_BUTTON, self.plotScan, self.btnRefresh)

    def loadData(self,fName="dump.dat"):
        temp=loadtxt("dump.dat",delimiter='\t')
        self.data=[]
        self.data.append(temp[:,0])
        self.data.append(temp[:,1])
        self.data.append(temp[:,2])
```

```
    def plotScan(self,fName="dump.dat"):
        close('all')
        self.loadData(fName)
        plot(self.data[0],self.data[1],"-", label="Y")
        plot(self.data[0],self.data[2],"-", label="Y90")
        title("RUS Scan - Spectrum")
        xlabel("Frequency (MHz)")
        ylabel("Voltage (V)")
        show()

class PlotGUI():
    def __init__(self):
        app = wx.PySimpleApp()
        frame = PLOTSCAN(parent=None, id=-1, title="RUS-NCPA")
        frame.Show()
        app.MainLoop()

plotGUI=PlotGUI()
```

C.3 Extracting Peak Data

```
from pylab import *
import scipy.optimize as optimize
import tkFileDialog
import os

def selectFile():
    file_path = tkFileDialog.askopenfilename()
    return file_path

def loadData(fname='scan.dat',delim='\t',unpack=True):
    data=array(loadtxt(fname, delimiter=delim, unpack=unpack))
    freq=array(data[0])
    amp00=array(data[1])
    amp90=array(data[2])
    del(data)
    return freq,amp00,amp90

def getSlice(freq,amp00,amp90,doneList=[]):
    fig=figure()
    ax1=subplot(1,1,1)
    ax1.plot(freq,amp00,'r-',label='In-Phase')
    ax1.plot(freq,amp90,'k-',label='Quadrature Phase')
    for doneF in doneList:
        ax1.axvline(x=doneF, color='r')
```

```
    title("RUS Scan - Spectrum (SLECT RANGE AND CLOSE THE FIGURE) ")
    xlabel('Freq (Hz)')
    ylabel('Signal (V)')
    legend()
    show()

    fLow,fHigh = ax1.get_xlim() #get current axis limits

    nStart=len(freq[freq<fLow]) #start index
    nEnd=len(freq[freq<fHigh])#end index

    amp00Clip = amp00[nStart:nEnd]
    amp90Clip = amp90[nStart:nEnd]
    freqClip=freq[nStart:nEnd]

    close(fig)

    del(freq)
    del(amp00)
    del(amp90)

    return freqClip,amp00Clip,amp90Clip

def selectPeaks(freq, amp00,amp90):
    plot(freq, amp90)
    draw()
    title("RUS Scan - Spectrum")
    xlabel("Frequency (Hz)")
    ylabel("Signal (V)")
    p=[]
    for i in range(4):
        print "Select Peak  Frequency # %d ( max = 4): LEFT mouse
            button >"%(i+1)
        title("Select Peak  Frequency # %d  ( max = 4 ) : LEFT mouse
            button"%(i+1))
        draw()
        p1 = ginput(n=1, timeout=0,show_clicks=True, mouse_add=1,
            mouse_pop=2, mouse_stop=3)
        if len(p1)==0:
            break
        p.append(p1[0][0])
        axvline(x=p1[0][0], color='r',linestyle='-')
        draw()
    close('all')
    return array(p)

def lorentz(x,x0,A,phi,Q,y0):
```

```
    y=A*(x/x0*cos(phi)+(1.0-(x/x0)**2)*Q*sin(phi))/((x/x0)**2+(1.0-(x/
        x0)**2)**2*Q**2)+y0
    return y

def bkgrnd(x,x0,a0,a1,a2,a3):
    y0=a0+a1*(x-x0)+a2*(x-x0)**2+a3*(x-x0)**3
    return y0

def errfunc(p,x,y):
    return L(x,p)-y

def L(x,*p):
    peaky=0
    for i in range(int((len(p)-5)/5)):
        peaky=peaky+lorentz(x,p[i*5+0],p[i*5+1],p[i*5+2],p[i*5+3],p[i
            *5+4])

    #add background
    peaky=peaky+bkgrnd(x,p[-5],p[-4],p[-3],p[-2],p[-1])
    return peaky

def fitPeaks(freq, amp00, amp90 ,f0):
    fig=figure()
    #start and end of the window
    f1=min(freq)
    f2=max(freq)

    #estimated initial amplitude
    A0=max(amp90)-min(amp90)

    #initial phase
    phi0=0.1

    #starting voltage level
    y0=amp90[1]

    #starting background level (same as y0)
    a0=y0

    #starting frequency
    freq0=freq[0]

    #background parameters
    a1=-1e-7
    a2=1e-14
    a3=1e-16
```

```
 #peaks info [freq, ampl, phi, Q, y0,.....,a0,a1,a2,a3]
p0=[]
for i in range(len(f0)): #populate parameter list
    #estimatted initial Q
    Q0=f0[i]/(f2-f1)*10
    p0.append(f0[i])
    p0.append(A0)
    p0.append(phi0)
    p0.append(Q0)
    p0.append(y0)

#background parameters at the end of the list
p0.append(freq0)
p0.append(a0)
p0.append(a1)
p0.append(a2)
p0.append(a3)

try:
    pp,success = optimize.curve_fit(L,freq,amp90,p0)
    ppx,successx = optimize.curve_fit(L,freq,amp00,pp)
except:
    print "Fit algorithm failed Try again"
    return 0,0
#perr = sqrt(diag(success)) #Errors of parameters
#print perr

#Generate signal with calculated parameters
F=linspace(min(freq),max(freq),len(freq)*10)
A90=0
A00=0
print "-"*50
print "Freq(MHz)\tQ\tA(V)\tphi(Rad)"
print "-"*50
for i in range(len(f0)):
    A90=A90+lorentz(F,pp[i*5+0],pp[i*5+1],pp[i*5+2],pp[i*5+3],pp[i
        *5+4])
    A00=A00+lorentz(F,ppx[i*5+0],ppx[i*5+1],ppx[i*5+2],ppx[i*5+3],
        ppx[i*5+4])
    print "%.6f\t%.2f\t%.2e\t%.3f"%(pp[i*5+0]/1e6,abs(pp[i*5+3]),
        abs(pp[i*5+1]),abs(pp[i*5+2]))
    print "%.6f\t%.2f\t%.2e\t%.3f"%(ppx[i*5+0]/1e6,abs(ppx[i*5+3]),
        abs(ppx[i*5+1]),abs(ppx[i*5+2]))

#add background
A90=A90+bkgrnd(F,pp[-5],pp[-4],pp[-3],pp[-2],pp[-1])
A00=A00+bkgrnd(F,ppx[-5],ppx[-4],ppx[-3],ppx[-2],ppx[-1])
```

```
    suptitle("RUS Peak Plot")

    subplot(2,2,1)
    title("Inphase")
    plot(freq, amp90)
    plot(F,A90,"r-")

    subplot(2,2,3)
    title("Quadrature")
    plot(freq, amp00)
    plot(F,A00,"r-")

    subplot(1,2,2)
    title("Cole-Cole")
    plot(amp00, amp90)
    plot(A00,A90,"r-")

    subplots_adjust(bottom=0.2)

    axSave = axes([0.81, 0.05, 0.1, 0.075])
    btnSave = Button(axSave, 'Save')
    msg=btnSave.on_clicked(saveFit)

    axFinish = axes([0.7, 0.05, 0.1, 0.075])
    btnFinish = Button(axFinish, 'Finish')
    msg=btnFinish.on_clicked(finishFit)

    axReFit = axes([0.59, 0.05, 0.1, 0.075])
    btnReFit = Button(axReFit, 'Re-Fit')
    msg=btnReFit.on_clicked(reDoFit)

    show()
    print msg

    return pp[0],abs(pp[3])

def reDoFit(event):
    print "Data Saved"
    return "Data Saved Return"

def saveFit(event):
    print "Data Saved"
    return "Data Saved Return"

def finishFit(event):
```

```
    print "Data Saved"
    print "Peak Fitting Done"
    return "Data Saved Return"

def output(FLIST,QLIST):
    for i in range(len(FLIST)):
        print "%.6f\t%.2f"%(FLIST[i]/1e6,QLIST[i])

filePath=selectFile()
print filePath
FLIST=[]
QLIST=[]

while(True):
    freq,amp00,amp90=loadData(filePath,unpack=True)
    freqClip,amp00Clip,amp90Clip=getSlice(freq, amp00,amp90,FLIST)
    f0=selectPeaks(freqClip,amp00Clip,amp90Clip)
    FF,QQ=fitPeaks(freqClip,amp00Clip,amp90Clip, f0)
    if FF>0:
        FLIST.append(FF)
        QLIST.append(QQ)
    output(FLIST,QLIST)
```

C.4 Fitting Elastic Constants

