Documentation

Allysa Mae Tuano – Chrysostomos Karakasis

## Isolating LTIB and LTIBA

LTIB and LTIBA are 2 points along the subjects leg on which the markers were placed to measure position over time. Eventually this data will be taken in order to determine position and orientation to effectively mimic a human’s foot/ankle as they walk at a normal pace. 2 points were recorded in order to determine orientation.

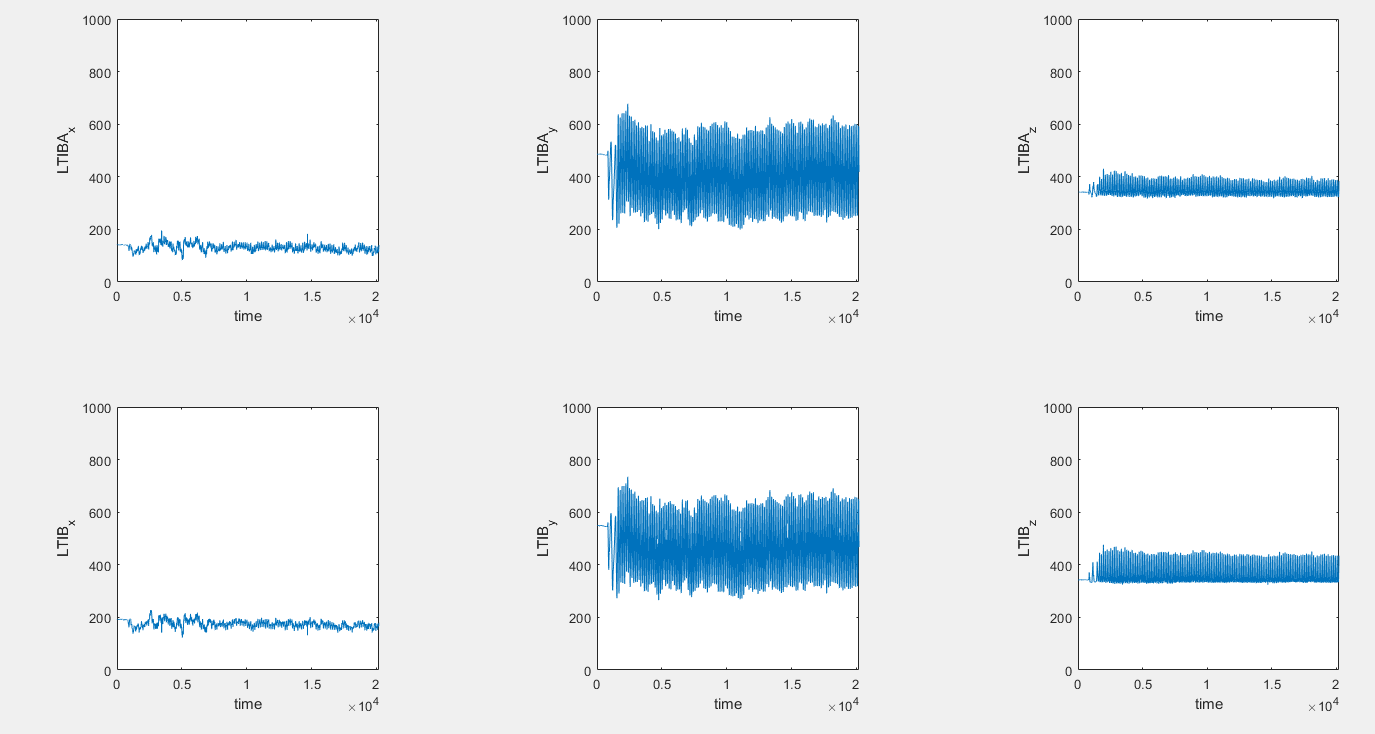


Figure 1:x,y,z vs time plots

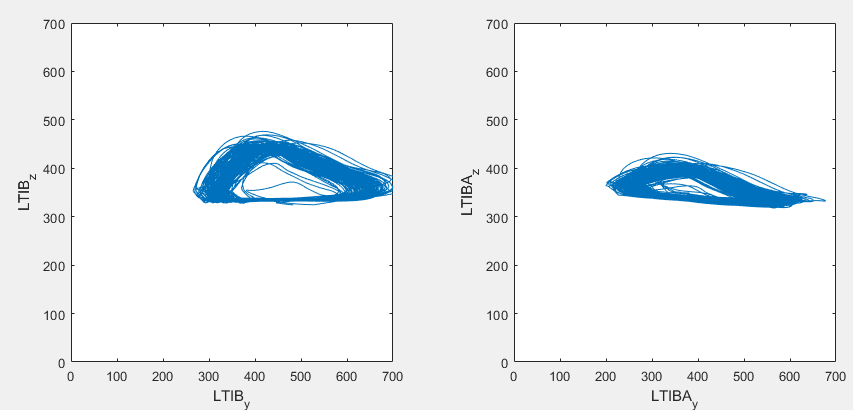


Figure 2: y vs z plots for LTIB and LTIBA

Does it make sense: Do you think the figures make sense (why)? Were there any missing frames in the data?

