Software Requirements Specification for Double Pendulum

Dong Chen

August 9, 2023

Contents

1 Reference Material

This section records information for easy reference.

1.1 Table of Units

The unit system used throughout is SI (Système International d'Unités). In addition to the basic units, several derived units are also used. For each unit, the Table of Units lists the symbol, a description, and the SI name.

Table 1: Table of Units

Symbol	Description	SI Name
kg	mass	kilogram
m	length	metre
N	force	newton
rad	angle	radian
s	time	second

1.2 Table of Symbols

The symbols used in this document are summarized in the Table of Symbols along with their units. Throughout the document, symbols in bold will represent vectors, and scalars otherwise. The symbols are listed in alphabetical order. For vector quantities, the units shown are for each component of the vector.

Table 2: Table of Symbols

Symbol	Description	Units
$a_{\text{x}1}$	Horizontal acceleration of the first object	$\frac{\text{m}}{\text{s}^2}$
$a_{\text{x}2}$	Horizontal acceleration of the second object	$\frac{\text{m}}{\text{s}^2}$
$a_{\text{y}1}$	Vertical acceleration of the first object	$\frac{\text{m}}{\text{s}^2}$
$a_{\text{y}2}$	Vertical acceleration of the second object	$\frac{\text{m}}{\text{s}^2}$
$\mathbf{a}(t)$	Acceleration	$\frac{\text{m}}{\text{s}^2}$
$\mathbf{F}$	Force	N
g	Magnitude of gravitational acceleration	$\frac{\text{m}}{\text{s}^2}$
$\mathbf{g}$	Gravitational acceleration	$\frac{\text{m}}{\text{s}^2}$
$\hat{\mathbf{i}}$	Unit vector	–

Continued on next page

Table 2: Table of Symbols (Continued)

Symbol	Description	Units
L_1	Length of the first rod	m
L_2	Length of the second rod	m
m	Mass	kg
m_1	Mass of the first object	kg
m_2	Mass of the second object	kg
p_{x1}	Horizontal position of the first object	m
p_{x2}	Horizontal position of the second object	m
p_{y1}	Vertical position of the first object	m
p_{y2}	Vertical position of the second object	m
$\mathbf{p}(t)$	Position	m
$\mathbf{T}$	Tension	N
$\mathbf{T}_1$	Tension of the first object	N
$\mathbf{T}_2$	Tension of the second object	N
t	Time	s
theta	Dependent variables	rad
v_{x1}	Horizontal velocity of the first object	$\frac{\text{m}}{\text{s}}$
v_{x2}	Horizontal velocity of the second object	$\frac{\text{m}}{\text{s}}$
v_{y1}	Vertical velocity of the first object	$\frac{\text{m}}{\text{s}}$
v_{y2}	Vertical velocity of the second object	$\frac{\text{m}}{\text{s}}$
$\mathbf{v}(t)$	Velocity	$\frac{\text{m}}{\text{s}}$
w_1	Angular velocity of the first object	$\frac{\text{rad}}{\text{s}}$
w_2	Angular velocity of the second object	$\frac{\text{rad}}{\text{s}}$
α_1	Angular acceleration of the first object	$\frac{\text{rad}}{\text{s}^2}$
α_2	Angular acceleration of the second object	$\frac{\text{rad}}{\text{s}^2}$
θ_1	Angle of the first rod	rad
θ_2	Angle of the second rod	rad
π	Ratio of circumference to diameter for any circle	–

1.3 Abbreviations and Acronyms

Table 3: Abbreviations and Acronyms

Abbreviation	Full Form
2D	Two-Dimensional
A	Assumption
DD	Data Definition
DblPend	Double Pendulum
GD	General Definition
GS	Goal Statement
IM	Instance Model
PS	Physical System Description
R	Requirement
RefBy	Referenced by
Refname	Reference Name
SRS	Software Requirements Specification
TM	Theoretical Model
Uncert.	Typical Uncertainty

2 Introduction

A pendulum consists of mass attached to the end of a rod and its moving curve is highly sensitive to initial conditions. Therefore, it is useful to have a program to simulate the motion of the pendulum to exhibit its chaotic characteristics. The program documented here is called Double Pendulum.

The following section provides an overview of the Software Requirements Specification (SRS) for Double Pendulum. This section explains the purpose of this document, the scope of the requirements, the characteristics of the intended reader, and the organization of the document.

2.1 Purpose of Document

The primary purpose of this document is to record the requirements of DblPend. Goals, assumptions, theoretical models, definitions, and other model derivation information are specified, allowing the reader to fully understand and verify the purpose and scientific basis of DblPend. With the exception of system constraints, this SRS will remain abstract, describing what problem is being solved, but not how to solve it.

This document will be used as a starting point for subsequent development phases, including writing the design specification and the software verification and validation plan. The design document will show how the requirements are to be realized, including decisions on the numerical algorithms and programming environment. The verification and validation plan will show the steps that will be used to increase confidence in the software documentation and the implementation. Although the SRS fits in a series of documents that follow the so-called waterfall model, the actual development process is not constrained in any way. Even when the waterfall model is not followed, as Parnas and Clements point out [6], the most logical way to present the documentation is still to "fake" a rational design process.

2.2 Scope of Requirements

The scope of the requirements includes the analysis of a two-dimensional (2D) pendulum motion problem with various initial conditions.

2.3 Characteristics of Intended Reader

Reviewers of this documentation should have an understanding of undergraduate level 2 physics, undergraduate level 1 calculus, and ordinary differential equations. The users of DblPend can have a lower level of expertise, as explained in Sec:User Characteristics.

2.4 Organization of Document

The organization of this document follows the template for an SRS for scientific computing software proposed by [5], [8], [9], and [7]. The presentation follows the standard pattern of presenting goals, theories, definitions, and assumptions. For readers that would like a more bottom up approach, they can start reading the instance models and trace back to find any additional information they require.

The goal statements are refined to the theoretical models and the theoretical models to the instance models.

