DESIGN GUIDELINES FOR IMPACT MECHANICAL FREQUENCY UP-CONVERSION PIEZOELECTRIC ENERGY HARVESTERS

A THESIS SUBMITTED TO THE GRADUATE DIVISION OF THE
UNIVERSITY OF HAWAIʻI AT MĀNOA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

CIVIL AND ENVIRONMENTAL ENGINEERING

DECEMBER 2016

By

Lawrence R. Corr

Thesis Committee:

David T. Ma, Chairperson

H. Ronald Riggs

Ian Robertson

Keywords: Mechanical Frequency Up-Conversion, Impact Excitation, Piezoelectric, Energy Harvesting

Abstract

Vibration energy harvesters based on the impact mechanical frequency up-conversion technique utilize an impactor, which gains kinetic energy from low frequency ambient environmental vibrations, to excite high frequency systems tuned to efficiently convert mechanical energy to electrical energy. In order to design energy harvesters to take full advantage of the impact mechanical frequency up-conversion technique, it is prudent to understand the mechanisms of energy transfer from the low frequency excitations, to the impactor, and finally to the high frequency systems. The purpose of this work is to develop design guidelines for impact mechanical frequency up-conversion piezoelectric energy harvesters. The specific objectives are to develop guidelines for:

- Maximum energy transfer from the impactor to high frequency system
- High frequency system design to maximize energy generation from piezoelectric device
- Impactor size and placement of high frequency system to maximize impactor / high frequency system interaction for a given excitation spectrum

Table of Contents

List of Tables

List of Figures

CHAPTER 1. INTRODUCTION AND LITERATURE REVIEW

Chapter 1 provides a brief review of Structural Health Monitoring, a general overview of energy harvesters, an introduction to the impact mechanical frequency up-conversion technique, and the purpose and objectives of this work.

1.1. Structural Health Monitoring

The ultimate goal of Structural Health Monitoring (SHM) programs for civil structures (e.g., buildings, bridges, power plants, tunnels, off-shore platforms, port facilities [Brownjohn, 2007]) is to continually determine the overall "health" of a structure, in real-time, by monitoring its response to environmental excitation. A structure's "health" can be defined as its ability to perform, and the remaining timeframe for which it can perform, the design function of the structure. The ideal SHM programs have the advantage of being able to identify deterioration or damage at the earliest possible stage (Doebling, Farrar & Prime, 1998) in order to develop the appropriate decisions on whether to repair or replace the structure, to avoid cataclysmic failure, and to optimize maintenance activities (Hu et al., 2014).

Typically, SHM consists of four major activities (Doebling et al., 1998; Tamas, 2012):

- Identify damage that has occurred to the structure
- Determine the location of the damage
- Assess the severity of the damage
- Estimate the remaining service life of the structure

Damage does not necessarily mean structural failure is imminent (e.g., structure is no longer able to perform the design function), but rather that the structure is no longer at optimal "health" (Farrar & Worden, 2007). Damage can be loosely defined as any change, either intentional or unintentional, in the material properties of the structure, the geometric properties of the structure, or both, which adversely affects the current or future performance of the design function of the structure (Doebling et al., 1998; Farrar & Worden, 2007).

SHM involves the measurement of the structure's response due to environmental excitation with globally located sensors, the identification of damage indicators from the measurements, and statistical analysis on the indicators to determine the current state of the structure's "health" (Farrar

& Worden, 2007). A SHM program is actually a damage detection and characterization strategy (Sohn et al., 2004; Amditis, 2010). It will prevent catastrophic failures and maximize the structure's availability, while minimizing downtime. The strategy will replace scheduled and periodic maintenance programs with performance-based programs (Farrar & Worden, 2007). It will also minimize the human involvement in maintenance, decreasing the respective labor requirements and human error, and ultimately improving safety and resiliency. The effectiveness of the damage and characterization strategy is only as effective as its ability to identify problematic issues in a timely manner; therefore, the current research direction is to supplement, possibly eliminate, the limited and intermittent inspection programs by continuous, real-time SHM programs (Brownjohn, 2007)

1.1.1. Local and Global Structural Health Monitoring

Both local and global techniques are used to perform the first three structural health monitoring (SHM) major activities listed above (Kim et al., 2007). Local SHM based on non-destructive evaluation is highly effective (Farrar & Worden, 2007) and is a mature technology. It is typically performed at the site of the damage and does not use global structure response data. Typical examples of local SHM techniques are (Chang, Flatau & Liu, 2003; Doebling et al., 1998):

- Visual inspections
- Acoustic emissions
- Ultrasonic measurements
- Impact-echoes
- Tap tests
- X-ray imaging
- Gamma ray imaging
- Radar technology
- Magnetic field methods
- Eddy-current methods
- Thermal field methods

These techniques are typically robust and relatively inexpensive, but they are also labor intensive, possibly requiring extensive equipment set-ups, and are not always possible due to structure layout (e.g., access required to both sides of a structural member). Also, these local techniques are typically performed either on time-based (i.e., periodic) maintenance activities (Farrar & Worden, 2007), or

when the damage is already known to exist on the structure (Doebling et al., 1998). Therefore, the typical local SHM technologies will not allow continuous, real-time monitoring of the structure's "health."

Over the last 30 years, a large portion of SHM research has been focused on global SHM techniques (Farrar & Worden, 2007). The ideal global SHM program involves a holistic combination of sensor technology, data transmission, computing technology, and processing power integrated with the structure (Brownjohn, 2007). Global SHM uses sensors (e.g., accelerometers, strain sensors, load cells, displacement transducers, GPS [global positioning system], terrestrial laser scanning, vision-based systems, laser Doppler vibrometers [Park, Shin, Choi & Kim, 2013]) to sense the global structural response (e.g., acceleration, strain, load, displacement, position, tilt, climate, curvature [Amditis, 2010]), with appropriate data interrogation and model development / updating methodologies to estimate the "health" of a structure. Typical examples of global SHM techniques are (Chang et al., 2003):

- Detecting shifts in modal frequencies and mode shapes due to damage
- Detecting changes in the slope or curvature of the structure's mode shapes
- Using imaging and pattern recognition methods to identify cracks in the structure
- Modifying the mass, stiffness, and damping matrices of structure's models to match measured data
- Using the Damage Locating Vector technique for mapping changes in flexibility
- Using artificial neural networks or statistical pattern-recognition approaches with the global response data to identify damage to the structure
- Using wavelet analysis, Hilbert-Huang analysis, or similar types of analyses to observe transient changes of vibration signals to eliminate the need for baseline signals
- Modifying the stiffness and damping of a structure with smart actuators to modify the global response of the structure to provide additional data for damage detection
- Using smart materials to excite and sense the response of the structure; using the data to identify damage to the structure

One major challenge of global SHM is identifying local, low order of magnitude, damage to the structure (e.g., corrosion, connection problems, material degradation), since these local conditions

will not have a major impact on the global response of the structure (Chang et al., 2003). Local damage will have an impact on the high frequency modes of the structure (Doebling et al., 1998); however, the higher modes will normally not be used in a global SHM program to detect the "health" of the structure due to sensor bandwidth requirements and due to the high number of sensors required: sensing over large regions of the structure with fine spatial resolution (Hu et al., 2014) (e.g., thumb rule to use 4 to 6 sensors over the length of the wave of interest – as frequency increases, the number of sensors increase).

There are many additional challenges to SHM to be addressed, including nontechnical ones, such as: (1) obtaining regulatory acceptance of the safety benefits of SHM and (2) obtaining civil structure owner acceptance of the economic benefits of SHM (Farrar & Worden, 2007). Another set of major technical challenges of global SHM is focused on the sensors used to obtain global response data. The current SHM sensor research is focused in many areas ranging from wireless data transmission (wires are difficult to install after construction is complete; wires may tend to impede structure's design function; wire infrastructure is expensive [Chang et al., 2003; Kim et al., 2007; Hackmann, Guo, Yan, Lu & Dyke, 2010]), to localized / decentralized data processing (minimizing the amount of data transmitted to a central location for processing; performing analysis locally and transmitting only pertinent information [Hackmann et al., 2010]), to sensor robustness (sensors must withstand the structure's environment and operate for the design life of the structure [Chang et al., 2003]), to the vulnerability to environmental signal noise and earthquake conditions, and to the sensing location requirements for large, complex structures (Amditis, 2010). Finally, a large and important area of SHM sensor research is on the development of an efficient and limitless power source for sensor operation (i.e., energy harvesting).

1.2. Energy Harvesting for Structural Health Monitoring

Over the past few decades, energy harvesting technology has received much interest from researchers (Xiao & Wang, 2014). The field of energy harvesting is broad and entails, not only the generation of energy from the environment, but also power management, sensor network implementation, types of sensors, network strategies, system designs, and dynamic power requirements (Park, Rosing, Todd, Farrar & Hodgkiss, 2008; Davidson & Mo, 2014). Energy harvesting, or energy scavenging, is the process of converting energy available in the general environment into a usable from of energy; typically electrical energy (Park et al., 2008; Beeby et al.,

2013). Even though large-scale solar and wind turbine energy harvesting are considered to be mature technologies, energy harvesting for small, low-power devices that can be used for autonomous structural health monitoring are still in the early developmental stages (Jiang, Wang, Li, Li & Yao, 2014).

1.2.1. General Classification of Energy Harvesters

At a basic level, all energy harvesters can be categorized by three characteristics: the excitation source, the type of transducer (i.e., energy conversion device), and the dynamics of the device. A brief summary of each of these characteristics is presented below.

1.2.1.1. Excitation Source

Energy harvesters need a source of energy to scavenge. The energy source will depend on the application (e.g., location, type of structure or system) for the needed device. Currently, energy harvesters use a range of energy sources, including some of the most exploited energy sources: solar radiation, acoustic radiation, thermal radiation, ocean waves / water flow, wind, mechanical vibrations (both transverse and rotational), aeroelastic vibrations (galloping, flutter, vortex induced vibration), human motion and body heat, vehicle motion (e.g., cars, trucks, trains, planes), and ambient radio frequency (RF) energy (Park et al., 2008; Davidson & Mo, 2014; Xiao & Wang, 2014; Maurya, Yan & Priya, 2015).

1.2.1.2. Transducer Type

For each type of energy source, there is typically one or more transducers that can be used to convert the environmental energy into a usable form of energy. Some of the most popular transducer types are presented below (Davidson & Mo, 2014; Jiang et al., 2014; Maurya et al., 2015; Park et al., 2008; Xiao & Wang, 2014).

- Photovoltaic – solar radiation
- Thermoelectric generators (TEGs) – thermal radiation
- Magnetic shape-memory alloys – thermal energy, vibrational energy
- Rectenna (Rectifying Antenna) – RF energy
- Micro-wind turbine – wind energy
- Piezoelectric material – vibrational energy, e.g.,

    - PVDF material (polyvinylidene difluoride), PP material (polyvinylene polymer), PMN-PT single crystal devices (lead magnesium niobate-lead titanate), PZN-PT (lead zinc niobate-lead titanate) and PZT (lead zirconate titanate) in the form of macro-fiber composite devices, "cymbal" piezoelectric transducer, patch (benders) devices, and stack devices
- Magnetostrictive – vibrational energy, e.g.,
    - galfenol, terfenol-d
- Electromagnetic – vibrational energy, e.g.,
    - A coil / permanent magnet / spring system
- Electrostatic – vibrational energy, e.g.,
    - Movement of charged capacitor plates of a variable capacitor against the electrostatic forces between the electrodes
- Electroactive – vibrational energy, e.g.,
    - Dielectric elastomers (DEs), ionic polymer-metal composites (IPMCs)

1.2.1.3. Harvester Dynamics

The dynamics of energy harvesters may be classified as linear (i.e., superposition principle applies) or non-linear (i.e., multiple equilibrium points; superposition principle does not apply). Linear energy harvesters are typically either High-Q (high quality factor – low system damping, narrow bandwidth) or Low-Q (low quality factor – higher system damping, wide bandwidth) resonators (Beeby et al., 2013, Trigona et al., 2010). The High-Q energy harvesters are more efficient at producing energy; however, due to the narrow bandwidth, any change in the excitation frequency of the energy source will have a major negative impact on harvesting efficiency. The Low-Q energy harvesters are not as efficient at producing energy as the High-Q harvesters, but they allow for some variation in energy source excitation frequency with a smaller negative impact on harvesting efficiency. Also, another popular use of linear harvesters is to group slightly mistuned High-Q harvesters together to create the effect of a single multi-frequency energy harvester (Xiao & Wang, 2014).

Non-linear energy harvesters have quite a range of variation, but typically couple a linear system with additional dynamics to create a non-linear effect (e.g., a simple cantilever beam couples with permanent magnets), or places a linear system in a condition that creates alternative equilibrium

points (e.g., pinned-pinned beam under a buckling condition). The non-linear harvesters have been developed to overcome the issues associated with the linear harvesters (Trigona et al., 2010, Xiao & Wang, 2014). Non-linear harvesters also use “impacts” or “stops” in the system to convert low frequency excitation energy into more efficient high frequency excitation for the energy harvesters. Also, quasi-linear energy harvesters (e.g., tunable harvesters) have been developed (Davidson & Mo, 2014). The quasi-linear harvesters are based on linear energy harvesters for the majority of the operation time, but are non-linear when the system dynamics are changed to improve the efficiency of the harvester.

1.2.2. Energy Harvesting for Autonomous Sensor Networks

An ultimate goal of structural health monitoring is to be able to embed a dense population of autonomous sensor networks on civil structures for investigating both local and global integrity of the structure. The current conventional paradigm for autonomous sensor network includes a sensing device(s), signal conditioning, analog-to-digital conversion, signal processing and analysis, power management, and wireless transmission of results / data (Godinez-Azcuaga & Ley, 2014; Zhou & Wu, 2015) – all requiring some source of power. This will never be practical if the sensor networks require power cables, or if included power supplies will need to be replaced during the lifetime of the structure (Park et al., 2008;). Energy harvesting technology is increasingly considered to be the key issue in developing viable autonomous sensor networks with extended lifetimes (Trigona et al., 2010).

1.3. Frequency Up-conversion Energy Harvesters

In general, most of the transducers that convert mechanical energy to electrical energy (e.g., piezoelectric, electromagnetic, electrostatic) operate most efficiently at high (>100 Hz) frequencies (Cui & Hu, 2014; Gu & Livermore, 2011; Liu, Lee, Kobayashi, Tay & Quan, 2012; Vijayan, Friswell, Khodaparast & Adhikari, 2015). However, most of the general ambient vibration motions (environmental mechanical energy), which are widely available for energy harvester excitation (Külah & Najafi, 2008), are non-resonant, of low (<30 Hz) frequencies and amplitudes, have flat acceleration spectrums, and drift as a function of time (Galchev, McCullagh, Peterson & Najafi, 2011; Halim, Khym & Park, 2013; Halim & Park, 2015; Külah & Najafi, 2004; Külah & Najafi, 2008; Roundy, Wright & Rabaey, 2003; Zorlu, Topal & Külah, 2011; Zorlu & Külah, 2013). It is generally recognized that for an ideal energy harvester based on a resonant mechanism, the maximum power output of the device is

directly proportional to the cube of the resonant frequency of the device (Liu et al., 2012; Mitcheson, Yeatman, Rao, Holmes & Green, 2008; Vijayan et al., 2015; Zorlu et al., 2011); therefore, as the operational frequency of the transducer decreases, so does the available power output of the device (Cui & Hu, 2014; Gu & Livermore, 2011,Halim et al., 2013; Halim & Park, 2015; Külah & Najafi, 2004; Haroun & Yamada, 2015; Külah & Najafi, 2008; Roundy et al., 2005). Also, to take advantage of the available low frequency excitation, it is typically required to have harvester devices that are of high mass and low stiffness. These types of requirements make energy harvesting from low frequency excitation, while maintaining small device sizes or high power density, a design challenge (Gu & Livermore, 2011; Liu et al., 2012; Shukla & Bell, 2015; Vijayan et al., 2015).

To improve the power efficiency and broaden the frequency band of operation of harvesters in the range of the available ambient vibrations, a non-linear energy harvester type has been developed that uses a concept called mechanical frequency up-conversion. Mechanical frequency up-conversion is defined as the process of exciting a high frequency system or device at its resonance frequency from a low frequency input source (Shukla & Bell, 2015). The use of mechanical frequency up-conversion for energy harvesters seems to have been introduced in literature around the 1996 to 2004 timeframe (Cui & Hu, 2014; Gu & Livermore, 2011; Külah & Najafi, 2004; Külah & Najafi, 2008; Umeda, Nakamura & Ueha, 1996; Zorlu et al., 2011). The basic concept of mechanical frequency up-conversion is to use a system that operates at low frequencies to either excite a system that operates at high frequencies, or interact with a second system to excite high frequency content in the low frequency system. When a high frequency system is used, it can then be utilized as the energy harvesting component, thus taking advantage of improved power generating efficiencies that result from high frequency operation.

Mechanical frequency up-conversion techniques for energy harvesting can be loosely categorized into three groups, as shown in the below schematic (see Figure 1). Figure 1(a) presents a low frequency resonance system excited by base motion. If the system amplitude exceeds a specific level, the system will impact with stoppers or barriers and will excite higher frequency modes of the system. The high frequency excitation can then be used to more efficiently convert mechanical to electrical energy. Figure 1(b) again presents a low frequency system excited by base motion (e.g., a pendulum that is excited by base motion). For this category of energy harvester, if the system amplitude exceeds a specific level, the low frequency system will impact high frequency systems,

transferring energy to the high frequency systems, which then convert mechanical energy to electrical energy. In the last category of mechanical frequency up-conversion energy harvesters, Figure 1(c), a non-resonance system is used to impact the high frequency systems, which are again used to convert mechanical energy to electrical energy. The non-resonance system may be a slipping block (or mass), or possibly a rolling and slipping sphere (or ball).

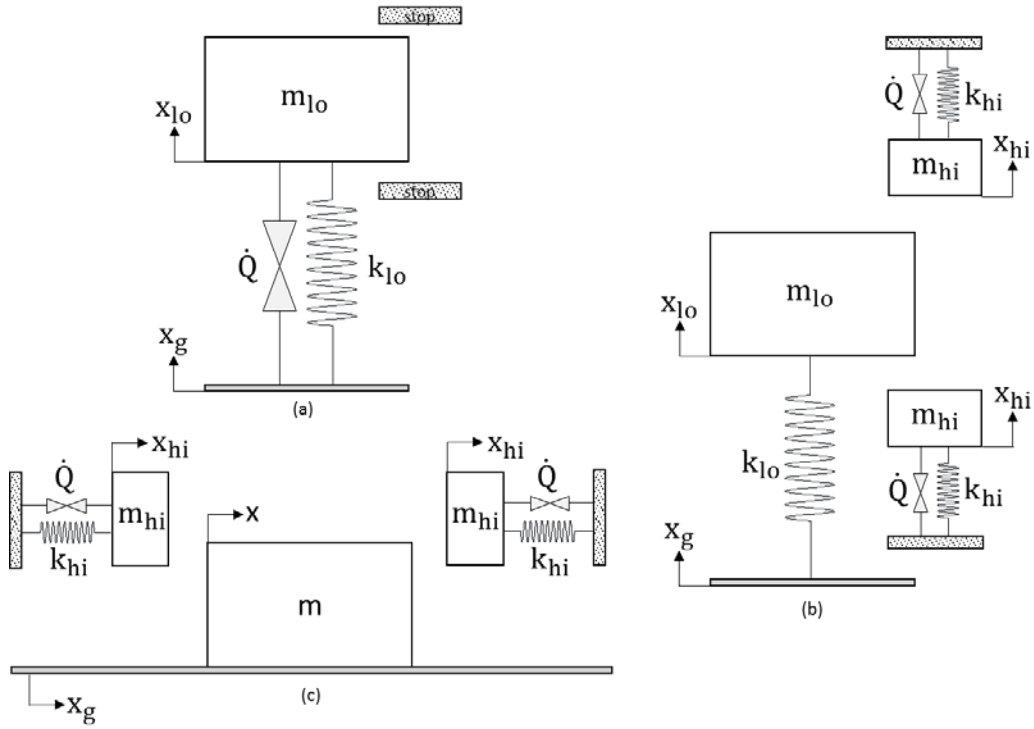


Figure 1 – General categories of frequency up-conversion energy harvesters: (a) high frequency excitation (stops), (b) low frequency system (lo) to high frequency system (hi) energy transfer, and (c) non-resonance system impact

1.4. Purpose of Thesis

The use of mechanical frequency up-conversion energy harvesters is seen as a breakthrough in generating power from low frequency ambient energy sources (Liu et al., 2012, Halim et al., 2013, Zorlu & Külah, 2013, Cui & Hu, 2014). However, the current state of literature provides little general guidance on the development of such devices. The purpose of this thesis is to develop design guidelines for impact mechanical frequency up-conversion piezoelectric energy harvesters. The specific objectives are to develop guidelines for:

- Maximum energy transfer from the impactor to high frequency system
- High frequency system design to maximize energy generation from piezoelectric device

- Impactor size and placement of high frequency system to maximize impactor / high frequency system interaction for a given excitation spectrum

1.5. Outline of Thesis

This thesis is divided into the following chapters. Chapter 2 presents the foundation for all of the system models used in this work, developing the equations of motion for each type of system. Chapter 3 describes the dynamics of the interaction of the impactor with the high frequency system. Chapter 4 presents how to maximize the energy transfer from the impactor to the high frequency system. In Chapter 5, the details of an experimental test set-up are presented, and experimental data is compared to numerical simulation data. Finally, in Chapter 6 the work of this thesis is summarized and suggestions for future work are presented.

CHAPTER 2. SYSTEM MODELLING

Chapter 2 provides the foundation for all models used in this work. The equation of motion for each type of system used to generate and substantiate the design guidelines are presented below. Note: all system dynamics are for planar motion (i.e., two dimensional motion); general motion (i.e., three dimensional motion) should be considered in future work.

2.1. Spring / Mass System

It is typically advantageous to use simple structural models when first exploring a new phenomenon. The simplicity of the models allows one to gain a fundamental understanding of the items to be investigated that can later be utilized on complex models, which more accurately predict the response of the structures of interest. Two simple models used in this work are the single degree of freedom spring / mass system and the multiple degree of freedom spring / mass system.

2.1.1. Single Degree of Freedom Equation of Motion

The response of most linear – time invariant structures can be approximated as a combination of simple single degree of freedom (SDOF) spring (k) / mass (m) systems, with an applied force (F) and possibly a damper (c). Therefore, it is prudent to review the derivation for the equation of motion for this type of system (see Figure 2).

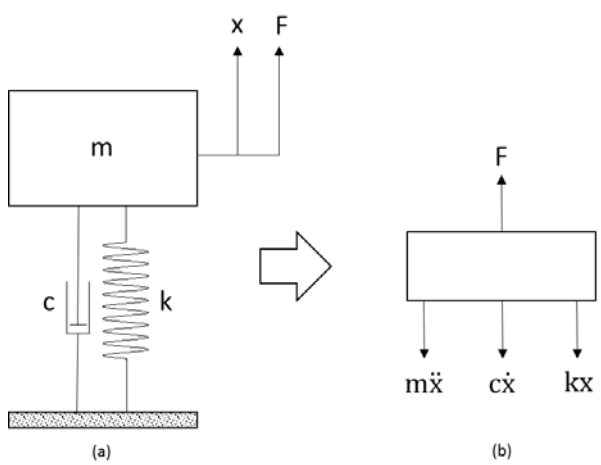


Figure 2 – Simple single degree of freedom: (a) schematic and (b) free body diagram

Summing the forces in the "x" direction and setting equal to zero provides the equation of motion for the single degree of freedom system.

$$m\ddot{x} + c\dot{x} + kx = F \qquad \text{Eq. 1}$$

2.1.2. Multiple Degree of Freedom Equation of Motion

It is usually instructive to use simplified models of structures to demonstrate and understand physical phenomena (e.g., impacts). One such model is the multiple degree of freedom (MDOF) spring / mass system, which is an extension of the single degree of freedom spring / mass system. The MDOF system provides an easy to understand representation of a structural response, yet it also provides multiple natural frequencies and mode shapes, adding some complexity to the overall response. The equation of motion for a five degree of freedom spring / mass system is presented below (see Figure 3).

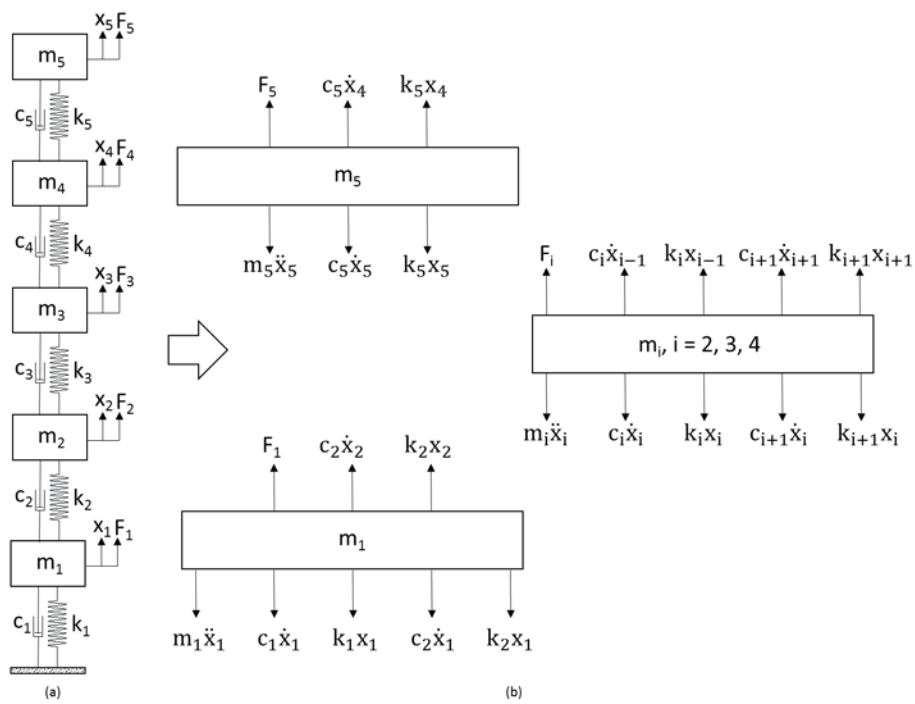


Figure 3 – Multiple degree of freedom system: (a) schematic and (b) free body diagram

Summing the forces in the "x" direction and setting equal to zero provides the equations of motion for the five degree of freedom system.

$$\begin{aligned} &m_1\ddot{x}_1 + (c_1 + c_2)\dot{x}_1 + (k_1 + k_2)x_1 - c_2\dot{x}_2 - k_2x_2 = F_1 \\ &m_i\ddot{x}_i + (c_i + c_{i+1})\dot{x}_i + (k_i + k_{i+1})x_i - c_i\dot{x}_{i-1} - k_ix_{i-1} - c_{i+1}\dot{x}_{i+1} - k_{i+1}x_{i+1} = F_i \\ &\qquad i = 2, 3, 4 \\ &m_5\ddot{x}_5 + c_5\dot{x}_5 + k_5x_5 - c_5\dot{x}_4 - k_5x_4 = F_5 \end{aligned} \qquad \text{Eq. 2}$$

Or, in matrix form:

$$\begin{bmatrix} m_1 & 0 & 0 & 0 & 0 \\ 0 & m_2 & 0 & 0 & 0 \\ 0 & 0 & m_3 & 0 & 0 \\ 0 & 0 & 0 & m_4 & 0 \\ 0 & 0 & 0 & 0 & m_5 \end{bmatrix} \begin{Bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \ddot{x}_3 \\ \ddot{x}_4 \\ \ddot{x}_5 \end{Bmatrix} + \begin{bmatrix} c_1 + c_2 & -c_2 & 0 & 0 & 0 \\ -c_2 & c_2 + c_3 & -c_3 & 0 & 0 \\ 0 & -c_3 & c_3 + c_4 & -c_4 & 0 \\ 0 & 0 & -c_4 & c_4 + c_5 & -c_5 \\ 0 & 0 & 0 & -c_5 & c_5 \end{bmatrix} \begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \end{Bmatrix} +$$

$$\begin{bmatrix} k_1 + k_2 & -k_2 & 0 & 0 & 0 \\ -k_2 & k_2 + k_3 & -k_3 & 0 & 0 \\ 0 & -k_3 & k_3 + k_4 & -k_4 & 0 \\ 0 & 0 & -k_4 & k_4 + k_5 & -k_5 \\ 0 & 0 & 0 & -k_5 & k_5 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{Bmatrix} = \begin{Bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \\ F_5 \end{Bmatrix} \qquad \text{Eq. 3}$$

$$[M]\{\ddot{x}\} + [C]\{\dot{x}\} + [K]\{x\} = \{F\}$$

where M, C, and K denote the mass, damping, and stiffness matrices, respectively. The modal response of the system can be obtained by using the following coordinate transformation from the results of the eigenvalue problem in non-standard form:

$$\begin{gathered} [K][\Phi] = [\Omega^2][M][\Phi] \\ [\Phi]^T[M][\Phi] = [I] \\ [\Phi]^T[K][\Phi] = [\Omega^2] \end{gathered} \qquad \text{Eq. 4}$$

$$\{x\} = [\Phi]\{q\}$$

where Φ denotes the modal matrix, Ω denotes a diagonal matrix of natural frequencies, I denotes the identity matrix, and q denotes a vector of the modal displacements. Therefore, the equation of motion in modal coordinates is:

$$\begin{gathered} [\Phi]^T[M][\Phi]\{\ddot{q}\} + [\Phi]^T[C][\Phi]\{\dot{q}\} + [\Phi]^T[K][\Phi]\{q\} = [\Phi]^T\{F\} \\ [I]\{\ddot{q}\} + [2\xi\Omega]\{\dot{q}\} + [\Omega^2]\{q\} = [\Phi]^T\{F\} \end{gathered} \qquad \text{Eq. 5}$$

Note that for simplicity, the damping matrix has been assumed to be mass proportional, stiffness proportional, or both, with a damping ratio of ξ. It is also useful to determine the energy of the system in terms of each mode of the system.

$$\begin{gathered} KE_i = \frac{1}{2}\{\dot{q}_i\}^T\{\dot{q}_i\} \\ PE_i = \frac{1}{2}\{q_i\}^T[\Omega^2]\{q_i\} \\ TE_i = KE_i + PE_i \end{gathered} \qquad \text{Eq. 6}$$

where KE_i, PE_i, and TE_i denote the kinetic energy, potential energy, and the total energy in the ith mode of the system. Note that the extension to fewer or more degrees of freedom is relatively straight forward.

2.2. Slipping Mass

It is quite possible that the impactor for a frequency up-conversion type energy harvester will be a block or mass that slips on a surface to contact and excite the high-frequency sub-systems. Therefore, the equations of motion of a mass slipping on a moving surface have been derived.

The surface is modeled as a simple spring / mass system (see Figure 4). This ensures that the excitation to the slipping mass is physical (i.e., the changes in velocity and acceleration can not be instantaneous), and it provides a convenient method of calculating the position, velocity, and the acceleration of the surface or "ground" and determining when the mass starts and stops slipping. It is assumed that the slipping mass is of such a small mass that it has no impact on the dynamics of the "ground."