* From what I saw, the plots make sense. A lot of the motion occurred in the y-z plane, which can be seen in the larger range of motion when compared to the x position vs time plots. some gaps can be seen in the LTIB\_y vs time plot but it does not have an effect on the LTIB\_zy plot.
* shapes of the y-z plots make sense as the curve along the top right of the overall shape corresponds with the leg swinging at the hip, similar to a pendulum. Along with this there is an almost flat shape along the bottom of the plot as the foot stays along the ground/treadmill, but I think it is interesting that the slopes along the bottom differ between LTIB and LTIBA.

Noise: Do you think there is any noise in the data?

* in terms of data collection, since there is minimal amounts of variation/outliers along the points of each individual curve/step path, there looks to be minimal noise.
* in terms of variation between each step, while they do converge to form the same general shape, there is a lot of variation between each step.

## Separating data based on Treadmill Velocity: Methodology and Reasoning

Plot (dLTIBy/dt, LTIBy) and (dLTIBAy/dt, LTIBAy)

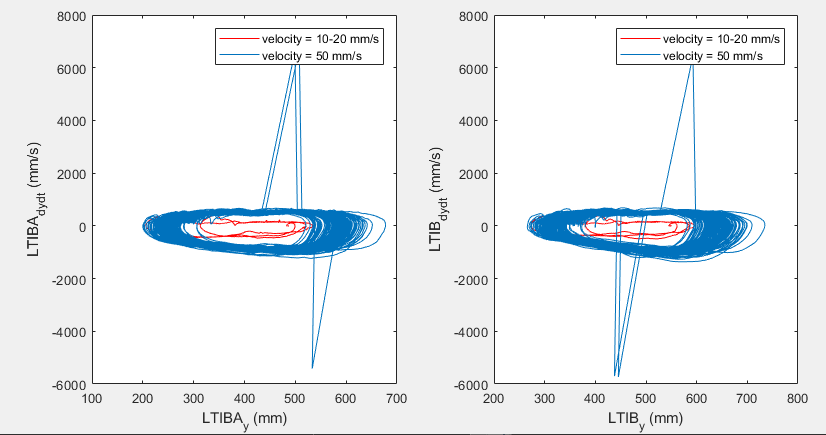
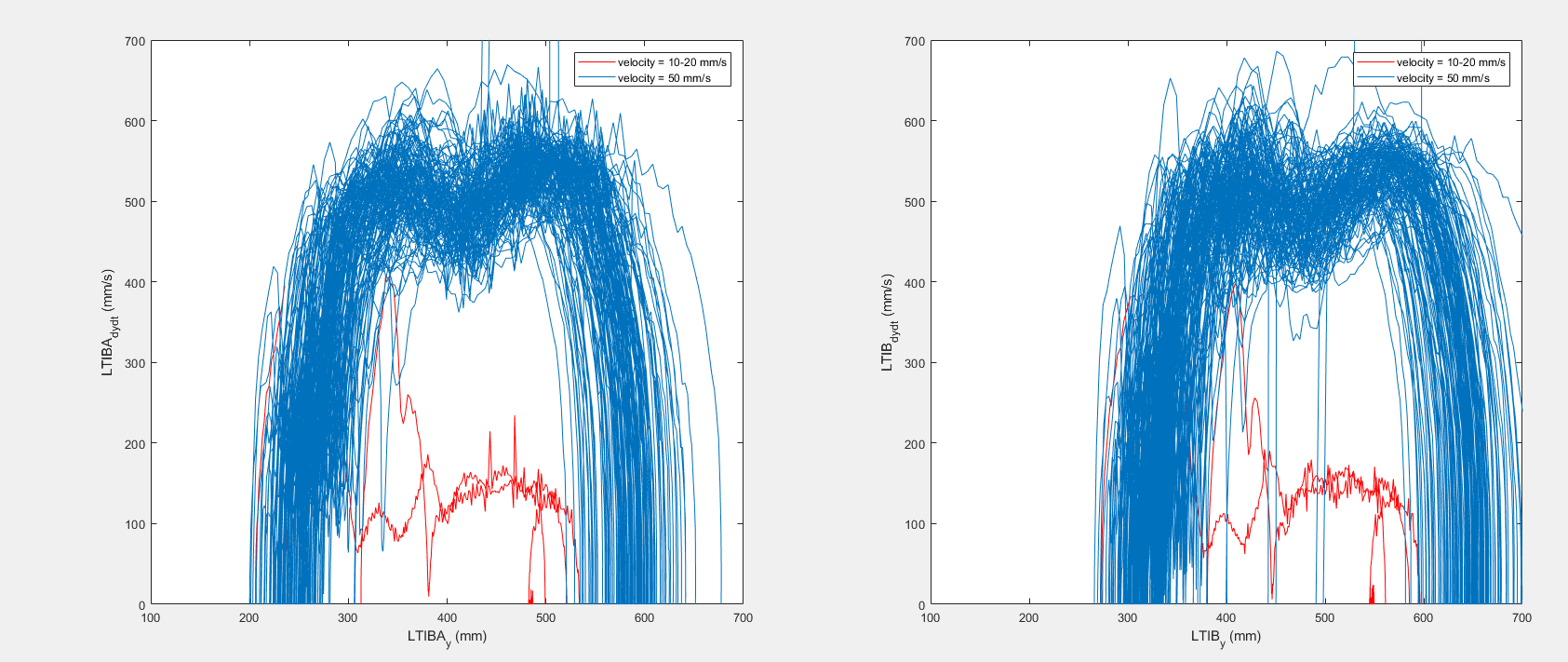


