An Analysis of Vibrations Experienced by Passengers on Champaign-Urbana Mass Transit District (MTD) Buses

Liz Zhang and Max Winga
Department of Physics, University of Illinois at Urbana-Champaign

Credit to Autumn Soliman, Daily Illini for the image

Table of Contents

Abstract

In this paper, we will examine vibrations in acceleration data collected from Champaign-Urbana Mass Transit District (CU-MTD) buses. We built data-collection devices using Arduino microcontrollers and various sensors to detect and record data with respect to acceleration, time, location, speed, temperature, and other factors. After constructing our devices and writing our data acquisition code, we conducted numerous trials in the field on MTD buses, encountering a variety of different situations and environmental factors. We also collaborated with the MTD to hold a series of controlled trials to test specific variables regarding the different bus types. Overall, we noticed interesting results concerning high-frequency vibrations possibly related to the engine on the bus. We were able to observe correlations in data between acceleration, location, speed, temperature, bus type, and numerous other factors. We ultimately found high-frequency vibrations of around 120 Hz on the hybrid and hydrogen buses, confirmed our hypothesis that the hydrogen buses are much smoother and more comfortable than previous bus types, and found some positive correlations between the acceleration of the bus and factors like speed and temperature. The results we found may be attributed to internal factors such as the engines, suspension, and HVAC systems on the buses. The findings from this research project suggest future research into correlations between bus vibrations and these characteristics of the buses, examination of the effects of these vibrations on passenger comfort, and overall more rigorously controlled trials.

Introduction

Public transportation is an integral component of the daily lives of many people around the world. Better public transportation decreases the number of cars on the road and congestion in large cities, improving commuters' lives and increasing economic productivity[1]. A key aspect of a good public transportation ecosystem is for the passenger experience to be as smooth as possible; nobody enjoys being jostled around left and right and up and down on their daily commute. Thus it is necessary to analyze the vibrations in acceleration experienced by passengers.

Champaign-Urbana is home to one of the premier public transport companies in the United States[2], the Champaign-Urbana Mass Transit District or MTD. Maintaining a fleet of 114 buses and running up to 103 at once, the MTD services the entire Champaign-Urbana area, including the 44,000 students who attend the University of Illinois at Urbana-Champaign (UIUC). As a large public transportation company, they update their fleet as technology improves to create more efficient and comfortable buses. Our goal is to examine the vibrations of these buses, both by conducting field tests to mimic true passenger experience and by holding controlled trials through collaboration with the MTD. We hope that our data and analysis will help to inform future bus model decisions, as well as provide more information overall about passengers' experiences on MTD buses.

The prompt for this project originated from a previous group taking PHYS 398 Design Like a Physicist that was studying the vibrations on Amtrak trains in Illinois. While testing their accelerometer device on the buses around Champaign-Urbana, they noticed 100 Hz vibrations with acceleration amplitudes of up to $0.5g$. As part of the same class with Professor George Gollin, we were tasked with investigating these vibrations and generally exploring the ride quality of MTD buses. In order to do this, we built two Arduino devices with the goals of recording acceleration, location, speed, and temperature data. In this paper, we will detail our methodology for the device design and code, examine the results that we obtained from over one million collected data points, and discuss the meaning of our findings and what would be interesting to look into in the future.

---

[1] *Subways, Strikes, and Slowdowns: The Impacts of Public Transit on Traffic Congestion*
[2] Gerard, W. *CUMTD awarded 'most innovative' transit agency*. WCIA.com

Methods

Goal for Data Collection Device

Our goal for the data collection device was to record the vibrational signal from MTD buses as experienced by the passengers onboard. The prompt for this project mentioned a potentially interesting 100 Hz signal, so we needed to read our accelerometer accurately enough to look for signals at this frequency scale. We also wanted to collect other data from the bus that might have some correlation with the vibrations. Specifically, we chose to record location, speed, and interior temperature data. We also keep track of other factors such as the type of bus, weather conditions, and the location of the device on the bus as we record the data.

Acceleration and GPS Recording Devices

In the beginning of the project, we focused on reducing the number of functions unrelated to reading the accelerometer that we were running in the data recording loop. The purpose of this was to increase the percentage of the vibrational signal that we recorded since we were most concerned with recording vibrations. We chose to put the GPS and temperature tracking on a separate device to allow the accelerometer to read data as often as possible.

Accelerometer Unit "A"

Unit A is an Arduino device that uses an Adafruit LSM9DS1 accelerometer breakout board to record the vibrational signal from the bus. It also utilizes an Adafruit Ultimate GPS chip and an Adafruit DS3231 real-time clock (RTC) chip to synchronize with the secondary device (Unit B). The data is then offloaded to a microSD chip for later analysis. We used an LCD and keypad as a user interface to monitor and control the device.

Adafruit LSM9DS1 Accelerometer

For this project, we chose to use an LSM9DS1 sensor as our accelerometer. This 9-DOF breakout board includes an accelerometer, gyroscope, and magnetometer packaged together. Since we only needed to collect data from the accelerometer, we disabled the gyroscope and magnetometer in order to achieve faster accelerometer reading speeds. The LSM9DS1 uses MEMS sensors to detect acceleration[3]. These sensors work via an array of variable capacitors which are allowed to move only in the direction of measurement and regulated by a spring. Based on the measured voltage which changes as a result of the distance between the plates on the capacitor changing, the acceleration is recorded.

[3]LSM9DS1 Reference Sheet - https://www.arduino.cc/en/Reference/ArduinoLSM9DS1

Figure 1

Diagram of a MEMS Sensor

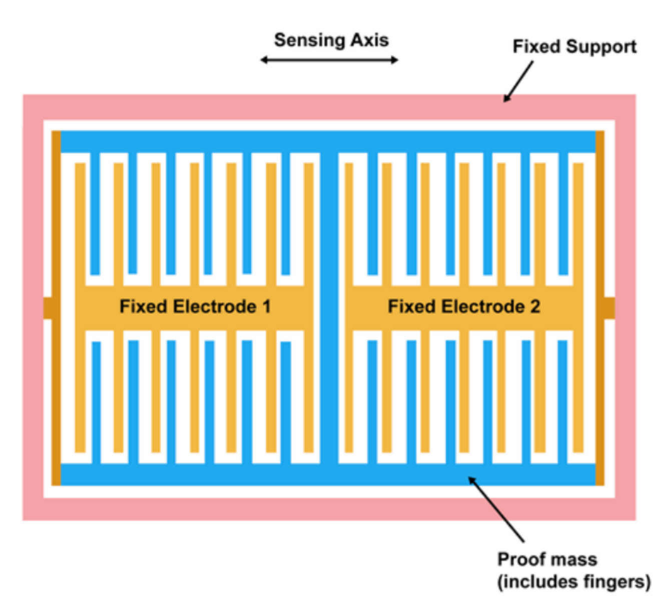

Figure 1. A representation of a MEMS sensor. The proof mass colored in blue is allowed to move to the left and right with springs resisting movement in either direction. The movement of this component changes the distance between the capacitors and thus changes the voltage across them which is measured and converted to a value of acceleration. Image from Thinkology[4].

Axis Orientation

The LSM9DS1 uses a left-handed coordinate system. We could not find any explanation for why this is the case, but it was important to note the positions of the axes relative to the bus for accurate data analysis. All of the data used for data analysis was collected with the printed circuit board (PCB) version of the device and was always oriented so that the positive x-axis was pointed to the left side of the bus (while facing in the direction of forward motion), the positive y-axis was pointed to the front of the bus, and the positive z-axis was pointed upwards. The z-axis always had an offset of +9.801 m/s^2 from the acceleration due to the gravity in Champaign, and we accounted for this where necessary during data analysis.

GPS and Temperature Unit “B”

Unit B is our secondary data collection unit, which collects time-synced GPS and temperature data from onboard the bus to supplement the acceleration data from Unit A. We collected the location and speed data using an Adafruit Ultimate GPS breakout chip. The temperature was recorded with an Adafruit BME680 sensor. We recorded this data and matched it to the accelerometer data in our analysis code using the time data recorded from the RTC for synchronization. This device also uses an LCD and keypad for the user interface.

[4] https://www.siliconsensing.com/technology/mems-accelerometers/

Components used in data collection devices

While they have different functions, both Unit A and Unit B are physically the same. We initially built the circuit using a breadboard layout, then moved on to using printed circuit boards (PCBs) in order to reduce the risk of physical damage to the device such as a wire coming out of the breadboard. The circuit diagram for the PCB was created by George Gollin and can be seen in Figure 2.

Figure 2

PCB Device Circuit Diagram

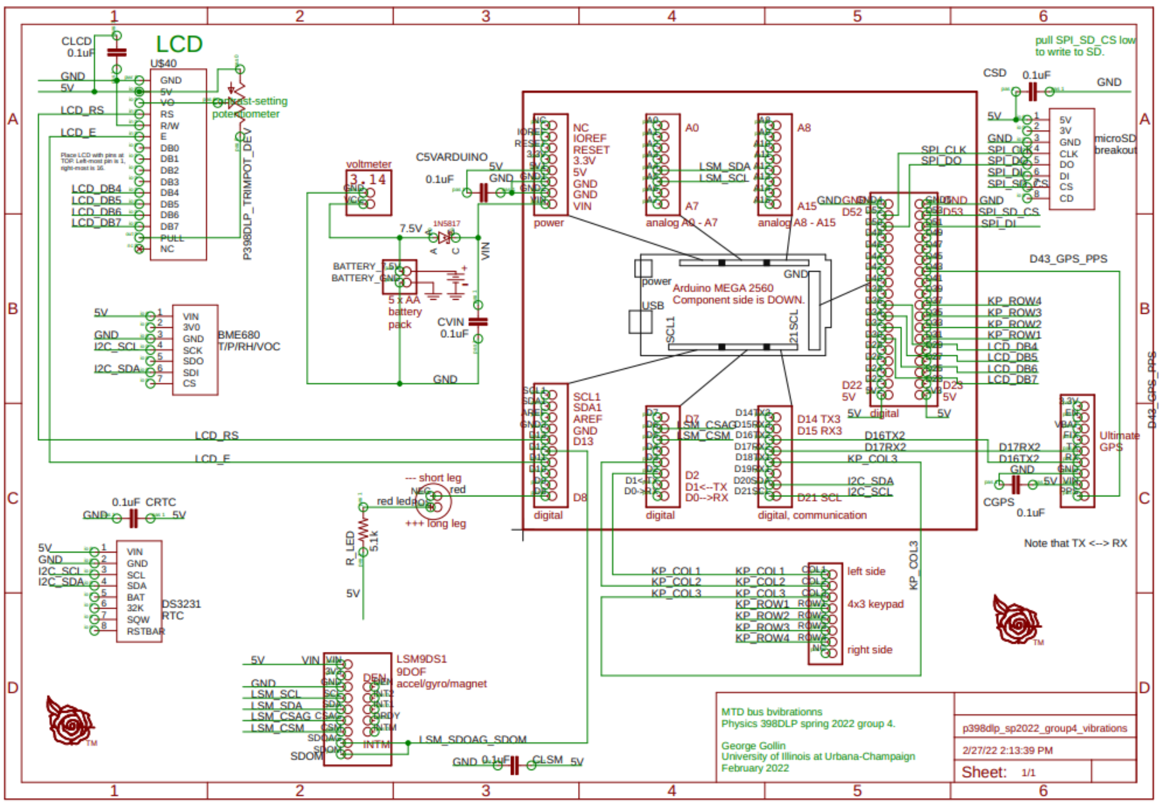

Figure 2. The circuit diagram used for the PCB for our recording devices.