```
C*************************************************************
C
      PROGRAM XYZMRQ
C
C*************************************************************
      IMPLICIT INTEGER (I-N)
C
      PARAMETER (MMAX=20,MDAT=60)
C
      INTEGER IA(MMAX)
      REAL X(MDAT),Y(MDAT),A(MMAX),FDATA(MDAT)
      REAL SIG(MDAT),ALPHA(MMAX,MMAX),COVAR(MMAX,MMAX)
C
      REAL*8 RHO,WSAV(MDAT),WIMP(MDAT)
      CHARACTER*50 HEADER, USED(MDAT)*3
      EXTERNAL FREQS
C
      COMMON /RHO/ RHO
      COMMON /NN/ NN
      COMMON /NMDS/ NMDS
      COMMON /NDATA/ NDATA
```

```
      COMMON /IA/ IA
      COMMON /X/ X
      COMMON /WSAV/ WSAV,WIMP
      COMMON /SIG/ SIG
C0.467176
400   FORMAT(A50)
C
      TOL=0.05
C
      OPEN(3,FILE='xyzmrq.dat',STATUS='OLD',ERR=101)
      GO TO 102
101   WRITE(*,'(A)') ' Error opening xyzmrq.dat.'
      GO TO 9900
C
102   READ(3,400) HEADER
      READ(3,*,ERR=101) C11,C12,C44,E1,E2,E3
      READ(3,*,ERR=101) (IA(I),I=1,6)
      READ(3,*,ERR=101) RHO,NN,NMDS
      DO 103 I=1,NMDS
      READ(3,*,ERR=101) FDATA(I)
103   CONTINUE
      CLOSE(3,ERR=9900)
C
      NDATA=0
      DO 105 I=1,NMDS
      IF(FDATA(I).LE.0.0) THEN
        USED(I)='EXC'
        GO TO 104
      ELSE
        IF(FDATA(I).GT.0.0) USED(I)='INC'
      END IF
      NDATA=NDATA+1
      X(NDATA)=FLOAT(I)
      Y(NDATA)=FDATA(I)
      GO TO 105
104   IF(FDATA(I).EQ.0.0) USED(I)='---'
      FDATA(I)=-FDATA(I)
105   CONTINUE
C
C     Parameter setup
C
C       write(*,*) c11,c12,c44,E1,E2,E3
      CALL INIT
      MA=6
      A(1)=C11
      A(2)=C11-2.0*C44
      A(3)=C44
```

```
      A(4)=E1
      A(5)=E2
      A(6)=E1*E2*E3
C
      ALAMDA=-1.
      SUMSQ=1.D23
      ITER=0
      OPEN(3,FILE='mrqout.txt',STATUS='UNKNOWN')
      WRITE(3,400) HEADER
      WRITE(3,'(1H )')
C
200   ITER=ITER+1
      OLAMDA=ALAMDA
      CHI0=SUMSQ
      WRITE(*,'(A20,I2)') '     Iteration number',ITER
C
      CALL MRQMIN(X,Y,SIG,NDATA,A,IA,MA,COVAR,ALPHA,MMAX,
     & SUMSQ,FREQS,ALAMDA)
C
      CHANGE=ABS((SUMSQ-CHI0)/CHI0)
      WRITE(*,'(5E12.4)') OLAMDA,ALAMDA,CHI0,SUMSQ,CHANGE
      WRITE(3,'(5E12.4)') OLAMDA,ALAMDA,CHI0,SUMSQ,CHANGE
C
C     IF(CHANGE.GT.TOL) GO TO 200
C
      A(2)=A(1)-2.0*A(3)
      WRITE(*,'(6F12.5)') (A(I),I=1,5),A(6)/(A(4)*A(5))
      WRITE(3,'(6F12.5)') (A(I),I=1,5),A(6)/(A(4)*A(5))
      SUMERR=0.0
      DO 8200 I=1,NMDS
      ERR=0.0
      IF(FDATA(I).GT.0.) ERR=100.*((FDATA(I)-WSAV(I))/WSAV(I))
      WRITE(3,'(I3,2F10.6,F8.3,A6)') I,FDATA(I),WSAV(I),ERR,USED(I)
      IF (I.LE.NMDS) WRITE(*,'(I3,2F10.6,F8.3,A6)')
     & I,FDATA(I),WSAV(I),ERR,USED(I)
      IF (USED(I) .EQ. 'INC') SUMERR=SUMERR+ABS(ERR)
8200  CONTINUE
       ERRAVG=SUMERR/NDATA
      WRITE(*,'(A43,F8.3)')
     & 'Average error magnitude of included peaks= ', ERRAVG
      WRITE(3,'(A43,F8.3)')
     & 'Average error magnitude of included peaks= ', ERRAVG
          WRITE(*,*) 'ISOTROPIC MODEL'
         WRITE(3,*) 'ISOTROPIC MODEL'
         WRITE(*,*) 'E= ',(A(1)-A(2))*(2.*A(2)+A(1))/(A(1)+A(2))
         WRITE(*,*) 'G= ', A(3)
         WRITE(*,*) 'SIGMA= ', A(2)/(A(1)+A(2))
```

```
            WRITE(3,*)  'E= ',(A(1)-A(2))*(2.*A(2)+A(1))/(A(1)+A(2))
            WRITE(3,*)  'G= ', A(3)
            WRITE(3,*)  'SIGMA= ', A(2)/(A(1)+A(2))

8210
          IF(CHANGE.EQ.0) THEN
              GO TO 9000
          ELSE
              GO TO 200
          END IF
C
9000      CLOSE(3, ERR=9900)
9900      STOP
          END
C
C*************************************************************
C
          SUBROUTINE FREQS(XVAL,A,YMOD,DYDA,MA)
C
C*************************************************************
          IMPLICIT INTEGER (I-N)
          IMPLICIT REAL*8 (A-H,O-Z)
C
          PARAMETER (MMAX=20,MDAT=60)
C
          INTEGER IA(MMAX)
          REAL XVAL,A(MA),YMOD,DYDA(MA),A0,X(MDAT)
C
          COMMON /ALST/ ALST(MMAX),EPS(MMAX)
          COMMON /WSAV/ WSAV(MDAT),WIMP(MDAT)
          COMMON /DYDAA/ DYDAA(MMAX,MDAT)
          COMMON /NDATA/ NDATA
          COMMON /X/ X
          COMMON /IA/ IA
C
10        DO 11 I=1,MA
          IF(A(I).NE.ALST(I)) GO TO 100
11        CONTINUE
          GO TO 200
C
100       CALL XYZBLK(A,MA,WSAV)
C
          DO 112 I=1,MA
          IF(IA(I).EQ.0) GO TO 112
          A0=A(I)
          A(I)=A0*(1.D0+EPS(I))
          DA=A(I)-A0
```