3 General System Description

This section provides general information about the system. It identifies the interfaces between the system and its environment, describes the user characteristics, and lists the system constraints.

3.1 System Context

Fig:sysCtxDiag shows the system context. A circle represents an entity external to the software, the user in this case. A rectangle represents the software system itself (DblPend). Arrows are used to show the data flow between the system and its environment.

The interaction between the product and the user is through an application programming interface. The responsibilities of the user and the system are as follows:

- User Responsibilities

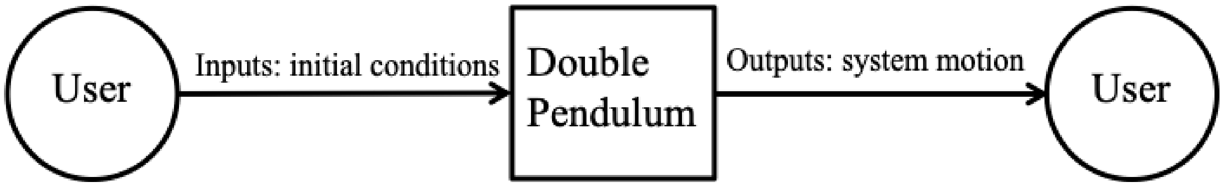

Figure 1: System Context

  - Provide initial conditions of the physical state of the motion and the input data related to the Double Pendulum, ensuring no errors in the data entry.
  - Ensure that consistent units are used for input variables.
  - Ensure required software assumptions are appropriate for any particular problem input to the software.

- DblPend Responsibilities
  - Detect data type mismatch, such as a string of characters input instead of a floating point number.
  - Determine if the inputs satisfy the required physical and software constraints.
  - Calculate the required outputs.
  - Generate the required graphs.

3.2 User Characteristics

The end user of DblPend should have an understanding of high school physics, high school calculus and ordinary differential equations.

3.3 System Constraints

There are no system constraints.

4 Specific System Description

This section first presents the problem description, which gives a high-level view of the problem to be solved. This is followed by the solution characteristics specification, which presents the assumptions, theories, and definitions that are used.

4.1 Problem Description

A system is needed to predict the motion of a pendulum.

4.1.1 Terminology and Definitions

This subsection provides a list of terms that are used in the subsequent sections and their meaning, with the purpose of reducing ambiguity and making it easier to correctly understand the requirements.

- Gravity: The force that attracts one physical body with mass to another.
- Cartesian coordinate system: A coordinate system that specifies each point uniquely in a plane by a set of numerical coordinates, which are the signed distances to the point from two fixed perpendicular oriented lines, measured in the same unit of length (from [2]).

4.1.2 Physical System Description

The physical system of DblPend, as shown in Fig:dblpend, includes the following elements:

PS1: The first rod (with length of the first rod L_1).

PS2: The second rod (with length of the second rod L_2).

PS3: The first object.

PS4: The second object.

4.1.3 Goal Statements

Given the masses, length of the rods, initial angle of the masses and the gravitational constant, the goal statement is:

motionMass: Calculate the motion of the masses.

4.2 Solution Characteristics Specification

The instance models that govern DblPend are presented in the Instance Model Section. The information to understand the meaning of the instance models and their derivation is also presented, so that the instance models can be verified.

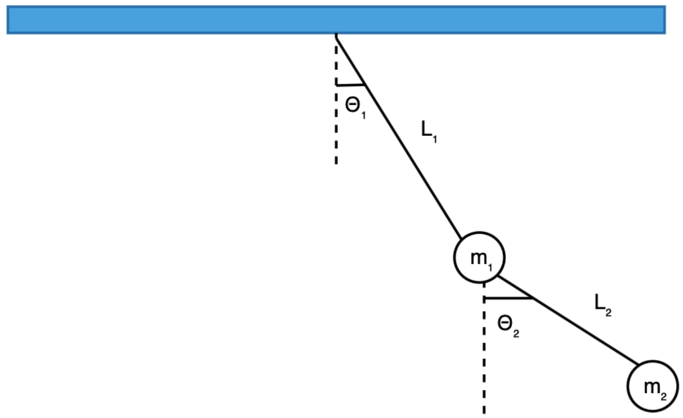

Figure 2: The physical system

4.2.1 Assumptions

This section simplifies the original problem and helps in developing the theoretical models by filling in the missing information for the physical system. The assumptions refine the scope by providing more detail.

twoDMotion: The pendulum motion is two-dimensional (2D).

cartSys: A Cartesian coordinate system is used.

cartSysR: The Cartesian coordinate system is right-handed where positive x-axis and y-axis point right up.

yAxisDir: The direction of the y-axis is directed opposite to gravity.

startOrigin: The first rod is attached to the origin.

firstPend: The first rod has two sides. One side attaches to the origin. Another side attaches to the first object.

secondPend: The second rod has two sides. One side attaches to the first object. Another side attaches to the second object.

4.2.2 Theoretical Models

This section focuses on the general equations and laws that DblPend is based on.

Refname	TM:acceleration
Label	Acceleration
Equation	$$\mathbf{a}(t) = \frac{d\mathbf{v}(t)}{dt}$$
Description	$\mathbf{a}(t)$ is the acceleration ($\frac{\text{m}}{\text{s}^2}$) t is the time (s) $\mathbf{v}(t)$ is the velocity ($\frac{\text{m}}{\text{s}}$)
Source	[1]
RefBy	

Refname	TM:velocity
Label	Velocity
Equation	$$\mathbf{v}(t) = \frac{d\mathbf{p}(t)}{dt}$$
Description	$\mathbf{v}(t)$ is the velocity ($\frac{\text{m}}{\text{s}}$) t is the time (s) $\mathbf{p}(t)$ is the position (m)
Source	[3]
RefBy	

Refname	TM:NewtonSecLawMot
Label	Newton's second law of motion
Equation	$$\mathbf{F} = m\mathbf{a}(t)$$
Description	$\mathbf{F}$ is the force (N) m is the mass (kg) $\mathbf{a}(t)$ is the acceleration ($\frac{\text{m}}{\text{s}^2}$)
Notes	The net force $\mathbf{F}$ on a body is proportional to the acceleration $\mathbf{a}(t)$ of the body, where m denotes the mass of the body as the constant of proportionality.
Source	–
RefBy	

4.2.3 General Definitions

This section collects the laws and equations that will be used to build the instance models.

