

Software Requirements Specification for CVT Simulator: subtitle describing software

Team #17, Baja Dynamics

Grace McKenna

Travis Wing

Cameron Dunn

Kai Arseneau

October 11, 2024

Contents

1	Reference Material	iv
1.1	Table of Units	iv
1.2	Table of Symbols	iv
1.3	Abbreviations and Acronyms	vi
1.4	Mathematical Notation	vi
2	Introduction	1
2.1	Purpose of Document	1
2.2	Scope of Requirements	1
2.3	Characteristics of Intended Reader	1
2.4	Organization of Document	2
3	General System Description	2
3.1	System Context	2
3.2	User Characteristics	3
3.3	System Constraints	3
4	Specific System Description	3
4.1	Problem Description	3
4.1.1	Terminology and Definitions	4
4.1.2	Physical System Description	5
4.1.3	Goal Statements	7
4.2	Solution Characteristics Specification	7
4.2.1	Types	8
4.2.2	Scope Decisions	8
4.2.3	Modelling Decisions	8
4.2.4	Assumptions	8
4.2.5	Theoretical Models	9
4.2.6	General Definitions	23
4.2.7	Data Definitions	24
4.2.8	Data Types	27
4.2.9	Instance Models	28
4.2.10	Input Data Constraints	36
4.2.11	Properties of a Correct Solution	38
5	Requirements	38
5.1	Functional Requirements	38
5.2	Nonfunctional Requirements	40
5.3	Rationale	40
6	Likely Changes	41

7	Unlikely Changes	41
8	Traceability Matrices and Graphs	41
9	Development Plan	44
10	Values of Auxiliary Constants	44

Revision History

Date	Version	Notes
Date 1	1.0	Notes
Date 2	1.1	Notes

1 Reference Material

This section records information for easy reference.

1.1 Table of Units

Throughout this document SI (Système International d'Unités) is employed as 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
°C	temperature	centigrade
J	energy	joule
W	power	watt ($W = J s^{-1}$)
N	force	newtons ($N = kg m s^{-2}$)
Nm	torque	newton metre ($Nm = kg m^2 s^{-2}$)
rad	angle	radian

1.2 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units. The choice of symbols was made to be consistent with the heat transfer literature and with existing documentation for solar water heating systems. The symbols are listed in alphabetical order.

symbol	unit	description
d_{center}	m	center to center distance
d_{prim}	m	primary spring pre-compression distance
d_{sec}	m	secondary spring pre-compression distance
d_{shift}	m	shift distance
f_{engine}	N m	engine torque function
f_{prim_angle}	rad	primary ramp geometry function
f_{prim_height}	m	primary ramp height function
f_{sec_ramp}	rad	secondary ramp geometry function
f_{sec_shift}	rad	secondary ramp rotation function

h_{belt}	m	belt height
h_{prim}	m	primary ramp output height
k_{prim}	N/m	primary spring rate
$k_{\text{sec.comp}}$	N/m	secondary compression spring rate
$k_{\text{sec.tor}}$	N m/rad	secondary torsional spring rate
l_{belt}	m	belt length
m_{belt}	kg	belt weight
m_{d}	kg	driver weight
m_{fly}	kg	flyweight mass
m_{v}	kg	vehicle weight
R_{gearbox}		gearbox reduction ratio
r_{fly}	m	flyweight radius
$r_{\text{prim.shaft}}$	m	primary shaft initial radius
$r_{\text{sec.shaft}}$	m	secondary shaft initial radius
$r_{\text{sec.ramp}}$	m	secondary ramp radius
T_{eng}	N m	engine torque
w_{belt}	m	belt width
ω_{engine}	rad/s	engine angular velocity
μ_{air}		air resistance coefficient
μ_{belt}		belt friction coefficient
μ_{traction}		traction coefficient
ϕ_{sheave}	rad	angle between sheaves
θ_{inc}	rad	angle on incline
θ_{prim}	rad	primary ramp output angle
θ_{sec}	rad	secondary spring pre-torsional rotation
θ_{shift}	rad	secondary shift rotation
θ_{ramp}	rad	secondary ramp output angle