```
      CALL XYZBLK(A,MA,WIMP)
      DO 111   J=1,NDATA
      JX=NINT(X(J))
      DYDAA(I,JX)=(WIMP(JX)-WSAV(JX))/DA
111   CONTINUE
      A(I)=A0
112   CONTINUE
C
      DO 113  I=1,MA
      ALST(I)=A(I)
113   CONTINUE
C
200   IX=NINT(XVAL)
      YMOD=WSAV(IX)
      DO 201  I=1,MA
      DYDA(I)=DYDAA(I,IX)
201   CONTINUE
C
      RETURN
      END
C
C*************************************************************
C
      SUBROUTINE XYZBLK(A,MA,WSAV)
C
C*************************************************************
      IMPLICIT INTEGER (I-N)
      IMPLICIT REAL*8 (A-H,O-Z)
C
      PARAMETER (MMAX=20,MDAT=60)
      PARAMETER(NRR=256,TWOPI=6.283185307795864769)
C
      REAL A(MA)
C
      DIMENSION LB(NRR),MB(NRR),NB(NRR),IC(NRR),LMN(3)
      DIMENSION E(NRR,NRR),GAMMA(NRR,NRR),W(NRR),V(NRR,NRR),P(NRR)
      DIMENSION D(NRR),WSAV(MDAT)
C
      COMMON /RHO/ RHO
      COMMON /NN/ NN
      COMMON /NMDS/ NMDS
      COMMON /CIJKL/ C(3,3,3,3)
      COMMON /IA/ IA(MMAX)
C

      A(2)=A(1)-2.0*A(3)
```

```
C
C       C11, C22, C33
41      C(1,1,1,1)=A(1)
        C(2,2,2,2)=A(1)
        C(3,3,3,3)=A(1)
C       C12, C23, C13
        C(1,1,2,2)=A(2)
        C(2,2,1,1)=A(2)
        C(1,1,3,3)=A(2)
        C(3,3,1,1)=A(2)
        C(2,2,3,3)=A(2)
        C(3,3,2,2)=A(2)
C       C44
        C(2,3,2,3)=A(3)
        C(2,3,3,2)=A(3)
        C(3,2,2,3)=A(3)
        C(3,2,3,2)=A(3)
C       C55
        C(1,3,1,3)=A(3)
        C(1,3,3,1)=A(3)
        C(3,1,1,3)=A(3)
        C(3,1,3,1)=A(3)
C       C66
        C(2,1,2,1)=A(3)
        C(2,1,1,2)=A(3)
        C(1,2,2,1)=A(3)
        C(1,2,1,2)=A(3)
C
        E1=A(4)
        E2=A(5)
        E3=A(6)/(E1*E2)
        NNP1=NN+1
        NNP3=NN+3
C
        DO 43 I=1,NMDS
        WSAV(I)=1.D64
43      CONTINUE
C
C  normalization factors for the calculation of Gamma
C
        E11=4.D0/(E1*E1)
        E22=4.D0/(E2*E2)
        E33=4.D0/(E3*E3)
        E23=4.D0/(E2*E3)
        E13=4.D0/(E1*E3)
        E12=4.D0/(E1*E2)
C
```

```
C        Loop for eight blocks
C
         DO 8000 L0=1,2
         DO 8000 M0=1,2
         DO 8000 N0=1,2
C
         IG=0
         DO 2 I=1,3
         LMN(1)=L0
         LMN(2)=M0
         LMN(3)=N0
         LMN(I)=LMN(I)+1
         IF(LMN(I).EQ.3) LMN(I)=1
C
         DO 2 L=LMN(1),NNP1,2
         DO 2 M=LMN(2),NNP1,2
         DO 2 N=LMN(3),NNP1,2
         IF(L+M+N.GT.NNP3) GO TO 2
         IG=IG+1
         IC(IG)=I
         LB(IG)=L-1
         MB(IG)=M-1
         NB(IG)=N-1
2        CONTINUE
         NR=IG
C
C  calculate Gamma and E
C
         DO 3 IG=1,NR
         DO 3 JG=IG,NR
         E(IG,JG)=0.D0
         GAMMA(IG,JG)=0.D0
          I=IC(IG)
          J=IC(JG)
          LS=LB(IG)+LB(JG)
          MS=MB(IG)+MB(JG)
          NS=NB(IG)+NB(JG)
          GAMMA(IG,JG)=
     &   C(I,1,J,1)*DFLOAT(LB(IG)*LB(JG))*F(LS-2,MS,NS)*E11
     & +C(I,2,J,2)*DFLOAT(MB(IG)*MB(JG))*F(LS,MS-2,NS)*E22
     & +C(I,3,J,3)*DFLOAT(NB(IG)*NB(JG))*F(LS,MS,NS-2)*E33
     & +(C(I,1,J,2)*DFLOAT(LB(IG)*MB(JG))+C(I,2,J,1)*
     & DFLOAT(MB(IG)*LB(JG)))*F(LS-1,MS-1,NS)*E12
     & +(C(I,1,J,3)*DFLOAT(LB(IG)*NB(JG))+C(I,3,J,1)*
     & DFLOAT(NB(IG)*LB(JG)))*F(LS-1,MS,NS-1)*E13
     & +(C(I,2,J,3)*DFLOAT(MB(IG)*NB(JG))+C(I,3,J,2)*
     & DFLOAT(NB(IG)*MB(JG)))*F(LS,MS-1,NS-1)*E23
```

```
C
      GAMMA(JG,IG)=GAMMA(IG,JG)
      IF(I.EQ.J) E(IG,JG)=F(LS,MS,NS)
      E(JG,IG)=E(IG,JG)
3     CONTINUE
C
      CALL CHOLDC(E,NR,NRR,P)
C
      OOP=1.D0/P(1)
      DO 21 I=1,NR
      V(I,1)=GAMMA(I,1)*OOP
21    CONTINUE
      DO 23 J=2,NR
      OOP=1.D0/P(J)
      DO 23 I=J,NR
      SUM=GAMMA(I,J)
      DO 22 K=1,J-1
      SUM=SUM-V(I,K)*E(J,K)
22    CONTINUE
      V(I,J)=SUM*OOP
23    CONTINUE
      GAMMA(1,1)=V(1,1)/P(1)
      DO 25 I=2,NR
      SUM=V(I,1)
      DO 24 K=1,I-1
      SUM=SUM-E(I,K)*GAMMA(K,1)
24    CONTINUE
      GAMMA(I,1)=SUM/P(I)
      GAMMA(1,I)=GAMMA(I,1)
25    CONTINUE
      DO 27 J=2,NR
      DO 27 I=J,NR
      SUM=V(I,J)
      DO 26 K=1,I-1
      SUM=SUM-E(I,K)*GAMMA(J,K)
26    CONTINUE
      GAMMA(I,J)=SUM/P(I)
      GAMMA(J,I)=GAMMA(I,J)
27    CONTINUE
C
      CALL TRED2(GAMMA,NR,NRR,W,D)
C
      CALL TQLI(W,D,NR,NRR,GAMMA)
C
C     Save lowest frequencies.
C
60    DO 64 J=1,NR
```