Refname	**GD:velocityX1**
Label	The x-component of velocity of the first object
Units	$\frac{\text{m}}{\text{s}}$
Equation	$$v_{\text{x1}} = w_1 L_1 \cos\left(\theta_1\right)$$
Description	v_{x1} is the horizontal velocity of the first object ($\frac{\text{m}}{\text{s}}$) w_1 is the angular velocity of the first object ($\frac{\text{rad}}{\text{s}}$) L_1 is the length of the first rod (m) θ_1 is the angle of the first rod (rad)
Source	–
RefBy	

Detailed derivation of the x-component of velocity: At a given point in time, velocity is defined in DD:positionGDD

$$\mathbf{v}(t) = \frac{d\mathbf{p}(t)}{dt}$$

We also know the horizontal position that is defined in DD:positionXDD1

$$p_{\text{x1}} = L_1 \sin\left(\theta_1\right)$$

Applying this,

$$v_{\text{x1}} = \frac{dL_1 \sin\left(\theta_1\right)}{dt}$$

L_1 is constant with respect to time, so

$$v_{\text{x1}} = L_1 \frac{d\sin\left(\theta_1\right)}{dt}$$

Therefore, using the chain rule,

$$v_{\text{x1}} = w_1 L_1 \cos\left(\theta_1\right)$$

Refname	GD:velocityY1
Label	The y-component of velocity of the first object
Units	$\frac{\text{m}}{\text{s}}$
Equation	$$v_{\text{y1}} = w_1 L_1 \sin(\theta_1)$$
Description	v_{y1} is the vertical velocity of the first object ($\frac{\text{m}}{\text{s}}$) w_1 is the angular velocity of the first object ($\frac{\text{rad}}{\text{s}}$) L_1 is the length of the first rod (m) θ_1 is the angle of the first rod (rad)
Source	–
RefBy	

Detailed derivation of the y-component of velocity: At a given point in time, velocity is defined in DD:positionGDD

$$\mathbf{v}(t) = \frac{d\mathbf{p}(t)}{dt}$$

We also know the vertical position that is defined in DD:positionYDD1

$$p_{\text{y1}} = -L_1 \cos(\theta_1)$$

Applying this,

$$v_{\text{y1}} = -\left(\frac{dL_1 \cos(\theta_1)}{dt}\right)$$

L_1 is constant with respect to time, so

$$v_{\text{y1}} = -L_1 \frac{d \cos(\theta_1)}{dt}$$

Therefore, using the chain rule,

$$v_{\text{y1}} = w_1 L_1 \sin(\theta_1)$$

Refname	GD:velocityX2
Label	The x-component of velocity of the second object
Units	$\frac{\text{m}}{\text{s}}$
Equation	$$v_{\text{x2}} = v_{\text{x1}} + w_2 L_2 \cos(\theta_2)$$
Description	v_{x2} is the horizontal velocity of the second object ($\frac{\text{m}}{\text{s}}$) v_{x1} is the horizontal velocity of the first object ($\frac{\text{m}}{\text{s}}$) w_2 is the angular velocity of the second object ($\frac{\text{rad}}{\text{s}}$) L_2 is the length of the second rod (m) θ_2 is the angle of the second rod (rad)
Source	–
RefBy	

Detailed derivation of the x-component of velocity: At a given point in time, velocity is defined in DD:positionGDD

$$\mathbf{v}(t) = \frac{d\mathbf{p}(t)}{dt}$$

We also know the horizontal position that is defined in DD:positionXDD2

$$p_{\text{x2}} = p_{\text{x1}} + L_2 \sin(\theta_2)$$

Applying this,

$$v_{\text{x2}} = \frac{dp_{\text{x1}} + L_2 \sin(\theta_2)}{dt}$$

L_1 is constant with respect to time, so

$$v_{\text{x2}} = v_{\text{x1}} + w_2 L_2 \cos(\theta_2)$$

Refname	**GD:velocityY2**
Label	The y-component of velocity of the second object
Units	$\frac{\text{m}}{\text{s}}$
Equation	$$v_{\text{y2}} = v_{\text{y1}} + w_{2} L_{2} \sin\left(\theta_{2}\right)$$
Description	v_{y2} is the vertical velocity of the second object ($\frac{\text{m}}{\text{s}}$) v_{y1} is the vertical velocity of the first object ($\frac{\text{m}}{\text{s}}$) w_{2} is the angular velocity of the second object ($\frac{\text{rad}}{\text{s}}$) L_{2} is the length of the second rod (m) θ_{2} is the angle of the second rod (rad)
Source	–
RefBy	

Detailed derivation of the y-component of velocity: At a given point in time, velocity is defined in DD:positionGDD

$$\mathbf{v}(t) = \frac{d\mathbf{p}(t)}{dt}$$

We also know the vertical position that is defined in DD:positionYDD2

$$p_{\text{y2}} = p_{\text{y1}} - L_{2} \cos\left(\theta_{2}\right)$$

Applying this,

$$v_{\text{y2}} = -\left(\frac{dp_{\text{y1}} - L_{2} \cos\left(\theta_{2}\right)}{dt}\right)$$