Figure 3: Phase plots of LTIBy and LIBTAy

To separate the data, we can see that on the phase plot there are 2 distinct paths the marker takes (shown in fig 1):

1. The smaller circle representing the smaller steps taken at the slower speed (10-20 cm/s) (blue)
2. The larger circle representing the larger strides taken at the faster speed (50 cm/s) (red)

To determine the criteria/threshold velocity to separate the two velocities, I focused on the (dLTIBy/dt, LTIBy) with no particular reason, though this could be repeated on (dLTIBAy/dt, LTIBAy) with similar results. (dLTIBy/dt, LTIBy) had more data points/lines which would cross through the center of the phase plots and so I wanted to ensure I could properly separate the data and clearly see the distinction between the paths.

A velocity threshold of 400mm/s was chosen as there were a few spikes that eventually would converge to the smaller velocity phase plot, most likely due to either anticipation in the transition of the greater velocity or just miscellaneous movements.

## Averaging the curve

To average the curve, I make an assumption that each individual step/circle takes approximately the same amount of data points. With this assumption, I hope to be able to divide up all of the data into these intervals, line up the points, then average them to get the average curve.

The first thing I do is then determine the interval of a single loop by looking at how often the v50 data crosses over 0, which should represent half the loop. (this was done within LTIB\_LTIBA\_data.m, but I will insert the modified script to get the plot from figure 4, this will prevent figure 3 from being able to plot as it modifies the matricies’ dimensions)

|  |
| --- |
| LTIB\_LTIBA\_data.m |
| data = readtable('Subject\_BH\_2\_27\_2020\_allysa01.csv');  %Isolate the LTIB and LTIBA columns.  %Plot X, Y, Z over time and Y over Z for each one of the markers.    t = str2double(data.Var1(3:end));  LTIBA\_y = str2double(data.Var25(3:end));  LTIB\_y = str2double(data.Var28(3:end));    Position = [LTIBA\_y LTIB\_y];    %multiply by 100 to get mm/s  LTIBA\_dydt = diff(Position(:,1),1,1)\*100;  LTIB\_dydt = diff(Position(:,2),1,1)\*100;  Velocity = [ LTIBA\_dydt LTIB\_dydt];    % Phase Plots  Label\_x2 = ["LTIBA\_y (mm)", "LTIB\_y (mm)"];  Label\_y2 = ["LTIBA\_d\_y\_d\_t (mm/s)", "LTIB\_d\_y\_d\_t (mm/s)"];  v20 = zeros(1551,2);    test = zeros(255,1);  value = 0;  counter = 2;    for i = 1:length(Velocity(:,2))  temp2 = Velocity(i,2);  temp1 = Velocity(i,1);  if temp2 > 400  break  end  v20(i,2) = temp2;  v20(i,1) = temp1;  end    v50 = zeros(length(Velocity)-length(v20),2);  for i = length(v20)+1:1:length(Velocity)+1  v50(i+1-length(v20),1) = Velocity(i-1,1);  v50(i+1-length(v20),2) = Velocity(i-1,2);    % % This section of code was used to determine interval of loops  % % To run this, move the i values +1  if v50(i+1-length(v20)-1,1) < 0 && v50(i+1-length(v20),1) > 0  disp('delta=')  disp(i+1-length(v20)-value)  test(counter) = i+1-length(v20)-value;  value = i+1-length(v20);  counter = counter + 1;  elseif v50(i+1-length(v20)-1,1) > 0 && v50(i+1-length(v20),1) < 0  disp('delta=')  disp(i+1-length(v20)-value)  test(counter) = i+1-length(v20)-value;  value = i+1-length(v20);  counter = counter + 1;  end    end    plot(test)  xlabel('number of points')  ylabel('interval') |

