CAS 741: Problem Statement

Dynamical Systems: Multi-Pendulum

Karol Serkis
serkiskj@mcmaster.ca

December 14, 2018

1 Revision History

Table 1: Revision History

Date	**Developer(s)**	**Change**
December 14, 2018	Karol Serkis	First revision of document

2 Symbols, Abbreviations and Acronyms

See SRS Documentation at CAS-741-Pendula SRS [give url —SS]
[Also add any additional symbols, abbreviations or acronyms —SS]
The symbols are listed in alphabetical order.

symbol	description
A	Assumption
DD	Data Definition
FT	Functional Test
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
MG	Module Guide
MIS	Module Interface Specification
NF	Non-Functional Requirement
R	Requirement
SRS	Software Requirements Specification
T	Test
VnV	Verification and Validation

2.1 Table of Units

Throughout this document SI (Système International d'Unités) is employedas the unit system. In addition to the basic units, several derived units are used as described below. For each unit, the symbol is given followed by a description of the unit and the SI name.

symbol	unit	SI
m	length	metre
kg	mass	kilogram
s	time	second
°	angle	degree

Table 2: Table of Units

Contents

List of Tables

List of Figures

3 Introduction

The purpose of the document is to provide the User Guide for instructions in testing the Multi-Pendulum Simulation software with respect to the requirements (see SRS document). SRS template is based on (Smith and Lai, 2005) & (Smith et al., 2007) (ex. based on the principle of information hiding (Parnas, 1972)). User Guide consists of outlining software execution and input instructions for the software's various modules. These tests are created to ensure that the units satisfy the software's functional and nonfunctional requirements. The tests can be traced to a particular module. The module should be traced to a particular requirement.

4 Before executing Multi-Pendulum Simulation

The user responsibilities are described in the SRS document, nevertheless it is worth mentioning that a minimum of two files are required to execute Multi-Pendulum Simulation which are:

- a plot trajectory of the pendula initialized
- a plot of movement over Kinetic Energy and Potential Energy

The Multi-Pendulum Simulation program solution that only focuses on multi-pendulum simulations (double & triple pendula and beyond) and tracking the chaotic motion of the system.

5 Starting Multi-Pendulum Simulation

After downloading the source files and checking that the requirement from the dependencies are met, the main.py file can be executed and two windows should appear. Do not close these windows since there is no way of reopening them without starting the software again. The first window corresponds to Multi-Pendulum Simulation flow with all the different buttons that represent the different steps of execution which are (details of each step will be describe later in the document):

- Pass the user input from the command-line options
- Simulate the plot trajectory of the pendula.

6 Load files

To pass the user input from the command-line options in Multi-Pendulum Simulation one must do this:

An error message should appear if the loading process was not performed properly. At the end of the loading process a new window with the plot trajectory simulation and KE and PE plot should appear.

7 Simulate the plot trajectory

Once the command-line options loaded the simulation process. Please be patient, this step can take some time. At the end of the simulation, three windows will appear:

8 Simulate the plot KE and PE

Once the command-line options loaded the simulation process. Please be patient, this step can take some time. At the end of the simulation, three windows will appear:

References

David L. Parnas. On the criteria to be used in decomposing systems into modules. *Comm. ACM*, 15(2):1053–1058, December 1972.

W. Spencer Smith and Lei Lai. A new requirements template for scientific computing. In J. Ralyté, P. Àgerfalk, and N. Kraiem, editors, *Proceedings of the First International Workshop on Situational Requirements Engineering Processes – Methods, Techniques and Tools to Support Situation-Specific Requirements Engineering Processes, SREP'05*, pages 107–121, Paris, France, 2005. In conjunction with 13th IEEE International Requirements Engineering Conference.

W. Spencer Smith, Lei Lai, and Ridha Khedri. Requirements analysis for engineering computation: A systematic approach for improving software reliability. *Reliable Computing, Special Issue on Reliable Engineering Computation*, 13(1):83–107, February 2007.