Therefore, using the chain rule,

$$v_{\text{y2}} = v_{\text{y1}} + w_{2} L_{2} \sin\left(\theta_{2}\right)$$

Refname	GD:accelerationX1
Label	The x-component of acceleration of the first object
Units	$\frac{\text{m}}{\text{s}^2}$
Equation	$$a_{\text{x1}} = -{w_1}^2 L_1 \sin(\theta_1) + \alpha_1 L_1 \cos(\theta_1)$$
Description	a_{x1} is the horizontal acceleration of the first object ($\frac{\text{m}}{\text{s}^2}$) w_1 is the angular velocity of the first object ($\frac{\text{rad}}{\text{s}}$) L_1 is the length of the first rod (m) θ_1 is the angle of the first rod (rad) α_1 is the angular acceleration of the first object ($\frac{\text{rad}}{\text{s}^2}$)
Source	–
RefBy	IM:calOfAngularAcceleration2

Detailed derivation of the x-component of acceleration: Our acceleration is:

$$\mathbf{a}(t) = \frac{d\mathbf{v}(t)}{dt}$$

Earlier, we found the horizontal velocity to be

$$v_{\text{x1}} = w_1 L_1 \cos(\theta_1)$$

Applying this to our equation for acceleration

$$a_{\text{x1}} = \frac{d w_1 L_1 \cos(\theta_1)}{dt}$$

By the product and chain rules, we find

$$a_{\text{x1}} = \frac{dw_1}{dt} L_1 \cos(\theta_1) - w_1 L_1 \sin(\theta_1) \frac{d\theta_1}{dt}$$

Simplifying,

$$a_{\text{x1}} = -{w_1}^2 L_1 \sin(\theta_1) + \alpha_1 L_1 \cos(\theta_1)$$

Refname	GD:accelerationY1
Label	The y-component of acceleration of the first object
Units	$\frac{\text{m}}{\text{s}^2}$
Equation	$$a_{\text{y1}} = {w_1}^2 L_1 \cos(\theta_1) + \alpha_1 L_1 \sin(\theta_1)$$
Description	a_{y1} is the vertical acceleration of the first object ($\frac{\text{m}}{\text{s}^2}$) w_1 is the angular velocity of the first object ($\frac{\text{rad}}{\text{s}}$) L_1 is the length of the first rod (m) θ_1 is the angle of the first rod (rad) α_1 is the angular acceleration of the first object ($\frac{\text{rad}}{\text{s}^2}$)
Source	–
RefBy	IM:calOfAngularAcceleration2

Detailed derivation of the y-component of acceleration: Our acceleration is:

$$\mathbf{a}(t) = \frac{d\mathbf{v}(t)}{dt}$$

Earlier, we found the vertical velocity to be

$$v_{\text{y1}} = w_1 L_1 \sin(\theta_1)$$

Applying this to our equation for acceleration

$$a_{\text{y1}} = \frac{dw_1 L_1 \sin(\theta_1)}{dt}$$

By the product and chain rules, we find

$$a_{\text{y1}} = \frac{dw_1}{dt} L_1 \sin(\theta_1) + w_1 L_1 \cos(\theta_1) \frac{d\theta_1}{dt}$$

Simplifying,

$$a_{\text{y1}} = {w_1}^2 L_1 \cos(\theta_1) + \alpha_1 L_1 \sin(\theta_1)$$

Refname	GD:accelerationX2
Label	The x-component of acceleration of the second object
Units	$\frac{\text{m}}{\text{s}^2}$
Equation	$$a_{\text{x2}} = a_{\text{x1}} - {w_2}^2 L_2 \sin(\theta_2) + \alpha_2 L_2 \cos(\theta_2)$$
Description	a_{x2} is the horizontal acceleration of the second object ($\frac{\text{m}}{\text{s}^2}$) a_{x1} is the horizontal acceleration of the first object ($\frac{\text{m}}{\text{s}^2}$) w_2 is the angular velocity of the second object ($\frac{\text{rad}}{\text{s}}$) L_2 is the length of the second rod (m) θ_2 is the angle of the second rod (rad) α_2 is the angular acceleration of the second object ($\frac{\text{rad}}{\text{s}^2}$)
Source	–
RefBy	IM:calOfAngularAcceleration2

Detailed derivation of the x-component of acceleration: Our acceleration is:

$$\mathbf{a}(t) = \frac{d\mathbf{v}(t)}{dt}$$

Earlier, we found the horizontal velocity to be

$$v_{\text{x2}} = v_{\text{x1}} + w_2 L_2 \cos(\theta_2)$$

Applying this to our equation for acceleration

$$a_{\text{x2}} = \frac{dv_{\text{x1}} + w_2 L_2 \cos(\theta_2)}{dt}$$

By the product and chain rules, we find

$$a_{\text{x2}} = a_{\text{x1}} - {w_2}^2 L_2 \sin(\theta_2) + \alpha_2 L_2 \cos(\theta_2)$$

Refname	**GD:accelerationY2**
Label	The y-component of acceleration of the second object
Units	$\frac{\text{m}}{\text{s}^2}$
Equation	$$a_{\text{y2}} = a_{\text{y1}} + {w_2}^2 L_2 \cos(\theta_2) + \alpha_2 L_2 \sin(\theta_2)$$
Description	a_{y2} is the vertical acceleration of the second object ($\frac{\text{m}}{\text{s}^2}$) a_{y1} is the vertical acceleration of the first object ($\frac{\text{m}}{\text{s}^2}$) w_2 is the angular velocity of the second object ($\frac{\text{rad}}{\text{s}}$) L_2 is the length of the second rod (m) θ_2 is the angle of the second rod (rad) α_2 is the angular acceleration of the second object ($\frac{\text{rad}}{\text{s}^2}$)
Source	–
RefBy	IM:calOfAngularAcceleration2

Detailed derivation of the y-component of acceleration: Our acceleration is:

$$\mathbf{a}(t) = \frac{d\mathbf{v}(t)}{dt}$$

Earlier, we found the horizontal velocity to be

$$v_{\text{y2}} = v_{\text{y1}} + w_2 L_2 \sin(\theta_2)$$

Applying this to our equation for acceleration

$$a_{\text{y2}} = \frac{dv_{\text{y1}} + w_2 L_2 \sin(\theta_2)}{dt}$$

By the product and chain rules, we find

$$a_{\text{y2}} = a_{\text{y1}} + {w_2}^2 L_2 \cos(\theta_2) + \alpha_2 L_2 \sin(\theta_2)$$

