Naval Applications of Enhanced Temperature, Vibration and Power Monitoring

by
Ryan David Zachar

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degrees of
Naval Engineer
and
Master of Science in Engineering and Management
at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June 2015

Author .....
Department of Mechanical Engineering
May 15, 2015

Certified by .....
Steven B. Leeb
Professor
Thesis Supervisor

Certified by .....
Peter Lindahl
Postdoctoral Associate
Thesis Supervisor

Certified by .....
John Donnal
Doctoral Candidate
Thesis Supervisor

Accepted by .....
Patrick Hale
Director, System Design and Management Fellows Program
Thesis Reader

Accepted by .....
David Hardt
Chairman, Department Committee on Graduate Students

Naval Applications of Enhanced Temperature, Vibration and Power Monitoring

by

Ryan David Zachar

Submitted to the Department of Mechanical Engineering
on May 15, 2015, in partial fulfillment of the
requirements for the degrees of
Naval Engineer
and
Master of Science in Engineering and Management

Abstract

Navy ships require reliable information regarding their power and mechanical systems in order to perform their mission effectively. While today's shipboard systems are quite sophisticated, there are areas for improvement in monitoring individual loads, managing the loads to fit the ships mission, and continuously monitoring mechanical equipment. This thesis presents a method to continuously assess the condition of a rotating machinery system using vibration analysis during the machine's spin-down. A method to determine the thermal storage capacity of a structure, so that HVAC loads can be more effectively managed, is also explained. Finally, the potential impacts of a Non-Intrusive Load Monitor (NILM) on a ship are investigated.

Thesis Supervisor: Steven B. Leeb
Title: Professor

Thesis Supervisor: Peter Lindahl
Title: Postdoctoral Associate

Thesis Supervisor: John Donnal
Title: Doctoral Candidate

Acknowledgments

I am extremely grateful to the US Navy for giving me the opportunity to study at MIT. Thank you also to the 2N staff for always being there to support me both academically and with all of the other administrative matters. I got to spend 3 years with some great people in the 2N program, thank you for being great friends, and for sticking together during those super fun classes our first year.

Thank you also to the US Navy and US Army personnel outside the program who let us run field tests on your equipment, especially the crew of USS CHAMPION. You were very accommodating and made what could have been a huge hurdle very manageable.

To Professor Leeb, thank you for giving me the opportunity to do some real, relevant research and challenging me throughout the past year more than I ever thought. And finally, I wouldn't be where I am today without Team NILM: John Donnal, Pete Lindahl, and Will Cotta. I learned so much from all of you and had a blast on all our trips and lunches. I will miss getting to work with you all on a daily basis.

THIS PAGE INTENTIONALLY LEFT BLANK

Contents

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 1

The Roll of Energy and Data in Ships

The US Navy is budgeted to spend over $11.2 billion on ship operations and maintenance in fiscal year 2015 [1]. Included in this figure is the routine and corrective maintenance on ships equipment as well as fuel to power the vessels. Continued fiscal pressure presents the opportunity for the US Navy to utilize promising new technologies, which can reduce this figure and support a more effective fighting force. This thesis explores continuous vibration monitoring of motors and generators in an effort to diagnose a potential failure before it occurs. It also investigates the use of thermal storage and power monitoring to more effectively manage power use.

Vibration analysis is a common method used to asses the mechanical health of a machine. Periodic vibration measurements can be used as a tool for condition based maintenance, informing the operator when repairs need to be made. Rotating machines such as engines and motors need to be balanced and properly mounted to function correctly, and excessive vibration can indicate an abnormality. Excessive vibration also leads to increased noise levels, which may be critical in certain applications where low radiated noise is a mission requirement.

For example, the US Navy's fleet of Mine Countermeasures Ships (MCMs) employ resilient mounts on much of their engineering machinery, including the Ship Service Diesel Generators (SSDGs), in order to minimize radiated noise from the machines to the hull. This is important since the ships are designed to operate in an environment where radiated must be kept to a minimum, and a SSDG can be one of the most

substantial contributors. The health of the mount is difficult to determine since its encasement precludes a simple visual inspection. Vibration and acoustic analysis is currently used to determine when a mount may need to be inspected. Navy engineers perform vibration assessments by measuring the vibration above and below the mount when it is at a steady state operating speed. These vibration readings can be put through a spectral density function which shows the data distribution over a band of frequencies. A reduction in vibration from the top of on mount to below the mount can be checked against certain thresholds in frequency space to detect if the mount is operating correctly. Another way to check the mounts is through a far field acoustic measurement. The ship goes out to an area of water with an array of hydrophones on the bottom, and turns on and off equipment in order to determine actual radiated noise.

One issue with these methods of inspection is the measurements are taken infrequently, often with years in between assessments. While far field acoustic measurements may offer the best picture of a ship's noise profile, they are expensive and time consuming. The currently performed above and below mount vibration readings are effective, however there are many more machines which need periodic testing than personnel available to perform the tests. Often mount degradations are not diagnosed until they have already failed, at which point the ship must enter a lengthy repair period. Continuous monitoring allows for several more data points and a trend line to emerge which can alert the organization to potential failure much earlier. While continuously monitoring the generator at steady state would incur an unmanageable amount of data, measuring vibration only during spin-down and coupling it to rotor speed can provide a deeper look into the machine's condition, without the need for large data storage and processing.

A complete system for measuring vibration and rotor speed (via back EMF) was designed and deployed on several types of rotating machines. The methodology was also tested on a lab rig consisting of a DC motor driving an AC generator, where several adverse conditions such as imbalances and failing mounts were imposed. Field test on generators of various sizes were also conducted, including testing on three

generators of an active MCM, both in-port and underway. Chapter 2 contains an explanation of the process used to create the transfer functions which visualize the machines natural resonance, and Chapter 3 contains information on the testing and results of several rotating machines, including those in the lab and in the field.

Any ship requires power to function, however the vast amount of technologically advanced electronics and weapons make power especially important on naval vessels. A loss of power can render a ship unable to communicate, perform its mission, or most importantly, defend itself. Most ships also require liquid fuels in order to generate electricity, an especially limited resource which must be conserved wherever possible. US Navy ships also do not have a sophisticated way of identifying priority loads and steering power to them during times of high demand with limited supply. Legacy power systems include a multi-stage load shed system which will shed non-vital loads in the case of generator failure, however this is only used in emergencies and requires a time consuming resetting of switchboards to return to a normal condition. Furthermore, newer weapons and sensors are requiring an increasing quantity of power, and the ship designer currently has little choice but to add generation capacity. An alternative may exist by temporarily turning off or throttling back loads which consume a large quantity of power, without significantly affecting shipboard operations. This effectively adds to the power available for short term, high energy loads without having to add additional generating capacity.

One of the largest consumers of power on ships is heating, ventilation, and air conditioning (HVAC). A collection of air conditioning plants, fan coil units, and resistive heaters work to maintain the proper operating environment for a ship's equipment as well as crew comfort. One opportunity for shifting power demand in existing loads is to take advantage of thermal storage and reduce the HVAC loads temporarily without significantly affecting equipment or crew comfort. Using the Base Camp Integration Laboratory (BCIL) at Fort Devens, MA as a testing ground, a model was developed to determine the impact on a structure, or ship compartment, if heating or cooling were to be turned off for a period of time. This method is flexible and can eventually be applied to a ship and provide information for an energy

management system. The method and field test results are presented in Chapter 4.

Although shipboard power control systems are continuously updated, there are still many opportunities to improve how the loads are monitored, managed, and backed up in case of a loss of generating capability. Naval architects must plan for required and future loads when determining the size of the ship's electrical generating plant. Unfortunately, these determinations are just estimates and are of limited use after the ship begins its service life. Any sort of in-service monitoring beyond overall consumption requires an expensive installation of sub-meters throughout the ship. A non-intrusive load monitor (NILM) provides effective determinations of individual loads while the ship is in service. These determinations allow for energy score-keeping at a ship or system level as well as inferences of human activity and how they correspond to the ship's daily battle rhythm. The NILM can also facilitate condition based maintenance of equipment by detecting when the power draw is abnormal. Information from these three aspects of the NILM makes it possible for a more intelligent power system to be designed that gives the operator greater knowledge of their electrical system. This increased knowledge of power use can lead to energy savings if changes are made based on the use patterns identified. Non-contact sensors, which allow an inexpensive, rapid installation solution for NILM were tested on two active US Navy ships to determine their feasibility on ship power cables. A complete NILM with non-contact sensors was also tested on an MCM during in-port and underway periods. Chapter 5 contains an explanation of these tests.

Chapter 2

Continuous Vibration Monitoring Theory

This chapter was co-written with William Cotta.

Most vibrations measurements are taken at steady state. A sensor is typically attached to a machine via a mounting stud or a magnet for a certain period of time. Tools such as spectral densities and spectrograms can give vibration levels over a frequency band, which can be used to perform diagnostic tests. Frequency response functions (FRFs) offer a more complete picture of a system, and are commonly used to determine the natural resonance and damping characteristics of a system as well as the noise transfer paths. Frequency response can be measured in several ways. Two common methods are excitation with an impulse hammer or with a shaker [2]. In its most basic form, a FRF is a representation of the response of the system, in this case vibration, to certain frequencies. For rotating machines, an alternative to these methods exists which requires no external excitation. Every time the machine spins up or spins down it passes every frequency between 0 and steady state operating speed. The machine becomes its own network analyzer just by going through its normal start up and shut down routine. Since the machine is sweeping through an entire frequency band it will resonate at a certain frequency, or rotor speed, corresponding to its natural frequency.

This research project of vibration sensing during machinery spin-down is an extension of the work originally performed by Chirs Schantz [3]. Vibration measurements need only be taken during the period the machine is spinning down, and if rotor speed is known the measurements can be used to create an estimated Frequency Response Function (FRF). The estimated FRF, hereafter referred to as a Vibration Transfer Function (VTF), can be used to compare machinery spin-downs to known baselines, allowing the condition of the machine and its associated hardware to be tracked every time the machine shuts down. This amounts to near continuous monitoring, since it does not require human interaction to conduct the test and and the equipment is always in place on the machine. There are two major trends which can be observed from a VTF. First, as the mounts degrade and their hardness changes, the resonant peak will shift in frequency. Second, as an imbalance develops in the rotor the magnitude of the resonant peak will increase, with little to no shift in the frequency.

2.1 Derivation of Transfer Function

A motor on resilient mounts can be modeled as an eccentric mass vibrator connected to a baseplate by a spring and damper, as shown in Fig. 2-1. The spinning motor generates a force which is counteracted by the resilient mount. The equation of motion for a spring mass damper is

$$m\ddot{x}_m(t) + c\dot{x}_m(t) + kx_m(t) = \mu mr\omega^2 cos(\omega t) \tag{2.1}$$

where m is the mass of the motor, $x_m(t)$ is the position of the motor, c is the damping coefficient, k is the spring constant, and $\mu mr\omega^2 cos(\omega t)$ is equal to the force generated by the spinning motor, $F_e(t)$.

If Eqn. 2.1 is rewritten in terns of acceleration rather than position, the new equation of motion is

$$ma_m(t) + c\int_{-\infty}^{t} a_m(\tau)d\tau + k\iint_{-\infty}^{t} a_m(\tau)d\tau = F_e(t) \tag{2.2}$$

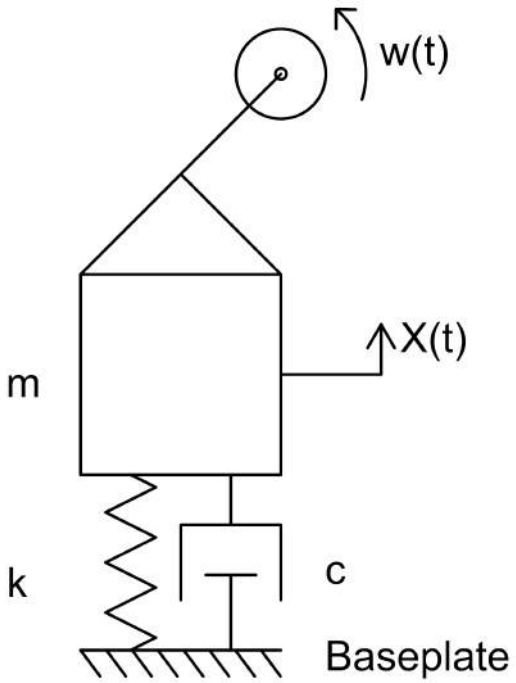


Figure 2-1: Free Body Diagram of Eccentric Mass Vibrator

where $a_m(t)$ is the acceleration of the motor. The equation of motion can be transformed into the Laplace domain as

$$A_m(s)(m + \frac{c}{s} + \frac{k}{s^2}) = F_e(s) \tag{2.3}$$

where s is the Laplace variable, and $A_m(s)$ is the Laplace transform of acceleration from Eqn. 2.2. Rearranging variables results in the transfer function from force imparted on the motor due to the motor spinning, F_e, to the acceleration which it causes, A_m. This transfer function is

$$\frac{A_m}{F_e} = \frac{s^2}{ms^2 + cs + k} \tag{2.4}$$

The damping ratio can be defined as

$$\zeta = \frac{c}{c_c} \tag{2.5}$$

where $c_c = 2\sqrt{km}$. ζ is observed to be less than .1 for practical isolation arrangements, such that the transmissibility in the isolation range is similar to that for zero damping [4]. Thus, c can be ignored when operating speeds are away from resonance. At a motor's operating speed, the magnitude of the force transmitted through the mounts to the baseplate is primarily through the spring and follows Hooke's law,

$$F_t = k \cdot x(t) \tag{2.6}$$

where F_t is the force transmitted, k is the spring constant, and $x(t)$ is the amplitude of the position at operating speed, ω, at time t. If the simple harmonic motion is assumed, then the equation of motion, as derived in [5], is given by

$$x(t) = Acos(\omega t) \tag{2.7}$$

where A is the maximum displacement from equilibrium. Position can be shown as proportional to acceleration by taking the second derivative and substituting,

$$a(t) = x''(t) = -\omega^2 Acos(\omega t) \tag{2.8}$$

$$a(t) = -x(t)\omega^2 \tag{2.9}$$

$$x(t) = \frac{a(t)}{\omega^2} \tag{2.10}$$

The natural frequency of the system is defined as

$$\omega_n = \sqrt{\frac{k}{m}} \tag{2.11}$$

therefore k is equivalent to $m\omega_n^2$. If k and $x(t)$ are substituted in Eqn. 2.6, then

$$F_t = \frac{m\omega_n^2 a(t)}{\omega^2} \tag{2.12}$$

For diagnostic purposes the magnitude of the force transmitted, F_t, is the interesting

metric. If the magnitude of Eqn 2.13 is taken then

$$|F_t| = \frac{m\omega_n^2 A_m}{\omega^2} \tag{2.13}$$

where

$$A_m = |a(t)| = max\ measured\ acceleration \tag{2.14}$$

A_m would be the maximum reading on an accelerometer. It is possible to estimate the force transmitted compared to a known measurement as the spring constant changes, which can be caused a change in mount condition. Estimating the force transmitted relies on the assumption that the mass of the motor is not changing, and uses the methods presented in the following sections to calculate the natural frequency, ω_n. By measuring the maximum magnitude of the acceleration of the motor at operating speed, the ratio of force transmitted through the mounts from a baseline level can be estimated as

$$\frac{|F_t'|}{|F_t|} = \frac{|A_m'|\omega_n'^2}{|A_m|\omega_n^2} \tag{2.15}$$

where the variables with prime superscripts indicate current or predicted conditions and variables without superscripts indicate a baseline. Using this method requires a reliable baseline condition for future comparisons. Continuous vibration monitoring does not attempt to fully characterize a system; instead the goal is to detect changes in the frequency response, and most importantly the natural frequency. A shift in natural frequency can indicate a change in the condition of the mounts. The following sections demonstrate how a machine can be monitored by tracking changes in natural frequency using only a single voltage sensor and an accelerometer. The resonant frequency can be calculated using time frequency analysis of the spin-down period of a machine, which permits the calculation of the change in force transmitted to the baseplate.

2.2 Constructing a VTF

2.2.1 Time Frequency Analysis

Rotating electrical machines provide a unique opportunity for time frequency analysis of vibrations, since it is possible to measure the rotational speed driving the motor. The speed of an AC induction machine can be estimated through voltage measurements of the phase lines and knowledge of the pole pairs of the machine. The method used here for estimating the transfer function of rotational speed to mount vibration is based on the method presented by E. Feron et al, in [6]. The authors attempt to understand aeroelastic behavior in an F/A-18 research aircraft by generating a known input vibration in the wingtips. The modified procedure for evaluation of a motor or generator system is:

1. Perform the time-frequency analysis of both system inputs (rotational speed) and outputs (vibration acceleration).

2. For each frequency of interest, remove the parts of the output that have not been excited by the input.

3. Obtain estimates of the frequency response using the clean time-frequency representations of the signals.

As outlined in [3], transfer function estimation through spin down analysis can be susceptible to noise from other pieces of machinery rotating in the same space. This method attempts to reduce uncorrelated noise by ignoring vibrations outside the area corresponding to a windowed input.

2.2.2 Voltage Preprocessing from Back EMF Data

The method begins with 3 raw data streams; 2 voltage and 1 acceleration. The first step is to extract the speed from a back EMF sensor, which returns 2 voltage streams obtained through non-contact sensors on the phase lines of a motor or generator as

it spins down. Noise from the environment and other phase lines is inherent in non-contact sensing, and the 2 sensors may have differing voltages as well as background waveforms which need to be reconciled in order to produce the machine's true voltage. A 1 degree polynomial fit of 1 sensor to the other when the machine is off and the voltage is known to be zero is calculated. The difference between the first sensor and the new fitted data stream is the true voltage. A more detailed explanation of the non contact voltage sensor can be found in Chapter 3.

An example voltage plot for the start-up and spin-down of a generator can be seen in Fig. 2-2a. Zooming in to the area where the machine is off, as shown in Fig. 2-2b, offers an example of the voltage sensor fitting. Even though the machine is off there are still 60 Hz waveforms in the raw data, however if the two streams are subtracted the resulting plot is zero when the machine is off, and an accurate voltage from the machine when it is on or spinning down.

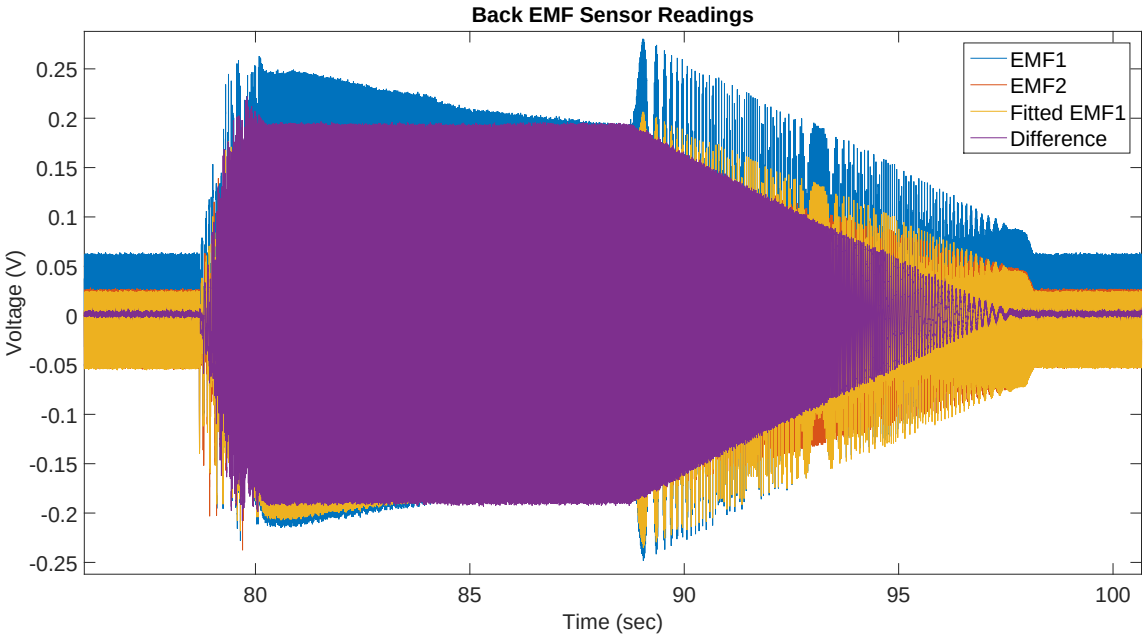
2.2.3 Hilbert Transform Speed Extraction

A Hilbert transform can be used to extract the spectral envelope of the voltage waveform [7]. Once a motor is disconnected from its power supply, the rotor will continue spinning due to its inertia, generating voltage in the stator. The amplitude of the voltage in the stator has a linear relationship to the speed as shown by

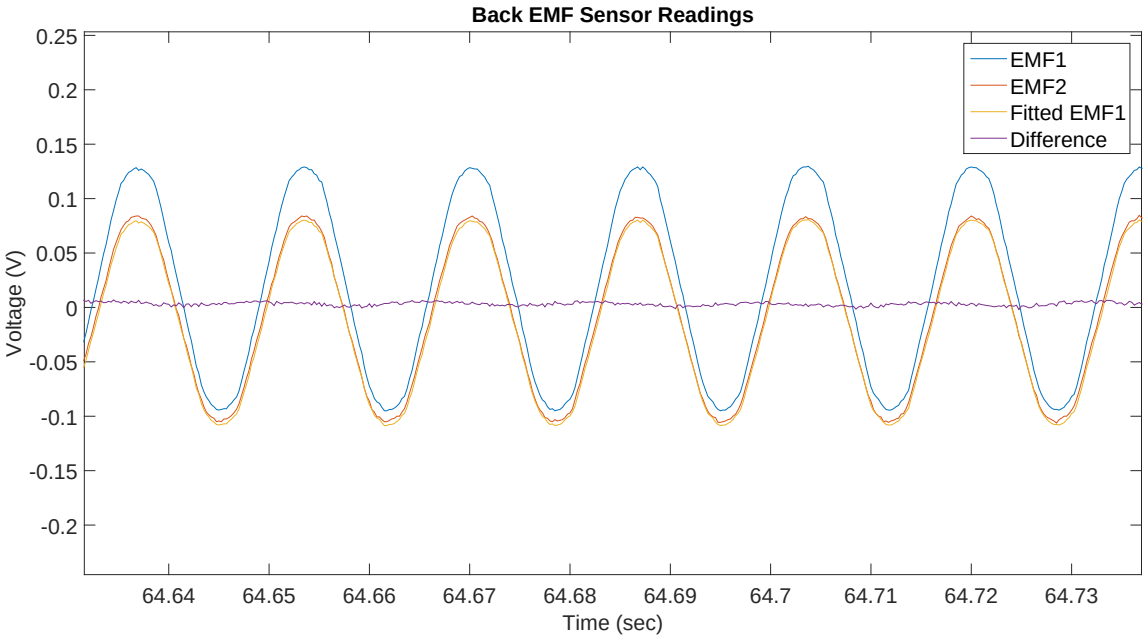
$$\omega = \frac{V \cdot k_{motor}}{P} \tag{2.16}$$

where V is the voltage sensor output, P is the number of pole pairs in the motor, ω is the motor speed in Hz, and k_{motor} is a calibration constant.

A detailed description of using the Hilbert transform to extract a spectral envelope is given in [7]. A summary is provided here. Assuming that the useful content of a signal appears at the frequency ω_c, and the amplitude of that useful information, $A(t)$, is assumed to be changing slowly compared to ω_c so as to be constant, the envelope can be extracted. In the following example, shown in Fig. 2-3, the enveloping signal is at 5 Hz with an amplitude of 1, yet it is corrupted with 60 Hz noise. The true



(a) Voltage Sensor from Start up and Spin Down of 3 Phase Induction Motor



(b) Voltage Sensor Fitting

Figure 2-2: Preprocessing Voltage Sensors

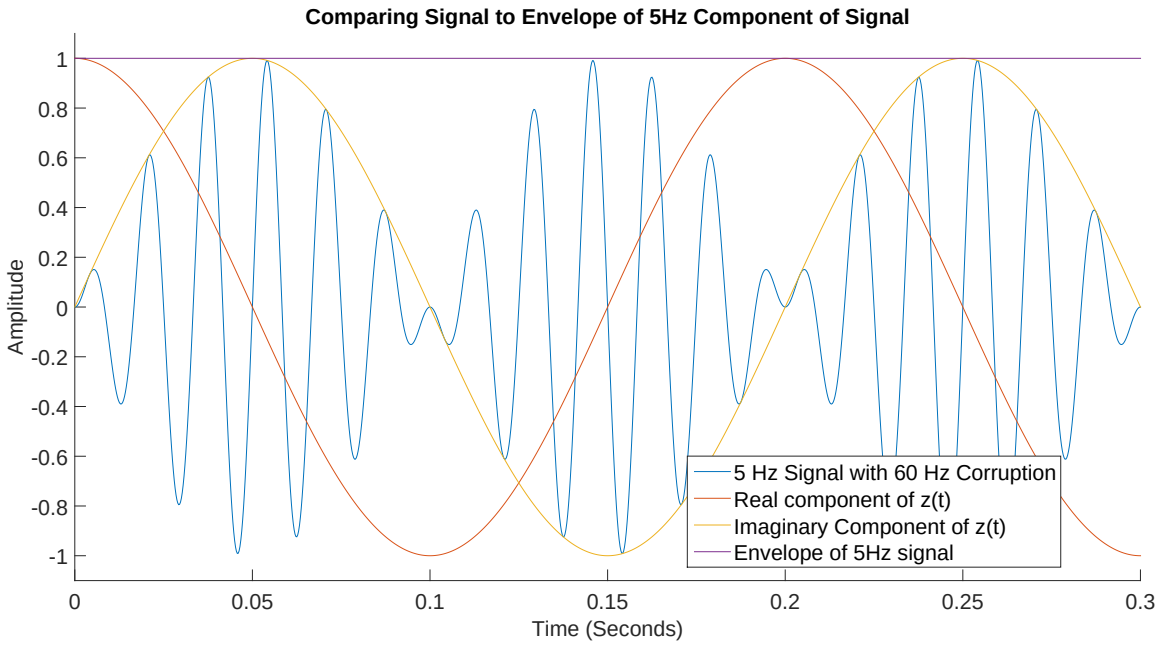


Figure 2-3: Extracting Envelope of 5Hz Signal using the Hilbert Transform

signal is given as

$$x(t) = sin(5 \cdot 2\pi \cdot t) \tag{2.17}$$

To extract the envelope, an analytic signal is considered,

$$z(t) = A(t)x(\omega_c t) + jH(A(t)x(\omega_c t)) \tag{2.18}$$

It is a property of the Hilbert transform that $H(sin(\theta)) = -cos(\theta)$, where H indicates the Hilbert transform, making the analytic signal $z(t) = A(t)(sin(5{\cdot}2\pi{\cdot}\theta) - jcos(5{\cdot}2\pi{\cdot}\theta))$. The extracted envelope of the signal is the approximately equal to the magnitude of $A(t)$,

$$|z(t)| = |A(t)| \tag{2.19}$$

The results for extracting the envelope of the 5Hz signal are seen in Fig. 2-3. An example of the extracted voltage envelope of a motor spin down is shown in Fig. 2-4. This envelope can be easily converted to speed in Hz with Eqn. 2.16, using the known speed at the start to calculate k_{motor}.

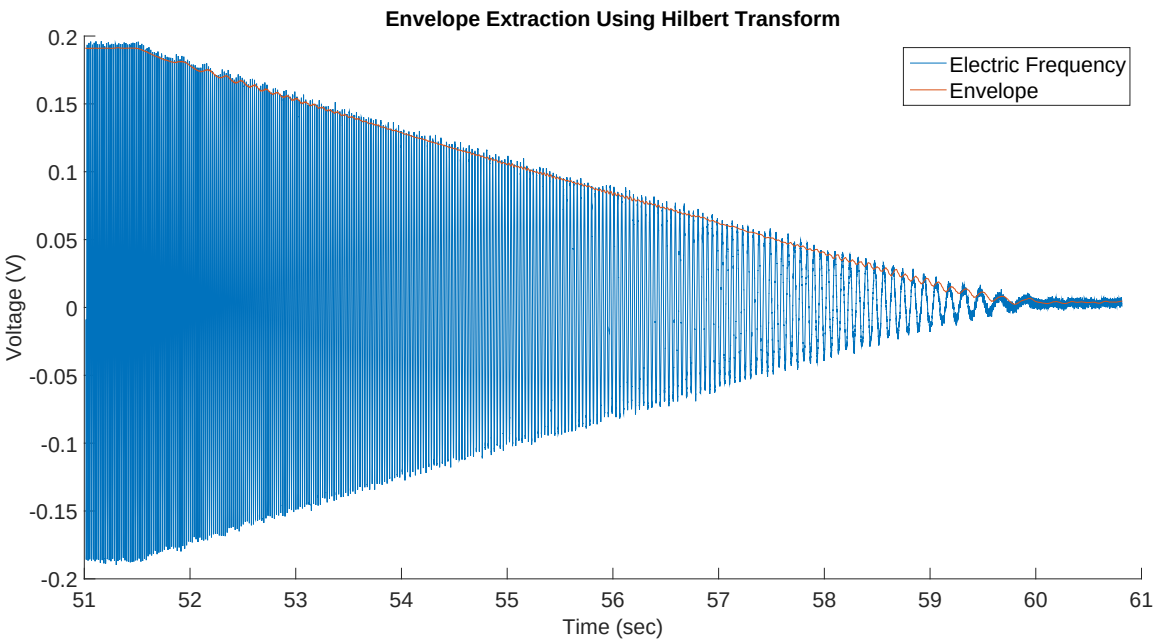


Figure 2-4: Extracting Envelope of Motor Spindown

2.2.4 Zero Crossing Frequency Extraction

In some cases the Hilbert transform is ineffective at extracting speed due to clipping in the sensor output. Clipping occurs when the voltage exceeds the maximum voltage the sensor is able to read. The speed, however, is still recoverable by calculating the frequency of the voltage signal through zero crossings. An example of clipped sensor output can be seen in Fig. 2-5, with zero crossings marked with red circles. Zero crossings were estimated through linear interpolation of points on either side of the zero crossing. The frequency is then determined by taking the inverse of the time difference between zero crossings. The speed is fitted with an exponential decay to ensure a smooth speed curve, and verify that the data conforms to the well established system equation of motion given as

$$\omega(t) = Ae^{-t/\tau} \tag{2.20}$$

where $t = 0$ is the start of the spin down, τ is the moment of inertia divided by the viscous friction constant, $\omega(t)$ is the shaft speed of the motor, and A is the operating

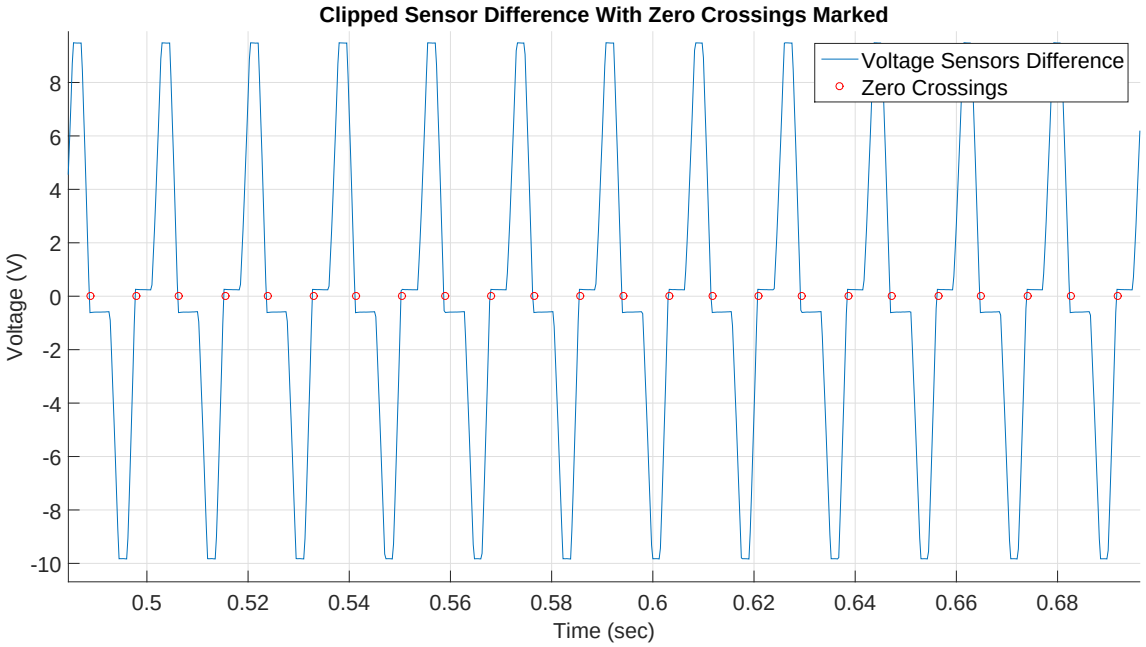


Figure 2-5: Clipped Sensor Voltage with Zero Crossings Marked

speed of the motor. To confirm the validity of the method, the extracted speed curve is compared to the spectrogram of obtained using data from the vibration sensor in Fig. 2-6.