1.3 Abbreviations and Acronyms

symbol	description
A	Assumption
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
TM	Theoretical Model
RPM	Revolutions Per Minute
CVT	Continuous Variable Transmission
SEC	Secondary
PRIM	Primary

1.4 Mathematical Notation

N/A

2 Introduction

The McMaster Baja engineering team is facing challenges in tuning their Continuous Variable Transmission (CVT). The current process of tuning the CVT is extensive, time consuming and requires testing of multiple physical components leading to possible inconsistencies due to weather and wear. These inconsistencies then complicate the tuning process of the CVT's torque transfer directly affecting the performance of the Baja vehicles. To address these issues, our team aims to develop a CVT simulation tool that uses mathematical models, a rendering engine and a user-friendly interface. Our solution aims to streamline the optimization process allowing Baja members to simulate real-world factors and virtually test various tuning parameters. Our simulation tool will be validated against data collected by the McMaster Baja Data Acquisition team to ensure reliability and accuracy.

This introduction will outline the key objectives of this document to provide a structured guide for the development of the CVT tool and outline of the system's requirements. Additionally, the scope will explore technical and functional assumptions and will intend to streamline the complex tuning process of a CVT. A detailed overview of this document will provide guidance to the reader regarding the structure and flow of the document.

2.1 Purpose of Document

The purpose of this Software Requirements Specification (SRS) is to provide an outline of the system requirements given for the development of a CVT simulation tool. This ensures that all stakeholders including members of the Baja team, Dr. Smith and the developers understand the projects objectives and constraints. This document will serve as a communication tool to align expectations, provide guidance during the design of this tool and aid in the development and testing phases of this project. This SRS will be referenced throughout the development of the system to ensure the tool is within the scope and meets the outlined requirements.

2.2 Scope of Requirements

This model will simulate as a closed system and will not consider environmental factors such as temperature, humidity, and wind. It also will not consider any unintended damage, forces or effects on the components of the CVT system. The system will not account for any relative displacement between the components of the CVT system due to driver input or external forces. Further, the system will be delimited to 2 spatial dimensions, being forwards/backwards and up/down, and will not consider any lateral movement of the vehicle.

2.3 Characteristics of Intended Reader

Readers or reviewers of this SRS document should have a solid understanding of the principles of Newtonian physics and basic calculus. Specifically, it is recommended that they

have completed courses in first-year mechanics, Calculus I, Calculus II, and Calculus III. Knowledge of the components in a Continuous Variable Transmission (CVT) and a basic understanding of how these parts work together will also be beneficial. Proficiency in these areas will be sufficient to comprehend the material in this document.

2.4 Organization of Document

This document is structured as follows:

- Section 3 discusses the general context and description of the system
- Section 4 details the specific system description, goals, and definitions
- Section 5 covers the system requirements
- Section 6 outlines the likely changes for the system
- Section 7 discusses the unlikely changes for the system
- Section 8 covers the traceability of the requirements

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

Figure 1 shows the system context. A circle represents an external entity outside the software. A rectangle represents the software system itself (CVT Simulator). Arrows are used to show the data flow between the system and its environment.

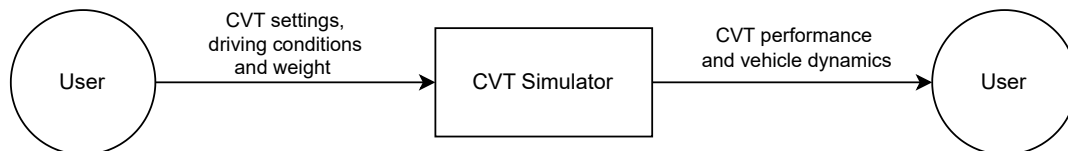


Figure 1: System Context

- User Responsibilities:

- Provide the input data related to the tunable components of the CVT system.
-
- CVT Simulator Responsibilities:
 - Accept user input data to simulate the CVT system, validating against set constraints.
 - Calculate the output data based on the input data and the CVT system model.
 - Display the results of the simulation.