Table 1

List of Device Components

#	Component Name	Purpose of component
1	Adafruit LSM9DS1 Accelerometer	For vibration reading
2	Adafruit BME680	For temperature data
3	Adafruit MEGA 2560	Processor
4	Keypad	For function selection
5	Adafruit DS3231 real-time Clock	For precise time tracking
6	LCD Display	For visual feedback
7	Adafruit Ultimate GPS Breakout v3	For GPS and time sync
8	Adafruit MicroSD Breakout	For data output

Table 1. Figures 3 and 4 use the numbering from this list to identify the components.

Figure 3

Breadboard Layout of the Device

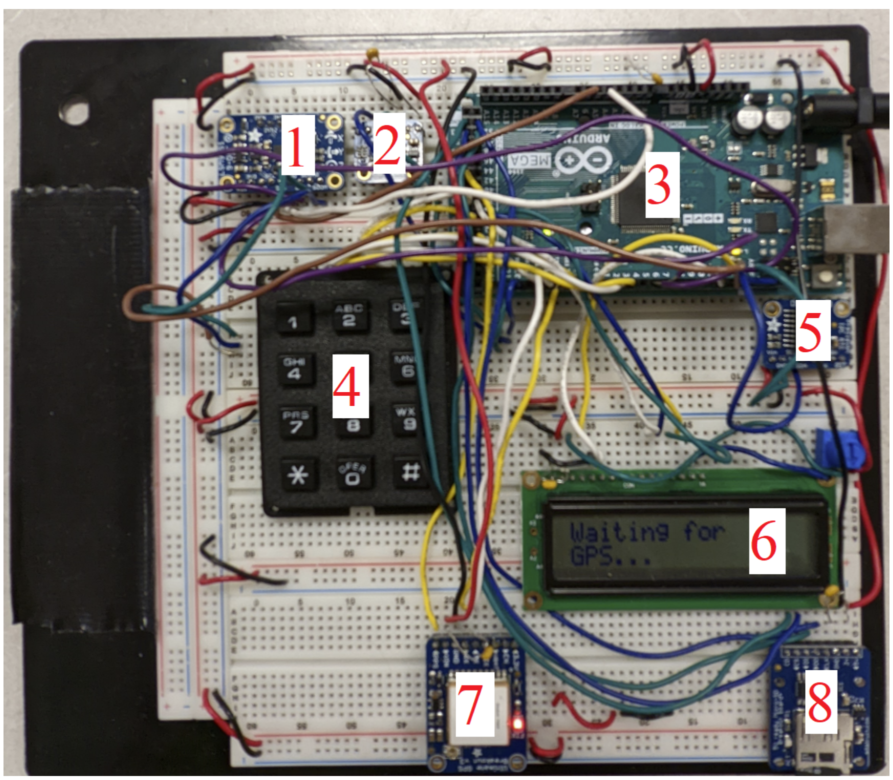

Figure 3. Here, we used a breadboard in order to experiment and easily make adjustments to the data collection device during our initial building phase. The components are labeled according to Table 1.

Figure 4

Device on Printed Circuit Board (PCB)

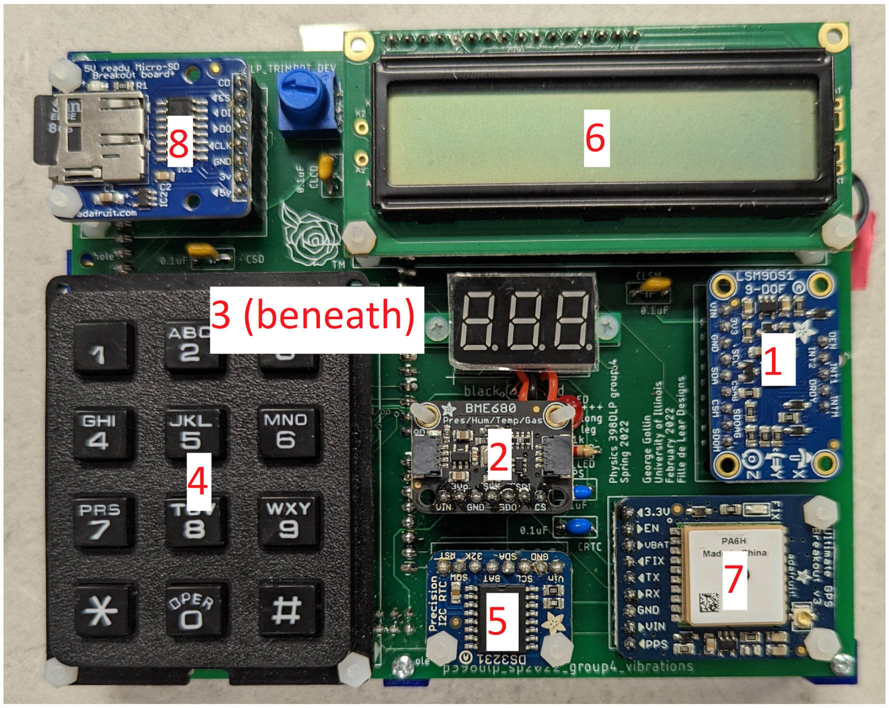

Figure 4. The final device version on a PCB. The components are labeled according to Table 1.

<u>Data Acquisition (DAQ) Code Overview</u>

This section will serve as a brief overview of how our DAQ code functions and the design decisions that we made around it. For background, Arduino code runs two main functions that are written by the user. There is a setup function that runs once when the program starts, followed by a loop function that loops continuously.

Unit A: Setup

The setup function does as its name implies and runs the setup functions for all of the components used in the device. It also syncs the real-time clock with the GPS-provided time. This syncing is important because it allows us to line up the data accurately between Unit A and Unit B

Unit A: Loop

Our loop function waits for keypad input to determine which of the secondary functions run. Pressing a key on the keypad will start the corresponding function.

[1] recordData()

Function 1 records accelerometer data. First, it creates a comma-separated list (CSV) file on the microSD where the data is saved. It then syncs the real-time clock (RTC) time with the Arduino's built-in function to record milliseconds since compiling [millis()]. Saving the time immediately before and after reading the accelerometer allows us to timestamp each reading to the millisecond, as the read loop runs at a very consistent 1.435 ms per point of accelerometer data. The snippet of code shown in Figure 5 is the main loop which reads the accelerometer and offloads to the microSD.

The reason that we don't continuously read the accelerometer and offload the data afterwards is that the Arduino MEGA 2560 only has 8 KB of memory, and the array to store the accelerometer data fills up the available memory very quickly. This means that we need to take some time to offload the data from the array in the Arduino's memory onto the CSV file on the microSD. If we were to offload directly to the microSD without using the array, this would decrease the read speed and we would lose signal clarity. This leads to the explanation for two very important terms for our data collection: "datapoints" and "cycles".

Datapoint: One specific measurement of the acceleration in the x, y, and z directions. A datapoint requires 3x16 bits of memory[5]. Each datapoint takes about 1.435 ms to collect.
Cycle: A single iteration of the (accelerometer read loop)→(offload to microSD) algorithm. Each cycle contains numerous datapoints, and data from multiple cycles are stored in the same CSV file. One cycle takes about 450 ms to collect datapoints and about 1.4 seconds to offload data to the microSD.

We mostly kept the number of datapoints per cycle constant at 300, but for the testing done towards the end of our data collection period, we increased this rate to 350 datapoints per cycle to allow us to look at a slightly wider range of frequencies in our Fourier analysis. We adjusted the number of cycles per iteration of the recordData() function regularly in order to adjust the total amount of time that the device ran based on when we were getting on and off the buses.

In order to study the full spectrum of vibrational frequency, we needed our accelerometer data to be sampled at the highest rate possible. Initially, using the default Adafruit LSM9DS1 library and the I2C communication standard, we were locked at a frequency of 119 Hz, which was too low to properly read the 100 Hz signal that we anticipated finding (from the results of the previous

---

[5] *LSM9DS1 Datasheet*. ST. (n.d.) - https://www.st.com/resource/en/datasheet/lsm9ds1.pdf

group which had led to the prompt for this project). By editing the Adafruit LSM9DS1 library to disable the magnetometer and gyroscope, which reduced the computational cost of each reading of the LSM9DS1, we managed to increase the sampling rate of the accelerometer to approximately 698 Hz. We also switched to using software SPI communication with the breakout board, which further increased the sampling rate. The difference in clarity made by increasing the sampling rate is shown in Figure 6.

