Bungee Lab Final Report

MEAM 347 - Group 14
Anthony Saldutti, Luyando Kwenda, Ling Xu

I. Introduction

The purpose of this lab was to maximize the displacement of a jumper using a bungee cord constructed from string and rubber bands given the height of the drop, maximum allowable acceleration interval during the entire fall, and mass of the jumper. We will report the results of the experiment and how we arrived at our bungee configuration from characterizing the rubber bands, collecting acceleration data, and deriving an appropriate model from our calculations.

II. Rubber Band Characterization

For this activity, we conducted MTS tests of varied rubber band configurations. The tests were focused on even-numbered values of rubber bands in series and parallel such that, if needed, we could easily average the data sets together to obtain an estimate of odd-numbered results in the interest of time. The data obtained ranged from 2-12 rubber bands in parallel and from 1-2 in series.

Using this data, we determined cubic models of best fit for each of the data sets and obtained the equations representing these curves. From there, each of the coefficients were obtained for each equation and then normalized based on the number of rubber bands in series and in parallel. This was done based on the formula below in order to obtain a normalized equation which would be dependent on the number of rubber bands in series and in parallel so that displacement calculations could be standardized. The normalized force is ideally the equation of 1 rubber band. Then, we examined this normalized equation relative to the original data sets and compared them to each other to determine the validity of the new equation. Additionally, we derived the equation from the equivalent spring constant of springs in parallel and in series:

$$F_{actual} = c n_p F_{norm}\left(\frac{x}{n_s}\right)$$

In the equation above, c is a constant we chose to create a better fit for the MTS test results, n_p represents the number of bands in parallel and n_s the number in series.

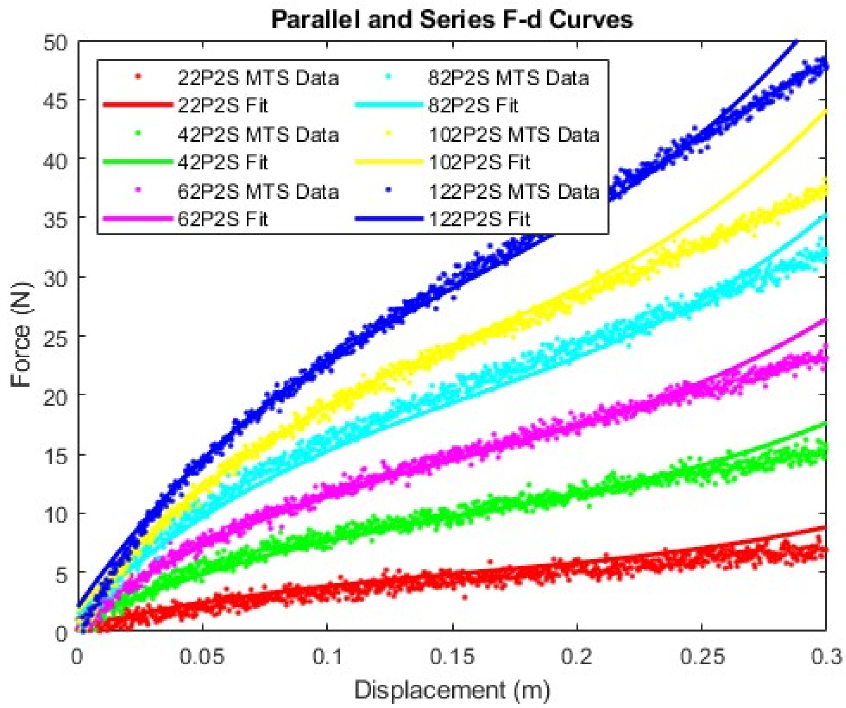

Figure 1: Fitted data using Normalized Equation

III. Using the Accelerometer

In this section, we discuss the processes used to measure acceleration of a falling body and the experimental setup of the accelerometer. Using the Arduino, Adafruit shield, accelerometer, and battery, we conducted preliminary tests to measure acceleration of Deku, our stuffed-animal jumper, when dropping down the staircase. Figure 2a shows the accelerometer wiring and Figure 2b shows the setup for dropping the attached test subject. The accelerometer was wired with I2C and the SD card was wired with SPI. Data is written to the SD card and exported to a text file, and the highest acceleration value is extracted from the measured x, y, and z components to show the maximum magnitude of acceleration at the drop height. The initial length of the rubber band configuration was 18.11in, with 15 rubber bands that were each in series with another rubber band in parallel.