3.2 User Characteristics

The expected user of this application is someone who is very familiar with drivetrains and CVTs but not necessarily familiar with software. The software expects the user to know all of the components of a CVT and how they can be modified or adjusted. The user should also have enough of an understanding of undergraduate physics and calculus to be able to make sense of the outputs of the software. On the software side, the user should be able to navigate a GUI and know how to input data into the software.

3.3 System Constraints

The system should be able to run on any machine that supports Python 3.9 or later and Unity projects.

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, definitions and finally the instance models.

4.1 Problem Description

The CVT simulator is intended to address the challenges faced by the McMaster Baja racing team when they tune their CVT. The current tuning process is time-consuming and requires physical testing of multiple components and configurations. This leads to inconsistencies in the tuning process due to weather and wear. The CVT simulator will allow the Baja team to virtually test different tuning parameters and simulate real-world factors that affect the CVT's performance.

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:

- RPM - Revolutions Per Minute, a measure of rotational speed.
- CVT - Continuous Variable Transmission
- Torque - The rotational force produced by the engine or seen as load by the wheels.
- Primary RPM - The RPM of the primary CVT.
- Secondary RPM - The RPM of the secondary CVT.
- Belt - The rubber V-belt that connects the primary and secondary CVT.
- Primary ramp - The ramp that the fly weights in the primary CVT push against. Highlighted green in Figure 2.
- Flyweight - The weights that experience centrifugal force and push against the primary ramp. Highlighted yellow in Figure 2.
- Helix - The helical ramp that the secondary CVT pushes against. Highlighted bronze in Figure 4.
- CVT System - This is the entire system that includes the primary and secondary CVT and the belt. The primary CVT is connected to the engine and the secondary CVT is connected to the wheels.