Refname	**GD:xForce1**
Label	Horizontal force on the first object
Units	N
Equation	$$\mathbf{F} = m\mathbf{a}(t) = -\mathbf{T}_1 \sin(\theta_1) + \mathbf{T}_2 \sin(\theta_2)$$
Description	$\mathbf{F}$ is the force (N) m is the mass (kg) $\mathbf{a}(t)$ is the acceleration ($\frac{\text{m}}{\text{s}^2}$) $\mathbf{T}_1$ is the tension of the first object (N) θ_1 is the angle of the first rod (rad) $\mathbf{T}_2$ is the tension of the second object (N) θ_2 is the angle of the second rod (rad)
Source	–
RefBy	IM:calOfAngularAcceleration2

Detailed derivation of force on the first object:

$$\mathbf{F} = m\mathbf{a}(t) = -\mathbf{T}_1 \sin(\theta_1) + \mathbf{T}_2 \sin(\theta_2)$$

Refname	**GD:yForce1**
Label	Vertical force on the first object
Units	N
Equation	$$\mathbf{F} = m\mathbf{a}(t) = \mathbf{T}_1 \cos(\theta_1) - \mathbf{T}_2 \cos(\theta_2) - m_1\mathbf{g}$$
Description	$\mathbf{F}$ is the force (N) m is the mass (kg) $\mathbf{a}(t)$ is the acceleration ($\frac{\text{m}}{\text{s}^2}$) $\mathbf{T}_1$ is the tension of the first object (N) θ_1 is the angle of the first rod (rad) $\mathbf{T}_2$ is the tension of the second object (N) θ_2 is the angle of the second rod (rad) m_1 is the mass of the first object (kg) $\mathbf{g}$ is the gravitational acceleration ($\frac{\text{m}}{\text{s}^2}$)
Source	–
RefBy	IM:calOfAngularAcceleration2

Detailed derivation of force on the first object:

$$\mathbf{F} = m\mathbf{a}(t) = \mathbf{T}_1 \cos(\theta_1) - \mathbf{T}_2 \cos(\theta_2) - m_1\mathbf{g}$$

Refname	GD:xForce2
Label	Horizontal force on the second object
Units	N
Equation	$$\mathbf{F} = m\mathbf{a}(t) = -\mathbf{T}_2 \sin(\theta_2)$$
Description	$\mathbf{F}$ is the force (N) m is the mass (kg) $\mathbf{a}(t)$ is the acceleration ($\frac{\text{m}}{\text{s}^2}$) $\mathbf{T}_2$ is the tension of the second object (N) θ_2 is the angle of the second rod (rad)
Source	–
RefBy	IM:calOfAngularAcceleration2

Detailed derivation of force on the second object:

$$\mathbf{F} = m\mathbf{a}(t) = -\mathbf{T}_2 \sin(\theta_2)$$

Refname	**GD:yForce2**
Label	Vertical force on the second object
Units	N
Equation	$$\mathbf{F} = m\mathbf{a}(t) = \mathbf{T}_2 \cos(\theta_2) - m_2\mathbf{g}$$
Description	$\mathbf{F}$ is the force (N) m is the mass (kg) $\mathbf{a}(t)$ is the acceleration ($\frac{\mathrm{m}}{\mathrm{s}^2}$) $\mathbf{T}_2$ is the tension of the second object (N) θ_2 is the angle of the second rod (rad) m_2 is the mass of the second object (kg) $\mathbf{g}$ is the gravitational acceleration ($\frac{\mathrm{m}}{\mathrm{s}^2}$)
Source	–
RefBy	IM:calOfAngularAcceleration2

Detailed derivation of force on the second object:

$$\mathbf{F} = m\mathbf{a}(t) = \mathbf{T}_2 \cos(\theta_2) - m_2\mathbf{g}$$

4.2.4 Data Definitions

This section collects and defines all the data needed to build the instance models.

Refname	**DD:positionGDD**
Label	Velocity
Symbol	$\mathbf{v}(t)$
Units	$\frac{\text{m}}{\text{s}}$
Equation	$$\mathbf{v}(t) = \frac{d\mathbf{p}(t)}{dt}$$
Description	$\mathbf{v}(t)$ is the velocity ($\frac{\text{m}}{\text{s}}$) t is the time (s) $\mathbf{p}(t)$ is the position (m)
Source	–
RefBy	GD:velocityY2, GD:velocityY1, GD:velocityX2, and GD:velocityX1

Refname	DD:positionXDD1
Label	Horizontal position of the first object
Symbol	p_{x1}
Units	m
Equation	$$p_{\mathrm{x1}} = L_1 \sin\left(\theta_1\right)$$
Description	p_{x1} is the horizontal position of the first object (m) L_1 is the length of the first rod (m) θ_1 is the angle of the first rod (rad)
Notes	p_{x1} is the horizontal position p_{x1} is shown in Fig:dblpend.
Source	–
RefBy	GD:velocityX1

Refname	**DD:positionYDD1**
Label	Vertical position of the first object
Symbol	p_{y1}
Units	m
Equation	$$p_{\text{y1}} = -L_1 \cos\left(\theta_1\right)$$
Description	p_{y1} is the vertical position of the first object (m) L_1 is the length of the first rod (m) θ_1 is the angle of the first rod (rad)
Notes	p_{y1} is the vertical position p_{y1} is shown in Fig:dblpend.
Source	–
RefBy	GD:velocityY1

Refname	**DD:positionXDD2**
Label	Horizontal position of the second object
Symbol	p_{x2}
Units	m
Equation	$$p_{\text{x2}} = p_{\text{x1}} + L_2 \sin(\theta_2)$$
Description	p_{x2} is the horizontal position of the second object (m) p_{x1} is the horizontal position of the first object (m) L_2 is the length of the second rod (m) θ_2 is the angle of the second rod (rad)
Notes	p_{x2} is the horizontal position p_{x2} is shown in Fig:dblpend.
Source	–
RefBy	GD:velocityX2