2.2.5 Virtual Input

The method presented in this section models an electric motor or generator mounted on resilient mounts as a single degree of freedom spring mass damper with an eccentric mass vibration, as shown in Fig. 2-1. The equation which governs the motion of the mass, Eqn. 2.1, is equivalent to single degree of freedom spring mass damper system experiencing forced vibration, where $F_e(t) = \mu m r \omega^2 cos(\omega t)$. Using extracted rotor speed and acceleration on the surface, the input force and output response can be used to produce a VTF. According to Eqn. 2.1, the force generated by the eccentric mass is proportional to speed squared, therefore the original speed envelope must be squared so that it is proportional to input force. A virtual signal of the speed envelope squared is created, which squares the speed and provides a clean input

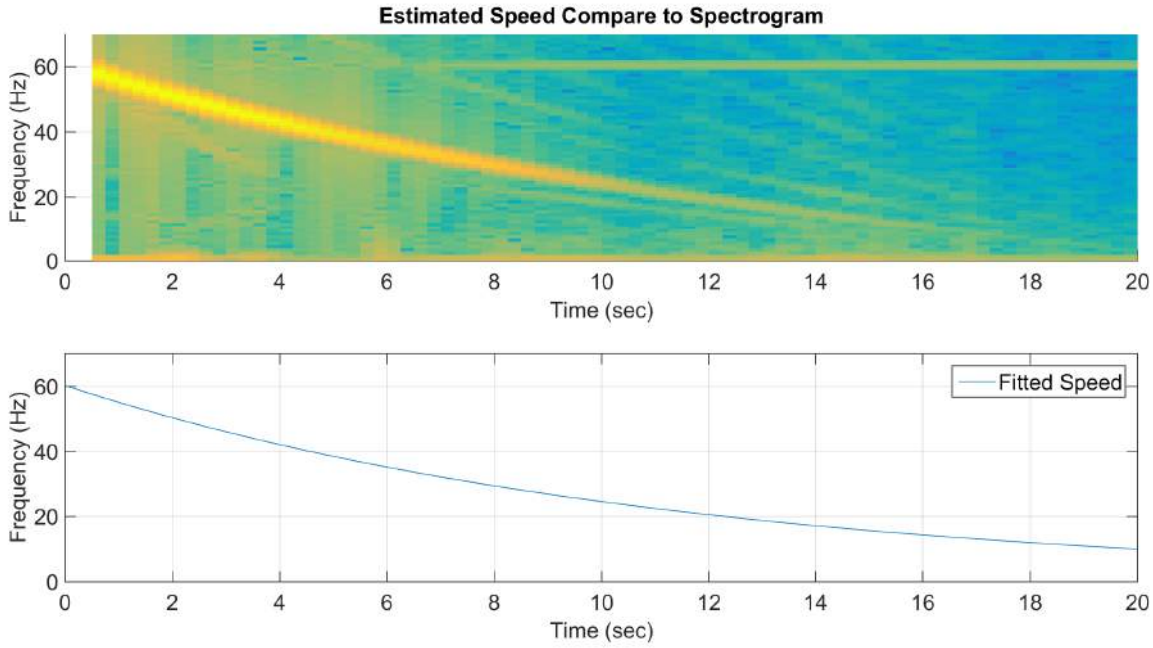


Figure 2-6: Comparison of Voltage Sensor Spectrogram and Extracted Speed Curve

signal for Fourier analysis. This signal is known as the virtual input, and is given by

$$\omega_{virtual}(t) = \frac{\omega(t)^2}{\omega(t=0)^2} cos\Big(\int_0^t 2\pi\omega(\tau)d\tau\Big) \tag{2.21}$$

where $\omega_{virtual}$ is the virtual input and ω is the speed at time t. An example of a virtual input can be seen in Fig. 2-7, with the corresponding speed envelope the virtual input was derived from.

2.2.6 Hanning Window Masked Inputs and Outputs

As a motor spins down it imparts a frequency sweep, probing all the frequencies from the operating speed to 0 for resonances. The voltage sensor measures the time varying electric frequency, known as the chirp, generated by the electromagnetic forces pushed back through the voltage connectors. Fourier analysis excels at steady state, however a fundamental problem of using the Fourier Transform for a chirp is that though the transform is reversible, it is impossible to know at a given time what the frequency spectrum looks like. Instead, the Fourier Transform provides the frequency spectrum

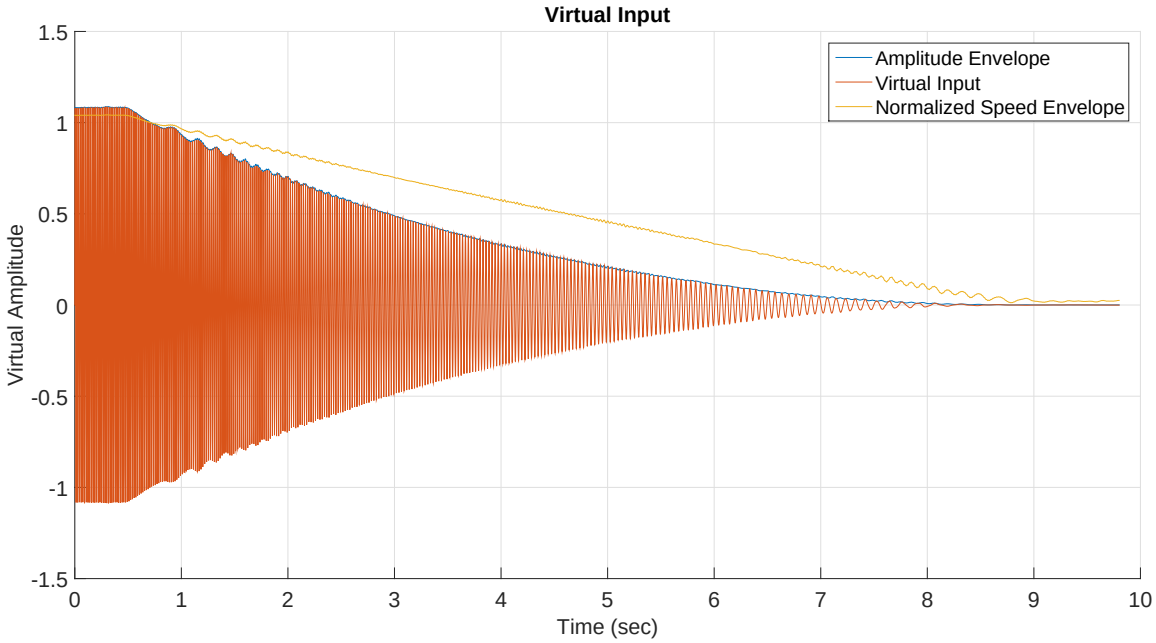


Figure 2-7: Comparison of Normalized Speed Envelope to Virtual Input

for the entire window of time. Using a Short Time Fourier Transform (STFT) can provide small windows which can be analyzed. This allows for the frequency spectrum to be calculated at smaller windows, and a time frequency representation of the signal to be created.

A Hanning window is used to mask the input and output of the system in order to minimize vibration information from adjacent frequencies. The input to the system is the virtual input described previously, and the output is the acceleration measurement in g. The center of the window is the frequency of interest, and the width of the Hanning window is bounded by two parameters, the Nyquist Frequency and the Rayleigh Frequency. The Nyquist Frequency sets the upper bound for frequencies which can be observed at half of the sample frequency. The Rayleigh Frequency sets the lower bound for frequencies which can be observed based on the size of the window in time. The Rayleigh frequency is equal to $\frac{1}{T_w}$, where T_w is the length of the time window. For example, the lowest frequency observable in a 1 second time window is 1 Hz, and the lowest frequency observable in a 0.5 second time window is 2 Hz. The

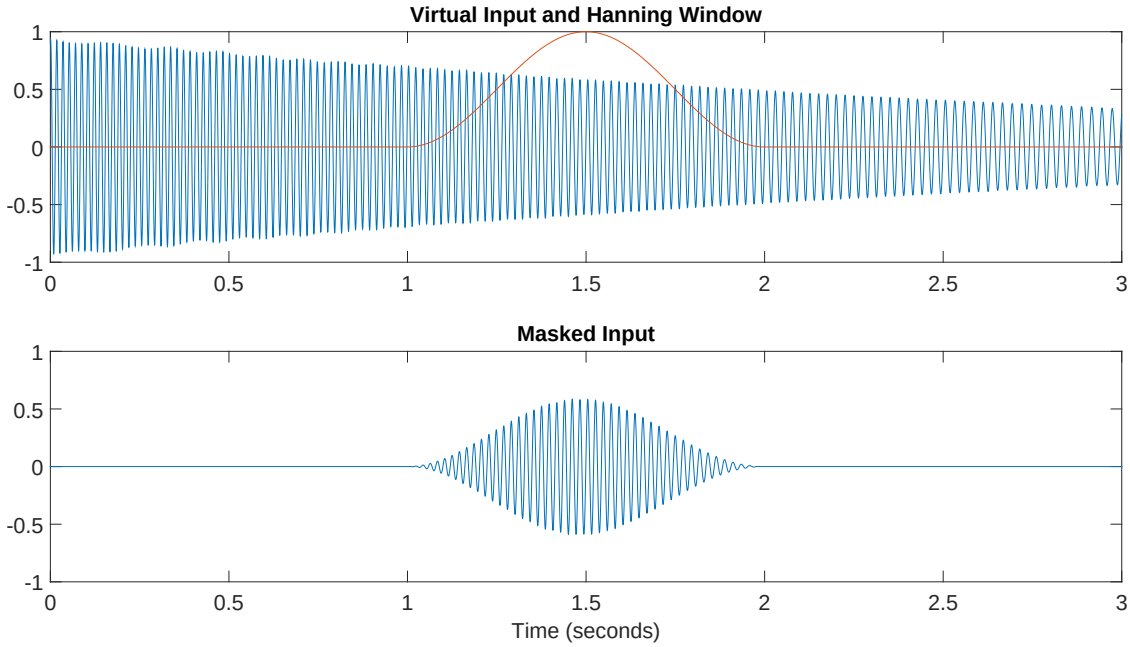


Figure 2-8: Masked Input

size of the Hanning window is determined by

$$n_s = f_s \cdot T_w \tag{2.22}$$

where f_s is the sample frequency. The Hanning window size equation contains the tunable parameters T_w for time window length in seconds. The spacing of Hanning windows is also important, and determined by a tunable overlap parameter, o_v, which specifies the overlap between time windows, where 1 is equal to 100% overlap and 0 is 0% overlap. The Hanning windows are then convolved with the inputs and outputs. As an example, the results of the first Hanning Window convolution for the first 0.25 seconds of the spin down for the input and output are shown in Figs. 2-8 and Fig. 2-9, respectively. Each Hanning window provides a frequency spectrum for a known period of time; by using Fourier analysis on each window a time frequency representation of the motor spin down can be calculated. The Hanning windows as well as the masked inputs and outputs are shown in Fig. 2-10. For clarity only every tenth window is shown.

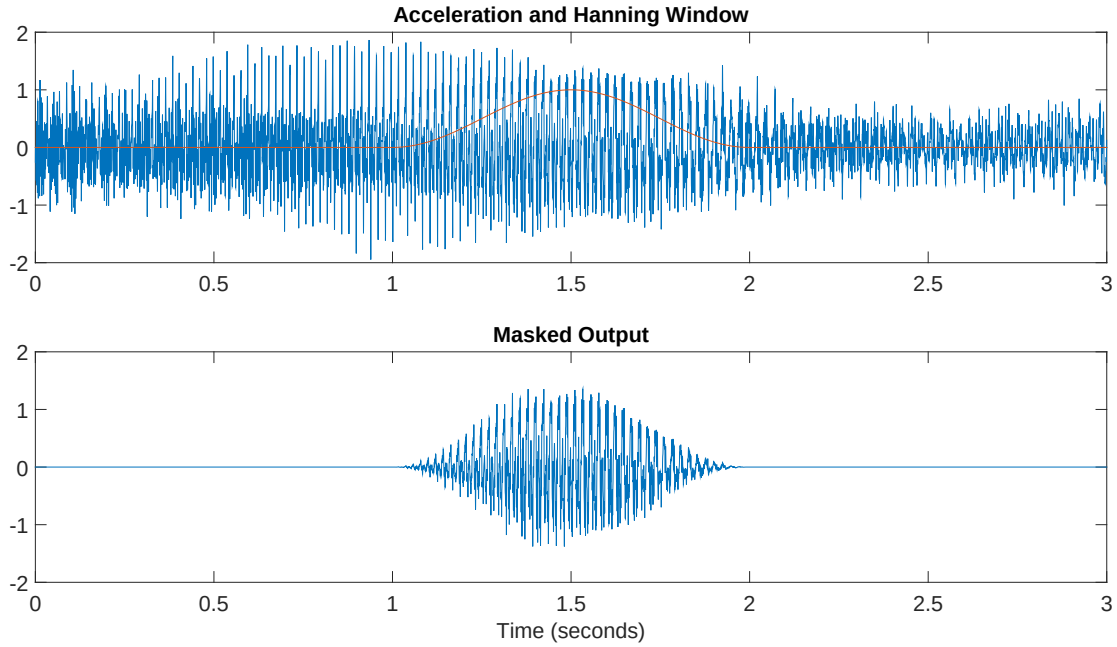


Figure 2-9: Masked Output

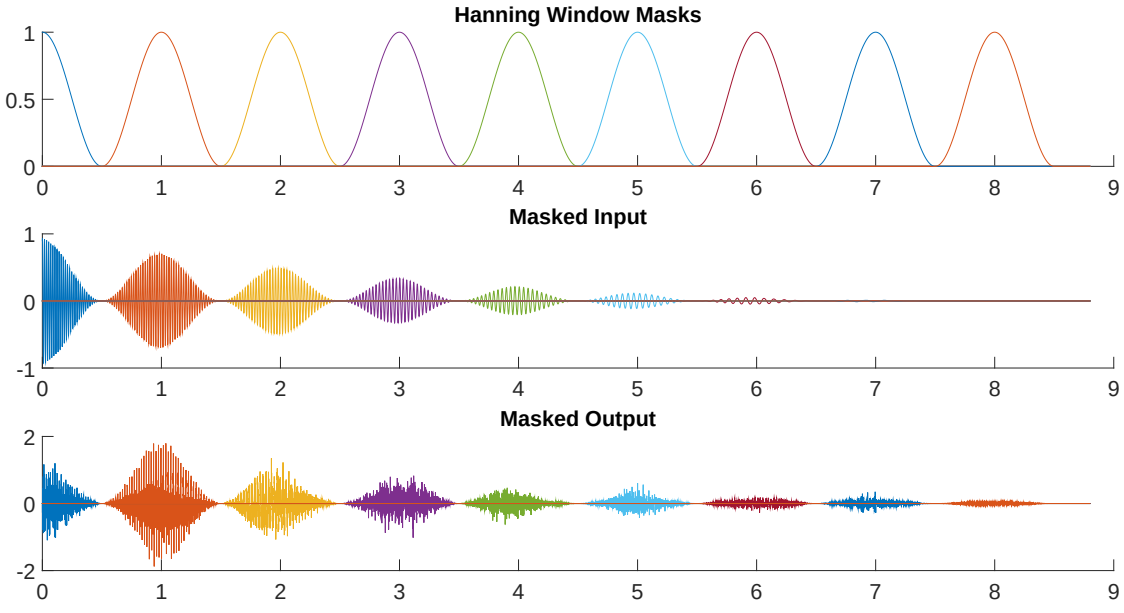


Figure 2-10: Hanning Windows and Masked Inputs and Outputs

2.2.7 Vibration Transfer Function Estimation

The VTF is calculated by taking the STFT of the masked inputs and outputs. STFT utilizes the standard Fourier Transform, but allows for Time-Frequency Analysis by dividing the signal into time windows. Each window is analyzed, providing the estimated response at each frequency in the window. A motor spin-down has an input, the speed of the motor which is related to the force imparted, and an output, the acceleration of the motor caused by the force from the motor rotating. By dividing the STFT of the output signal by the STFT of the input signal, the response at each frequency can be identified. In order to accurately use the STFT in analyzing the signals, the system must be in steady state or quasi steady state. Since the system can only have frequencies in the output which exist in the input, the only frequency content in the output considered valid and not noise is the frequency content where the input is non-zero. Because there is a lag in the acceleration of the motor in response to speed as the motor spins down, it is not actually in steady state.

To account for this and still allow the method to be used, the STFTs of the outputs and inputs for each window are calculated. Envelopes of all output STFTs and all input STFTs are calculated, and divided to produce a VTF. The Fast Fourier Transform (FFT) of a masked input and output are shown in Fig. 2-11. The dashed lines show the envelope of all of the FFTs of the masked inputs and outputs. Using these envelopes, the VTF in Fig. 2-12, in blue, is calculated by dividing the output envelope by the input envelope. With this step, the constraints of a single degree of freedom are still used, and only frequencies which have been excited can exist. This method accounts for lag as the motor spins down, allowing any frequency content in the output that has been excited to be considered valid. If the motor was assumed to be in pure steady state, and only the straight FFTs of the full output and input were calculated and divided, then lag would cause extreme values known to be false, as shown by the orange curve in Fig. 2-12.

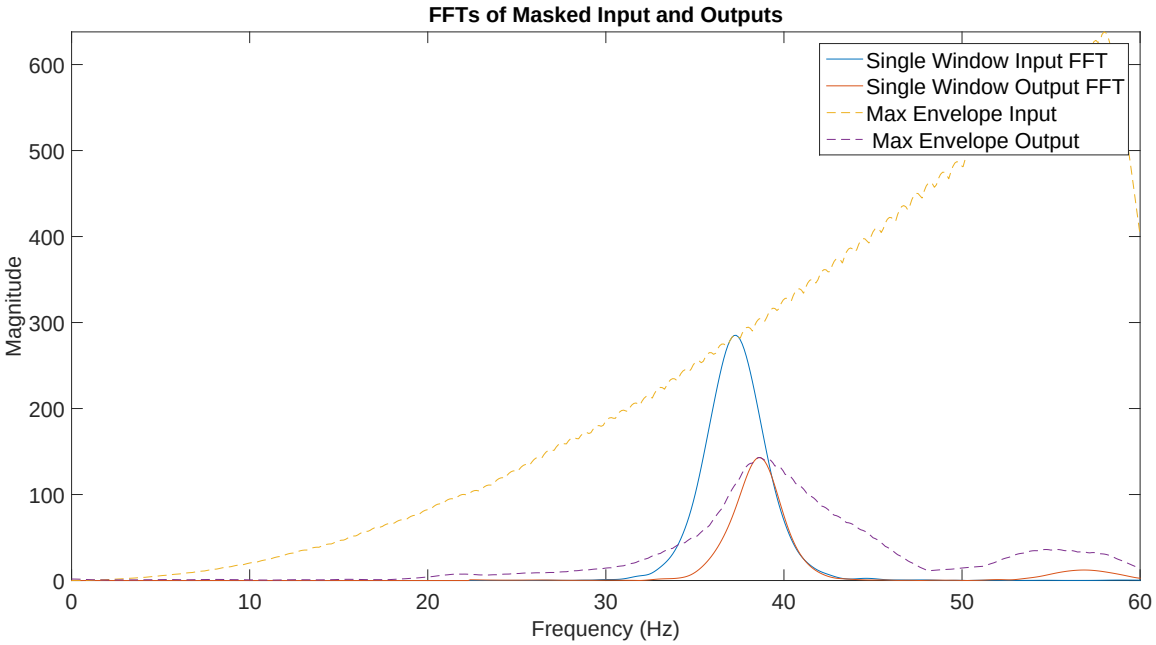


Figure 2-11: Fast Fourier Transform of Masked Inputs and Masked Outputs with Envelopes

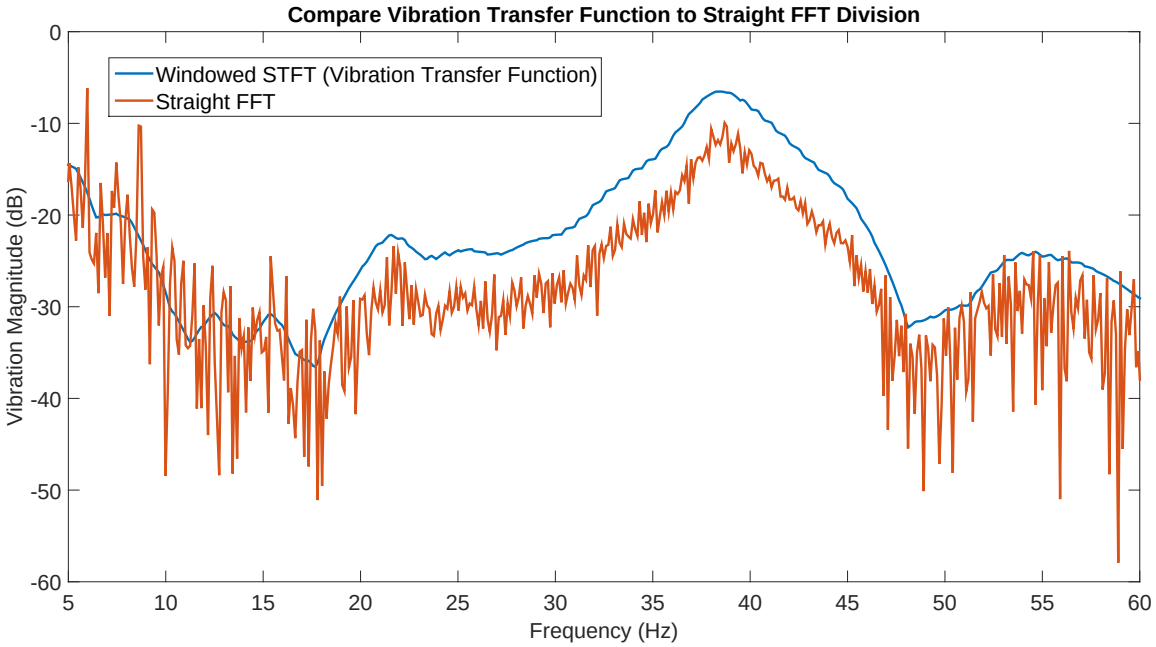


Figure 2-12: Estimated Vibration Transfer Function

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 3

Vibration Tests and Results

The method to create VTFs described in Chapter 2 was applied onto 5 sets of rotating machines; 2 in the lab and 3 in the field. The systems tested in the lab include a purpose-built test platform with a coupled DC motor and AC generator as well as a test stand with two 5 KW generators driven by DC motors. The first field tests were conducted on nine 60 KW generators at the US Army BCIL. The final 2 field tests were conducted on board an active MCM in San Diego, CA. The ship's 3 SSDGs and 2 auxiliary seawater (ASW) pumps were tested over a week long period. The equipment, test setup and results for each experiment set are presented in the following sections.

3.1 Hardware used for Speed and Vibration Data Collection

A mobile, rugged kit was developed with all of the sensors and tools needed to obtain the necessary data to create the VTFs. The Vibration Information Box- Portable (VIB-P) was constructed to deliver high quality sensors to the field in a self contained case which requires only a laptop and standard outlet to operate. The shell of the VIB-P is a Nanuk 920 locking case with an aluminum plate to provide mounting for various data acquisition and power supply modules. Underneath the plate is a power strip for the power supplies and storage for all necessary sensors, cables and

other accessories. Detailed drawings and parts lists for the VIB-P can be found in Appendix A. Fig. 3-1 shows the VIB-P.

(a) VIB-P with Storage Door Open

(b) Close in View of VIB-P

Figure 3-1: VIB-P

The VIB-P contains 2 Omega® ACC786A single axis industrial accelerometers and their associated power supplies. These power supplies are each connected to a LabJack UE9. There are several options for sensor orientation and placement. For example, the sensors can be mounted on two separate locations of the machine, above and below one of the machine's mounts, or on two separate axis (such as the x and y axis). Each accelerometer has a tap which accepts a 1/4-28 stud for mounting. Since it would be impractical to mount a stud on every machine being tested, an alternative method of attachment was needed. Magnets can be used when the sensor must be moved often, however they could not be used on the MCM because the SSDGs are made of a non-ferrous material. In order to accommodate this a set of aluminum mounting blocks were machined, each with a hole to accept a 1/4-28 stud, allowing it to attach to the accelerometer. The mounting block is attached to the measurement point via a 3M Command^{TM} heavy duty mounting strip. Fig. 3-2 shows the accelerometer and its mounting block.

The VIB-P also contains a back EMF sensor. This sensor measures winding

Figure 3-2: Accelerometer and Mounting Block

voltages of an electric machine, which can be used to create a virtual input of speed for the VTF. Although there are other methods to obtain speed from a rotating machine, a back EMF sensor is the most desirable. Most generators have a tachometer installed which gives the operator revolutions per minute (RPM) to ensure the machine is running properly. On US Navy ships some even record this number at a certain intervals as part of the Integrated Condition Assessment System (ICAS). Nevertheless accessing this data and integrating it into the vibration sensor system is problematic and it is unclear if this data is of high enough fidelity to create a VTF. An independent tachometer could be mounted on the machine to record the spin-down speeds and be easily integrated into the vibration sensor system, however it is not ideal for a portable system. Installing and removing an independent tachometer, especially on a large machine such as a generator, can take considerable time and effort.

A back EMF sensor is the preferred method for gathering spin-down speed because it is accurate, portable and easy to install in the field. It does not measure the speed directly, rather it measures the residual voltage as the machine spins down. Voltage is related to machine speed as outlined in Section 2.2. In order to use the back EMF sensor, the steady state operating speed before shutdown must be used as a reference point. The remaining speeds extracted from the back EMF signal are dependent on the initial speed given. Fig. 3-3 from [3] shows a demonstration of how the speed estimate determined from zero-crossings of back EMF data compares to the speed obtained from a shaft encoder.

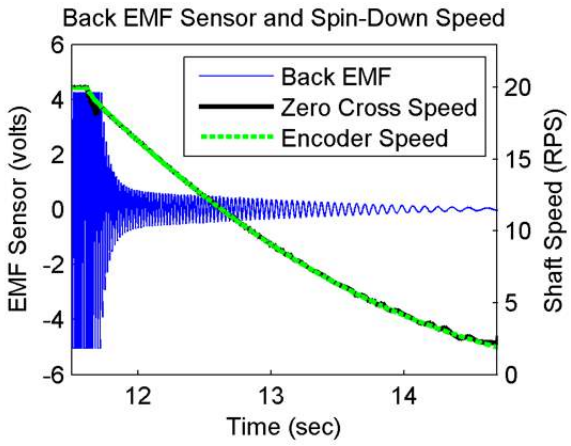


Figure 3-3: Back EMF Zero Cross and Encoder Speeds

The back EMF sensor uses differential capacitive sensing to detect the electric field generated by the phase lines to the motor or a generator. Fig. 3-4 is a schematic of two identical halves of the back EMF sensor. Three copper plates electrically connected to a circuit board are zip tied onto the phase lines coming directly off of the generator. 1 plate connects to the + side of the first AD8421 differential amplifier, a second plate connects to the (-) side of the first AD8421 as well as the (+) side of the second AD8421, and the third plate connects to the - side of this second amplifier. The voltages generated on the sensors correspond to the line-line voltages of the generator, V_{A-B}, V_{B-C} and V_{C-A}. If the first, second and third plates read phases A, B and C respectively then 1 amplifier would read $G_a(V_A - V_B)$ and the second would read $G_a(V_B - V_C)$, where G_a is the gain of the amplifier, V_A is the voltage off of phase A, V_B is the voltage off of phase B and V_C is the voltage off of phase C. This method allows any background 60 Hz disturbances to be subtracted out in further processing.

The back EMF sensor is connected to a power supply via an Ethernet cable, which outputs the 2 back EMF measurements as ±5V signals. Fig. 3-5a shows the back EMF sensor and Fig. 3-5b shows the sensor leads connected to the phase lines of of a MCM generator. The 2 vibration and 1 back EMF power supplies connect to a LabJack UE9, which converts the analog data into a digital representation with 16-bit precision. The LabJack is connected to a computer via either Ethernet or USB. A

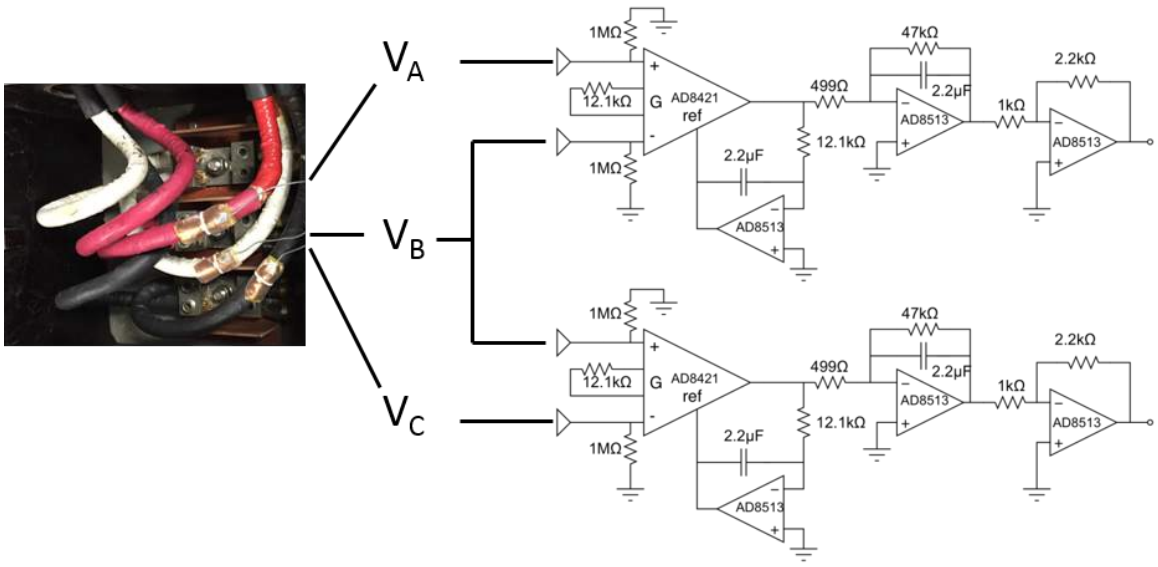
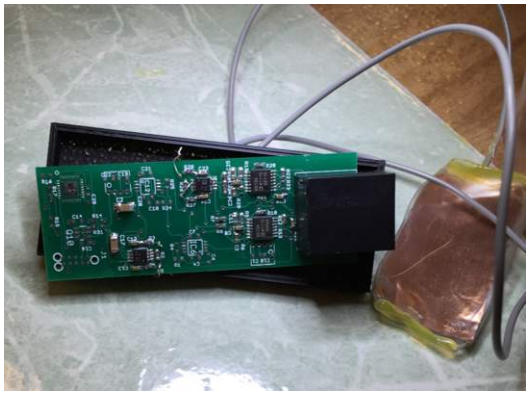


Figure 3-4: Back EMF Sensor Schematic

program called LJStreamUD is available from `www.labjack.com`, and offers plug and play installation on a PC. Data can also be read from the LabJack with a Python code known as *ethstream*, created by Jim Paris [9]. The VIB-P proved to be a rugged, easy to use, all-in-one tool kit for measuring the data needed to construct a VTF both in the lab and in the field. The following sections describe these experiments.