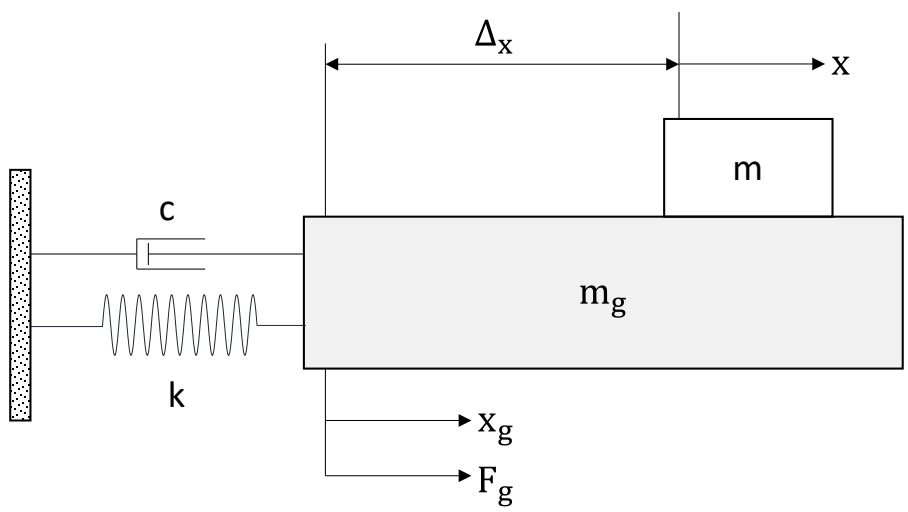


Figure 4 – Slipping mass on moving ground

It can be shown (see section 2.1) that the equation of motion for the "ground" can be determined by:

$$m_g\ddot{x}_g + c\dot{x}_g + kx_g = F_g \qquad \text{Eq. 7}$$

where m_g, c, and k denote the mass, damping, and stiffness of the "ground" system, respectively; x_g, $\dot{x}_g$, and $\ddot{x}_g$ denote the position, velocity, and the acceleration of the "ground" system, respectively; and F_g denotes the "ground" excitation force. As seen in Figure 4, for the no slip condition, the position, velocity, and acceleration of the mass (x, $\dot{x}$, and $\ddot{x}$) are related to the "ground" position, velocity, and acceleration:

$$\begin{aligned} x &= \Delta_x + x_g \\ \dot{x} &= \dot{x}_g \\ \ddot{x} &= \ddot{x}_g \end{aligned} \qquad \text{Eq. 8}$$

Figure 5 presents the forces acting on the slipping mass. The only force that can cause any acceleration of the mass is the friction force caused either by the potential or actual movement between the mass and the "ground." As long as there is no slip between the mass and "ground," the mass will have the same position (with a possible offset Δ_x), velocity, and acceleration as the "ground." However, once the mass starts to slip, the mass is governed by the kinetic friction force.

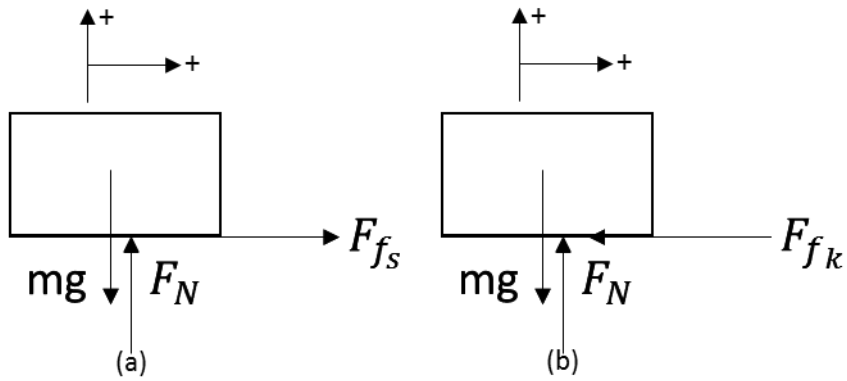


Figure 5 – Free body diagram for slipping mass: (a) no-slip condition, (b) slipping condition

The direction of the static friction force, $\mathrm{F_{f_s}}$ – see Figure 5(a), may be determined by assuming that there is no friction between the "ground" and the mass, moving the "ground" in the positive direction, and determining the direction the mass would move with respect to the "ground." The static friction force will act in the direction opposite to this motion to maintain the no slip condition. Since there is no slip between the "ground" and the mass (i.e., no motion), no work is performed on the mass. However, as the mass slips, the direction of the kinetic friction force, $\mathrm{F_{f_k}}$ – see Figure 5(b), will always be in the direction opposite to the actual slipping motion. Summing the forces in each direction and setting equal to the accelerations provides the equations of motion for the mass. For the no slip condition, the acceleration of the mass is equal to the acceleration of the "ground" (see Eq. 8 and Eq. 9).

$$\begin{gathered} \mathrm{F_N = mg} \\ \mathrm{F_{f_s} = m\ddot{x} = m\ddot{x}_g} \\ \left|\mathrm{F_{f_s}}\right| \leq \mu_s \mathrm{mg} \end{gathered} \qquad \text{Eq. 9}$$

Therefore, the static friction force varies as a function of the ground acceleration. For the slip condition, the acceleration of the mass is independent of the acceleration of the "ground."

$$\begin{gathered} \mathrm{F_N = mg} \\ \left|\mathrm{F_{f_k}}\right| = \mu_k \mathrm{F_N} = \mu_k \mathrm{mg} \\ \mu_k \mathrm{mg} = \mathrm{m}|\ddot{x}| \end{gathered} \qquad \text{Eq. 10}$$

Or:

$$|\ddot{x}| = \mu_k g \qquad \text{Eq. 11}$$

From Eq. 9, it is clear that as long as the magnitude of the "ground" acceleration is less than or equal to the static friction coefficient multiplied by the gravitational constant ($|\ddot{x}_g| \leq \mu_s g$), the mass will not slip and will have the same position (possibly with an offset), velocity, and acceleration as the "ground." However, if this acceleration magnitude is exceeded, the mass will have a constant acceleration in the opposite direction of the relative velocity of the mass and the "ground," or:

$$\ddot{x} = -1 \times \mathrm{sgn}(\dot{x} - \dot{x}_g)\mu_k g \qquad \text{Eq. 12}$$

where $\mathrm{sgn}(\blacksquare)$ denotes the signum function of $\blacksquare$, and $\dot{x} - \dot{x}_g$ is the relative velocity between the mass and the "ground." Since during the slipping condition the mass acceleration is constant, and noting that:

$$\begin{aligned} \ddot{x} &= \frac{d\dot{x}}{dt} \\ \dot{x} &= \frac{dx}{dt} \end{aligned} \qquad \text{Eq. 13}$$

integration can be used to obtain the equations of motion for the mass.

$$\begin{aligned} \ddot{x} &= -1 \times \mathrm{sgn}(\dot{x} - \dot{x}_g)\mu_k g \\ \dot{x} &= \dot{x}_o - \mathrm{sgn}(\dot{x} - \dot{x}_g)\mu_k g \times (t - t_o) \\ x &= x_o + \dot{x}_o \times (t - t_o) - \mathrm{sgn}(\dot{x} - \dot{x}_g)\mu_k g \times (t - t_o)^2 \end{aligned} \qquad \text{Eq. 14}$$

where t_o denotes the time at the onset of slipping, and x_o and $\dot{x}_o$ denote the mass position and velocity at the onset of slipping (i.e., at t_o). The mass will continue to slip, under constant acceleration, until there is no relative motion between the mass and the "ground," or:

$$\dot{x} - \dot{x}_g = 0 \qquad \text{Eq. 15}$$

To obtain the time response of this system due to a known input, Eq. 7 is first used to solve for the motion of the "ground" system. Once (if) the acceleration of the "ground" reaches the slipping condition ($\ddot{x}_g > \mu_s g$), the simulation of the "ground" system is stopped, and Eq. 8 is used to calculate the mass dynamics for the time until the onset of slipping. Then, using the values of the last time step as the set of initial conditions, Eq. 7 and Eq. 14 are simultaneously solved to obtain the time response of both the "ground" and mass systems until the non-slip condition is satisfied ($\dot{x} - \dot{x}_g = 0$). This simulation cycle, switching between the non-slip and slipping dynamics, continues until the desired end time of the simulation.

2.3. Slipping and Rolling Sphere

A rolling and slipping metal sphere, or ball, that can contact and excite the high-frequency sub-systems is also a good prospect as an impactor for a mechanical frequency up-conversion type energy harvester. Therefore, the equations of motion of a slipping and rolling ball on a moving surface have been derived.

The surface is modeled as a simple spring / mass system (see Figure 6). This ensures that the excitation to the slipping mass is physical (i.e., the changes in velocity and acceleration can not be instantaneous), and it provides a convenient method of calculating the position, velocity, and the acceleration of the surface or "ground" and determining when the ball starts and stops slipping. It is assumed that the rolling and slipping ball is of such a small mass that it has no impact on the dynamics of the "ground."

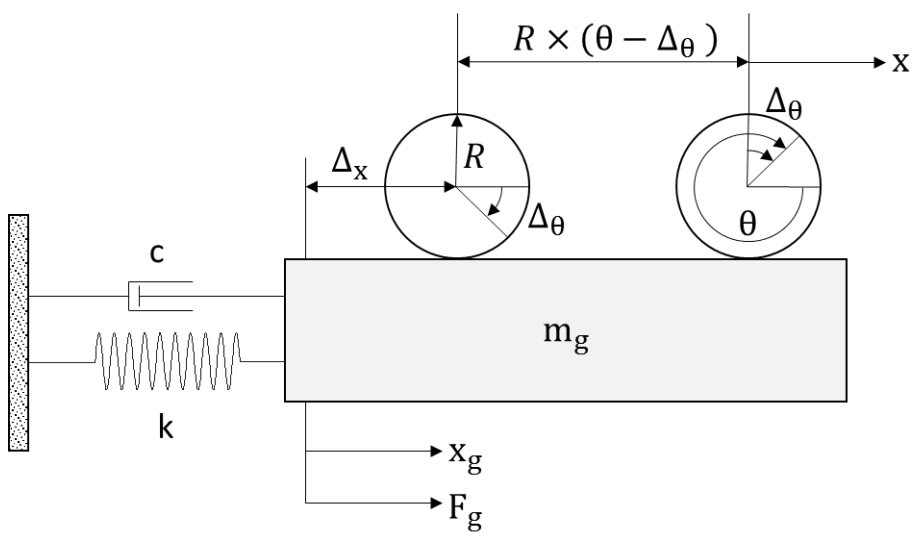


Figure 6 – Slipping and rolling ball on moving ground

It can be shown that the equation of motion for the "ground" can be determined by Eq. 7 (see section 2.1). As seen in Figure 6, for the no slip condition, the translational position, velocity, and acceleration of the ball (x, $\dot{x}$, and $\ddot{x}$) are related to the "ground" position, velocity, and acceleration:

$$\begin{aligned} x &= \Delta_x + x_g + R(\theta - \Delta_\theta) \\ \dot{x} &= \dot{x}_g + R\dot{\theta} \\ \ddot{x} &= \ddot{x}_g + R\ddot{\theta} \end{aligned} \qquad \text{Eq. 16}$$

where θ, $\dot{\theta}$, and $\ddot{\theta}$ denote the rotational position, velocity, and acceleration of the ball, respectively, Δ_θ denotes a possible rotational offset, and R denotes the radius of the ball. Figure 7 presents the

forces acting on the rolling and slipping ball. The only force that can cause any acceleration of the ball is the friction force caused either by the potential or actual movement between the ball and the "ground." As long as there is no slip between the ball and the "ground," the ball will have a position, velocity, and acceleration that is dependent on the "ground" (see Eq. 16). However, once the ball starts to slip, the ball is governed by the kinetic friction force (see Figure 7(b) and Figure 7(c)).

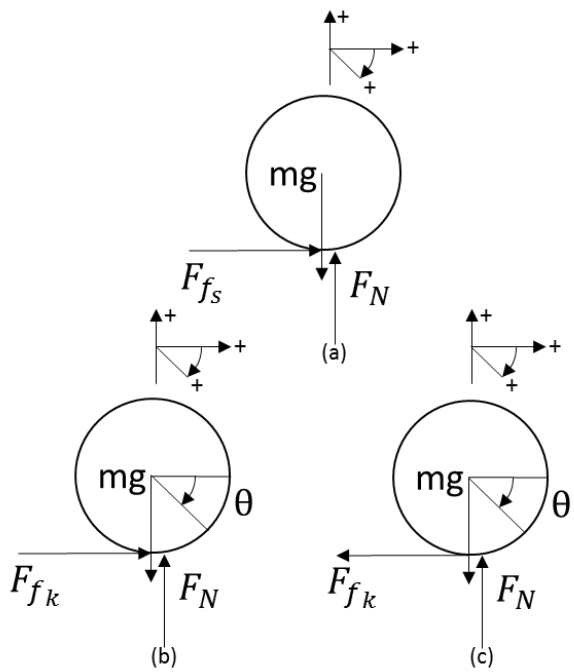


Figure 7 – Free body diagram for slipping and rolling ball: (a) no-slip condition, (b) backward slipping condition, (c) forward slipping condition

The direction of the static friction force, F_{f_s} – see Figure 7(a), may be determined by assuming that there is no friction between the "ground" and the ball, moving the "ground" in the positive direction, and determining the direction the ball would move with respect to the "ground." The static friction force will act in the direction opposite to this motion to maintain the no slip condition. Since there is no slip between the "ground" and the point of the ball that contacts the ground, no work is performed on the mass. However, as the ball slips, the direction of the kinetic friction force, F_{f_k}, will depend on the type of slipping motion that occurs. For forward slipping (see Figure 7(c)), the relative velocity of the ball's center of mass is greater than the velocity of the contact point with respect to the ball's center of mass (i.e., the ball is not rotating fast enough to match the relative velocity of the ball / "ground"; therefore, the kinetic friction force acts to decelerate the ball's center of mass, while accelerating the ball's rotation). For backward slipping (see Figure 7(b)), the relative velocity of the

ball's center of mass is less than the rotational velocity at the contact point with respect to the ball's center of mass (i.e., the ball is rotating too fast to match the relative velocity of the ball / "ground"; therefore, the kinetic friction force acts to accelerate the ball's center of mass, while decelerating the ball's rotation).

Summing the forces (moments) in each direction and setting equal to the translational (rotational) accelerations provides the equations of motion for the ball.

$$\begin{gathered} F_N = mg \\ F_{f_s} = m\ddot{x} \\ \left|F_{f_s}\right| \leq \mu_s mg \\ -F_{f_s} R = I_o \ddot{\theta} \end{gathered} \qquad \text{Eq. 17}$$

where I_o is the mass moment of inertia of the ball (i.e., $2/5mR^2$). For the no slip condition, the rotational acceleration of the ball is related to the acceleration of the "ground" and the ball's center of mass (see Eq. 16).

$$\ddot{\theta} = \frac{\ddot{x} - \ddot{x}_g}{R} \qquad \text{Eq. 18}$$

By substituting Eq. 18 into Eq. 17, $\ddot{x}$, $\ddot{\theta}$, and F_{f_s} can be determined.

$$\begin{gathered} \ddot{x} = \frac{I_o \ddot{x}_g}{mR^2 + I_o} \\ \ddot{\theta} = \frac{-mR\ddot{x}_g}{mR^2 + I_o} \\ F_{f_s} = \frac{mI_o \ddot{x}_g}{mR^2 + I_o} \end{gathered} \qquad \text{Eq. 19}$$

Due to the following relationships:

$$\begin{gathered} \ddot{x} = \frac{d\dot{x}}{dt} \\ \dot{x} = \frac{dx}{dt} \\ \ddot{\theta} = \frac{d\dot{\theta}}{dt} \\ \dot{\theta} = \frac{d\theta}{dt} \end{gathered} \qquad \text{Eq. 20}$$

integration can be used to determine the ball's position and velocity as a function of time.

$$
\begin{aligned}
\ddot{x} &= \frac{I_o \ddot{x}_g}{mR^2 + I_o} \\
\dot{x} &= \dot{x}_o + \int_{t_o}^{t} \ddot{x}(\tau) d\tau \\
x &= x_o + \dot{x}_o \times (t - t_o) + \int_{t_o}^{t} \int_{t_o'}^{t'} \ddot{x}(\tau) d\tau \, dt' \\
\ddot{\theta} &= \frac{-mR\ddot{x}_g}{mR^2 + I_o} \\
\dot{\theta} &= \dot{\theta}_o + \int_{t_o}^{t} \ddot{\theta}(\tau) d\tau \\
\theta &= \theta_o + \dot{\theta}_o \times (t - t_o) + \int_{t_o}^{t} \int_{t_o'}^{t'} \ddot{\theta}(\tau) d\tau \, dt'
\end{aligned}
\qquad \text{Eq. 21}
$$

where t_o denotes the time at the onset of non-slip condition, x_o and $\dot{x}_o$ denote the ball position and velocity at the onset of non-slip condition, and θ_o and $\dot{\theta}_o$ denote the ball rotational position and velocity at the onset of non-slip condition (i.e., at t_o). Note that once θ_o and $\dot{\theta}_o$ have been calculated, either Eq. 21 or Eq. 16 can be used to calculate x_o and $\dot{x}_o$. For the slip condition, the translational and rotational acceleration of the ball are independent from the acceleration of the "ground" and from each other.

$$
\begin{gathered}
F_N = mg \\
\left|F_{f_k}\right| = \mu_k F_N = \mu_k mg \\
\mu_k mg = m|\ddot{x}| \\
|-\mu_k mgR| = I_o \left|\ddot{\theta}\right|
\end{gathered}
\qquad \text{Eq. 22}
$$

Or:

$$
\begin{gathered}
|\ddot{x}| = \mu_k g \\
\left|\ddot{\theta}\right| = \left|-\frac{m\mu_k gR}{I_o}\right|
\end{gathered}
\qquad \text{Eq. 23}
$$

From Eq. 17, it is clear that as long as the magnitude of the ball's acceleration is less than or equal to the static friction coefficient multiplied by the gravitational constant ($|\ddot{x}| \leq \mu_s g$), the ball will not slip and will have a position (possibly with an offset), velocity, and acceleration that is related to the "ground." However, if this acceleration magnitude is exceeded, the ball will have a constant translational acceleration in the opposite direction of the relative velocity of the ball's center of mass and the ball's contact point on the "ground" with respect to the ball's center of mass (i.e., the relative velocity between the "ground" and the ball's contact point). Likewise, the rotational acceleration will be constant and in the same direction as the relative velocity between the ball's contact point and the "ground," or:

$$
\begin{gathered}
\ddot{x} = -1 \times \mathrm{sgn}(\dot{x} - \dot{x}_g - R\dot{\theta})\mu_k g \\
\ddot{\theta} = +1 \times \mathrm{sgn}(\dot{x} - \dot{x}_g - R\dot{\theta}) \frac{m\mu_k gR}{I_o}
\end{gathered}
\qquad \text{Eq. 24}
$$

where $\mathrm{sgn}(\blacksquare)$ denotes the signum function of $\blacksquare$, and $\dot{x} - \dot{x}_g$ is the relative velocity of the ball's center of mass between the ball and the "ground," and $R\dot{\theta}$ is the velocity of the ball's contact point on the "ground" with respect to the ball's center of mass. Since during the slipping condition the ball's acceleration is constant, and noting the relationships in Eq. 16, integration can be used to obtain the equations of motion for the ball as a function of time.

$$\begin{aligned}
\ddot{x} &= -1 \times \mathrm{sgn}(\dot{x} - \dot{x}_g - R\dot{\theta})\mu_k g \\
\dot{x} &= \dot{x}_o - 1 \times \mathrm{sgn}(\dot{x} - \dot{x}_g - R\dot{\theta})\mu_k g \times (t - t_o) \\
x &= x_o + \dot{x}_o \times (t - t_o) - 1 \times \mathrm{sgn}(\dot{x} - \dot{x}_g - R\dot{\theta})\mu_k g \times (t - t_o)^2 \\
\ddot{\theta} &= +1 \times \mathrm{sgn}(\dot{x} - \dot{x}_g - R\dot{\theta})\frac{m\mu_k g R}{I_o} \\
\dot{\theta} &= \dot{\theta}_o + 1 \times \mathrm{sgn}(\dot{x} - \dot{x}_g - R\dot{\theta})\frac{m\mu_k g R}{I_o} \times (t - t_o) \\
\theta &= \theta_o + \dot{\theta}_o \times (t - t_o) + 1 \times \mathrm{sgn}(\dot{x} - \dot{x}_g - R\dot{\theta})\frac{m\mu_k g R}{I_o} \times (t - t_o)^2
\end{aligned} \qquad \text{Eq. 25}$$

The mass will continue to slip, under constant acceleration, until there is no relative motion between the ball's contact point and the "ground," or:

$$\dot{x} - \dot{x}_g - R\dot{\theta} = 0 \qquad \text{Eq. 26}$$

To obtain the time response of this system due to a known input, Eq. 7 is first used to solve for the motion of the "ground" system. Once (if) the acceleration of the "ground" reaches a level such that the slipping condition is satisfied ($|\ddot{x}| > \mu_s g$, or similarly, from Eq. 19, $|\ddot{x}_g| > \mu_s g(mR^2 + I_o)/I_o$), the simulation of the "ground" system is stopped, and Eq. 21, or a combination of Eq. 21 and Eq. 16, is used to calculate the ball dynamics for the time until the onset of slipping. Then, using the values of the last time step as the set of initial conditions, Eq. 7 and Eq. 25 are simultaneously solved to obtain the time response of both the "ground" and ball systems until the non-slip condition is satisfied ($\dot{x} - \dot{x}_g - R\dot{\theta} = 0$). This simulation cycle, switching between the non-slip and slipping dynamics, continues until the desired end time of the simulation.

2.4. Cantilever Beams and Piezoelectric Devices

Many of the energy harvesting systems utilize a cantilever beam with attached piezoelectric devices as a source of energy. Therefore, these types of devices will also be used in this work. The equation of motion for a simple one dimensional cantilever beam system with attached piezoelectric devices will be presented. This type of simplistic model is quick to develop and easy to use, yet it provides an additional layer of complexity to the simple multiple degree of freedom spring / mass system. Then, a method of converting the results from a finite element analysis program into a state space model

will be presented. A finite element analysis model should be utilized when additional fidelity is required, either for the structural system or the piezoelectric device. The finite element analysis model will also allow the exploration of the structure's response in more than one dimension. First, however, the general derivation for a generic structure with both piezoelectric devices used as sensors and actuators will be presented.

2.4.1. Structures with Piezoelectric Devices

An excellent derivation of the equations of motion for structures that are coupled with piezoelectric actuators and sensors is given by Hagood (Hagood, Chung & von Flotow, 1990). Therefore, only a brief review of this method is given for completeness. Interested readers are directed to the previous work for the complete derivation.

The mass, stiffness, and electromechanical coupling matrices can be derived from the general form of Hamilton's principle (Crandall, Karnopp, Kurtz & Pridmore-Brown, 1968; Petyt, 1990):

$$\int_{t_1}^{t_2}(\delta(T - U + W_e) + \delta W_{nc})dt = 0 \qquad \text{Eq. 27}$$

where the work performed by the inductance of the piezoelectric materials is negligible; T and U denote the kinetic energy and the potential energy of the system, respectively (only elastic strain energy is of interest); and W_e and W_{nc} are the work done by the capacitance of the piezoelectric materials and the non-conservative work done by external forces to the system, respectively. The kinetic energy, the potential energy, the electrical work, and the non-conservative work for the structure, actuators, and the sensors are defined by (Hagood et al., 1990; Petyt, 1990; Weaver, Timoshenko & Young, 1990; Clark, Saunders & Gibbs, 1998):

$$\begin{aligned}
T &= \int_{v_b}\left(\frac{1}{2}\rho_b\{\dot{u}\}^T\{\dot{u}\}\right)dv + \int_{v_s}\left(\frac{1}{2}\rho_s\{\dot{u}\}^T\{\dot{u}\}\right)dv + \int_{v_a}\left(\frac{1}{2}\rho_a\{\dot{u}\}^T\{\dot{u}\}\right)dv \\
U &= \int_{v_b}\left(\frac{1}{2}[S_b]^T[T_b]\right)dv + \int_{v_s}\left(\frac{1}{2}[S_s]^T[T_s]\right)dv + \int_{v_a}\left(\frac{1}{2}[S_a]^T[T_a]\right)dv \\
W_e &= \int_{v_s}\left(\frac{1}{2}[E_s]^T[D_s]\right)dv + \int_{v_a}\left(\frac{1}{2}[E_a]^T[D_a]\right)dv \\
\delta W_{nc} &= \sum_i f_i\delta u_i + \sum_m q_m\delta\varphi_m
\end{aligned} \qquad \text{Eq. 28}$$

where $(\bullet)_b$, $(\bullet)_s$, $(\bullet)_a$, and $(\bullet)^T$ denote the structure, sensor, and actuator properties, and the transpose of $(\bullet)$, respectively; v, ρ, E, and D denote the volume, density, electric field applied to piezoelectric material (V/m), and the dielectric displacement (C/m^2), respectively; $T_\bullet$ and $S_\bullet$ are the engineering stress (N/m^2) and strains (m/m), respectively; q_m and φ_m are the charge and the electric potential

applied to the piezoelectric material, respectively; and u_i and f_i are generalized displacements of the system and the generalized external forces applied to the system, respectively.

The constitutive laws that describe the behavior of the piezoelectric materials must take into account both the electrical and mechanical effects. These electromechanical equations come in many forms and are problem dependent. For this derivation, the equations in matrix form for the piezoelectric actuators and sensors are (Ikeda, 1990; Waanders, 1991; Dosch, Inman & Garcia, 1992):

$$\begin{bmatrix}\{D_a\}\\ \{T_a\}\end{bmatrix} = \begin{bmatrix}[R_E]^T[\varepsilon_a^S][R_E] & [R_E]^T[e_a][R_a]\\ -[R_a]^T[e_a]^T[R_E] & [R_a]^T[c_a^E][R_a]\end{bmatrix}\begin{bmatrix}\{E_a\}\\ \{S_a\}\end{bmatrix}$$
$$\begin{bmatrix}\{D_s\}\\ \{T_s\}\end{bmatrix} = \begin{bmatrix}[R_E]^T[\varepsilon_s^S][R_E] & [R_E]^T[e_s][R_s]\\ -[R_s]^T[e_s]^T[R_E] & [R_s]^T[c_s^E][R_s]\end{bmatrix}\begin{bmatrix}\{E_s\}\\ \{S_s\}\end{bmatrix} \qquad \text{Eq. 29}$$

where ε^S is the dielectric permittivity (F/m) measured under constant strain; e is the piezoelectric e-constant (C/m^2); c^E is the elastic modulus (N/m^2) measured under constant electric field; R_s and R_a are the sensors' and actuators' engineering strain rotation matrices, respectively; R_E is the electrical field rotation matrix; and $R_\bullet^T$ denotes the transpose of $R_\bullet$. The constitutive laws that describe the behavior of the structure are simply:

$$\{T_b\} = [c_b]\{S_b\} \qquad \text{Eq. 30}$$

where c_b is the elastic modulus of the structure material. Note that it is assumed that the structure is already in global coordinates. Substituting Eq. 28 through Eq. 30 into Eq. 27 and performing the variation, the equations of motion for a structure coupled with piezoelectric actuators and sensors are (Hagood et al., 1990):

$$[M_b + M_s + M_a]\{\ddot{x}\} + [C_b + C_s + C_a]\{\dot{x}\} + [K_b + K_s + K_a]\{x\} - [\Theta_a]\{V_{app}\} = [F]\{f\}$$
$$[\Theta_s]^T\{x\} + \left[(C_p^S)_s\right]\{V_s\} = \{0\} \qquad \text{Eq. 31}$$
$$[\Theta_a]^T\{x\} + \left[(C_p^S)_a\right]\{V_a\} = \{Q^{app}\}$$

where x is a vector of generalized coordinates, f is a vector of generalized forces, Q^{app} is a vector of applied charges to the piezoelectric actuators, V_{app} is a vector of applied voltages to the piezoelectric actuators, V_s and V_a are the voltages across the piezoelectric sensors and actuators, and F is the force input influence matrix. M_b, M_s, and M_a are the mass matrices for the structure, sensors, and actuators, respectively, and are defined as:

$$[M_b] = \int_{v_b} \rho_b [\Psi_w]^T [\Psi_w] dv$$
$$[M_s] = \int_{v_s} \rho_s [\Psi_w]^T [\Psi_w] dv \qquad \text{Eq. 32}$$
$$[M_a] = \int_{v_a} \rho_a [\Psi_w]^T [\Psi_w] dv$$

where Ψ_w is either typical shape functions for finite element derivations or assumed mode shapes for Rayleigh-Ritz or Galerkin methods. K_b, K_s, and K_a are the structure, sensors, and actuators stiffness matrices, respectively, and are defined as:

$$[K_b] = \int_{v_b} \left[[L_w][\Psi_w]\right]^T [c_b][L_w][\Psi_w] dv$$
$$[K_s] = \int_{v_s} \left[[L_w][\Psi_w]\right]^T [R_s]^T [c_s^E][R_s][L_w][\Psi_w] dv \qquad \text{Eq. 33}$$
$$[K_a] = \int_{v_a} \left[[L_w][\Psi_w]\right]^T [R_a]^T [c_a^E][R_a][L_w][\Psi_w] dv$$

where L_w is the elastic differential operator, dependent on the type of structure. Θ_a and Θ_s are the electromechanical coupling matrices for the actuators and the sensors, respectively, and are defined as:

$$[\Theta_s] = \int_{v_s} \left[[L_w][\Psi_w]\right]^T [R_s]^T [e_s]^T [R_E] \left[L_\varphi\right] \left[\Psi_\varphi\right] dv$$
$$[\Theta_a] = \int_{v_a} \left[[L_w][\Psi_w]\right]^T [R_a]^T [e_a]^T [R_E] \left[L_\varphi\right] \left[\Psi_\varphi\right] dv \qquad \text{Eq. 34}$$

where L_φ is the electrical differential operator (the gradient); and Ψ_φ is the electrical field shape function. Finally, $(C_p^S)_s$ and $(C_p^S)_a$ are the piezoelectric capacitance matrices for the sensors and actuators, respectively, and are defined as:

$$\left[(C_p^S)_s\right] = \int_{v_s} \left[\left[L_\varphi\right]\left[\Psi_\varphi\right]\right]^T [R_E]^T \left[\varepsilon_s^S\right] [R_E] \left[L_\varphi\right] \left[\Psi_\varphi\right] dv$$
$$\left[(C_p^S)_a\right] = \int_{v_a} \left[\left[L_\varphi\right]\left[\Psi_\varphi\right]\right]^T [R_E]^T \left[\varepsilon_a^S\right] [R_E] \left[L_\varphi\right] \left[\Psi_\varphi\right] dv \qquad \text{Eq. 35}$$

A constant electric field is assumed over the thickness of the piezoelectric actuators; therefore, the electrical field shape function (Ψ_φ) will be a linear function.

2.4.2. One Dimensional Beam System with Piezoelectric Devices

A finite element model of a simple one dimensional beam system with a bonded piezoelectric actuator (i.e., no sensors) was developed for this work (see Figure 8(a)). The beam was modeled as a uniform, homogeneous flexible beam that was clamped at one end. It is assumed that the beam was an Euler-Bernoulli beam that only undergoes pure bending. It is also assumed that the piezoelectric device is perfectly bonded to the beam and that any dynamics associated with the bonding material

were negligible. Using the information above and the below definitions, the equation of motion for the cantilever beam with piezoelectric device was derived.