Refname	DD:positionYDD2
Label	Vertical position of the second object
Symbol	p_{y2}
Units	m
Equation	$$p_{\text{y2}} = p_{\text{y1}} - L_2 \cos(\theta_2)$$
Description	p_{y2} is the vertical position of the second object (m) p_{y1} is the vertical position of the first object (m) L_2 is the length of the second rod (m) θ_2 is the angle of the second rod (rad)
Notes	p_{y2} is the vertical position p_{y2} is shown in Fig:dblpend.
Source	–
RefBy	GD:velocityY2

Refname	**DD:accelerationGDD**
Label	Acceleration
Symbol	$\mathbf{a}(t)$
Units	$\frac{\text{m}}{\text{s}^2}$
Equation	$$\mathbf{a}(t) = \frac{d\mathbf{v}(t)}{dt}$$
Description	$\mathbf{a}(t)$ is the acceleration ($\frac{\text{m}}{\text{s}^2}$) t is the time (s) $\mathbf{v}(t)$ is the velocity ($\frac{\text{m}}{\text{s}}$)
Source	–
RefBy	

Refname	DD:forceGDD
Label	Force
Symbol	$\mathbf{F}$
Units	N
Equation	$$\mathbf{F} = m\mathbf{a}(t)$$
Description	$\mathbf{F}$ is the force (N) m is the mass (kg) $\mathbf{a}(t)$ is the acceleration ($\frac{\text{m}}{\text{s}^2}$)
Source	–
RefBy	

4.2.5 Instance Models

This section transforms the problem defined in the problem description into one which is expressed in mathematical terms. It uses concrete symbols defined in the data definitions to replace the abstract symbols in the models identified in theoretical models and general definitions.

Refname	IM:calOfAngularAcceleration1
Label	Calculation of angular acceleration
Input	L_1, L_2, m_1, m_2, θ_1, θ_2
Output	α_1
Input Constraints	$L_1 > 0$ $L_2 > 0$ $m_1 > 0$ $m_2 > 0$
Output Constraints	$\alpha_1 > 0$
Equation	$\alpha_1\left(\theta_1, \theta_2, w_1, w_2\right) = \frac{-g\left(2m_1 + m_2\right)\sin\left(\theta_1\right) - m_2 g \sin\left(\theta_1 - 2\theta_2\right) - 2\sin\left(\theta_1 - \theta_2\right) m_2 \left({w_2}^2\right.}{L_1\left(2m_1 + m_2 - m_2\cos\left(2\theta_1 - 2\theta_2\right)\right)}$
Description	α_1 is the angular acceleration of the first object ($\frac{\text{rad}}{\text{s}^2}$) θ_1 is the angle of the first rod (rad) θ_2 is the angle of the second rod (rad) w_1 is the angular velocity of the first object ($\frac{\text{rad}}{\text{s}}$) w_2 is the angular velocity of the second object ($\frac{\text{rad}}{\text{s}}$) g is the magnitude of gravitational acceleration ($\frac{\text{m}}{\text{s}^2}$) m_1 is the mass of the first object (kg) m_2 is the mass of the second object (kg) L_2 is the length of the second rod (m) L_1 is the length of the first rod (m)
Source	–
RefBy	FR:Output-Values, FR:Calculate-Angular-Position-Of-Mass, and IM:calOfAngularAcceleration2

Refname	**IM:calOfAngularAcceleration2**
Label	Calculation of angular acceleration
Input	L_1, L_2, m_1, m_2, θ_1, θ_2
Output	α_2
Input Constraints	$L_1 > 0$ $L_2 > 0$ $m_1 > 0$ $m_2 > 0$
Output Constraints	$\alpha_2 > 0$
Equation	$$\alpha_2\left(\theta_1, \theta_2, w_1, w_2\right) = \frac{2\sin\left(\theta_1 - \theta_2\right)\left({w_1}^2 L_1\left(m_1 + m_2\right) + g\left(m_1 + m_2\right)\cos\left(\theta_1\right) + {w_2}^2 L_2 m_2 \mathrm{c}\right.}{L_2\left(2m_1 + m_2 - m_2\cos\left(2\theta_1 - 2\theta_2\right)\right)}$$
Description	α_2 is the angular acceleration of the second object ($\frac{\text{rad}}{\text{s}^2}$) θ_1 is the angle of the first rod (rad) θ_2 is the angle of the second rod (rad) w_1 is the angular velocity of the first object ($\frac{\text{rad}}{\text{s}}$) w_2 is the angular velocity of the second object ($\frac{\text{rad}}{\text{s}}$) L_1 is the length of the first rod (m) m_1 is the mass of the first object (kg) m_2 is the mass of the second object (kg) g is the magnitude of gravitational acceleration ($\frac{\text{m}}{\text{s}^2}$) L_2 is the length of the second rod (m)
Source	–
RefBy	FR:Output-Values, FR:Calculate-Angular-Position-Of-Mass, and IM:calOfAngularAcceleration2

Detailed derivation of angle of the second rod: By solving equations GD:xForce2 and GD:yForce2 for $\mathbf{T}_2 \sin(\theta_2)$ and $\mathbf{T}_2 \cos(\theta_2)$ and then substituting into equation GD:xForce1 and GD:yForce1 , We can get equations 1 and 2:

$$m_1 a_{\text{x}1} = -\mathbf{T}_1 \sin(\theta_1) - m_2 a_{\text{x}2}$$

$$m_1 a_{\text{y}1} = \mathbf{T}_1 \cos(\theta_1) - m_2 a_{\text{y}2} - m_2 g - m_1 g$$