(a) Back EMF Sensor

(b) Sensor on Phase Lines

Figure 3-5: Back EMF Sensors

3.2 Lab Motor Tests

A test stand consisting of multiple motors was constructed in the Laboratory for Electromagnetic and Electronic Systems (LEES) in order to develop and test the methods for constructing VTFs. The Motor Mount Test Platform (MMTP) contains a DC permanent magnet motor with duel couplings. One end is coupled to a 3 phase induction motor, which emulates a generator, and the other is connected to a single phase synchronous AC motor, which is disabled so it acts only only a flywheel. The purpose of this flywheel is to add mass to the rotation and extend the spin down time to a similar length as that of the MCM generators. The MMTP is depicted in Fig. 3-6. The DC driving motor is in the center, the induction motor is on the right, and the flywheel is on the left. Detailed drawings and construction information for the MMTP can be found in Appendix B.

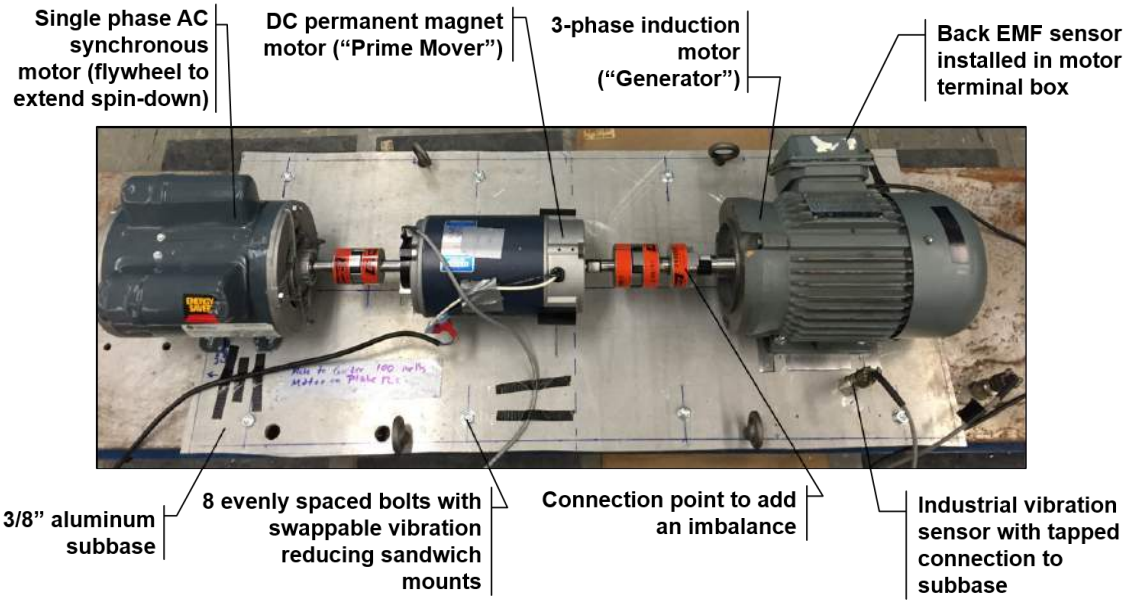


Figure 3-6: Motor Mount Test Platform

The three machines are mounted onto a single aluminum subbase, which emulates the subbase typical of a typical ship generator set. The subbase is mounted to a steel box girder with vibration reducing mounts at eight points. A second shaft coupler is attached to the induction motor as an attachment point for an imbalance. A 1/4-20 screw fits into the insert for the set screw of the coupling, and washers can be added

to create a mass which will imbalance the machine. This is done to simulate a rotor imbalance, a type of machinery fault, which will appear as an increase in vibration magnitude at resonance. The MMTP is designed so the mounts can be changed out easily. The subbase simply needs to be lifted off of the mounts while they are changed and then set and bolted back down. There are 4 eye-bolts which allow the assembly to be lifted with a shop crane via a four-leg bridle. The motors never need to be realigned after initial alignment since they are fixed to the plate and never move. This is critical as it reduces the chance that a factor other than a mount change or imbalance would affect the vibration of the machine.

Five commercial vibration dampening mounts of different durometer were used to emulate a scenario where the mounts stiffness increases or decreases over time. The mounts were all cylindrical and had the same dimensions: diameter of 3.81 cm and height of 2.54 cm. Durometer is a measure of hardness often used to describe the rubber used in vibration dampening mounts. Different scales are used for durometer depending on the hardness of the material; the A scale is used for softer materials such as elastomers, while the D scale is used for a harder material such as a plastic. All mounts used for testing on the MMTP use the A scale. The five mounts had durometers of 30A, 40A, 50A, 60A and 70A. The mounts are made of silicone which is sandwiched between two steel plates. There is a 3/8"-16 mounting stud on one end and a 3/8"-16 insert on the other for mounting to the aluminum subbase and steel girder. Fig. 3-7 shows the 5 types of vibration reducing mounts used for testing with the MMTP.

An estimated resonant frequency for the system with each mount can be calculated if the mount properties and system weight are known using the equation

$$\omega_n = \frac{1}{2\pi}\sqrt{\frac{k}{M}} \tag{3.1}$$

where k is the axial stiffness and M is the mass per mount. M is simply the total mass of the system (57.24 kg) divided by 8 since there are 8 mounts. k is determined

Figure 3-7: Vibration Reducing Mounts used with MMTP

using

$$k = \frac{AE}{L} \tag{3.2}$$

where A is the cross sectional area of the mount (1.14×10^{-3} m), E is Young's Modulus, and L is the length, or height, of the mount (2.54×10^{-2} m).

Equations exists which exploit a linear relationship between durometer, which is a hardness measurement, and Young's modulus, which typifies stiffness of an elastic material. For example, Mix and Giacomin present a method for determining a standardized Young's modulus from the durometer which takes into account different indentation mechanics in [10]. Their method is applied in a simple to use mobile app, Duromod$^{\text{TM}}$, which easily converts the durometer to Young's modulus. Table 3.1 shows the estimated resonant frequencies for each mount durometer.

	30A	40A	50A	60A	70A
Estimated Resonant Frequency (Hz)	15.5	18.9	22.8	27.6	34.2

Table 3.1: Estimated Resonant Frequencies

The MMTP is powered by a DC power supply, which connects to the DC motor that in turn drives the other two machines. The power supply provides constant voltage, and the operator can vary the voltage to change the operating speed of the motor. A US Digital E3 shaft encoder delivers real time revolutions per minute (RPM) to the operator, which can easily be converted to Hz.

Sensors from VIB-P were installed on the MMTP using a 1/4-28 stud and taps

in the subbase and steel girder. The three leads from the back EMF sensor were originally zip tied onto the three phase lines of the induction generator, however initial tests revealed that the resulting back EMF pick-up was insufficient for reliably inferring speed. Therefore, the leads were removed from their protective plastic and wrapped directly on the wire and covered with electrical tape. The entire sensor unit was sealed in the metal terminal box to minimize background 60 Hz noise. Although this departs from how the sensors are attached in the field, this was deemed acceptable based on the small size of the test motor and the fact that even the smallest generators tested produced large enough back EMF to be distinguishable from 60 Hz interference. Two test procedures were run on the MMTP: steady state and spin-down. Steady state tests were used to verify the VTFs produced via the spin-down tests. The following 2 sections describe these tests and their results.

3.2.1 Steady State Tests

A actual FRF can be created by running a motor at several steady state speeds between 0 an its operating speed and measuring the vibration response at each speed. This FRF can be compared to a VTF created by spin-down analysis to determine how well the method is estimating vibration levels at known frequencies. Multiple steady state tests were conducted on versions of the MMTP until consistent results which agreed with the theory were obtained. The original MMTP, shown in Fig. 3-8 included two subbases, one for the driving DC motor and a separate plate for the induction motor. The original MMTP also only had a single power supply and no automation, meaning the steady state tests were conducted by manually setting individual voltages and running the MMTP at the desired speeds each for 35 seconds. The RMS of the last 30 seconds of vibration for each speed was taken, divided by speed squared and plotted against speed to produce a steady state FRF. These early steady state tests of the MMTP did not show a clear resonant peak, which led to a belief that the multiple mounting surfaces were causing the true resonance to be masked in some way.

Although this demonstrated an interesting phenomenon, it was desired to have

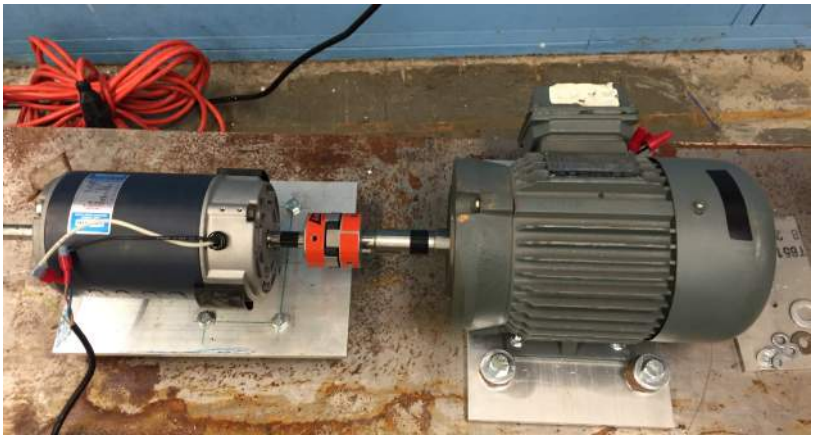


Figure 3-8: Original MMTP with Multiple Subbases

only a single resonant peak so as to replicate a field generator as best as possible. Studying the effects of multiple mounting plates on a coupled machine's resonance is an area for future study. In order to eliminate the multiple peaks, a single subbase system was designed and built, as described in the previous section. The first mounts installed were 50A durometer vibration dampening sandwich mounts with a 5 cm diameter and 5 cm height. A steady state test was run on this configuration from speeds 5-60 Hz, and a clear resonance was identified at 28.5 Hz. Fig. 3-9 displays the steady state RMS vs frequency plot for this configuration.

The presence of a distinct resonant peak in the plot indicated that the MMTP was robust enough to serve as the base setup for future experiments with minor changes to the setup and method. The original 50A durometer 5 cm diameter mounts were substituted by the 5 types of mounts described in the previous section in order to control every part of the mounting except for the durometer. An automated system to control the DC power supply and allow steady state FRFs to be generated easier and more accurately was also incorporated. This automated system included a method to eliminate the rapid increase in vibration magnitude at low frequencies. FRFs were generated for 10 MMTP configurations: a baseline with each of the 5 mounts with no imbalance and an identical set of tests with a 17g imbalance placed on the rotor.

The automated steady state Python script interfaces directly with the VIB-P and MMTP via a dedicated Linux workstation. The script can be used to command these

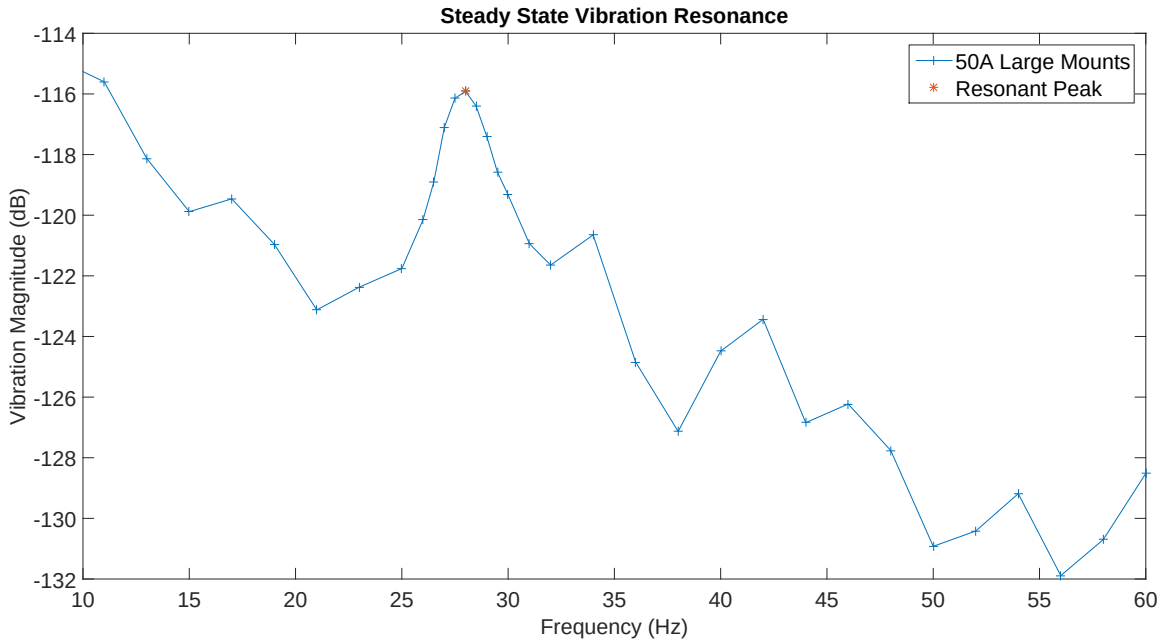


Figure 3-9: MMTP with Single Subbase Steady State RMS Response

speeds, measure the vibration, and create the FRF. After the voltage is commanded and the motor begins spinning, the script uses a shaft encoder to determine the speed of the motor. After the next voltage is commanded, the vibration data is read from the previous speed using *ethstream* and stored in a temporary file. The first 3 seconds of data is removed to account for the system settling, and the remaining data is run through a function to determine Welch's power spectral density estimate [11]. This returns a steady state fast Fourier transform (FFT) for the period of vibration data. The peak of the power spectral density at a frequency closest to but greater than the measured speed of the motor is taken as the vibration magnitude at the measured speed. Fig. 3-10 shows the steady state FFT at an individual frequency, as well as the measured speed and point estimate for vibration magnitude. This magnitude is divided by the speed squared and plotted to produce the FRF, shown in Fig. 3-11. The script controls the MMTP by commanding voltages at certain intervals and time durations as specified by the user. The voltage commanded is the one output by the first power supply, which must be converted from a 0-60 V voltage to a 0-5 V voltage to drive the large power supply, which will deliver to the motor a voltage

between 0-200V. The factor between voltage commanded to the small power supply and voltage driven to the motor is 3.33. The motor spins at 60 Hz around 86.6 V, meaning the maximum voltage set in the script is 26 V. The minimum voltage was typically set to 5 V, which corresponds to about 11.5 Hz. This is the minimum speed frequency where useful vibration data was observed. The user can also specify the time to hold at each voltage, in addition to a built in 5 seconds for each step. A 4 second user specified hold time, for a total of 9 seconds at each voltage, was deemed adequate. The full script, written by John Donnel, can be found in B.

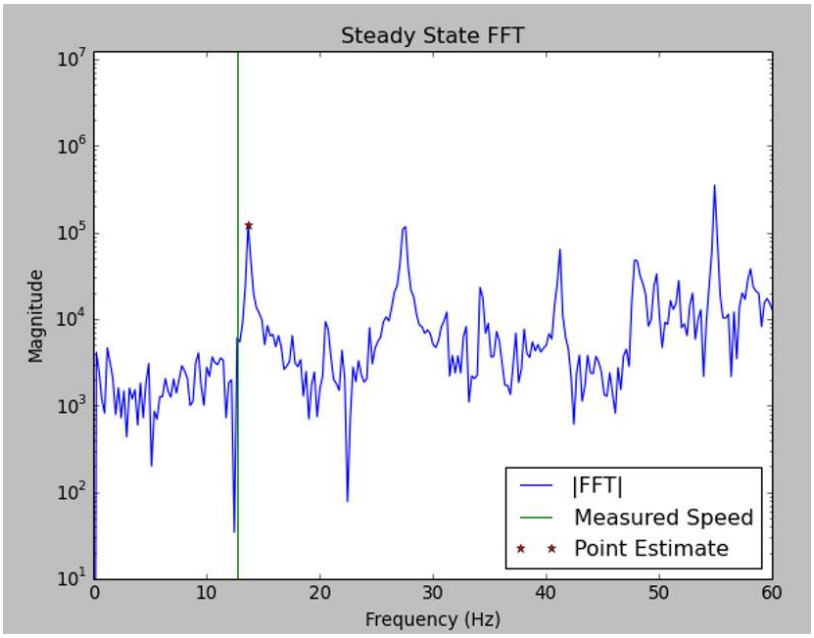


Figure 3-10: FFT at Individual Frequency

The results from the 10 MMTP configurations on which the automated steady state script was run can be seen in Fig. 3-12. The resonant peaks moved in a matter that was expected. The mount with the lowest resonant frequency was the 30A durometer, and the resonant frequency rose as the durometer rose. The magnitude of the resonant peak also rose when an imbalance was imposed without a significant shift in resonant frequency compared to the baseline condition with the same mount. The imbalanced cases actually produced clearer FRFs than the baseline cases, which makes sense since there is a better signal to noise ratio with higher vibration levels. The baseline cases often had multiple humps in the data near the resonant frequency, while

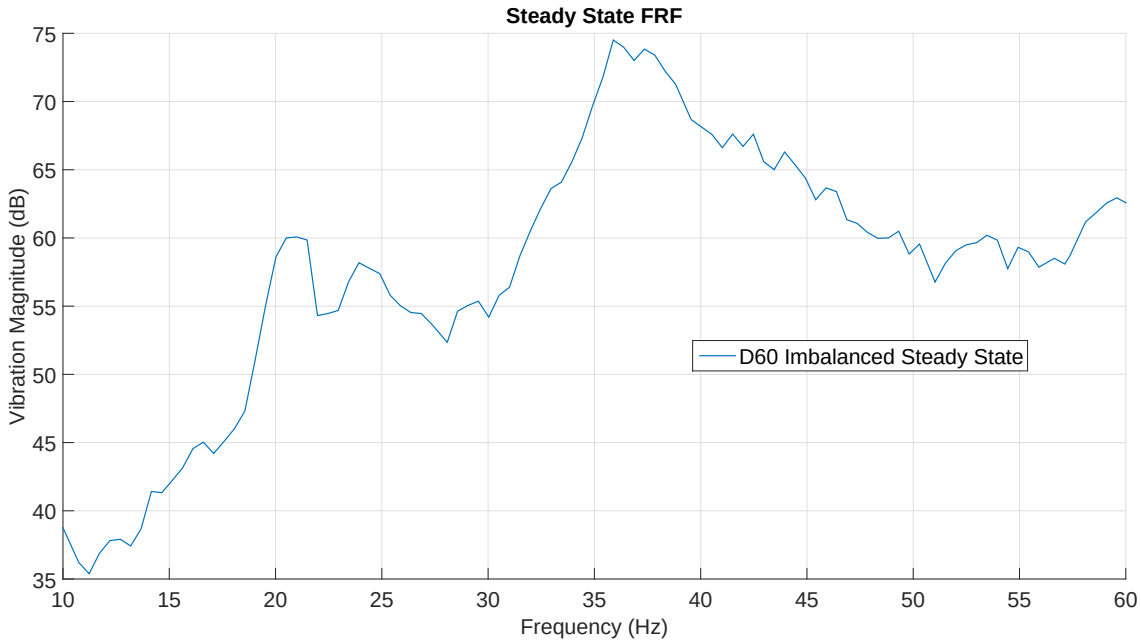


Figure 3-11: Steady State Frequency Response Function

the imbalanced cases always had very clear resonant peaks. The primary purpose of the steady state FRF generation tests was to ensure the mount frequencies changed as expected and that the estimated FRFs, or VTFs derived from the spindowns produced similar results. The first part of this is shown in Fig. 3-12, and the second part will be investigated in the following section.

3.2.2 Spin-down Tests

Testing the impacts of different mounts and imbalances using spin-downs and VTFs was the primary goal of the MMTP since it most closely matched the tests conducted on the field motors and generators. The MMTP configurations run through the spin-down tests were the exact same as those run through the steady state tests. Spin-down tests were initially conducted by bringing the DC motor to 3600 RPM by referencing the encoder reading and adjusting the power supply voltage. After a stabilization period of at least 5 seconds, the large DC power supply was turned off at the power switch, so there was no power provided to the motor and it could spin down under open-loop conditions. After 5 seconds in an "off" condition, the power supply was

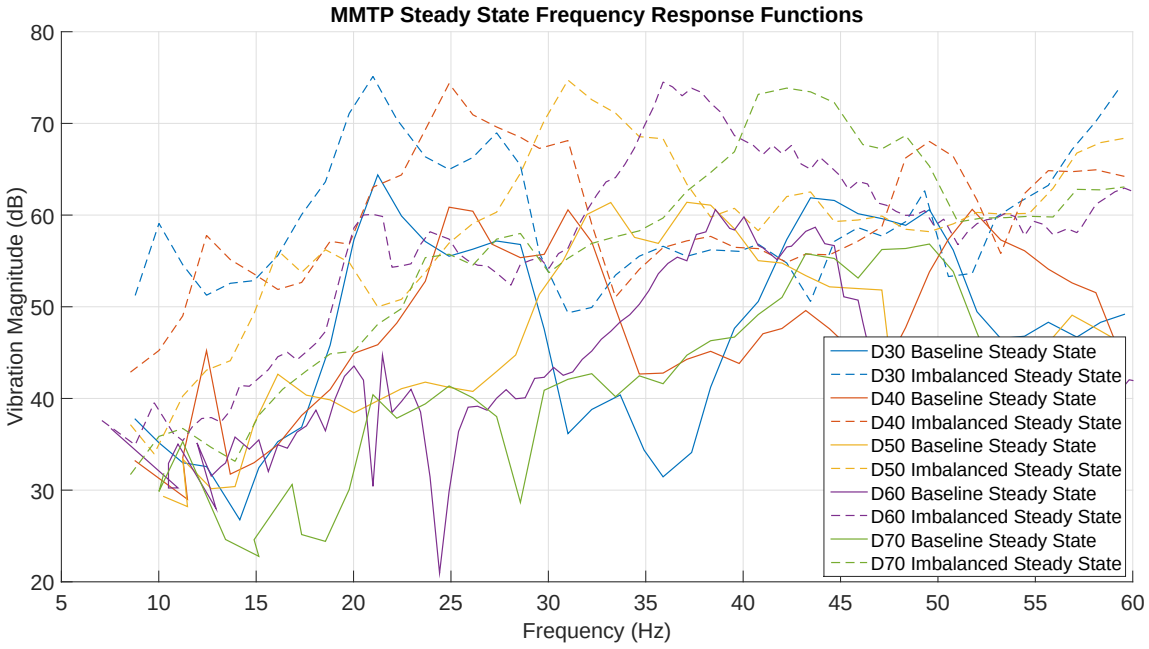


Figure 3-12: Steady State FRFs for 10 MMTP Configurations

restarted and the motor brought to 3600 RPM again for a second run. A total of 4 spin-own runs were conducted for each mount/imbalance configuration. A function was eventually added to the automated steady state FRF generator which included a spin-down analysis portion. See Appendix B for detailed test procedures.

The spin-down tests on the MMTP produced VTFs which generally agreed with predictions. As the durometer of the mount increased, so did the resonant frequencies on the VTF. Table 3.2 contains the resonant frequencies and peak amplitudes. The resonant frequencies on the VTFs did not match the predicted frequencies derived based on mount characteristics. One of the possible reasons for this is the MMTP has several degrees of freedom. Although vibration is only being measured in the vertical (z) direction, the system is actually moving in the x and y direction, as well as pitching, rolling and yawing. Since the MMTP is moving in so many directions it does not agree with a perfect eccentric spring mass damper theoretical system. Nevertheless as long as the sensor is at the same point and axis, comparisons can drawn which provide valuable information about the system.

An increase in vibration magnitude at resonance was seen when an imbalance

	Predicted Freq. (Hz)	Baseline Freq. (Hz)	Imbalanced Freq. (Hz)	Baseline Mag. (dB)	Imbalanced Mag. (dB)
30A	15.5	22.2	21.8	-6.1	5.8
40A	18.9	26.1	25.6	-4.8	7.14
50A	22.8	32.7	31.2	-5.7	6.3
60A	27.6	38.5	36.4	-6.4	5.7
70A	34.2	44.3	41.4	-4.5	7.3

Table 3.2: Predicted and Actual Resonant Frequencies and Magnitudes

was applied for each mount. As the durometer became lower, the imbalance also tended to shift the resonant frequency even lower. Fig. 3-13 shows the VTFs of the 10 MMTP configuration tests. Each configuration has all four spin-downs plotted in order to see the grouping within the same configuration. During the majority of each spin-down, and most importantly during the time of resonance, the VTFs for the same configuration are nearly identical, which inspires confidence in the method and the control of the experiment.

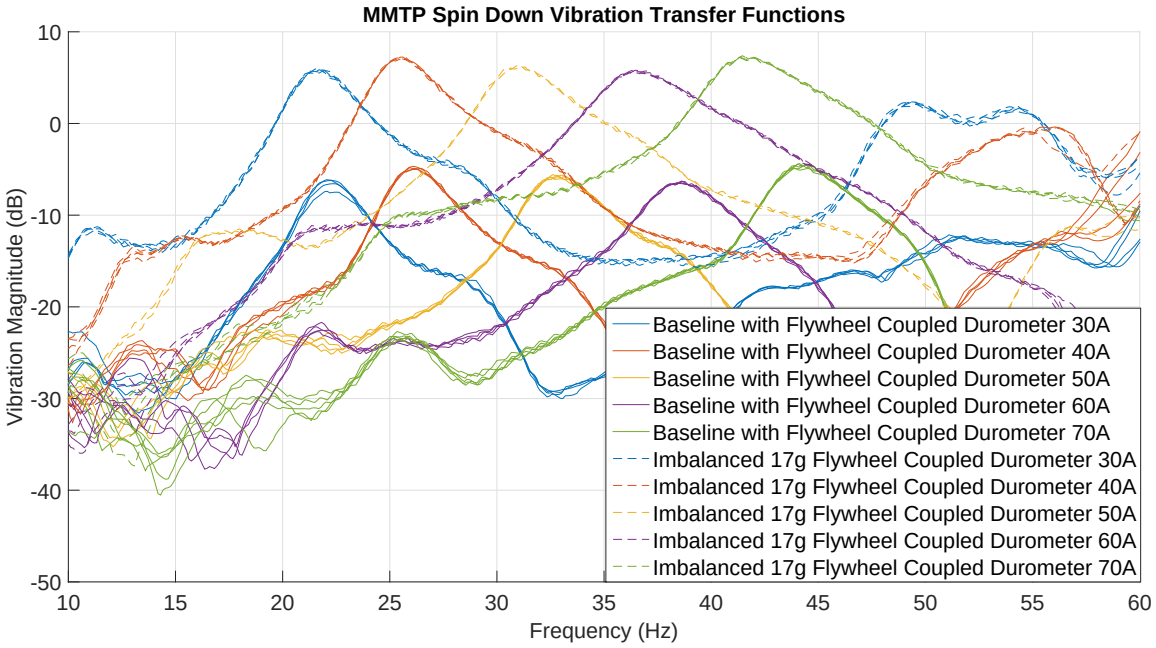


Figure 3-13: VTFs from 10 MMTP Configurations

The VTFs produced using the spin-down method were also compared to the FRFs produced in steady state tests. Fig. 3-14 shows an example of a steady state FRF

plotted over a spin-down VTF. For this test the motor was run in steady state at 105 frequencies. Although they are not exact copies of each other, the curves are very similar and show virtually the same resonant peak. The spin-down VTF may appear much smoother because the virtual input spans more frequencies than the steady state FRF. It also has an envelope taken of the response's FFT while only the pure response is plotted for the FRF. For clarity this is the only comparison shown, however the FRFs and VTFs for each of the 10 configurations all matched in similar ways.

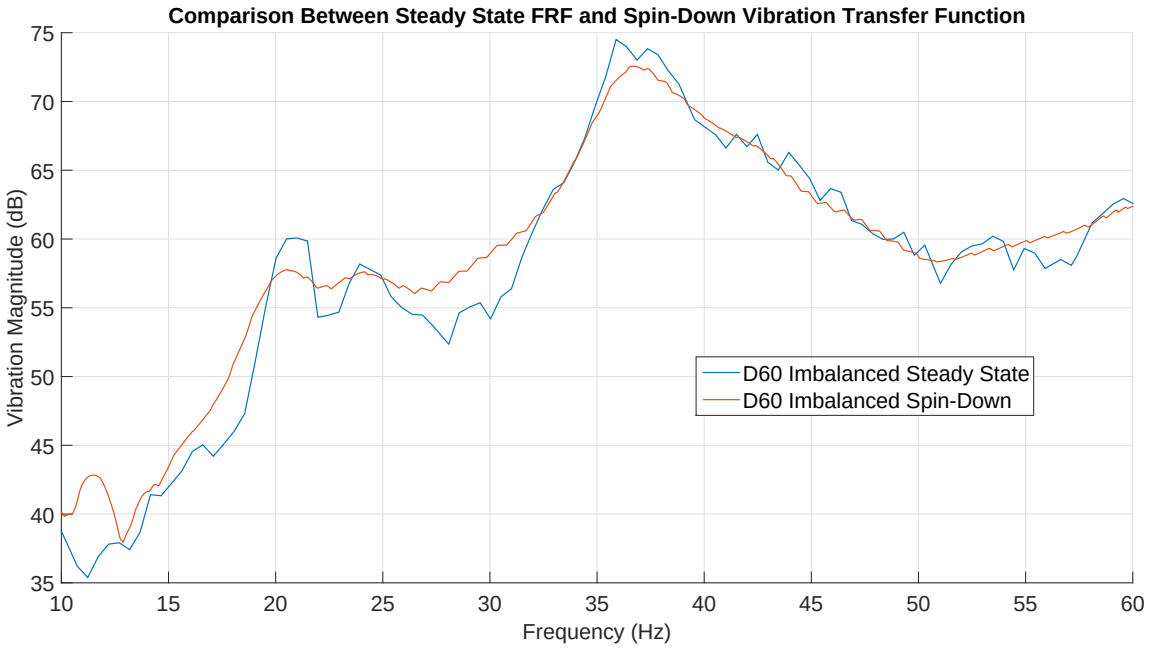


Figure 3-14: Steady State FRF vs. Spin-Down VTF for D60 Imbalanced Configuration

The MMTP was also used to test different sensor configurations. Although the MMTP is equipped with a mounting stud for the accelerometer, field tests on the MCM used a mounting block with a Command^{TM} strip. In order to ensure that the mounting strip did not affect the measurements, a second accelerometer was mounted to the subbase using the mounting block to see if it had any effect. A comparison of FRFs showed no difference in the resonant peak, confirming that the mounting block was an acceptable way to temporarily mount the accelerometers to a machine.