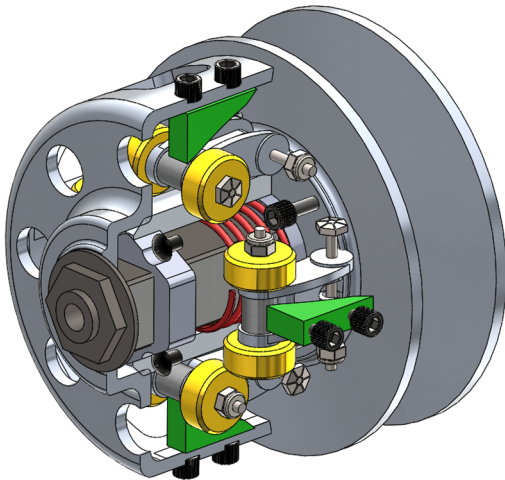


Figure 2: Primary CVT

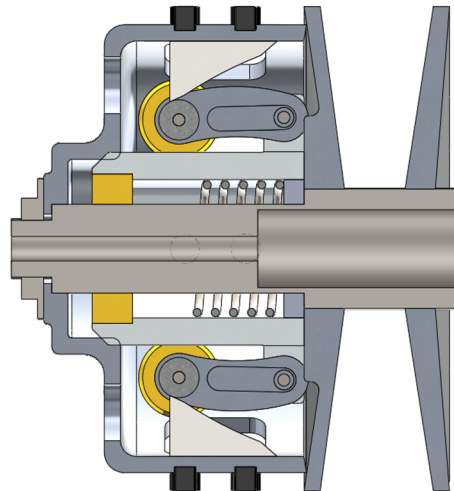


Figure 3: Primary CVT Side View

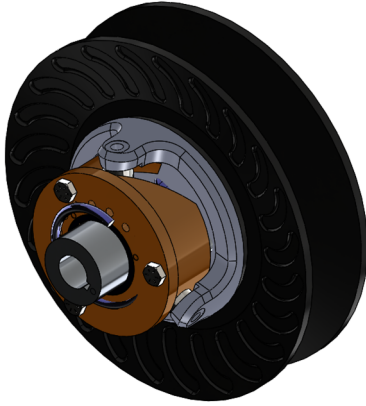


Figure 4: Secondary CVT

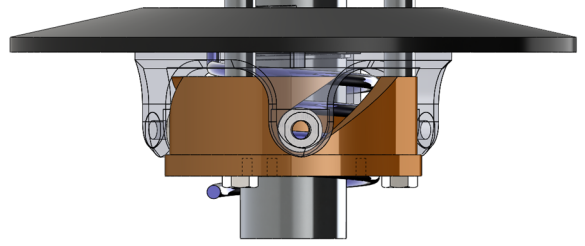


Figure 5: Secondary CVT Side View

Note: While the McMaster Baja Team's CVT differs slightly from these figures, its mechanics remain widely equivalent.

4.1.2 Physical System Description

The physical system of CVT Simulator, as shown in Figures 2-5, includes the following elements:

PS1: Flyweights

PS1a: mass (m_{fly}) in kg.

PS1b: initial radius (r_{fly}) in m.

PS2: Primary spring

PS2a: spring constant (k_{prim}) in N/m.

PS2b: pre-compression distance (d_{prim}) in m.

PS3: Primary ramp geometry function ($f_{\text{prim_angle}}$)

PS3a: shift distance (d_{shift}) in m.

PS3b: output angle (θ_{prim}) in radians.

PS4: Primary ramp height function ($f_{\text{prim_height}}$)

PS4a: shift distance (d_{shift}) in m.

PS4b: output height (h_{prim}) in m.

PS5: Secondary spring

PS5a: compression spring constant ($k_{\text{sec_comp}}$) in N/m.

PS5b: torsional spring rate ($k_{\text{sec_tor}}$) in Nm/rad.

PS5c: pre-compression distance (d_{sec}) in m.

PS5d: pre-torsional rotation (θ_{sec}) in radians.

PS6: Secondary ramp geometry function ($f_{\text{sec_ramp}}$)

PS6a: shift distance (d_{shift}) in m.

PS6b: output angle (θ_{ramp}) in radians.

PS7: Secondary ramp rotation function ($f_{\text{sec_shift}}$)

PS7a: shift distance (d_{shift}) in m.

PS7b: output angle (θ_{shift}) in radians.

PS8: Weight

PS8a: driver weight (m_{d}) in kg.

PS8b: vehicle weight (m_{v}) in kg.

PS9: Engine torque function (f_{engine})

PS9a: input angular velocity (ω_{engine}) in rad/s.

PS9b: output torque (T_{eng}) in Nm.

PS10: Belt

PS18a: belt length (l_{belt}) in m.

PS18b: belt width (w_{belt}) in m.

PS18c: belt height (h_{belt}) in m.

PS18d: belt weight (m_{belt}) in kg.

PS10b: friction coefficient (μ_{belt}), unitless.

PS11: Total reduction

PS11a: gearbox reduction ratio (R_{gearbox}), unitless.

PS11b: wheel radius (r_{wheel}) in m.

PS12: Coefficient of air resistance (μ_{air}), unitless.

PS13: Center to Center distance (d_{center}) in m.

PS14: Angle on Incline (θ_{inc}) in radians.

PS15: Secondary Ramp Radius ($r_{\text{sec_ramp}}$) in m.

PS16: Traction (μ_{traction}), unitless.

PS16: CVT Geometry

PS16a: Primary shaft initial radius ($r_{\text{prim_shaft}}$) in m.

PS16b: Secondary shaft initial radius ($r_{\text{sec_shaft}}$) in m.

PS17: Angle between sheaves (ϕ_{sheave}) in radians.