Multiply the equation 1 by $\cos(\theta_1)$ and the equation 2 by $\sin(\theta_1)$ and rearrange to get:

$$\mathbf{T}_1 \sin(\theta_1) \cos(\theta_1) = -\cos(\theta_1) (m_1 a_{\text{x}1} + m_2 a_{\text{x}2})$$

$$\mathbf{T}_1 \sin(\theta_1) \cos(\theta_1) = \sin(\theta_1) \left(m_1 a_{\text{y}1} + m_2 a_{\text{y}2} + m_2 g + m_1 g\right)$$

This leads to the equation 3

$$\sin(\theta_1) \left(m_1 a_{\text{y}1} + m_2 a_{\text{y}2} + m_2 g + m_1 g\right) = -\cos(\theta_1) (m_1 a_{\text{x}1} + m_2 a_{\text{x}2})$$

Next, multiply equation GD:xForce2 by $\cos(\theta_2)$ and equation GD:yForce2 by $\sin(\theta_2)$ and rearrange to get:

$$\mathbf{T}_2 \sin(\theta_2) \cos(\theta_2) = -\cos(\theta_2) m_2 a_{\text{x}2}$$

$$\mathbf{T}_1 \sin(\theta_2) \cos(\theta_2) = \sin(\theta_2) \left(m_2 a_{\text{y}2} + m_2 g\right)$$

which leads to equation 4

$$\sin(\theta_2) \left(m_2 a_{\text{y}2} + m_2 g\right) = -\cos(\theta_2) m_2 a_{\text{x}2}$$

By giving equations GD:accelerationX1 and GD:accelerationX2 and GD:accelerationY1 and GD:accelerationY2 plus additional two equations, 3 and 4, we can get IM:calOfAngularAcceleration1 and IM:calOfAngularAcceleration2 via a computer algebra program:

4.2.6 Data Constraints

The Data Constraints Table shows the data constraints on the input variables. The column for physical constraints gives the physical limitations on the range of values that can be taken by the variable. The uncertainty column provides an estimate of the confidence with which the physical quantities can be measured. This information would be part of the input if one were performing an uncertainty quantification exercise. The constraints are conservative to give the user of the model the flexibility to experiment with unusual situations. The column of typical values is intended to provide a feel for a common scenario.

Table 4: Input Data Constraints

Var	Physical Constraints	Typical Value	Uncert.
L_1	$L_1 > 0$	1.0 m	10%
L_2	$L_2 > 0$	1.0 m	10%
m_1	$m_1 > 0$	0.5 kg	10%
m_2	$m_2 > 0$	0.5 kg	10%

4.2.7 Properties of a Correct Solution

The Data Constraints Table shows the data constraints on the output variables. The column for physical constraints gives the physical limitations on the range of values that can be taken by the variable.

Table 5: Output Data Constraints

Var	Physical Constraints
θ_1	$\theta_1 > 0$
θ_2	$\theta_2 > 0$

5 Requirements

This section provides the functional requirements, the tasks and behaviours that the software is expected to complete, and the non-functional requirements, the qualities that the software is expected to exhibit.

5.1 Functional Requirements

This section provides the functional requirements, the tasks and behaviours that the software is expected to complete.

Input-Values: Input the values from Tab:ReqInputs.

erify-Input-Values: Check the entered input values to ensure that they do not exceed the data constraints. If any of the input values are out of bounds, an error message is displayed and the calculations stop.

-Position-Of-Mass: Calculate the following values: α_1 and α_2 (from IM:calOfAngularAcceleration1) (from IM:calOfAngularAcceleration2).

Output-Values: Output α_1 and α_2 (from IM:calOfAngularAcceleration1 and IM:calOfAngularAcceleration2).

Table 6: Required Inputs following FR:Input-Values

Symbol	Description	Units
L_1	Length of the first rod	m
L_2	Length of the second rod	m
m_1	Mass of the first object	kg
m_2	Mass of the second object	kg

5.2 Non-Functional Requirements

This section provides the non-functional requirements, the qualities that the software is expected to exhibit.

Correct: The outputs of the code have the properties of a correct solution.

Portable: The code is able to be run in different environments.

6 Traceability Matrices and Graphs

The purpose of the traceability matrices is to provide easy references on what has to be additionally modified if a certain component is changed. Every time a component is changed, the items in the column of that component that are marked with an "X" should be modified as well. Tab:TraceMatAvsA shows the dependencies of the assumptions on each other. Tab:TraceMatAvsAll shows the dependencies of the data definitions, theoretical models, general definitions, instance models, requirements, likely changes, and unlikely changes on the assumptions. Tab:TraceMatRefvsRef shows the dependencies of the data definitions, theoretical models, general definitions, and instance models on each other. Tab:TraceMatAllvsR shows the dependencies of the requirements and goal statements on the data definitions, theoretical models, general definitions, and instance models.

Table 7: Traceability Matrix Showing the Connections Between Assumptions and Other A

	A:twoDMotion	**A:cartSys**	**A:cartSysR**	**A:yAxisDir**	**A:startOrigin**	**A:firs**
A:twoDMotion						
A:cartSys						
A:cartSysR						
A:yAxisDir						
A:startOrigin						
A:firstPend						

Table 7: Traceability Matrix Showing the Connections Between Assumptions and Other Assump

	A:twoDMotion	**A:cartSys**	**A:cartSysR**	**A:yAxisDir**	**A:startOrigin**	**A:firs**
A:secondPend						

Table 8: Traceability Matrix Showing the Connections Between Assumpti

	A:twoDMotion	**A:cartSys**	**A:cartSysR**	**A:yAxisDir**
DD:positionGDD				
DD:positionXDD1				
DD:positionYDD1				
DD:positionXDD2				
DD:positionYDD2				
DD:accelerationGDD				
DD:forceGDD				
TM:acceleration				
TM:velocity				
TM:NewtonSecLawMot				
GD:velocityX1				
GD:velocityY1				
GD:velocityX2				
GD:velocityY2				
GD:accelerationX1				
GD:accelerationY1				
GD:accelerationX2				
GD:accelerationY2				
GD:xForce1				
GD:yForce1				
GD:xForce2				
GD:yForce2				
IM:calOfAngularAcceleration1				
IM:calOfAngularAcceleration2				
FR:Input-Values				