Fig. 3-15 shows the VTFs from the two accelerometers in spin-down tests with 30A durometer mounts.

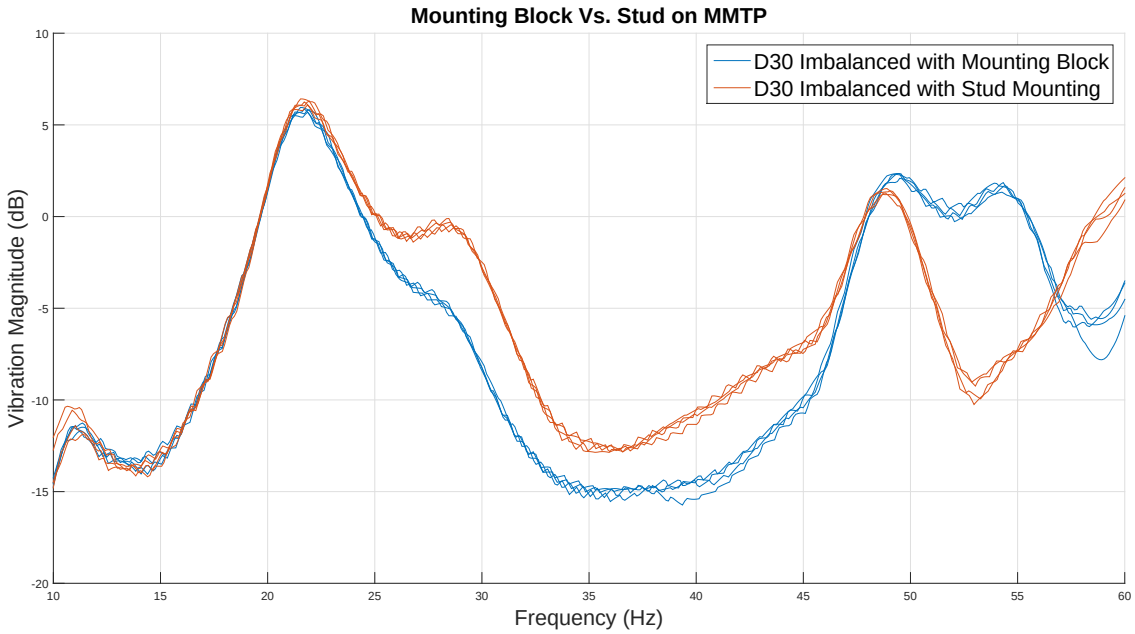


Figure 3-15: Mounting Method Comparison for D30 Imbalanced FRFs

Tests were also conducted to see the effect of different sensor orientations on the VTF. For these cases the x-direction can be thought of as being the long dimension and going lengthwise along the MMTP subbase. The y-direction covers the width, or short portion, of the subbase. The z-direction is the direction used for every test up until this point, and comes out of the subbase. Fig 3-16 shows VTFs for spin-downs with each sensor orientation. When the sensor is in the y-direction the resonant peak appears to shift to the left significantly. When it is in the x-direction there is not a clear resonant peak. Since the results were so consistent with the normal mounting orientation it was determined it was acceptable to use this for all other tests. Nevertheless the vastly different VTFs exposed just how many degrees of freedom are inherent in the MMTP. Since the vibration reducing sandwich mounts are flexible, the MMTP is actually free to move in 6 directions, as mentioned previously. Understanding each of the motions of the MMTP and their impact on VTFs is an area for future study.

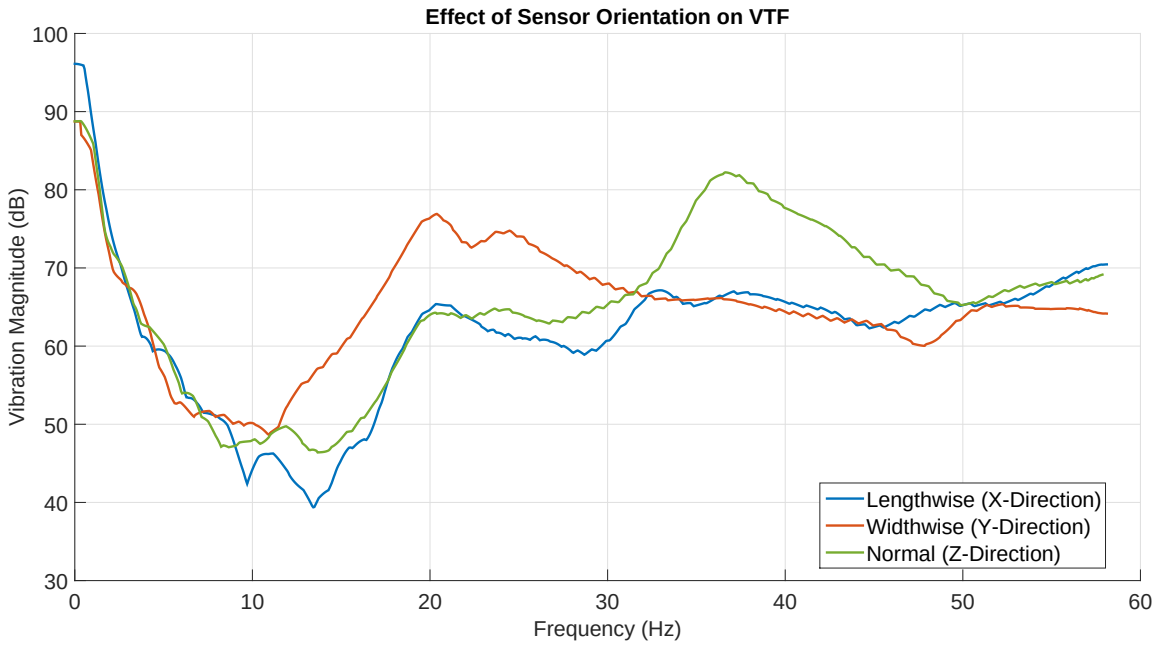


Figure 3-16: Comparison of VTFs for Different Sensor Orientations on MMTP with D60 Mounts

The MMTP proved an invaluable test setup for developing, testing and verifying the method used to construct VTFs. Seeing that the VTFs matched the steady state FRFs, and that the resonant peak moved as expected was enough to prove that the method to construct VTFs was suitable to run with other tests on non-purpose built platforms. There is still much work that can be done with the MMTP. The effect of a single bad mount, different subbase balancing and external vibrations below the subbase are just a few possible tests which may lead to interesting results. The MMTP can also be used to integrate a smaller sensor than those included with the VIB-P, including those envisioned by the vibration assessment monitoring point with integrated recovery of energy (VAMPIRE) project.

3.3 Lab Generator Tests

A pair of two 5 KW generators used for DC electric drive research in LEES were used for concept testing the VIB-P. These were the first generators on which the back EMF sensor was installed, and were an important development tool for the VTF

generation algorithm. Each generator is driven by a belt from 2 DC motors coupled in parallel. There are no vibration dampening mounts; the generator and motors are mounted directly to a steel strut, which is bolted to a bench. The two generators are identical and during normal operation are braced to each other with an aluminum bar connecting to the top of each. Two accelerometers were mounted 90° to each other directly on the generator being tested. The z sensor is vertical (coming out of the bench) and the y sensor is horizontal (lengthwise along the bench). The back EMF sensor is zip tied onto the 3 phase lines residing in the terminal box on top of the generator. Fig. 3-17 shows the test setup and sensor placement on 1 of the 2 generators.

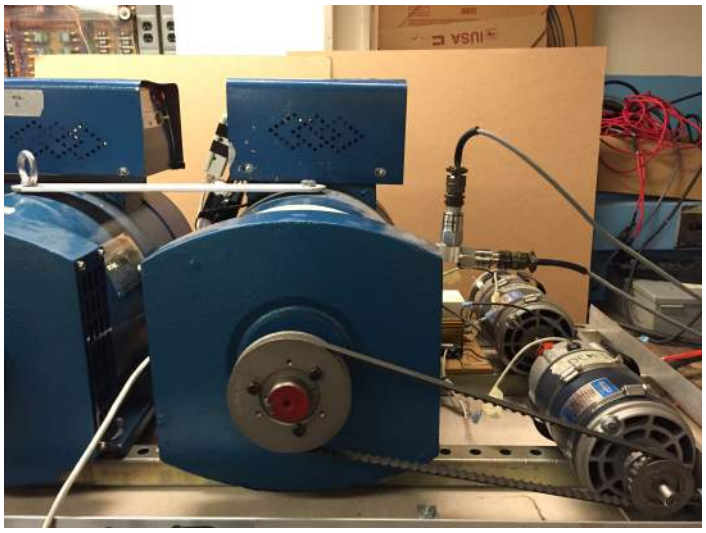


Figure 3-17: Sensor Mounting on Lab Generators

There were two periods of testing on the lab generators, conducted on separate days. For the first day, each generator was run through 3 spin-downs in each of 3 configurations, for a total of 9 spin-downs per generator. The first configuration was a baseline, where the generator was unaltered from its normal state of operation. The second configuration involved loosening a single mounting bolt, and the third had a weight added to the outside of the generator's belt drive, creating an imbalance on the rotor (the loose bolt was tightened before this test commenced). The system which runs the generator setup operates by first bringing the generator up to its steady state speed with the field current applied. The operator then turns off the field current

in order to simulate the generator being in a cool down state before finally cutting power to the DC motors. The spin-down lasts approximately 11 seconds.

The back EMF and vibration data was run through the same VTF generation method that was used for the MMTP, with the speed extracted from the voltage difference using the Hilbert transform. Results for individual generators in the first day of testing showed little difference in VTFs between the configurations. All spin-downs with the x-axis sensor had a characteristic resonance at approximately 8.3 Hz. There was a clear difference between the magnitudes at resonance between the 2 generators for each configuration, with generator 2's being higher. There was not a significant change in resonant frequency or magnitude as the bolts were loosened or the imbalance added. Since the resonant frequency is consistent between the two generators and only the magnitude changes, the VTF predicts a possible greater imbalance in generator 2. Fig. 3-18 shows the combined VTFs of the lab generators from the first day of testing.

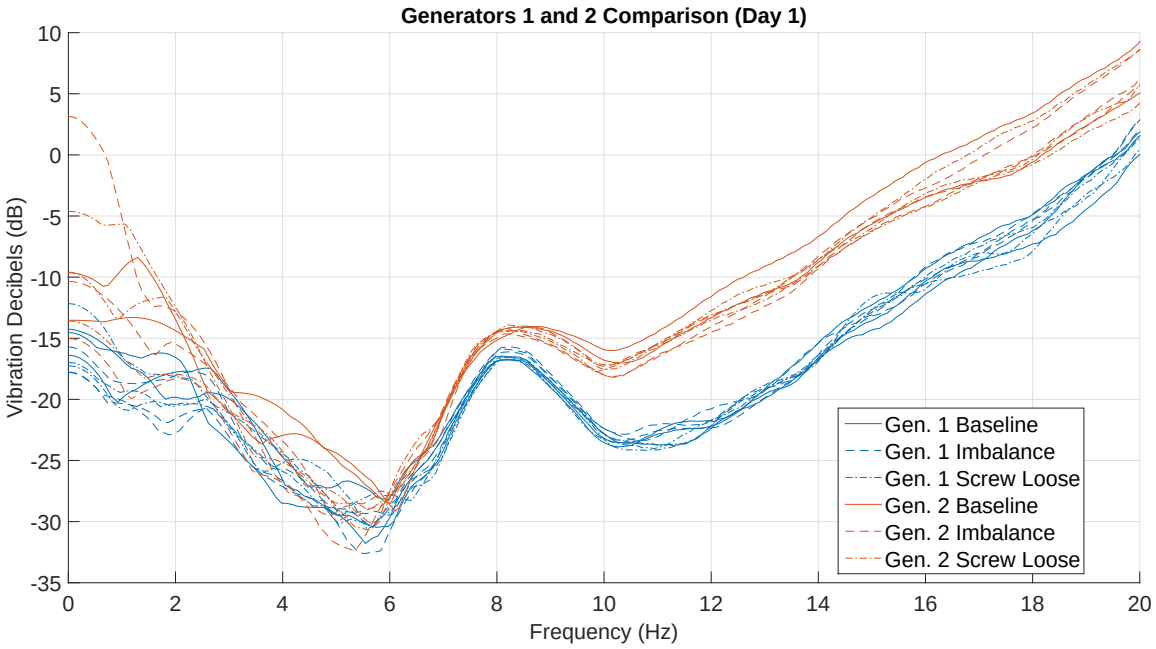


Figure 3-18: Day 1 of Testing on Lab Generators

The goal of the second day was to force the VTFs within generator data sets to move through more extreme changes due to alterations in mounting and an imbalance.

Since the goal was only to see the effects in the context of one machine, only generator 1 was tested. Instead of only loosening 1 bolt like the first day, all 4 mounting bolts were loosened. The weight of the imbalance was also increased significantly. Finally, the stabilizing bar was removed for all configurations in order to to give the generator more freedom to move in the destabilizing cases. For day 2 testing, the 5 configurations tested were baseline, 4 bolts loose, heavy imbalance, and baseline with bar reattached. Results from these tests can be seen in Fig. 3-19. All cases have nearly the same resonant frequency, however the imbalanced case has a magnitude which is clearly higher, supporting the theory that an imbalance raises the vibration magnitude at resonance, but doesn't shift the peak. The experiment with 4 loose bolts did not shift the resonant frequency, however they did increase the magnitude very slightly. In the baseline with bar experiment, the vibration levels during higher frequencies were much lower. This makes sense because the stabilizing bar is likely in place to reduce vibration when the machine is running at steady state.

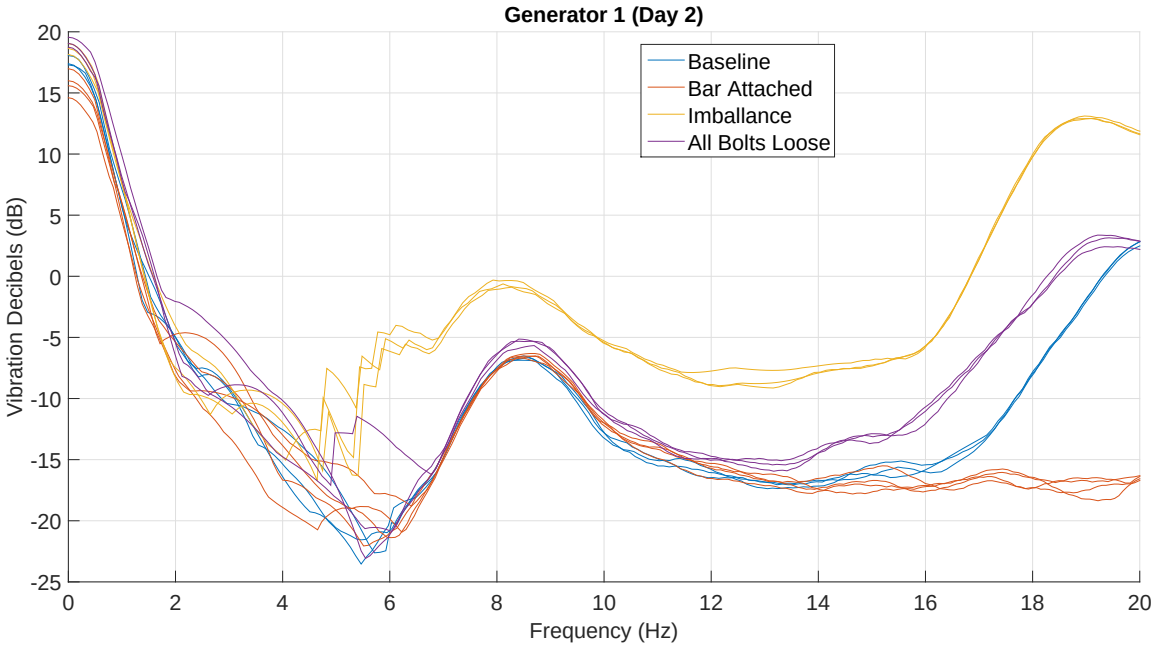


Figure 3-19: Day 2 of Testing on Lab Generators

All previous VTFs were generated using vibration data from the vertical accelerometer only. One of the goals of the lab generator tests was to see if there

were any differences in VTFs if the sensor orientation was different. The VTFs for day 2's tests were regenerated using the horizontal accelerometer's data and overlaid onto the VTFs from the vertical sensor, as can be seen in Fig. 3-20. There was no discernible resonance when the horizontal accelerometer was used, reinforcing the fact that the vertical sensor was best for future applications.

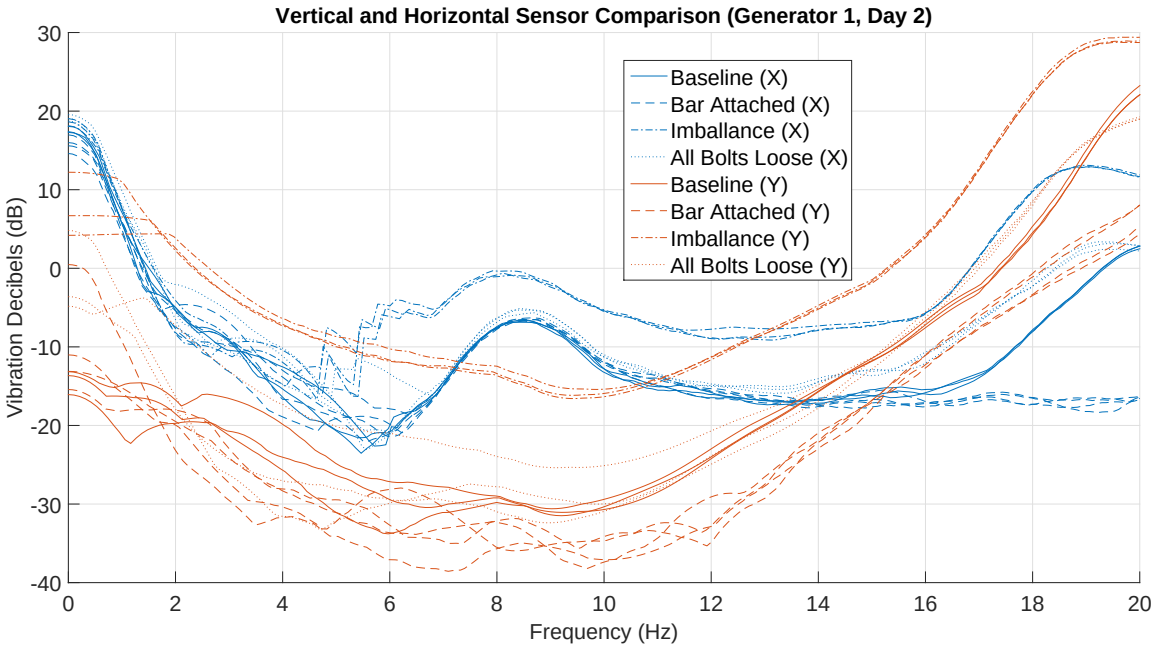


Figure 3-20: Impact of Sensor Axis on FRFs

The lab generator testing had two primary goals: to serve as a systems integration test for the VIB-P and to assist in the development of the VTF generation algorithm. The first goal was fully achieved as these tests were the first of the fully functional VIB-P. The back EMF sensor proved it could pick up the voltage from a generator during spin-down and reliably estimate the speed. The second goal was partially achieved. Although reliable VTFs were eventually generated from the data, there was no shift in resonant frequency observed in any of the tests. Since the generator was hard mounted, it was not possible to alter the mounting to force a shift other than loosening the screws, which did not seem to have a significant effect.

3.4 Fort Devens Generators

Following successful tests on the lab generator, the VIB-P was brought to the BCIL for testing on their field generators. The BCIL is a fully functional, 150 person deployable base camp used by the US Army for training and to test new technologies. Its electrical distribution system can be configured to run off of a micro-grid powered by 18 low noise, field portable generators, each rated at 60 KW. Each generator is driven at 1800 RPM by a 6 cylinder diesel engine. The generators and diesel engines are each mounted on a skid and enclosed in a case to protect the components and allow easy movement. A total of 9 Devens generators were run through a series of spin-downs. A fully instrumented Devens generator can be seen in Fig. 3-21. The back EMF sensor leads were zip tied onto the phase lines on the left side of the assembly, and the accelerometers were placed in various areas around the prime mover on the right side. All tested generators were identical and there were no known issues reported to the test team.

Figure 3-21: Fully Instrumented Generator at Fort Devens

Each generator was run through 3-7 spin-downs. For each spin-down, one of the accelerometers was placed on the mounting plate directly attached to the diesel

engine and above a rubber vibration reducing mount, as seen in Fig. 3-22. This allowed the generators to be compared to each other and verify the consistency of VTF generation. The other vibration sensor was placed in various locations to allow comparison between different areas of the generator set and different orientations. The other areas included the enclosure base, forward mounting plate, and top of the diesel engine.

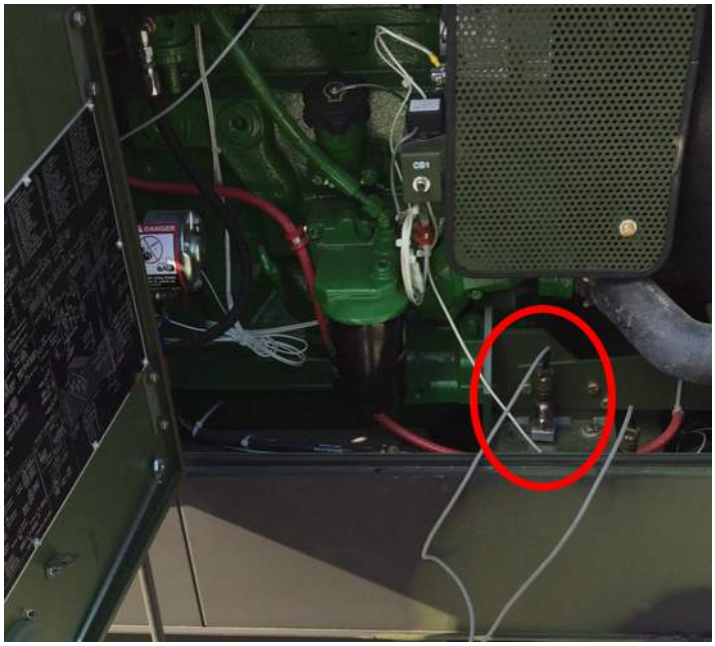


Figure 3-22: Sensor Placement on Devens Generator

The Devens generators have a very short spin-down time, roughly only 2 seconds. The data was first run through the original VTF code which uses a Hilbert transform to determine the spin-down speeds. However the transform fit from the back EMF data had significant ripple, and was unsuitable for use in the rest of the VTF generation algorithm. The zero crossing method, which was developed for the MCM generator spin-downs, was used instead since it was able to produce a smoother, more realistic curve.

The grouping of VTFs for spin-downs on each individual generator was very consistent. Fig. 3-23 shows the VTFs generated for the first 3 spin-downs of each generator using the primary sensor, which was above the mounting plate for the diesel generator. 6 of the 9 generators had very similar VTFs; even though there did not appear

to be a clear resonace the curves all leveled out around 17 Hz. Each generator had a vastly different VTF above this frequency, with the magnitudes near the steady state operating speed spanning a range of over 30 dB. The VTF for Generator 14 had the most significant deviation from the other VTFs and was shifted to the left and elevated. Since this generator presented the most interesting case, it was investigated further.

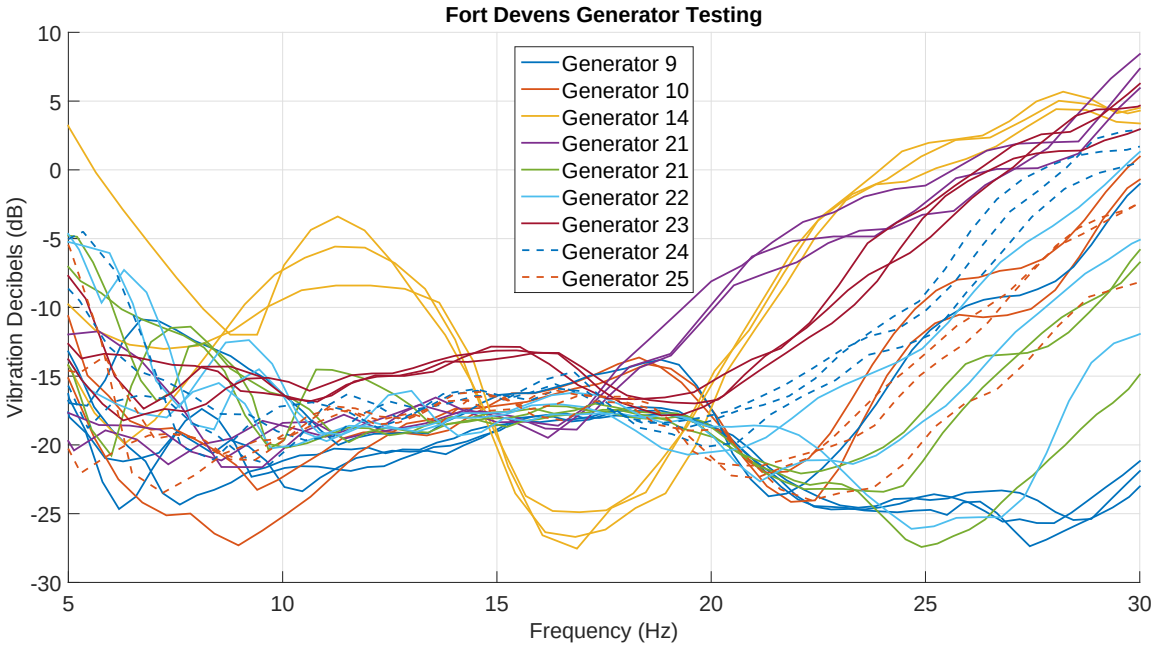


Figure 3-23: Devens Generators Spin-down FRFs

One theory for such a shift was that the mounts might not have been tightened to the proper level. Even though tests on the lab generator did not show a significant shift when the mounts were loosened, this possibility was still investigated since the Devens generators have actual vibration reducing mounts. To test the effect of loose bolts, the 8 bolts connecting the MMTP to it's mounts were loosened. A comparison of the VTFs between the baseline condition and the loose bolt condition can be seen in Fig. 3-24. There was no noticeable shift in resonant frequency between the baseline and loose bolts case, and the magnitudes were identical, making the possibility of a loose mount causing such a large change in resonance on the Devens generator unlikely.

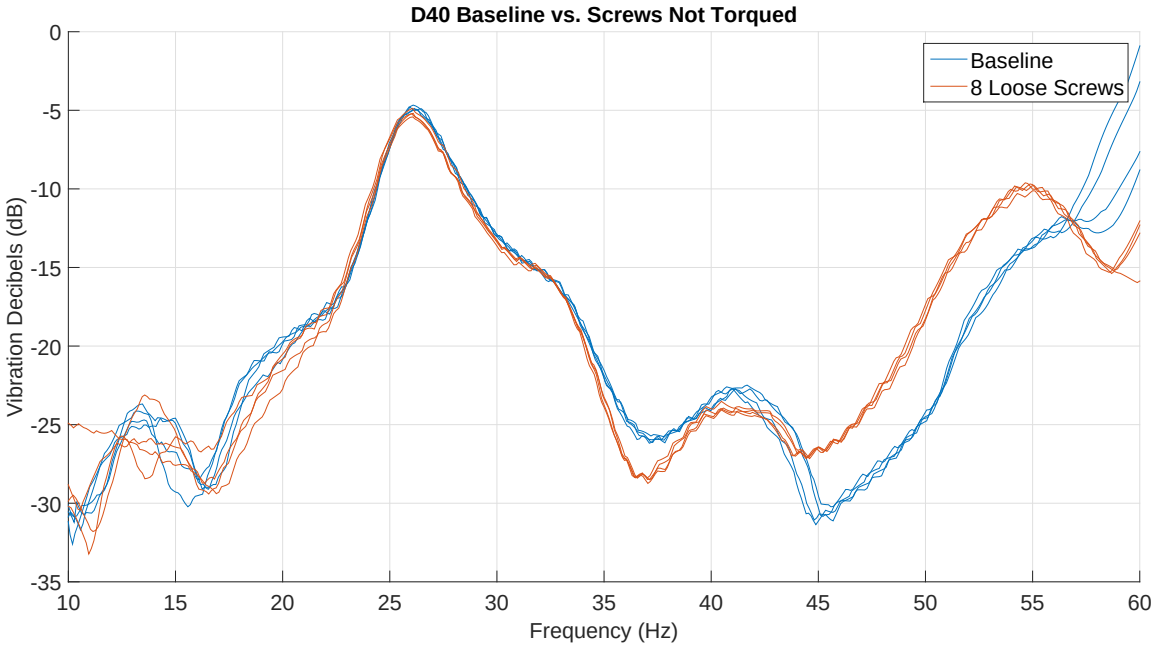


Figure 3-24: Impact of Loosening 8 Bolts on MMTP

The primary goal of the tests on the Devens generators was to test the VIB-P and VTF generation algorithm on a machine in the field. These were the first machines tested that were not in the lab, and since there were 9 available for testing the team was able to see how consistent the machines were with each other. The fact that 2/3 of the machines had identical VTFs validated the test system and associated VTF generation code. One unique aspect of the Devens generators compared to the other machines tested was the extremely short spin-down time. It is unknown if the Devens generators operate on a closed-loop system, where the spin-down is artificially braked, or an open-loop which would allow a natural spin-down. This may partially explain why the VTFs were so different at high frequencies and why there lacked a clear resonance. Investigating the effect of this shorter spin-down on the VTF generation is an area for future study.

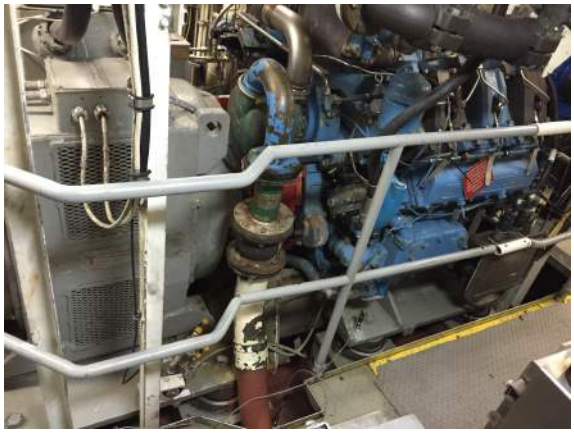
3.5 MCM Tests

Testing generators on in-service MCM generators was the primary motivation for developing the VIB-P and VTF generation method. Several spin-down tests were conducted on the USS CHAMPION (MCM 4) between 14-22 February 2015. CHAMPION is a 68 m, 1312 t *Avenger* class mine countermeasures ship commissioned in 1991. The ship is designed to hunt mines with sonar and remotely operated vehicles as well as sweep for mines with an influence sweep cable. Since the ships are designed to operate in a minefield, there are several design characteristics which make them unique to other navy ships. Specifically, Douglas fir coated with glass-reinforced plastic is used for the hull in order to lower the magnetic signature. The machinery is also designed to have low magnetic and acoustic signatures. The main engines and SSDGs are made of non-ferrous materials and have resilient mounts which reduce vibration and transmitted noise. CHAMPION is depicted in Fig. 3-25.