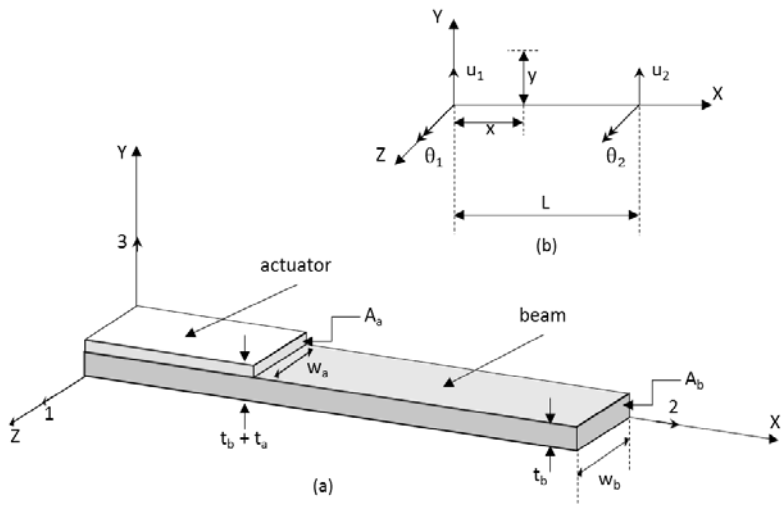


Figure 8 – Schematic for piezoelectric beam: (a) beam in global coordinates (X, Y, Z) with actuator in local coordinates (1, 2, 3) and (b) details for finite elements

Since the one dimensional beam elements are being used in this model, the shape function used in Eq. 32 through Eq. 34 was assumed to be a cubic displacement function. Therefore, each node had two degrees of freedom associated with it: transverse displacement and the slope or rotation. The finite element shape function and elastic operator were defined as (Petyt -1990; Weaver et al., 1990):

$$[\Psi_w] = \left[1 - \frac{3x^2}{L^2} + \frac{2x^3}{L^3} \quad x - \frac{2x^2}{L} + \frac{x^3}{L^2} \quad \frac{3x^2}{L^2} - \frac{2x^3}{L^2} \quad -\frac{x^2}{L} + \frac{x^3}{L^2}\right]$$
$$[L_w] = \left[-y\frac{d^2}{dx^2} \quad 0 \quad 0 \quad 0 \quad 0 \quad 0\right]^T \qquad \text{Eq. 36}$$

A constant electric field is assumed over the thickness of the piezoelectric device; therefore, the electrical field shape function and the electrical differential operator used in Eq. 34 and Eq. 35 were defined by (Hagood et al., 1990):

$$[\Psi_\varphi] = \left[0 \quad \frac{y - t_b}{t_a} \quad 0\right]^T$$
$$[L_\varphi \Psi_\varphi] = \left[0 \quad \frac{1}{t_a} \quad 0\right]^T \qquad \text{Eq. 37}$$

Finally, to rotate the piezoelectric device from the local coordinates into global coordinates, the following engineering strain and electric field rotation matrices were used.

$$[R_E] = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$[R_a] = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \qquad \text{Eq. 38}$$

Using Eq. 32 through Eq. 38, the following mass, stiffness, electromechanical coupling, and capacitance matrices were developed for each element in the finite element model.

$$[M_b] = A_b \rho_b \begin{bmatrix} \frac{13L}{35} & \frac{11L^2}{210} & \frac{9L}{70} & \frac{-13L^2}{420} \\ \frac{11L^2}{210} & \frac{L^3}{105} & \frac{13L^2}{420} & \frac{-L^3}{140} \\ \frac{9L}{70} & \frac{13L^2}{420} & \frac{13L^2}{35} & \frac{-11L^2}{210} \\ \frac{-13L^2}{420} & \frac{-L^3}{140} & \frac{-11L^2}{210} & \frac{L^3}{105} \end{bmatrix}$$

$$[M_a] = A_a \rho_a \begin{bmatrix} \frac{13L}{35} & \frac{11L^2}{210} & \frac{9L}{70} & \frac{-13L^2}{420} \\ \frac{11L^2}{210} & \frac{L^3}{105} & \frac{13L^2}{420} & \frac{-L^3}{140} \\ \frac{9L}{70} & \frac{13L^2}{420} & \frac{13L^2}{35} & \frac{-11L^2}{210} \\ \frac{-13L^2}{420} & \frac{-L^3}{140} & \frac{-11L^2}{210} & \frac{L^3}{105} \end{bmatrix} \qquad \text{Eq. 39}$$

$$[K_b] = (c_{11})_b I_b \begin{bmatrix} \frac{12}{L^3} & \frac{6}{L^2} & \frac{-12}{L^3} & \frac{6}{L^2} \\ \frac{6}{L^2} & \frac{4}{L} & \frac{-6}{L^2} & \frac{2}{L} \\ \frac{-12}{L^3} & \frac{-6}{L^2} & \frac{12}{L^3} & \frac{-6}{L^2} \\ \frac{6}{L^2} & \frac{2}{L} & \frac{-6}{L^2} & \frac{4}{L} \end{bmatrix}$$

$$[K_a] = (c_{11}^S)_a I_a \begin{bmatrix} \frac{12}{L^3} & \frac{6}{L^2} & \frac{-12}{L^3} & \frac{6}{L^2} \\ \frac{6}{L^2} & \frac{4}{L} & \frac{-6}{L^2} & \frac{2}{L} \\ \frac{-12}{L^3} & \frac{-6}{L^2} & \frac{12}{L^3} & \frac{-6}{L^2} \\ \frac{6}{L^2} & \frac{2}{L} & \frac{-6}{L^2} & \frac{4}{L} \end{bmatrix} \qquad \text{Eq. 40}$$

$$[\Theta] = e_{31} w_a \frac{t_b + t_a}{2} \begin{bmatrix} 0 \\ 1 \\ 0 \\ -1 \end{bmatrix}$$

$$(C_p^S)_a = \frac{(\varepsilon_3^S)_a L w_a}{t_a} \qquad \text{Eq. 41}$$

where A, w, t, and L denote the cross-sectional area, width, thickness, and length, respectively, of either the beam or the piezoelectric device. I_b and I_a denote the second moment of area about the neutral axis ($\bar{x}$) for the beam and piezoelectric device, respectively, and are defined as:

$$
\begin{aligned}
I_b &= w_b \frac{t_b^3}{12} \text{ (without piezoelectric}, \bar{x} = \frac{t_b}{2}) \\
I_b &= w_b \frac{t_b^3}{12} + w_b t_b (\frac{t_b}{2} - \bar{x})^2 \text{ (with piezoelectric)} \\
I_a &= w_a \frac{t_a^3}{12} + w_a t_a (\frac{t_a}{2} + t_b - \bar{x})^2 \\
\bar{x} &= \frac{\frac{(c_{11}^S)_a}{(c_{11})_b} t_a \left(\frac{t_a}{2} + t_b\right) + \frac{t_b^2}{2}}{\frac{(c_{11}^S)_a}{(c_{11})_b} t_a + t_b}
\end{aligned}
\qquad \text{Eq. 42}
$$

The mass and the stiffness matrices of the piezoelectric actuator are only added to the beam's stiffness and mass matrices if they actually cover that specific finite element. Also, since there is only one lead (contact) connected to the actuator, $[\Theta]$ will only be non-zero for those finite elements that contain the two edges (perpendicular to the "x" direction) of the actuator. It was also assumed that the damping matrix was either mass proportional, stiffness proportional, or both. This type of approximation has been used by other researchers and is well suited for the finite element work performed in this thesis (Dosch et al., 1992; Fanson & Caughey, 1987; Gopinathan, Pajunen, Neelakanta & Arockaisamy, 1995).

2.4.3. Finite Element Analysis Beam System with Piezoelectric Device

In practical applications, it is sometimes useful to use a finite element program (e.g., ANSYS, ABAQUS, SAP2000) to develop the system (structure, actuators, and sensors). The results of the finite element analysis can then be used to develop a linear time invariant state space model. The following process was used to convert finite element models, which were developed in commercial software, into state space models.

- Perform a modal analysis of the system over the frequency range of interest, ensuring that the modes of the system are mass normalized (see Eq. 4). Damping should not be included in the modal analysis.
- Obtain the natural frequencies (ω_n) and the modal displacement at each of the input degrees of freedom and the output degrees of freedom (θ_{ij}). N_{modes}, N_{inputs}, and $N_{outputs}$ denote the total number of modes, system inputs, and system outputs, respectively.
    - $\omega_n^j, j = 1, 2, \cdots, N_{modes}$

- $\theta_{ij}^{input}, i = 1, 2, \cdots, N_{inputs};\ j = 1, 2, \cdots, N_{modes}$
- $\phi_{ij}^{output}, i = 1, 2, \cdots, N_{outputs};\ j = 1, 2, \cdots, N_{modes}$

- Perform a static analysis of the system separately for each of the system inputs. Apply a unit input at each input degree of freedom (e.g., unit force, unit moment, unit charge), and obtain the static general displacement from each of the outputs.
  - $x_{ij}^{static}, i = 1, 2, \cdots, N_{outputs};\ j = 1, 2, \cdots, N_{inputs}$
- Perform a harmonic analysis over the frequency range of interest for each of the system inputs. Obtain the complex frequency results for each of the system's outputs.
  - $H(\omega)_{ij}^{harmonic}, i = 1, 2, \cdots, N_{outputs};\ j = 1, 2, \cdots, N_{inputs}$
- Using the results from the finite element analysis, form the state space matrices.
  - System matrix (A) – Note that an appropriate amount of modal damping (ξ_i) may be added to the state space system. 0_{mxn} and I_{mxn} denote mxn dimensioned zero and identity matrices, respectively.
    - $[A] = \begin{bmatrix} 0_{N_{modes} \times N_{modes}} & I_{N_{modes} \times N_{modes}} \\ -\mathrm{diag}(\omega_n^2) & -\mathrm{diag}(2\xi\omega_n) \end{bmatrix}$
  - Input matrix (B)
    - $[B] = \begin{bmatrix} 0_{N_{modes} \times N_{inputs}} \\ \Theta_{N_{modes} \times N_{inputs}} \end{bmatrix}$
  - Output matrix (C)
    - $[C] = \begin{bmatrix} \Phi_{N_{outputs} \times N_{modes}} & 0_{N_{outputs} \times N_{modes}} \end{bmatrix}$
  - Feedthrough matrix (D)
    - $[D] = \left[X_{N_{outputs} \times N_{inputs}} + [C][A]^{-1}[B] \right]$
- Verify the response of the state space model by calculating and comparing the frequency response of the state space model with the finite element analysis harmonic results.

2.4.4. Model Examples

To demonstrate the development of models using the techniques presented in sections 2.4.2 and 2.4.3, the results of a long, flexible cantilever beam model, with and without a piezoelectric device – similar to Figure 8, are compared below. Analytical results (only for the non-piezoelectric device model) are compared to the results from state space models generated from (1) the results from a one dimensional beam finite element model developed in MATLAB and (2) the results from a three

dimensional solid finite element model developed in ANSYS. The properties of the model are presented in Table 1.

Table 1 – Properties for cantilever beam models

Parameter	Beam Value	Piezo Value
length	36"	10.3"
width	1"	1"
thickness	1/4"	1/4"
density	0.29 lb/in^3	0.29 lb/in^3
modulus	29,000 ksi	See Eq. 43

For the beams with a piezoelectric device, lead magnesium niobate – lead titanate (PMN-(33%)PT) single crystal material was used. The pseudo-tetragonal 4mm symmetry constant strain relative permittivity matrix ($\varepsilon^S/\varepsilon_o$), the piezoelectric constant stress matrix (e), and the short circuit stiffness matrix (c^E) for the PMN-(33%)PT material are presented in Eq. 43 (Zhang, Jiang & Cao, 2001).

$$\varepsilon^S/\varepsilon_o = \begin{bmatrix} 1434 & 0 & 0 \\ 0 & 1434 & 0 \\ 0 & 0 & 680 \end{bmatrix}$$
$$e = \begin{bmatrix} 0 & 0 & 0 & 0 & 10.1 & 0 \\ 0 & 0 & 0 & 10.1 & 0 & 0 \\ -3.9 & -3.9 & 20.3 & 0 & 0 & 0 \end{bmatrix} \frac{C}{m^2}$$
$$c^E = \begin{bmatrix} 11.5 & 10.3 & 10.2 & 0 & 0 & 0 \\ 10.3 & 11.5 & 10.2 & 0 & 0 & 0 \\ 10.2 & 10.2 & 10.3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 6.9 & 0 & 0 \\ 0 & 0 & 0 & 0 & 6.9 & 0 \\ 0 & 0 & 0 & 0 & 0 & 6.6 \end{bmatrix} \times 10^{10} \frac{N}{m^2} \qquad \text{Eq. 43}$$

The results of the non-piezoelectric cantilever beams are compared in Figures 9, 10, and 11. Figure 9 presents a comparison of the natural frequencies of a non-piezoelectric cantilever beam as calculated via an analytical model (Analytical), a one dimensional beam finite element analysis state space model (1-D FEA (SS)), and a three dimensional solid finite element analysis state space model (ANSYS (SS)). As shown in the figure, there is good agreement between the models, even up to the 10th bending mode of the cantilevered beam.

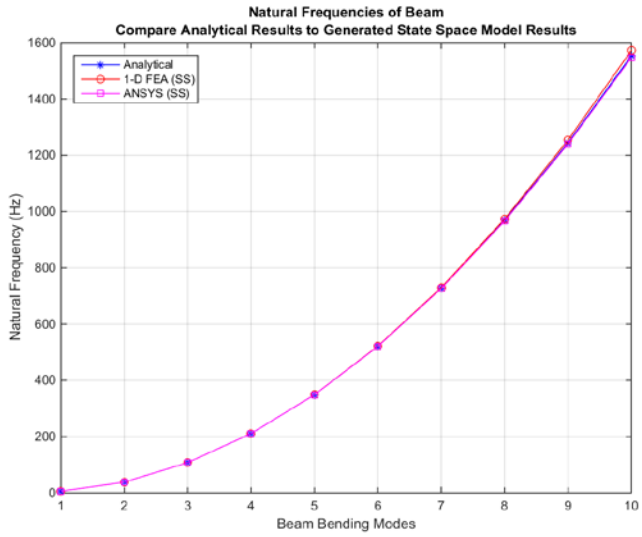


Figure 9 – Comparison of beam natural frequencies

Figure 10 presents the power sum of the harmonic frequency response of the cantilever beam models with a force located at the end of the beam. It is clear from this figure that again, the generated state space models compare well to the results obtained using ANSYS (ANSYS (FEA) – no analytical harmonic results were generated). The ANSYS results do, however, show a larger amplitude of response at the natural frequencies of the cantilever beam. This is a result of providing a small amount of modal damping to the state space models ($\xi = 0.5\%$), whereas the ANSYS model had zero damping. Figure 11 presents a comparison of the static analysis results, with a static unit force applied to the end of the cantilever beam.

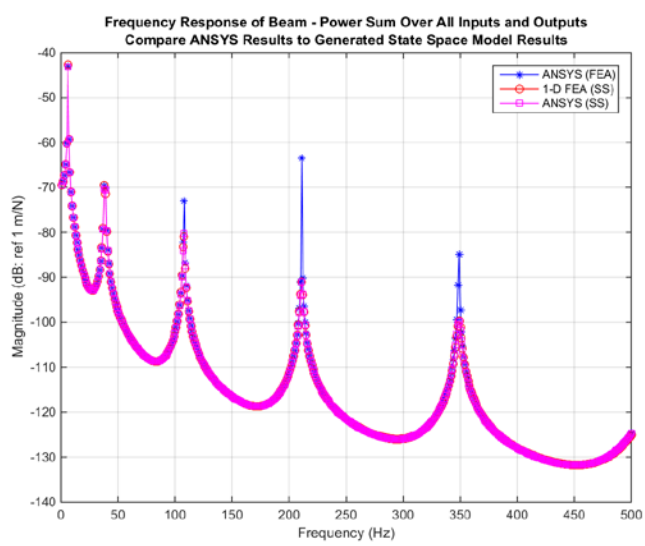


Figure 10 – Comparison of harmonic sweep analysis results: power sum over all outputs

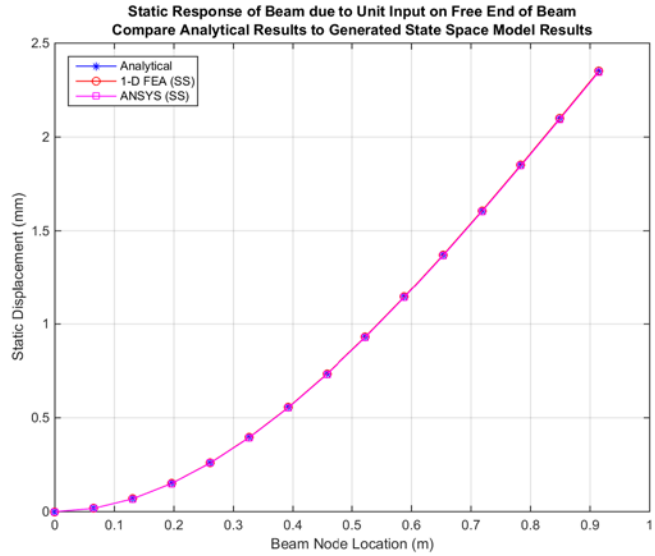


Figure 11 – Comparison of static analysis results

The results for the cantilever with a piezoelectric device, as presented in Figures 12, 13, and 14, are similar to the non-piezoelectric device cantilever results. Figure 12 presents the similar displacement frequency responses of the models for either a unit force input or a unit charge input, while Figure 13 presents the voltage frequency responses. Again, the results of the one dimensional beam finite element model closely match the results of the more refined three dimensional solid finite element model. The slight differences can be attributed to isotropic vs. non-isotropic material properties used in the respective models. Finally, Figure 14 presents the static displacement results.

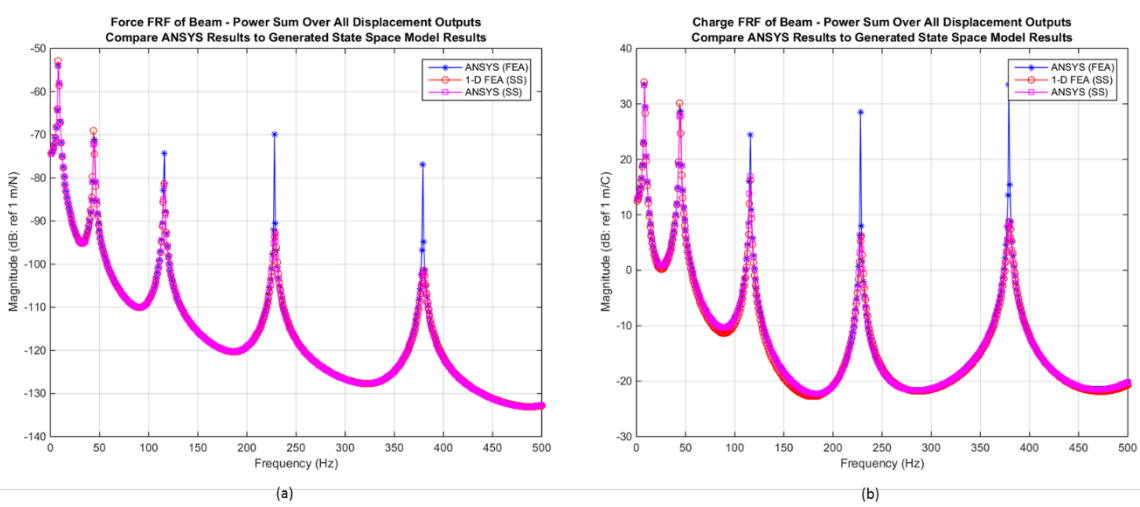


Figure 12 – Power sum displacement frequency response of piezoelectric device cantilever beam: (a) unit force input and (b) unit charge input

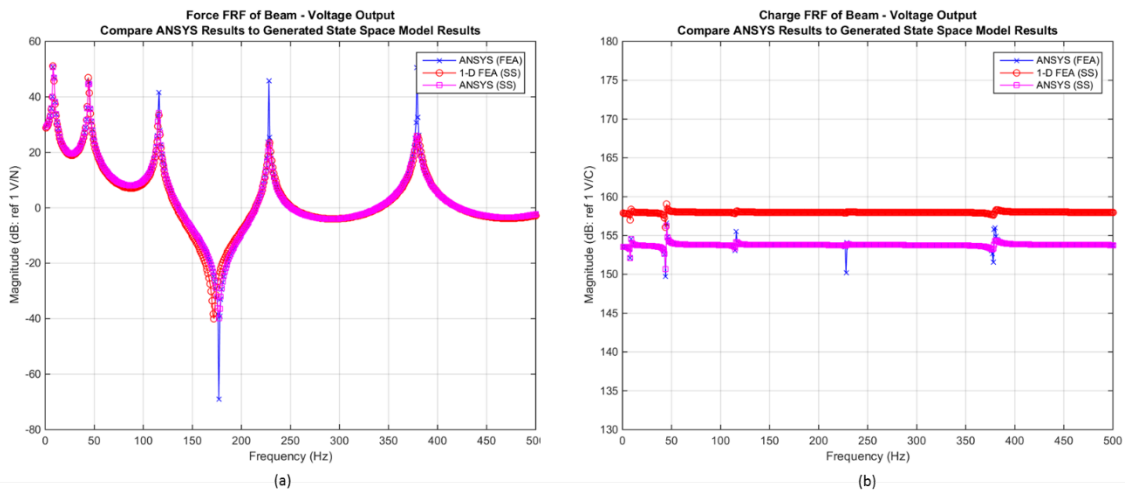


Figure 13 – Voltage frequency response of piezoelectric device cantilever beam: (a) unit force input and (b) unit charge input

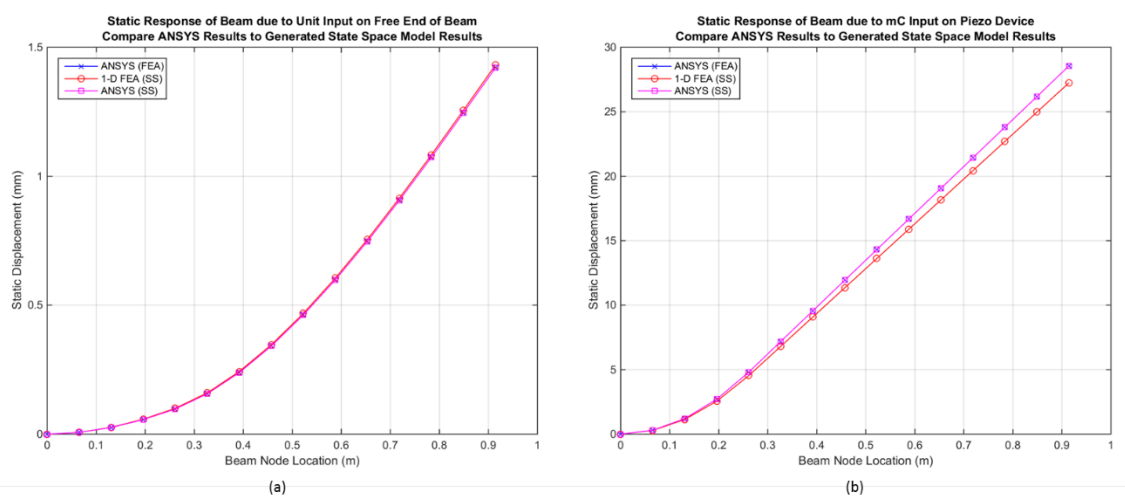


Figure 14 – Static response of piezoelectric device cantilever beam: (a) unit force input and (b) unit charge input (x 10^{-3})

It is clear from these results that the one dimensional beam finite element analysis model closely matches a three dimensional solid finite element analysis model, that the conversion of the results from the one dimensional beam and the three dimensional solid finite element analysis models into state space models is valid, and the result of the state space model closely matches the analytical results for the cantilever beam, with and without a piezoelectric device.

CHAPTER 3. IMPACT DYNAMICS

It is easy to conclude, after a brief review of literature on the topic of impact dynamics, that the accurate modeling of elastic bodies undergoing collision and deformation is difficult at best, and quite impossible at the worst (Faik & Witteman, 2000). Some of the factors that influence the behavior of the impacted bodies are the geometry of interacting bodies, material properties, elastic – plastic and shock wave propagation, hydrodynamic flow, finite strains and deflections, strain rate effects, work hardening, thermal and frictional effects, surface adhesion, gravity, kinetic effects, energy dissipation, and the initiation and propagation of failure in the colliding materials (He & Wettlaufer, 2014; Zukas, 1980). Also, there currently is a lack of understanding of essential related phenomena (e.g., friction, fracture, non-linear deformation) and typically, problem specific information is not accurately known (e.g., geometric information, pertinent environmental conditions, initial conditions of impacting bodies) (Chatterjee, 1997). Fortunately, for low velocity impacts, such as with the mechanical frequency up-conversion technique, the Hertzian Theory of Impacts may be used. Note that low velocity impacts are generally defined as the relative velocity of the impactor is much less than the speed of sound in the target material, or generally less than two meters per second (Grady, 1988; Faik & Witteman, 2000; Zukas, 1980). However, experiments have shown that Hertzian theory may be used for higher impact velocities, as long as no plastic deformation (i.e., elastic limits are not exceeded) takes place (Davis, Serayssol & Hinch, 1986).

3.1. Hertzian Theory of Impacts

When two bodies (spheres) are in contact, a deformation takes place, resulting in a contact force. If the surfaces of the bodies are continuous and non-conforming, the contact area is small compared to the geometry of the bodies, the resulting strains are small, the material properties are isotropic and linearly elastic (no plastic deformation occurs), and the surfaces are frictionless (i.e., sufficiently smooth), Hertzian theory can be used to determine the relationship between the deformation and the resulting contact force (Faik & Witteman, 2000; Grady, 1988; Johnson, 1995; Sun & Yang, 1980; Yang & Sun, 1981; Van de Wouw, de Kraker, Van Campen & Nijmeijer, 2003).

$$\mathrm{F} = \mathrm{K}\delta^{3/2} \qquad \text{Eq. 44}$$

where F denotes the contact force between the two impacting bodies, δ denotes the relative deformation, and K denotes a constant depending on the spheres' radii (R_i) and elastic properties (E_i – modulus of elasticity, and ν_i – Poisson ratio).

$$K = \frac{4}{3}\sqrt{\frac{R_1 R_2}{R_1+R_2}}\left(\frac{k_1 k_2}{k_1+k_2}\right)$$
$$k_i = \frac{E_i}{1+\nu_i^2} \qquad \text{Eq. 45}$$

Since the contact force is the derivative of the potential energy of the elastic deformation with respect to the actual deformation, an expression for the potential energy (U) may also be developed (Patrıcio, 2004; Popov, 2010).

$$U = \frac{2}{5}K\delta^{5/2} \qquad \text{Eq. 46}$$

This expression is valid for static deformations; however, it can be used as a good approximation for deformation potential energy for impacts with velocities that are much smaller than the sound velocities of the bodies involved (Patrıcio, 2004). Note that during an impact, the kinetic energy of the impactor is converted into the strain energy of the deformation. Then as the bodies return to the original pre-impact shapes, this strain energy is transferred back into kinetic energy, depending on the mass ratios of the target and the impactor. Therefore, the total energy before and after the impact, including any energy dissipated during the impact (if any is assumed), is conserved (Cross, 1999a; Cross, 1999b; Davis et al., 1986).

3.2. Impactor Model

The Hertzian contact law presented in Eq. 44 can be used to develop a simple model of the impactor and how it interacts with a target. Then, traditional vibration theory can be used to generate the response of the entire target, including any energy harvesting components (Cross, 1999b; Eringen, 1952; Hutzler, Delaney, Weaire & MacLeod, 2004; Nagurka & Huang, 2004; Van de Wouw et al., 2003; Zukas, 1980). The simplified impactor model is presented in Figure 15.

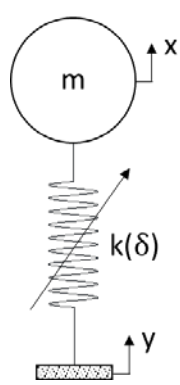


Figure 15 – Simplified impactor model

The impactor is modelled as a point mass with a non-linear spring.

$$m\ddot{x} + k(\delta)x = k(\delta)y \qquad \text{Eq. 47}$$

Note that if it is required to model the impactor with a coefficient of restitution that is less than 1 (i.e., energy is lost during the impact), a damping element could be added to the impactor model (Cross, 1999b; Nagurka & Huang, 2004; Van de Wouw et al., 2003). However, since it was assumed that both the impactor and target are metal for this work, the effects of viscous damping were assumed small and ignored (Faik & Witteman, 2000). For spring-mass systems (see section 2.1.2, for example), the impactor can be directly incorporated into the equations of motion, for when the impactor interacts with the target, and removed from the system when not interacting with the target (see Figure 16(a)). For general state space models, the impactor equation of motion is incorporated into a feedback loop with the state space model. The state space model will then be in an open loop condition, when not interacting with the impactor (see Figure 16(b)).

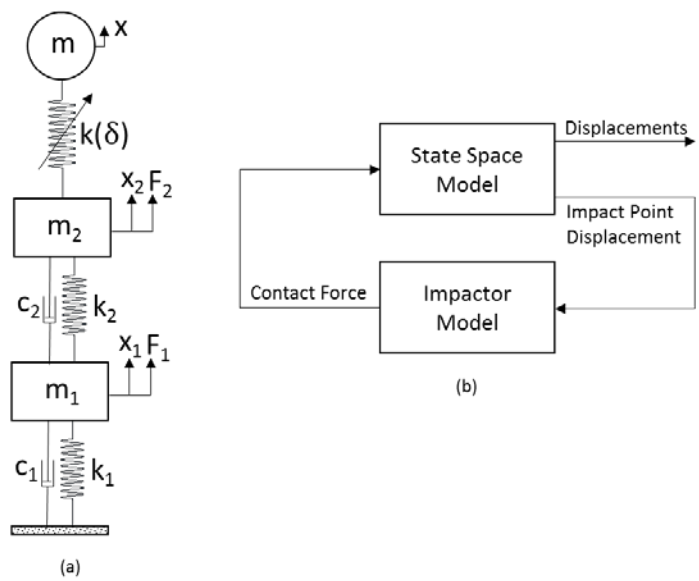


Figure 16 – Interaction of impactor and target models: (a) spring-mass systems and (b) general state space models

The following assumptions, with respect to impactor interaction with a target, were used in the system modelling.

- At the time of impact (t_o), the impact spring was not deformed.
- The mass of the impactor was equal to or less then the equivalent mass of the target; therefore, the impactor would not continue "through" the impact (i.e., no multiple hits from an impactor interaction).

- The impactor interaction was only valid while $x_{impactor} - x_{target} \leq 0$ (i.e., the impact spring could not "pull" the impactor point mass towards the target).
- Once the impactor separates from the target (i.e., no longer in contact), it can no longer interact with the target – it is removed from any further system interaction.
- The deformation and contact force relationship was the same for both the compression and recoil dynamics.