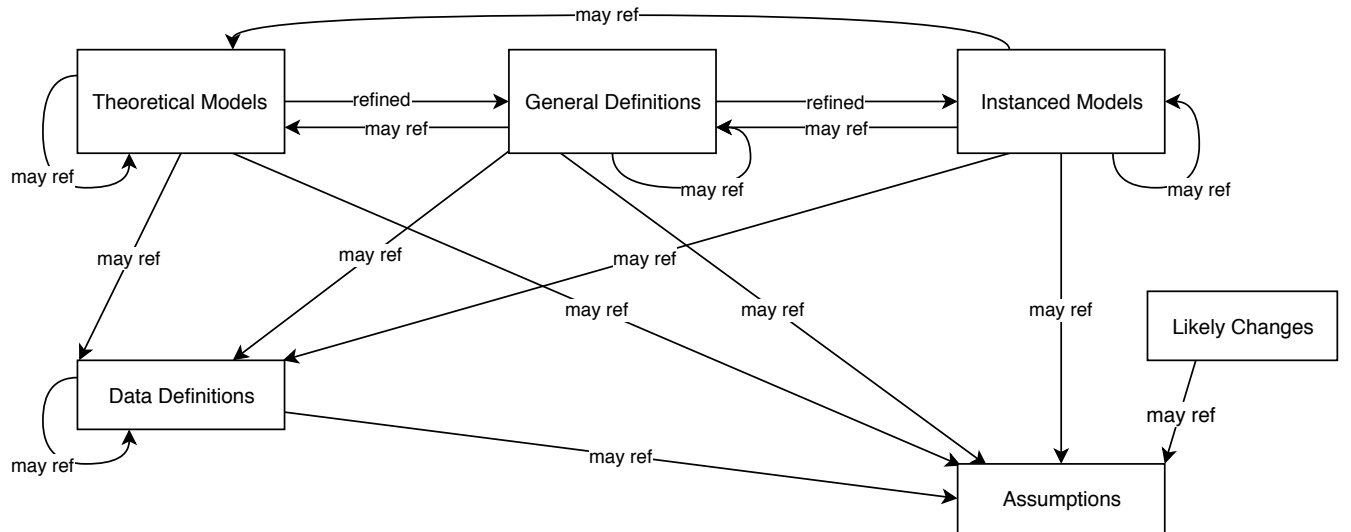
4.1.3 Goal Statements

GS1: Simulates the kinematics of the vehicle

GS2: Simulates the output of the engine

GS3: Simulates the CVT system over time

4.2 Solution Characteristics Specification



The instance models that govern CVT Simulator are presented in Subsection 4.2.9. The information to understand the meaning of the instance models and their derivation is also presented, so that the instance models can be verified.

4.2.1 Types

N/A

4.2.2 Scope Decisions

N/A

4.2.3 Modelling Decisions

N/A

4.2.4 Assumptions

- A1: **No Material Variations By Temperature** Ignore the effect of temperature on the material properties, excluding the belt.
- A2: **Negligible Gravitational Influence:** The orientation of the CVT system and gravitational effects are considered negligible due to the high rotational speeds involved in the system's operation.
- A3: **Rigidity of Belts:** Elasticity of belts are assumed negligible through its length, and the components are considered ideally rigid under normal operating conditions. The belt may still rotate about the pulleys.
- A4: **No Vibrational Effects:** Vibrational effects from the vehicle or external environment are assumed to be minimal and are not factored into the transmission's operation in this model.
- A5: **Ignore Wear and Tear:** Component wear over time, including belt/chain wear and pulley degradation, is assumed negligible in the short-term operational model.
- A6: **No Frictional Losses in Non-critical Components:** Frictional losses in non-critical components (e.g., bearings, shafts) are assumed negligible, focusing only on friction relevant to the CVT system's core mechanics.
- A7: **Full Throttle Consistent Power Engine:** The engine is modeled to operate at full throttle with consistent power curves and peak performance, maintaining a uniform response throughout the analysis. Load feedback will be integrated to adjust RPM and power output as necessary.
- A8: **Perfect Axis Alignment:** The CVT's axes are aligned with one another.
- A9: **No Belt Slippage:** The CVT will not slip relative to the sheaves.
- A10: **Ignore Irrelevant Forces:** Only forces that directly contribute to the car's dynamics or the CVT's shifting behaviours are considered, allowing the simplification of dimensions where applicable.