Figure 5

Snippet of Data Recording Loop for Unit A

```
for (int c = 0; c < cycles; c++) {                // Loop for set number of cycles
  millis1 = millis();                             // Save time right before reading begins
  for (int i = 0; i < datapoints; i++) {          // Loop for set number of datapoints
    lsm.read();                                   // Read the accelerometer
    lsm.getEvent(&a, &m, &g, &temp);              // Get the data from the reading
    accel_data[i][0] = a.acceleration.x;          // Save the x value into the array
    accel_data[i][1] = a.acceleration.y;          // ..y
    accel_data[i][2] = a.acceleration.z;          // ..z
  }                                               //
  millis2 = millis();                             // Save time right after reading ends
  //////////////////print to microSD///////////////////
  for (int j = 0; j < datapoints; j++) {
    x = accel_data[j][0] - x_off;                 // Adjust for linear offset
    y = accel_data[j][1] - y_off;                 // The offset is prerecorded and
    z = accel_data[j][2] - z_off;                 // saved in the x,y,z_off variables
    myFile.print(millis1); myFile.print(",");     // Print data into the CSV
    myFile.print(millis2); myFile.print(",");     // ..
    myFile.print(x, 6); myFile.print(",");        // ..
    myFile.print(y, 6); myFile.print(",");        // ..
    myFile.print(z, 6); myFile.println();         // ..
  }
```

Figure 5. Code from Function 1 which records acceleration data from the LSM9DS1 and offloads it to the microSD.

Figure 6

698 Hz (red) vs. 119 Hz (blue) Reading of a 100 Hz Sine Curve (black-dashed)

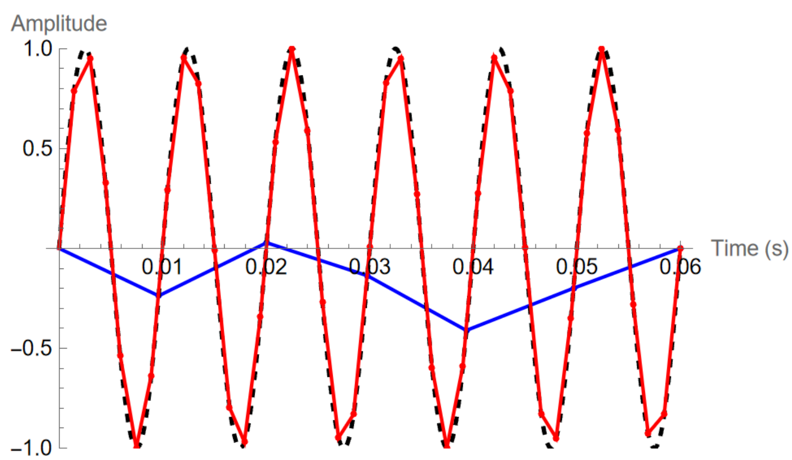

Figure 6. This graph demonstrates why increasing the sampling rate of our accelerometer is essential to analyzing the full vibrational spectrum of the buses. The red and blue curves are the result of reading points from the sine curve at different frequencies. The blue 119 Hz reading doesn't come close to properly representing the complexity of the basic 100 Hz sine curve, while the red 698 Hz reading clearly outlines the important details of the curve.

[2] calibrateAccelerometer()

Function 2 is used for calibrating the accelerometer. Through testing, we found the error on the LSM9DS1 to be mostly constant and linear with a bit of noise as shown in Figure 7, so adjusting for it was trivial; we simply needed to find the value of this offset and adjust for it when recording data. We accomplish this by placing the unit level on the ground with the z-axis facing upwards. Using the fact that the acceleration from gravity is constant in the z-axis direction, we can read the accelerometer, average the values read, and then adjust the x and y axes to 0 m/s^2 and the z-axis to 9.801 m/s^2 (local value of *g* in Urbana-Champaign). These values are stored in the x_off, y_off, and z_off variables and the adjustment happens in the data recording loop as seen in Figure 5.

Figure 7

Acceleration Data from Stationary LSM9DS1

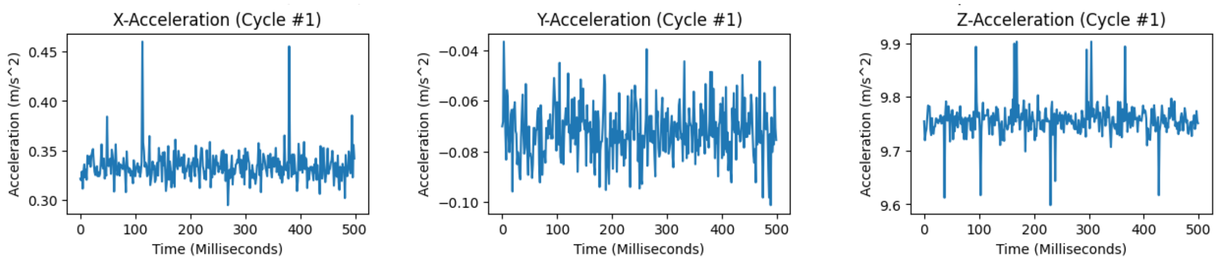

Figure 7. This graph shows the small fluctuations in data collected from the LSM9DS1 while it is completely stationary on the floor. These variations are relatively small in magnitude, but still require calibration to account for. Notice that while the x- and y-axes are approximately centered at zero as expected, the z-axis is centered at around 9.801 m/s^2 due to gravity.

[3] displayInfo()

Function 3 displays the current number of cycles per loop and datapoints per cycle to the LCD.

[4] setDatapoints()

Function 4 was used to set the number of datapoints per cycle while the program was running; however, due to memory issues with the Arduino, we disabled it.

[5] setCycles()

Function 5 is used to change the number of cycles per loop of the main data-recording function.

Unit B: Setup

The setup function for Unit B is similar to Unit A's setup function. It sets up all of the necessary breakout boards and syncs the RTC with the GPS.

Unit B: Loop

The Unit B loop is much less complicated than the Unit A loop. It only has one function, which records GPS latitude, longitude, and speed along with temperature from the BME680. These data are saved in CSV format on the microSD along with timestamps (which are accurate to the millisecond, since they are synced using Arduino's millis() function) for correlation with data from Unit A. While Unit B is the less complex of the two devices, we have had the most issues with data from this unit.

We have had several different issues with the GPS printing out data properly. We are unsure of how these came to be and have had incredible difficulty nailing down the causes or even being able to reproduce them. As a result, only some of the Unit B data we recorded has usable GPS data attached to it. Luckily, when we have problematic data it is easy to spot, so we can still use what is good. Additionally, even when the GPS has failed, the BME680 has remained consistent and has never had an issue[6]. Although these problems have resulted in our location analysis being very limited, our data from Unit A is still useful on its own without location and speed data associated with it.

Data Collection Methodology

Outside of the information that is directly recorded with the devices, we also considered other variables that may have some effect on the vibrations, such as the type of bus, the weather, the location of the device on the bus, and various events that occurred during the data collection.

Bus Type

MTD currently runs 3 main types of buses: Diesel, Hybrid (40 ft single body and 60 ft articulated), and Hydrogen. We took note of which type of bus we were recording on under the assumption that the different models would have different vibrational profiles due to changes in engine type and suspension. MTD is currently phasing out the diesel buses and introducing new hydrogen buses into its fleet. Along with the decreased engine noise, all buses from 2020 onwards (both hybrid and hydrogen) have newer adaptive pneumatic suspension, which allows for a much smoother ride over rougher terrain versus the older airbag suspension.

[6] Bless you BME680!

Weather

We took note of the weather conditions as we recorded data in the event that rain or snow would have an impact on the vibrational spectrum. Unfortunately, we were unable to get enough variety in weather conditions to analyze on this basis.

Device Location on Bus

The consideration of where to place the device on the bus was important since we noticed that different areas of the bus experienced vibrations differently. We chose to record our data with the device taped to the floor of the bus. The tape prevented the device from moving and being on the floor allowed us to get the best signal from the bus. We recorded the vibrations in various locations inside the buses to get a fuller picture of the bus. Most of the vibrations that we recorded were likely produced by components on the bus such as the engine, drivetrain, suspension, temperature control, cooling system for the batteries, etc. These components are located in different locations on the different types of buses, so changing locations in the bus could cause a significant difference between vibrations in the front and back of the same bus.

Notes on Conditions During Testing

In addition to the more specific categories, we took assorted notes on what happened during the testing. These were events such as running over particularly rough sections of road, disturbances to the device, if we heard the air conditioning turn on, or anything else we deemed relevant.

Collaboration with MTD

A key aspect of the scientific method is controlling certain variables to better understand the relationships between all aspects of a system. Unfortunately, this was difficult when testing buses in the wild as they went about their regular routes. Traffic changes, some stops are missed, and pretty much nothing remains consistent. In order to control some of these variables and to learn more about the buses, we contacted Rebecca Bolt and Josh Berbaum from MTD. Initially, we spoke over a video call about the company and, most importantly, the buses. Speaking with them was how we learned about the different models, suspension types, and locations of certain noise-producing components. Additionally, they extended the offer for us to visit the MTD Garage and run some controlled tests on the buses, especially the diesel and hydrogen buses which are difficult to find in the wild. When we went to MTD we had a few specific goals. We first rode around their practice route twice on both a diesel and hydrogen bus to compare the vibrations between them along the same route. We also rode on a hybrid bus at specific speeds to observe the effect that speed has on the bus while on the same road surface.

Results

Introduction

This section of the paper contains the results of our data analysis. We will walk through our findings and point out interesting aspects of what we found. Two of the main properties of the vibrations that we will focus on are the Fourier response and the root-mean-square deviation (RMSD). The Fourier response will tell us about the frequencies that are the strongest in the overall signal and how much overlapping noise there is. The RMSD of the acceleration is the metric we will use to measure the "bumpiness" of the ride.

Fourier Transforms

The Fourier transform is a key element in analyzing any oscillatory function. It decomposes the waveform into the frequencies that make it up. We expect the vibrational signal from the bus to largely consist of a superposition of oscillatory functions, and by looking at the Fourier response we can observe exactly which frequencies make up the signal. To convert from real space to Fourier space, Equation 1 is used.