3.2.1. Linear vs. Non-Linear Spring Force

As an example, the response of a ½-inch diameter steel sphere impacting (head on) a rigid steel wall with a speed of 2 m/s is simulated. The relative deformation and velocity of the sphere is calculated. Also, the contact force between the sphere and wall is calculated. The results of this simulation are presented in Figure 17 and Figure 18. Note that if the non-linear behavior of the spring is replaced with a linear spring, response results within 5% to 10% of the non-linear spring results may be achieved. Since this is most likely within the margin of modeling error with respect to system material properties, geometry, tolerances, and boundary conditions, appropriately sized linear impact springs may be utilized to model impact simulations.

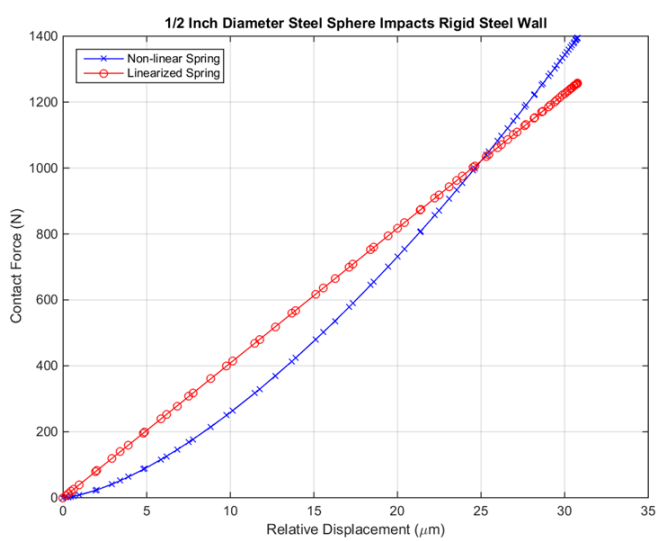


Figure 17 – Example impact spring force (non-linear and linear spring results)

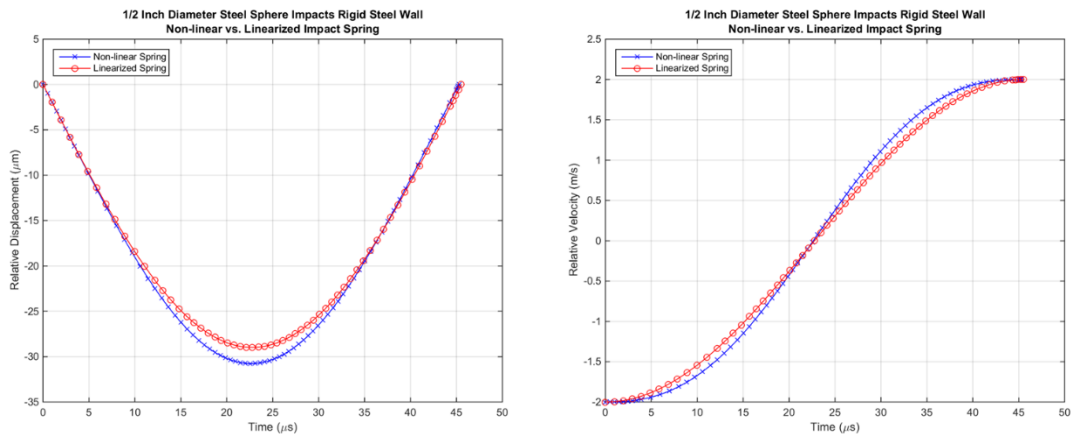


Figure 18 – Example relative displacement and velocity of impacting sphere (non-linear and linear spring results)

From Figure 18 it is clear that since the magnitude of the impactor velocity before and after the impact is the same, no energy was lost during the impact. Also, it is clear that the simulation must stop once the relative displacement reaches zero. At this time step, the sphere is no longer impacting the rigid wall and will start to move away from it with its original speed.

CHAPTER 4. MAXIMUM ENERGY TRANSFER

One goal of mechanical frequency up-conversion energy harvester design is to maximize the energy transfer within the harvester: the conversion of kinetic energy into strain and kinetic energy of the target, and then the conversion of target energy into electrical energy. Both of these topics are investigated below in the discussion of harvester design.

4.1. Impactor and Target Contact Point Design

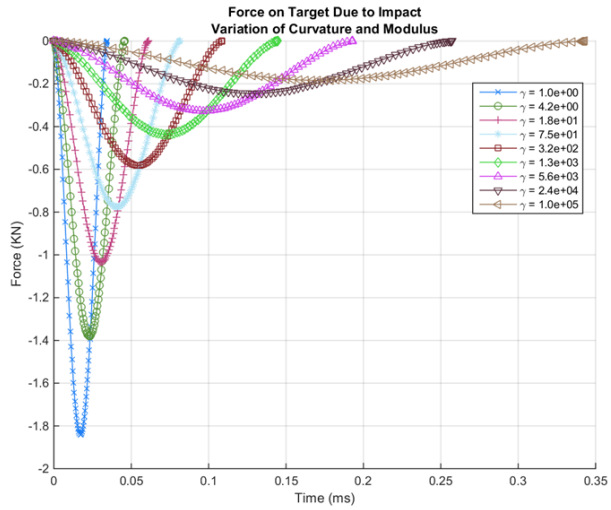
It is prudent to ask whether or not the shape or the material properties of the impact area on the target and impactor have an effect on the transfer of energy from the impactor to the target. It is clear from Eq. 46 that the radius of curvature and the Young's Modulus of the target and impactor do have some effect on the deformation and contact force. However, since the kinetic energy of the impactor is transferred into the strain energy of the deformation, the total strain energy of the impact is a known and fixed quantity; therefore, the radius of curvature and the Young's Modulus only influence the total relative deformation between the impactor and the target, which also shapes the force pulse applied to the target.

To demonstrate this effect, a simple numerical example is presented. Using the impactor model from section 3.2, a ½"-diameter steel sphere was impacted against a rigid (i.e., does not vibrate or move with respect to the impactor) target with varying radius of curvature and Young's modulus (note that the results are equally valid for a specified target while varying radius of curvature and modulus of the impactor, or varying the properties on both the target and impactor). From Eq. 45, the impact spring constant may be rewritten as a function of a non-dimensional parameter, γ.

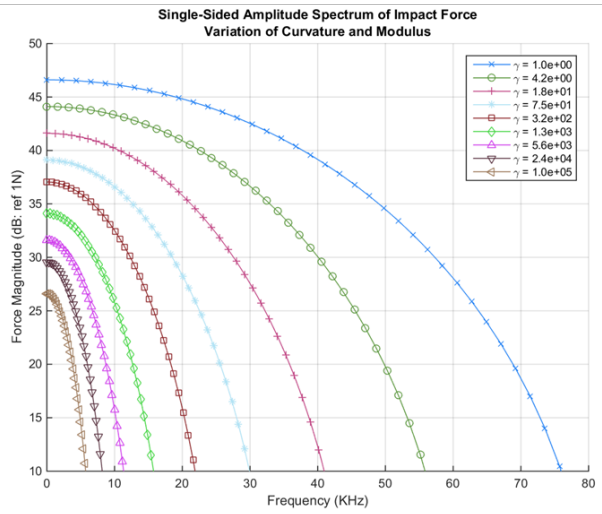
$$\begin{gathered} K(\gamma) = \frac{4}{3}\sqrt{\frac{R_1 k_1^2}{\gamma}}\ (1 \le \gamma \le \infty) \\ \gamma = (1+\alpha)(1+\beta)^2 \\ \alpha = \frac{R_1}{R_2}\ (0 \le \alpha \le \infty) \\ \beta = \frac{k_1}{k_2}\ (0 \le \beta \le \infty) \end{gathered} \qquad \text{Eq. 48}$$

As γ is varied from a value of 1.0 (a stiff flat walled target) to a large value (a soft pointed target), the total deformation, and hence the applied contact force, varies for a given initial impactor kinetic energy (see Figure 19(a)). The area under each of the impact force curves is constant – twice the initial impactor kinetic energy (deformation and restoration). Therefore, the radius of curvature and the Young's modulus of the target or the impactor have no effect on the actual energy transfer from

the impactor to the target. These properties do, however, have an effect on the shape of the impulse applied to the target. The lower the γ value, the "stiffer" the impact, thereby exciting high frequency structural modes in the target. Unfortunately, even with high values of γ to obtain a "soft" impact, since the timeframe of the impact is over a short period of time, the impact will always have the ability to excite high frequency structural modes in the target (see Figure 19(b)).



(a)



(b)

Figure 19 – Variation of impact force due to radius of curvature and material modulus (time domain and respective frequency domain response)

Note that the above analysis assumed ideal conditions and that the mass of the target and the impactor did not change as a function of radius of curvature or Young's modulus. As will be shown in the next sections, the relative mass between the impactor and the target does have a major impact on the transfer of energy from the impactor to the target.

4.2. Equivalent Mass of Impactor and Target

As shown in section 4.1, the shape and the material properties of the contact area for the impactor and the target have little influence on the amount of energy transfer from the impactor to the target body. They only influence the shape of the impact force pulse, thus controlling the highest frequency mode excited by the impact. Fortunately, just as with two billiard balls, two neutrons, or any other two impacting bodies, it can be shown that the relative mass between the two bodies influences the amount of energy transfer in the impact (Knief, 2008; Lamarsh, 1983; Sears, Zemansky & Young, 1987).

Since in this work the impactor will always be either a point mass or a block mass, the mass of the impactor is easily obtained. However, for a real system (e.g., a cantilever beam) it is much more difficult to obtain the "mass" of the system, because it is not the total mass of the system that dictates the transfer of energy, but the equivalent mass as seen by the impactor. To obtain this equivalent mass for systems that demonstrate linear spring / mass system behavior, it is assumed that the momentum at the moment of impact of the real system must be equal to an equivalent system with the correct equivalent mass. Given a real system modelled as a second order matrix differential equation (see Eq.3), the momentum for each degree of freedom of the system is given by (Petyt -1990):

$$\{P\} = \frac{\partial[T]}{\partial\dot{x}} = \frac{\partial}{\partial\dot{x}}\left(\frac{1}{2}\{\dot{x}\}^T[M]\{\dot{x}\}\right) = \frac{1}{2}\left([M]\{\dot{x}\} + [M]^T\{\dot{x}\}\right) = [M]\{\dot{x}\} \qquad \text{Eq. 49}$$

where P and T denote the system momentum vector and the system kinetic energy, respectively. Note that Eq. 49 assumes a symmetric mass matrix. To obtain the total momentum of the system in the direction of the impact, an influence vector ($\hat{r}$), which represents the results of a unit displacement of the system in the direction of impact (i.e., value of unity for degrees of freedom in the direction of interest), is used (note that $\hat{r}$ is also called the displacement results vector).

$$P_{total} = \{P\}^T\hat{r} = \{\dot{x}\}^T[M]\hat{r} \qquad \text{Eq. 50}$$

At the moment of impact, it is assumed that only the impact degree of freedom must assume the same speed as the impactor. This can be written as:

$$P_{total} = \{0 \quad \cdots \quad v_o \quad \cdots \quad 0\}[M]\hat{r} = v_o\{F\}^T[M]\hat{r} \qquad \text{Eq. 51}$$

where F is the generalized forcing vector that ensures the impact will occur at the correct degree of freedom on the real system. The total momentum of the real system, with the correct linear spring /

mass behavior, must be equal to an equivalent one dimensional system to obtain the correct equivalent mass. This must take into account linear momentum caused by the impact (assuming zero angular momentum).

$$\begin{aligned} M_{eq} v_o &= v_o \{F\}^T [M] \hat{r} \\ M_{eq} &= \{F\}^T [M] \hat{r} \end{aligned} \qquad \text{Eq. 52}$$

Once the equivalent mass of the real system has been calculated, the percent of energy remaining in the impactor after the impact ($\%E_i^{remaining}$) can be calculated (Lamarsh, 1983):

$$\%E_i^{remaining} = \left(\frac{m_i - M_{eq}}{m_i + M_{eq}}\right)^2 \times 100 \qquad \text{Eq. 53}$$

where m_i denotes the mass of the impactor. It is clear from Eq. 53 that only when the mass of the impactor is equal to the equivalent mass of the target system will all of the impactor's kinetic energy be transferred into the target.

4.3. Single Degree of Freedom System Results

To demonstrate the use of equivalent mass, a simple single degree of freedom system (see section 2.1.1) is impacted with a ½"-diameter steel sphere (see section 3.2 and Figure 20).

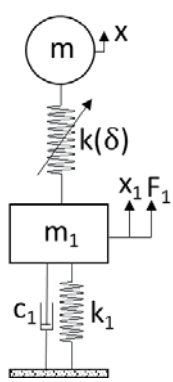
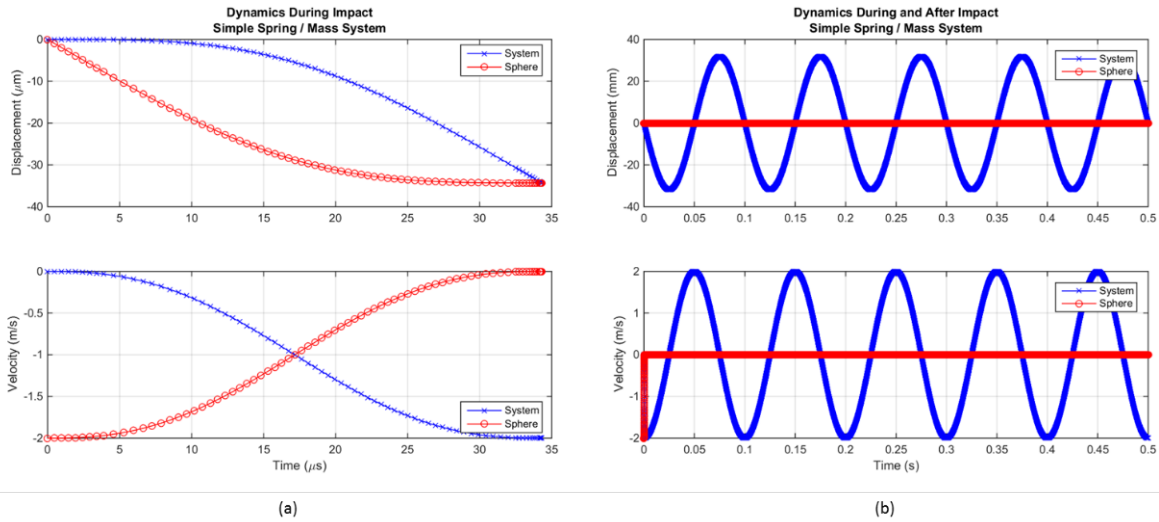


Figure 20 – Impact on a simple spring / mass system

Since the model is a single degree of freedom and the impact is "head on" (i.e., no applied moment due to the impact), it is clear from Eq. 53 that to obtain the maximum energy transfer from the impactor to the system, the mass of the system (M_{eq}) must be equal to the mass of the impactor. For convenience, it was assumed that the contact point on the system (i.e., the target) was flat (R = ∞) and that the target was the same material as the impactor. The stiffness of the system was set to obtain a system natural frequency of approximately 10 Hz. The response of the system during the

impact and after the rebound of the impactor are presented in Figure 21. The kinetic and potential energies of the system and impactor are presented in Figure 22.



(a) (b)

Figure 21 – Response of the simple spring / mass system due to impact: (a) during the impact and (b) for the total simulation time

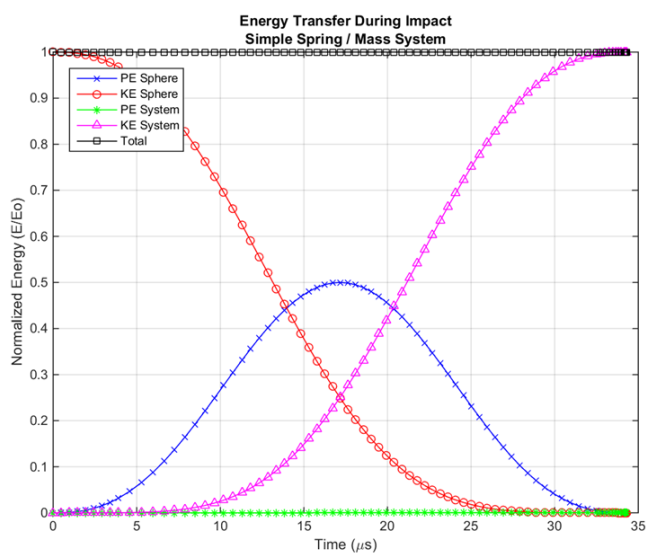


Figure 22 – Energy transfer between impactor and simple spring / mass system during impact

It is interesting to note from Figure 21(a) that at the end of the impact (approximately 34 μs), the impact spring is no longer deformed; therefore, it has zero potential energy. If the simulation were to be continued, the impact spring would then start to elongate, versus compress. Since this is physically impossible, the simulation of the impact stops and releases the impactor from the system.

Also note that at the end of the impact, the impactor has zero velocity; all of its initial kinetic energy has been transferred into the system. From Figure 22, note that due to the "softness" of the system as compared to the impact spring, very little of the impactor's kinetic energy has been transferred to the strain energy (i.e., potential energy) of the system. Most of the system's energy ended up as kinetic energy of the system. Also, note from Figure 22 the symmetry of the compression and recoil dynamics. It is also clear from these figures that no damping has been assumed both during and after the impact.

4.4. Multiple Degree of Freedom System Results

To expand on the single degree of freedom system demonstration, a multiple degree of freedom spring / mass system was developed. A five degree of freedom system (see section 2.1.2) is impacted with a ½"-diameter steel sphere (see section 3.2 and Figure 23).

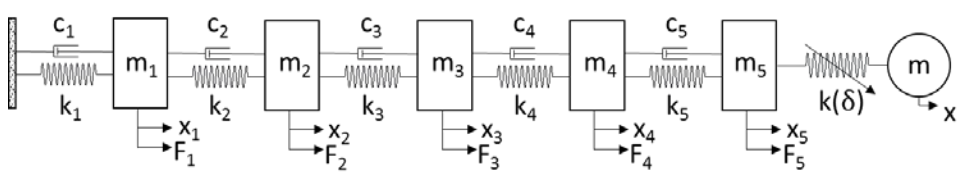


Figure 23 – Impact on a five degree of freedom spring / mass system

Since the mass matrix for this system is diagonal and the impact is "head on" (i.e., no applied moment due to the impact) to the fifth mass (top mass), it is clear from Eq. 52 that to obtain the maximum energy transfer from the impactor to the system, the top mass of the system (M_{eq}) must be equal to the mass of the impactor. As in the previous example, it was assumed that the contact point on the top mass (i.e., the target) was flat ($R = \infty$) and that it was the same material as the impactor. The mass of the remaining degrees of freedom was set by doubling the mass of the block above it (e.g., $m_4 = 2*m_5$, $m_3 = 2*m_4$). The stiffness of the system was set to obtain system natural frequencies spaced out from 0 Hz to approximately 1,000 Hz – all spring constants were set to the same value. The response of the system during the impact and after the rebound of the impactor are presented in Figure 24 and Figure 25. The kinetic and potential energies of the system and impactor are presented in Figure 26.

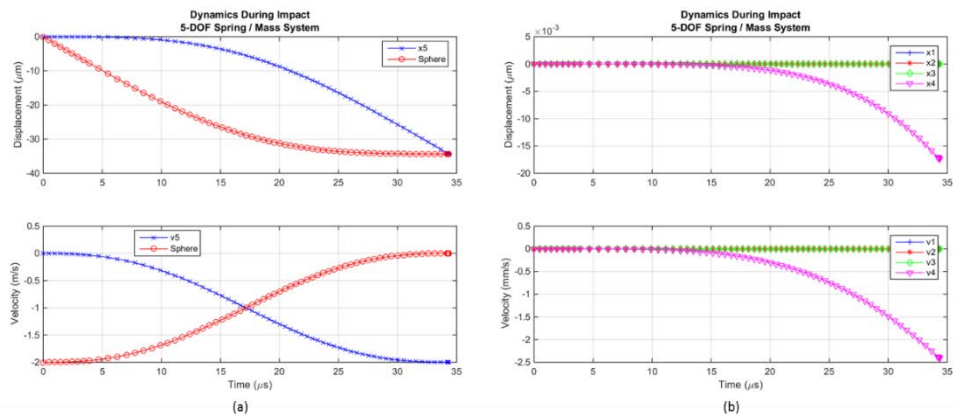


Figure 24 – Response of the 5-DOF spring / mass system due to impact during impact: (a) for impactor and target (mass 5) and (b) all other masses

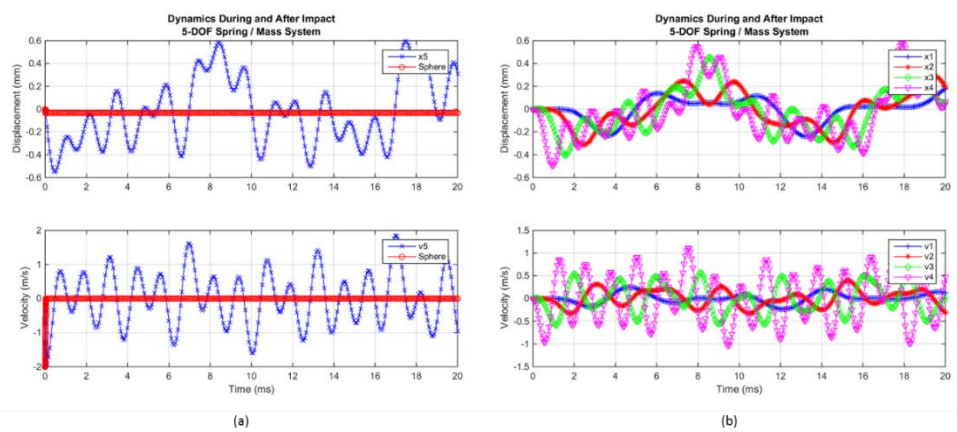


Figure 25 – Response of the 5-DOF spring / mass system due to impact for the total simulation time: (a) for impactor and target (mass 5) and (b) all other masses

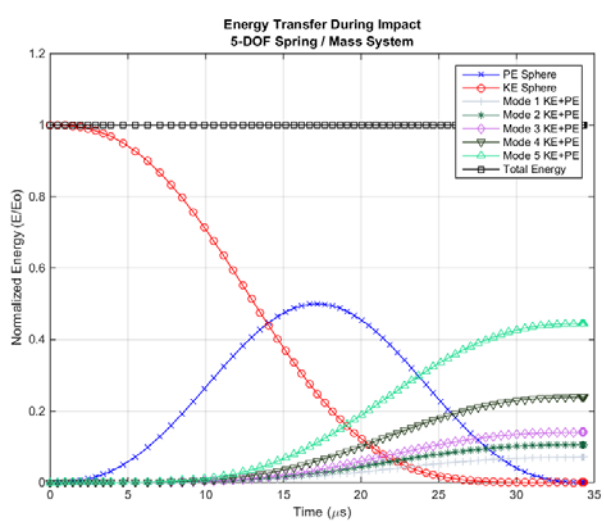


Figure 26– Energy transfer between impactor and 5-DOF spring / mass system during impact

Just as with the single degree of freedom case, Figure 24(a) shows that at the end of the impact (approximately 34 μs), the impact spring is no longer deformed; therefore, it has zero potential energy. If the simulation were to be continued, the impact spring would then start to elongate, versus compress. Since this is physically impossible, the simulation of the impact stops and releases the impactor from the system. Also note that again, just as in the single degree of freedom case, at the end of the impact, the impactor has zero velocity; all of its initial kinetic energy has been transferred into the system. From Figure 24(b), it is clear that the impact occurs over a timespan that is much too short for the rest of the system to respond to. However, this "slowness" to respond is a characteristic of the system properties and will depend on the actual system being impacted. Eventually, as shown in Figure 25(a) and (b), the entire system responds to the impact on the top mass. Figure 26 is again very similar to energy transfer that is seen for the single degree of freedom case; however, in this case, the kinetic energy from the impactor has been transferred to the entire system – the energy being shared between the modes of the system. The final amount of energy obtained by each mode is dependent on the system properties, the location of the impact, and the shape of the impact force applied to the system.

To demonstrate the energy transfer when the mass of the impactor does not match the equivalent mass of the target and to validate Eq. 53, additional simulations of the 5-DOF spring / mass system were conducted. For each simulation, the mass of the impactor was varied and then the total energy of the impactor was calculated. The simulation results were then compared to the expected results from using Eq. 53. Table 2 presents the results from the additional simulations. It is very clear that being able to match the mass of the impactor to the equivalent mass of the target is paramount in order to transfer as much energy as possible into the target system; however, up to a 40% miss-match with respect to relative mass will still ensure a majority of energy transfer into the target body. Table 2 also demonstrates the validity of Eq. 53 in estimating the amount of energy transfer.

Table 2 – Simulation results for energy transfer as a function of impactor mass

m_i/M_{eq}	Energy Remaining After Impact (%)			
	Impactor Energy		System Energy	
	Simulation	Expected	Simulation	Expected
0.2	44.5	44.4	55.5	55.6
0.4	18.4	18.4	81.6	81.6
0.6	6.3	6.2	93.7	93.8
0.8	1.2	1.2	98.8	98.8
1.0	0.0	0.0	100.0	100.0

4.5. Focusing Impact Energy into Specific Modes

It is clear from Figure 26 and as expected from Figure 19, all modes of the spring / mass system were excited. The amount of energy transferred from the impact to a specific mode depended on the impact location, initial energy of the impactor, and the shape of the input pulse from the impact (i.e., modulus and radius of curvature of the contact areas). In general, however, it is likely that the device used to convert mechanical energy to electrical energy in the energy harvester is more efficient if excited only at specific modes of the system. For example, consider the impact target of a pinned-pinned beam with a piezoelectric device that spans the entire length of the beam. Since the generated charge from the piezoelectric device is proportional to the strain in the beam, any mode that consists of both positive and negative strains will generate charges that would tend to cancel out, thus reducing the efficiency of the energy conversion. Therefore, the first mode of the beam would then generate the most charge as compared to the other modes of the beam (see Figure 27). It would be prudent to ensure that the impact only excites those modes of the target system that efficiently convert mechanical energy to electrical energy.

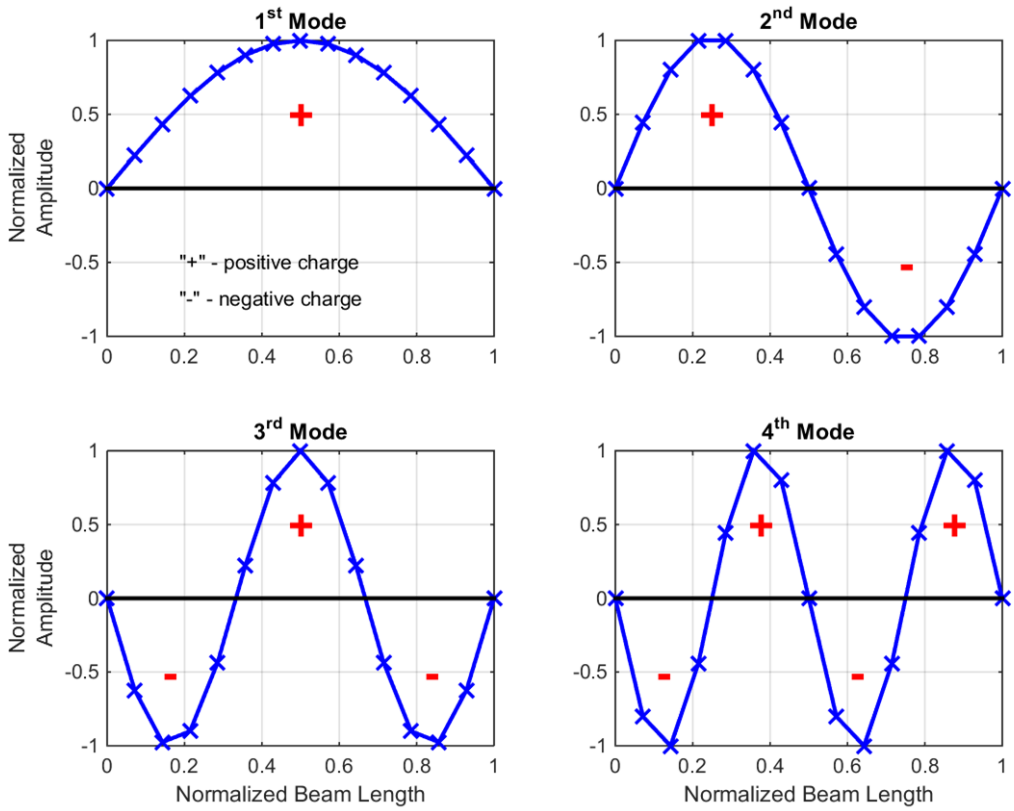


Figure 27 – Modes of a pinned-pinned piezoelectric beam: example of charge generation per mode

One method of achieving the desired "impact to modal" energy transfer is to utilize dynamic vibration absorber technology. The target system of the impact should be designed such that it is as close to a one dimensional spring / mass system as possible, thereby only exhibiting a single natural frequency. For example, a block of steel supported by a small diameter all-thread rod would be such a target system. Since it would be difficult to excite the very high frequency modes in the block or the rod, a majority of the impact energy would then be transferred into the fundamental spring / mass mode of the system. The energy conversion device is then connected to the one dimensional spring / mass system and is designed to behave as a dynamic vibration absorber for the target system. Therefore, by proper design of the energy conversion device (or dynamic vibration absorber), the majority of the impact energy will be transferred into the energy conversion device improving the efficiency of the energy harvester.

4.5.1. Focused Impact Energy for Multiple Degree of Freedom System

As a simple example, the five degree of freedom spring / mass system in section 4.4 is used as the energy conversion device and, therefore, is also used as the effective dynamic vibration absorber. The one degree of freedom spring / mass system in section 4.3 is used as the target system (see Figure 28).

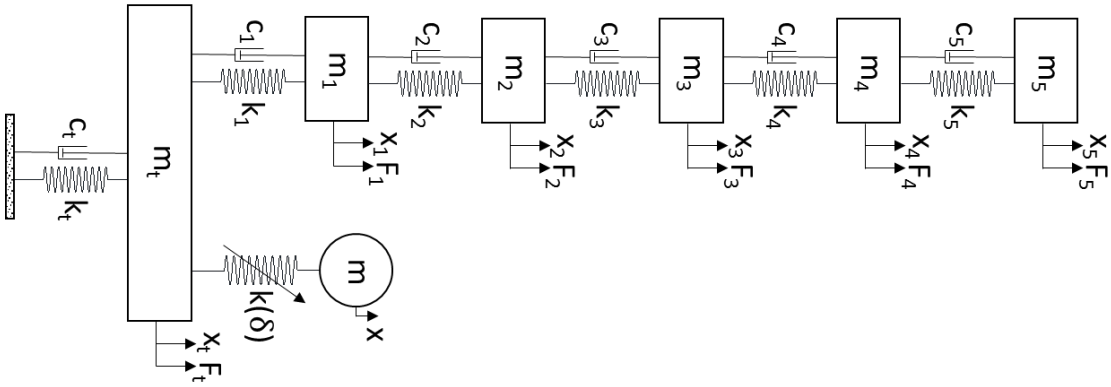


Figure 28 – System used as an example for focused energy transfer from impact to target mode

Assuming that the energy conversion device (the five degree of freedom spring / mass system) is most efficient when vibrating at the first mode, then the target system is designed to also vibrate at the first mode of the energy conversion device. The energy of the impact will excite the fundamental mode of the target system, which will then be transferred into the first mode of the excitation device, due to the "tuning" of the attached effective dynamic vibration absorber. The results are presented in Figure 29. As a second example, it is assumed that the conversion device is most efficient when vibrating at the third mode (see Figure 30).

