| | |
|---|---|
| MOVEMENT SCIENCE LAB<br>TEXAS SCOTTISH RITE HOSPITAL FOR CHILDREN | **Texas Scottish Rite Hospital**<br>**Research Department**<br>**Division of Movement Science** |

**Edgren Side Step Test Automated Events and Timings**

| | | | |
|---|---|---|---|
| **Effective Date:** | | **Document Number:** | 0.1 |

**Approval/Date by Movement Science Director/Assistant Director:**

Kirsten Tulchin-Francis, PhD
Director, Division of Movement Science

Kelly Jeans, MS
Assistant Director, Movement Science Lab Dallas

Sophia Ulman, PhD
Assistant Director, Movement Science Lab Frisco

| REVISION HISTORY | | |
|---|---|---|
| **Date** | **Section** | **Author** |
| 2/3/2021 | Original | Benjamin Randoing |

## PURPOSE

To further automate an existing algorithm to identify foot contact/off events and to calculate descriptive metrics characterizing subject movement.

## SCOPE

Considerable amount of work has been done to create a reliable algorithm to identify the split/lag times during the ESS test. Split time is defined as the travel time between end lines. Lag time is defined as the transition time at each end line. The algorithm currently requires users to pick points from the graph of the raw ACC data. This requirement is still a necessary step in the new code and can be modified into a more user-friendly experience. Additionally, the data points can be saved to the MATLAB workspace to automatically calculate the split and lab times along with saving the raw accelerometry data for each split/lag. The accelerometry data may be

implemented to calculate the peak acceleration (g), integral of accelerometry data, and stride frequency of a subject.

## DEFINITIONS/ABBREVIATIONS

1. **Definitions**
    - 1.1 Vector Magnitude of horizontal and perpendicular directions
    - 1.2 Start – 1st prominent peak
    - 1.3 FC – 1st data point at the bottom of the bout (after the prominent peak)
    - 1.4 FO – 1st prominent peak after the plateau following the FC event
    - 1.5 Script - Separate MATLAB program called in the primary program to execute a separate function
    - 1.6 Zero Offset - Mathematical adjustment to ensure horizontal and perpendicular accelerometer planes are represented at zero
    - 1.7 Split Time - Time from FO to FC for a single directional repetition
    - 1.8 Lag Time - Time between FC and FO when changing directions between repetitions
    - 1.9 Stride Frequency - Largest magnitude, non-zero frequency component of lateral movement
2. **Abbreviations**
    - 2.1 ACC – accelerometer
    - 2.2 FC – foot contact
    - 2.3 FO – foot off
    - 2.4 ESST - Edgren test
    - 2.5 .csv - CSV file format compatible with Microsoft excel
    - 2.6 .txt - Text file used to store individual trial start and end times
    - 2.7 .mat - MATLAB output file for temporary storage of accelerometry data

## RESPONSIBILITIES/END USERS

This data processing tool will be used by staff in MSL Dallas and Frisco.

## CODE OVERVIEW

1. The code initializes by prompting the user to answer whether the trial time values will be entered manually or read automatically from a .txt file. The subject number is also requested.
2. The user must select the left and right .csv accelerometer files. The code will read all ESST .txt files in the same folder as the selected .csv files.
3. The start and end times for each trial for the .txt files are identified and converted to frames in the .csv file. The corresponding frames are used to create new .mat files for each trail and foot.
4. The user may select to plot a single trial data or all trial data in succession within the folder of the csv files. If the individual option is selected, the user must select the corresponding .mat file. Otherwise, the code will automatically read in all .mat files in the corresponding folder.
5. The code will be routed to a separate script entitle ActigraphCommands.m to perform the zero offset and vector magnitude of the trial data. The left and right foot data is graphed to illustrate the horizontal axis data with the zero offset and the vector magnitude for each

foot. The user must select the appropriate points to indicate the FO and FC of feet during the trial.

6. The code will calculate the split, lag, and total time for the trial. The average right and left split and lag times are calculated.
7. An excel file is created in the same folder as the original .csv accelerometer data to include - for each foot - the split times, lag times, average split times, average lag times, total time, and selected time values when selecting the time on the corresponding plot.
8. Each repetition with the trial moving to the right and left is separated into six total repetition.
9. G Metric calculations are performed to obtain to trapezoidal integral for each repetition within a trial. The integrals are normalized by the split time for the specific repetition. The maximum acceleration (g) and stride frequency are obtained for each repetition as well.
10. The aforementioned G Metrics are saved into an excel file in the same folder as the original .csv accelerometer data.
11. All of the split, lag, and total times as well as the G Metrics are saved into a global excel file in a separate folder where the subject data of multiple subjects is saved.

**PROCEDURE**

1. Select whether trial times with be entered manually or read automatically by typing the appropriate response in the command window.
2. If manual entry is selected enter the start time and then the end time of the trial. Then, enter an appropriate name for the new .mat file of accelerometer data for the specific trial. If another entry is desired, select yes when prompted. Otherwise, proceed to the next step. An example automatic and manual input series is noted in Appendix B and C respectively.
3. Enter the appropriate subject number in the command window.
4. Select the Left then Right .csv files with the accelerometer data.
5. Select whether all trials or only a single trial is of interest for plotting and calculations.
6. If an individual trial is desired select the left and right .mat files corresponding to the appropriate trial. If all trials are desired, the trials will be read automatically.
7. A cross-hare will appear on the graphed image. Select the point of foot lifting and stepping on with the plotted vector magnitude data. This may be done by right clicking to select a point. Left clicking on the plot will zoom in or pan the graph window.