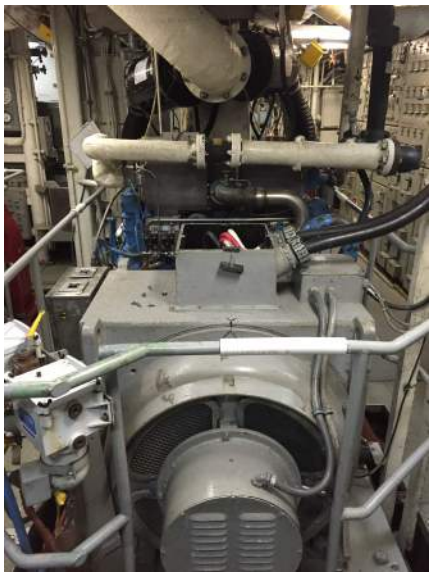
Figure 3-25: USS CHAMPION (MCM 4)

The SSDG is a 375 KW, 450 Volt, 601 Amp, 3-Phase, 60 Hz, 1800 RPM generator set. The prime mover is an Isotta-Fraschini 6-cylinder diesel engine. There are 3 SSDGs on each MCM; 1A and 1B are in a forward auxiliary machinery room (AMR) to the port and starboard of centerline. SSDG 2 is centerline in the main machinery room (MMR), aft of the AMR. There are 2 electrical buses, with SSDGs 1A and 1B feeding the forward bus and SSDG 2 feeding the aft. Although the buses can

be connected, the plant is generally operated with SSDG 2 always on-line feeding the aft bus, and either 1A or 1B feeding the forward bus. A side view of a SSDG on CHAMPION can be found in Fig. 3-26a and a front view in Fig. 3-26b. The generators go through a 5 minute unloaded cool down upon being removed from the bus before shutting down. The spin-down on an MCM SSDG is about 9 seconds.



(a) SSDG Side View



(b) SSDG Front View

Figure 3-26: SSDGs on CHAMPION

The accelerometers were placed in identical locations on each of the three generators during most of their respective tests. One sensor was placed on the subbase directly above one of the mounts near the forward portion of the diesel engine on the port side. The other was placed on the ship structure next to the mount on which the first sensor was placed above. In certain cases, one of the sensors was placed on top of the generator enclosure instead of the ship structure. The back EMF sensor was zip-tied to the phase lines in the terminal box above the generator, as shown previously in Fig. 3-5b. Although the back EMF sensor is mostly non-intrusive, it does require the generator to be "tagged-out", meaning the ship's crew has to mechanically and electrically isolate the machine, before the terminal box can be opened. Consequently multiple back EMF sensors were constructed and brought to the ship so each generator could have a back EMF sensor attached for the entire trip, reducing the disruption

to the ship's crew and increasing chances to capture spin-down data. The voltage picked up from the phase lines during the initial portion of the spin-down exceeded the 5 V limit of the back EMF sensor, meaning the sensor data clipped, rendering the Hilbert transform ineffective for extracting speed. The speed was instead extracted by detecting the zero crossings of the voltage sine wave, as described in Section 2.2.4. The operating speed of the generator is 1800 RPM, meaning the initial speed must be set to 30 Hz.

A total of 14 usable spin-downs were measured on CHAMPION; 5 for SSDG 1A, 5 for SSDG 1B, and 4 for SSDG 2. The first 2 spindowns for each generator took place while the ship was pier-side and on shore power, meaning that none of the generators were being used to power the ship. There was also minimal other auxiliary equipment running in the machinery rooms since the ship was not underway. Each generator was also spun down once more pier-side a few days later, however for each these spin-downs the ship was on ship's power, and the generators had been previously loaded. While underway, SSDG 1A and 1B each had 2 spin-downs measured, and SSDG 2 had 1.

3.5.1 MCM Generator Results

Each spin-down data set was run through the algorithm described in Section 2 to produce a VTF. Fig. 3-27 shows VTFs for each MCM spin-down, with data taken from the sensor above the resilient mount. There is consistent grouping within each generator as well as a clear resonant peak. The generators also have separation between each other, exposing the inherent physical differences between the machines. Generator 1A has a resonance around 18.25 Hz, generator 1B at 19 Hz, and generator 2 at 19.25 Hz. Nevertheless, the aggregate view of all of the generators does not give a completely clear picture of each test. Some were conducted in-port and others were underway, when several other pieces of machinery were running.

A more detailed view of the spin-downs for all runs, which focus only at the region of resonance, can be seen in Fig. 3-28. The first observation that can be made is the grouping of the two in-port spin-downs for each generator. These were the first tests

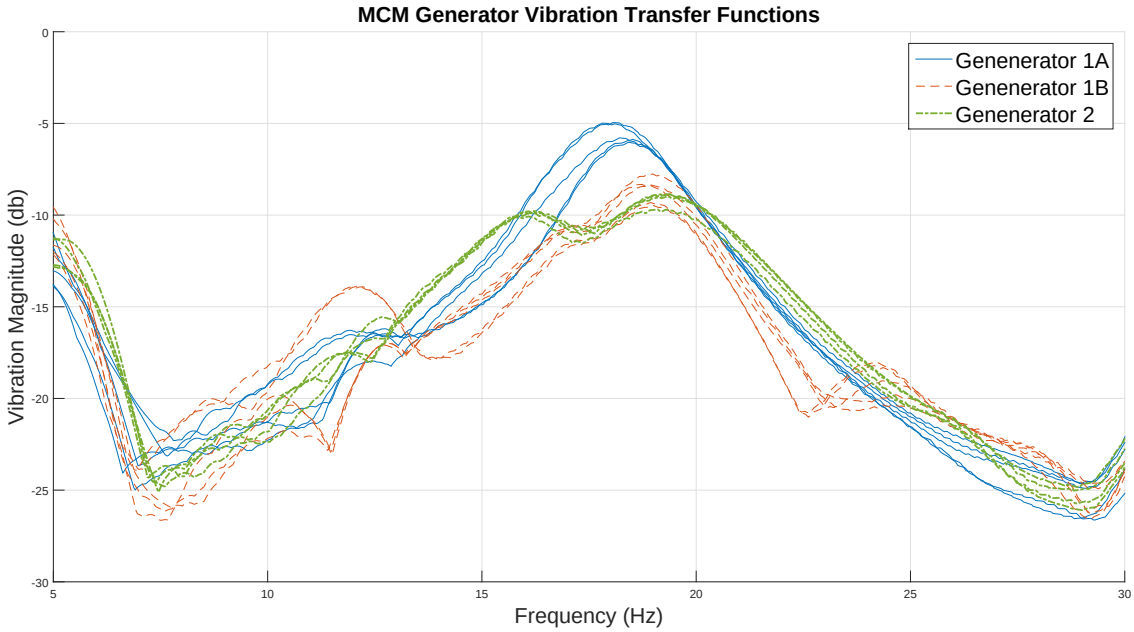


Figure 3-27: MCM Vibration Transfer Functions

conducted, and were done on a Saturday while the ship was in-port and on shore power, meaning there were few crew members on board and very little machinery running. For each spin-down the crew would start the generator, wait approximately 5 minutes and then put the generator in cool down. Since the generators were never loaded there was no difference between the generator being "on" or in cool down. As soon as the first spin-down was complete the process was repeated for a second spin-down. While the generators were being spun down there were no other generators running. In the AMR, where generators 1A and 1B are located, the other machinery running included an air conditioning (AC) plant as well as a few pumps. In the MMR, where generator 2 is located, there was only the periodic running of an air compressor. Although there is some variation between similar generators over the course of all tests, the consistency within each generator on 14FEB is very good. This is important because these were the only tests that were able to be conducted in completely repeatable circumstances. All other generator runs were completed at moments of opportunity, and the ship's condition varied for each spin-down. Two tests in the same condition with identical curves for each generator inspire confidence in the

method used to produce the VTFs and their applicability to real-world generators.

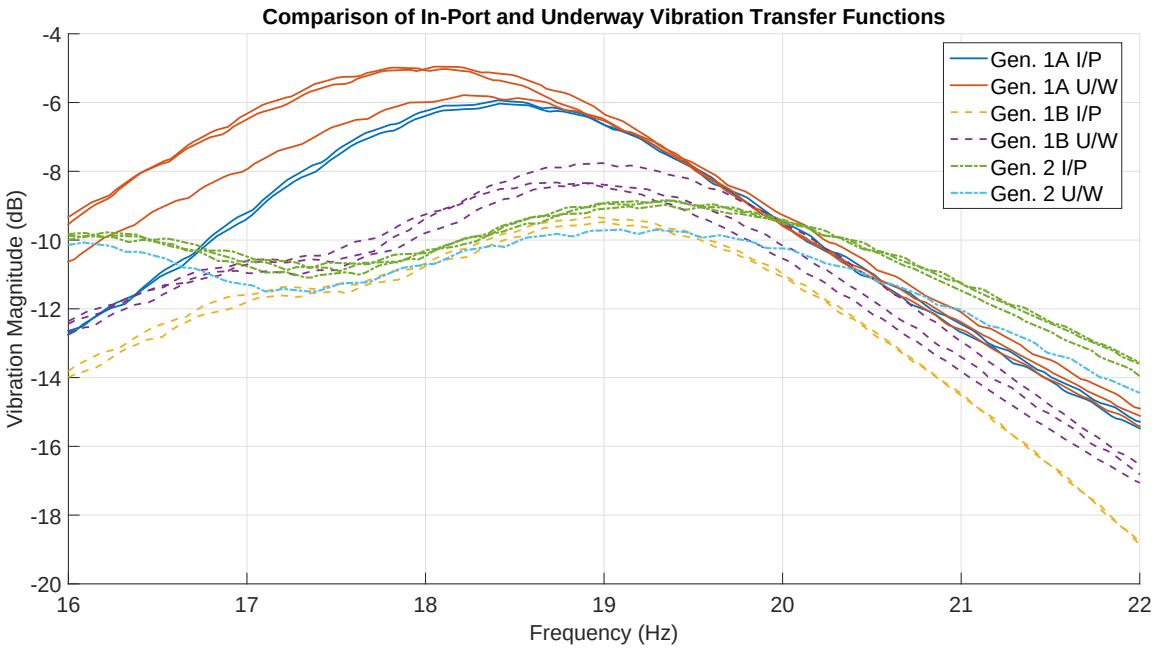


Figure 3-28: Detailed MCM Vibration Transfer Function Resonances

It was predicted that the underway runs would have higher vibration levels than in-port runs, since several additional pieces of equipment, including the main propulsion diesel engines (MPDEs), were also running. This was the case for SSDGs 1A and 1B; each resonance was slightly higher for the underway case. For SSDG 2, the grouping was very consistent for all runs except for the 1 underway runs, which was slightly lower. A more detailed investigation of the steady state data would reveal how much the external vibration effects the vibration levels above the mounts, and whether this has a significant impact on the resonance.

There were 3 spin-downs conducted which can be used to compare mount placement on the generator. Fig. 3-29 gives a comparison between one VTF generated from a sensor placed on top of the generator next to the terminal box, and the other placed directly above a mount. For each case the overall vibration and resonance was reduced. The 2 spin-downs for generator 1B had very similar shapes for both mounting locations; however there was a more significant difference in shape and resonance with the 1A spin-down.

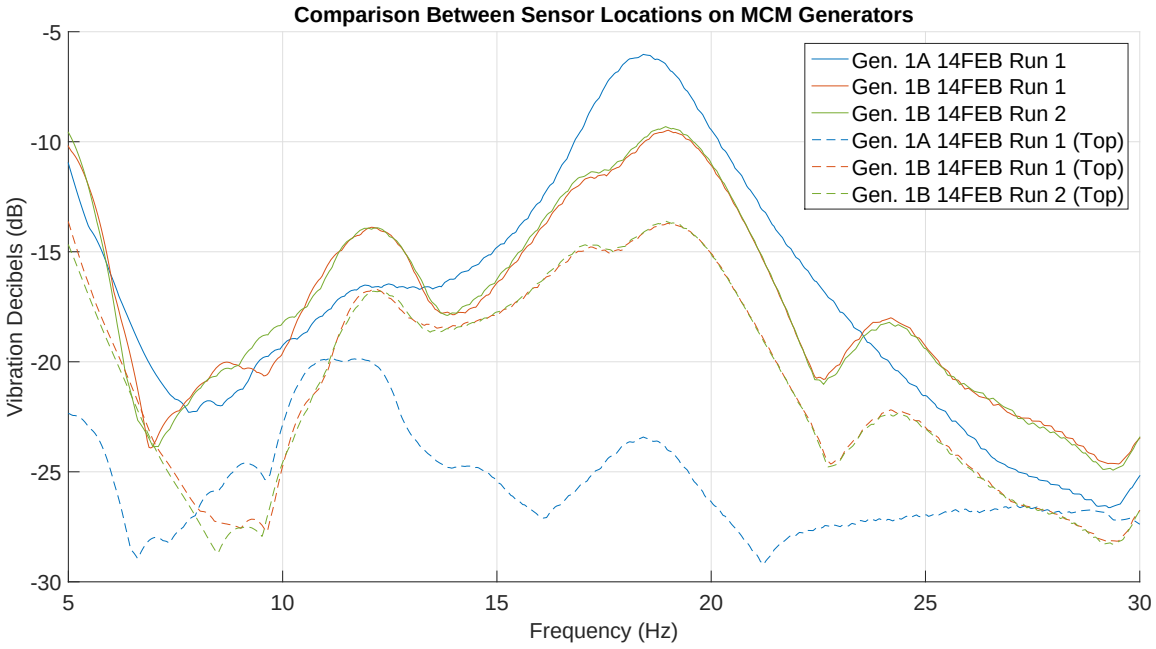


Figure 3-29: Impact of Placing Vibration Sensor on Top of Generator

3.5.2 MCM ASW Pump Results

The MCMs two auxiliary seawater (ASW) pumps were also investigated during the testing period on CHAMPION. The MCM has 2 ASW pumps which operate at 3545 RPM and are driven by a 3-phase AC motor rated at 20 HP. The purpose of the ASW pumps is to move water from the sea through the ASW cooling system, which is used to cool systems such as the AC plants. An ASW pump with the vibration sensors mounted above and below a mount is depicted in Fig. 3-30. The crew had noted that one of the ASW pumps had recently been overhauled. The last overhaul of the other ASW pump was unknown, however there were no known issues with it. Fig. 3-31 shows the resultant VTFs for each of the 8 spin downs.

The VTFs for ASW pumps 1 and 2 are quite different, however both do appear to have a slight peak around 45 Hz. The most interesting part of the ASW pump tests is the dramatic difference between ASW 1 and 2. At the projected resonance of 45 Hz, ASW 2 has a peak about 30 dB lower than ASW 1. The ship's port engineer confirmed that ASW 2 had been overhauled approximately 1 year prior to the test. Studying the change of the VTF for ASW 2 over time would potentially show a trend

Figure 3-30: MCM ASW Pump

corresponding to degradation of the pump system.

The MCM test demonstrated the feasibility of conducting spin-down vibration analysis with the VIB-P. The grouping of multiple runs for each machine was very consistent, and clear trends emerged when comparing machines. Further tests on MCM generators, as well as other pieces of rotating machinery, may give additional insight into the patterns for each machine. Studying the effects of the machines location relative to other vibration producing mechanisms would allow a full characterization of an ship's engine room. Finally, being able to diagnose the health of individual mounts based only on vibrations from a sensor located near the terminal box would allow further integration with VAMPIRE.

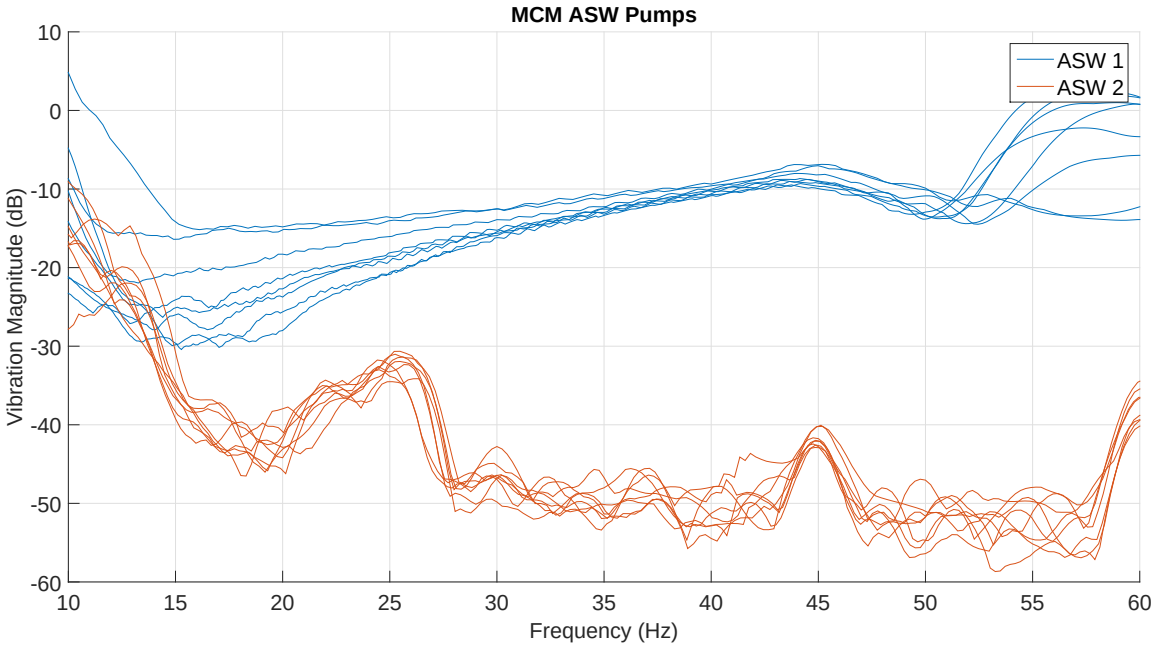


Figure 3-31: Results of 8 Spin-Downs of MCM ASW Pumps

Chapter 4

Thermal Energy Transfer Experiments

Recent proceedings have indicated a need for greater short term energy in military applications [12]. The U.S. Navy is actively conducting research in directed energy weaponry which consumes a large amount of energy for a short duration of time. These weapons include different forms of lasers as well as rail guns [13]. A large capacitor bank is typically necessary to build up the energy necessary to fire the weapon, and the time it takes to recharge the capacitor bank determines the maximum rate of fire. The energy available to charge these capacitors can be thought of as the total generating capacity minus the necessary loads at a battle condition.

Ship loads are often classified as being vital and non-vital. For example, the information systems necessary to run the ship and its weapons are vital, while the laundry and galley are not. Since HVAC systems are needed to keep vital loads working properly, they are also considered vital. Nevertheless, there is an opportunity to use some of the energy normally dedicated to HVAC in a short term situation if the thermal storage of a space is considered. When an HVAC system is turned off, the space does not immediately become unsuitable for equipment or people. Temperature degradation is a product of the thermal mass of the space and the positive and negative inputs which may increase or decrease temperature. Determining the thermal time constants for spaces on a ship can give the system engineer the information they

need to determine how long high energy HVAC loads can be turned off to free up capacity for short term loads. Capacity gained by reducing consumption or increasing efficiency is given a unit of energy known as a negawatt [14]. These negawatts are untapped energy which may be able to reduce the installed power requirement for a ship or enhance the capability of a high energy system.

A series of thermal energy transfer experiments were conducted at the BCIL at Fort Devens. The BCIL was chosen as the test location because they have several of the same type of structure, and the test team was able to secure frequent access since there exist related ongoing research at the facility. Although the tents are much different than a steel ship, they serve as an adequate surragate for basic model development and validation, and the methods for testing could be expanded to ships. Thermal storage is also an interest to the US Army, who operates the BCIL, since it offers the possibility for energy savings by reducing the total number of operating generators at a forward operating base while ensuring generators in use are operating at or near max efficiency.

4.1 Thermal Transfer Model

Thermal time constants are often determined for buildings as a mean of quantifying the thermal mass and efficiency of the structure. In its most basic description, the thermal time constant is the time needed for the structure to reach 63% of the final full temperature change [15]. For example, if a structure had an initial temperature of 40^o C and was cooled with a constant external temperature to 20^o C over a period of time, the thermal time constant would be the amount of time needed to reach 27.4^o C. A commonly used model for temperature change, as described in [16], treats the structure as a lumped mass and gives the change of the building's temperature over time with a constant outside temperature and internal loads as,

$$\ln\left(\frac{\theta}{\theta_i}\right) = -\left(\frac{L}{m_c}\right)t, \tag{4.1}$$

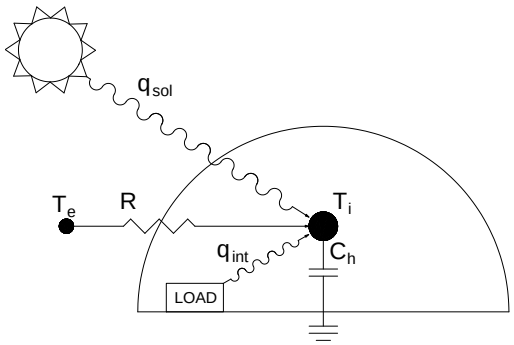
where L is the overall loss coefficient, m_c is the energy mass capacitance, t is the time of test (zero initial time), $\theta = T_i - T_e - G/L$ is the building temperature parameter, $\theta_i = T_{io} - T_e - G/L$ is the initial time building temperature parameter, G is the internal power of loads, T is the inside ambient temperature, T_i is the inside ambient temperature at the beginning of the test (time = 0), and T_e is the external temperature. If the structure is allowed to coast (no efforts to maintain a temperature set point) with a constant external temperature for a period of time then the logarithmic temperature parameter, $\ln(\theta/\theta_i)$, can be plotted against time. The negative reciprocal of its slope will produce a thermal time constant

$$\tau = \left(\frac{m_c}{L}\right) \tag{4.2}$$

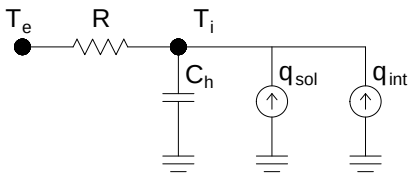
Thermal energy transfer experiments can be conducted in either direction; the structure can be heated or cooled and then coasted depending on the outside temperature. The thermal time constant should be the same in any case. Knowledge of the thermal time constant allows architects and building owners to quantify the thermal mass of a structure, and determine how efficient it can be. However, a major drawback of the above model is the outside temperature must remain constant. It is highly unlikely the temperature will remain constant for the duration of the test, especially if it is conducted during the day. Additionally, although the model does include consideration for internal loads, there is no input for external loads. If the test is to be conducted during the daytime, it is highly likely there will be solar irradiation which will contribute to the temperature change inside the structure.

A new model was developed by Dr. Peter Lindahl which uses state space equations to create a grey box model. A grey box model is appropriate because there is some information known about the system as well as experimental data, however many aspects of what is going on inside the system are unknown. Consequently the model has free parameters which can be estimated using the system identification tools in Matlab. The model can also have as many inputs as desired. Fig. 4-1a depicts a basic visual thermal model for a BCIL tent at Fort Devens and Fig. 4-1b, its elec-

trical equivalent. Known parameters include the external temperature, T_e, internal temperature, T_i, solar heat load, $q_{sol} = k \cdot sol(t)$, and internal heat load, q_{int}. The unknown parameters are the thermal resistance of the tent, R, the heat capacitance of the tent, C_h, and the constant for solar absorptivity, k.



(a) Parameter Schematic



(b) Electrical Equivalent of Tent Model

Figure 4-1: Tent Model Parameters

If system is modeled by the differential equation

$$C_h \frac{dT_i}{dt} = \frac{T_e - T_i}{R} + q_{sol} + q_{int} \tag{4.3}$$

then

$$\frac{dT_i}{dt} = -\frac{1}{C_h R} T_i + \frac{1}{C_h R} T_e + \frac{1}{C_A} q_s + \frac{1}{C_A} q_{int} \tag{4.4}$$

Using state space equations,

$$\dot{x}_t = Ax(t) + Bu(t) + Ke(t) \tag{4.5}$$

$$y(t) = Cx(t) + Du(t) + e(t) \tag{4.6}$$

where $x(t)$ is the state (internal tent temperature), $e(t)$ is an error term, t is time, and

$$u(t) = [T_e(t), P_i(t), Sol(t)]^T \tag{4.7}$$

where $T_e(t)$ is the outside temperature, $P_i(t)$ is the power of the internal load, and $Sol(t)$ is the solar power density. The coefficients in the equations represent the

unknowns in the system, including C_h, R, the thermal time constant, τ, and k. They are

$$A = \frac{-1}{\tau}$$
$$B = [\frac{1}{\tau}, \frac{1}{C_h}, \frac{k}{C_h}]$$
$$C = 1$$
$$D = [0, 0, 0]$$
$$K = 0$$
$$\tau = R \cdot C_h$$

The function inputs to the system are passed through as a 4X1 vector, with

$$par(1) = \frac{1}{\tau}$$
$$par(2) = \frac{1}{C_h}$$
$$par(3) = \frac{k}{C_h}$$
$$par(4) = T_{io}$$

where T_{io} is the initial air temperature in the tent.

Data is brought into the model as an *iddata* set containing all known quantities for the data set, including internal temperature, outside temperature, internal load, and solar irradiation. The *iddata* sets also contain notes regarding the data set, including time, location and any other amplifying information. The data sets are first trimmed to eliminate any unwanted data streams and data points from before and after the experiment started and ended. The data sets also must be detrended prior to being run through the model, so that the floor of the temperature streams is the temperature inside the structure at the beginning of the experiment. Each data stream is prepended with 5 data points; the data points for the temperature streams are given the value of the inside structure temperature at the beginning of

the experiment and all other streams have the data points set to 0.

The data sets are then merged into two sets of data sets, assuming there are multiple experiments. The first set is for training, meaning the data from this set will be used to determine the model fit. The second data set is for cross validation, where the actual data is compared to the fit derived by the training data set. There must be an initial guess for each unknown parameter for the grey box model to work. The initial guess for R can be based on the R-value of the insulation and surface area of the tent. Initial guesses for C_h, τ and k can be based on heuristics and the dimensions of the structure. The system is estimated by passing the training data set to the grey box model using the *greyest* function in Matlab, which fits the data over several iterations while minimizing a cost function. The output is the system estimate, from which the fitted parameters can be extracted using the function *pvec*, which will return a 4X1 vector, pv with fitted values for the input parameters. The estimated thermal time constant, τ, in minutes, is $\frac{pv(1)^{-1}}{60}$, C_h is $pv(2)^{-1}$, and k is $pv(3) \cdot C_h$. The fitted system can be plotted and compared against actual data from the training and cross validation data sets. This model is used with experimental data collected from tests at Fort Devens, with results presented in the following section.

4.2 Fort Devens Tent Tests

The BCIL has 16 HDT AirBeam^{TM} shelters, hereafter known as tents, which are used for housing soldiers during training and conducting energy related experiments. The tents each have 59.5 square meters of floor space and an inside height of 3 meters. They are arranged in pairs with an opening in between to create one long structure. High pressure air is used to support the tent, and they are each set up on a wooden base. Each tent connects to an environmental control unit (ECU), which provides heating and cooling through a duct spanning the length of the tent. The tents also have a removable thermal liner, which is always installed, and a sunshade, which is removed during the winter time. Fig. 4-2 shows a tent pair without the sunshade. The ECU is to the left of the vestibule. In most cases the tents are filled with 10 sets

of military bunks and foot lockers, however there are some tents which may be empty or only have a table and chairs.

Figure 4-2: Fort Devens Tent Pair

HOBO® U12 data loggers with a TMC1-HD external probe were used to collect temperature readings inside the tents. The probes have an accuracy of $\pm$ .25^{o}C, and come in 1' and 6' versions. 4 of the HOBOs used were single channel, and 1 was a 4 channel model which allowed 4 probes to be connected. The HOBOs are set and the data downloaded using software available from `www.onsetcomp.com`. The length of time the HOBO can record is a function of the sampling interval; for the Devens tent experiments the sampling interval was set to 10 seconds. The HOBO's clock is set based on the computers time when the device is launched.

Each HOBO sensor and associated probe was tested in an ESPEC LHU-113 temperature cabinet in order to verify their accuracies. The 5 HOBO sensors and 8 probes were placed in the cabinet as shown in Fig. 4-3. For this test the HOBO sampling interval was set to 1 second. The control temperature was first set to 30^{o}C, where it stayed for 15 minutes after settling, and was finally set to 40^{o}C and left for 25 minutes. Fig. 4-4 shows the temperature reading for each HOBO sensor and associated probe. TL1 through TL4 each have 1 probe, and TL5 has four probes. There is clearly some variation in the response times for each of the sensors, however once they settle on the control temperature the grouping between the sensors is relatively tight, especially as the curve flattens. This is acceptable since the time constant of

the probe is still much faster than the time constants of structure. Since the probes have an accuracy of $\pm$ .25^{o}C, there should be a maximum .5^{o}C difference between any two temperature lines. Except for the very beginning of each control temperature, the sensor readings all fall within this range. It was therefore determined that all of the HOBO sensors and probes were in proper condition and were suitable for field use.

Figure 4-3: HOBO Sensors in Environmental Cabinet

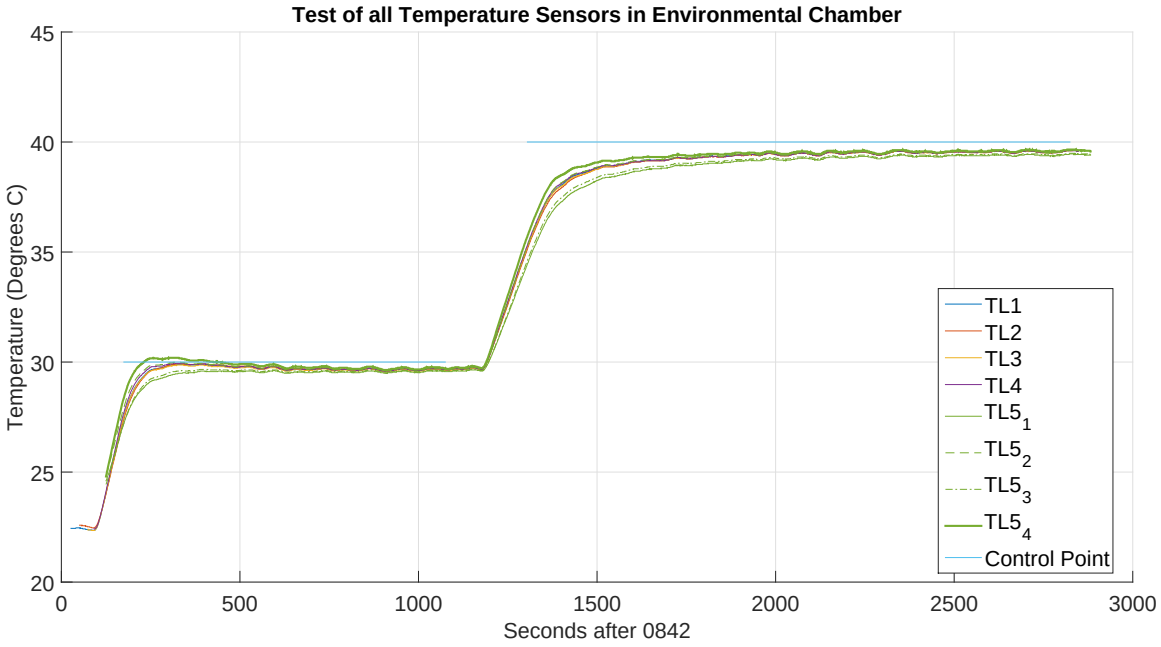


Figure 4-4: HOBO Tests in Temperature Cabinet

Outside weather data is retrieved from a weather station located in the center of the BCIL. The weather station has the ability to collect several types of weather measurements, including temperature, humidity, barometric pressure and solar irradiation. The sampling rate is 1 measurement every 60 seconds. Data from the weather station is stored by the operators of the BCIL in a MySQL database, which can easily be accessed via the Wattsworth web server (`wattsworth.net`), a network enabled database which retrieves and stores data from several sources [**?**]. When using this method, the data must be downloaded via VPN and converted from UNIX timestamps to a time Matlab recognizes so it can interface with the HOBO sensor data. Also, since the HOBO sampling interval is once every 10 seconds, the weather data is linearly interpolated to match the Hobo sample rate. The temperature sensor on the weather station has an accuracy of $\pm.5^o$ C.