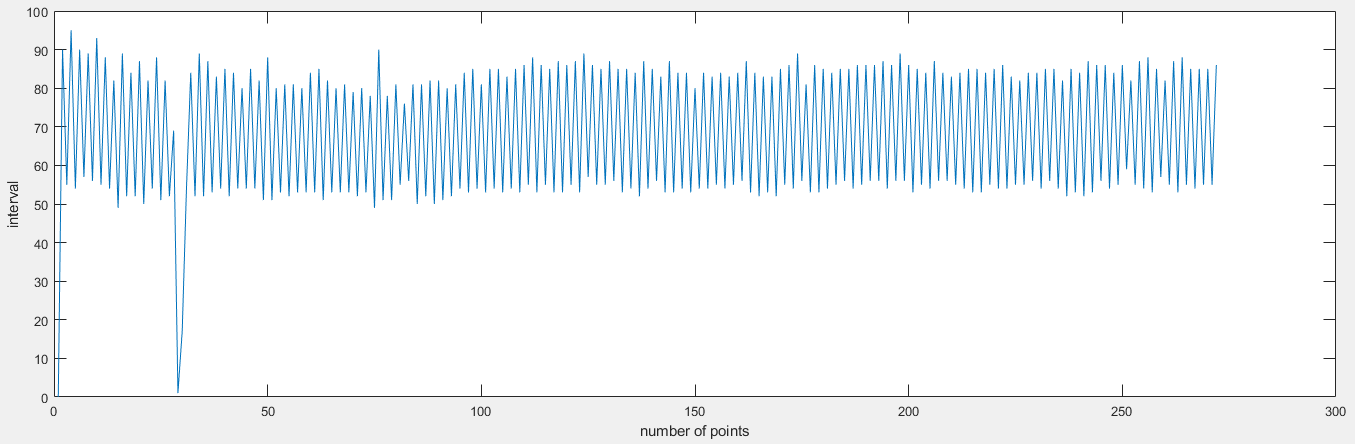


Figure : plot of points which cross over the x axis and the interval in which they occur

Looking at Figure 4, we can see that there are 3 major intervals that occur and we can make assumptions on these intervals

1. 1-2: this interval may be due to the high frequency of data points that were recorded in close proximity, or may be some noise from the data. So this interval may be ignored.
2. 50-60: I assume this represents the top half of the loop, which can be seen in figure 3, as it is smaller to the bottom half and I assume would have less data points.
3. 70-80: I assume this is the top half of the loop for similar reasons to 2.

So looking at this data, we can get a closer idea as to how many data points are in a single loop.

Storing this data in a variable called ‘Test’, I made a new script ‘CalculateInterval.m’ (shown Below) which would group values above 70 and then also take the remaining values and group them if they are greater than 50, to take out the values that occur from noise. I averaged both groups and then added them together to get an overall interval to be 138.6049 data points per loop.

|  |
| --- |
| CalculateInterval.m |
| countTop = 0;  countBot = 0;  top = zeros(134,1);  bot = zeros(130,1);  for i = 1:1:length(test)  if test(i)>70  countTop = countTop + 1;  top(countTop) = test(i);  elseif test(i)>50  countBot = countBot + 1;  bot(countBot) = test(i);  else  end  end  topAvg = mean(top);  botAvg = mean(bot);    disp(topAvg+botAvg) |

# Creating Average Curves (YZ-Plane)

Chrystomos developed an algorithm that would determine heel strikes from the z coordinate data.

With this data, the continuous gait cycle data could be divided into individual gait cycles.

The individual cycles were resampled and interpolated to then be averaged.