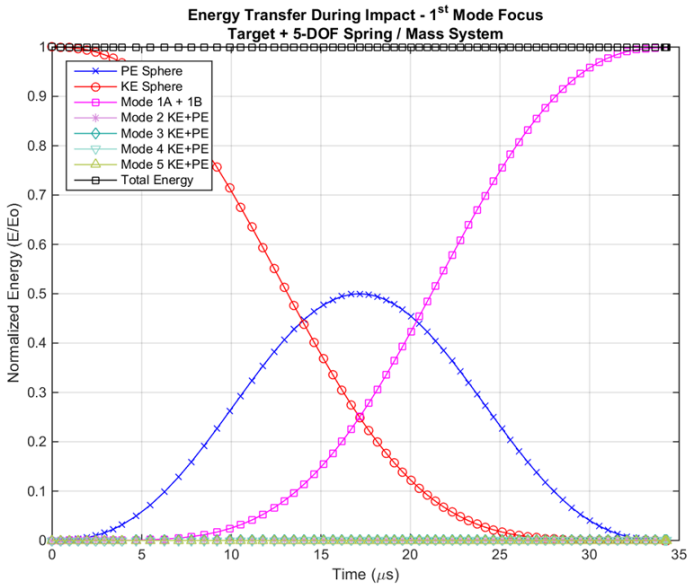


Figure 29 – Energy transfer between impactor, target, and 5-DOF spring / mass system during impact: focus on the 1st mode of 5-DOF spring / mass system

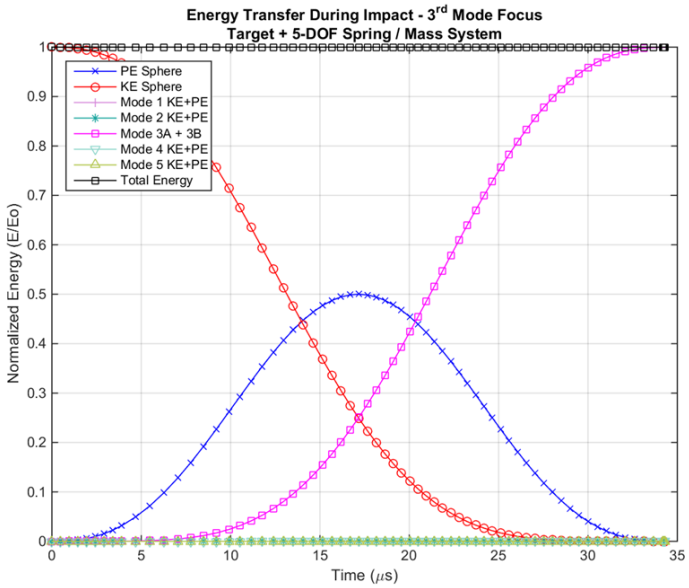


Figure 30 – Energy transfer between impactor, target, and 5-DOF spring / mass system during impact: focus on the 3rd mode of 5-DOF spring / mass system

As can be seen from Figures 29 and 30, by designing the target mass to provide a fundamental system natural frequency that approximately equals the mode of interest with respect to the energy conversion device, a majority of the impact energy can be focused into the corresponding modes,

while minimizing parasitic energy loss due to mechanical damping at other system frequencies and minimizing the reduction of conversion efficiency due to the generation of conflicting charges.

However, the use of dynamic vibration absorber technology does have a few disadvantages. The total mass of the energy conversion device (i.e., the dynamic vibration absorber) should be relatively small (< 5% to 10%) as compared to the total mass of the target to ensure a small impact of the dynamics of the one dimensional spring / mass system. If the energy conversion device is a design constraint, then the mass of the target, and therefore the impactor, must then be 90% to 95% more massive than the conversion device. However, if the impactor is a design constraint, this will then set the mass of the target, while the mass of the energy conversion system will need to be designed to be 90% to 95% of the impactor. In practical implementation, this should not be an issue.

Also, since the focus of this work is on piezoelectric devices, the electrical load of the conversion device will have an effect on the dynamic vibration absorber stiffness and damping, which will shift the mode of interest in real-time. Therefore, designing a target system to match the mode of interest in practice will be difficult. Nonetheless, the ability to focus the majority of the impact energy into system modes that increase the efficiency of the energy conversion device outweigh any of these design challenges.

4.6. Beam System Results

Multiple degree of freedom spring / mass systems are excellent tools for understanding the overall behavior of mechanical systems. Unfortunately, real systems typically can not be modelled with just simple spring / mass subsystems. Therefore, to verify that the concepts of equivalent mass and focused modal energy transfer also work with non-spring / mass systems, two cantilever beam models are used: one with a beam clamped to a non-moving base and impacted at the free end of the beam and one with a beam clamped to a moving one dimensional base. A similar beam model from section 2.4.2 was used for the numerical verification. The modelled beam is not a realistic energy generation device for energy harvesting, but it does have low frequency dynamics that helps with computational efficiency.

4.6.1. Non-Moving Base

To demonstrate the use of equivalent mass of the impact point with a real structure, a state space model generated from the results of a one dimensional piezoelectric cantilever beam finite element analysis model (similar to section 2.4.2; see Table 3 for properties) is used. It is assumed that the impact occurs at the tip of the cantilever beam (i.e., the free end). Unlike the previous spring / mass systems, because of the impact location, a non-zero moment arm now exists – the length of the beam; therefore, the equation of equivalent mass (Eq. 52) can not be used. Unfortunately, there is no simple, closed form formula to obtain the equivalent mass for the cantilever beam. The equivalent mass varies greatly as a function of the length of the beam. A very short beam – a very stiff structure – can be treated by assuming the beam is a rigid body rotating about a fixed point. Then, conservation of linear momentum, angular momentum, and kinetic energy can be used to estimate the equivalent mass of the beam. However, as the length of the beam increases, the flexibility of the beam causes the rigid body estimate to no longer be valid. A simple one dimensional numerical search can be conducted to find the beam's equivalent mass, and, therefore, the optimal mass of the impactor.

Table 3 – Properties for cantilever shunted piezoelectric beam model

Parameter	Beam Value	Piezo Value (top and bottom of beam)
length	36"	18"
width	3"	3"
thickness	1/8"	1/16"
density	0.29 lb/in^3	0.29 lb/in^3
modulus	29,000 ksi	See Eq. 43

To simulate an electrical load for the piezoelectric device, a simple resistor shunt was connected to the piezoelectric device (Hagood et al., 1990). The resistance of the resistor was tuned to obtain the maximum amount of damping to the cantilever beam (maximum energy output) for the first bending mode. The response of the system during the impact and after the rebound of the impactor are presented in Figure 31 and Figure 32. The kinetic and potential energies of the system and impactor are presented in Figure 33.

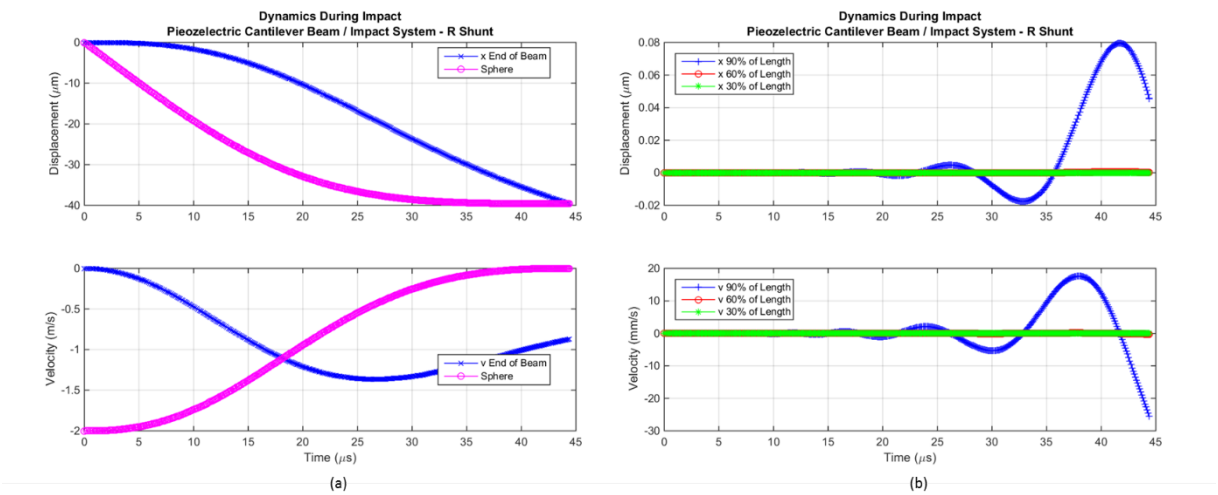


Figure 31 – Dynamic response of impactor and beam during impact: (a) impactor (sphere) and impact point (end of beam), and (b) 30%, 60%, and 90% of the beam's total length

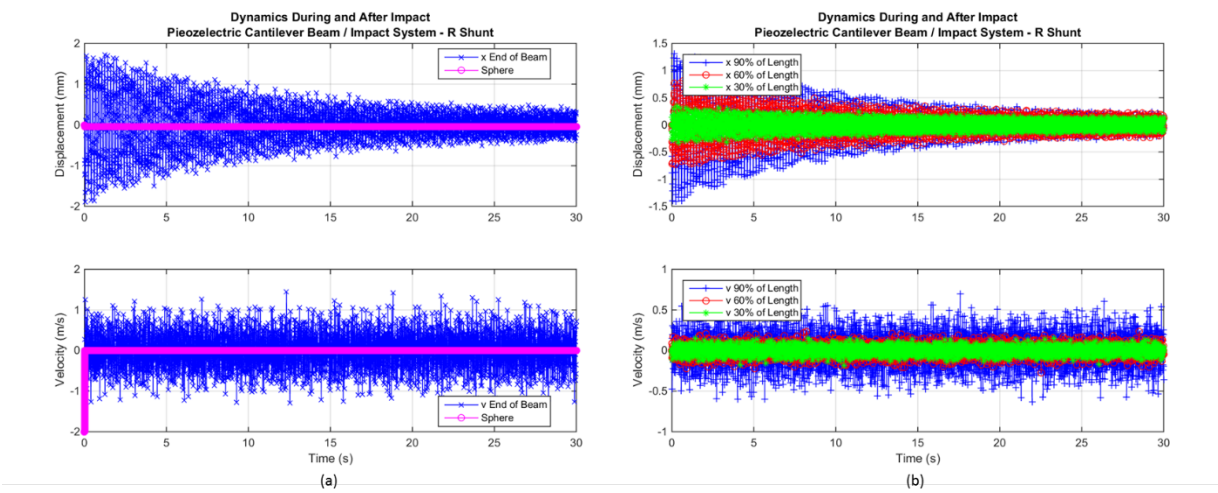


Figure 32 – Dynamic response of impactor and beam after impact: (a) impactor (sphere) and impact point (end of beam), and (b) 30%, 60%, and 90% of the beam's total length

Just as with the single and multiple degree of freedom system cases, Figure 31(a) shows that at the end of the impact (approximately 44 µs), the impact spring is no longer deformed; therefore, the simulation of the impact stops and releases the impactor from the system. Also note that again, just as in the single and multiple degree of freedom system cases, at the end of the impact, the impactor has zero velocity; all of its initial kinetic energy has been transferred into the system. From Figure 31(b), it is clear that the impact occurs over a timespan that is much too short for the rest of the system to respond – the length of the beam ensures that the effect of the impact is a localized effect. However, this "slowness" to respond is a characteristic of the system properties, especially the beam's length and stiffness, and will depend on the actual system being impacted. Eventually, as

shown in Figure 32(a) and (b), the entire system responds to the impact at the end of the beam. Figure 33 is again very similar to energy transfer that is seen for the single and multiple degree of freedom system cases – the kinetic energy from the impactor has been transferred to the entire system; the energy being shared between the modes of the system. The final amount of energy obtained by each mode is dependent on the system properties, the location of the impact, and the shape of the impact force applied to the system. Note that approximately 90% of the impact's energy was transferred into the higher modes of the system (> 200 Hz). Since the electrical load (piezoelectric device resistor shunt) was tuned to dissipate energy at the first bending mode of the system (approximately 5 Hz), very little energy, approximately 10% of the impact energy, was used by the load (or dissipated by the system) over the 30 seconds of simulation time. As shown in section 4.5.1, focusing the impact energy into a mode that is most efficient for energy production will provide a more efficient system for an energy harvester.

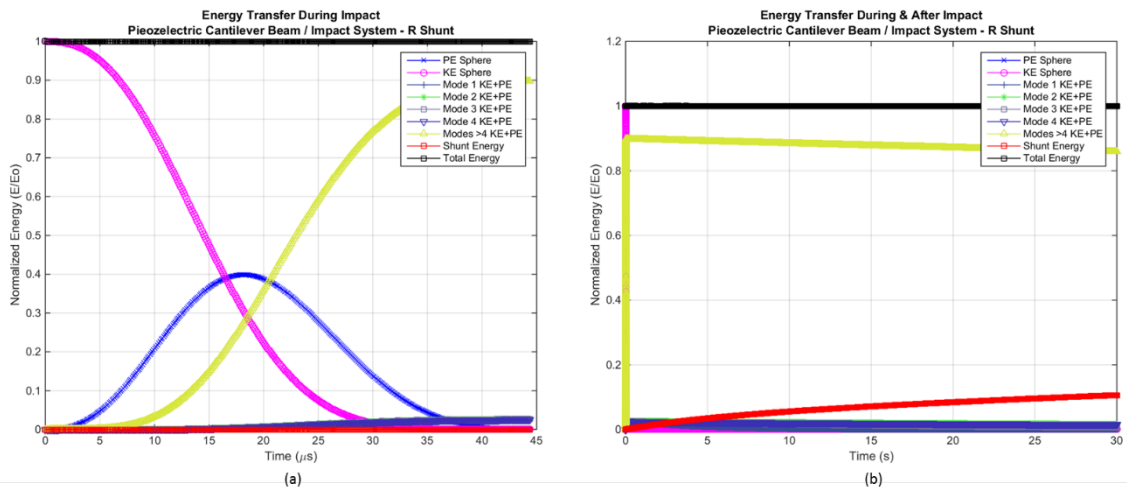


Figure 33 – System energy: (a) during the impact and (b) after the impact – shunt energy denotes energy used by electrical load

4.6.2. One Dimensional Moving Base

To demonstrate the ability to focus impact energy to a mode of interest of a real structure, the same state space model used above (section 4.6.1) was used; however, it was modified such that the cantilever beam was clamped to a one dimensional spring / mass system (see Figure 34), which was tuned to the first bending mode of the cantilever beam – the same mode the resistor shunt was tuned, and acted as the impact point. The mass of the base was set to 100 times the total mass of the piezoelectric beam, ensuring that the impactor, base, and beam system had the behavior of a

simple spring / mass system – thus allowing the use of Eq. 52 to obtain the equivalent mass of the system. The response of the system during the impact and after the rebound of the impactor are presented in Figure 35 and Figure 36. The kinetic and potential energies of the system and impactor are presented in Figure 37.

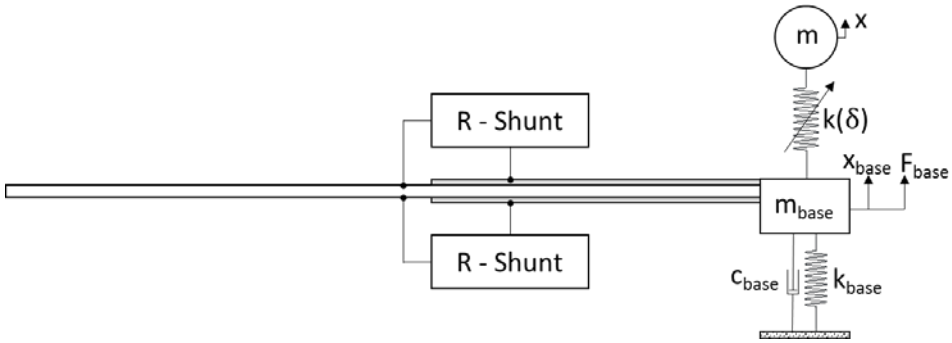


Figure 34 – Piezoelectric cantilever beam with base system – impact on base

Similar to the results of section 4.5.1, Figures 35, 36, and 37 demonstrate that attaching an energy generation device (e.g., piezoelectric beam) to a base spring / mass system tuned to the optimal energy producing mode of the system produces the desired results. Figure 37(a) clearly shows that nearly all of the impact energy was transferred into the first bending mode of the piezoelectric beam, and that the resistor shunt, Figure 37(b), was able to use (or dissipate) 90% of the generated energy over the 30 seconds of simulation time.

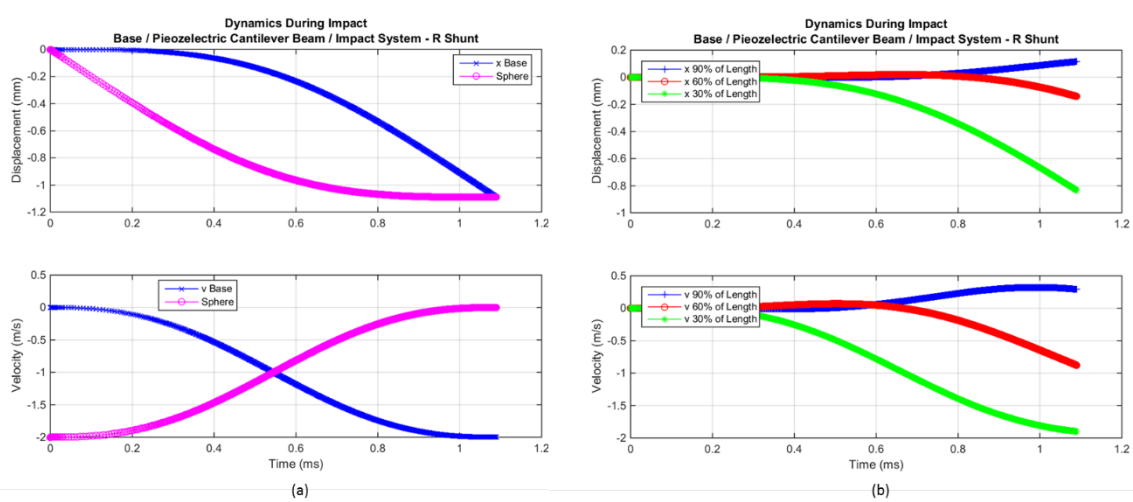


Figure 35 – Dynamic response of impactor, base, and beam during impact: (a) impactor (sphere) and impact point (end of beam), and (b) 30%, 60%, and 90% of the beam's total length

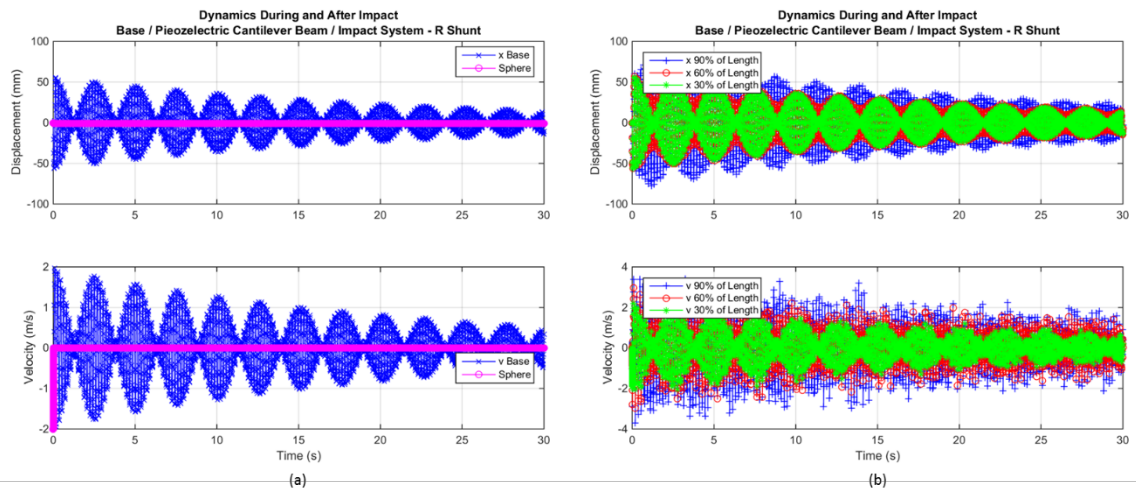


Figure 36 – Dynamic response of impactor, base, and beam after impact: (a) impactor (sphere) and impact point (end of beam), and (b) 30%, 60%, and 90% of the beam's total length

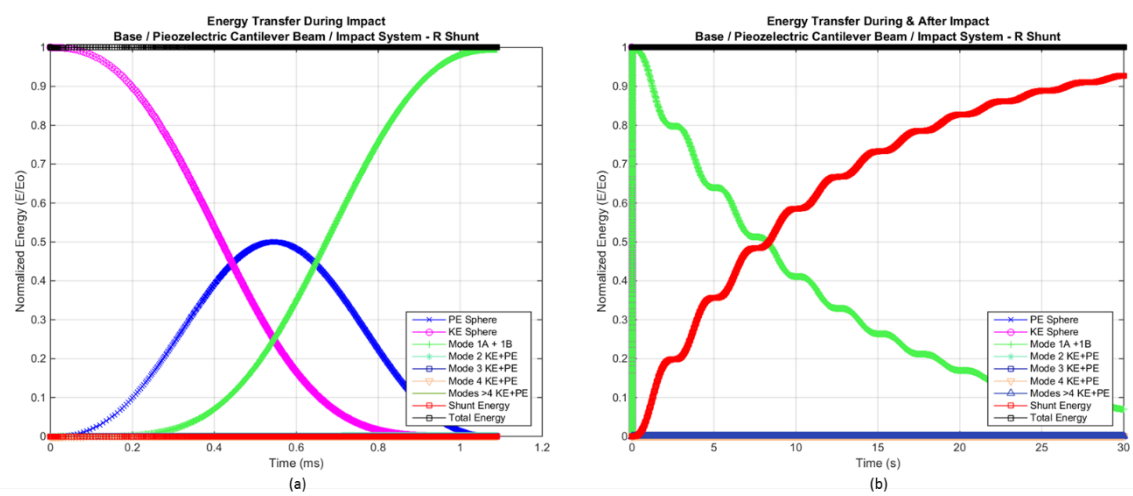


Figure 37 – System energy: (a) during the impact and (b) after the impact – shunt energy denotes energy used by electrical load

4.7. Conclusions

The results of this chapter demonstrate that the target system must be designed such that the majority of the impact energy transfers into the optimal energy generating mode for the energy harvester. It has been demonstrated that a good method to accomplish this is to treat the energy generating system as a dynamic vibration absorber attached to a target base spring / mass system, which is tuned to the mode of interest. By designing the base system to act as a simple linear spring

/ mass system, the effective mass of the system, hence the mass of the impactor, may then be obtained to achieve maximum energy transfer from the impactor to the target system. Or, if the mass of the impactor is already known, this provides a constraint in the design of the base / energy generating system.

CHAPTER 5. EXPERIMENTAL TEST SET-UP

It is prudent to develop a small experimental test assembly to verify the modeling work completed in the previous chapters before continuing to generate numerical data for the purpose of drawing design conclusions. This chapter describes the experimental test set-up, its operation, preliminary experimental data, and conclusions about the presented modeling techniques.

5.1. Power Spectral Density

It is assumed that the energy harvester will be designed for a specific structural system as its source of excitation, or at least designed with a pre-specified expected excitation frequency band and level. It is also assumed that the excitation source will be random in nature without any specific frequency characteristics (i.e., flat response with no specific modal content). Therefore, the power spectral density of the excitation source will be used as the main design input for the energy harvester. A few aspects of random vibrations and power spectral density are presented below; however, the interested reader is directed to the noted references for additional detailed information.

The power spectral density of a signal represents, depending on what response signal is being measured, the spectral distribution of energy in the signal, or the energy density associated with a specific frequency (Heinzel, Rüdiger & Schilling, 2002). It is assumed that the excitation is random, narrow band, stationary, zero mean, and normal. The power spectral density, $S(\omega)$, is defined (Rao, 1990) as the Fourier transform of the autocorrelation function, $R(\tau)$.

$$S(\omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty} R(\tau)e^{-i\omega\tau}\,d\tau \qquad \text{Eq. 54}$$

Or, after rearranging Eq. 54, the autocorrelation in terms of power spectral density can be found.

$$R(\tau) = \int_{-\infty}^{\infty} S(\omega)e^{-i\omega\tau}\,d\omega \qquad \text{Eq. 55}$$

Therefore, for a zero mean stationary random excitation, x(t), the variance of the signal may be obtained directly from the power spectral density.

$$\sigma_x^2 = R_x(0) = \int_{-\infty}^{\infty} S_x(\omega)\,d\omega \qquad \text{Eq. 56}$$

Also, if the frequency response function, $H(\omega)$, of the system forced by the random excitation is known, then the power spectral density and the variance of the response of the system are also known.

$$S_y(\omega) = |H(\omega)|^2 S_x(\omega)$$
$$\sigma_y^2 = R_y(0) = \int_{-\infty}^{\infty} |H(\omega)|^2 S_x(\omega)\, d\omega \qquad \text{Eq. 57}$$

Eq. 54 and Eq. 57 provide a set of powerful, yet simple, tools in the process of designing energy harvesters using the mechanical frequency up-conversion technique. By knowing the power spectral density of the excitation source, an estimate of the unconstrained response of the impactor and the excitation are immediately known; therefore, they can be used as design inputs, allowing the placement of the high frequency systems.

5.2. Time Signal from Power Spectral Density

Section 5.1 provides the methodology to obtain high level information (i.e., variance) from the response of the energy harvester due to a specific excitation input; however, for numerical simulations of the energy harvester response, it is desired to have time data. In order to obtain this numerical data, the known, or possibly the desired, power spectral density of the excitation needs to be converted from the frequency domain into the time domain. There are various techniques for performing this conversion, including finding a digital filter that will take a Gaussian white random noise signal as an input to provide the desired signal as an output; however, a simple technique that is typically used in Monte Carlo analysis was utilized (Shinozuka, 1972, Van de Wouw et al., 2003). Given a narrow band Gaussian random process with zero mean and power spectral density, $S(\omega_k)$, the corresponding time representation of the process, f(t), can be calculated as

$$f(t) = \sqrt{2} \sum_{k=1}^{N} (2 \times S(\omega_k)\Delta\omega)^{1/2} \times \cos(\omega_k t - \varphi_k) \qquad \text{Eq. 58}$$

where ω_k denotes the kth frequency (radians), $\Delta\omega$ denotes the frequency resolution, $N*\Delta\omega$ is the maximum frequency of the power spectral density, and φ_k denotes the kth phase angle, which has been randomly selected from a uniform distribution ranging from 0 to 2π.

5.3. Experimental Set-up – Shaker Table

A simple shaker table system was developed to help gain confidence in the modeling methodologies of previous chapters. A Parker Automation linear motor driven, high-speed / high-precision 406 LXR position table was used as the base of the experimental set-up (406T02LXRMP). The table had a 5 micrometer resolution encoder and had an overall travel of approximately 50 mm. The position table was hard mounted to a stiff and massive foundation to provide support for the system and to minimize floor “noise” from interfering with the response of the test system. An Aries Digital Drive

controller (AR-04CE), with revision 1.16 ACR Operating System, was used to control the desired position of the table. Parker Automation's single software environment, ACRView version 6.4.1, was used to program all desired position profiles and to obtain the desired acceleration power spectral density response of the system.

A slightly raised platform was connected to the table's carriage to act as a base for two parallel steel rails (3/16" wide, 12" long square bar), with adjustable separation lengths, that were used to support, and to guide, steel balls of various diameters. The platform also acted as the base for the end blocks, which were supported by short (~1" in length) 1/16" diameter all-thread rods, to simulate the high-frequency system for the up-conversion system. The distance between the two end blocks was also adjustable. The raised platform also allow for the capability to ensure that the rails were level in both the transverse and longitudinal directions. Figure 38 provides a schematic of the experimental system.

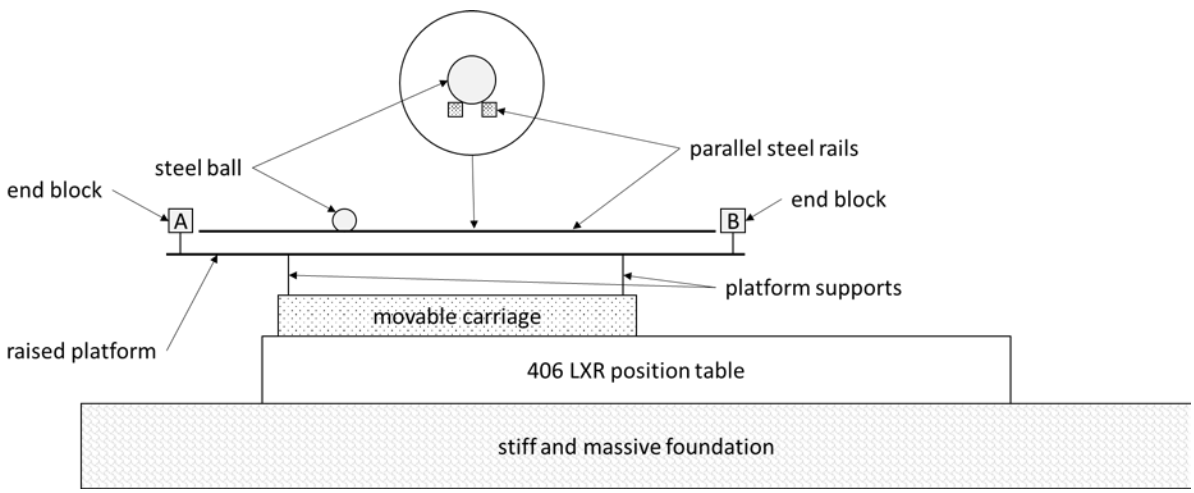


Figure 38 – Schematic of the experimental set-up

5.4. Data Acquisition Systems

The acceleration of the raised platform was measured using an Analog Devices ADXL335 (3-axis, $\pm$ 3g) accelerometer. The 10-bit analog to digital capability of an Arduino Uno microcontroller was used to sample the data from the accelerometer in the direction of interest. A MATLAB program was developed to communicate with the Arduino Uno to collect the acceleration time data for data processing, e.g., time to frequency transformation (see the appendix for sample MATLAB and Arduino programs). A sample rate of approximately 650 Hz to 700 Hz was achieved using the

Arduino / MATLAB combination; however, the sample rate was not always consistent throughout the sampling process. Therefore, the experimental data was always down sampled to a sample rate of 500 Hz.

A touch sensor was developed to count the number of times the steel ball impacted, or hit, the end blocks. A simple circuit was constructed, using the steel ball as a switch to complete the circuit (see Figure 39). The digital input and interrupt capabilities of an Arduino Uno microcontroller were used to count the number of hits during a test run and to send the running count and the total acquisition time to a computer display.

A speed sensor was developed to provide an estimate of the speed at which the steel ball was traveling when it hit the end blocks. A simple circuit was constructed using 5 mm 940 nm infrared emitter and infrared receiver light emitting diode pairs (see Figure 40). As the steel ball passed the infrared receivers, a jump in voltage was generated. The digital input capability of an Arduino Uno microcontroller was used to calculate the time difference between the voltage jumps and, knowing the distance between the two receivers, an estimate of speed was generated and sent to a computer display.