Temperature degradation experiments were performed on the Devens tents on four separate days: 27JUN14, 29SEP14, 14OCT14, and 20NOV14. This allowed tests in each direction to be performed, since air conditioning was being used for the first three days and heat for the last. An established test procedure was eventually developed, however there is some variation in test method, mostly related to time duration, between the dates. What did not change was the sensor placement. The HOBO sensors with only 1 probe, TL1-TL4, were zip tied to the top of the bunk closest to the middle of the tent, for a height of approximately 42". The probes coming from TL5 were placed in the center of the tent at different levels on a tripod, with heights similar to those specified in ASHRAE's Thermal Environmental Conditions for Human Occupancy Standard [17]. The probes were placed at 6", 48" and 72", whose average matches the 42" standard for the other sensors. Fig. 4-5 shows the interior of one of the Devens tents with the tripod and probes from TL5 set up. Table 4.1 includes the tent locations for each sensor on each test day.

In general, the test procedure first involved placing and turning on the sensor and setting the ECU to either the highest heat or AC setting (depending on the season). All lights in the tent were shut off and the tent doors were closed. After approximately 2 hours the ECU was turned completely off (not on vent), and the

Figure 4-5: Temperature Sensors in Fort Devens Tent

temperature was allowed to drift. Since the controls for the ECU are on the outside of the tent, the doors to the tent never needed to be opened during the experiment. After approximately 2 more hours the tents were entered and the experiment ended. Data was retrieved from the HOBO sensors and the tents were returned to their previous condition.

During the 20NOV14 tests an internal load was placed in 2 of the tents. A 1500 W Utilitech oscillating space heater, shown in Fig. 4-6a was used instead of the ECU to heat the tent for 2 hours, since it's total load could be more easily quantified. After 2 hours the tents were entered briefly to turn the heaters off and allow the tent to coast. The heaters were placed in the corners of tents 5 and 6, as shown in Fig. 4-6b. As a control the same heaters were placed in tents 7 and 8 with only the fan function on during the heating portion of the test, with the ECU off at all times just like tents 5 and 6. The goal was to try and make the added head the only variable when comparing the tents with internal loads and those without, and eliminate any effect the fan would have on the comparison.

	TL1	TL2	TL3	TL4	TL5
27JUN14	Tent 29	Hardstand	Outside	Tent 28	Outside
29SEP14	Tent 3	Tent 4	Tent 25	Outside	Tent 27
14OCT14 AM	Tent 3	Tent 2	Tent 1	Outside	Tent 4
14OCT14 PM	Tent 3	Tent 2	Tent 1	Outside	Tent 4
20NOV14	Tent 5	Tent 6	Tent 7	Tent 8	Tent 4

Table 4.1: HOBO Sensor Locations

(a) Space Heater used as Internal Load

(b) Space Heater in Tent Corner

Figure 4-6: Internal Load Heaters

4.3 Results from Fort Devens Tests

The thermal model described in Section 4.1 was applied to each of the Fort Devens data sets. There must be an initial guess for each unknown parameter in order for the grey box model to provide an estimation. The initial guess for τ was 35 minutes, based on the construction of the tent and an initial look at the raw temperature data from the first data set. The guess for R was based on the R-value given by the manufacturer for the thermal liner and the surface area of the tent. An R-value is a measure of thermal resistance often used to describe insulation properties in buildings. The higher the thermal resistance, the higher the R-value. The thermal liner has a stated R-value of 1.27 $m^2 \cdot K/W$, which was used as the initial guess for R [18].

For the Devens tent experiments there are three inputs: external temperature, solar irradiation, and internal loads. Real time external temperature and solar irradiation data is taken from the BCIL weather station, and internal loads represent the

heating or AC load applied to the tent. This load is quantified based on the rated power of the unit. The output of the model is the internal temperature of the tent. The model can fit a curve to the output data based on the input data, which can be used to determine thermal characteristics of the tent. These characteristics include the heat capacitance of the tent and the thermal time constant.

The data was organized into 11 data sets which encompassed all of the usable experimental data. The tents tested at Fort Devens are arranged so that there are always two tents connected laterally to each other with a small vestibule. Since this passage allows an exchange of air the two tents are treated as 1 structure for these experiments. Tests where there were HOBO sensors in each tent making up the structure had the temperature readings averaged to form only 1 data set. Table 4.2 contains a schedule of the 11 data sets. Data sets 1, 2, 3, 9 and 11 were randomly chosen and merged to form the training data set, which contains tests in both the heating and cooling directions. The cross validation data set is composed of merged data from data sets 4, 5, 6 and 10. Data sets 7 and 8 were not used because these tests were terminated early.

	Date	**Location**
Data Set 1	June 27	Tent 28
Data Set 2	June 27	Tent 29
Data Set 3	September 29	Tent 3/4
Data Set 4	September 29	Tent 1/2
Data Set 5	October 14 AM	Tent 1/2
Data Set 6	October 14 AM	Tent 3/4
Data Set 7	October 14 PM	Tent 1/2
Data Set 8	October 14 PM	Tent 3/4
Data Set 9	November 20	Tent 5/6
Data Set 10	November 20	Tent 7/8
Data Set 11	November 20	Tent 4

Table 4.2: Temperature Data Set Schedule

Figs. 4-7 and 4-8 show the temperature responses of 2 different training data sets, both as measured and as estimated by the model based on the inputs. Data set 1, shown in Fig. 4-7, contains data from a standard test during the 27JUN14 testing pe-

riod. The ECU was turned on for approximately 2 hours and then turned off, allowing the tent to coast for an additional 2 hours. The measured tent temperature, shown on the top plot in blue, is taken from the HOBO sensor. The fitted temperature, shown in orange, is what is predicted by the model based on the outside temperature and solar irradiation. There are no internal loads for this data set during the model fit period.

Data set 9, shown in Fig. 4-8 is from 20NOV14, and has an internal load accounted for during the heating period. The rating for each of the space heaters used was 1500 W, therefore the internal load was set to 3000 W during the heating period to account for 1 space heater in each tent. The model fit is applied during the entire 4 hour heating and coasting period for data set 9 because the internal load is precisely known.

As can be seen in both figures, the fit generated from simulation of the model matches the measured temperature reasonably well. The model predicted a thermal time constant, τ of approximately 44.6 minutes based on the training data sets. The estimates for C_h, R_{eff} and k were 1.47E6 J/K, 0.584 $m^2 \cdot K/W$, and 1.47E6 m^2, respectively. These value appears reasonable based on the size and material of the tents.

The cross validation model sets provide another opportunity to test the accuracy of the model. None of the data streams from these data sets are used to construct the model, the inputs are just run through the model fit parameters generated by the training data set. Fig. 4-9 shows measured and fitted data from data set 4, which was one of the 14OCT14 experiments. The coast time was 2 hours, and even though this data set was not used to construct the model, the model fit still matches the measured temperature well. A fit score was calculated for both the training and cross validation data sets. The fit score for the training data set, which is how well the model fit matches the measure tent temperature for each individual data set, was 75.8%. The fit score for the cross validation set was 72.4%. Considering the model typically only has 2 inputs, it fits the actual output data quite well.

4.4 Future Tests at Fort Devens

Although the model produces fits that seem to match well with the data, there were several areas of experimental error which could have skewed the model fit parameters. Most notably the 20NOV tests took place after the sunshades had been removed from the tents, potentially increasing the impact of solar irradiation during these tests. Determining and including a factor for the sunshade being on or off in the model may lead to a better fit.

The incidence angle of the sun on the tent could also alter the effect the solar irradiation is having on the tent temperature. Although the tents are generally orientated in the same direction, the tests were conducted at different times of the year and over the course of several hours during the day. The incidence angle changes continuously, and at certain angles the sun may not be affecting the tent structure in the same way. Several websites, such as `http://www.esrl.noaa.gov/`, give the azimuth and elevation of the sun given an input of latitude, longitude, date and time. These can be combined with a 3D CAD model of a tent to determine the sun angle's effect. Models of the 16 Devens tents were produced as surfaces in Rhino based on information from the tent manufacturer. The tents were overlaid over a scaled aerial image of the BCIL in order to get accurate spacing between the tents. A perspective view of the tent models can be seen in Fig. 4-10. The incident angle of the sun at 12:00 PM on 20NOV14 over Tent 1 was placed in the model as an example. Scripting in the Rhino programming tool Grasshopper could be used in the future to quantify the effect of different sun angles on the tent model.

Finally, the model likely imparts its own bias on the parameter values found during the fit. Specifically, the model treats a tent as a lumped mass, the tent surface as a lumped resistance, and the environment as a lumped heat source or sink. In reality there is heat transfer between areas of the tent, thermal storage is the tent liner, and variability in environmental conditions external to the tent, e.g. air space below the tent compared to air space above the tent. As more data is collected, higher order models allowing for these realities can be applied to reduce model bias.

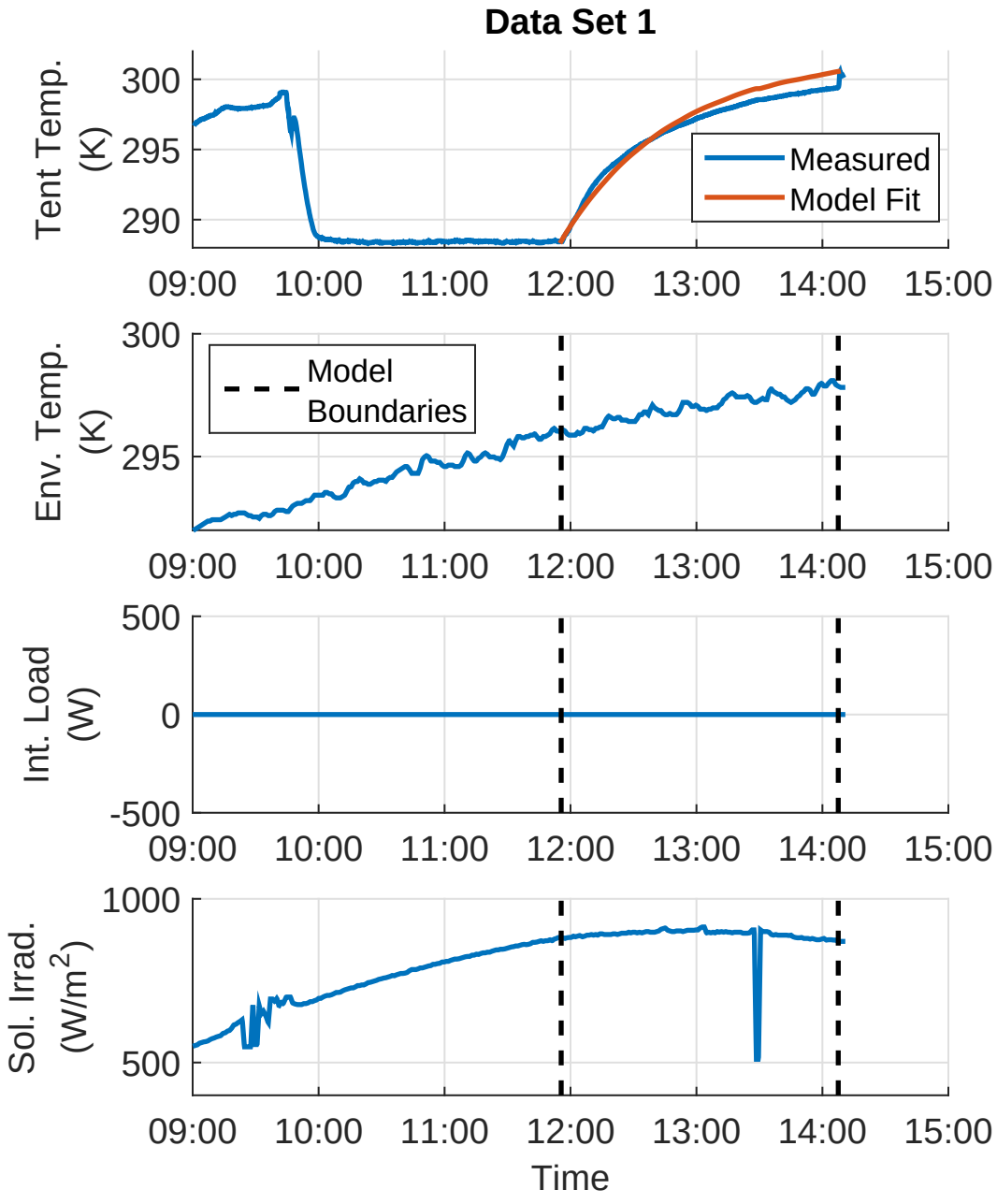


Figure 4-7: Measured and Modeled Responses of a Devens Tent

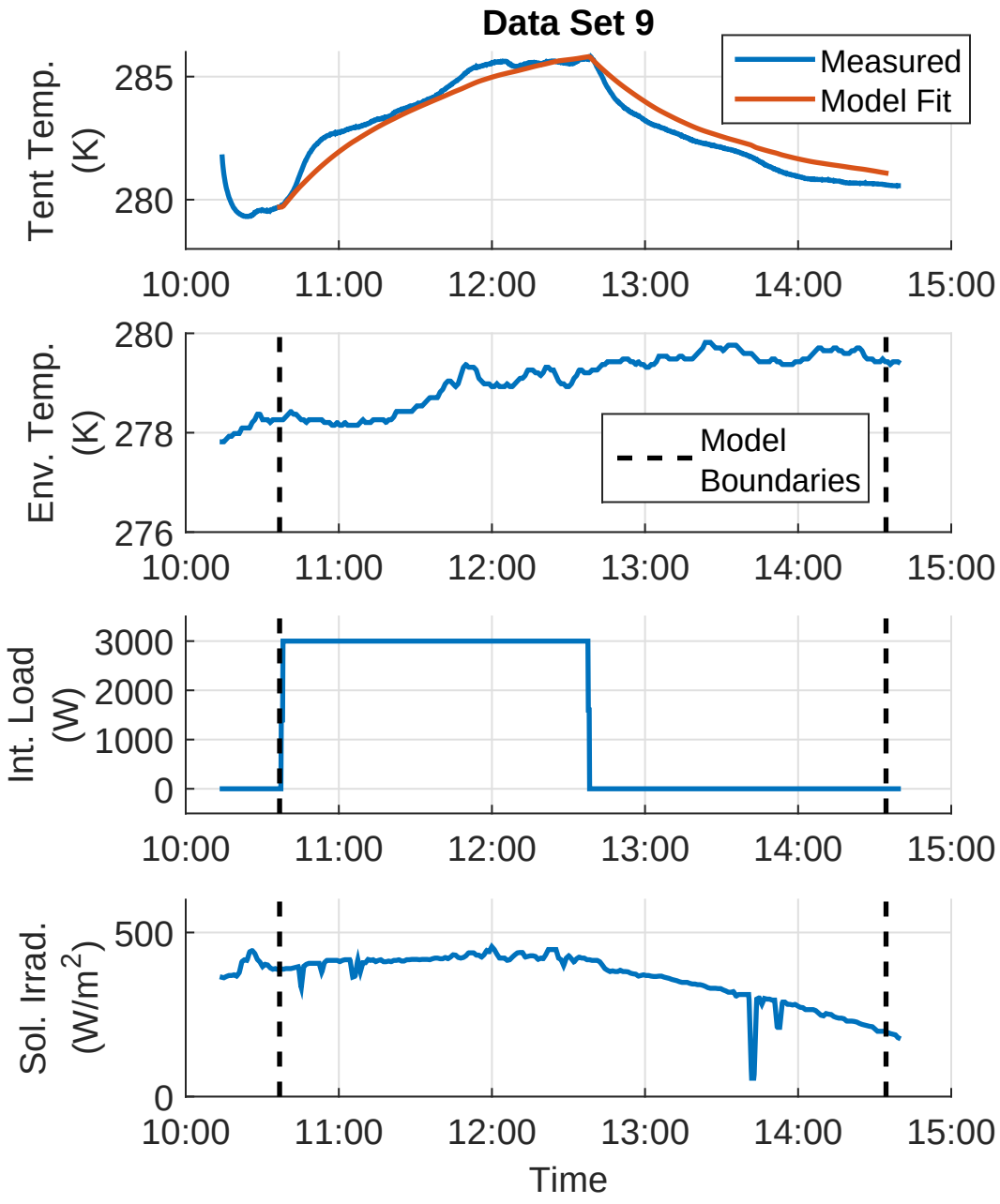


Figure 4-8: Measured and Modeled Responses of a Devens Tent with Internal Loads

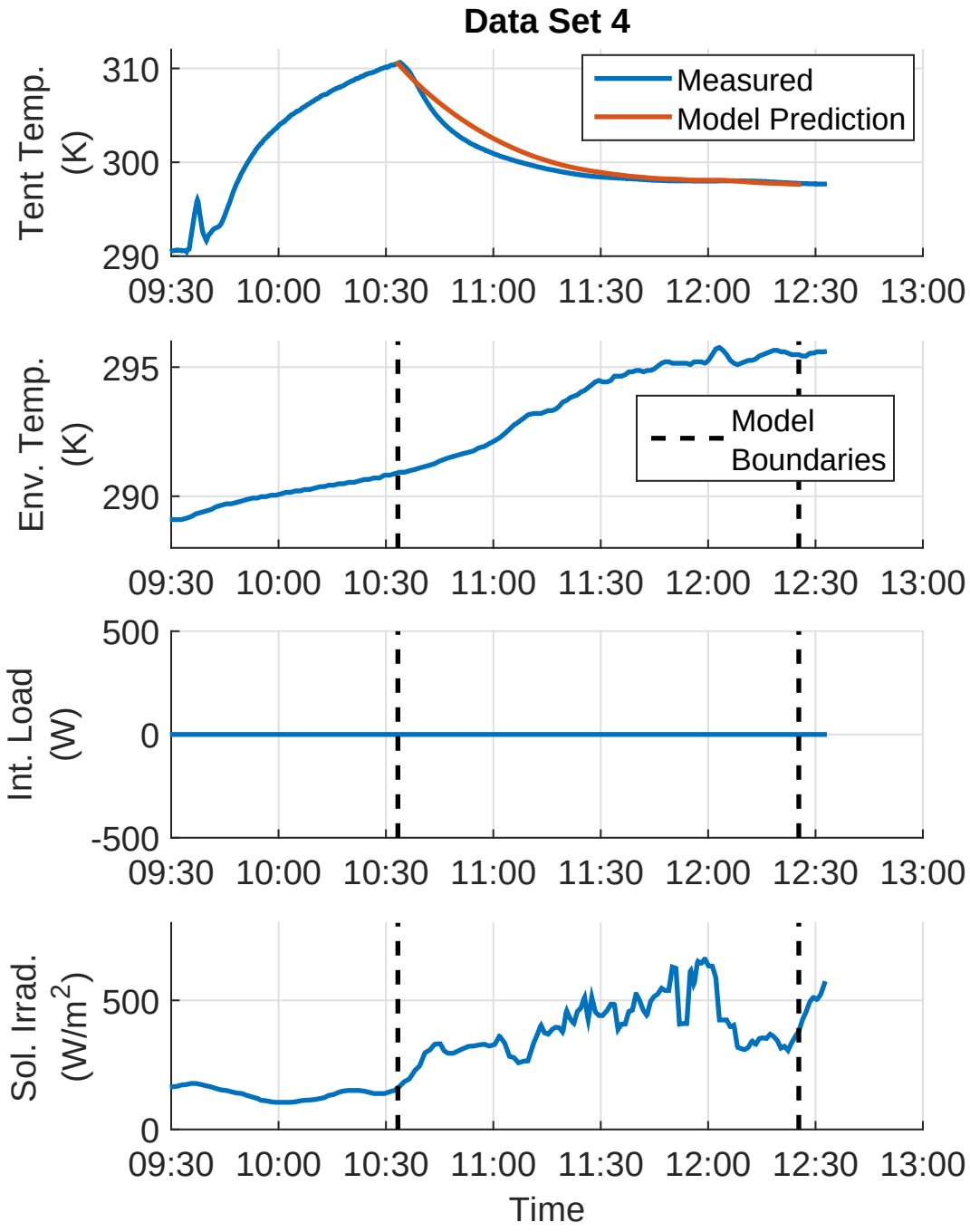


Figure 4-9: Measured and Modeled Responses Cross Validation Data Set

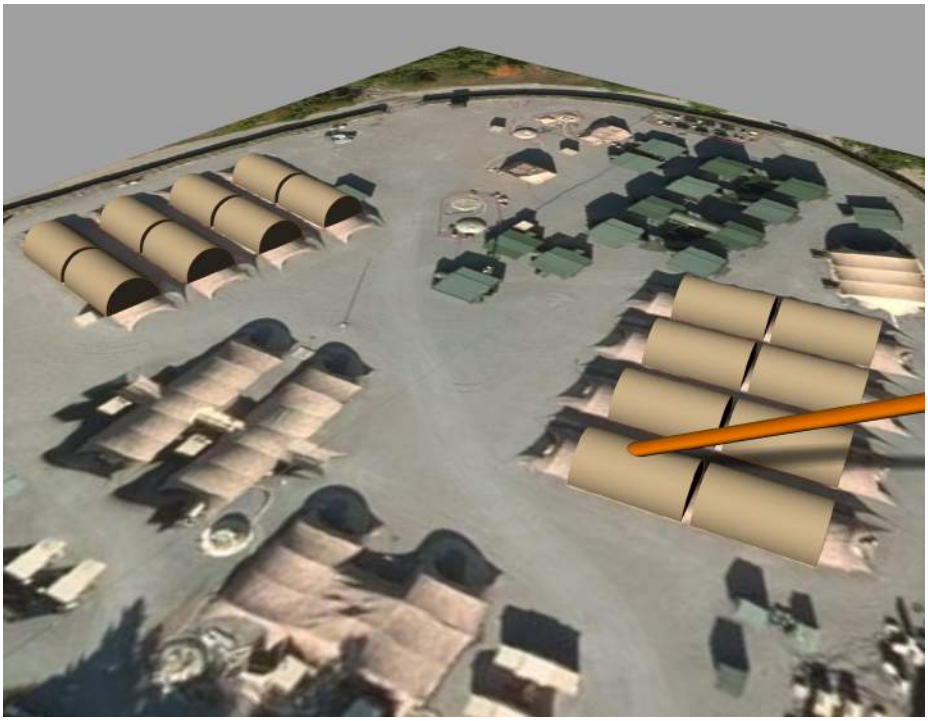


Figure 4-10: CAD Model of Devens Tents with Example Sun Angle

Chapter 5

NILM- An Electronic Stethoscope for the Navy

Energy availability and use on board ships continues to be an interest for the US Navy. Future weapons and sensors are projected to draw more energy than today's system, meaning that current existing loads need to be effectively quantified and managed. Beyond a total wattage produced by a generator or a total load on a bus, ship's crews have little energy information available about their ship. A Non-Intrusive Load Monitor (NILM) offers the opportunity to see into the ship's electrical plant with little installation or maintenance effort. A NILM provides much more than just a total load to operators, it continuously monitors and shows the actual power, allowing transients to be seen and loads to be identified. Information on aggregate power use, human activity and machinery diagnostics can be inferred from this data.

Traditional NILMs operate by measuring voltage and current at a main panel. Current is measured using current transducers, which need to be placed around each phase line coming into the panel. Voltage is measured from an unused breaker or other connection point. The voltage taps and current sensors feed into an equipment box, known as the NILM, which reads the current values and measures the voltage using voltage transducers, all while sampling the data and outputting to a computer via an Ethernet connection. A traditional NILM installation is depicted in Fig. 5-1, taken from [19]. These NILMs are very effective at accurately gathering data, and

are currently deployed in multiple military settings, including the BCIL and USCGC SPENCER (WMEC 905). They also, however, involve a non-trivial installation process. Power to the panel must be disconnected and an electrician must install the sensors. A non-contact, combined current and voltage sensor offers the opportunity to monitor loads without opening the breaker box or securing power; the sensors can simply be zip-tied to the outside of the wire and offer comparable performance to the traditional setup.

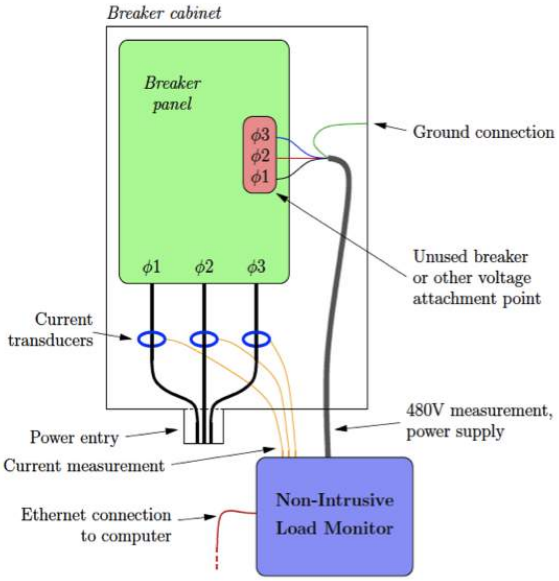


Figure 5-1: Traditional NILM Installation Schematic

A non-contact sensor, developed by John Donnal, exploits the electric and magnetic fields which reside around an energized wire to sense voltage and current, respectively [20]. Previously, the sensors had only been tested in a lab environment, and there was a desire to test the sensors in a real-world setting with different types of cables. A US Navy ship was a prime candidate since it has many different size power cables available to test sensors on. The ships also have several large loads, such as pumps and air conditioning plants, which cycle on and off frequently, allowing additional opportunities to make inferences on human activity and diagnose potential machinery faults. A total of 3 ships were visited in order to test non-contact sensing technology; the first two ship visits only tested the sensor on several different types of cables, and the third ship visit deployed an actual NILM with non-contact sensors

and recorded data while the ship was in-port and underway.

5.1 Initial Non-Contact Sensing Tests

The standards for power cables aboard US Navy ships can be found in the detail specification MIL-DTL-24643B. Depending on the requirements of the systems, there may be several types of power cables installed aboard the ship. Some cables are required to be shielded, while others require a braided metal armor. A common cable used for power and lighting may look like the LSTSGU type, which comes in ratings between 10-400 amps. A stranded conductor is wrapped in silicone insulation, and is wrapped further with a glass braid along with the 2 other phases with fillers in between. After a layer of tape there is an outer polyolefin jacket. These types cables are ubiquitous on US Navy ships, and can be found anywhere from a generator switchboard to a downstream lighting power panel. Fig. 5-2 is a depiction of an LSTSGU cable, courtesy of Seacoast Electric Company, Inc. Most cables to not have the braided metal armor, nor are they shielded.

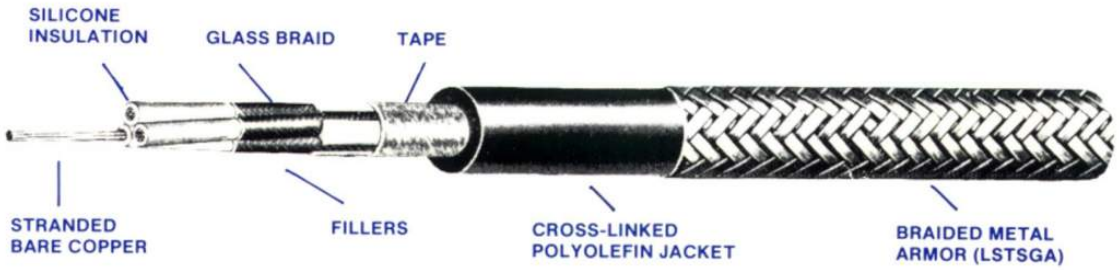


Figure 5-2: LSTSGU Cable

The first ship the non-contact sensor was tested on was the USS LAKE CHAMPLAIN (CG-57), a 567', 9600 LT guided missile cruiser commissioned in 1988. The sensor simply connected to a power supply via an Ethernet cable, which connected to a LabJack UE9 for data acquisition. Only straight current and voltage on a single phase were read, no attempt was made to extract actual power from the data. Several different size cables coming out of Load Center 41 were tested, as well as cables going

in an out of a 3-phase power panel and a Fan Coil Unit (FCU). Each test returned similar voltage and current waveforms.

The second ship tested was the USS NEW ORLEANS (LPD 18), a 684', 24,000 LT, landing platform dock commissioned in 2008. The sensor was held up to a supply cable of a FCU while it was turned on and off several times in order to see a change in current, which was observed. The sensor was also held up to several different size cables off of a 3-phase power panel, as seen in Fig. 5-3. Each cable returned a strong voltage and current through the sensor. Although the tests on LAKE CHAMPLAIN and NEW ORLEANS were very basic, they proved the ability of a non-contact sensor to work on cables featured prominently on US Navy ships.

Figure 5-3: Sailor Holding Non-Contact Sensor to Power Cable on NEW ORLEANS

5.2 Non-Contact NILM on MCM

A full non-contact NILM sensor and data acquisition system was tested aboard USS CHAMPION (MCM 4), concurrently with the vibration tests discussed in 3. The sensors used were a further iteration than the ones tested during the two previous ship visits, however the general sensing method remained the same. The non-contact NILM system consists of 4 parts. The main sensor senses voltage and current from

one phase, while two daughter boards connected via ribbon cable measure the current from the two other phases on the cable. The fourth part of the system is a small device which collects, stores, and transfers the data. This device connects to the main sensor via ribbon cable and requires a 12 V power supply. It has the ability to host its own WiFi, allowing the operator to easily see and manipulate the data.

The ship was pier-side and on shore power when the NILM was first installed. The MCM class has 3 shore power cables which feed into a bus tie unit on the number 1 switchboard in the AMR. The bus tie unit services loads on the 1S switchboard and also connects to the number 2 switchboard in the MMR. The total electric load on the ship while on shore power is shared by the 3 cables. Each shore power cable is an LSTSGU-400 type, meaning it is a standard 3 wire unshielded, unarmored cable rated to 400 amps. The NILM was originally installed on one of the shore power cables, as shown in Fig. 5-4. The 3 sensors are approximately 120^o off from each other to capture each phase.

