Test Report: Time_Freq_Analysis

Elizabeth Hofer

December 30, 2020

1 Revision History

Date	Version	Notes
27.12.20	1.0	Initial Release

2 Symbols, Abbreviations and Acronyms

symbol	**description**
T	Test
STFT	Short Time Fourier Transform
FFT	Fast Fourier Transform

Contents

List of Tables

List of Figures

This document review the Verification and Validation specifications as outlined in the VnVPlan (https://github.com/liziscool/cas741_project/blob/master/docs/VnVPlan/VnVPlan.pdf) for Time_Freq_Analysis.

3 Functional Requirements Evaluation

R1 *Program shall take a sequence of numbers representing the signal to be analysed as input. All other inputs will have defaults, but program shall accept user inputs for those as well.*
Tentatively Met. Due to anomalies/misunderstandings in implementation the frequency range cannot be specified. This feature will either be not implemented (as it may not be necessary), or will be implemented in future updates

R2 *Program shall notify user if an input value is illegal or out of bound*
Met.

R3 *The output shall be a time frequency representation of the signal in the specified time period and over the specified frequency range.*
Met

R4 *The time-frequency representations of simple input signals (such as sinusoids of a constant frequency or an impulse) should be comparable to existing time-frequency transforms of that signal.*
Implimentation of Testing, in progress.

4 Non-functional Requirements Evaluation

R5 *Program shall plot time-frequency representation as a heat map.*
Met.

R6 *The time complexity for this program should be $O(n)$.*
Honestly I have no idea and I don't even know how I would have gone about testing this. I had previous implementations that took forever and I have since fixed that. It runs 'quick enough' for my purposes.

R7 *Program will not have a graphical user interface but should still be easy to use, the input parameters besides the signal shall all have default*

values, there should be at most 6 optional inputs.
Met.

R8 *The program code should be clear and readable.*
Partially met. Will be improved in future versions.

R9 *The program should easily integrate with other software programs.*
Met.

R10 *The program should minimize spectral leakage.*
Testing of this requirement was not completed. It will be completed in future versions.

5 Comparison to Existing Implementation

This section will not be appropriate for every project.

6 Unit Testing

7 Changes Due to Testing

8 Automated Testing

9 Trace to Requirements

10 Trace to Modules

11 Code Coverage Metrics