```
      W1=W(J)
      IF(W1.LT.1.D-4) GO TO 64
      DIFMAX=-1.D0
      DO 61 I=1,NMDS
      DIF=WSAV(I)-W1
      IF(DIF.LE.DIFMAX) GO TO 61
      DIFMAX=DIF
      IMAX=I
61    CONTINUE
      IF(DIFMAX.LE.0.D0) GO TO 64
      WSAV(IMAX)=W1
64    CONTINUE
C
8000  CONTINUE
C
C     Sort frequencies.
C
      DO 8012 I=1,NMDS-1
      K=I
      TMP=WSAV(I)
      DO 8011 J=I+1,NMDS
      IF(WSAV(J).GT.TMP) GO TO 8011
      K=J
      TMP=WSAV(J)
8011  CONTINUE
      IF(K.EQ.I) GO TO 8012
      WSAV(K)=WSAV(I)
      WSAV(I)=TMP
8012  CONTINUE
C
      DO 8020 I=1,NMDS
      WSAV(I)=DSQRT(WSAV(I)/RHO)/TWOPI
8020  CONTINUE
C
9000  RETURN
      END
C
C*************************************************************
C
      SUBROUTINE INIT
C
C*************************************************************
      IMPLICIT INTEGER (I-N)
      IMPLICIT REAL*8 (A-H,O-Z)
C
      PARAMETER (MMAX=20,MDAT=60)
C
```

```
      REAL SIG(MDAT)
C
      COMMON /ALST/ ALST(MMAX),EPS(MMAX)
      COMMON /CIJKL/ C(3,3,3,3)
      COMMON /DYDAA/ DYDAA(MMAX,MDAT)
      COMMON /SIG/ SIG
C
      DO 11 I=1,MMAX
      ALST(I)=0.D0
      EPS(I)=1.D-5
11    CONTINUE
C
      DO 12 I=1,3
      DO 12 J=1,3
      DO 12 K=1,3
      DO 12 L=1,3
      C(I,J,K,L)=0.D0
12    CONTINUE
C
      DO 13 I=1,MMAX
      DO 13 J=1,MDAT
      DYDAA(I,J)=0.D0
13    CONTINUE
C
      DO 14 I=1,MDAT
      SIG(I)=1.D0
14    CONTINUE
C
      RETURN
      END
C
C*****************************************************************
C
      FUNCTION F(IP,IQ,IR)
C
C*****************************************************************
      IMPLICIT INTEGER (I-N)
      IMPLICIT REAL*8 (A-H,O-Z)
C
      F=0.00D+00
      IF((IP.LT.0).OR.(IQ.LT.0).OR.(IR.LT.0)) RETURN
      IF((MOD(IP,2).NE.0).OR.(MOD(IQ,2).NE.0).OR.(MOD(IR,2).NE.0))
     & RETURN
C
      IP1=IP+1
      IQ1=IQ+1
      IR1=IR+1
```

```
C
      F=8.D0/DFLOAT(IP1*IQ1*IR1)
      RETURN
      END
C
C*****************************************************************
C*****************************************************************
C
      SUBROUTINE CHOLDC(A,N,NP,P)
        IMPLICIT INTEGER (I-N)
        IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 A(NP,NP),P(N)
      DO 13 I=1,N
        DO 12 J=I,N
          SUM=A(I,J)
          DO 11 K=I-1,1,-1
            SUM=SUM-A(I,K)*A(J,K)
11        CONTINUE
          IF(I.EQ.J)THEN
            IF(SUM.LE.0.)PAUSE 'CHOLDC FAILED'
            P(I)=DSQRT(SUM)
          ELSE
            A(J,I)=SUM/P(I)
          ENDIF
12      CONTINUE
13    CONTINUE
      RETURN
      END
C
      SUBROUTINE TRED2(A,N,NP,D,E)
        IMPLICIT INTEGER (I-N)
        IMPLICIT REAL*8 (A-H,O-Z)
      REAL*8 A(NP,NP),D(NP),E(NP)
      DO 18 I=N,2,-1
        L=I-1
        H=0.
        SCALE=0.
        IF(L.GT.1)THEN
          DO 11 K=1,L
            SCALE=SCALE+DABS(A(I,K))
11        CONTINUE
          IF(SCALE.EQ.0.)THEN
            E(I)=A(I,L)
          ELSE
            DO 12 K=1,L
              A(I,K)=A(I,K)/SCALE
              H=H+A(I,K)**2
```

```
12              CONTINUE
                F=A(I,L)
                G=-DSIGN(DSQRT(H),F)
                E(I)=SCALE*G
                H=H-F*G
                A(I,L)=F-G
                F=0.
                DO 15 J=1,L
C       OMIT FOLLOWING LINE IF FINDING ONLY EIGENVALUES
C                 A(J,I)=A(I,J)/H
                  G=0.
                  DO 13 K=1,J
                    G=G+A(J,K)*A(I,K)
13                CONTINUE
                  DO 14 K=J+1,L
                    G=G+A(K,J)*A(I,K)
14                CONTINUE
                  E(J)=G/H
                  F=F+E(J)*A(I,J)
15              CONTINUE
                HH=F/(H+H)
                DO 17 J=1,L
                  F=A(I,J)
                  G=E(J)-HH*F
                  E(J)=G
                  DO 16 K=1,J
                    A(J,K)=A(J,K)-F*E(K)-G*A(I,K)
16                CONTINUE
17              CONTINUE
              ENDIF
            ELSE
              E(I)=A(I,L)
            ENDIF
            D(I)=H
18      CONTINUE
C       OMIT FOLLOWING LINE IF FINDING ONLY EIGENVALUES.
C        D(1)=0.
        E(1)=0.
        DO 24 I=1,N
C       DELETE LINES FROM HERE ...
C         L=I-1
C         IF(D(I).NE.0.)THEN
C           DO 22 J=1,L
C             G=0.
C             DO 19 K=1,L
C               G=G+A(I,K)*A(K,J)
C19           CONTINUE
```

```
C                DO 21 K=1,L
C                   A(K,J)=A(K,J)-G*A(K,I)
C21              CONTINUE
C22            CONTINUE
C            ENDIF
C        ... TO HERE WHEN FINDING ONLY EIGENVALUES.
           D(I)=A(I,I)
C        ALSO DELETE LINES FROM HERE ...
C           A(I,I)=1.
C           DO 23 J=1,L
C              A(I,J)=0.
C              A(J,I)=0.
C23         CONTINUE
C        ... TO HERE WHEN FINDING ONLY EIGENVALUES.
24       CONTINUE
         RETURN
         END
C
         SUBROUTINE TQLI(D,E,N,NP,Z)
           IMPLICIT INTEGER (I-N)
           IMPLICIT REAL*8 (A-H,O-Z)
         REAL*8 D(NP),E(NP),Z(NP,NP)
CU       USES DPYTHAG
         DO 11 I=2,N
           E(I-1)=E(I)
11       CONTINUE
         E(N)=0.
         DO 15 L=1,N
           ITER=0
1          DO 12 M=L,N-1
             DD=DABS(D(M))+DABS(D(M+1))
             IF (DABS(E(M))+DD.EQ.DD) GOTO 2
12         CONTINUE
           M=N
2          IF(M.NE.L)THEN
             IF(ITER.EQ.30)PAUSE 'TOO MANY ITERATIONS IN TQLI'
             ITER=ITER+1
             G=(D(L+1)-D(L))/(2.*E(L))
             R=DPYTHAG(G,1.D0)
             G=D(M)-D(L)+E(L)/(G+DSIGN(R,G))
             S=1.
             C=1.
             P=0.
             DO 14 I=M-1,L,-1
               F=S*E(I)
               B=C*E(I)
               R=DPYTHAG(F,G)
```