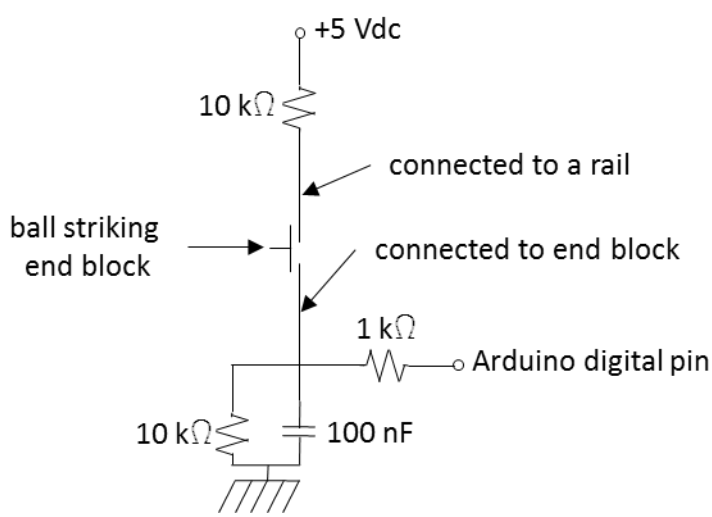


Figure 39 – Touch sensor circuit – one for each end block

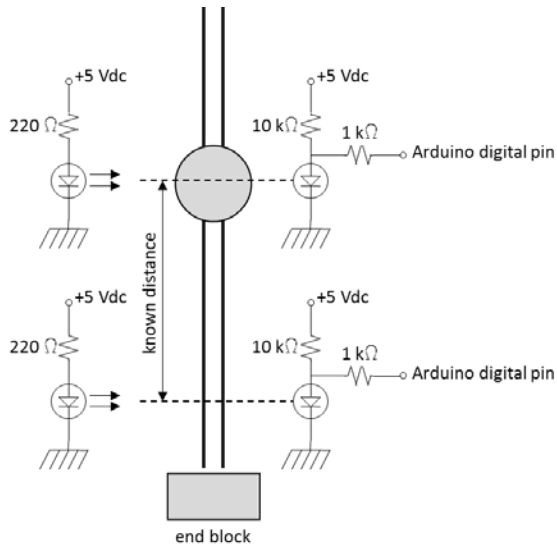


Figure 40 – Speed sensor circuit – one for each end block

5.5. Position Table Profile

The Aries Digital Drive controller was programmed to move the position table carriage to a specific position, arriving at the desired position at the end of a specific time interval. The minimum time interval the controller could achieve without causing errors in the desired position profile was 30 milliseconds. Therefore, the highest theoretical frequency content possible in the position profile was approximately 16 Hz (i.e., ½ * 1/[0.03 seconds] – Nyquist frequency). However, practically speaking, it is typically understood that to obtain "good" frequency data, the highest frequency content of the profile should be approximately 1/5 to 1/10 of the maximum possible frequency. The maximum frequency content expected in the position profile was therefore limited to 5 Hz.

Another limitation on frequency content was due to the total travel of the carriage. The total travel was approximately 50 mm; however, the controller would shut down the table if it would calculate that the table could exceed the limits of the table due to the dynamics of the system (i.e., time to decelerate and stop would be greater than the remaining distance to the hard stops of the table). Therefore, the total travel was limited to a total of 40 mm (± 20 mm) to avoid exceeding any limits associated with travel. As expected, the lower the frequency content of the position profile, the larger the travel of the position table. Also, the higher the expected magnitude of the response, the larger the travel of the position table. These limits put a restriction on the low frequency content of

the position profile to 1 Hz, and a limitation of the expected maximum power spectral density acceleration response of the table.

With the upper frequency content set, F_{high} = 5 Hz, the lower frequency content set, F_{low} = 1 Hz, and with a maximum power spectral density acceleration response, S_1 (see Figure 41), a MATLAB program was developed to calculate a position profile – a location for the table every 30 milliseconds – using Eq. 58. The MATLAB program also generated an ACRView program to carry out the motion for the table, which was copied into the controller's software environment and downloaded directly into the controller.

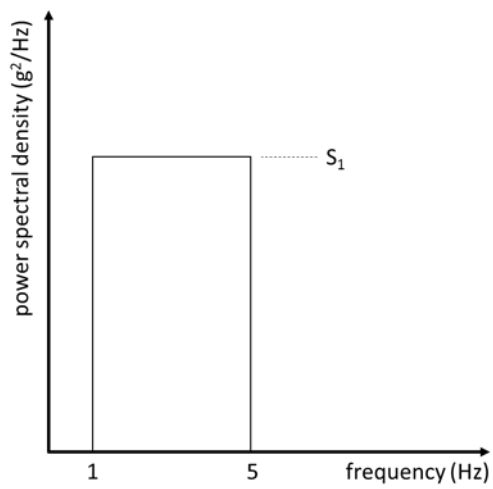


Figure 41 – Desired ideal acceleration power spectral density for position table

Note that the maximum acceleration power spectral density, S_1, to be reasonably sure (i.e., to a high probability) that the table travel limits (40 mm) are not exceeded may be estimated by using Eq. 54 and letting a fraction of the travel limit equal to the position variance:

$$S_1^{\text{No Limit}} = \left(\frac{\text{total travel limit (mm)}/2}{\text{PF}}\right)^2 \times \frac{3(2\pi)^4}{(9.8)^2(1000)^2} \times \left(\frac{F_{\text{high}}^3 F_{\text{low}}^3}{F_{\text{high}}^3 - F_{\text{low}}^3}\right)\left(\frac{g^2}{\text{Hz}}\right) \qquad \text{Eq. 59}$$

where PF denotes the probability factor of not exceeding the travel limit (i.e., PF = 2 for a 95.5% chance of not exceeding limit, PF = 3 for a 99.7% chance of not exceeding limit, PF = 4 for a 99.99% chance of not exceeding limit). Note that the same analysis can be performed to calculate the maximum acceleration power spectral density, S_1, to be reasonably sure (i.e., to a high probability) that the steel ball does not slip. From Section 2.3, the steel ball will not slip if the acceleration of the ball is less than $\mu_s g$; therefore, setting the acceleration variance of the ball times the probability factor equal to μ_s provides the limiting acceleration variance (in units of g). Also, from Section 2.3

the frequency response acceleration function of the steel ball is a constant (2/7), independent of the steel ball's mass and diameter. Therefore, from Eq. 54 the magnitude limit for position table acceleration to prevent ball slip is:

$$S_1^{\text{No Slip}} = \left(\frac{\mu_s}{PF}\right)^2 \times \frac{1}{\left|{}^2/_7\right|^2} \times \left(\frac{1}{F_{high} - F_{low}}\right) \left(\frac{g^2}{Hz}\right) \qquad \text{Eq. 60}$$

For the current test set-up, the acceleration limit to prevent exceeding the table travel limitation, $S_1^{\text{No Limit}}$, will always be smaller than the acceleration limit to prevent steel ball slipping, $S_1^{\text{No Slip}}$; therefore, the no slip assumption was always valid to a high degree of probability.

Two additional limitations on the test assembly were identified during testing. First, the speed of the table could not exceed approximately 200 mm/s between position points. It appears that this was more of a controller issue / limitation, than a position table limitation. To ensure this limit was not exceeded, once the position profile was calculated from the desired power spectral density function, the speed between each position point was calculated. An intermediate position point was added to the profile between any two points for which the speed limit was violated. Finally, the Aries controller could not contain more than approximately 130 seconds to 150 seconds of profile data. If a longer position profile was downloaded to the controller, the controller would cease to function. To overcome this limitation, position profiles of 130 seconds were used for all test runs; however, after executing approximately 130 seconds of position data, the controller would then loop back to start again at the beginning of the profile. The system would continuously operate in this manner, until the program was terminated.

5.6. Experimental Results

A few test runs were conducted on the experimental test set-up to verify the operation of the system and to gain confidence in the modeling methodologies presented in previous chapters.

5.6.1. Position Table Acceleration

To verify the use of Eq. 58 and the development of code for the ACRView controller software, a few acceleration power spectral density profiles with varying maximum amplitude and different frequency ranges were developed. The profiles were then converted into a position time domain signal and written as part of an ACRView program, which was then downloaded to the position table controller. Acceleration time data from the platform, measured as per section 5.4, was then

recorded in MATLAB for approximately 5 minutes to generate the actual power spectral density of the platform. The calculated and desired acceleration power spectral density for the platform are presented in Figures 42 and 43, and are discussed in section 5.6.3.

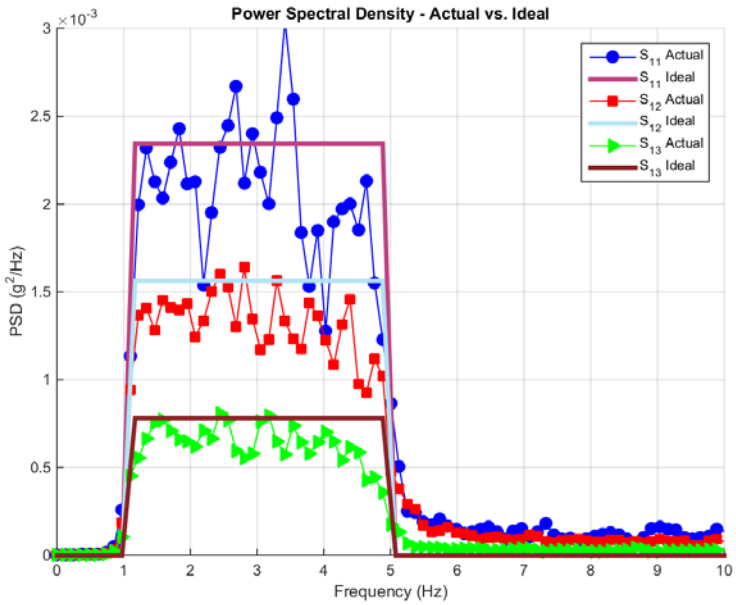


Figure 42 – Platform acceleration power spectral density: actual and ideal (1 Hz to 5 Hz)

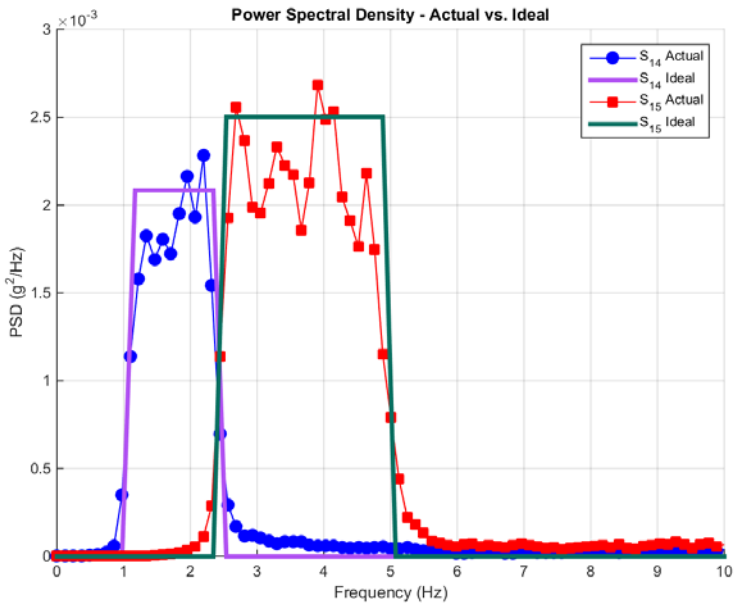


Figure 43 – Platform acceleration power spectral density: actual and ideal
(1 Hz to 2.5 Hz, 2.5 Hz to 5 Hz)

5.6.2. Number of Hits on End Blocks

To verify that the impactor / ground modeling technique of section 2.3 estimates the behavior of the test assembly, five different random position profiles were generated using a fixed ideal acceleration power spectrum density function and the technique presented in section 5.2. Then, the position profiles were downloaded to the Aries controller and used with the table end blocks at three different separation distances: 5”, 7.5”, and 10”. Over a period of 10 minutes, at 2 minute intervals, the number of times the steel ball contacted the end blocks was counted. The same five random excitations were also used to simulate the response of the system, so that the number of simulated impacts could also be counted. The experimental and numerical results are presented in Tables 4, 5, and 6; in Figures 44 and 45; and are discussed in section 5.6.3. Note that only nominal properties were used in the numerical simulations, and that no optimization of these values was conducted to reduce the differences between results.

Table 4 – Experimental impactor hits on end blocks for different random excitations

Distance (in)	Time (min)	Run 1		Run 2		Run 3		Run 4		Run 5	
		A-hits	B-hits	A-hits	B-hits	A-hits	B-hits	A-hits	B-hits	A-hits	B-hits
5	2	22	32	14	23	17	23	23	34	12	19
5	4	43	61	44	49	35	53	43	61	35	46
5	6	64	88	64	71	47	74	62	90	50	68
5	8	84	109	92	99	59	83	84	118	73	86
5	10	107	132	113	124	77	111	107	151	94	105
7.5	2	11	13	25	16	15	7	17	13	11	15
7.5	4	18	26	45	36	42	25	37	28	26	29
7.5	6	38	38	59	54	58	39	55	37	51	42
7.5	8	48	51	77	65	72	57	77	60	68	58
7.5	10	76	70	87	83	92	71	101	76	81	71
10	2	4	14	20	16	13	15	24	14	23	13
10	4	15	30	45	29	33	28	48	33	42	23
10	6	34	40	57	44	47	47	72	48	65	40
10	8	46	60	67	57	68	63	97	62	78	56
10	10	70	75	80	67	88	68	106	85	88	77

Table 5 – Numerical impactor hits on end blocks for different random excitations

Distance (in)	Time (min)	Run 1		Run 2		Run 3		Run 4		Run 5	
		A-hits	B-hits	A-hits	B-hits	A-hits	B-hits	A-hits	B-hits	A-hits	B-hits
5	2	22	22	14	14	15	15	16	17	19	20
5	4	43	44	34	40	30	29	36	34	41	43
5	6	65	68	55	64	49	48	57	53	56	58
5	8	83	85	70	81	72	73	75	68	77	83
5	10	108	108	93	106	86	86	102	92	96	105
7.5	2	14	16	9	10	4	7	6	5	9	10
7.5	4	29	31	29	27	14	17	20	19	22	26
7.5	6	41	39	42	38	29	33	33	36	35	39
7.5	8	54	52	57	52	43	48	43	46	53	56
7.5	10	67	64	67	65	57	62	54	59	60	66
10	2	11	11	8	5	5	5	14	12	7	9
10	4	18	18	19	19	14	14	24	19	19	22
10	6	25	26	28	29	23	21	35	29	26	32
10	8	32	35	41	43	34	31	47	40	35	43
10	10	41	44	50	53	45	45	58	51	39	49

Table 6 – Average impactor hits per minute on a single end block at various separation distances

Distance (in)	Experiment	Numerical	% Error
5.0	11.2	9.8	12%
7.5	8.1	6.2	23%
10.0	8.0	4.8	41%

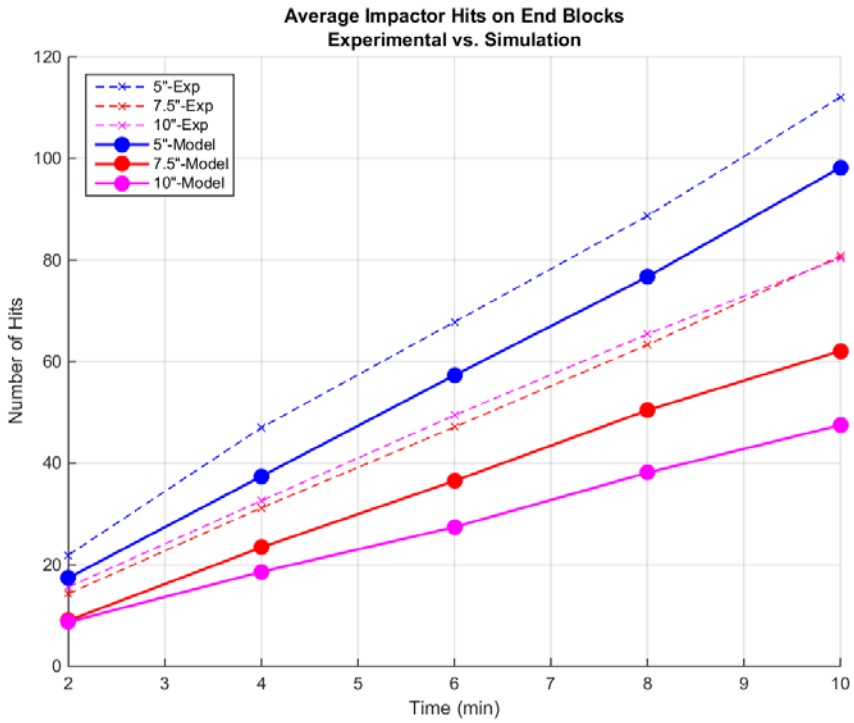


Figure 44 – Average impactor hits as a function of time

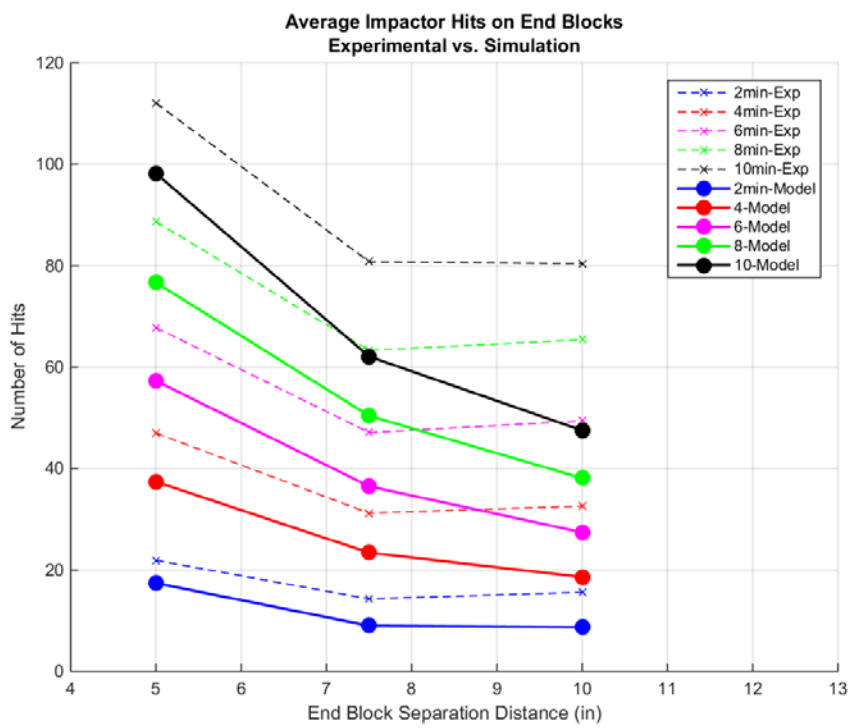


Figure 45 – Average impactor hits as a function of end block separation distance

5.6.3. Conclusions: Numerical Simulations and Experimental Data

Figures 42 and 43 present the calculated and desired acceleration power spectral density of the platform. These figures indicate that the overall technique, going from a desired acceleration power spectral density profile to a controller position profile, does provide the desired effect – providing an experimental acceleration power spectral density that estimates the desired profile. However, note that the overall acceleration response between 4 Hz to 5 Hz is lower than at the lower frequencies. This is due to the controller limitation of only being able to move the table to a position every 30 milliseconds. A faster position table controller would allow more energy at the higher frequencies and enable a better fit to the ideal acceleration power spectral density profile. Also, note the variability of the experimental acceleration response. Longer sampling times (e.g., much greater than 5 minutes) and additional input signal variation (e.g., position signals longer than 130 seconds) would be needed to obtain a smoother acceleration power spectral density.

Tables 4, 5, and 6 and Figures 44 and 45 present data on the number of times the steel ball (i.e., the impactor) made contact, or the number of hits, with either end block A or B (see Figure 38). From these tables and figures, it is clear that the numerical model of the system does provide the same type of behavior of the real system, especially when the distance between end blocks is such that many hits occur per minute; however, a few differences can be noted. First, from the data tables, it is clear that for the numerical simulations the number of hits on the end blocks are nearly the same between A and B for each time period of evaluation; however, from the experimental data, the number of hits on the end blocks tend to be different. This is most likely attributed to the construction of the experimental test assembly. Although care was given to ensure the steel rails were straight and level, in actuality, the rails were slightly bent (in both longitudinal and transverse directions); therefore, preventing a truly level set of rails with a constant separation distance on which the steel ball could travel.

Also, note that the numerical simulations actually predicted a lower hit count than was measured. This is most likely attributed to how the impactor contact with the end blocks was modeled in the simulations. First, it was assumed that the impact between the steel ball (impactor) and the end blocks was perfectly elastic (i.e., no kinetic energy was lost during the impact – a coefficient of restitution of 1.0). Clearly, in actuality this type of impact is not physically possible and that some energy during the impact must be lost to the system. Secondly, to help with numerical stability, it

was assumed that the steel ball would “stick” to the end blocks (i.e., have the same position and speed as the impacting end block) if the steel ball’s speed after an impact with an end block was nearly zero and the end block was accelerating in a direction towards the opposite end block. The steel ball would then be “released” from the end block once the end block started to accelerate in a direction away from the opposite end block, either transitioning from a positive to a negative acceleration, or transitioning from a negative to a positive acceleration. Without the “stick” to the end block assumption, the tolerances of the ordinary differential equation solvers would, in some impacts, allow the steel ball to move through the end blocks, ending up on the opposite side of the end block, which is clearly not physical. However, the tolerances chosen to implement the “stick” assumption may have been set too large (i.e., the speed of the steel ball after impact was too high to “stick”); therefore, causing a lower simulated number of hits with the end blocks.

Finally, the largest factor in the discrepancy between numerical simulation and experimental results was most likely how the dynamics of the end blocks contributed to the overall response of the steel ball. It was assumed that the duration of the impact was of such short duration that the dynamics of the end blocks were negligible (i.e., the end blocks act as rigid walls during the impact). It was also assumed that the dynamic response of the end blocks after each impact would dissipate (become negligible) before another impact could occur, and it was assumed that the ground motion did not excite the dynamics of the end blocks. Through observation of the test assembly, it was easy to verify that the ground motion (platform motion) did excite the end blocks; however, the overall effect of the dynamics of the end blocks could not be easily verified by direct observation. So, the modeling assumptions of the impact dynamics between the steel ball and the end blocks most likely had a large effect on the number of simulated hits being less than actually measured.

Another possible contributor to the differences in the numerical and experimental results may be due to the dynamics of the position table foundation and of the raised platform. It was assumed in the numerical simulations that the dynamics of both the foundation and the raised platform could be neglected; however, in the experimental acceleration results, high frequency dynamics from the platform and foundations could actually be observed.

CHAPTER 6. SUMMARY AND RECOMMENDED FUTURE WORK

This chapter provides a brief summary of the objectives of this work, describes how the objectives were satisfied, and provides a few recommended activities to continue this work on impact mechanical frequency up-conversion piezoelectric energy harvesters.

6.1. Summary

The purpose of this thesis was to develop design guidelines for impact mechanical frequency up-conversion piezoelectric energy harvesters. The specific objectives established in section 1.4 were to develop guidelines for:

- Maximum energy transfer from the impactor to high frequency system
- High frequency system design to maximize energy generation from piezoelectric device
- Impactor size and placement of high frequency system to maximize impactor / high frequency system interaction for a given excitation spectrum

Chapter 3 provides design guidelines to ensure maximum energy transfer from the impactor to high frequency system. The numerical simulation data indicate that the mass of the impactor and the high frequency system should be nearly equal to achieve maximum energy transfer between the two impacting objects. Chapter 4 provides high frequency system design guidelines to maximize energy generation from piezoelectric devices. The numerical simulation data indicate that the high frequency system should be designed as a single modal frequency system (i.e., system has a single dominate mode of vibration, e.g., a solid steel block supported by a small diameter all-thread rod), and the piezoelectric power generating device should be designed as a dynamic vibration absorber, attached to, and with a matching natural frequency of, the single modal frequency system. Finally, Chapter 5 takes the first few steps in developing design guidelines for impactor size and placement of high frequency system to maximize impactor / high frequency system interaction for a given excitation spectrum. The numerical simulation and experimental results indicate that the high frequency systems should be separated at a distance that is much less than the variance of either of the ground motion or the unconstrained impactor motion; however, the development of specific design guidance for the separation distance and the effects of the impactor size is left as recommended future work.

6.2. Recommended Future Work

This section provides a few recommended activities that would be beneficial for the continued work on developing design guidelines for impact mechanical frequency up-conversion piezoelectric energy harvesters. In order to develop specific design guidelines, further numerical simulations and experimental test cases must be conducted. More confidence in the numerical model must be gained by obtaining additional experimental data. Only then can numerical simulations be conducted to generate the data necessary to develop the specific design guidelines. The recommended future activities are categorized into two groups (i.e., numerical modeling and experimental equipment) and are focused on work that will provide the required confidence in the numerical simulations.

- Numerical Modeling – The below activities will add complexity to an already fairly complex numerical model; thus, a balance must be made between the behavior of the simulated results and the speed at which it takes the numerical models to complete.
    - Add the dynamics of the raised platform to the system model.
    - Add the dynamics of the position table foundation to the system model, unless the foundation can be stiffened against vibrations (see Experimental Equipment section).
    - Explicitly model the impact between the impactor and the end blocks, allowing for end block dynamics. This would entail multiple impacts due to the vibration of the end blocks.
- Experimental Equipment – Improvements to the experimental set-up would allow for obtaining better data; therefore, providing additional supporting information for numerical simulations.
    - Stiffen the position table foundation with respect to vibrations.
    - Redesign the raised platform and rail system. A stiff plate with a machined channel to act as the rail system is recommended. This will prevent the ability of having a variable rail separation distance, but the benefit from a uniform and level rail system outweighs this drawback.
    - Use a signal analyzer for obtaining acceleration power spectral density results. The sample rate of the Arduino Uno was adequate for this work; however, higher sample rates would provide much “cleaner” frequency data.

- Redesign the touch and speed sensors. The simple sensors developed, along with their circuits, were not robust and did not always provide reliable data. The improved touch sensors would allow for smaller end block separation distances (i.e., be able to count an increase in impacts per minute). The improved speed sensors would allow verification of energy transfer information from the numerical simulations.
- Increase the speed of the Aries controller, allowing for higher frequency excitation to the impactor. The Aries Digital Drive controller (AR-04CE) has been deemed "obsolete" by Parker Automation due to issues with the controller. The new replacement controller, Intelligent Parker Amplifier (IPA), may provide the increase in frequency response of the position table system, while also allowing longer position profiles to be loaded into the controller.

APPENDIX A. SAMPLE ARDUINO UNO PROGRAMS

This appendix provides sample Arduino Uno sample programs for performing analog to digital conversions (A.1), acting as a touch sensor (A.2), and acting as a speed sensor (A.3).