Equation 1
Fourier Transform

$$\widehat{f}(\omega) = \int_{-\infty}^{\infty} f(t)e^{-2\pi it\omega}dt$$

Here, $f(t)$ is the function which we are converting and ω is the frequency. Notice that the resulting $\widehat{f}(\omega)$ is a function of ω. The amplitude of the function is called the Fourier response and corresponds to how strongly the given ω occurs in $f(t)$.

The process of calculating the Fourier transform is a bit different when doing it for a limited number of discrete data points such as with our data. The main restriction this brings is that we can only look at frequencies with corresponding periods longer than the time between each data point and shorter than the overall time of the reading. We use the Python library NumPy's fast Fourier transform (numpy.fft.fft[7]). For our data, since we have about 300 datapoints per cycle, we can only look at frequencies between ~2 Hz to ~350 Hz. The resulting Fourier transform can show us whether the signal is primarily composed of a few discrete frequencies overlapping or a noisy signal composed of many frequencies, and which frequencies are the strongest.

---

[7] numpy.fft.fft - NumPy v1.22 Manual. (n.d.) - https://numpy.org/doc/stable/reference/generated/numpy.fft.fft.html

Root-Mean-Square Deviation (RMSD)

The root-mean-square deviation is the metric through which we are judging the "bumpiness" of the ride of the bus. RMSD is a measure of how much the data varies from the mean. Because the vibrations are oscillating, we cannot just look at the average of the height of the peaks because it will simply equal the mean. When we use RMSD, we are taking the RMSD of the acceleration in a specific direction for each cycle and then mapping over time. This is specifically to examine the relative scale of the oscillations. We are removing the mean so any constant acceleration from gravity or from the bus turning will not change the RMSD. The RMSD will only relate to the amplitude of the oscillations and will not be affected by constant accelerations such as from gravity or the bus going around corners. A higher RMSD means that the amplitude of the acceleration vibrations is bigger. Equation 2 defines RMSD.

Equation 2

Root-Mean-Square Deviation

$$RMSD_{cycle} = \sqrt{E[a^2] - E[a]^2}$$

$$Where\ E[a] = \frac{1}{n}\sum_n a_n\ ,\ \ E[a^2] = \frac{1}{n}\sum_n a_n^2\ ,\ \ 1 \le n \le datapoints\ in\ cycle$$

Terminology

We have previously defined some of these words but for reference, these are the key terms that will be used in our data analysis.

Datapoints: One specific measurement of the acceleration in the x, y, and z directions.
Cycles: A single iteration of the (accelerometer read loop)→(offload to microSD) algorithm.
Run: The file created by the main data collection function is called a run and it consists of several cycles, usually between 30 and 100.
Ride/Day: Sometimes we will view data in terms of a day or a full ride which will be a collection of runs on the same bus.

Note on Calibration

We have noticed while looking back at some of our data that it appears that the unit we were using was not calibrated properly. Since the LSM9DS1 has a small linear offset in the error, this just means that our data which should be centered on zero (or 9.801 m/s^2 for the z-axis) is slightly above or below it. This won't have an effect on our Fourier or RMSD analysis, since we normalize our data by subtracting the mean before conducting each of these analyses.

Figure 8.

Sample Acceleration Data and Fourier Response, 3/27/22

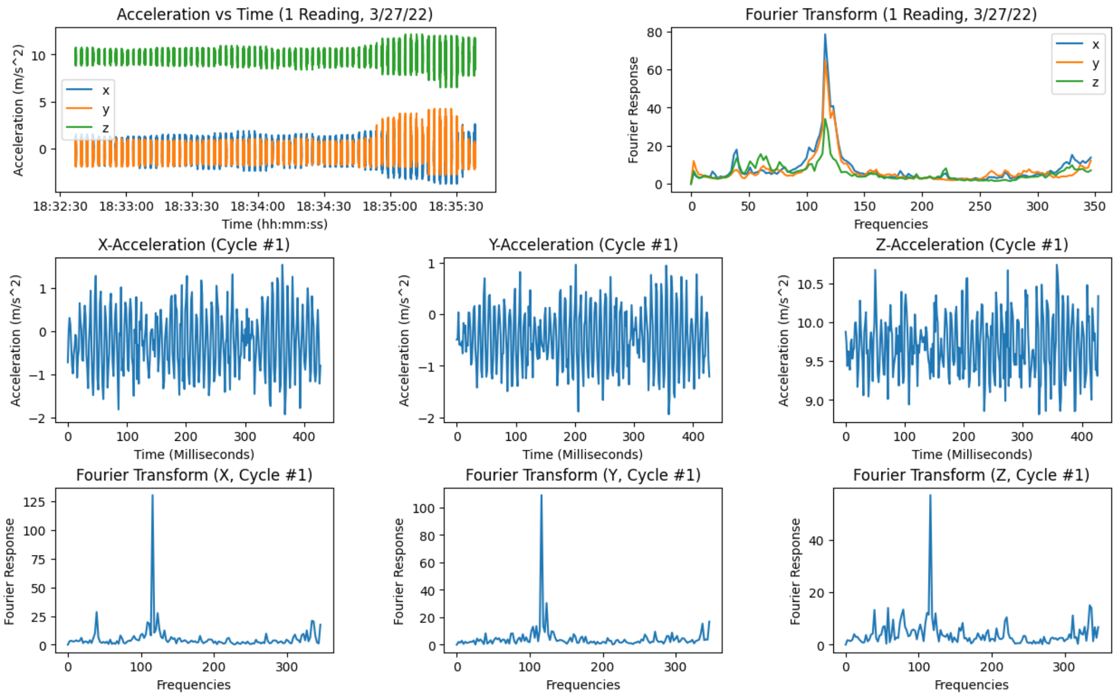

Figure 8. These graphs represent a sample of the data we collect with our Unit A devices. The plot on the top left shows accelerations in the x, y, and z directions over time (note that each of the vertical lines is the contribution of one cycle of measurement and the gaps are from when the device wasn't recording, this graph does not show any periodicity of the acceleration). The corresponding plot to its right depicts the Fourier transform of these data. The six plots below show the acceleration in the x, y, and z directions for a single cycle, as well as the Fourier transforms over the respective axis.

From an initial overview of our hybrid bus field data, we can immediately see some promising results regarding individual peak frequencies in our Fourier analysis. From the Fourier transforms in Figure 8, we can clearly observe a strong peak frequency at about 120 Hz, which aligns with our prior expectation of approximately 100 Hz, 0.5*g* vibrations. The amplitudes of these vibrations seem to vary by about ± 1 m/s^2 from the mean. Another interesting feature of this data is the presence of double peaks in the Fourier transform, which is visible in the acceleration-time graphs as beats with a frequency of about 6.67 Hz (see the x-acceleration subplot of Figure 8 for a clear example of this.) Besides the prominent frequency peaks in the Fourier analysis, a key feature of this data is the offset of the means from their expected values. While the mean of the z-acceleration appears to be around 9.8 m/s^2 (the anticipated value), the means of the x- and y-acceleration seem to be offset from zero by about -0.5 m/s^2. This is likely because the calibration was slightly off for the accelerometer during this reading.

Note that Figure 8 is not necessarily representative of our aggregate data, and is merely presented as a sample of the type and order of magnitude of the data that we have collected. Readings range from 10-200 cycles and cycles range from 300-350 datapoints across our data.

Figure 9

Overall Fourier Transform Comparison – Across Days

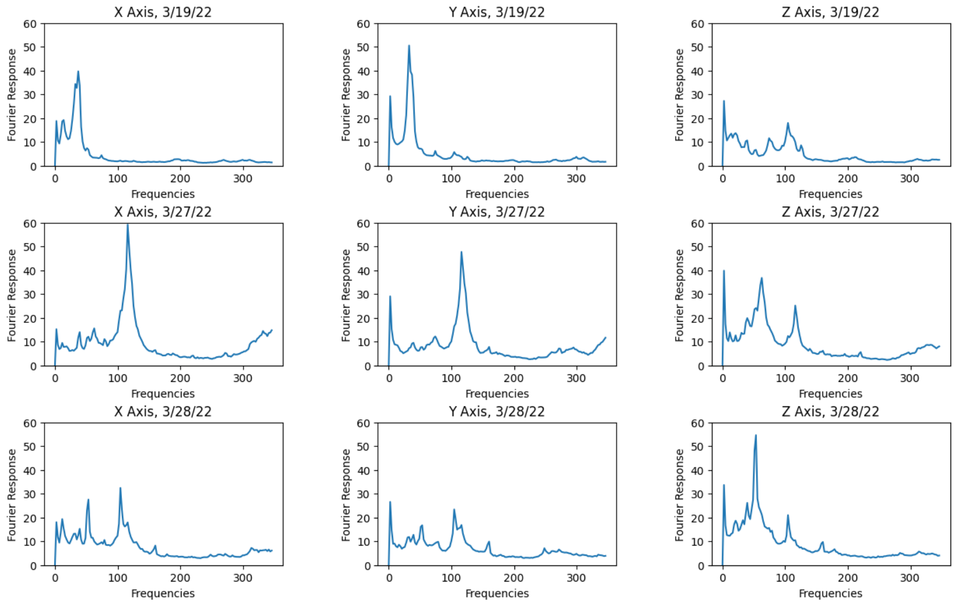

Figure 9. This figure shows Fourier transforms of our acceleration data across different days, for each of the x, y, and z axes. These overall Fourier transforms are individually calculated for each cycle, then averaged over a full day. Each day has approximately 10 readings, and each reading has approximately 70 cycles. These data were all collected on 40 ft hybrid buses.

Figure 9 presents a more holistic view of the data collected from our hybrid bus trials, spanning across different days, buses, and bus routes. Since these plots average across many differing bus conditions, any features that we can view and extrapolate here are indicative of significant vibrational frequencies that are present on many different runs, and could be reasonably expected to be felt by passengers. Although there is no single clear peak frequency present across all days and all axes, a few distinct frequencies stand out as significant. All three of the z-axis graphs, for instance, show some sort of frequency peak at about 120 Hz, similar to the peak found in Figure 8, as well as a lower-frequency signal at about 70 Hz. These frequency peaks can be seen on the other plots as well, in addition to an even lower 40 Hz peak in the x- and y-axis plots. Finally, all of the subplots show a peak at an extremely low frequency of about 2 Hz, although this can most likely be attributed to noise in our data rather than some significant feature of the bus ride itself.