Table 8: Traceability Matrix Showing the Connections Between Assumptions an

	A:twoDMotion	**A:cartSys**	**A:cartSysR**	**A:yAxisDir**
FR:Verify-Input-Values				
FR:Calculate-Angular-Position-Of-Mass				
FR:Output-Values				
NFR:Correct				
NFR:Portable				

	DD:positionGDD	**DD:positionXDD1**	**DD:positionYDD1**	**D**
DD:positionGDD				
DD:positionXDD1				
DD:positionYDD1				
DD:positionXDD2				
DD:positionYDD2				
DD:accelerationGDD				
DD:forceGDD				
TM:acceleration				
TM:velocity				
TM:NewtonSecLawMot				
GD:velocityX1	X	X		
GD:velocityY1	X		X	
GD:velocityX2	X			X
GD:velocityY2	X			
GD:accelerationX1				
GD:accelerationY1				
GD:accelerationX2				
GD:accelerationY2				
GD:xForce1				
GD:yForce1				
GD:xForce2				

Figure 3: TraceGraphAvsA

	DD:positionGDD	**DD:positionXDD1**	**DD:positionYDD1**	**D**
GD:yForce2				
IM:calOfAngularAcceleration1				
IM:calOfAngularAcceleration2				

	DD:positionGDD	**DD:positionXDD1**	**DD:positionY**
GS:motionMass			
FR:Input-Values			
FR:Verify-Input-Values			
FR:Calculate-Angular-Position-Of-Mass			
FR:Output-Values			
NFR:Correct			
NFR:Portable			

The purpose of the traceability graphs is also to provide easy references on what has to be additionally modified if a certain component is changed. The arrows in the graphs represent dependencies. The component at the tail of an arrow is depended on by the component at the head of that arrow. Therefore, if a component is changed, the components that it points to should also be changed. Fig:TraceGraphAvsA shows the dependencies of assumptions on each other. Fig:TraceGraphAvsAll shows the dependencies of data definitions, theoretical models, general definitions, instance models, requirements, likely changes, and unlikely changes on the assumptions. Fig:TraceGraphRefvsRef shows the dependencies of data definitions, theoretical models, general definitions, and instance models on each other. Fig:TraceGraphAllvsR shows the dependencies of requirements and goal statements on the data definitions, theoretical models, general definitions, and instance models. Fig:TraceGraphAllvsAll shows the dependencies of dependencies of assumptions, models, definitions, requirements, goals, and changes with each other.

For convenience, the following graphs can be found at the links below:

Figure 4: TraceGraphAvsAll

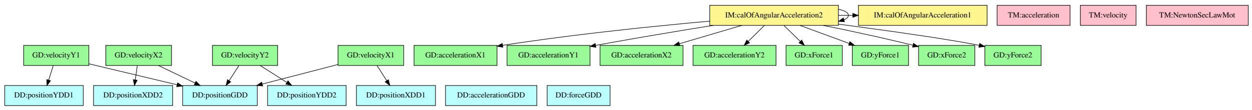

Figure 5: TraceGraphRefvsRef

- TraceGraphAvsA
- TraceGraphAvsAll
- TraceGraphRefvsRef
- TraceGraphAllvsR
- TraceGraphAllvsAll

7 Values of Auxiliary Constants

There are no auxiliary constants.

8 References

[1] Wikipedia Contributors. *Acceleration*. https://en.wikipedia.org/wiki/Acceleration. June 2019.

[2] Wikipedia Contributors. *Cartesian coordinate system*. https://en.wikipedia.org/wiki/Cartesian_coordinate_system. June 2019.

[3] Wikipedia Contributors. *Velocity*. https://en.wikipedia.org/wiki/Velocity. June 2019.

[4] R. C. Hibbeler. *Engineering Mechanics: Dynamics*. Pearson Prentice Hall, 2004.

[5] Nirmitha Koothoor. “A Document Driven Approach to Certifying Scientific Computing Software”. MA thesis. Hamilton, ON, Canada: McMaster University, 2013.

Figure 6: TraceGraphAllvsR

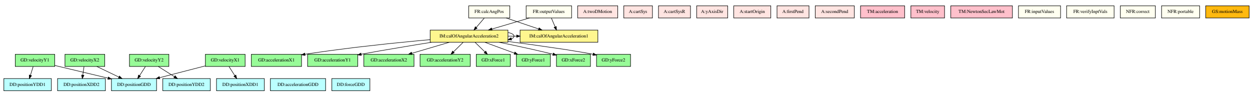

Figure 7: TraceGraphAllvsAll

[6] David L. Parnas and P. C. Clements. “A rational design process: How and why to fake it”. In: *IEEE Transactions on Software Engineering* 12.2 (Feb. 1986), pp. 251–257.

[7] W. Spencer Smith and Nirmitha Koothoor. “A Document-Driven Method for Certifying Scientific Computing Software for Use in Nuclear Safety Analysis”. In: *Nuclear Engineering and Technology* 48.2 (Apr. 2016), pp. 404–418.

[8] W. Spencer Smith and Lei Lai. “A new requirements template for scientific computing”. In: *Proceedings of the First International Workshop on Situational Requirements Engineering Processes - Methods, Techniques and Tools to Support Situation-Specific Requirements Engineering Processes, SREP’05*. Ed. by PJ Agerfalk, N. Kraiem, and J. Ralyte. In conjunction with 13th IEEE International Requirements Engineering Conference, Paris, France, 2005, pp. 107–121.

[9] W. Spencer Smith, Lei Lai, and Ridha Khedri. “Requirements Analysis for Engineering Computation: A Systematic Approach for Improving Software Reliability”. In: *Reliable Computing, Special Issue on Reliable Engineering Computation* 13.1 (Feb. 2007), pp. 83–107.