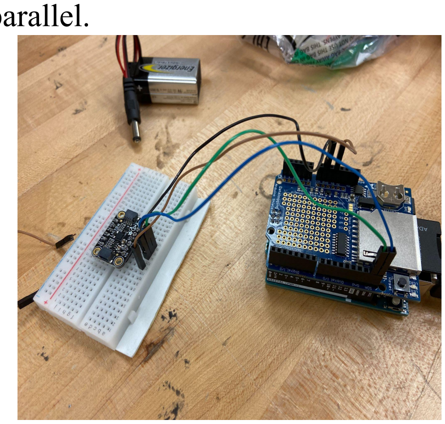

Figure 2: a) Accelerometer wiring

Figure 2: b.) Test drop setup

The plot of the acceleration gathered from the accelerometer data, with respect to time, using the fact that there is a delay of 200ms between each data collection point, is shown in Figure 2. We see that the acceleration stays between the interval of 8.69 m/s^s and -1.06 m/s^2 for this configuration over around 35 seconds of testing time. The position displacement was a little less than 5 feet (48 inches) for the jumper. Thus, the rubber band configuration stretched 29.89 inches. In the next test, we would have to rescale the acceleration values so that the initial value is 0m/s^2. We initially had the accelerometer running at a frequency of 50Hz which produced inaccurate readings but we later changed the code to run at the maximum frequency, 400Hz, decreased the time step interval to 100ms, and took the magnitude of the acceleration vector for more accuracy.

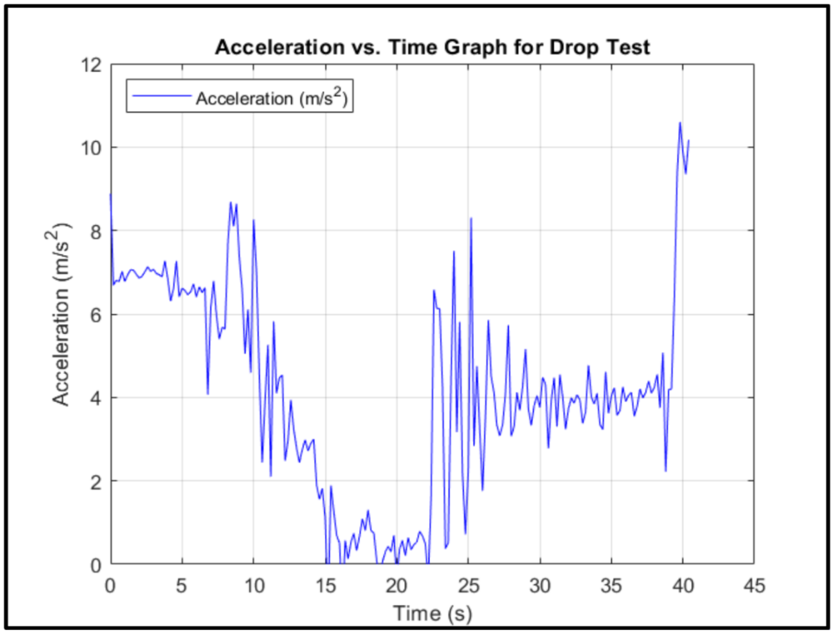

Figure 2: Collected Acceleration Data

IV. Model

We created a MATLAB function that takes in mass, height, and max acceleration as inputs. Taking the coefficients of the normalized non-linear fit of force as a function of extension adjusted for the number of in parallel and in series rubber bands, we used energy balance of the jumper just before it is pulled upwards and the energy at the top of the jump to set the area under the force-extension curve equal to the gravitational potential energy by taking an integral:

$$\int_{0}^{\Delta(x)} F_{actual}(x) \ = \ mgh$$

The upper limit of the integral gave the maximum displacement of the rubber bands.

The force function integrated (from Activity 1) is below:

$$F_{actual} = cn_p F_{norm}(\frac{x}{n_s})$$

where $F_{norm}(X) = 322.7883X^3 - 153.8337X^2 + 38.8496X + 0.2610$

The goal is to minimize the displacement of the rubber band chain while staying within the maximum acceleration limits. Right before the jumper rebounds, we are at maximum acceleration and can find the force needed from the rubber band configuration. The equation is then looped through multiple configurations of parallel and series creating 400 possible combinations of theoretical rubber band bungee cords. During this process, we look at different variable bounds to determine whether a given configuration is valid for the properties of the analyzed jump.

Our group made assumptions that the drag of our jumper and the mass of the rubber band configuration was negligible, since the force from drag and gravitational force are one order of magnitude smaller than that of the payload and force from maximum acceleration at the bottom of the drop.

Our constraints included:

- Checking $h - l_0 - \Delta(x) > 0$, which is the height minus the initial rubber band cord length minus the displacement calculated from estimated forces in our energy balance in order to determine if the rubber band cord is too long post-stretch for the drop, causing the jumper to hit the ground. As we loop through each possible in-series configuration, l_0 is calculated to be the length of each rubber band when tied together (0.055 m), multiplied by the number in series at that point in the loop.
- Another consideration is requiring that $\Delta(x)_F < \Delta(x)_a$ where the displacement estimated from the calculated force in energy balance should be less than the displacement estimated from the maximum acceleration given, where $a_{max} = (F_{max} - mg) / m$ so that the displacement estimated from the maximum acceleration given is when $F_{max} = a_{max}m + mg = F_{actual}(\Delta(x)_a)$

From this point, the ideal rubber band cord in the given case would correspond to the remaining data set with the lowest height minus the initial rubber band cord length minus the displacement calculated from estimated forces. This would represent the rubber band cord which has its maximum extension location closest to the floor. We also included a factor of safety to account for potential energy loss by multiplying mgh by 0.95.