Figure 10

Overall Fourier Transforms Comparison – Bus Types, Moving

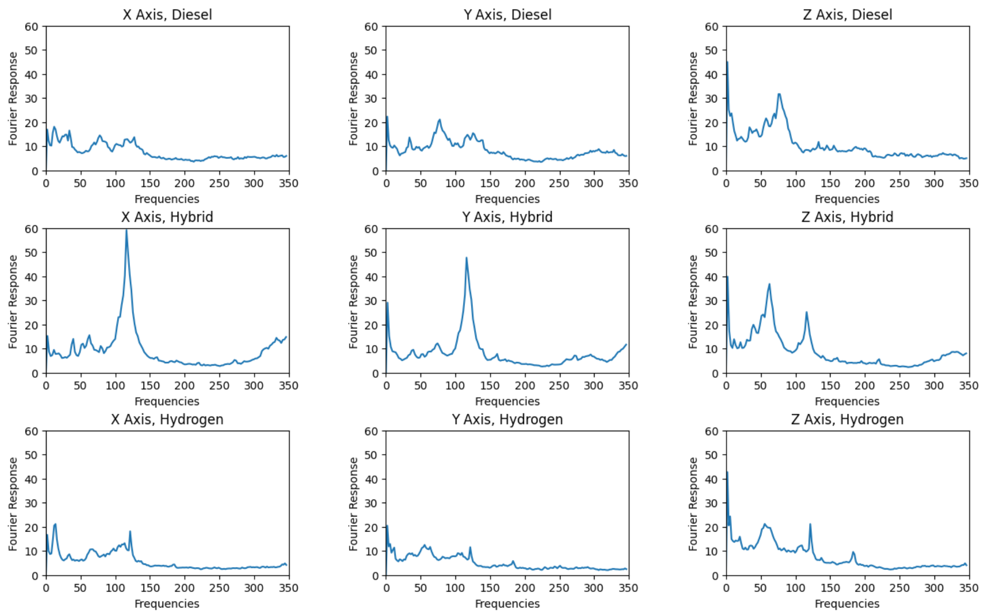

Figure 10. This figure shows averages of Fourier transforms of our acceleration data across different bus types, calculated in the same manner as Figure 9. Note that in this case, each bus type had about 6 corresponding readings. The data for these plots were all collected when the buses were in motion and driving at significant speeds along Champaign-Urbana roads, as opposed to sitting idle.

Next, we move from an overview of our field data to a comparative study between our hybrid bus field tests and the controlled tests we conducted with the MTD on their diesel and hydrogen buses. Figure 10 shows a similar Fourier analysis to Figure 9 but spans different bus types as opposed to different days. For our controlled diesel and hydrogen bus trials, we attempted to control variables such as bus route and speed; therefore, differences in our data can most likely be associated with some features of the buses themselves, such as the engine or the suspension system, rather than outside variables. Once again, besides the consistent 2 Hz peaks from noise in the data, there do not seem to be any strong similarities or consistent trends in the Fourier transforms of the different bus types. In general, the diesel and hybrid buses show a mix of low-frequency and high-frequency (over 250 Hz) signals, while the signals from the hydrogen bus seem to be very concentrated below 150 Hz.

Figure 11

Overall Fourier Transforms Comparison – Bus Types, Idle, Scaled

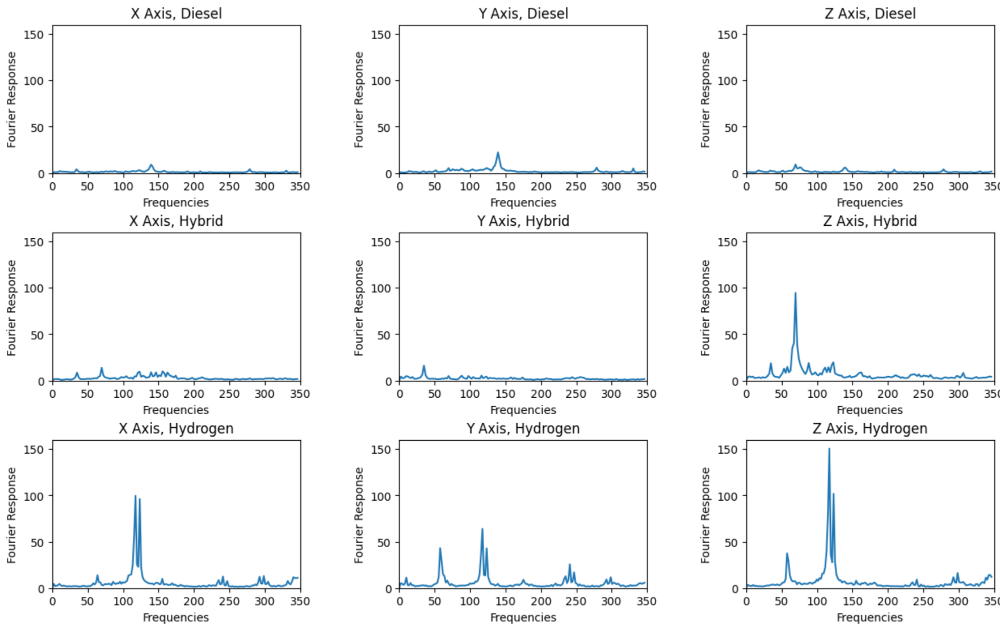

Figure 11. Similarly to Figure 10, the data for these Fourier Transforms are taken from the MTD's three different bus types: diesel, hybrid, and hydrogen. These plots, however, depict data collected when the bus's engines were turned on but the buses themselves were not moving; therefore, none of the acceleration/vibrational frequencies can be attributed to the movement of the bus. Note that in this figure, the y-axes of the subplots are all scaled equally from 0 to 150, in order to accurately represent the comparative magnitude of the Fourier responses.

Figure 12

Overall Fourier Transforms Comparison – Bus Types, Idle, Unscaled

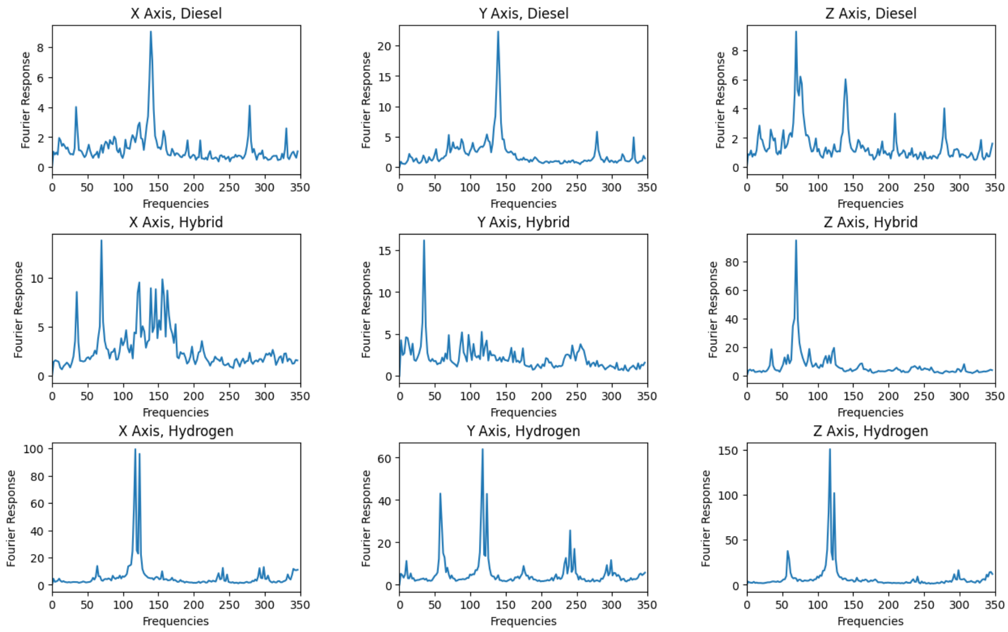

Figure 12. This figure shows the same data as Figure 11, but on these subplots the y-axes are not set to the same scale, but scaled relative to the range of the depicted dataset. This is in order to better observe the specific nuances of the frequency peaks for each bus type and axis, since Figure 11 may obscure some of these details.

While the analysis of the vibrations of different bus types while moving in Figure 10 did not necessarily show any strong trends, Figure 11 and Figure 12 show much stronger trends across the different buses when they are idle (unmoving but with their engines turned on). From the unscaled Fourier transform graphs in Figure 12, we can clearly see that the number of large frequency peaks decreases with each type of bus. The diesel buses have a strong frequency peak at about 140 Hz across all axes, and smaller peaks at around 40 and 270 Hz. The z-axis for the diesel buses especially features numerous distinct peaks in the Fourier transform, more than we have seen for any other plot. While the x-axis plot for the hybrid bus also seems to feature many peaks, especially in the 100-200 Hz range, the y- and z-axis plots only show strong peaks in a few frequencies. Most interestingly, the Fourier transforms for the hydrogen bus show a few extremely strong peaks, especially at 120-130 Hz. Looking at the equally-scaled plots in Figure 11, we can further observe that this frequency peak is extraordinarily strong, dwarfing the peaks we viewed from the diesel and hybrid buses. This indicates that the vibrations in the hydrogen bus were much clearer and at specific frequencies than the other bus types, which had much more distorted signals.