4.2.5 Theoretical Models

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

General Theories

This portion will specify the existing theories that are used in the development of the CVT simulation tool.

RefName: TM:HL

Label: Hookes Law (Displacement)

Equation: $F = k\Delta x$

Description: F is the force (N)

k is the spring constant (N/m)

Δx is the displacement (m)

Notes: In the context of this project Hookes law is used in both a torsional and linear way.

Source: <https://www.britannica.com/science/Hookes-law>

Ref. By: IM4

Preconditions for TM:HL: None

Derivation for TM:HL: Not Applicable

RefName: TM:CF

Label: Centrifugal Force

Equation: $F_c = m\omega^2 r$

Description: F is the force (N)

m is the mass (kg)

ω is the angular velocity (rad/s)

r is the radius (m)

Notes: None.

Source: <https://www.omnicalculator.com/physics/centrifugal-force>

Ref. By: IM4

Preconditions for TM:CF: None

Derivation for TM:CF: Not Applicable

RefName: TM:AR

Label: Air Resistance

Equation: $F_d = \frac{1}{2}\rho v^2 C_d A$

Description: F is the force (N)
 ρ is the density of the fluid (kg/m³)
 v is the velocity of the object (m/s)
 C_d is the drag coefficient
 A is the cross-sectional area of the object (m²)

Notes: None.

Source: https://softschools.com/formulas/physics/air_resistance_formula/85/

Ref. By: IM1

Preconditions for TM:AR: None

Derivation for TM:AR: Not Applicable

RefName: TM:ET

Label: Engine Torque

Equation: $\tau = \frac{P}{\omega}$

Description: τ is the torque (Nm)

P is the power (W)

ω is the angular velocity (rad/s)

Notes: None.

Source: <https://powertestdyno.com/how-to-calculate-horsepower/>

Ref. By:

Preconditions for TM:ET: None

Derivation for TM:ET: Not Applicable

RefName: **TM:GR**

Label: Gearing

Equation: $\tau_{\text{output}} = \tau_{\text{input}} \times R_{\text{gear}}$

Description: τ_{output} is the output torque (Nm)

τ_{input} is the input torque (Nm)

R_{gear} is the gear reduction (unitless)

Notes: None.

Source: <https://science.howstuffworks.com/transport/engines-equipment/gear-ratio.htm>

Ref. By: IM1

Preconditions for TM:GR: None

Derivation for TM:GR: Not Applicable

RefName: TM:FR

Label: Static Friction Formula

Equation: $F_f = \mu_s F_n$

Description: μ_s is the coefficient of friction (unitless)

F_f is the force of friction (N)

F_n is the normal force (N)

Notes: None.

Source: <https://en.wikipedia.org/wiki/Friction>

Ref. By: IM1, IM7

Preconditions for TM:FR: None

Derivation for TM:FR: Not Applicable

RefName: TM:DE

Label: Density

Equation: $\rho = \frac{m}{V}$

Description: ρ is the density (kg/m³)
 m is the mass (kg)
 V is the volume (m³)

Notes: None.

Source: <https://www.britannica.com/science/density-physics>

Ref. By: IM1

Preconditions for TM:DE: None

Derivation for TM:DE: Not Applicable

RefName: TM:CE

Label: Capstan Equation

Equation: $T_1 = T_2 e^{\mu\theta}$

Description: T_1 is the tension on the first side of the belt (N)

T_2 is the tension on the second side of the belt (N)

μ is the coefficient of friction

θ is the angle of wrap of the belt (radians)

Notes: None.

Source: <https://hackaday.com/2021/01/26/cable-mechanism-maths-designing-against-the-capstan-equation/>

Ref. By: IM7

Preconditions for TM:CE: None

Derivation for TM:CE: Not Applicable

RefName: TM:N2

Label: Newtons 2nd Law

Equation: $F = ma$

Description: F is the force (N)

m is the mass (kg)

a is the acceleration (m/s^2)

Notes: Newtons 2nd law is used repeatedly in several Instance models

Source: <https://www.physicsclassroom.com/class/newtlaws/lesson-3/newton-s-second-law>

Ref. By: IM1, IM6, IM7

Preconditions for TM:N2: None

Derivation for TM:N2: Not Applicable

Helper Theories

This portion will define auxiliary theories to aid in further derivation below.