```
                E(I+1)=R
                IF (R.EQ.0.) THEN
                  D(I+1)=D(I+1)-P
                  E(M)=0.
                  GOTO 1
                ENDIF
                S=F/R
                C=G/R
                G=D(I+1)-P
                R=(D(I)-G)*S+2.*C*B
                P=S*R
                D(I+1)=G+P
                G=C*R-B
C       OMIT LINES FROM HERE ...
C                DO 13 K=1,N
C                  F=Z(K,I+1)
C                  Z(K,I+1)=S*Z(K,I)+C*F
C                  Z(K,I)=C*Z(K,I)-S*F
C13              CONTINUE
C       ... TO HERE WHEN FINDING ONLY EIGENVALUES.
14              CONTINUE
                D(L)=D(L)-P
                E(L)=G
                E(M)=0.
                GOTO 1
              ENDIF
15      CONTINUE
        RETURN
        END
C
        FUNCTION DPYTHAG(A,B)
        DOUBLE PRECISION A,B,DPYTHAG
        DOUBLE PRECISION ABSA,ABSB
        ABSA=DABS(A)
        ABSB=DABS(B)
        IF (ABSA.GT.ABSB) THEN
          DPYTHAG=ABSA*DSQRT(1.0D0+(ABSB/ABSA)**2)
        ELSE
          IF (ABSB.EQ.0.0D0) THEN
            DPYTHAG=0.0D0
          ELSE
            DPYTHAG=ABSB*DSQRT(1.0D0+(ABSA/ABSB)**2)
          ENDIF
        ENDIF
        RETURN
        END
C
```

```
      SUBROUTINE mrqmin(x,y,sig,ndata,a,ia,ma,covar,alpha,nca,chisq,
     *funcs,alamda)
      INTEGER ma,nca,ndata,ia(ma),MMAX
      REAL alamda,chisq,a(ma),alpha(nca,nca),covar(nca,nca),
     *sig(ndata),x(ndata),y(ndata)
      EXTERNAL funcs
      PARAMETER (MMAX=20)
CU    USES covsrt,gaussj,mrqcof
      INTEGER j,k,l,m,mfit
      REAL ochisq,atry(MMAX),beta(MMAX),da(MMAX)
      SAVE ochisq,atry,beta,da,mfit
      if(alamda.lt.0.)then
        mfit=0
        do 11 j=1,ma
          if (ia(j).ne.0) mfit=mfit+1
11      continue
        alamda=0.001
        call mrqcof(x,y,sig,ndata,a,ia,ma,alpha,beta,nca,chisq,funcs)
        ochisq=chisq
        do 12 j=1,ma
          atry(j)=a(j)
12      continue
      endif
      j=0
      do 14 l=1,ma
        if(ia(l).ne.0) then
          j=j+1
          k=0
          do 13 m=1,ma
            if(ia(m).ne.0) then
              k=k+1
              covar(j,k)=alpha(j,k)
            endif
13        continue
          covar(j,j)=alpha(j,j)*(1.+alamda)
          da(j)=beta(j)
        endif
14    continue
      call gaussj(covar,mfit,nca,da,1,1)
      if(alamda.eq.0.)then
        call covsrt(covar,nca,ma,ia,mfit)
        return
      endif
      j=0
      do 15 l=1,ma
        if(ia(l).ne.0) then
          j=j+1
```

```
          atry(l)=a(l)+da(j)
        endif
15    continue
      call mrqcof(x,y,sig,ndata,atry,ia,ma,covar,da,nca,chisq,funcs)
      if(chisq.lt.ochisq)then
        alamda=0.1*alamda
        ochisq=chisq
        j=0
        do 17 l=1,ma
          if(ia(l).ne.0) then
            j=j+1
            k=0
            do 16 m=1,ma
              if(ia(m).ne.0) then
                k=k+1
                alpha(j,k)=covar(j,k)
              endif
16          continue
            beta(j)=da(j)
            a(l)=atry(l)
          endif
17      continue
      else
        alamda=10.*alamda
        chisq=ochisq
      endif
      return
      END
C
      SUBROUTINE mrqcof(x,y,sig,ndata,a,ia,ma,alpha,beta,nalp,chisq,
     *funcs)
      INTEGER ma,nalp,ndata,ia(ma),MMAX
      REAL chisq,a(ma),alpha(nalp,nalp),beta(ma),sig(ndata),x(ndata),
     *y(ndata)
      EXTERNAL funcs
      PARAMETER (MMAX=20)
      INTEGER mfit,i,j,k,l,m
      REAL dy,sig2i,wt,ymod,dyda(MMAX)
      mfit=0
      do 11 j=1,ma
        if (ia(j).ne.0) mfit=mfit+1
11    continue
      do 13 j=1,mfit
        do 12 k=1,j
          alpha(j,k)=0.
12      continue
        beta(j)=0.
```

```
13      continue
        chisq=0.
        do 16 i=1,ndata
          call funcs(x(i),a,ymod,dyda,ma)
          sig2i=1./(sig(i)*sig(i))
          dy=y(i)-ymod
          j=0
          do 15 l=1,ma
            if(ia(l).ne.0) then
              j=j+1
              wt=dyda(l)*sig2i
              k=0
              do 14 m=1,l
                if(ia(m).ne.0) then
                  k=k+1
                  alpha(j,k)=alpha(j,k)+wt*dyda(m)
                endif
14            continue
              beta(j)=beta(j)+dy*wt
            endif
15        continue
          chisq=chisq+dy*dy*sig2i
16      continue
        do 18 j=2,mfit
          do 17 k=1,j-1
            alpha(k,j)=alpha(j,k)
17        continue
18      continue
        return
        END
C
        SUBROUTINE gaussj(a,n,np,b,m,mp)
        INTEGER m,mp,n,np,NMAX
        REAL a(np,np),b(np,mp)
        PARAMETER (NMAX=50)
        INTEGER i,icol,irow,j,k,l,ll,indxc(NMAX),indxr(NMAX),ipiv(NMAX)
        REAL big,dum,pivinv
        do 11 j=1,n
          ipiv(j)=0
11      continue
        do 22 i=1,n
          big=0.
          do 13 j=1,n
            if(ipiv(j).ne.1)then
              do 12 k=1,n
                if (ipiv(k).eq.0) then
                  if (abs(a(j,k)).ge.big)then
```

```
                big=abs(a(j,k))
                irow=j
                icol=k
              endif
            else if (ipiv(k).gt.1) then
              pause 'singular matrix in gaussj'
            endif
12          continue
          endif
13      continue
        ipiv(icol)=ipiv(icol)+1
        if (irow.ne.icol) then
          do 14 l=1,n
            dum=a(irow,l)
            a(irow,l)=a(icol,l)
            a(icol,l)=dum
14        continue
          do 15 l=1,m
            dum=b(irow,l)
            b(irow,l)=b(icol,l)
            b(icol,l)=dum
15        continue
        endif
        indxr(i)=irow
        indxc(i)=icol
        if (a(icol,icol).eq.0.) pause 'singular matrix in gaussj'
        pivinv=1./a(icol,icol)
        a(icol,icol)=1.
        do 16 l=1,n
          a(icol,l)=a(icol,l)*pivinv
16      continue
        do 17 l=1,m
          b(icol,l)=b(icol,l)*pivinv
17      continue
        do 21 ll=1,n
          if(ll.ne.icol)then
            dum=a(ll,icol)
            a(ll,icol)=0.
            do 18 l=1,n
              a(ll,l)=a(ll,l)-a(icol,l)*dum
18          continue
            do 19 l=1,m
              b(ll,l)=b(ll,l)-b(icol,l)*dum
19          continue
          endif
21      continue
22    continue
```

```
      do 24 l=n,1,-1
        if(indxr(l).ne.indxc(l))then
          do 23 k=1,n
            dum=a(k,indxr(l))
            a(k,indxr(l))=a(k,indxc(l))
            a(k,indxc(l))=dum
23        continue
        endif
24    continue
      return
      END
C
      SUBROUTINE covsrt(covar,npc,ma,ia,mfit)
      INTEGER ma,mfit,npc,ia(ma)
      REAL covar(npc,npc)
      INTEGER i,j,k
      REAL swap
      do 12 i=mfit+1,ma
        do 11 j=1,i
          covar(i,j)=0.
          covar(j,i)=0.
11      continue
12    continue
      k=mfit
      do 15 j=ma,1,-1
        if(ia(j).ne.0)then
          do 13 i=1,ma
            swap=covar(i,k)
            covar(i,k)=covar(i,j)
            covar(i,j)=swap
13        continue
          do 14 i=1,ma
            swap=covar(k,i)
            covar(k,i)=covar(j,i)
            covar(j,i)=swap
14        continue
          k=k-1
        endif
15    continue
      return
      END
```

C.5 Analyzing Temperature and Pressure Dependent Resonance Data