Figure 13

Acceleration-Time Graph and Fourier Response for Idle Hydrogen Bus

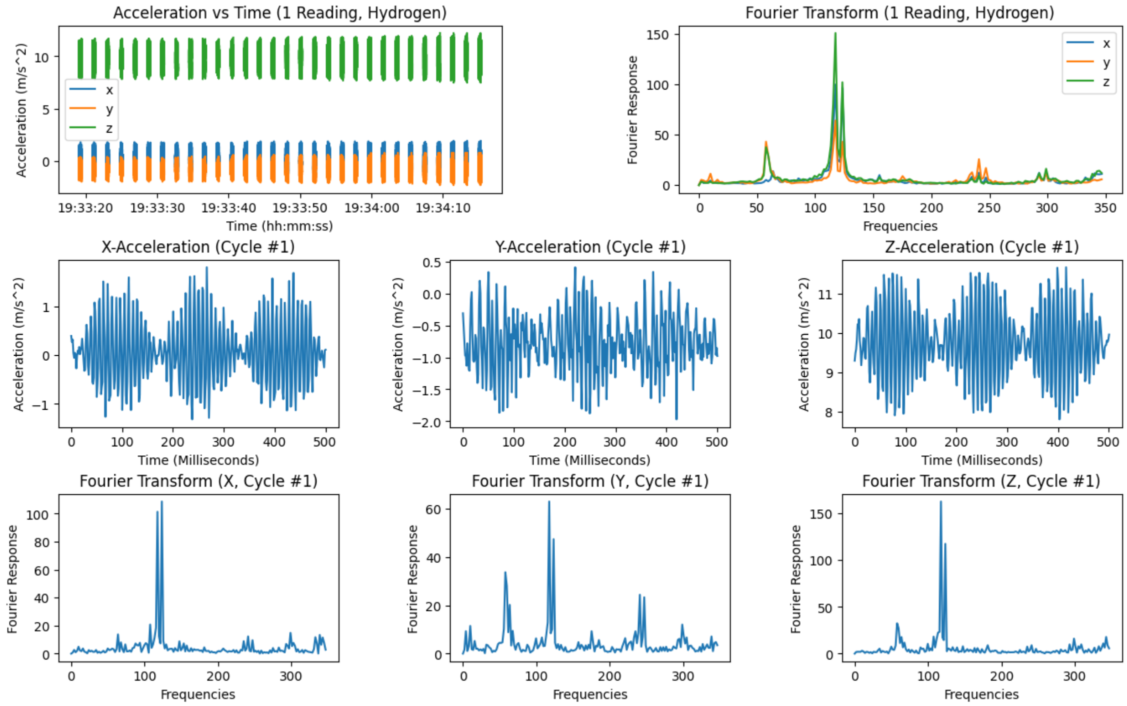

Figure 13. Here is the acceleration vs. time and Fourier transform breakdown of the idle test on the hydrogen bus seen in Figure 10. The top left plot is the full reading and the split x, y, z plots are only of cycle one. During this run the device was taped to the floor of an idling hydrogen bus. The single-cycle Fourier transforms are seen in the bottom row scaled individually. Note again that each of the vertical lines in the graph on the top left is the contribution of one cycle of measurement and the gaps are from when the device wasn't recording, so it does not show any periodicity of the acceleration.

Since the Fourier transform for the idle hydrogen bus in Figure 11 showed an extremely interesting strong peak frequency at 120-130 Hz, we isolated the acceleration graph for this specific reading in Figure 13 in order to directly observe the effect of these high-frequency close signals on the actual acceleration-time graph. Similarly to Figure 8, beats of about 6 Hz are clearly visible in the individual-cycle acceleration graphs (especially in the z-axis). This corresponds to the difference between the two strong peaks in the Fourier analysis. The extreme clarity of these signals suggests not only that the bus inherently experiences some sort of 120-130 Hz vibration, but also that there is very little noise on the hydrogen bus that would obscure these distinct signals.

Figure 14

Acceleration RMSD Comparison Across Bus Types

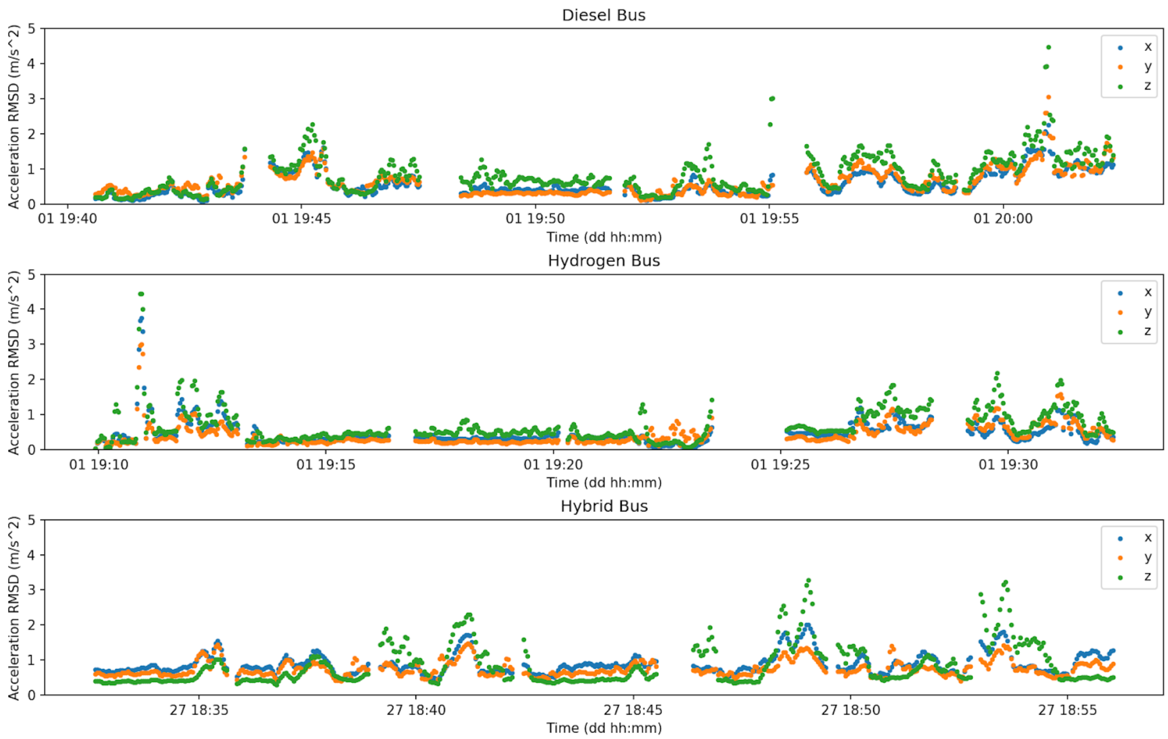

Figure 14. To create this figure, we plotted the RMSD in acceleration for each cycle, reading, and specific bus type over time. We separated the RMSD for the x, y, and z acceleration data. A higher acceleration RMSD indicates that the bus ride was bumpier.

Comparing the vibrations between the types of buses was one of the things we were most interested in studying. The top and middle plots from Figure 14 are from diesel and hydrogen buses respectively and were taken along the same test route from when we collaborated with MTD. The start of the timelines for each plot is slightly offset but observing the general pattern is interesting because it is very clear that they were traveling over the same road. The bottom plot is from a hybrid bus on a different route so it is not as applicable for the comparison, but still offers an interesting comparison between the relative acceleration RMSD of each bus type.

The beginning of the hydrogen bus plot has a very large spike. We don't know exactly what event may have caused it, but the rest of the plots are unaffected. The hydrogen bus (other than the initial peak) maintains a much lower acceleration RMSD than the diesel bus and the hybrid bus. This confirms our experience that the hydrogen buses are much smoother than any of the other models. Further testing over specific terrain and comparing against all three bus types would be interesting in the future to determine how the buses handle over all types of terrain. Additionally, for the diesel and hydrogen buses, we conducted our tests on different locations of the bus; the first third of data were in the front of the bus, the next third were in the middle, and the final third were from the back of the bus. Again, although there are clear differences between each of these sections of our data (for instance, the middle third of each run seems to be more stable than the other two), it is difficult to directly attribute these differences to the different sections of the bus, since other confounding variables, such as the road conditions, are also present.

Besides comparing the different bus types, it is also interesting to consider the differences in acceleration RMSD for different axes. From Figure 14, it appears that across all bus types, the z-axis consistently experiences the most change in acceleration RMSD, indicating that it is the "bumpiest". Changes in acceleration RMSD in the x and y direction are to be expected since these occur when the bus turns, starts, and stops; however, changes in the z-axis acceleration RMSD are not necessarily inherent to the bus ride and are most likely caused by some outside factor.

Figure 15

Z-Axis Displacement of Different Bus Types

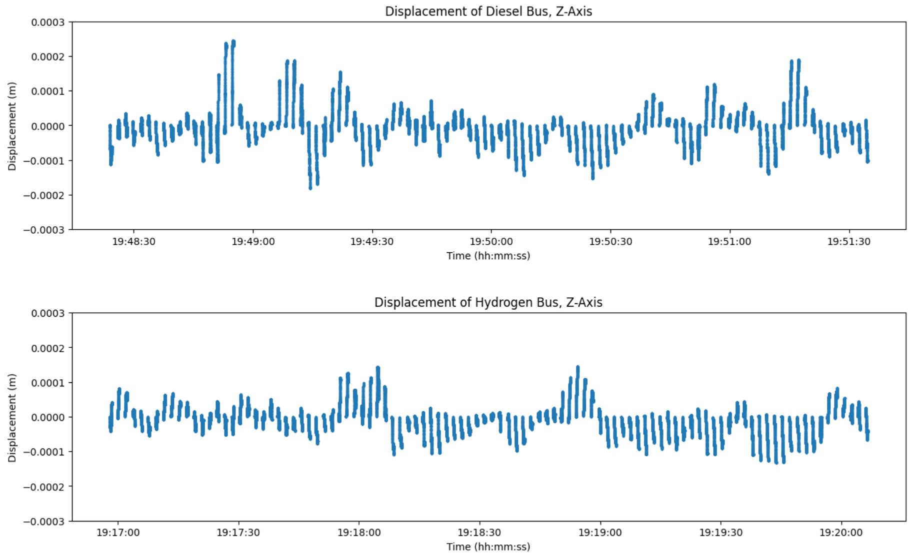

Figure 15. To generate these graphs, we converted our acceleration data into the physical displacement of the bus (measured in meters) by using Equation 3. Note once again that each of the vertical lines in the graph on the top left is the contribution of one cycle of measurement and the gaps are from when the device wasn't recording, so it does not show any periodicity of the acceleration. This is specifically run #3 from the data recorded shown in Figure 14.