A.1. Analog to Digital Conversion

```
/*****************************************************************
A2D_Speed2Matlab_RevA (from ReadAnalogVoltage_Play_RevC)

  0 to +5 Volts
  Read 2 channels and send to MATLAB

  LRCorr, 06Sep16
  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
  Communications Matlab <--> Arduino
  Arduino file 1 for use with Matlab file 1
*****************************************************************/
// Constants:
   int ledPin=13;
   int SpeedPin1_A = A5;    // analog pin for A front sensor
   int SpeedPin2_A = A4;    // analog pin for A back sensor
   //int SpeedPin1_A = A3;    // analog pin for B front sensor
   //int SpeedPin2_A = A2;    // analog pin for B back sensor

// Variables:
   double t = 0;
   double s = 0;

// the setup routine runs once when you press reset:
void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(115200);
   digitalWrite(ledPin,HIGH);
   establishContact();  // send a byte to establish contact until receiver responds
   digitalWrite(ledPin,LOW);
```

```
  s=micros();
}//void setup()

// the loop routine runs over and over again forever:
void loop() {
  // read the input on analog pin 0:
  int sensorValue1A = analogRead(SpeedPin1_A);
  int sensorValue2A = analogRead(SpeedPin2_A);
  t=micros()-s;
  Serial.println(t);
  Serial.println(sensorValue1A);
  Serial.println(sensorValue2A);
}//void loop()

 void establishContact() {
    while (Serial.available() <= 0) {
      // Serial.write('A');   // send a capital A
      Serial.println('A');   // send a capital A
      //Serial.println('A', BYTE);   // send a capital A
      delay(300);
    }//while(Serial …
 }//void estabishContact()
```

A.1. Touch Sensor

```
/*****************************************************************
BallHitDetection_RevD
 *
* Based on StateChangeDetection program in the Arduino Examples under Digital.
 *
 * LRCorr, 19Aug16
*****************************************************************/

// Constants:
const int  HitPin_A = 2;    // the pin that the pushbutton is attached to
const int  HitPin_B = 3;    // the pin that the pushbutton is attached to
const int  DebounceDelay = 400; //milli-seconds to delay after a hit

// Variables:
volatile int HitCount_A;      // number of hits on A
int lastHitCount_A = 0;       // previous number of hits on A
volatile int HitCount_B;      // number of hits on B
int lastHitCount_B = 0;       // previous number of hits on B
double stot = 0;      // start time of program
double temp = 0;    // used for various calcs

void setup() {
  // initialize the button pin as a input:
  pinMode(HitPin_A, INPUT);
  pinMode(HitPin_B, INPUT);
  // initialize serial communication:
  Serial.begin(9600);
  // initialize interrupt
  attachInterrupt(digitalPinToInterrupt(HitPin_A),HitCounter_A,RISING);
  attachInterrupt(digitalPinToInterrupt(HitPin_B),HitCounter_B,RISING);
  Serial.println("-----> Ready to Count Hits...");
  Serial.println(" ");
  Serial.println(" ");
  Serial.println(" ");
```

```
}//void setup()

void HitCounter_A(){
  HitCount_A++;
}

void HitCounter_B(){
  HitCount_B++;
}

void loop() {
  if (HitCount_A != lastHitCount_A){
      detachInterrupt(digitalPinToInterrupt(HitPin_A));
      Serial.println(" ");
      Serial.print("---> Total time observing speed = ");
      temp = (millis() - stot)/1e3/60;
      Serial.print(temp,2);
      Serial.println(" minutes");
      Serial.print("Number of Hits:  A = ");
      Serial.print(HitCount_A);
      Serial.print(", B = ");
      Serial.println(HitCount_B);
      delay(DebounceDelay);
      lastHitCount_A = HitCount_A;
      attachInterrupt(digitalPinToInterrupt(HitPin_A),HitCounter_A,RISING);
  }//if(HitCount_A ...

  if (HitCount_B != lastHitCount_B){
      detachInterrupt(digitalPinToInterrupt(HitPin_B));
      Serial.println(" ");
      Serial.print("---> Total time observing speed = ");
      temp = (millis() - stot)/1e3/60;
      Serial.print(temp,2);
      Serial.println(" minutes");
      Serial.print("Number of Hits:  A = ");
```

```
    Serial.print(HitCount_A);
    Serial.print(", B = ");
    Serial.println(HitCount_B);
    delay(DebounceDelay);
    lastHitCount_B = HitCount_B;
    attachInterrupt(digitalPinToInterrupt(HitPin_B),HitCounter_B,RISING);
  }//if(HitCount_B ...

}//void loop()
```

A.2. Speed Sensor

```
/*****************************************************************
  BallSpeed_RevC (based on RevB)

  Program to calculate the average speed of the ball per hit on "end."
  No comms with MATLAB.

  LRCorr, 13Sep16
 *****************************************************************
//Logic
int TOWARDS_A = 1;      // 1 = ball moving towards Aend
int TOWARDS_B = 1;      // 1 = ball moving towards Aend
int READING_SPEED = 0;   // 1 = inprocess of finding speed
int RESET = 0;          // 1 = senor 1A has been set high and then back to low

//Constants
int SpeedPin1_A = A5;   // analog pin for A front sensor
int SpeedPin2_A = A4;   // analog pin for A back sensor
int SpeedPin1_B = A3;   // analog pin for B front sensor
int SpeedPin2_B = A2;   // analog pin for B back sensor
int NumAverages_A = 0;     // number of speed averages
int NumAverages_B = 0;     // number of speed averages
float PassLevel1_A = 4.0;   // % of background increase to cause a pass
float PassLevel2_A = 8.0;   // % of background increase to cause a pass
float PassLevel1_B = 4.0;   // % of background increase to cause a pass
float PassLevel2_B = 8.0;   // % of background increase to cause a pass
double Distance_A = 22.225; // distance between A front and back sensors (mm) ~14/16"
double Distance_B = 22.225; // distance between A front and back sensors (mm) ~14/16"
double TimedOut = 2 * 1e6; // READING_SPEED timeout (us)- did not "see" ball hit

//Variables
int Background1_A = 0;   // background reading of A front sensor
int Background2_A = 0;   // background reading of A back sensor
int Background1_B = 0;   // background reading of B front sensor
int Background2_B = 0;   // background reading of B back sensor
```

```
int Pass1_A = 0;        // value that indicates a pass on A front sensor
int Pass2_A = 0;        // value that indicates a pass on A back sensor
int Pass1_B = 0;        // value that indicates a pass on A front sensor
int Pass2_B = 0;        // value that indicates a pass on A back sensor
int voltageValue1_A = 0;
int voltageValue2_A = 0;
int voltageValue1_B = 0;
int voltageValue2_B = 0;
double s = 0;      // start time of program
double stot = 0;      // start time of program
double tpass = 0; // time pass front sensor
double Vave_A = 0;  // cumulative moving average of speed for A
double Vave_B = 0;  // cumulative moving average of speed for B
double temp = 0;    // used for various calcs

// the setup routine runs once when you press reset:
void setup() {
  // initialize serial communication at 9600 bits per second:
  Serial.begin(9600);
  Serial.println("---> Obtaining background for speed sensors");
  Serial.println("Time Remaining (s) = ");
  for (int mm = 0; mm < 15; mm++) {
    delay(1000);
    voltageValue1_A = analogRead(SpeedPin1_A);
    voltageValue2_A = analogRead(SpeedPin2_A);
    voltageValue1_B = analogRead(SpeedPin1_B);
    voltageValue2_B = analogRead(SpeedPin2_B);
    temp = (voltageValue1_A + mm * Background1_A) / (mm + 1);
    Background1_A = temp;
    temp = (voltageValue2_A + mm * Background2_A) / (mm + 1);
    Background2_A = temp;
    temp = (voltageValue1_B + mm * Background1_B) / (mm + 1);
    Background1_B = temp;
    temp = (voltageValue2_B + mm * Background2_B) / (mm + 1);
    Background2_B = temp;
```

```
  Serial.print(15 - mm);
  Serial.print(", ");
}
Pass1_A = Background1_A * (PassLevel1_A);
Pass2_A = Background2_A * (PassLevel2_A);
Pass1_B = Background1_B * (PassLevel1_B);
Pass2_B = Background2_B * (PassLevel2_B);
Serial.println(" ");
Serial.println(" ");
Serial.println(" ");
Serial.println("---> A Background Levels");
Serial.print("A1 = ");
Serial.print(Background1_A);
Serial.print(", A2 = ");
Serial.println(Background2_A);
Serial.println(" ");
Serial.print("Pass 1 =");
Serial.println(Pass1_A);
Serial.print("Pass 2 =");
Serial.println(Pass2_A);
Serial.println(" ");
Serial.println("---> B Background Levels");
Serial.print("B1 = ");
Serial.print(Background1_B);
Serial.print(", B2 = ");
Serial.println(Background2_B);
Serial.println(" ");
Serial.print("Pass 1 =");
Serial.println(Pass1_B);
Serial.print("Pass 2 =");
Serial.println(Pass2_B);
Serial.println(" ");
Serial.println(" ");
Serial.println(" ");
s = micros();
```

```
  stot = millis();
  Serial.println("---> Calculating Average Speed of Ball");
} //void setup()

// the loop routine runs over and over again forever:
void loop() {
  // read A front sensor:
  voltageValue1_A = analogRead(SpeedPin1_A);
  voltageValue1_B = analogRead(SpeedPin1_B);
  ///////////////////
  // for A end
  ///////////////////
  if (voltageValue1_A > Pass1_A && TOWARDS_A ) {  // ball is moving towards A end
    tpass = micros() - s;
    READING_SPEED = 1;
    Serial.println("  ");
    Serial.println("---> Passed Sensor 1A going towards end");
    Serial.println("  ");
    while (READING_SPEED) {
      while (voltageValue1_A > Pass1_A && !RESET) { //read 1A until ball is passed sensor
        voltageValue1_A = analogRead(SpeedPin1_A);
        voltageValue2_A = analogRead(SpeedPin2_A);
      }//while(voltageValue1_A ...)
      RESET = 1; //sensor 1A is no longer above pass value
      voltageValue1_A = analogRead(SpeedPin1_A);
      voltageValue2_A = analogRead(SpeedPin2_A);
      if (voltageValue1_A > Pass1_A) { // ball did not hit - reversed direction
        READING_SPEED = 0;
        RESET = 0;
        Serial.println("---> No Hit - Reversed Direction!");
        while (voltageValue1_A > Pass1_A) { //read 1A until ball is passed sensor
          voltageValue1_A = analogRead(SpeedPin1_A);
          voltageValue2_A = analogRead(SpeedPin2_A);
        }//while(voltageValue1_A ...)
      }
```

```
    else if (voltageValue2_A > Pass2_A) { // ball is going to hit
      temp = (Distance_A / (micros() - s - tpass) * 1e6  + NumAverages_A * Vave_A) / (NumAverages_A + 1);
      Vave_A = temp;
      TOWARDS_A = 0;
      READING_SPEED = 0;
      RESET = 0;
      NumAverages_A++;
      Serial.println("  ");
      Serial.print("---> Total time observing speed = ");
      temp = (millis() - stot)/1e3/60;
      Serial.print(temp,2);
      Serial.println(" minutes");
      Serial.print("---> Number of A Averages = ");
      Serial.print(NumAverages_A);
      Serial.print(", Ave Speed A = ");
      Serial.print(Vave_A,0);
      Serial.println(" mm/s");
      Serial.print("---> Number of B Averages = ");
      Serial.print(NumAverages_B);
      Serial.print(", Ave Speed B = ");
      Serial.print(Vave_B,0);
      Serial.println(" mm/s");
      Serial.println("  ");
    }
    else if (micros() - s - tpass > TimedOut) { // ball never passed 1A - reversed direction
      READING_SPEED = 0;
      RESET = 0;
      Serial.println("---> No Hit - Timed Out!");
    }
    else {
      // keep looking for a pass on front or back sensor
    } //if(voltageValue1_A ...
  } //while(READING_SPEED)
}
else if (voltageValue1_A > Pass1_A && !TOWARDS_A) { //ball is moving away from end
```

```
  Serial.println("  ");
  Serial.println("---> Passed Sensor 1A going away from end");
  Serial.println("  ");
  TOWARDS_A = 1;
  while (voltageValue1_A > Pass1_A) { //read 1A until ball is passed sensor
    voltageValue1_A = analogRead(SpeedPin1_A);
    voltageValue2_A = analogRead(SpeedPin2_A);
  }//while(voltageValue1_A ...)
}//elseif(volgateValue1_A ...)
else {
  // keep looking for a pass on front sensor
} //if(voltageValue1_A ...
//////////////////
// for B end
//////////////////
if (voltageValue1_B > Pass1_B && TOWARDS_B ) {  // ball is moving towards B end
  tpass = micros() - s;
  READING_SPEED = 1;
  Serial.println("  ");
  Serial.println("---> Passed Sensor 1B going towards end");
  Serial.println("  ");
  //delay(1);  // small delay for bounce
  while (READING_SPEED) {
    while (voltageValue1_B > Pass1_B && !RESET) { //read 1B until ball is passed sensor
      voltageValue1_B = analogRead(SpeedPin1_B);
      voltageValue2_B = analogRead(SpeedPin2_B);
    }//while(voltageValue1_B ...)
    RESET = 1; //sensor 1B is no longer above pass value
    voltageValue1_B = analogRead(SpeedPin1_B);
    voltageValue2_B = analogRead(SpeedPin2_B);
    if (voltageValue1_B > Pass1_B) { // ball did not hit - reversed direction
      READING_SPEED = 0;
      RESET = 0;
      Serial.println("---> No Hit - Reversed Direction!");
      while (voltageValue1_B > Pass1_B) { //read 1B until ball is passed sensor
```

```
    voltageValue1_B = analogRead(SpeedPin1_B);
    voltageValue2_B = analogRead(SpeedPin2_B);
  }//while(voltageValue1_B ...)
}
else if (voltageValue2_B > Pass2_B) { // ball is going to hit
  temp = (Distance_B / (micros() - s - tpass) * 1e6  + NumAverages_B * Vave_B) / (NumAverages_B + 1);
  Vave_B = temp;
  TOWARDS_B = 0;
  READING_SPEED = 0;
  RESET = 0;
  NumAverages_B++;
  Serial.println("  ");
  Serial.print("---> Total time observing speed = ");
  temp = (millis() - stot)/1e3/60;
  Serial.print(temp,1);
  Serial.println(" minutes");
  Serial.print("---> Number of A Averages = ");
  Serial.print(NumAverages_A);
  Serial.print(", Ave Speed A = ");
  Serial.print(Vave_A,0);
  Serial.println(" mm/s");
  Serial.print("---> Number of B Averages = ");
  Serial.print(NumAverages_B);
  Serial.print(", Ave Speed B = ");
  Serial.print(Vave_B,0);
  Serial.println(" mm/s");
  Serial.println("  ");
}
else if (micros() - s - tpass > TimedOut) { // ball never passed 1A - reversed direction
  READING_SPEED = 0;
  RESET = 0;
  Serial.println("---> No Hit - Timed Out!");
}
else {
  // keep looking for a pass on front or back sensor
```

```
      } //if(voltageValue1_B ...
    } //while(READING_SPEED)
  }
  else if (voltageValue1_B > Pass1_B && !TOWARDS_B) { //ball is moving away from end
    Serial.println("  ");
    Serial.println("---> Passed Sensor 1B going away from end");
    Serial.println("  ");
    TOWARDS_B = 1;
    while (voltageValue1_B > Pass1_B) { //read 1B until ball is passed sensor
      voltageValue1_B = analogRead(SpeedPin1_B);
      voltageValue2_B = analogRead(SpeedPin2_B);
    }//while(voltageValue1_B ...)
  }//elseif(volgateValue1_B ...)
  else {
    // keep looking for a pass on front sensor
  } //if(voltageValue1_B ...
} //void loop()
```

APPENDIX B. SAMPLE MATLAB PROGRAMS

This appendix provides sample MATLAB programs for performing analog to digital conversions (B.1), converting a power spectral density function into a time function (B.2), and converting a time function into an ACRView program (B.3).

B.1. Analog to Digital Conversion

```
%%%%%%%%%%%%%%%%%%%%%%%%%%
% A2D_Speed2Matlab_RevA.m
%
% From A2D_Play_RevD.m
% Sample two Arduino analog channels, convert to voltage
% 0 to +5V
% integer values only, no conversion
%
% LRCorr, 06Sep16
%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
close all
clc
priorPorts = instrfind; % finds any existing Serial Ports in MATLAB
delete(priorPorts); % and deletes them

numSec=1/8*60;  %time of data recording
MaxFs=1000; %max assumed sample frequency
t=zeros(1,numSec*MaxFs);
v1=zeros(1,numSec*MaxFs);
v2=zeros(1,numSec*MaxFs);

s1 = serial('COM3');    % define serial port
s1.BaudRate=115200;               % define baud rate
set(s1, 'terminator', 'LF');    % define the terminator for println
fopen(s1);

try                             % use try catch to ensure fclose
                                % signal the arduino to start collection
w=fscanf(s1,'%s');              % must define the input % d or %s, etc.
if (w=='A')
    display(['Collecting data']);
    fprintf(s1,'%s\n','A');     % establishContact just wants
                                % something in the buffer
end

i=0;
t0=tic;
while (toc(t0)<=numSec)
    i=i+1;
    t(i)=fscanf(s1,'%f')/1e6;  %reading micro seconds on Arduino
    v1(i)=fscanf(s1,'%d');      % must define the input % d, %f, %s, etc.
    v2(i)=fscanf(s1,'%d');      % must define the input % d, %f, %s, etc.
end
fclose(s1);
```

```
delete(s1);
clear s1
catch exception
    disp('---> Something is wrong')
    fclose(s1);                 % always, always want to close s1
    delete(s1);
    clear s1
    throw(exception);
end

%remove the zeros from t and v (due to allocation at the begining)
index=find(0==t);
t=t(1:index(1)-1);
v1=v1(1:index(1)-1);
v2=v2(1:index(1)-1);

to=linspace(t(1),t(end),floor((t(end)-t(1))*500));  %down sample to 500Hz
vo1=interp1(t,v1,to);
vo2=interp1(t,v2,to);

figure(1)
plot(t,v1,'bx-',t,v2,'*r-')                  % another interesting graph
xlabel('Time (s)')
ylabel('Sendor Voltage (bits)')
grid on
zoom on
legend('Front Sensor','Back Sensor')
title('Matlab reading Arduino')

SR=mean(t(2:end)-t(1:end-1));
disp('----->')
disp(['---> Mean sample rate is ',num2str(SR),' seconds'])
disp('----->')
disp(['---> Mean sample frequency is ',num2str(1/SR),' Hz'])

figure(1)
filename1=input('Filename to store data?','s');
if isempty(filename1)
    disp('----->')
    disp('----->  No filename given; therefore, NO SAVE')
    disp('----->')
else
    disp('----->')
    disp(['----->  Saving to file:',filename1,'.mat'])
    disp('----->')
    save(filename1,'to','vo1','vo2')
    print('-dpng',strcat(filename1,'.png'));
end%if,isempty
```

B.2. PSD to Time

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% PSD_to_Time_Play_RevC.m
%
% from RevA
%
% RevB is used to create position signal for table
% This position signal is taken directly from the desired acceleration PSD
% by integrating in the frequency domain (divide by wn^2).
% RevB also adds color to the signal by stringing together segments
% developed using a different set of random numbers.
% RevC will be used for simulations, but will include the coloring of RevB.
%
% LRCorr, 15Sep16
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
close all

NN=pwd;

%%%%%%%%%%%%%%%%%%%%%
% Ball and Ground Properties
%%%%%%%%%%%%%%%%%%%%%
R1=1/2*(7/16)*2.54/100;  %radius of impactor (m)

rho=7850;                 %density of steel (kg/m^3)

m2=rho*4/3*pi*R1^3; %mass of impactor (kg)

r=R1;  %radius of ball (m)
I=2/5*m2*r^2;             %mass moment of inertia of ball (kg m^2)

m1=m2*1e4;  %mass of "ground" (kg)

Fd=15;  %frequency of ground system (Hz)
Td=1/Fd;  %period of system (1/Hz)

k1=(2*pi*1/Td)^2*m1; %stiffness of "ground" (N/m)

c1=2*.9*(2*pi*1/Td)*m1;  %damping of "ground" (N/m*s)

g=9.8;   %acceleration due to gravity (m/s^2)
mus=0.7; %static coefficient of friction (steel on steel)
muk=0.6; %kinetic coefficient of friction (steel on steel)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Desired specturm of ground
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
p = nextpow2(1000);
n = 2^p;    % # of points
dt = 1/200; % time step (s)
fs=1/dt;    % sample freq (Hz)
T = n*dt;   % total time
df = 1/n/dt; % frequency step
t = [ 0:n/2 , -n/2+1:-1 ] * dt; % time axis
f = [ 0:n/2 , -n/2+1:-1 ] * df; % freqency axis
```

```
Flo = 1; % low cut-off frequency, Hz
Fhi = 5; % high cut-off frequency, Hz
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% added 1/40 to ensure stroke of table OK
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

So=(mus/4)^2/(I/(m2*r^2+I))^2/(Fhi-Flo)/40;

% So to ensure no slip condition for ball
%   mus/2 - 95.5% sure
%   mus/3 - 99.7% sure
%   mus/4 - 99.9% sure
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Sxx = So*ones(1,n);
Sxx(find(f < -Fhi)) = 0;
Sxx(find(f >  Fhi)) = 0;
Sxx(find(-Flo < f & f < Flo)) = 0;

figure
plot(f,Sxx)
xlabel('Frequency (Hz)')
ylabel('PDS (g^2/Hz)')
grid on
title('Desired Ground PSD')
%%%%%%%%%%%%%%%%%%%%
% EOM
%%%%%%%%%%%%%%%%%%%%
%Ground driven by accel input
Factor=0.01;
A1=[0 1;-k1/(m1) -c1/(m1)];
B1=[0;-9.81*Factor];
C1=[1000 0;0 1;-k1/(m1)/9.8 -c1/(m1)/9.8];
D1=[0;0;-1*Factor];

sys2=ss(A1,B1,C1,D1);
sys2.name='Ground Model';
sys2.outputname{1}='Ground Displacement (mm)';
sys2.outputname{2}='Ground Velocity (m/s)';
sys2.outputname{3}='Ground Accleration (g)';
sys2.inputname='Filtered Excitation';
%%%%%%%%%%%%%%%%%%%%
% Obtain input PSD
%%%%%%%%%%%%%%%%%%%%
H2=abs(squeeze(freqresp(sys2(3,1),f(1:n/2+1),'Hz'))).^2;

%one sided
```

```
PSDu=Sxx(1:n/2+1)./H2';

figure
plot(f,Sxx,'b',f(1:n/2+1),H2'.*PSDu,'r--')
xlabel('Frequency (Hz)')
ylabel('PDS (g^2/Hz)')
grid on
title('Desired System PSDs')
legend('Desired PSD','Calculated PSD')

%two sided
PSDu=[PSDu PSDu(end-1:-1:2)];

figure
plot(f,Sxx,'b',f(1:n/2+1),Sxx(1:n/2+1)./H2','r')
xlabel('Frequency (Hz)')
ylabel('PDS (g^2/Hz)')
grid on
title('Desired System PSDs')
legend('Output PSD','Input PSD')
%%%%%%%%%%%%%%%%%%%%%
% Obtain input time waveform
%%%%%%%%%%%%%%%%%%%%%
Tstep=n*dt;
Tend=30*Tstep;                  % approximate end of time (sec)
NumSteps=ceil(Tend/Tstep);
Tend=n*dt*NumSteps;        % end of time (sec)
tt=0:dt:Tend-dt;            % time vector


rng('shuffle');

%one sided
PSD2u=Sxx(1:n/2+1)./H2';
theta = 2*pi*rand(1,n/2); % random phase angle
theta(1) = 0; % for real-valued signals
theta(n/2+1) = 0; % for real-valued signals
tt_temp=0:dt:Tstep-dt;
ff=zeros(length(tt_temp),1);
for mm=1:NumSteps
    for kk=1:n/2+1
        ff=ff+sqrt(PSD2u(kk)*df)*cos(2*pi*f(kk)*tt_temp'+theta(kk));
    end%for,kk
    ff=sqrt(2)*ff*1.0225;
    ff_tot(:,mm)=ff;
    ff=zeros(length(tt_temp),1);
    theta = 2*pi*rand(1,n/2); % random phase angle
    theta(1)=0;
    theta(n/2+1) = 0; % for real-valued signals
end%for,mm

close all
plot(ff_tot)
legend('1','2','3','4')

%connect the time series together without adding high freq content
```

```
tt_new=[0:dt:Tstep-dt]';
MaxRate=mean(mean(abs(ff_tot(2:end,:)-ff_tot(1:end-1,:)),1))/2;
ff_new=ff_tot(:,1);
for mm=2:NumSteps
    N=ceil(abs(ff_tot(end,mm-1)-ff_tot(1,mm))/MaxRate);
    Insert_Sig=linspace(ff_tot(end,mm-1),ff_tot(1,mm),N)';
    Insert_Sig=Insert_Sig(2:end-1);
    ff_new=[ff_new;Insert_Sig;ff_tot(:,mm)];
    tt_new=[tt_new;...
            tt_new(end)+dt*[1:length(Insert_Sig)]';...
            tt_new(end)+dt*length(Insert_Sig)+dt+[0:dt:Tstep-dt]'];
end%for,mm

ff=ff_new;
tt=tt_new;

D=designfilt('lowpassiir', 'PassbandFrequency', Fhi, 'StopbandFrequency',
1.5*Fhi, 'PassbandRipple', 1, 'StopbandAttenuation', 60, 'SampleRate',
200);
ff=filter(D,ff);

% obtain PS and PSD from input

% set info
df_desired=(f(2)-f(1));  %desired freq increment (Hz)
df_desired=.1;  %desired freq increment (Hz)
S=ff;           %signal
t=tt;

NFFT=2^nextpow2(fs/df_desired);
if NFFT<256
    NFFT=256;  %minimum fft length
end%if,NFFT
df=fs/NFFT; %actual freq increment
%number of averages
NumAve=floor(length(S)/NFFT)*2-1;
%NumAve=20;       %number of 50% averages
disp('------->')
disp('------->')
disp(['-------> Number of Averages: ',num2str(NumAve)])
disp('------->')
disp('------->')
%cut the time signal into segments
if ~rem(NumAve,2)
    NumCuts=NumAve+2; %NumAve is even
else
    NumCuts=NumAve+1; %NumAve is odd
end%if~rem
%cut the waveform
Len=floor(length(t)/NumCuts);
for mm=1:NumCuts
    Sc(mm,:)=S((mm-1)*Len+1:mm*Len,1);
end%for,mm
%combine waveforms
oo=1;
for mm=1:2:NumAve
```

```
    Scc(oo,:)=[Sc(mm,:),Sc(mm+1,:)];
    oo=oo+1;
end%for,mm
for mm=2:2:NumAve
    Scc(oo,:)=[Sc(mm,:),Sc(mm+1,:)];
    oo=oo+1;
end%for,mm
if size(Scc,2)>NFFT
    disp('----->')
    disp('-----> Warning: time signal will be truncated')
    disp('----->')
elseif size(Scc,2)<NFFT
    disp('----->')
    disp('-----> Warning: time signal will be zero padded')
    disp('----->')
end%if,size(Scc,2)
%apply window
w=hann(size(Scc,2))';
for mm=1:size(Scc,1)
    Scc(mm,:)=w.*Scc(mm,:);
end%for,mm
%obtain PSD
fuh=[0:NFFT/2]*df;                  % one-sided
                                  % double all freqs except 0 and Nyquist
                                  % Zero frequency (DC) and the Nyquist
frequency do not occur twice

Xx=fft(Scc,NFFT,2);
Xx=Xx(:,1:NFFT/2+1);
PSuh=2*abs(Xx).^2/sum(w).^2;
PSDuh=2*abs(Xx).^2/sum(w.^2)/fs;
PSuh=mean(PSuh,1);
LSuh=sqrt(PSuh);
PSDuh=mean(PSDuh,1);

figure
plot(f,PSDu,'b',fuh,PSDuh,'r')
xlabel('Frequency (Hz)')
ylabel('PDS (g^2/Hz)')
grid on
title('Desired Inpt PSD to Ground System')
legend('Desired','Actual')
%%%%%%%%%%%%%%%%%%%%
% Excite system & obtain PSD of Ground Accel
%%%%%%%%%%%%%%%%%%%%
clear Len Sc S Scc

Y=lsim(sys2,ff,tt);

% obtain PS and PSD from output

% set info
df_desired=(f(2)-f(1));  %desired freq increment (Hz)
df_desired=.5;  %desired freq increment (Hz)
S=Y(:,3);           %signal
t=tt;
```

```
NFFT=2^nextpow2(fs/df_desired);
if NFFT<256
    NFFT=256;  %minimum fft length
end%if,NFFT
df=fs/NFFT; %actual freq increment
%number of averages
NumAve=floor(length(S)/NFFT)*2-1;
disp('------->')
disp('------->')
disp(['-------> Number of Averages: ',num2str(NumAve)])
disp('------->')
disp('------->')
%cut the time signal into segments
if ~rem(NumAve,2)
    NumCuts=NumAve+2; %NumAve is even
else
    NumCuts=NumAve+1; %NumAve is odd
end%if~rem
%cut the waveform
Len=floor(length(t)/NumCuts);
for mm=1:NumCuts
    Sc(mm,:)=S((mm-1)*Len+1:mm*Len,1);
end%for,mm
%combine waveforms
oo=1;
for mm=1:2:NumAve
    Scc(oo,:)=[Sc(mm,:),Sc(mm+1,:)];
    oo=oo+1;
end%for,mm
for mm=2:2:NumAve
    Scc(oo,:)=[Sc(mm,:),Sc(mm+1,:)];
    oo=oo+1;
end%for,mm
if size(Scc,2)>NFFT
    disp('----->')
    disp('-----> Warning: time signal will be truncated')
    disp('----->')
elseif size(Scc,2)<NFFT
    disp('----->')
    disp('-----> Warning: time signal will be zero padded')
    disp('----->')
end%if,size(Scc,2)
%apply window
w=hann(size(Scc,2))';
for mm=1:size(Scc,1)
    Scc(mm,:)=w.*Scc(mm,:);
end%for,mm
%obtain PSD
ft=[0:NFFT/2]*df;                   % one-sided
                                   % double all freqs except 0 and Nyquist
                                   % Zero frequency (DC) and the Nyquist
frequency do not occur twice

Xx=fft(Scc,NFFT,2);
Xx=Xx(:,1:NFFT/2+1);
```

```
PSt=2*abs(Xx).^2/sum(w).^2;
PSDt=2*abs(Xx).^2/sum(w.^2)/fs;
PSt=mean(PSt,1);
LSt=sqrt(PSt);
PSDt=mean(PSDt,1);

figure
plot(f(1:n/2+1),Sxx(1:n/2+1),'b',ft,PSDt,'r')
xlabel('Frequency (Hz)')
ylabel('PDS (g^2/Hz)')
grid on
title('Desired Ground PSD - based on ff')
legend('Desired','Actual')


disp('---> Variance of Ground Accel (Desired)')
P=So*(Fhi-Flo);
disp(sprintf('%10.4f\r',P))

disp('---> Variance of Ground Accel  (time domain)')
P=cov(Y(:,3));
disp(sprintf('%10.4f\r',P))

disp('---> Variance of Ground Accel  (freq domain)')
P=trapz(ft,PSDt);
disp(sprintf('%10.4f\r',P))

disp('---> % Difference (Desired / freq domain)')
P=(So*(Fhi-Flo)-P)/(So*(Fhi-Flo))*100;
disp(sprintf('%10.4f\r',P))


figure
histogram(Y(:,1))
xlabel('Response')
ylabel('Count')
title(sys2.outputname{1})
grid on
hold on
Ax=axis;
h=plot(-sqrt(cov(Y(:,1))*[1 1]),Ax(3:4),'g',sqrt(cov(Y(:,1))*[1
1]),Ax(3:4),'g');
set(h,'linewidth',3)
drawnow
axPos = get(gca,'Position');
axX=axis;
xstart=axPos(1);
ystart=axPos(2);
xend=axPos(1)+axPos(3);
yend=axPos(2)+axPos(4);
xtemp1=(xend-xstart)/(axX(2)-axX(1))*(-sqrt(cov(Y(:,1)))-axX(1))+xstart;
xtemp2=(xend-xstart)/(axX(2)-axX(1))*(sqrt(cov(Y(:,1)))-axX(1))+xstart;
ytemp=(yend-ystart)/(axX(4)-axX(3))*(Ax(4)*0.5-axX(3))+ystart;
h=annotation('doublearrow',[xtemp1 xtemp2],[ytemp ytemp]);
set(h,'color','g');
set(h,'linewidth',3);
```

```
h=text(0,(Ax(4)*0.6),'2\sigma');
set(h,'color','g');
h.FontSize=18;
h.FontWeight='bold';
h.HorizontalAlignment='center';

figure
histogram(Y(:,2))
xlabel('Response')
ylabel('Count')
title(sys2.outputname{2})
grid on
hold on
Ax=axis;
h=plot(-sqrt(cov(Y(:,2))*[1 1]),Ax(3:4),'g',sqrt(cov(Y(:,2))*[1
1]),Ax(3:4),'g');
set(h,'linewidth',3)
drawnow
axPos = get(gca,'Position');
axX=axis;
xstart=axPos(1);
ystart=axPos(2);
xend=axPos(1)+axPos(3);
yend=axPos(2)+axPos(4);
xtemp1=(xend-xstart)/(axX(2)-axX(1))*(-sqrt(cov(Y(:,2)))-axX(1))+xstart;
xtemp2=(xend-xstart)/(axX(2)-axX(1))*(sqrt(cov(Y(:,2)))-axX(1))+xstart;
ytemp=(yend-ystart)/(axX(4)-axX(3))*(Ax(4)*0.5-axX(3))+ystart;
h=annotation('doublearrow',[xtemp1 xtemp2],[ytemp ytemp]);
set(h,'color','g');
set(h,'linewidth',3);
h=text(0,(Ax(4)*0.6),'2\sigma');
set(h,'color','g');
h.FontSize=18;
h.FontWeight='bold';
h.HorizontalAlignment='center';

figure
histogram(Y(:,3))
xlabel('Response')
ylabel('Count')
title(sys2.outputname{3})
grid on
hold on
Ax=axis;
h=plot(-sqrt(cov(Y(:,3))*[1 1]),Ax(3:4),'g',sqrt(cov(Y(:,3))*[1
1]),Ax(3:4),'g');
set(h,'linewidth',3)
drawnow
axPos = get(gca,'Position');
axX=axis;
xstart=axPos(1);
ystart=axPos(2);
xend=axPos(1)+axPos(3);
yend=axPos(2)+axPos(4);
xtemp1=(xend-xstart)/(axX(2)-axX(1))*(-sqrt(cov(Y(:,3)))-axX(1))+xstart;
xtemp2=(xend-xstart)/(axX(2)-axX(1))*(sqrt(cov(Y(:,3)))-axX(1))+xstart;
```

```
ytemp=(yend-ystart)/(axX(4)-axX(3))*(Ax(4)*0.5-axX(3))+ystart;
h=annotation('doublearrow',[xtemp1 xtemp2],[ytemp ytemp]);
set(h,'color','g');
set(h,'linewidth',3);
h=text(0,(Ax(4)*0.6),'2\sigma');
set(h,'color','g');
h.FontSize=18;
h.FontWeight='bold';
h.HorizontalAlignment='center';