Figure 5-4: Non-Contact NILM Installation on USS CHAMPION (MCM 4)

One of the purposes of the instalation of the NILM was to see how well it could

pick out individual loads. Since the NILM was very far upstream, basically at the initial generation point as far as the ship was concerned, there were hundreds of loads being monitored on the NILM. These consumers included lighting, computers, fans, pumps and galley equipment. One of the most interesting loads observed was the starting and stopping of one of the ship's AC plants. The ship was required to start and stop the number 2 AC plant several times as part of an assessment; four of these startup transients can be seen in Fig. 5-5. The transient can be identified based on the characteristic spike in power as the machine starts up. The first and third transients show a normal startup, followed by about 20 seconds of steady state run time where the power levels off at about half of the startup peak, and finally by a shutdown. The fourth transient does not show a shutdown, as this was the last test and the machine was to remain on. The second transient shows only a startup spike, but no leveling off characteristic of steady state operation. This is because the AC failed to start properly during this run; after the AC started it immediately shut itself down for an unknown reason. This lack of proper start can clearly be seen by the NILM, and could have served as an indicator that there was an improper start had the crew been operating the AC plant remotely.

Figure 5-5: MCM Number 2 AC Transients

The ship eventually shifted from shore power to ship's power in preparation for getting underway. The NILM sensors were moved to 1 of the 2 LSTSGU-300 cables coming from SSDG 1B, meaning the NILM was now monitoring half of the load being serviced by the 1B generator. Fig 5-6 shows approximately 1 hour of electrical

power data while the ship was underway. More information on specific loads would be needed to effectively use the data, however the MCM test proved that the NILM could function on the some of the largest power cables used on board ships.

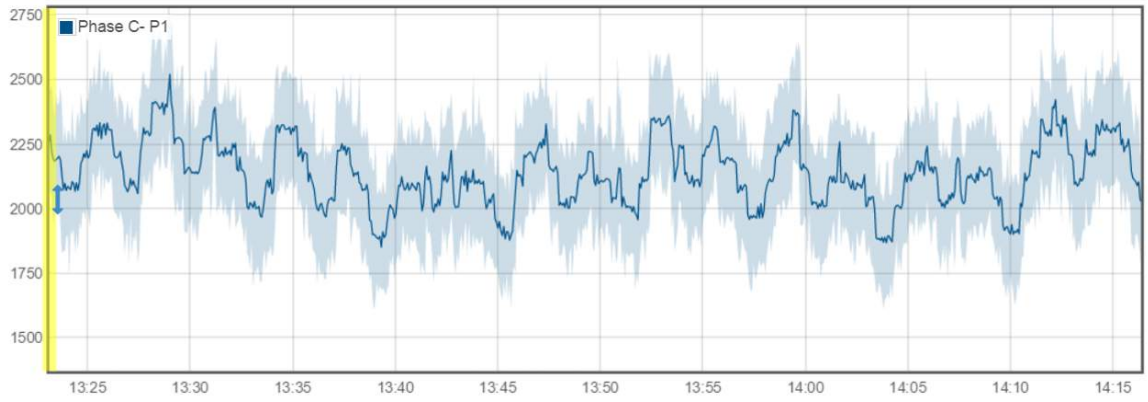


Figure 5-6: MCM SSDG 1B Transients

The NILM has great promise for future use aboard US Navy ships. The traditional NILM is well suited for permanant instalation while the non-contact NILM excels with temporary instalations and instances where then sensor may need to be moved. If the NILM is installed on a ship for a longer period of time, machine learning algorithms and other load identification techniques can be used to identify loads easily. Once the loads are identified then tools can be used to tell operators how long machines are running and how often they start and stop. They can also be used for condition based maintenance by identifying if a machine has a different transient than normal.

THIS PAGE INTENTIONALLY LEFT BLANK

Appendix A

VIB-P Operation and Construction

Use the following procedure to operate the VIB-P in a field test setting:

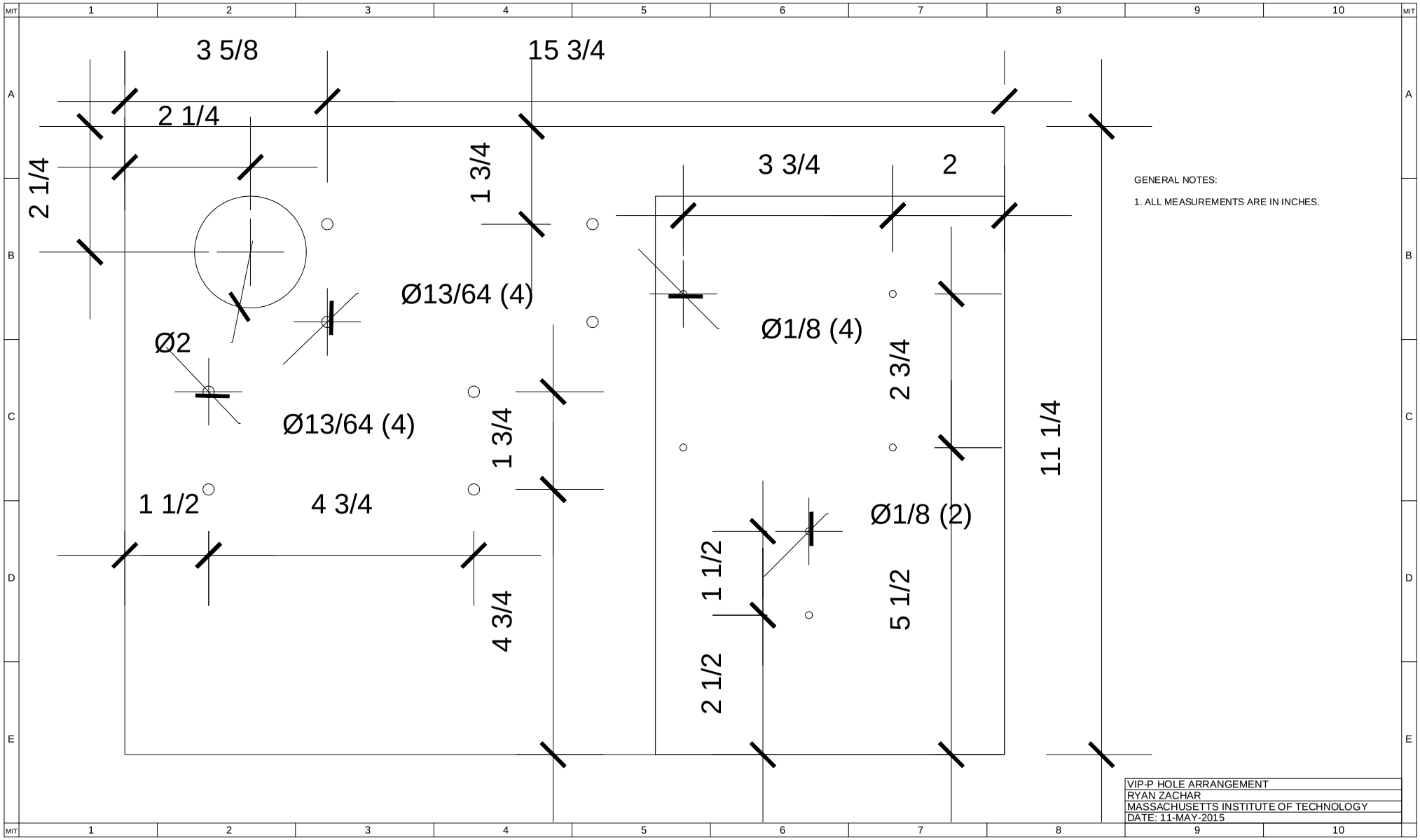
1. Electrically and mechanically isolate the machine.
2. Attach the back EMF sensor to each phase line in the terminal box using at least 1 zip tie.
3. Mount the vibration senors to the machine using the mounting block with either a magnet or Command strip.
4. Connect vibration sensor 515 to power supply 1095, and sensor 516 to sensor 1096.
5. Open LJStreamUD. If needed download from `www.labjack.com`.
6. Ensure the stream rate in LJStreamUD is 4 kHz.
7. Make working directory a new folder where only current setup's test will be written.
8. Make filename in LJStreamUD something unique for the test.
9. Start LabJack stream on LJStreamUD and run for at least 10 seconds.
10. Start and stop machine in accordance with established procedures.
11. If doing multiple runs continuously, wait at least 5 seconds in between runs.
12. Stop LabJack stream at least 5 seconds after final spin-down.

The VIB-P contains the following parts:

Part	**Quantity**	**Available From**
ACC786A Accelerometer	2	www.omega.com
ACC-PS3 Power Supply	2	www.omega.com
ACC-CB7-10 Cable	2	www.omega.com
LabJack UE9	1	www.labjack.com
Back EMF Sensor	3	MIT LEES
Back EMF Power Supply	1	MIT LEES
Nanuk 920 Case	1	www.amazon.com
Various Mounting Screws	10	mcmaster-carr.com
15' Extension Cord	1	mcmaster-carr.com
USB Cable	1	www.digikey.com
15' Ethernet Cable	1	mcmaster-carr.com
6 Outlet Surge Protector	1	mcmaster-carr.com

Table A.1: VIB-P Parts List

A hole drilling schematic is included in the following drawing.


3 5/8
15 3/4
2 1/4
2 1/4
1 3/4
3 3/4
2
Ø13/64 (4)
Ø1/8 (4)
Ø2
2 3/4
11 1/4
Ø13/64 (4)
1 3/4
1 1/2
4 3/4
Ø1/8 (2)
1 1/2
5 1/2
4 3/4
2 1/2
GENERAL NOTES:
1. ALL MEASUREMENTS ARE IN INCHES.
VIP-P HOLE ARRANGEMENT
RYAN ZACHAR
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
DATE: 11-MAY-2015

THIS PAGE INTENTIONALLY LEFT BLANK

Appendix B

MMTP Operation and Construction

Use the following procedure to operate the MMTP for steady state testing:

1. Ensure VIB-P is connected and powered up in accordance with steps 1-4 in Appendix A.
2. Ensure shaft guards are in place, and that all connections are tight and orderly.
3. Ensure dedicated Linux workstation is connected to the shaft encoder via USB, the PSP-603 via USB, and the LabJack via USB and Ethernet.
4. Power on PSP-603 and HP 6010A power supplies.
5. Open the *motor_control.py* script on the dedicated Linux machine.
6. Edit STEP_SIZE to adjust for resolution, remember 1 V commanded by the program is approximately 2.3 Hz on the motor.
7. Make sure the *motor_control.py* script is saved.
8. Run the script *motor_control.py filename* where *filename* is the name of the test.
9. Follow prompts in the terminal window to create the VTF.
10. After the script is complete the FRF and VTF files are saved in the $motor_control$ folder as $filename\_sd\_frf.dat$ and $filename\_ss\_frf.dat$.
11. Shut down all DC and VIB-P power supplies.

To change the mounts on the MMTP:

1. Ensure all power supplies are off.
2. Loosen 8 mounting nuts with 9/16" socket.
3. Attached shop crane to 4 lifting eyes and lift platform slowly.
4. Move platform away from the machine, no wires should need disconnecting.
5. Remove and replace mounts, using 9/16" wrench to remove nut from under girder flange.
6. Place washer and nut on bolt under the flange but do not tighten yet.
7. Open hydraulic valve 1/8 of a turn, MMTP should lower very slowly.
8. Guide MMTP onto the mounts.
9. Tighten bolts under the flange with wrench.
10. Replace top washers and nuts, use torque wrench to tighten bolts to 100 in-lbs.
11. Replace all shaft guards and do visual check for any debris on the system.

A parts list for the MMTP can be found in Table B.1. A hole drilling schematic is included in the drawing following the table.

Part	Quantity	Available From
Instek PSP-603 Power Supply	1	www.testequity.com
HP 6010A Power Supply	1	www.testequity.com
0-60 V to 0-5 V Converter	1	MIT LEES
AC 3-Phase Induction Motor	1	MIT LEES
DC Permanent Magnet Motor	1	MIT LEES
AC Single Phase Motor	1	MIT LEES
36.5" X 15"X3/8" Aluminum Plate	1	www.mcmaster.com
Extreme Temperature Vibration-Dampening Sandwich Mount Male/Female 5/16"-18 thread, 60 lb. compression cap (30A)	8	www.mcmaster.com
Extreme Temperature Vibration-Dampening Sandwich Mount Male/Female 5/16"-18 thread, 90 lb. compression cap (40A)	8	www.mcmaster.com
Extreme Temperature Vibration-Dampening Sandwich Mount Male/Female 5/16"-18 thread, 125 lb. compression cap (50A)	8	www.mcmaster.com
Extreme Temperature Vibration-Dampening Sandwich Mount Male/Female 5/16"-18 thread, 175 lb. compression cap (60A)	8	www.mcmaster.com
Extreme Temperature Vibration-Dampening Sandwich Mount Male/Female 5/16"-18 thread, 225 lb. compression cap (70A)	8	www.mcmaster.com
5/16"-18 Thread, 1-1/2" Long, Fully Threaded Stud	40	www.mcmaster.com
5/16"-18 Thread, 1-1/2" Long, Steel Cap Screw	8	www.mcmaster.com
5/16"-24 Thread, 1-1/2" Long, Steel Cap Screw	2	www.mcmaster.com
5/16"-18 Thread Hex Nut	24	www.mcmaster.com
5/16"-18 Screw Size Washers, Small OD	24	www.mcmaster.com
Flexible Shaft Coupling. Hub for 5/8" Diameter Shaft, 2-5/32" Overall Length, Iron	1	www.mcmaster.com
Flexible Shaft Coupling. Hub for 7/8" Diameter Shaft, 2-5/32" Overall Length, Iron	1	www.mcmaster.com
Flexible Shaft Coupling. Hub for 1" Diameter Shaft, 2-5/32" Overall Length, Iron	1	www.mcmaster.com
Flexible Shaft Coupling. Hub for 5/8" Diameter Shaft, 2-1/8" Overall Length, Iron	2	www.mcmaster.com
Buna-N Spider for 2-7/64" OD Flexible Shaft Coupling	1	www.mcmaster.com
Buna-N Spider for 1-3/4" OD Flexible Shaft Coupling	1	www.mcmaster.com
Steel Machine Key, Oversized With Square Ends, 3/16" Square, 1" Length	3	www.mcmaster.com
Steel Shim Stock, .01" Thick, 6" Width	2'	www.mcmaster.com
Steel Eyebolt with Shoulder. 3/8"-16 Thread Size, 5/8" Long Thread	4	www.mcmaster.com

Table B.1: MMTP Parts List

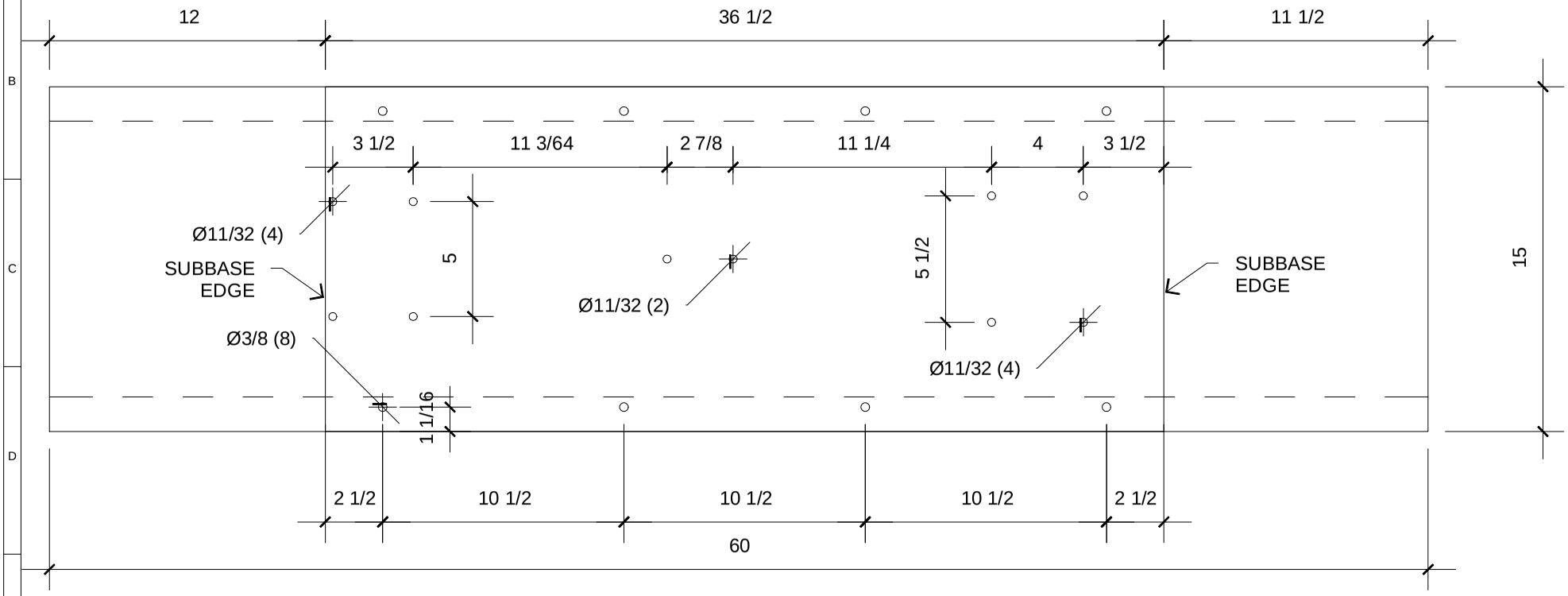
GENERAL NOTES:

1. ALL MEASUREMENTS ARE IN INCHES.

2. STEEL BOX GIRDER IS 60 X 15, AND ALUMINUM SUBBASE IS 26.5 X 15.

3. DASHED LINES REPRESENT WEB OF BOX GIRDER; ADEQUATE CLEARANCE FOR WASHER AND NUT MUST BE ACCOUNTED FOR WHEN DRILLING.

4. ALL MOTOR HOLES SHALL BE OFF CENTERLINE.



MMTP SUBBASE HOLE ARRANGEMENT
RYAN ZACHAR
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
DATE: 11-MAY-2015

Appendix C

Code Used for Experiments and Data Processing

Section C.1 contains the *motor_control.py* code to run the MMTP and produce at steady state FRF.

Section C.2 contains the 4 files used to generate a VTF. The *analysis.m* file (C.2.1) loads the data and calls the $frf\_function.m$ function (C.2.2) which creates the actual VTF. The function *extract_speed_hilbert.m* (C.2.3) is used if the Hilbert transform is used to extract the speed, and *extract_speed_zeroX.m* (C.2.4) is used if the zero crossing method is chosen instead.

C.1 MMTP Automated Steady State

```
#!/usr/bin/python

"""
Runs the motor test bench
John Donnal 2015
"""

import serial
import time
import threading
import pdb
import subprocess
import matplotlib.pyplot as plt
import numpy as np
from scipy.signal import welch
import signal
import os
import argparse

import frf_analysis

##### EDIT TO ADJUST FRF RESOLUTION ##


STEP_SIZE = 3.5 #in volts, .5 V for real tests
MAX_VOLTAGE = 26
MIN_VOLTAGE = 5 #normally 5
HIGH_SPEED_HOLD_TIME = 4 #in seconds (at each speed)
LOW_SPEED_HOLD_TIME  = 4 # (below 20 rps)
SPEED_BOUND = 2 # +/- Hz bounds around measured speed
#                           to look for peak of the steady state FFT
##########################################


DEV_SUPPLY = "usb-Prolific_Technology_Inc._USB-Serial_Controller-if00-port0
DEV_ENCODER = "usb-US_Digital_USB__-__QSB_73724-if00-port0"
COUNTS_PER_REV = 7200
ser_supply = None
ser_encoder = None

def initialize():
```

```
global ser_supply, ser_encoder
ser_supply = serial.Serial("/dev/serial/by-id/%s"%DEV_SUPPLY,
                           2400,dsrdtr=True,timeout=1)
ser_supply.setDTR(True)
ser_encoder = serial.Serial("/dev/serial/by-id/%s"%DEV_ENCODER,
                            230400,timeout=None)
#stop any streaming and flush the input buffer
ser_encoder.write("R0E\r\n")
time.sleep(0.5)
ser_encoder.flushInput()
#set up timestamps
ser_encoder.write("W1507\r\n")
ser_encoder.readline() #consume the output
print("Initializing encoder...")
ser_encoder.write("R14\r\n")
val = ser_encoder.readline()
ser_no = val[3:8]
type_code = int(val[8])
firmware = val[9:11]
if(type_code==0):
    dev_type="QSB-D"
elif(type_code==1):
    dev_type="QSB-M"
elif(type_code==2):
    dev_type="QSB-S"
else:
    dev_type="unknown device type %s"%type_code
ts = int(val[11:19],16)
print("[%d] Detected [%s]: #%s v%s"%(ts, dev_type, ser_no,firmware))
#configure the encoder for quadrature reads
ser_encoder.write("W000\r\n")
ser_encoder.readline()
#configure the counter to free run in quadrature mode
ser_encoder.write("W031\r\n")
ser_encoder.readline()
#enable counting in the up direction
ser_encoder.write("W040\r\n")
ser_encoder.readline()
#set the interval rate to maximum
ser_encoder.write("W0C0\r\n")
ser_encoder.readline()
#set the reporting threshold to 10 (0xA) ticks
ser_encoder.write("W0BA\r\n")
ser_encoder.readline()
```

```
def read_speed():
    last_ts = None
    last_ticks = None
    last_val = ""
    i=0
    accm = 0
    num_vals = 0
    #flush any input
    ser_encoder.flushInput()
    #stream the encoder values
    ser_encoder.write("S0E\r\n")
    while(i<500):
        val = ser_encoder.readline()
        #check to see if there is bogus stuff here!
        #looks like a problem with the QSB itself
        val=val.replace("\x00","")
        try:
            ticks = int(val[3:11],16)
            ts = int(val[11:19],16)
        except ValueError:
            #extremely bad line from QSB itself
            print("bad line from QSB: %s"%val.rstrip())
            last_ts = None
            continue
        #calculate the velocity
        if(last_ts is None):
            last_ts = ts
            last_ticks = ticks
            continue
        ts_diff = float(abs(ts-last_ts))/1000.0 #in seconds
        if(ts_diff == 0): #skip this measurement
            continue
        tick_diff = abs(ticks-last_ticks)
        velocity = (float(tick_diff)/COUNTS_PER_REV)/ts_diff
        accm+=velocity
        num_vals += 1
        #print("%.3f %d : %.2f RPS"%(ts_diff,tick_diff,velocity))
        last_ticks = ticks
        last_ts = ts
        last_val = val
        i+=1
    #stop streaming values
    ser_encoder.write("R0E\r\n")
    time.sleep(1)
    ser_encoder.flushInput()
```

```
    return accm/num_vals

def read_vibes(sample_length):
    channel = 2
    dead_time = 3      #discard the first 3 seconds b/c of settling
    Fs = 8000
    lines = (sample_length+dead_time)*Fs
    tmp_file = open("tmp.dat",'w')
    FNULL = open(os.devnull,'w')
    subprocess.call(["ethstream","-L","-l","%d"%lines,"-C",
                     "%d"%channel],stdout=tmp_file,stderr=FNULL)
    tmp_file.close()
    #open up the data file with numpy
    vals = np.loadtxt("tmp.dat")
    vals = vals[dead_time*Fs:lines]
    #detrend the data
    mean = np.mean(vals)
    vals = vals-mean
    #run pwelch to get steady state frequency info
    #Vals = np.array(welch(vals,fs=Fs,nperseg=4096,nfft=4096*4))
    nfft = 1
    while nfft<len(vals):
        nfft =nfft*2
    Vals = np.abs(np.fft.fft(vals,n=nfft))
    freq = np.linspace(0,Fs,nfft)
    #return (vibe,freq) tuple
    return (Vals,freq)

def shutdown():
    ser_supply.close()
    ser_encoder.close()

def set_voltage(value):
    if(value > 60 or value < 0):
        print("Error, invalid voltage %f"%value)
        return
#     print("Setting output to %05.2f volts"%value)
    ser_supply.write("SV %05.2f\r"%value);

def enable_voltage():
    ser_supply.write("KOE\r")

def disable_voltage():
    ser_supply.write("KOD\r")
```

```
def main(filename):

#DEMO of wierd accel output
#    for i in range(1,8):
#        print("sampling for %d sec"%i)
#        lines=(i+2)*8000
#        channel = 2
#        tmp_file = open("tmp.dat",'a')
#        time.sleep(5)
#        subprocess.call(["ethstream","-L","-l","%d"%lines,"-C",
#                         "%d"%channel],stdout=tmp_file)
#        tmp_file.close()

#DEMO of reading vibes (duration of read shouldn't matter)
#    print("reading vibes for 1 sec")
#    read_vibes(1)
#    print("reading vibes for 4 sec")
#    read_vibes(4)
#    print("reading vibes for 8 sec")
#    read_vibes(8)
#    print("reading vibes for 1 sec")
#    read_vibes(1)

    initialize()
    enable_voltage()
    time.sleep(1)
    #set up the FRF plot
    figFRF = plt.figure()
    plt.title('Vibration Transfer Function')
    plt.xlabel('Frequency (Hz)')
    plt.ylabel('Magnitude')
    axFRF = figFRF.add_subplot(111)
    axFRF.set_xlim(0,60)
    #axFRF.set_ylim(0,200);
    d_spinup, = axFRF.plot([], [],label='Steady State')
    d_spindown, = axFRF.plot([], [],label='Spin Down')
    axFRF.legend(loc=4)

    #set up the pwelch plot
    figWelch = plt.figure()
    axWelch = figWelch.add_subplot(111)
    plt.title('Steady State FFT')
    plt.xlabel('Frequency (Hz)')
    plt.ylabel('Magnitude')
```

```
axWelch.set_xlim(0,60)
axWelch.set_ylim(1e-3,1e5)
d_welch, = axWelch.semilogy([],[],label='|FFT|')
d_speed, = axWelch.semilogy([],[],label='Measured Speed')
d_pt_est, = axWelch.semilogy([],[],'*',
                             label='Point Estimate')
#show the plots
axWelch.legend(loc=4)
plt.show(block=False)
#initialize the arrays
speeds = []
vibes = []
frf = []
print("Spinning Up")
for i in np.arange(MIN_VOLTAGE,MAX_VOLTAGE,STEP_SIZE):
    set_voltage(i)
    time.sleep(1)
    speed=read_speed()
    if(speed>20): #high speed
        (vibe,freq) = read_vibes(HIGH_SPEED_HOLD_TIME)
    else: #low speed, read for longer
        (vibe,freq) = read_vibes(LOW_SPEED_HOLD_TIME)
    #now plot the vibe_freq and the measured speed
    #the maximum of vibe_freq should match with measured speed
    d_welch.set_xdata(freq)
    d_welch.set_ydata(vibe)
    figWelch.canvas.draw()
    #find the point estimate, the maximum vibration response
    #withing physially possible frequencies
    #the max should be within +/- SPEED_BOUND of measured speed,
    vibe_segment = vibe[(freq>speed) &
                        (freq<speed+SPEED_BOUND)]
    freq_segment = freq[(freq>speed) &
                        (freq<speed+SPEED_BOUND)]
    max_vibe = np.max(vibe_segment)
    max_idx = np.argmax(vibe_segment)
    #set the plot bounds appropriately
    axWelch.set_ylim(1e1,100*np.max(vibe_segment))
    #vertical line at the measured speed
    d_speed.set_xdata([speed,speed])
    d_speed.set_ydata([1e-3,100*np.max(vibe_segment)])

    d_pt_est.set_xdata(freq_segment[max_idx])
    d_pt_est.set_ydata(max_vibe)
    figWelch.canvas.draw()
```

```
        speeds.append(freq_segment[max_idx]) #use the *actual* speed
        vibes.append(max_vibe)
        #add the point estimate to the steady state FRF
        new_vibe_pt = max_vibe/(freq_segment[max_idx]**2)
        frf.append(20*np.log10(new_vibe_pt))
        d_spinup.set_ydata(frf)
        d_spinup.set_xdata(speeds)
        axFRF.set_ylim(np.min(frf),np.max(frf)+10)
        figFRF.canvas.draw()

    #------ now do a FRF spindown --------
    # Let the motor run at steady state RPS for 3 seconds
    # then turn off. To get this right we start a subprocess
    # that waits 6 seconds and then cuts power and in the main
    # thread we collect data with ethstream
    # 1.) first 3 seconds are removed b/c of accelerometer settling
    # 2.) next 3 seconds are steady state,
    # 3.) the next 3 seconds are the spindown
    # 4.) the last 2 seconds are the back EMF steady state
    #
    # ----------,
    # |    |    | \
    # |    |    |  \
    # |    |    |   ,
    # | 1 | 2 |  3   | .. | 4 |
    #
    #
    #we already calculated the speed here, so just let the
    #motor run
    t = SpinDownThread()
    #set up the lab jack
    channels = "0,1,2,3"
    dead_time = 3      #discard the first 3 seconds b/c of settling
    Fs = 8000
    lines = 20*Fs
    tmp_file = open("tmp.dat",'w')
    FNULL = open(os.devnull,'w')
    t.start()
    subprocess.call(["ethstream","-L","-l","%d"%lines,"-C",
                     "%s"%channels],stdout=tmp_file,stderr=FNULL)
    t.join() #should already be done
    tmp_file.close()
    #open up the data file with numpy
    vals = np.loadtxt("tmp.dat")
    #detrend the data
```

```
mean = np.mean(vals)
vals = vals-mean
vals = vals[3*Fs:]
#make a time series and add it to the array
ts = np.linspace(0,17,17*Fs)

data = np.hstack((ts.reshape(len(ts),1),vals))
# build the config object
config = {
    "DEAD_TIME_L"    : 15*Fs,
    "DEAD_TIME_R"    : 17*Fs,
    "SPIN_DOWN_L"    : 1*Fs,
    "SPIN_DOWN_R"    : 12*Fs,
    "STEADY_STATE_L": 1*Fs,
    "STEADY_STATE_R": 2*Fs,
    "FS"             : 8e3,
    "RPS"            : speeds[len(speeds)-1]
}
(sd_freq,sd_frf)=frf_analysis.analyze(data,config)
#convert to decibels
sd_frf = 20*np.log10(sd_frf)
#scale (done empirically)
sd_frf = sd_frf*7/8
#plot it against the spin up (steady state)
d_spindown.set_ydata(sd_frf)
d_spindown.set_xdata(sd_freq)


print ("Saving data")
#save data to filename_ss_frf.dat and filename_sd_frf.dat
np.savetxt("%s_ss_frf.dat"%filename,
           np.transpose(np.vstack((speeds,frf))),
           delimiter=" ")
np.savetxt("%s_sd_frf.dat"%filename,
           np.transpose(np.vstack((sd_freq,sd_frf))),
           delimiter=" ")

figFRF.canvas.draw()
print ("All Done!")
#turn the motor off completely
set_voltage(0)
disable_voltage()
time.sleep(1)
shutdown()
raw_input("Press any key to exit")
```

```
def handler(signum,frame):
    print("stopping immediately!")
    time.sleep(0.5)
    set_voltage(0)
    time.sleep(10.5)
    shutdown()
    exit(1)


class SpinDownThread (threading.Thread):
    def __init__(self):
        threading.Thread.__init__(self)

    def run(self):
        print "Starting spindown... "
        # Get lock to synchronize threads
        time.sleep(7)
        print "Turning off motor!"
        set_voltage(0)

if __name__=="__main__":
    #set up the signal handler
    signal.signal(signal.SIGUSR1, handler)
    signal.signal(signal.SIGINT, handler)
    signal.signal(signal.SIGTERM, handler)
    #parse input arguments
    parser = argparse.ArgumentParser(description=
                                        "Estimate frequency response function"
    parser.add_argument('filename', help='filename for output')
    args = parser.parse_args()
    if(args.filename==None or args.filename==""):
        print("error, please specify filename")
        exit(1)
    main(args.filename)
```