Although we chose acceleration RMSD as our variable of choice to measure the bumpiness of a passenger's ride, we thought it would also be useful to examine the displacement experienced by the passengers as well, providing another metric for our analysis. To convert our acceleration data into displacement data, we used Equation 3 as an extremely rough estimate of a double-integration of our acceleration data, assuming that at each time step the acceleration stays constant (n in Equation 3 represents the number of datapoints in a cycle; we iterated this algorithm over each cycle and repeated for a full run). We chose to examine the displacement solely in the z-direction because we could be fully confident that displacements in this direction were due to "bumpiness" rather than factors like forward motion or turning of the bus. From Figure 15, we observed that our estimated displacements had an order of magnitude of about 10^{-4} meters, or 0.1 millimeters, which would be extremely small for a passenger to experience. However, the displacements experienced by passengers on the diesel bus still seem to be larger than the displacements experienced by passengers on the hydrogen bus, again indicating that the hydrogen bus traveled more smoothly.

Equation 3

Displacement

$$x_k = x_{k-1} + \frac{1}{2}a_{k-1}\Delta t^2 \qquad k = 1, \dots, n;\ x_0 = 0$$

Figure 16

Speed vs Acceleration RMSD

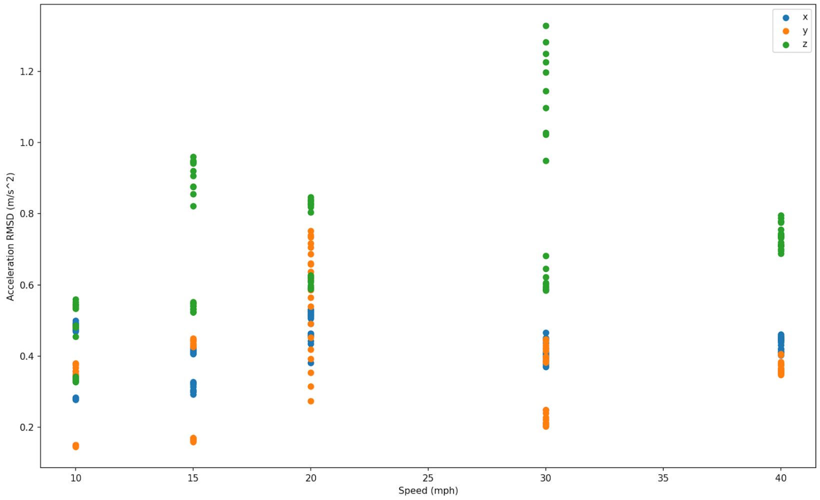

Figure 16. This graph compares the speed of the bus against the RMSD of the acceleration of the bus in the x, y, and z directions. To collect this data, we collaborated with the MTD to record data on the bus while it traveled at a constant speed. For each speed, we collected about 20 cycles of data, resulting in multiple acceleration RMSD data points per speed and per direction.

Figure 17

Temperature vs Acceleration RMSD

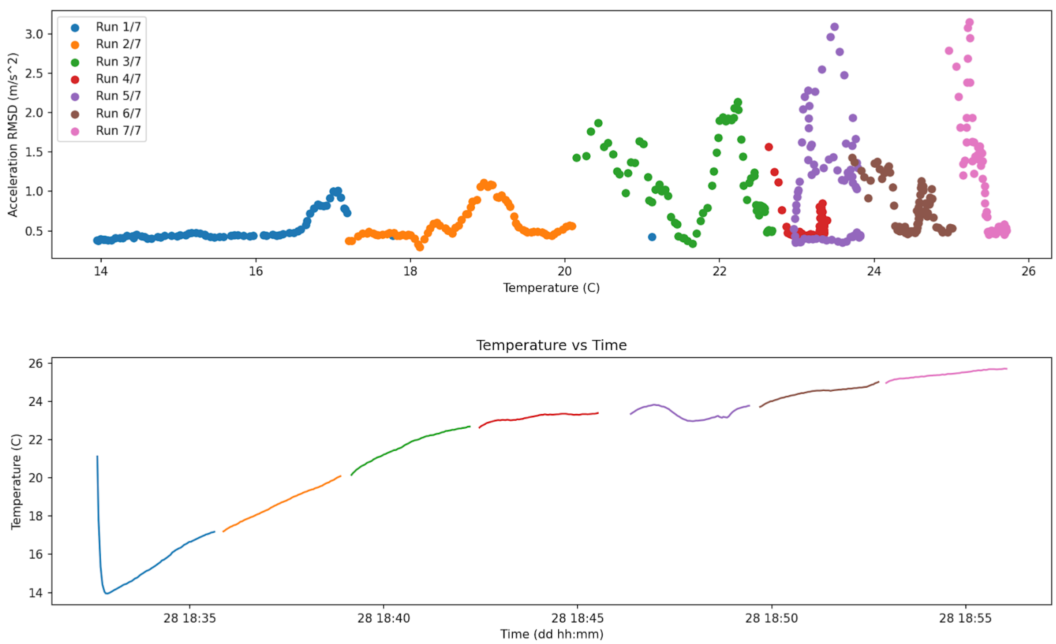

Figure 17. These plots show the temperature in relation to the movement of the bus: the top subplot shows the temperature vs the RMSD of the acceleration for each cycle, while the bottom subplot shows the change in the temperature over time during our bus ride. Each color represents a different run from the data collected on March 27, 2022, and roughly happens to correspond to a different temperature range.

For Figures 16-18, we pivoted from comparing different bus types to trying to relate acceleration with other variables we collected data on, namely speed of the bus, temperature on the bus, and location of the bus. The data for Figure 16 were collected during controlled trials with the MTD; for each of the speeds shown, the driver held the bus at a constant speed as shown by the speedometer for the entire duration of our data collection. We expected to see a positive correlation between the speed and acceleration RMSD. However, the data we collected seems to be so widely spread that it is difficult to derive any significant correlation. One confounding factor for these data was the fact that while the 10, 15, and 20 mph tests were all conducted by driving in circular patterns around a relatively flat parking lot, the 30 mph and 40 mph trials were conducted on roads due to safety concerns. Indeed, if we restrict our view to the 10, 15, and 20 mph tiers, there does seem to be some possible positive correlation, but it is not nearly as significant as we expected.

We hypothesized that the temperature of the bus might be positively correlated to the acceleration due to the vibration of the HVAC system. The results from Figure 17 seem to align with this hypothesis. The top plot in Figure 17 shows a mostly positive trend; as the temperature increases, the peaks in the acceleration RMSD also seem to increase, although the trend varies between these high peaks and a baseline acceleration RMSD of about 0.5. The only runs which seem to deviate from this trend are 4/7 and 6/7 (red and brown, respectively); it is interesting to note that the 24-25℃ region especially has a comparatively very low range of acceleration RMSD. It should be noted, however, that since this is a relatively small dataset, the increase in acceleration RMSD could be due to a number of other factors such as the bus being on a different part of the route. Further (preferably controlled) testing would need to be conducted before any conclusions could be drawn about a correlation between temperature and the vibrations.

Finally, Figure 18 shows the relationship between the acceleration and position of the bus, in an attempt to locate “bumpier” areas of the roads. The specific route shown in Figure 18 is the MTD’s 13 Silver route, with the acceleration RMSD overlaying a map showing the roads. There are a few regions that clearly show greater “bumpiness”, such as north of the South Quad and nearby the Illini Grove. Since the 13 Silver route traces over each road twice, these results are fairly replicable. Interestingly, some of these high-bumpiness areas seem to correspond to bus stops, such as the Gregory at Library stop which is located north of the South Quad. This might indicate that bus drivers tend to decelerate suddenly at certain bus stops.