Time=tt;
Input_Time=ff;
Freq=f(1:n/2+1);
Input_Freq=Sxx(1:n/2+1);
sysGround=sys2;
%%%%%%%%%%%%%%%%%%%%

filename=strcat('Input_',num2str(Flo),'_to_',num2str(Fhi),'Hz_',num2str(rou
nd(Tend/60)),'min.mat');
%filename=strcat('TableTest_',num2str(Flo),'_to_',num2str(Fhi),'Hz_',num2st
r(round(Tend)),'sec_RevA.mat');
save(filename,'sysGround','Time','Input_Time','Freq','Input_Freq','Flo','Fh
i','So','Y')

disp(['---> Saved to filename = ',filename])

%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%
% Excite system & obtain PSD of Ground Displacement
%%%%%%%%%%%%%%%%%%%%
% obtain PS and PSD from output

% set info
df_desired=(f(2)-f(1));  %desired freq increment (Hz)
df_desired=.5;  %desired freq increment (Hz)
S=Y(:,1);           %signal
t=tt;

NFFT=2^nextpow2(fs/df_desired);
if NFFT<256
    NFFT=256;  %minimum fft length
end%if,NFFT
df=fs/NFFT; %actual freq increment
%number of averages
NumAve=floor(length(S)/NFFT)*2-1;
disp('------->')
disp('------->')
disp(['-------> Number of Averages: ',num2str(NumAve)])
disp('------->')
disp('------->')
%cut the time signal into segments
if ~rem(NumAve,2)
    NumCuts=NumAve+2; %NumAve is even
else
    NumCuts=NumAve+1; %NumAve is odd
```

```
end%if~rem
%cut the waveform
Len=floor(length(t)/NumCuts);
for mm=1:NumCuts
    Sc(mm,:)=S((mm-1)*Len+1:mm*Len,1);
end%for,mm
%combine waveforms
oo=1;
for mm=1:2:NumAve
    Scc(oo,:)=[Sc(mm,:),Sc(mm+1,:)];
    oo=oo+1;
end%for,mm
for mm=2:2:NumAve
    Scc(oo,:)=[Sc(mm,:),Sc(mm+1,:)];
    oo=oo+1;
end%for,mm
if size(Scc,2)>NFFT
    disp('----->')
    disp('-----> Warning: time signal will be truncated')
    disp('----->')
elseif size(Scc,2)<NFFT
    disp('----->')
    disp('-----> Warning: time signal will be zero padded')
    disp('----->')
end%if,size(Scc,2)
%apply window
w=hann(size(Scc,2))';
for mm=1:size(Scc,1)
    Scc(mm,:)=w.*Scc(mm,:);
end%for,mm
%obtain PSD
ft=[0:NFFT/2]*df;                   % one-sided
                                   % double all freqs except 0 and Nyquist
                                   % Zero frequency (DC) and the Nyquist
frequency do not occur twice

Xx=fft(Scc,NFFT,2);
Xx=Xx(:,1:NFFT/2+1);
PSt=2*abs(Xx).^2/sum(w).^2;
PSDt=2*abs(Xx).^2/sum(w.^2)/fs;
PSt=mean(PSt,1);
LSt=sqrt(PSt);
PSDt=mean(PSDt,1);

Fin=fuh(find(1<fuh));
PSDin=PSDuh(find(1<fuh));
HS=abs(squeeze(freqresp(sys2(1,1),Fin,'Hz'))).^2;

figure
plot(Fin,HS'.*PSDin,'bx-',ft,PSDt,'ro-')
xlabel('Frequency (Hz)')
ylabel('PDS (mm^2/Hz)')
grid on
title('Ball Displacement PSD')
legend('Predicted','Actual')
```

```
disp('---> Standard Deviation of Ground Displacement (mm) (Expected)')
P=sqrt(trapz(Fin,HS'.*PSDin));
disp(sprintf('%10.4f\r',P))

disp('---> Standard Deviation of Ground Displacement (mm)  (time domain)')
P=sqrt(cov(Y(:,1)));
disp(sprintf('%10.4f\r',P))

disp('---> Standard Deviation of Ground Displacement (mm)  (freq domain)')
P=sqrt(trapz(ft,PSDt));
disp(sprintf('%10.4f\r',P))
```

B.3. Time to ACRView

```
%%%%%%%%%%%%%%%%%%%%%
% CreateTablePosition_RevA.m
%
% create program for Aries Controller (ACRView).
%
% LRCorr, 26May16
%%%%%%%%%%%%%%%%%%%%%
clear all
close all

NN=pwd;
%%%%%%%%%%%%%%%%%%%%%
% load in data
%%%%%%%%%%%%%%%%%%%%%
cd('C:\Users\Larry\Documents\MS_Degree\Thesis\MATLAB_ThesisWork')
filename='TableTest_1_to_5Hz_154sec_RevA';
load(strcat(filename,'.mat'))
cd(NN)
%%%%%%%%%%%%%%%%%%%%%
% plot data
%%%%%%%%%%%%%%%%%%%%%
figure(1)
plot(Time,Y(:,1),'bx-')
xlabel('Time (s)')
ylabel('Position (mm)')
grid on
zoom on

%resample data for a sample rate of Td s (50Hz)
index=find(130<Time);  %start with 1 second of data
Td=0.03; %sample rate in seconds (** BEST WE CAN DO WITH TABLE - 0.03 sec
To=linspace(Time(1),Time(index(1)),(Time(index(1))-Time(1))/Td);
Po=interp1(Time,Y(:,1),To);  %ground motion in (mm)

%%%%%%%%%%%%%%%%%%%%%
% don't allow the table velocity to go any faster than 200 mm/s
% this is just set due to observation with the table - no science here
% should not impact signal too much (in the noise...)
%%%%%%%%%%%%%%%%%%%%%
SpeedLimit=200;  %table speed limit (mm/s)
PPo=Po(1);
TTo=To(1);
Num_Changed=0;
for mm=2:length(Po)
    if abs(Po(mm)-Po(mm-1))/Td>SpeedLimit
        PPo=[PPo (Po(mm)+Po(mm-1))/2 Po(mm)];
        TTo=[TTo TTo(end)+Td TTo(end)+2*Td];
        [Po(mm-1) (Po(mm)+Po(mm-1))/2 Po(mm)];
        Num_Changed=Num_Changed+1;
    else
        PPo=[PPo Po(mm)];
        TTo=[TTo TTo(end)+Td];
    end%if,abs
end%for,mm
```

```
Po=PPo;
To=TTo;
clear PPo TTo

%do it again
PPo=Po(1);
TTo=To(1);
for mm=2:length(Po)
    if abs(Po(mm)-Po(mm-1))/Td>SpeedLimit
        PPo=[PPo (Po(mm)+Po(mm-1))/2 Po(mm)];
        TTo=[TTo TTo(end)+Td TTo(end)+2*Td];
        [PPo(mm-1) (Po(mm)+Po(mm-1))/2 Po(mm)];
        Num_Changed=Num_Changed+1;
    else
        PPo=[PPo Po(mm)];
        TTo=[TTo TTo(end)+Td];
    end%if,abs
end%for,mm
disp(['Number of points added = ',num2str(Num_Changed)])

Po=PPo;
To=TTo;

%don't allow Po to be larger than +/- 20 mm - table length (+/- 24 mm)
%this will add in some high freqs, but is better for system
disp(['Number of points limited > 0 = ',num2str(length(Po(find(Po>20))))])
disp(['Number of points limited < 0 = ',num2str(length(Po(find(Po>20))))])
Po(find(Po>20))=20;
Po(find(Po<-20))=-20;

% since in a loop, make sure the last Po is close to the first Po
index=find(abs(Po)<=abs(Po(1))+0.5);
Po=Po(1:index(end));
To=To(1:index(end));


figure(2)
plot(Time,Y(:,1),'bx-',To,Po,'ro-')
xlabel('Time (s)')
ylabel('Position (mm)')
grid on
zoom on

if To(end)~= Time(end)
    index1=strfind(filename,'_Rev');
    index2=strfind(filename(index1-1:-1:1),'_');
    index2=index2(1);
    filename=strcat(filename(1:index1-
index2),num2str(round(To(end))),'sec',filename(index1:end));
end%if,To(end)
%%%%%%%%%%%%%%%%%%%%%%
% write code for controller
%%%%%%%%%%%%%%%%%%%%%%
[fileID,errmsg]=fopen(strcat(filename,'.txt'),'w');
fprintf(fileID,'%s\r\n','PROGRAM');
```

```
fprintf(fileID,'%s\r\n',strjoin({'''PROGRAM 0 - Data Sample Rate =
',num2str(Td)}));
fprintf(fileID,'%s\r\n',strjoin({'''Position Data from
',filename,'.mat'}));
fprintf(fileID,'%s\r\n',strjoin({'''Data file created on',date}));
fprintf(fileID,'%s\r\n','''REM %%%%%%%%%%%%%%%%%%%%');
fprintf(fileID,'%s\r\n','''REM - Center the Table');
fprintf(fileID,'%s\r\n','''REM %%%%%%%%%%%%%%%%%%%%');
fprintf(fileID,'%s\r\n','''Set PID Gains for centering the table');
fprintf(fileID,'%s\r\n','PGAIN AXIS0 0.002441400');
fprintf(fileID,'%s\r\n','IGAIN AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','ILIMIT AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','IDELAY AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','DGAIN AXIS0 0.000010000');
fprintf(fileID,'%s\r\n','DWIDTH AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','FFVEL AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','FFACC AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','TLM AXIS0 10.000000000');
fprintf(fileID,'%s\r\n','FBVEL AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','''Set JOG Speed');
fprintf(fileID,'%s\r\n','jog acc x 50');
fprintf(fileID,'%s\r\n','jog dec x 50');
fprintf(fileID,'%s\r\n','jog vel x 5');
fprintf(fileID,'%s\r\n','''Set MOV Speed');
fprintf(fileID,'%s\r\n','tmov off');
fprintf(fileID,'%s\r\n','acc 50');
fprintf(fileID,'%s\r\n','dec 50');
fprintf(fileID,'%s\r\n','stp 50');
fprintf(fileID,'%s\r\n','vel 5');
fprintf(fileID,'%s\r\n','''Engage the Drive');
fprintf(fileID,'%s\r\n','drive on x');
fprintf(fileID,'%s\r\n','''JOG Home');
fprintf(fileID,'%s\r\n','jog home x-1');
fprintf(fileID,'%s\r\n','print "jogging home"');
fprintf(fileID,'%s\r\n','print "Waiting for jog (bit 792) to clear"');
fprintf(fileID,'%s\r\n','INH -792');
fprintf(fileID,'%s\r\n','print "Bit 792 is clear"');
fprintf(fileID,'%s\r\n','print "waiting 10 seconds"');
fprintf(fileID,'%s\r\n','dwl 10');
fprintf(fileID,'%s\r\n','''Center Table');
fprintf(fileID,'%s\r\n','x-11.695');
fprintf(fileID,'%s\r\n','print "Waiting for mov (bit 516) to clear"');
fprintf(fileID,'%s\r\n','INH -516');
fprintf(fileID,'%s\r\n','print "Bit 516 is clear"');
fprintf(fileID,'%s\r\n','''Reset Position');
fprintf(fileID,'%s\r\n','res x');
fprintf(fileID,'%s\r\n','ren x');
fprintf(fileID,'%s\r\n','print "waiting 10 seconds for reset of
position"');
fprintf(fileID,'%s\r\n','dwl 10');
fprintf(fileID,'%s\r\n','print "table is centered"');
fprintf(fileID,'%s %9.7f\r\n','X',Po(1));
fprintf(fileID,'%s\r\n','print "Waiting for mov (bit 516) to clear"');
fprintf(fileID,'%s\r\n','INH -516');
fprintf(fileID,'%s\r\n','print "Bit 516 is clear"');
fprintf(fileID,'%s\r\n','''Turn on MBUF (move buffer)');
```

```
fprintf(fileID,'%s\r\n','clear');
fprintf(fileID,'%s\r\n','dim mbuf (50)');
fprintf(fileID,'%s\r\n','mbuf on');
fprintf(fileID,'%s\r\n','dwl 1');
fprintf(fileID,'%s\r\n','''REM %%%%%%%%%%%%%%%%%%%%');
fprintf(fileID,'%s\r\n','''REM - Move Table per Data');
fprintf(fileID,'%s\r\n','''REM %%%%%%%%%%%%%%%%%%%%');
fprintf(fileID,'%s\r\n','print "Executing Position"');
fprintf(fileID,'%s\r\n','''Turn on time moves');
fprintf(fileID,'%s\r\n','tmov on');
fprintf(fileID,'%s\r\n',strjoin({'tmov ',num2str(Td)}));
%
fprintf(fileID,'%s\r\n','''Set MOV Speed limits for system');
fprintf(fileID,'%s\r\n','acc 49000');
fprintf(fileID,'%s\r\n','dec 49000');
fprintf(fileID,'%s\r\n','stp 0');
%
fprintf(fileID,'%s\r\n','vel 3000');
%
fprintf(fileID,'%s\r\n','''Set PID Gains for moving the table');
fprintf(fileID,'%s\r\n','PGAIN AXIS0 0.02');
fprintf(fileID,'%s\r\n','DGAIN AXIS0 0.00001');
fprintf(fileID,'%s\r\n','IGAIN AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','ILIMIT AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','IDELAY AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','DWIDTH AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','FFVEL AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','FFACC AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','TLM AXIS0 10.000000000');
fprintf(fileID,'%s\r\n','FBVEL AXIS0 0.000000000');
%%%%%%%%%%%%%%%%%%%%
%for testing only
%%%%%%%%%%%%%%%%%%%%
fprintf(fileID,'%s\r\n','set bit128');
fprintf(fileID,'%s\r\n','_LOOP');
%%%%%%%%%%%%%%%%%%%%
fprintf(fileID,'%s\r\n','''Define positions for table');
for mm=1:length(Po)
    fprintf(fileID,'%s %9.7f\r\n','X',Po(mm));
end%for,mm
%%%%%%%%%%%%%%%%%%%%
%for testing only
%%%%%%%%%%%%%%%%%%%%
%fprintf(fileID,'%s\r\n','GOTO LOOP');
fprintf(fileID,'%s\r\n','if (bit 128) then GOTO LOOP');
%%%%%%%%%%%%%%%%%%%%
fprintf(fileID,'%s\r\n','''Turn off MBUF (move buffer)');
fprintf(fileID,'%s\r\n','mbuf off');
fprintf(fileID,'%s\r\n','print "Finish with Move"');
fprintf(fileID,'%s\r\n','''Set PID Gains for centering the table');
fprintf(fileID,'%s\r\n','PGAIN AXIS0 0.002441400');
fprintf(fileID,'%s\r\n','IGAIN AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','ILIMIT AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','IDELAY AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','DGAIN AXIS0 0.000010000');
fprintf(fileID,'%s\r\n','DWIDTH AXIS0 0.000000000');
```

```
fprintf(fileID,'%s\r\n','FFVEL AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','FFACC AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','TLM AXIS0 10.000000000');
fprintf(fileID,'%s\r\n','FBVEL AXIS0 0.000000000');
fprintf(fileID,'%s\r\n','''Set MOV Speed');
fprintf(fileID,'%s\r\n','tmov off');
fprintf(fileID,'%s\r\n','acc 50');
fprintf(fileID,'%s\r\n','dec 50');
fprintf(fileID,'%s\r\n','stp 50');
fprintf(fileID,'%s\r\n','vel 5');
fprintf(fileID,'%s\r\n','print "waiting 10 seconds"');
fprintf(fileID,'%s\r\n','dwl 10');
fprintf(fileID,'%s\r\n','print "centering table"');
fprintf(fileID,'%s\r\n','x 0');
fprintf(fileID,'%s\r\n','print "Waiting for mov (bit 516) to clear"');
fprintf(fileID,'%s\r\n','INH -516');
fprintf(fileID,'%s\r\n','print "Bit 516 is clear"');
fprintf(fileID,'%s\r\n','print "table is centered"');
fprintf(fileID,'%s\r\n','''Disengage the Drive');
fprintf(fileID,'%s\r\n','drive off x');
fprintf(fileID,'%s','ENDP');
fclose(fileID);

disp(['---> Saved to filename = ',filename])
```

References

Amditis, A. (2010). *Structural Health Monitoring (SHM) today. The MEMSCON project approach* [PowerPoint slides]. Retrieved from http://www.memscon.com/content/content/files/02.%20AMDITIS%20ANGELOS%20MEMSCON%20project_.ppt.

Beeby, S. P., Wang, L., Zhu, D., Weddell, A. S., Merrett, G. V., Stark, B., Szarka, G. & Al-Hashimi, B. M. (2013). A comparison of power output from linear and nonlinear kinetic energy harvesters using real vibration data. *Smart Materials and Structures, 22*(7), Article ID 075022, 15 pages

Brownjohn, J. M. W. (2007). Structural health monitoring for civil infrastructure. Philosophical Transactions of the Royal Society, 365, 589-622. doi:10.1098/rsta.2006.1925.

Castaneda, N. E. *Structural Health Monitoring: from algorithms to implementations* [PowerPoint slides]. Retrieved from ftp://ftp.ecn.purdue.edu/spujol/CE575/Presentations/SHM_PRESENTATION%20(Nestor%20Castaneda).ppt.

Chang, P. C., Flatau, A. & Liu, S. C. (2003). Review paper: health monitoring of civil infrastructure. *Structural Health Monitoring, 2*(3), 257-267. doi: 10.1177/1475921703036169.

Chatterjee, A. (1997). Rigid body collisions: some general considerations, new collision laws, and some experimental data. Ithaca, NY: Cornell University.

Clark, R., Saunders, W. & Gibbs, G., (1988). Adaptive Structures, 1st Ed, John Wiley & Sons, Inc., New York, pp. 163-170.

Crandall, S.H., Karnopp, D.C., Kurtz, E.F. & Pridmore-Brown, D.C. (1968). Dynamics of mechanical and electromechanical systems. Robert E. Krieger Publishing Co., Malabar Florida.

Cross, R. (1999a). The bounce of a ball. American Journal of Physics, 67(3), 222-227.

Cross, R. (1999b). Impact of a ball with a bat or racket. American Journal of Physics, 67(8), 692-702.

Cui, X. & Hu, J. (2014). An impact-type vibration energy harvester using the pendulum and its Multiphysics finite element analysis. *21st International Congress on Sound and Vibration (ICSV21)*. Beijing, China.

Davidson, J. & Mo, C. (2014). Recent advances in energy harvesting technologies for structural health monitoring applications. *Smart Materials Research, 2014*. doi: 10.1155/2014/410316

Davis, R. H., Serayssol, J. M. & Hinch, E. J. (1986). The elastohydrodynamic collision of two spheres. Journal of Fluid Mechanics, 163, 479-497.

Doebling, S.W., Farrar, C. R. & Prime, M.B. (1998). A summary review of vibration-based damage identification methods. *The Shock and Vibration Digest, 30*(2), 91–105.

Doebling, S. W., Farrar, C. R., Prime, M. B. & Shevitz, D. W. (1996). Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: a literature review. *Los Alamos National Laboratory report, LA-13070-MS*.

Dosch, J. J., Inman, D. J. & Garcia, E. (1992). A self-sensing piezoelectric actuator for collocated control. Journal of Intelligent Material, Systems, and Structures, Vol. 3, January 3, 1992, pp. 166 – 185.

Eringen, A. C. (1952). Transverse impact on beams and plates with arbitrary edge conditions. Office of Naval Research, Department of the Navy, Technical Report No. 2 (AD No. AD008174).

Faik, S. & Witteman, H. (2000). Modeling of impact dynamics: a literature survey. 2000 International ADAMS User Conference.

Fanson, J.L. & Caughey, T.K. (1987). Positive position feedback control for large space structures. Proceedings 28th AIAA/ASME/ASC/AHS Structures and Structural Dynamics and Materials Conference, Monteray, California, pp. 588 – 598.

Farrar, C. R. & Worden, K. (2007). An introduction to structural health monitoring. *Philosophical Transactions of the Royal Society, 365*, 303-315. doi:10.1098/rsta.2006.1928.

Galchev, T. V., McCullagh, J., Peterson, R. L. & Najafi, K. (2011). Harvesting traffic-induced vibrations for structural health monitoring of bridges. *Journal of Micromechanics and Microengineering, 21*(10), 104005. doi: 10.1088/0960-1317/21/10/104005

Godinez-Azcuaga, V. & Ley, O. (2014). New strategies for SHM based on a multichannel wireless AE node. *Proc. SPIE 9063, Nondestructive Characterization for Composite Materials, Aerospace Engineering, Civil Infrastructure, and Homeland Security 2014, 9063*, 906302-1-8. doi:10.1117/12.2045343

Gopinathan, M., Pajunen, G.A., Neelakanta, P.S. & Arockaisamy, M. (1995). Recursive estimation of displacement and velocity in a cantilever beam using a measured set of distributed strain data. Journal of Intelligent Material Systems and Structures, Vol. 6, July 1995, pp. 537 – 549.

Grady, J. E. (1988). Contact force history and dynamic response due to the impact of a soft projectile. NASA Technical Memorandum 100961.

Gu, L. & Livermore, C. (2011). Impact-driven, frequency up-converting coupled vibration energy harvesting device for low frequency operation. *Smart Materials and Structures, 20*(4), 045004. doi: 10.1088/0964-1726/20/4/045004

Hagood, N. W., Chung, W. H. & von Flotow, A. (1990). Modelling of piezoelectric actuator dynamics for active structural control. Proceedings of the AIAA/ASME/ASCE/AHS/ASC 31st Structures, Structural Dynamics and Materials Conference, Long Beach, CA, AIAA-90-1097-CP, April 2-4 1990, pp. 2242-2256.

Hagood, N. W. & von Flotow, A. (1991). Damping of structural vibrations with piezoelectric materials and passive electric networks. Journal of Sound and Vibration, vol. 146(2), 1991, pp. 243-268.

Hackmann, G., Guo, W., Yan, G., Lu, C. & Dyke, S. (2010). Cyber-physical codesign of distributed structural health monitoring with wireless sensor networks. in: ICCPS 2010, 2010.

Halim, M. A., Khym, S. & Park, J. Y. (2013). Frequency up-converted wide bandwidth piezoelectric energy harvester using mechanical impact. *Journal of Applied Physics, 114*(4), 044902. doi: 10.1063/1.4816249

Halim, M. A. & Park, J. Y. (2015). Piezoceramic based wideband energy harvester using impact-enhanced dynamic magnifier for low frequency vibration. *Ceramics International, 41.* S702–S707. http://dx.doi.org/10.1016/j.ceramint.2015.03.143

Haroun, A. & Yamada, I. (2015). Study of electromagnetic vibration energy harvesting with free/impact motion for low frequency operation. *Journal of Sound and Vibration, 349*, 389-402. http://dx.doi.org/10.1016/j.jsv.2015.03.048

He, A. & Wettlaufer, J. S. (2014). Hertz beyond belief. Soft matter, 10(13), 2264-2269.

Heinzel, G., Rüdiger, A. & Schilling, R. (2002). Spectrum and spectral density estimation by the Discrete Fourier transform (DFT), including a comprehensive list of window functions and some new at-top windows.

Hu, Y., Rieutort-Louis, W. S. A., Sanz-Robinson, J., Huang, L., Glisic, B., Strum, J. C., Wagner, S. & Verma, N. (2014). Large-scale sensing system combining large-area electronics and CMOS ICs for structural-health. *IEEE Journal of Solid-State Circuits, 49*(2), 513-523.

Hutzler, S., Delaney, G., Weaire, D. & MacLeod, F. (2004). Rocking Newton's cradle. American Journal of Physics, 72(12), 1508-1516.

Ikeda, T. (1990). Fundamentals of piezoelectricity, Oxford University Press.

Jiang, X., Wang, J., Li, Y., Li, J. & Yao, J. (2014). Energy harvesting for powering wireless sensor networks in low-frequency and large-force environments. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 299*(11), 1953–1964. doi: 10.1177/0954406214551038

Johnson, K. L. (1995). Contact mechanics. Cambridge University Press, Cambridge.

Kim, S., Pakzad, S., Culler, D., Demmel, J., Fenves, G., Glaser, S. & Turon, M. (2007). Health monitoring of civil infrastructures using wireless sensor networks. In IPSN, 2007.

Knief, R. A. (2008). Nuclear engineering: theory and technology of commercial nuclear power. 2nd Ed, American Nuclear Society, Inc., pp. 102-105.

Külah, H. & Najafi, K. (2004). An electromagnetic micro power generator for low-frequency environmental vibrations. *In Micro Electro Mechanical Systems, 2004. 17th IEEE International Conference on MEMS,* 237-240.

Külah, H. & Najafi, K. (2008). Energy scavenging from low-frequency vibrations by using frequency up-conversion for wireless sensor applications. *Sensors Journal, IEEE, 8*(3), 261-268. doi: 10.1109/JSEN.2008.917125

Lamarsh, J. R. (1983). Introduction to nuclear engineering. 2nd Ed, Addison-Wesley Publishing Company, Inc., pp. 57-61.

Liu, H., Lee, C., Kobayashi, T., Tay, C. J. & Quan, C. (2012). Piezoelectric MEMS-based wideband energy harvesting systems using a frequency-up-conversion cantilever stopper. *Sensors and Actuators A: Physical, 186*, 242-248. doi:10.1016/j.sna.2012.01.033

Maurya, D., Yan, Y. & Priya, S. (2015). 5 Piezoelectric Materials for Energy Harvesting. *Advanced Materials for Clean Energy, 143*, 144-174.

Mitcheson, P. D., Yeatman, E. M., Rao, G. K., Holmes, A. S. & Green, T. C. (2008). Energy harvesting from human and machine motion for wireless electronic devices. *Proceedings of the IEEE, 96*(9), 1457-1486. doi: 10.1109/JPROC.2008.927494

Nagurka, M. & Huang, S. (2004). A mass-spring-damper model of a bouncing ball. In American Control Conference, 2004. Proceedings of the 2004 (Vol. 1, pp. 499-504). IEEE.

Park, H. S., Shin, Y., Choi, S. W. & Kim, Y. (2013). An integrative structural health monitoring system for the local/global responses of a large-scale irregular building under construction. *Sensors, 13*, 9085-9103. doi:10.3390/s130709085.

Park, G., Rosing, T., Todd, M. D., Farrar, C. R. & Hodgkiss, W. (2008). Energy Harvesting for Structural Health Monitoring Sensor Networks. *ASCE Journal of Infrastructure Systems, 14*(1), 64-79.

Patricio, P. (2004). The Hertz contact in chain elastic collisions. American journal of physics, 72(12), 1488-1491.

Petyt, M. (1990). Introduction to finite element vibration analysis. 1st Ed, New York: the Press Syndicate of the University of Cambridge.

Popov, V. (2010). Contact mechanics and friction: physical principles and applications. Springer Science & Business Media. doi: 10.1007/978-3-642-10803-7.

Rao, S. S. (1990). Mechanical Vibrations. 2nd Ed, Addison-Wesley Publishing Company.

Roundy, S., Wright, P. K. & Rabaey, J. (2003). A study of low level vibrations as a power source for wireless sensor nodes. *Computer communications, 26*(11), 1131-1144.

Roundy, S., Leland, E. S., Baker, J., Carleton, E., Reilly, E., Lai, E. & Sundararajan, V. (2005). Improving power output for vibration-based energy scavengers. *Pervasive Computing, IEEE, 4*(1), 28-36.

Sears, F. W., Zemansky, M. W. & Young, H. D. (1987). University Physics. 7th Ed, Addison-Wesley Publishing Company, Inc., pp. 192-196.

Shinozuka, M. (1972). Monte Carlo solution of structural dynamics. *Computers & Structures*, *2*(5-6), 855-874.

Shukla, R. & Bell, A. J. (2015). PENDEXE: A novel energy harvesting concept for low frequency human waistline. *Sensors and Actuators A: Physical, 222*, 39-47. http://dx.doi.org/10.1016/j.sna.2014.11.016

Sohn, H., Farrar, C. R., Hemez, F. M., Shunk, D. D., Stinemates, D. W., Nadler, B. R. & Czarnecki, J. J. (2004). A Review of Structural Health Monitoring Literature: 1996 – 2001. *Los Alamos National Laboratory report, LA-13976-MS*.

Sun, C. T. & Yang, S. H. (1980). Contact law and impact responses of laminated composites. NASA Technical Memorandum CR 165460 (CLM-80-1).

Tamas, N. (2012). *ISIS and SAMCO Educational Module 5: An Introduction to Structural Health Monitoring* [PDF document]. Retrieved from http://www.ct.upt.ro/users/TamasNagyGyorgy/Tehnici_Experimentale/3_SHM__Nagy-Gyorgy_T_2012_12_03.pdf

Trigona, C., Baglio, S., Ando, A., Ferrari, M., Ferrari, V. & Guizzetti, M. (2010). Improved Energy Harvesting from Wideband Vibrations by Nonlinear Piezoelectric Converters. *Sensors and Actuators A, 162*(2), pp.425-431. doi:10.1016/j.sna.2010.05.022

Umeda, M., Nakamura, K. & Ueha, S. (1996). Analysis of the transformation of mechanical impact energy to electric energy using piezoelectric vibrator. *Japanese Journal of Applied Physics, 35*(5S), 3267.

Van de Wouw, N., de Kraker, A., Van Campen, D. H. & Nijmeijer, H. (2003). Non-linear dynamics of a stochastically excited beam system with impact. International journal of non-linear mechanics, 38(5), 767-779.

Vijayan, K., Friswell, M. I., Khodaparast, H. H. & Adhikari, S. (2015). Non-linear energy harvesting from coupled impacting beams. *International Journal of Mechanical Sciences, 96*, 101-109. http://dx.doi.org/10.1016/j.ijmecsci.2015.03.001

Waanders, J. W. (1991). Piezoelectric ceramics: properties and applications. Philips Components, Marketing Communications.

Weaver, W. Jr., Timoshenko, S. P. & Young, D. H. (1990). Vibration problems in engineering. 5th Ed, New York: John Wiley & Sons, Inc, pp. 511-531.

Xiao, H. & Wang, X. (2014). A review of piezoelectric vibration energy harvesting techniques. *International Review of Mechanical Engineering, 8*(3), 609-620.

Yang, S. H. & Sun, C. T. (1981). Indentation law for composite laminates. NASA Technical Memorandum 159884 (CLM-81-1).

Zhang, R., Jiang, B. & Cao, W. (2001). Elastic, piezoelectric, and dielectric properties of multidomain 0.67 Pb (Mg1/3Nb2/3) O3-0.33 PbTiO3 single crystals. Journal of Applied Physics, 90(7), 3471-3475. doi: 10.1063/1.1390494.

Zhou, G. & Wu, J. (2015). Unifying energy harvesting, sensing, and communication for ultra-low power structure health monitoring. *EURASIP Journal on Wireless Communications and Networking, 2015*(1), 1-7. doi: 10.1186/s13638-015-0342-1

Zorlu, Ö., Topal, E. T. & Külah, H. (2011). A vibration-based electromagnetic energy harvester using mechanical frequency up-conversion method. Sensors Journal, *IEEE, 11*(2), 481-488. doi: 10.1109/JSEN.2010.2059007

Zorlu, Ö. & Külah, H. (2013). A MEMS-based energy harvester for generating energy from non-resonant environmental vibrations. *Sensors and Actuators A: Physical, 202*, 124-134. http://dx.doi.org/10.1016/j.sna.2013.01.032

Zukas, J. A. (1980). Impact dynamics: theory and experiment. US Army Armament Research and Development Command, Ballistic Research Laboratory Technical Report ARBRL-TR-02271.