```
,,,
```

RUS Lab – NCPA, University of Mississippi

```
*This program reads in following files
  [1] 'temp_hdr.dat' = Contains all the sample dimensions and initial
     fitting parameters
        Sample Tag
        [C11], [C12], [C44], [e1], [e2], [e3]
        1,0,1,0,0,0
        [rho], [order of fit], [peaks]
        [CTE]

  C is stiffness matrix in GPa
  e is for dimension in cm
  3rd line is enable-1/disable-0 fit variables
  Density (rho) is in g/cm^-1
  Default value of the 'order of fit'=12
  'Peaks' is number of peaks need to consider for fitting
  Coefficient of thermal expansion (CTE) is dimension less.


  [2] 'temp_freqs.dat' = Contains lists of resonance frequencies at
     each temperature (spectra file) in the following format
        temp1   temp2    temp3
        fr1     fr1      fr1
        fr2     fr2      fr2
        ...
  All the temperatures in celcius (C) and frequencies in MHz

*Compiled fortran code 'auto_isomrq_varC44.exe' will be used to run
   fitting algorithm
*Output results will be saved in the output folder

'''


__author__="Sumudu Tennakoon"
__email__="sptennak@go.olemiss.edu"
__date__ ="$Feb 1, 2013 4:58:26 PM$"

from pylab import *
import os
import time

RUS_FIT_EXEC='./cubicmrq.exe' #executable fitting algorithm
RUS_FIT_INPUT='xyzmrq.dat' #input file for fitting algorithm (
   autogenerated by following program)
TEMP_SPECTRA='temp_freqs.dat' #input file for fitting termperature data
TEMP_HDR='temp_hdr.dat'
FIT_SPECTRA='fit_freqs.dat'
```

```
FIT_ERROR='fit_error.dat'
RESULTS_FILE='fit_results.txt'
MRQ_DUMP='mrqout.txt'
OUT_DIR='output/'

class RUSDATA:
    def __init__(self):
        self.SampleTag="Sample"
        #Elastic constants
        self.C11=0.0
        self.C12=0.0
        self.C44=0.0
        #Sample dimensions
        self.E1=0.0
        self.E2=0.0
        self.E3=0.0
        self.RHO=0 #Density
        #Variability of parameters
        self.varC11=0
        self.varC12=0
        self.varC44=0
        self.varE1=0
        self.varE2=0
        self.varE3=0
        #FitParameters
        self.OP=12 #Order of polynomial
        self.N=10 #Number of peaks
        self.CTE=0.0
        self.temps=[]
        self.freqs=[]

    def readData(self):
        #Open header file
        f = open(TEMP_HDR, 'r')
        tempHDR=f.readlines()
        f.close()#release file object
        #Open spectrum file
        f = open(TEMP_SPECTRA, 'r')
        tempSpectra = array([line.split('\t') for line in f],dtype='
            float')
        f.close() #release file object
        self.SampleTag=tempHDR[0][:-2] #Sample and test information
        #Initial elastic constants and dimensions
        tempHDR[1]=array(tempHDR[1].split(','),dtype='float')
        #C0=split(tempHDR[1],',')
        self.C11=tempHDR[1][0] # unit: GPa
        self.C12=tempHDR[1][1] # unit: GPa
```

```
        self.C44=tempHDR[1][2] # unit: GPa
        self.E1=tempHDR[1][3] # unit: cm
        self.E2=tempHDR[1][4] # unit: cm
        self.E3=tempHDR[1][5] # unit: cm
        #variability of parameters
        tempHDR[2]=array(tempHDR[2].split(','),dtype='byte')
        self.varC11=tempHDR[2][0]
        self.varC12=tempHDR[2][1]
        self.varC44=tempHDR[2][2]
        self.varE1=tempHDR[2][3]
        self.varE2=tempHDR[2][4]
        self.varE3=tempHDR[2][5]
        #Density and fit parameters
        tempHDR[3]=array(tempHDR[3].split(','))
        self.RHO=float(tempHDR[3][0]) #Density    unit: g/cm^-3
        self.OP=int(tempHDR[3][1]) #Order of polynomial
        self.N=int(tempHDR[3][2]) #Number of frequencies
        self.CTE=float(tempHDR[4])  # unit: 1/C
        #Update temperature and frequency data in data object
        self.temps=tempSpectra[0]
        self.freqs=tempSpectra[1:].T
        del(tempHDR) #delete temporary array
        del(tempSpectra) #delete temporary array

    def plotFreqs(self,n=0):
        fig=plt.figure(n)
        i=0
        for fr in self.freqs.T:
                plot(self.temps,fr,'o-',label="Mode␣%d"%i)
                i=i+1
        xlabel('Temperature␣(C)')
        ylabel('Frequency␣(MHz)')
        legend()


class RUSFIT:
    def __init__(self):
        self.T=0.0
        self.E=0.0
        self.G=0.0
        self.n=0.0
        self.C11=0.0
        self.C12=0.0
        self.C44=0.0
        self.aveError=0.0
        self.inFreqs=[]
        self.fitFreqs=[]
```

```
        self.fitErrors=[]
        self.data=RUSDATA()

    def readData(self):
        self.data.readData()

    def displayFitInfo(self):
        text="#"*70+"\n"
        text=text+"\n"+"High Temperature RUS : Data Fitting : RUS Lab -
            NCPA"
        text=text+"\n"+"\n"+"#"*70+"\n"
        text=text+"\n"+" Elastic constants(Initial Parameters)"
        text=text+"\n"+"C11=%.2f GPa\nC12=%.2f GPa\nC44=%.2f GPa"%(self
            .data.C11,self.data.C12,self.data.C44)
        text=text+"\n"+"-"*70
        text=text+"\n"+"Order of Polynomial"
        text=text+"\n"+"OP=%d"%self.data.OP
        text=text+"\n"+"-"*70
        text=text+"\n"+"Number of Peaks"
        text=text+"\n"+"N=%d"%self.data.N
        text=text+"\n"+"_"*70

        return text

    def displayExpansionInfo(self):
        text="\n"+ "Thermal expansion of the sample"
        text=text+"\n"+ "T(C)  \tE1(cm)\tE2(cm)\tE3(cm)\tRHO(g/cm^3)\t"
        text=text+"\n"+"-"*70
        for i in range(len(self.data.RHO)):
            text=text+"\n"+"%06.2f\t%.4f\t%.4f\t%.4f\t%.4f\t"%(self.
                data.temps[i],self.data.E1[i],self.data.E2[i],self.data.
                E3[i],self.data.RHO[i])
        text=text+"\n"+"_"*70
        return text

    def calcThermExpansion(self,t0,t,CTE,E1,E2,E3,RHO):
        dt=t-t0  #temperature difference (unit: C)
        M=RHO*E1*E2*E3
#       M=9.1387
        E1=E1*(1.0+CTE*dt)#New length (unit:cm)
        E2=E2*(1.0+CTE*dt)
        E3=E3*(1.0+CTE*dt)
        RHO=M/(E1*E2*E3)
        return E1,E2,E3,RHO