V. Results and Analysis

At the drop conditions of 7.091 meters, weight of 750 grams, and max acceleration of 5G's that were inputted into the model, the resulting configuration generated was 17 rubber bands in series and 5 in parallel. The model calculated that with this configuration, $h - l_0 - \Delta(x) = 1.53$ meters, for $\Delta(x)$ was the optimized displacement calculated in the energy balance. Thus, we accounted for the leftover 1.5 meters with the equivalent length in string. The rubber bands in parallel were connected in series with knots of string.

During the final demo, we attached the accelerometer setup and extra washers as weights on a bottle of sanitizing spray to our bungee configuration. In the drop, we were still about 3 feet short from reaching the ground, and our maximum acceleration only reached 2.91G's (we found the maximum acceleration, 28.56 m/s^2, by applying the "max" function in MATLAB to our raw data stored in the SD card). The analysis of our acceleration data is as follows:

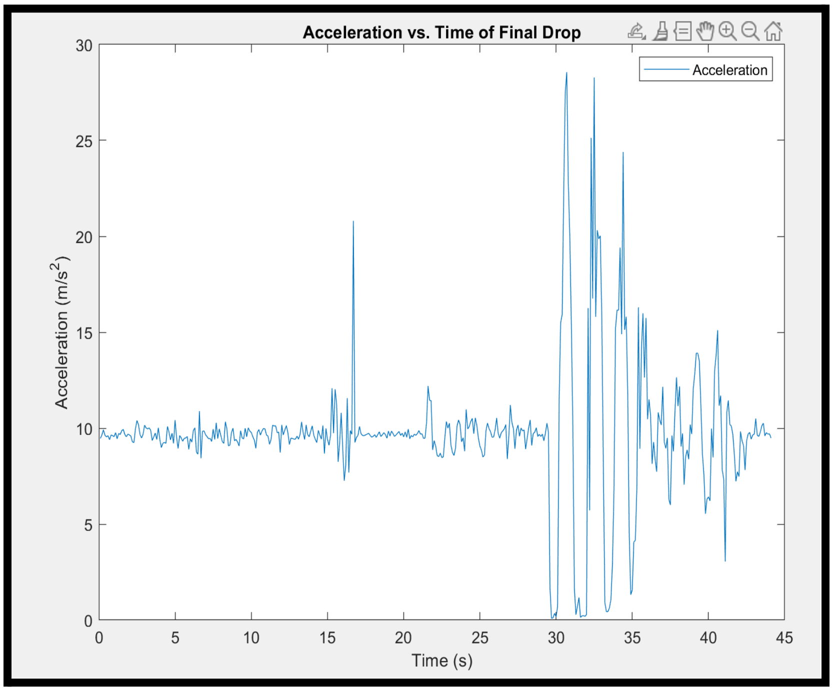

Figure 3: Final drop acceleration data

Errors

The error in maximum acceleration may be a result of our lack of consideration for the lower bound of acceleration, where we only checked to ensure the force from the rubber bands would be less than the force due to the maximum acceleration, but did not take into account a minimum required number of G's. Additionally, we did not adjust our model for characterizing how the bungee would change with respect to additional string, which may have yielded the extra gap between the end height of the bungee and the ground. Furthermore, we realized that we should have corroborated our acceleration data with other acceleration sensors to ensure accuracy, such as an iPhone sensor. In general, the model significantly underestimated the stiffness of the rubber bands configurations, thereby overpredicting the maximum extension of the cord.

Possible Corrections

The outcomes from our pre-demo experimental data had shown our maximum acceleration to be around 2G's; thus, given the time, we should have explored configurations with a higher percentage of string rather than rubber bands in the attempt to increase our maximum acceleration value. Additionally, our model should be corrected to include the initial length with added string and an extra filter to exclude values such that the resulting configuration yields displacements with maximum acceleration lower than the lower bound of G's. Furthermore, to obtain a more accurate drop-model of the rubber bands itself, we could utilize the chain models of rubber elasticity to calculate the stored energy function discussed by Dr. Purohit, with $P = G(\lambda - 1/\lambda^2)$ for P is the derivative of the stored energy function with respect to extension of the rubber band, λ, and G is the shear modulus. In this way, we would be able to incorporate the potential loss at a molecular level and compare it to our experimental data instead of solely constructing a cubic fit.

VI. Appendix

1. Link to video of the drop: https://drive.google.com/file/d/1XxQFU4c4YA7wlE58dcBy1VTFDrqqoMqz/view?usp=sharing
2. Acceleration data:
    a. Raw data:
        i. DATALOG.txt
        ii. DATALOG1.txt (final drop)
3. Code:
    a. Normalizing MTS Data Code:
        i. fitplotdata.m
    b. Model:
        i. plotDataPS.m
        ii. integrals_rubber_bandPS.m
        iii. getData.m
        iv. match_configPS.m (script that you run with inputted parameters)