Figure 18

Acceleration RMSD vs Location

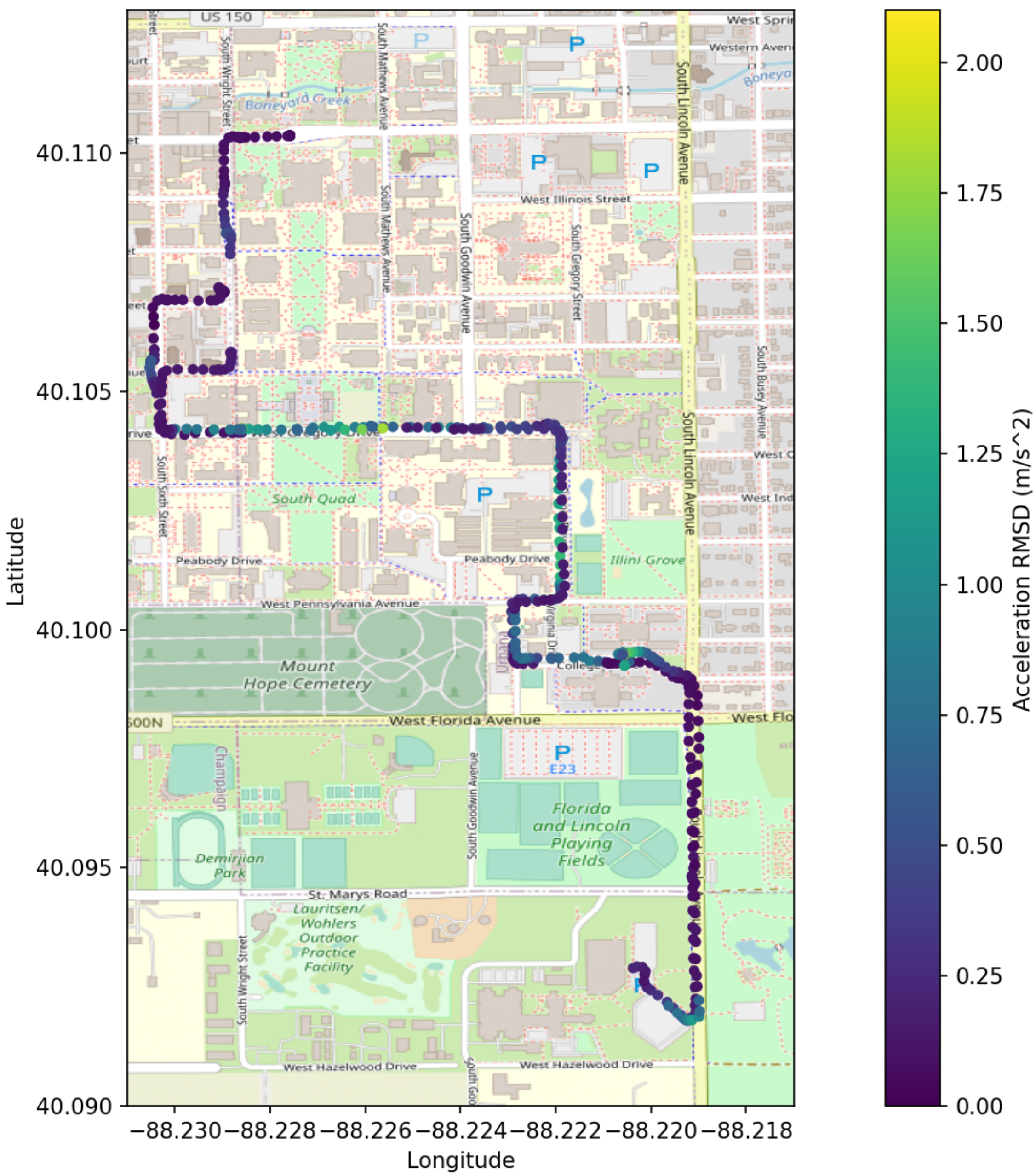

Figure 18. This figure shows the RMSD of acceleration (in the x direction), represented by the color of each dot, plotted against the location of the bus. The plot is overlaid on a map of Champaign-Urbana, allowing us to observe the specific sections of the road which may be bumpier and cause greater changes in acceleration on the bus. Background map supplied by OpenStreetMap[8].

[8] https://www.openstreetmap.org/export#map=15/40.1076/-88.2236

Discussion

Overview of Results

In summary, our research focused on the vibrational analysis of passenger experiences on the MTD's hybrid buses, comparison tests between the hybrid, diesel, and hydrogen buses, and correlations between bus acceleration and other environmental factors. In our hybrid bus vibrational analysis, we found consistent vibrations with a frequency of around 120 Hz with an amplitude of about 1 m/s^2, which aligns with our prior expectation of 100 Hz vibrations but does not reach 0.5*g* in magnitude. We also found some lower-frequency signals at around 40 and 70 Hz, although these were not as significant. In general, vibrations along the x, y, and z axes seemed to be fairly similar in frequency.

When analyzing the vibrations of the different types of buses, we found that although the diesel and hybrid buses showed a wide range of frequencies, the hydrogen buses only had a few dominant frequencies in their vibrations. Furthermore, the dominant frequencies in the hydrogen bus were about 6 Hz apart, resulting in clearly-visible beats in the acceleration graphs. This, along with a comparison of the acceleration RMSD across bus types and an estimate of the displacement experienced by passengers, indicated that the hydrogen bus traveled much more smoothly than the other two types of buses, a fact which was corroborated by our personal observations.

We tried to relate the acceleration RMSD of the bus to specific factors, with mixed results. We were able to find a very slight positive correlation between speed and acceleration, as well as a stronger but more greatly varying positive correlation between temperature and acceleration. There was a much clearer connection between acceleration and position of the bus, which may have been related to bumps in the road or sudden decelerations due to bus stops.

Future Research

As with any project, by the time we conclude this study and reflect on our process, there are certain things we would have liked to do better or would like to look further into. Due to the nature of this project as intended for a semester-long course, we faced time restrictions and as a result, there are some things that we could have done better or that we didn't have the time to implement. The two most important of these are variable isolation and better recording device design. We would also be interested in further applying our findings by looking at the effects these vibrations have on individuals.

Variable Isolation

While we did find some interesting results as to the primary frequencies of the vibrations and the different ride quality between bus types, some more controlled testing would be necessary to draw further conclusions about the precise origins of the vibrations. As we collect data, one of the biggest factors impacting our ability to compare them are the road and weather conditions. Most of our data come from hopping on different buses running routes around campus and therefore the nature of the movement of the bus is not controlled. This occurs both from the buses driving on different roads as well as the nature of the traffic patterns changing even across rides on the same routes. When we did testing with MTD, we were able to control more variables, but not to a very precise level. For example, when doing speed tests in order to produce Figure 15, we did the first few tests while the bus did circles in a parking lot, but the higher speed tests had to be done on the road due to safety concerns. This meant that the road conditions weren't consistent, making it difficult to draw solid conclusions.

If we were to continue our testing, it would be very interesting to spend more time with MTD and try to conduct very controlled runs along the same strip of road to isolate the individual factors we want to test. Some factors we would be most interested in examining further are the speed of the bus, road condition, heating/air conditioning vibrations, the location on the bus, interior temperature, and weather conditions. Isolating these would give us a much more complete picture of the vibrational patterns and could provide better insight for future bus design.

Device Design Improvements

Initially, we separated data collecting into two separate devices with the intention of increasing the amount of time during which the accelerometer could be reading to give us a fuller scope of what was happening on the bus. One of the major trade-offs to implementing this was that it meant that we had to match up the GPS data with the accelerometer in our data analysis code and keep track of twice as many files. By the time we got to data analysis, we realized that the length of the "gap time" during the cycles when the accelerometer was not reading didn't matter as much, because our analysis was much more focused on the individual cycles. This means that in a future design we could include everything in one device which would make recording and analyzing more data much more efficient at the cost of a slightly longer “gap time”.

Another trade-off we made for the sake of time was using the Arduino Mega 2560. This Arduino only has 8 KB of memory onboard, which limits each cycle to about 350 datapoints. With a microcontroller containing more memory, we would be able to record more datapoints in each cycle and also detect a wider range of frequencies.

Effects of Vibrations on the Human Body

Our study was focused on observing the vibrations of the buses and breaking apart the signal. A clear follow-up to what we have found is asking how these vibrations affect the comfort of the passengers on the bus. Anecdotally, we can say that the newer hydrogen buses with adaptive suspension are much more comfortable to ride in than the diesel or even the newer hybrid buses with adaptive suspension. Unfortunately, though, that is as much as we can say given the extent of the study we conducted. From looking at similar studies that have been conducted in the past, we know that the effect of vibrations on the body is very complex. An April 2016 study[9] also examining vibrations in public transport looked at comfort based both on location of the vibration on the body. They also weighed the impact of different frequencies differently for different ranges of frequency. Since we only collected data from the floor of the bus, any analysis we tried to conduct regarding the effects of vibrations on the passengers would be extremely limited. Continuing our study to be able to fully detail passenger impact would be a logical next step and almost certainly would yield interesting results. Additionally, although many of the interesting vibrations we found are relatively high-frequency (around 120 Hz), much of the existing literature only examines vibrations applied to the human body of up to 80 Hz. In the future, it may be worthwhile to focus more research on the impact of higher-frequency vibrations on the human body, since we found these to be very prominent on the MTD buses.

Possible Causes of Vibrations

We can see in our Fourier analysis (Figures 8-13) that the vibrational pattern is often a result of just a few primary frequencies. During our meeting with Rebekka Bolt and Josh Berbaum from the MTD, we discussed where these frequencies are likely originating from. As could be expected, most of the noise is likely coming from the engine and the suspension. This explains why the newer hybrid and hydrogen busses are much smoother than the older diesel busses as they have fewer moving parts and better suspension systems. Other contributions to the vibrations come from the temperature control units and other auxiliary components. More

[9] Zhao, H., Guo, L.-L., & Zeng, X.-Y. (2016, April 14). Evaluation of bus vibration comfort

research is needed regarding the specific vibrational frequencies that may be caused as a result of these components.

References

1. Anderson, M. L. (2013, August). *Subways, Strikes, and Slowdowns: The Impacts of Public Transit on Traffic Congestion*. Transportation For America. Retrieved April 22, 2022, from https://t4america.org/portfolio/la-transit-strike/#

2. Gerard, W. (2022, April 26). *CUMTD awarded 'most innovative' transit agency*. WCIA.com. Retrieved April 28, 2022, from https://www.wcia.com/news/cumtd-awarded-most-innovative-transit-agency/

3. Arduino_lsm9ds1. Arduino_LSM9DS1 - Arduino Reference. (n.d.). Retrieved April 15, 2022, from https://www.arduino.cc/en/Reference/ArduinoLSM9DS1

4. Thinkology. (n.d.). MEMS accelerometers. Silicon Sensing. Retrieved March 12, 2022, from https://www.siliconsensing.com/technology/mems-accelerometers/

5. Datasheet - LSM6DST - INEMO inertial module: Always-on 3D ... (n.d.). Retrieved April 15, 2022, from https://www.st.com/resource/en/datasheet/lsm6dst.pdf

7. *Numpy.fft.fft*¶. numpy.fft.fft - NumPy v1.22 Manual. (n.d.). Retrieved April 28, 2022, from https://numpy.org/doc/stable/reference/generated/numpy.fft.fft.html

8. Export. OpenStreetMap. (n.d.). Retrieved March 10, 2022, from https://www.openstreetmap.org/export#map=15/40.1076/-88.2236

9. Zhao, H., Guo, L.-L., & Zeng, X.-Y. (2016, April 14). Evaluation of bus vibration comfort based on passenger crowdsourcing mode. Mathematical Problems in Engineering. Retrieved April 15, 2022, from https://www.hindawi.com/journals/mpe/2016/2132454/

Make an HTML map (google/leaflet) from a GPS file. GPS Visualizer. (n.d.). Retrieved April 15, 2022, from https://www.gpsvisualizer.com/map_input

Sekulić, D., Dedović, V., Rusov, S., Šalinić, S., & Obradović, A. (2013, April 16). Analysis of vibration effects on the comfort of intercity bus users by oscillatory model with ten degrees of freedom. Applied Mathematical Modelling. Retrieved April 15, 2022, from https://www.sciencedirect.com/science/article/pii/S0307904X13002345

All data and code used for the study available upon request.

Special thanks to Professor George Gollin and Shubhang Goswami for their invaluable assistance and guidance throughout this process.

Also, special thanks to Rebekka Bolt, Josh Berbaum, and the MTD for their cooperation and participation in our study.