    def computeDims(self):
        t=self.data.temps
```

```
        t0=t[0]
        CTE=self.data.CTE
        E1=self.data.E1
        E2=self.data.E2
        E3=self.data.E3
        RHO=self.data.RHO
        NT=len(t)
        e1=zeros(NT)
        e2=zeros(NT)
        e3=zeros(NT)
        rho=zeros(NT)
        for i in range(NT):
            e1[i],e2[i],e3[i],rho[i]=self.calcThermExpansion(t0,t[i],
                CTE,E1,E2,E3,RHO)
        self.data.E1=e1
        self.data.E2=e2
        self.data.E3=e3
        self.data.RHO=rho

    def genFitFile(self,i=0,C11i=0,C12i=0,C44i=0):
        if i==0:
            C11i=self.data.C11
            C12i=self.data.C12
            C44i=self.data.C44
        print "Initial Cij : %f\t%f\t%f"%( C11i,  C12i, C44i)
        text=self.data.SampleTag+": Temperature = %.2f"%self.data.temps
            [i]
        #Cij values were divided by 100 to convert GPaa->mbar
        text=text+"\n"+"%.4f,%.4f,%.4f,%.4f,%.4f,%.4f"%(C11i/100,C12i
            /100,C44i/100,self.data.E1[i],self.data.E2[i],self.data.E3[i
            ])
        #text=text+"\n"+"%.4f,%.4f,%.4f,%.4f,%.4f,%.4f"%(self.data.C11
            /100,self.data.C12/100,self.data.C44/100,self.data.E1[i],
            self.data.E2[i],self.data.E3[i])
        text=text+"\n"+"%d,%d,%d,%d,%d,%d"%(self.data.varC11,self.data.
            varC12,self.data.varC44,self.data.varE1,self.data.varE2,self
            .data.varE3)
        text=text+"\n"+"%.4f,%d,%d"%(self.data.RHO[i],self.data.OP,self
            .data.N)
        for f in self.data.freqs[i]:
            text=text+"\n"+"%f"%f
        return text

    def updateFitResults(self,results):
        self.T=results[0]
        self.E=results[1]
        self.G=results[2]
```

```
        self.n=results[3]
        self.C11=results[4]
        self.C12=results[5]
        self.C44=results[6]
        self.aveError=results[7]

    def displayFitOutput(self,out):
        text=""
        for i in range(len(self.T)):
            text=text+"%.2f\t"%self.T[i]
        text=text+"\n"
        for j in range(int(self.data.N)):
            for k in range(len(self.T)):
                text=text+"%f\t"%out[k][j]
            text=text+"\n"
        return text

    def displayResults(self):
        text="#"*70
        text=text+"\n"+"%s : Fit Results"%self.data.SampleTag
        text=text+"\n"+" units: T (C); E,G,C11,C12,C44 (GPa)"
        text=text+"\n"+"#"*70+"\n"
        text=text+"\n"+"T\tE\tG\tn\tC11\tC12\tC44\tAve.Err."
        text=text+"\n"+"-"*70
        for i in range(len(self.T)):
            text=text+"\n"+"%05.1f\t%.2f\t%.2f\t%.3f\t%.2f\t%.2f\t%.2f\
               t%.3f"%(self.T[i],self.E[i],self.G[i],self.n[i],self.C11
               [i],self.C12[i],self.C44[i],self.aveError[i])
        text=text+"\n"+"_"*70
        return text

    def plotTable(self, xVals, yVals, xLabel="X", YLabel="Y", pStyle="o
       -", pTitile="Plot X vs. Y", figNum=0 ):
        fig=plt.figure(figNum)
        i=0
        for i in range(len(yVals)):
            plot(xVals, yVals[i], pStyle, label="Mode %d"%i)
            i=i+1
        xlabel(xLabel)
        ylabel(YLabel)
        title(pTitile)
        legend()

    def plotResults(self,n=0):
        fig=plt.figure(n)
        suptitle(self.data.SampleTag) #main title to subplots
        subplot(221)
```

```
        plot(self.T,self.E,"b-o",label="Young's Modulus")
        xlabel('Temperature (C)')
        ylabel('E (GPa)')
        legend()
        subplot(222)
        plot(self.T,self.G,"g-o",label="Shear Modulus")
        xlabel('Temperature (C)')
        ylabel('G (GPa)')
        legend()
        subplot(223)
        plot(self.T,self.n,"r-o",label="Poisson's Ratio")
        xlabel('Temperature (C)')
        ylabel('n')
        legend()
        ylim(min(self.n)*.95,max(self.n)*1.05)
        subplot(224)
        plot(self.T,self.aveError,"k-o",label="Average Error %")
        xlabel('Temperature (C)')
        ylabel('%')
        legend()
        ylim(min(self.aveError)*.95,max(self.aveError)*1.05)

    def runFit(self,plotIt=False):
        start=time.time()
        #Read frequency data file and header file
        self.data.readData()
        print "Data file read ..."
        print self.displayFitInfo() #Display fit information and
            initial parameters
        #Compute dimensions at each temperature
        self.computeDims()
        print self.displayExpansionInfo() #Display thermal expansion
            data
        print "Fiting algorithm started ..."
        results=[]
        for i in range(len(self.data.temps)):
            print "Current Temp: %.2f"%self.data.temps[i]
            #Input data file for the fitting algorithm
            if i == 0:
                fitFile=self.genFitFile(i) #Generate data file
            else:
                fitFile=self.genFitFile(i,C11,C12,C44) #Generate data
                    file
            f = open(RUS_FIT_INPUT, 'w')
            f.write(fitFile)
            f.close()
            mrqout=os.popen(RUS_FIT_EXEC).readlines()#Open pipe and
```

```
        execute fitting algorithm returns a file objet
    for line in mrqout[-7-self.data.N:]: #print last iteration
        of fitting
         print line[:-1]
    print "-"*70
    #Generate fit frequency and error lists
    inFreq=[]
    fitFreq=[]
    fitError=[]
    for line in mrqout[-5-self.data.N:-5]:
         inFreq.append(line.split()[1])
         fitFreq.append(line.split()[2])
         fitError.append(line.split()[3])
    self.inFreqs.append(inFreq)
    self.fitFreqs.append(fitFreq)
    self.fitErrors.append(fitError)
    #Write output dump file
    f = open("%sT%06.2f_%s"%(OUT_DIR, self.data.temps[i],
        MRQ_DUMP), 'w')
    f.write(''.join(mrqout)) #connect lines in output
    f.close()
    #Extract fit results from output string
    C11=float(mrqout[-(self.data.N+6)].split()[0])*100 #unit:
        GPa (*100 to convert output in mbar to GPa)
    C12=float(mrqout[-(self.data.N+6)].split()[1])*100 #unit:
        GPa
    C44=float(mrqout[-(self.data.N+6)].split()[2])*100 #unit:
        GPa
    E=float(mrqout[-3].split()[-1])*100 #unit:GPa
    G=C44 #unit:GPa
    n=mrqout[-1].split()[-1]
    aveError=mrqout[-5].split()[-1]
    results.append([self.data.temps[i],E,G,n,C11,C12,C44,
        aveError])
#Transpose results and convert to float array
results=array(results,dtype=float).T
print "Fit complete..."
print "_"*70
#Update Fit object variables
self.updateFitResults(results)
#Display Results
text=self.displayResults()
print text
#Write Fit results to a file
f = open("%s%s"%(OUT_DIR,RESULTS_FILE),'w')
f.write(text)
f.close()
```