RefName: **TM:D2R**

Label: Shift Distance to Pulley Radius

Equation: $\Delta r = \frac{d_{\text{shift}}}{2 \tan(\frac{\phi}{2})}$

Description: d_{shift} is the axial shift distance, or sheave displacement (m)
 Δr is the change in effective radius (m)
 ϕ is the angle between sheaves (rad)

Notes: Derived based on geometry of the sheaves and belt

Source: See below for derivation

Ref. By:

Preconditions for [TM:D2R](#): None

Derivation for [TM:D2R](#): Not Applicable

RefName: TM:D2RA

Label: Shift Distance to Ramp Angle

Equation: $\theta_{\text{ramp}} = f_{\text{prim}}(d_{\text{shift}})$

Description: d_{shift} is the axial shift distance, or sheave displacement (m)

θ_{ramp} is the instantaneous ramp angle (rad)

$f_{\text{prim}}(d)$ is the function to express the ramp geometry (m \rightarrow rad)

Notes: The user will input the geometry of the ramp, which will then be interpreted to generate f_{prim}

Source: See below for derivation

Ref. By:

Preconditions for [TM:D2RA](#): None

Derivation for [TM:D2RA](#): Not Applicable

RefName: TM:D2HA

Label: Shift Distance to Helix Angle

Equation: $\theta_{\text{ramp}} = f_{\text{sec}}(d_{\text{shift}})$

Description: d_{shift} is the axial shift distance, or sheave displacement (m)

θ_{ramp} is the instantaneous helix angle (rad)

$f_{\text{prim}}(d)$ is the function to express the helix geometry (m \rightarrow rad)

Notes: The user will input the geometry of the helix, which will then be interpreted to generate f_{sec}

Source: See below for derivation

Ref. By:

Preconditions for TM:D2HA: None

Derivation for TM:D2HA: Not Applicable

RefName: TM:HT

Label: Hookes Law (Torsional)

Equation: $F = k\Delta x$

Description: F is the force (N)
 k is the spring constant (Nm/rad)
 Δx is the displacement (m)

Notes: In the context of this project Hookes law is used in both a torsional and linear way.

Source: <https://www.britannica.com/science/Hookes-law>

Ref. By:

Preconditions for TM:HT: None

Derivation for TM:HT: Not Applicable

4.2.6 General Definitions

Number	GD1
Label	Helix Side Force
SI Units	$\text{N} = \text{kg m s}^{-2}$
Equation	$F_H = \frac{T_{\text{eng}} R_{\text{CVT}}}{2r \tan(\beta)}$
Description	<p>F_H is the side force due to the helix (N)</p> <p>T_{eng} is the engine torque (N-m)</p> <p>R_{CVT} is the CVT ratio (unitless)</p> <p>r is the radius of the ramps from the shaft (m)</p> <p>β is the helix angle (rad)</p>
Source	
Ref. By	IM5

Number	GD2
Label	Spring Side Force
SI Units	$\text{N} = \text{kg m s}^{-2}$
Equation	$F_S = \frac{k_{\text{tor}}(\theta_0 + \theta)}{2r \tan(\beta)} + k_{\text{comp}}(x_0 + x)$
Description	<p>F_S is the side force due to the spring (N)</p> <p>k_{tor} is the torsional spring constant (N m/rad)</p> <p>θ_0 is the initial torsion (rad)</p> <p>θ is the amount of angular torsion (rad)</p> <p>k_{comp} is the compression spring constant (N/m)</p> <p>x_0 is the initial compression (m)</p> <p>x is the amount of compression (m)</p> <p>r is the radius of the ramps from the shaft (m)</p> <p>θ is the amount of angular torsion (rad)</p>
Source	
Ref. By	IM5

4.2.7 Data Definitions

This section collects and defines all the data needed to build the instance models. The dimension of each quantity is also given.