C.2 VTF Generation

C.2.1 Data Specific Analysis File

```
%% i17_cf_d50_SD1
%load the data
clear all
load('i_17_cf_d50_SD1.mat');

%Call frf_function, see frf_function.m for explanation of each input
DEAD_TIME_L = 1*4000;
DEAD_TIME_R = 3*4000;
SPIN_DOWN_L = 19.54*4000;
SPIN_DOWN_R = 29.32*4000;
STEADY_STATE_L = 2000;
STEADY_STATE_R = 4000;
FS = 4e3;
RPS = 60;
filename = 'imbalanced_d50/i_17_cf_d50_SD1_1'

[i_17_cf_d50_SD1_1] = frf_function( raw, FS, RPS, filename,...
                        DEAD_TIME_L, DEAD_TIME_R, SPIN_DOWN_L,...
                        SPIN_DOWN_R, STEADY_STATE_L, STEADY_STATE_R,1);

%% Plotting
close all
figure
hold on
plot(i_17_cf_d50_SD1_1(:,1),i_17_cf_d50_SD1_1(:,2))

xlabel('Frequency (Hz)');
ylabel('Magnitude');
title('FRF Estimate');
legend('VTF');
xlim([5,RPS])

save('imbalanced_d50/i_17_cf_d50_SD1_out','i_17_cf_d50_SD1_1')
```

C.2.2 VTF Parent Function

```
function [ freq_resp_out ] = frf_function( raw, FS, RPS, PolePairs,...
                                    DEAD_TIME_L    , DEAD_TIME_R,...
                                    SPIN_DOWN_L    , SPIN_DOWN_R,...
                                    SPINDOWN_VALID_L, SPINDOWN_VALID_R,...
                                    plt, speed_extraction_method, varargin);
%% Inputs
%{
 raw : Raw data
 FS : Sample frequency in Hz, typically (4000).
 RPS : Revolutions per second the motor spins at steady state in Hz.
 PolePairs : Number of pole pairs in a motor. An easy way to determine
             this is to look at the operating speed of the motor. A 3600
            RPM motor has 1 pole pair, 1800 RPM is 2 pole pairs
 DEAD_TIME_L : Row index when the motor was off, left bound
 DEAD_TIME_R : Row index when the motor was off, right bound
 SPIN_DOWN_L : Row index when the motor starts spinning down, left bound.
               If the sensor output is clipped, the amplitude of the
               electric frequency envelope may not drop right away. It is
               important to confirm the spin down start once the two emf
               sensors have been fitted to each other, and the correct
               difference is calculated.
 SPIN_DOWN_R : Row index when the motor stops spinning, right bound.
 SPINDOWN_VALID_L : Zero Crossing, Using the zero crossing method, an
                        exponential is fit to the spindown speed envelope.
                        This boundary is a row index used to fit the
                        exponential to the spindown, and is relative to the
                       SPIN_DOWN_L value. Typically this value is between
                        200-500.
                    Hilbert Transform, Using the hilbert transform method
                        for speed extraction, this is a row index for a
                        time when the motor was spinning at steady state.
 SPINDOWN_VALID_R : Row index, right bound of the corresponding
                    SPINDOWN_VALID_L.
 plt : Boolean (0 or 1) flag to indicate whether or not auxilary plots
       showcasing the steps of the method should be displayed. 0 indicates
       no plots, 1 indicates plots.
 speed_extraction_method : valid options are 'hilbert' or 'zeroX'
%}
%%
clc
%% Round indices
DEAD_TIME_L     = round(DEAD_TIME_L);
DEAD_TIME_R     = round(DEAD_TIME_R);
```

```
SPIN_DOWN_L     = round(SPIN_DOWN_L);
SPIN_DOWN_R     = round(SPIN_DOWN_R);
SPINDOWN_VALID_L = round(SPINDOWN_VALID_L);
SPINDOWN_VALID_R = round(SPINDOWN_VALID_R);

%% Verify inputs
try
    plt;
catch
    plt =0;
end
try
    speed_extraction_method;
catch;
    speed_extraction_method = 0;
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if plt == 1;
    actual_fs = 1/mean(diff(raw(:,1)))
    pause
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
TIME_IDX    = 1;
ACCEL_X_IDX = 5;
ACCEL_Y_IDX = 4;
EMF_1_IDX   = 2;
EMF_2_IDX   = 3;
% Analysis Routine
% 1.) Process EMF Data to find electric frequency
% 2.) Only use spindown data from here on
% 3.) Find Zero Crossings and Convert to Speed Envelope
% 4.) Fit an exponential to the Speed Envelope and extrapolate out the
%     speed
% 5.) Construct virtual input from speed envelope
% 6.) Detrend Accelerations
% 7.) Build Hanning Windows
% 8.) Construct the Masked Inputs and Outputs
%% Select Speed Extraction Method
if speed_extraction_method == 'zeroX';
    [ virtual_input,accel_x,accel_y ] = extract_speed_zeroX(raw, ...
        EMF_1_IDX, EMF_2_IDX,...
        ACCEL_X_IDX, ACCEL_Y_IDX,...
        TIME_IDX,...
        DEAD_TIME_L, DEAD_TIME_R,...
        SPIN_DOWN_L, SPIN_DOWN_R,...
```

```
        SPINDOWN_VALID_L, SPINDOWN_VALID_R,...
        FS, RPS, PolePairs, plt...
        );
elseif speed_extraction_method == 'hilbert';
    [ virtual_input,accel_x,accel_y ] = extract_speed_hilbert( raw,...
    EMF_1_IDX      , EMF_2_IDX      ,...
    ACCEL_X_IDX    , ACCEL_Y_IDX    ,...
    TIME_IDX,...
    DEAD_TIME_L    , DEAD_TIME_R,...
    SPIN_DOWN_L    , SPIN_DOWN_R,...
    SPINDOWN_VALID_L, SPINDOWN_VALID_R,...
    RPS            , plt);
else;
    fprintf('No speed extraction method specified, defaulting to hilbert')
    [ virtual_input,accel_x,accel_y ] = extract_speed_hilbert( raw,...
    EMF_1_IDX      , EMF_2_IDX      ,...
    ACCEL_X_IDX    , ACCEL_Y_IDX    ,...
    TIME_IDX,...
    DEAD_TIME_L    , DEAD_TIME_R,...
    SPIN_DOWN_L    , SPIN_DOWN_R,...
    SPINDOWN_VALID_L, SPINDOWN_VALID_R,...
    RPS            , plt);
end
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Time Frequency Analysis
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Time window and overlap
tw = 1; %seconds
ov = 0.9; %percent
%Find out how many samples this is and create a hanning window
ns = round(FS*tw);
%han_win = [hanning(ns);zeros(length(virtual_input)-ns,1)];
han_win = hanning(ns);
%Calculate the number of windowing intervals given the overlap
num_wins = floor(length(virtual_input)/round(ns*(1-ov)));
%Pre allocate the time and frequency vectors
msk_input = zeros(length(virtual_input),num_wins);
msk_output = zeros(length(virtual_input),num_wins);
MSK_input = zeros(length(virtual_input),num_wins);
MSK_output = zeros(length(virtual_input),num_wins);
mFRF = zeros(length(virtual_input),num_wins);

fprintf('---Time Frequency Analysis using Hanning Window STFT---\n');
```

```
fprintf('\t constructing masks, please wait...\n');
%now build all the masks
%close all
figure
for i=0:num_wins-1
    %shift the hanning window by 1-overlap % each time
    %carefully build them so we start with a hanning window peak
    %don't wrap around (no circshift, even though it's easy :P )
    %1.) Find the center of the hanning window in the main array
    center_point = round(ns*(1-ov))*i;
    msk = zeros(length(virtual_input),1);
    mask_start_idx = center_point - round(ns/2);
    mask_length = ns+mask_start_idx-1;
    han_start_idx = 0;
    han_end_idx = ns;
    %for the beginning remove the left half of the hanning window
    if(mask_start_idx<=0)
        han_start_idx = -mask_start_idx;
        mask_length = ns-han_start_idx;
        mask_start_idx = 1;
    end
    %for the end remove the right half of the hanning window
    if(mask_length>length(msk))
        han_end_idx = ns - (mask_length-length(msk));
        mask_length = length(msk);
    end

    %.) Now add the hanning window to the mask (which is all 0's)
    msk(mask_start_idx:mask_length) = han_win(han_start_idx+1:han_end_idx);
    msk_input(:,i+1) = msk.*virtual_input;
    msk_output(:,i+1) = msk.*accel_x;
    %plot the results (every 10th window)
    if plt ==1;
        if(mod(i,10)==0)
            x = [0:1:size(msk_input,1)-1]./FS;
            subplot(3,1,1); hold on
            %title('Hanning Window Masks','Font',36);
            plot(x,msk,'Font',24)
            subplot(3,1,2); hold on
            %title('Masked Input');
            plot(x,msk_input(:,i+1));
            subplot(3,1,3); hold on
            %title('Masked Output');
            plot(x,msk_output(:,i+1));
```

```
        end
    end
end

if plt == 1;
    size(virtual_input)
    size(accel_x)
    figure
    msk =  zeros(length(virtual_input),1);
    size(msk)
    msk(ns:2*ns-1) = hanning(ns);
    % Plot Virtual Input and Mask
    x = [0:1:size(msk_input,1)-1]./FS;
    size(msk)
    for i = 1:size(msk,1);
        v_in(i) = msk(i,1)*virtual_input(i,1);
        a_out(i) = msk(i,1)*accel_x(i,1);
    end
    subplot(211)
    plot(x(1:3*ns),virtual_input(1:ns*3,1),x(1:3*ns),msk(1:ns*3,1))
    title('Virtual Input and Hanning Window')
    subplot(212)
    plot(x(1:3*ns),v_in(1:3*ns))
    title('Masked Input')
    xlabel('Time (seconds)')

    figure
    subplot(211)
    plot(x(1:3*ns),accel_x(1:ns*3,1),x(1:3*ns),msk(1:ns*3,1))
    title('Acceleration and Hanning Window')
    subplot(212)
    plot(x(1:3*ns),a_out(1:3*ns))
    title('Masked Output')
    xlabel('Time (seconds)')
    pause

end
fprintf('Step 6: Time Window Input and Output\n');
fprintf('\t Press [enter]\n');
if plt == 1;
    pause
end

%now take the FFT's of the input and output vectors
%create a frequency vector for the x-axis
```

```
freq =linspace(0,FS,length(msk_input(:,1)));

%figure out the envelope
for i=1:num_wins
    %only allow casual responses- mask frequencies below
    %the current windowed input. Find the maximum frequency
    %between RPS and 0
    %of the input and zero out everything 15 Hz below it for
    %both the input and output
    MSK_input(:,i) = fft(msk_input(:,i));
    max_freq = round(RPS/FS*length(msk_input(:,i)));
    [M,k] = max(MSK_input(1:max_freq,i));
    center_freq = freq(k);
    MSK_output(:,i) = fft(msk_output(:,i));
    MSK_input(freq<(center_freq-15),i)=0;
    MSK_output(freq<(center_freq-15),i)=0;
end

MSK_output_env=max(MSK_output,[],2);
MSK_input_env=max(MSK_input,[],2);

if plt == 1;
    figure; hold on;
    title('Calculating FFT masks');
    xlabel('Frequency (Hz)');
    ylabel('Magnitude');
end

if plt == 1;
    for i=1:num_wins
        plot(freq,abs(MSK_input(:,i)));
        hold on
        plot(freq,abs(MSK_output(:,i)));
        plot(freq,abs(MSK_input_env),'--');
        plot(freq,abs(MSK_output_env),'--');
        if(i==1)
            fprintf('Step 7: Find the envelopes of the FFT\n');
            fprintf('\t [animation]\n');
        end
        legend('Single Window Input FFT','Single Window Output FFT',...
            'Max Envelope Input','Max Envelope Output');
        xlim([0,RPS])
        ylim([0,max(abs(MSK_input_env))])
        title('FFTs of Masked Input and Outputs')
```

```
        xlabel('Frequency (Hz)')
        ylabel('Magnitude')
        pause(0.2)
        hold off
    end
end

fprintf('Step 8: Divide output by input to compute FRF\n');
figure
env_FRF=abs(MSK_output_env)./abs(MSK_input_env);
plot(freq,mag2db(env_FRF),'LineWidth',2.0)
xlabel('Frequency (Hz)');
ylabel('Vibration Magnitude (dB)');
title('Vibration Transfer Function Estimate');
legend('Vibration Transfer Function');
xlim([5,RPS])
fprintf('\t Press [enter]\n');
if plt == 1;
    pause
end
%Provide results from raw FFT division to show the
%benefit of hanning windows- considering short windows
%removes 'noise' from harmonic frequency excitations spilling over
%
fprintf('Compare Vibration Transfer Function to Straight FFT\n');
figure
RAW_input = abs(fft(virtual_input));
RAW_output = abs(fft(accel_x))./2;
plot(freq,mag2db(env_FRF),'LineWidth',2.0)
hold on
plot(freq,mag2db(RAW_output./(RAW_input)),'LineWidth',2.0);

hold on;
xlim([5,RPS])
% ylim([0,500])
xlabel('Frequency (Hz)');
ylabel('Vibration Magnitude (dB)');
title('Compare Vibration Transfer Function to Straight FFT Division');
legend('Windowed STFT (Vibration Transfer Function)','Straight FFT');
fprintf('\t All Done\n');

freq_resp_out = [freq',mag2db(env_FRF)];
end
```

C.2.3 Hilbert Transform Speed Extraction

```
function [ virtual_input,accel_x,accel_y ] = extract_speed_hilbert( data,..
    EMF_1_IDX       , EMF_2_IDX       ,...
    ACCEL_X_IDX     , ACCEL_Y_IDX     ,...
    TIME_IDX,...
    DEAD_TIME_L     , DEAD_TIME_R,...
    SPIN_DOWN_L     , SPIN_DOWN_R,...
    STEADY_STATE_L, STEADY_STATE_R,...
    RPS             , plt)
text_size = 18;
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Step 1: Process EMF Data to find the electric frequency
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
emf1 = data(:,EMF_1_IDX);
emf2 = data(:,EMF_2_IDX);
ts   = data(:,TIME_IDX);
% Find a linear fit between the emf sensors when the motor is not running
% Use this fit to scale emf2 to emf1, calculate the difference, the result
% is the EMF without the background "noise"
p = polyfit(emf1(DEAD_TIME_L:DEAD_TIME_R),...
    emf2(DEAD_TIME_L:DEAD_TIME_R),1);
emf1_fit = emf1*p(1)+p(2);
emf_diff = emf2-emf1_fit;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
close all
if plt == 1;
    figure
    plot(ts,emf1)
    hold on
    plot(ts,emf2)
    plot(ts,emf1_fit)
    plot(ts,emf_diff)
    leg = legend('EMF1','EMF2','Fitted EMF1','Difference');
    set(leg,'FontSize',text_size)
    ylabel('Voltage (V)','FontSize',text_size)
    xlabel('Time (sec)','FontSize',text_size)
    title('Back EMF Sensor Readings','FontSize',text_size)
    set(gca,'FontSize',text_size)
    fprintf('\t Press [enter]\n')
    pause
end
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
% #### Step 2: Extract speed envelope from electric frequency
%      Use hilbert transform to find the envelope of elec freq
%      process 'too much' so we can remove the ends (+/- ov)
%      this allows us to remove the startup of the low pass filter
%      and the discontinuity at the end for the Hilbert
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fprintf('Step 2:  Extract speed envelope from electric frequency\n')
ov = 300;
elec_freq_env = abs(hilbert(emf_diff(SPIN_DOWN_L-ov:SPIN_DOWN_R+ov)));
%     low pass (moving average filter)
elec_freq_env=filter(ones(200,1),200,elec_freq_env);
%    remove the overlap (cuts off the tails so we just have SPIN_DOWN
elec_freq_env = elec_freq_env(ov:length(elec_freq_env)-ov-1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if plt == 1;
    %    Plot envelope extraction routine
    figure
    hold on
    size(elec_freq_env);
    temp_ts = ts(SPIN_DOWN_L-ov:SPIN_DOWN_L-ov+size(elec_freq_env,1)-1);
    plot(temp_ts,emf_diff(SPIN_DOWN_L:SPIN_DOWN_R));
    plot(temp_ts,elec_freq_env);
    title('Envelope Extraction Using Hilbert Transform',...
        'FontSize',text_size);
    leg = legend('Electric Frequency','Envelope');
    set(leg,'FontSize',text_size)
    xlabel('Time (sec)','FontSize',text_size);
    ylabel('Voltage (V)','FontSize',text_size);
    set(gca,'FontSize',text_size)
    fprintf('\t Press [enter]\n')
    pause
end
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% #### Step 3: Convert Envelope to speed  ####
%  find the mean EMF during steady state at rated RPS
%  use the ratio of rated RPS to EMF to figure out scaling
%  constant to convert elec_freq_env to speed_env
fprintf('Step 3: Convert Envelope to speed\n');
STEADY_STATE_L;
STEADY_STATE_R;
ss_emf = mean(elec_freq_env(STEADY_STATE_L:STEADY_STATE_R));
k = RPS/ss_emf;
speed_env = elec_freq_env*k;
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if plt == 1;
    %   Plot the motor speed
    figure
    plot(ts(SPIN_DOWN_L:SPIN_DOWN_R),speed_env);
    title('Motor Speed During Spindown','FontSize',text_size);
    xlabel('Time (sec)','FontSize',text_size);
    ylabel('Speed (Hz)','FontSize',text_size);
    set(gca,'FontSize',text_size)
    fprintf('\t Press [enter]\n')
    pause
end
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% #### Step 4: Construct virtual input from speed (I) ####
%   using speed_env construct a virtual input using Chris'
%   algorithm
fprintf('Step 4: Construct virtual input from speed (I)\n');
spindown_ts = ts(SPIN_DOWN_L:SPIN_DOWN_R)-ts(SPIN_DOWN_L);
%   normalize the speed squared
virtual_amp = (speed_env.^2)./RPS^2;
virtual_input = virtual_amp.*sin(cumtrapz(spindown_ts,2*pi*speed_env));

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if plt == 1;
    %   Plot the virtual input components
    % Normalize the speed envelope to confirm we are getting
    % a squared envelope
    norm_speed = speed_env./RPS;
    figure
    hold on
    plot(spindown_ts,virtual_amp);
    plot(spindown_ts,virtual_input);
    plot(spindown_ts,norm_speed);
    grid on
    title('Virtual Input','FontSize',text_size);
    xlabel('Time (sec)','FontSize',text_size);
    ylabel('Virtual Amplitude','FontSize',text_size);
    leg = legend('Amplitude Envelope','Virtual Input',...
        'Normalized Speed Envelope');
    set(gca,'FontSize',text_size)
    fprintf('\t Press [enter]\n')
    pause
end
```

```
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% #### Step 5: Detrend acceleration to remove gravity (O)
%   extract the spindown interval and remove the mean, scale to g's
%   conversion to g's is 100mV/g
fprintf('Step 5: Convert accelerometer measurements to g (O)\n');
accel_x = data(SPIN_DOWN_L:SPIN_DOWN_R,ACCEL_X_IDX);
accel_x = (accel_x-mean(accel_x))./0.1;
accel_y = data(SPIN_DOWN_L:SPIN_DOWN_R,ACCEL_Y_IDX);
accel_y = (accel_y-mean(accel_y))./0.1;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
if plt == 1;
    % Plot the results
    figure
    hold on
    plot(ts(SPIN_DOWN_L:SPIN_DOWN_R),accel_x);
    plot(ts(SPIN_DOWN_L:SPIN_DOWN_R),accel_y);
    leg = legend('Acceleration Sensor A','Acceleration Sensor B');
    set(leg,'FontSize',text_size)
    set(gca,'FontSize',text_size)
    xlabel('Time (sec)','FontSize',text_size);
    ylabel('Acceleration (g)','FontSize',text_size);
    title('Accelerometer Measurements During Spindown',...
          'FontSize',text_size);
    fprintf('\t Press [enter]\n');
    pause
end
%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% #### Step 6: Format outputs
%   trim the outputs so we are just looking at spindown range
fprintf('Step 6: Trim the inputs to the right range\n');
L = 4000;
virtual_input = virtual_input(L:end);
accel_x = accel_x(L:end);
accel_y = accel_y(L:end);
spindown_ts = spindown_ts(L:end);
spindown_speed_env = speed_env(L:end);

colors= [         0    0.4470    0.7410;...
    0.8500    0.3250    0.0980;...
    0.9290    0.6940    0.1250;...
    0.4940    0.1840    0.5560;...
    0.4660    0.6740    0.1880;...
    0.3010    0.7450    0.9330;...
```

```
    0.6350      0.0780      0.1840];

subplot(311)
plot(spindown_ts,virtual_input,'color',colors(1,1:3));
title('Trimmed Inputs to FRF Finder','FontSize',text_size)
ylabel('Virtual Input','FontSize',text_size)
set(gca,'FontSize',text_size)
subplot(312)
plot(spindown_ts,accel_x,'color',colors(2,1:3));
ylabel('Acceleration A (g)','FontSize',text_size);
set(gca,'FontSize',text_size)
subplot(313)
plot(spindown_ts,accel_y,'color',colors(3,1:3));
ylabel('Acceleration B (g)','FontSize',text_size);
set(gca,'FontSize',text_size)
xlabel('Time (sec)')
fprintf('\t Press [enter]\n');
%%
end
```

C.2.4 Zero Crossing Speed Extraction

```
function [ virtual_input,accel_x,accel_y ] = extract_speed_zeroX(raw, ...
    EMF_1_IDX, EMF_2_IDX,...
    ACCEL_X_IDX, ACCEL_Y_IDX,...
    TIME_IDX,...
    DEAD_TIME_L, DEAD_TIME_R,...
    SPIN_DOWN_L, SPIN_DOWN_R,...
    SPINDOWN_VALID_L, SPINDOWN_VALID_R,...
    FS, RPS, PolePairs, plt...
    );
%% Step 1: Find EMF Difference
fprintf('Step 1: Find EMF Difference\n')
emf1 = raw(:,EMF_1_IDX);
emf2 = raw(:,EMF_2_IDX);

p = polyfit(emf1(DEAD_TIME_L:DEAD_TIME_R),...
    emf2(DEAD_TIME_L:DEAD_TIME_R),1);
emf1_fit = emf1*p(1)+p(2);
emf_diff = emf2-emf1_fit;
if plt ==1;
    figure;hold on;
        plot(raw(:,1),emf1);
        plot(raw(:,1),emf2);
        plot(raw(:,1),emf_diff);
        grid on
        ylabel('EMF Sensor Output (V)')
        xlabel('Time (s)')
        title('EMF Fitting')
        legend('EMF1','EMF2','EMF Difference')
    pause
end

%% Step 2: Only use spindown data from here on
fprintf('Step 2: Only use spindown data from here on\n')
data = raw(SPIN_DOWN_L:SPIN_DOWN_R,:);
emf_diff = emf_diff(SPIN_DOWN_L:SPIN_DOWN_R,:);
ts = data(:,1) - data(1,1); % Time w\ t=0 being the start of the spindown

%% Step 3: Find Zero Crossings and Convert to Speed Envelope
fprintf('Step 3: Find Zero Crossings and Convert to Speed Enevelope\n')
% Find Zero Crossings
zero_x = [];
for i=1:size(data,1)-1;
    if emf_diff(i,1)*emf_diff(i+1,1) < 0;
```

```
        % Look for 2 sequential values where the sign changes indicating a
        % Zero Crossing. From here we just assume the crossing is linear
        % and find the crossing using y=mx+b with 2 known points.
        m = (emf_diff(i+1,1) - emf_diff(i,1));
        b = emf_diff(i,1) - (m*i);
        z = -b/m;
        zero_x = [zero_x;z];
    end
end
% Calculate speed using zero crossings on emf
speed_z = [0,0]; % Initialize speed vector
for i = 1:size(zero_x,1)-1;
    % Iterate through the zero crossings. The inverse of the difference
    % in times between
    % 2 succesive zero crossings multipled by (FS/2) is equal to speed. We
    % use (FS/2), because 2 zero crossings only gives us half a period of
    % the waveform.
    t1 = zero_x(i,1);
    t2 = zero_x(i+1,1);
    z = (FS/2)/(t2 - t1);
    speed_z(i,1) = zero_x(i,1);
    speed_z(i,2) = z;
end
speed_z(:,2) = filter(ones(10,1),10,speed_z(:,2)); % Filter speed_z
% Speed Z col1 = index of zero crossings
%         col2 = estimated speed
if plt ==1;
    figure;hold on;
        plot(ts,emf_diff)
        scatter(zero_x(:,1)/FS,zeros(size(zero_x,1),1),'r');
        %ylim([-10,60]);
        grid on
        title('Clipped Sensor Difference With Zero Crossings Marked')
        legend('Voltage Sensors Difference','Zero Crossings')
        xlabel('Time (seconds)')
        ylabel('Voltage (V)')
    pause
end
%% Step 4: Fit an exponential to the speed_z vector
fprintf('Step 4: Fit an exponential to the speed_z vector\n')
s = SPINDOWN_VALID_L;
e = SPINDOWN_VALID_R;

z_fit_data = speed_z;
z_fit_data = z_fit_data(s < z_fit_data(:,1),:);
```

```
z_fit_data = z_fit_data(e > z_fit_data(:,1),:);
z_fit = fit(z_fit_data(:,1)./FS,z_fit_data(:,2),'exp1')
speed_z_fit = feval(z_fit,ts(:,1));

speed = speed_z_fit./PolePairs;

if plt == 1;
    figure;hold on;
        plot(speed_z(:,1)./FS,speed_z(:,2));
        plot(ts,speed_z_fit);
        plot(ts,emf_diff);
        ylim([-10,75]);
    pause
end
if plt ==1;
    figure
        title('Estimated Speed Compare to Spectrogram')
        subplot(211)
            spectrogram(emf_diff,4000,3000,4000,4000,'yaxis')
            ylim([0,70])
            xlim([0,20])
        subplot(212)
            plot(ts,speed_z_fit);
            ylim([0,70])
            xlim([0,20])
            grid on
            xlabel('Time (s)')
            ylabel('Frequency (Hz)')
    pause
end

%% Step 5: Build the Virtual Input
fprintf('Step 5: Build the Virtual Input\n')
virtual_amp = (speed.^2)./RPS^2;
virtual_input = virtual_amp.*sin(cumtrapz(ts,2*pi*speed));

if plt ==1;
    norm_speed = speed./RPS;
    figure;hold on;
    plot(ts,virtual_amp);
    plot(ts,virtual_input);
    plot(ts,norm_speed);
    title('Building the Virtual Input');
    ylabel('Time (secs)');
    xlabel('Virtual Amplitude');
```

```
    legend('Amplitude Envelope','Virtual Input',...
        'Normalized Speed Envelope');
    pause
end
%% Step 6: Detrend Accelerations
fprintf('Step 6: Detrend Accelerations\n')
accel_x = data(:,ACCEL_X_IDX);
accel_x = (accel_x-mean(accel_x))./0.1;
accel_y = data(:,ACCEL_Y_IDX);
accel_y = (accel_y-mean(accel_y))./0.1;

end
```

THIS PAGE INTENTIONALLY LEFT BLANK

Bibliography

[1] Department of the Navy. "Highlights of the Department of the Navy FY 2015 Budget," Information Card, Office of Budget, Department of the Navy, http://www.secnav.navy.mil/fmc/fmb/Documents/15pres/Budget_Card.pdf. Accessed: May 2015.

[2] Brandt, A., "Noise and Vibration Analysis; Signal Analysis and Experimental Procedures." 2011, John Wiley and Sons Ltd: Chichester, UK, ISBN:9780470746448

[3] Schantz, Christopher James. "Methods for Non-Intrusive Sensing and System Monitoring." Thesis, Massachusetts Institute of Technology, 2014. http://dspace.mit.edu/handle/1721.1/92173.

[4] Beranek, Leo L., and I. L. Ver, eds. Noise and Vibration Control Engineering: Principles and Applications. New York: Wiley, 1992.

[5] Rao, Singiresu S. Mechanical Vibrations. Upper Saddle River, N.J..: Prentice Hall, c2011., 2011.

[6] Feron, E., M. Brenner, J. Paduano, and A. Turevskiy. "Time-Frequency Analysis for Transfer Function Estimation and Application to Flutter Clearance." Journal of Guidance, Control, and Dynamics 21, no. 3 (April 16, 2015): 375-82. doi:10.2514/2.4269.

[7] John Donnal, Robert Cox, Uzoma Orji, Christopher Schantz, Jin Moon, Steven B. Leeb, Jim Paris, Andrew Goshorn, Kevin Thomas, Jayme Dubinsky. "VAMPIRE: Accessing a Life-Blood of Information for Maintenance and Damage Assessment." American Society of Naval Engineers Proceedings, 2012.

[8] Donnal, John Sebastian. "Home NILM: A Comprehensive Energy Monitoring Toolkit." Thesis, Massachusetts Institute of Technology, 2013. http://dspace.mit.edu/handle/1721.1/82386.

[9] Paris, James. "A Comprehensive System for Non-Intrusive Load Monitoring and Diagnostics." Thesis, Massachusetts Institute of Technology, 2013. http://dspace.mit.edu/handle/1721.1/84720.

[10] A. Mix, A. Giacomin, "Standardized Polymer Durometry."Journal of Testing and Evaluation, vol 39, no. 4, pp. 1-10, July 2011.

[11] Welch, Peter D. "The Use of Fast Fourier Transform for the Estimation of Power Spectra: A Method Based on Time Averaging over Short, Modified Periodograms." IEEE Transactions on Audio and Electroacoustics 15, no. 2 (June 1967): 70-73. doi:10.1109/TAU.1967.1161901.

[12] NAVSEA. "Naval Power Systems Technology Development Roadmap," Document, Naval Sea Systems Command PMS 320, Department of the Navy, 2013.

[13] "More Railguns and Lasers, Less Gunpowder – the Navy's Future High-Tech Weaponry." Network World, February 5, 2015. http://www.networkworld.com/article/2880535.

[14] Bartram, L., J. Rodgers, and K. Muise. "Chasing the Negawatt: Visualization for Sustainable Living." IEEE Computer Graphics and Applications 30, no. 3 (May 2010): 8-14. doi:10.1109/MCG.2010.50.

[15] J. Karlsson. "Possibilities of using thermal mass in buildings to save energy, cut power consumption peaks and increase the thermal comfort," Licentiate Thesis, Division of Building Materials, Lund Institute of Technology, Lund, Sweden, 2012.

[16] T. Newell, B. Newell, "Thermal Mass Design." ASHRAE Journal, vol 53, no. 3, pp. 70-76, March 2011.

[17] ASHRAE, "ANSI/ASHRAE Standard 55-2010: Thermal Environmental Conditions for Human Occupancy." 2010, American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.: Atlanta, GA

[18] "Extreme Weather Liner." Camel Manufacturing. Accessed June 29, 2014. http://camelmfg.com/products-ewl.php.

[19] M. Gillman. "Interpreting Human Activity from Electrical Consumption Data through Non-Intrusive Load Monitoring," S.M. Thesis, Dept. Elect. Eng. and Comp. Sci., Massachusetts Institute of Technology, Cambridge, MA, 2014.

[20] Donnal, J.S., and S.B. Leeb. “Noncontact Power Meter.” Sensors Journal, IEEE 15, no. 2 (February 2015): 1161-69. doi:10.1109/JSEN.2014.2359619.