```
        #update fit frequency output list
        self.inFreqs=array(self.inFreqs,dtype=float)
        self.fitFreqs=array(self.fitFreqs,dtype=float)
        self.fitErrors=array(self.fitErrors,dtype=float)
        print "_"*70
        #Generate display fit frequencies
        text= self.displayFitOutput(self.fitFreqs)
        #Wite output to text file
        f = open("%s%s"%(OUT_DIR,FIT_SPECTRA),'w')
        f.write(text)
        f.close()
        #Generate display fit errors
        text=self.displayFitOutput(self.fitErrors)
        #Wite output to text file
        f = open("%s%s"%(OUT_DIR,FIT_ERROR),'w')
        f.write(text)
        f.close()
        #calculate time elapsed
        elapsed=time.time()-start
        print "Elapsed time: %.3f"%elapsed
        #Plot Results
        if plotIt==True:
            self.plotResults(0)
            self.plotTable(self.T, self.fitFreqs.T, xLabel="Temperature
                (C)", YLabel="Fit Frequencies (MHz)", pStyle="o-",
                pTitile=" Fit Frequencies vs. Temperature",figNum=1)
            self.plotTable(self.T, self.fitErrors.T, xLabel="
                Temperature (C)", YLabel="Errors (%)", pStyle="o-",
                pTitile=" Fit Errors vs. Temperature",figNum=2)
        del(mrqout)
        show()
        return results

def main():
    rusFit=RUSFIT()
    results=rusFit.runFit(plotIt=True)
    show()
    return rusFit

main()
```

APPENDIX D

PULSE ECHO DATA ACQUISITION COMPUTER PROGRAMS

APPENDIX D

PULSE ECHO DATA ACQUISITION COMPUTER PROGRAMS

Details on computer interfacing to laboratory equipment were discussed in the appendix C. Computer programs written in Python for pulse echo data acquisition and analysis are listed below.

D.1 PULSE ECHO DATA ACQUISITION

```
#Project : Pulse-Echo Data Aquisition System
#Date Started : 06/16/2010
#Loc : RUS LAB, NCPA, University of Mississippi, MS USA
#ver : 1.0
#Author : Sumudu Tennakoon

import visa
import time
from pylab import *
import os
close('all')
ch="CH1"
sample="PMN-PT"
fpath=""
rm = visa.ResourceManager()
print rm.list_resources()
myinst = rm.open_resource('TCPIP0::130.74.33.199::INSTR')
print(myinst.query("*IDN?"))
#myinst=visa.instrument("GPIB1::6")
myinst.write("*IDN?")
print "Instrument : " + myinst.read()

myinst.write("DISplay?")
print "Display settings : " + myinst.read()
myinst.write("MATH?")
print "Math : " + myinst.read()
myinst.write("MEASUrement?")
print "MEASUrement : " + myinst.read()
```

```
#print '\n----------'
#FRAMES=int(raw_input("No. of Frames to scan>>>"))
#print '----------\n'

FRAMES=1 #number of frames
i=0 #starting frame
avgsignal=zeros(10000) #final waveform data

for i in range(FRAMES):
    start=time.time()
    try:
        myinst.write("DATa:SOUrce "+ch)
        myinst.write("DATa:SOUrce?") #returns data channel
        print "DATA:SOURCE : " + myinst.read()
        myinst.write("DATa:ENCDG ASCIi;WIDTH 2") #sets encoding method
            (ASCII integer)
        myinst.write("DATa:ENCDG?")
        print "DATA:ENCODING : " + myinst.read()
        myinst.write("HORIZONTAL:RECORDLENGTH 10000") #sets record
            length (10000 data points)
        myinst.write("DATa:START 1") #sets starting point to record
            data
        myinst.write("DATa:STOP 10000") #sets ending point to record
            data
        myinst.write("HEADER OFF") #no header information

    except:
        print "Error 01 !!!" #Initialization error

    elapsed1=time.time()
    #myinst.write("ACQUIRE?")
    #print "ACQUIRE: " + myinst.read()

    try:
        myinst.write("HORizontal:MAIn:SECdiv?") #returns time scale
        hscale=myinst.read()
        print "Horizontal Scale : " + hscale + " sec/div"

        myinst.write(ch+":SCAle?") #returns voltage scale for CH1
        vscale=myinst.read()
        print "Vertical Scale : " + vscale + " volts/div"

    except:
        print "Error 02 !!!" #scale read error

    elapsed2=time.time()
```

```
        #myinst.write("CH1:YUNIT?")
        #print myinst.read()
        try:
            myinst.write("ACQUIRE:STATE RUN") #start data acquisition
        except:
            print "Error3!!!"

        try:
            myinst.write("CURVe?") #returns waveform in the oscilloscope
                display
            scope=myinst.read()

            wavefm=zeros(10000)
            for j in range(10000):
                #print "[%d] %f "%(j,float(scope.split(',')[j]))
                wavefm[j]=float(scope.split(',')[j]) #splits ascii string
                    in to values
                j=j+1

            #myinst.write("WFMPRE:CH1:NR_PT?;YOFF?;XINCR?PTOFF?XUNIT?;YUNIT
                ?")
            #print myinst.read()

        except:
            print "Error 03 !!!"     #splitting ascii string error

        if(FRAMES >1):
            avgsignal=(avgsignal+wavefm) #add current waveform to final
                waveform and calculate average
        else:
            avgsignal=wavefm

        elapsed3=time.time()
        print '\n----------------------------------------------------'
        print "Total time to process frame %i : %f"%(i,(elapsed3-start))
        print '----------------------------------------------------\n'

        #time.sleep(1)

        i=i+1

    try:
        t=linspace(0,float(hscale)*10,10000)
        #wavefm=wavefm/65536*float(vscale)*8
        resol=65536/10 #10 divisions for 65536 levels (range -32768 t0
            32767)
```

```
        avgsignal=avgsignal/resol*float(vscale)/FRAMES
        #print len(t)
        #print float(vscale)
        #print '\n----------------------------------------------'
        #fname=raw_input("InserFilename >>>")
        #print '----------------------------------------------\n'
        #test=raw_input('TestID >')
        #temp="ROOM"
        test = "SampleB_Dir001_Shear_23C_back1"
        fname=sample+'_'+test
        fpath=fpath+fname+'.dat'
        print fpath
        print os.getcwd()
        savetxt(fpath,(t,avgsignal))

        plot(t,avgsignal,'-')
        title("Pulse Echo : RUS Lab, NCPA : Sample " + fname)
        xlabel("Time (s)")
        ylabel("Voltage (V)")
        grid()
        show()
    except:
        print "Error 04 : Plot Error!!!"
```

VITA

Sumudu Prasanna Tennakoon earned his Bachelor of Science degree in physics from the University of Kelaniya, Kelaniya, Sri Lanka in 2008. He started his postgraduate education upon receiving a generous graduate assistantship from the University of Mississippi (UM) in August 2009. Sumudu earned M.A. in physics from the UM in 2013. He worked as a teaching assistant for one year and joined the Resonant ultrasound spectroscopy (RUS) group at the National Center for Physical Acoustics (NCPA) as a research assistant in the summer of 2010. Since then he worked on several projects of studying elastic properties of materials at extreme environments; mainly novel and energy related materials. He also worked on a project characterizing acoustic sensors to be used in structural heath monitoring systems. While working these projects he acquired the skills on scientific computing, design and implement instrumentation, data acquisition systems, RUS and pulse-echo techniques as acoustic methods of material characterization. In early 2012 Sumudu started working on his dissertation project under the supervision of Dr. Joseph Gladden to study temperature and pressure effects on elastic properties of relaxor ferroelectric PMN-PT material after reading some fascinating journal articles about this promising transduction material. Sumudu gave several research presentations in local, regional, national and international conferences. He won the first place winner in podium presentations at the UM Graduate Student Council Research Symposium held April 2015.

During his time at the UM as a graduate student, Sumudu took leadership roles in several university organizations and, also took part in outreach activities. In March 2016, Sumudu was inducted into the 2015-2016 class of Who's Who Among Students in American Universities and Colleges. He was also inducted as a member of Sigma Pi Sigma, the Physics honor society in May 2016.