**REFERENCES**


1. Carlos Adrian Vargas Aguilera (2021). ginput2.m v3.1 (Nov 2009) (https://www.mathworks.com/matlabcentral/fileexchange/20645-ginput2-m-v3-1-nov-2009), MATLAB Central File Exchange. Retrieved February 6, 2021.
2. Stevens Jr, Wilshaw R., Anthony M. Anderson, and Kirsten Tulchin-Francis. "Validation of Accelerometry Data to Identify Movement Patterns During Agility Testing." *Frontiers in sports and active living* 2 (2020).

## APPENDICES

APPENDIX A - Example Plot of Vector Magnitude and Horizontal Data with Zero Offset

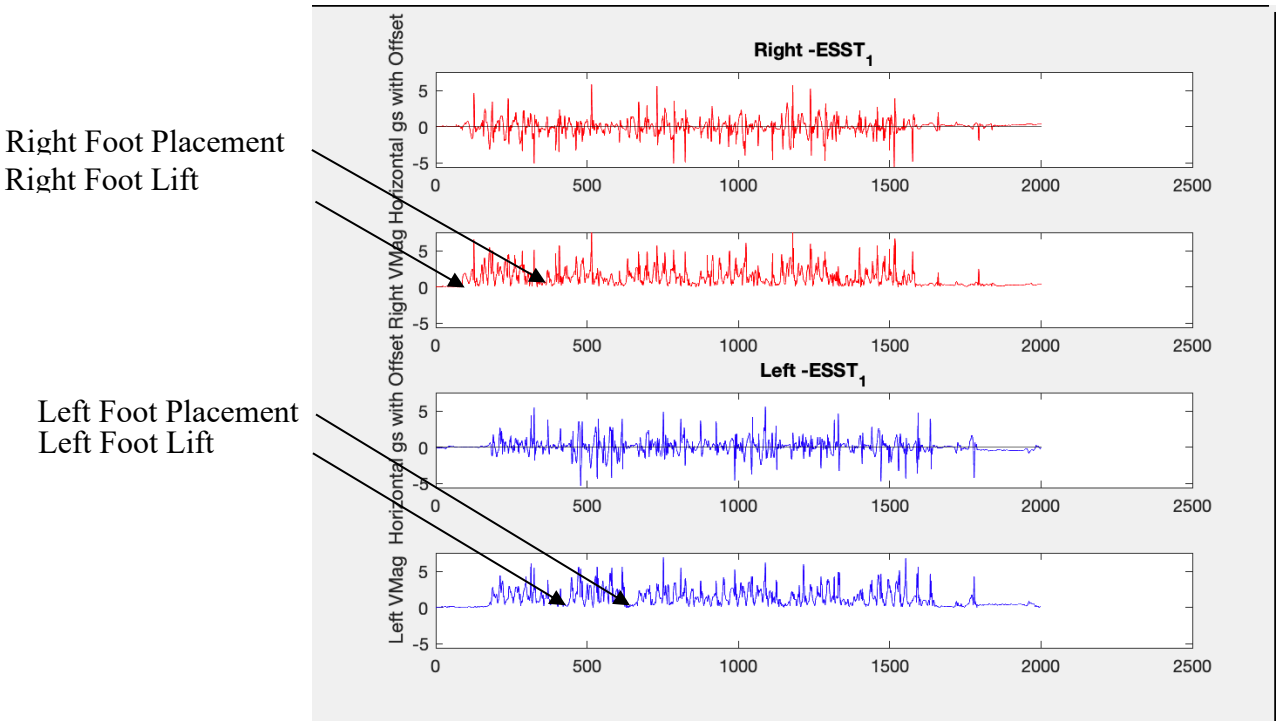


Figure 1. Right and Left Foot Vector Magnitude and Horizontal Data with Zero Offset

APPENDIX B - Example Prompt Series for Automatic Operation

```
Manual Input? [Y/N] N
Subject Study Number? 123
Please Select Left CSV File
Please Select Right CSV File
Plot All or Individual ESST? [ALL, IND]? IND
Please Select Left MAT File
Please Select Right CSV File
    'ESST_1L.mat'    'ESST_1R.mat'
```

| ERROR TRACKING | | |
|---|---|---|
| **Error Message** | **Approx. Line # (.m File)** | **Solution Steps** |
| Index exceeds the number of array elements (0).<br><br>Error in Attempt_ACCDataExtractAutomatic_ESST (line 709)<br>disp(['xValue ', num2str(xArray(j))]); | 709 | This indicates the window for selecting points was closed prior to selecting all points. A solution is to re-run the test to completion. |
| Index exceeds array bounds.<br><br>Error in Attempt_ACCDataManual (line 88)<br>minDiff = (double(tempStart(2))-double(csvStart(2)));<br><br>Error in Attempt_ACCDataExtractAutomatic_ESST (line 14)<br>Attempt_ACCDataManual | 88 | This is an error if entering trial times manually. If the times entered are chronologically before the true start time, the code will have a negative time value and run this error. The solution is to ensure the times are in 24 hr equivalents (ie: 14:00) and correct. Then, re-run the program. |