Number	DD1
Label	Coefficient of Friction
Symbol	μ
SI Units	unitless
Equation	$F_f = \mu F_n$
Description	μ is the coefficient of friction F_f is the force of friction F_n is the normal force
Sources	None.
Ref. By	IM1, IM7

Number	DD2
Label	Compressional Spring Constant
Symbol	k
SI Units	N/m
Equation	$F = k\Delta x$
Description	k is the spring constant (N/m) F is the force (N) Δx is the displacement (m)
Sources	None.
Ref. By	IM4

Number	DD3
Label	Torsional Spring Constant
Symbol	k
SI Units	N m/rad
Equation	$\tau = k\Delta\theta$
Description	k is the torsional spring constant (N m/rad) τ is the torque (N m) θ is the angle of rotation (radians)
Sources	None.
Ref. By	IM5

Number	DD4
Label	Gravitational Constant
Symbol	g
SI Units	m/s ²
Equation	$F_g = mg$
Description	g is the gravitational constant (m/s ²) F_g is the force of gravity m is the mass
Sources	None.
Ref. By	IM1

Number	DD5
Label	Drag Coefficient
Symbol	C_d
SI Units	unitless
Equation	$F_d = \frac{1}{2}\rho v^2 C_d A$
Description	C_d is the drag coefficient F_d is the force of drag ρ is the density of the fluid v is the velocity of the object A is the cross-sectional area of the object
Sources	None.
Ref. By	IM1

Number	DD6
Label	Rolling Resistance Coefficient
Symbol	C_{rr}
SI Units	unitless
Equation	$F_{rr} = C_{rr} F_n$
Description	C_{rr} is the rolling resistance coefficient F_{rr} is the force of rolling resistance F_n is the normal force
Sources	None.
Ref. By	IM1

4.2.8 Data Types

N/A

4.2.9 Instance Models

This section transforms the problem defined in Section 4.1 into one which is expressed in mathematical terms. It uses concrete symbols defined in Section 4.2.7 to replace the abstract symbols in the models identified in Sections 4.2.5 and 4.2.6.

The goal statement GS1 are solved by Instance Models 1, 2 and 3

The goal statement GS2 are solved by Instance Models 8

The goal statement GS3 are solved by Instance Models 4, 5, 6 and 7

Number	IM1
Label	Acceleration
Input	$r_{\text{gear}}, r_{\text{wheel}}, \rho, C_D, A, \theta, T_{\text{CVT}}, C_{\text{rr}}, m_v, m_d$
Equation	$a(t) = \frac{\left(\frac{T_{\text{eng}}(t) \cdot R_{\text{cvt}}(t)}{r_{\text{gear}} + r_{\text{wheel}}} \right) - (C_{\text{rr}} F_g \sin(\theta_{\text{inc}})) - \left(\frac{1}{2} \rho v^2 C_D A \right) - (F_g \cos(\theta_{\text{inc}}))}{m_v + m_d}$
Output	acceleration $a(t)$
Description	<p>r_{gear} is the radius of the gear (m)</p> <p>r_{wheel} is the radius of the wheel (m)</p> <p>ρ is the density of the fluid (kg/m³)</p> <p>C_D is the drag coefficient (unitless)</p> <p>A is the cross-sectional area of the object (m²)</p> <p>θ_{inc} is the angle of incline (rad)</p> <p>C_{rr} is the coefficient of rolling resistance (unitless)</p> <p>F_g is the force of gravity (N)</p> <p>m_v is the mass of the vehicle (kg)</p> <p>m_d is the mass of the driver(kg)</p> <p>$a(t)$ is the function of acceleration over time (m/s²)</p> <p>$R_{\text{cvt}}(t)$ is the current ratio of the CVT as a function of time (unitless)</p> <p>$T_{\text{eng}}(t)$ is the torque seen at the engine as a function of time (Nm)</p>
Sources	IM5
Ref. By	-

Derivation of Acceleration

Using [Newton's 2nd Law](#), the acceleration of the vehicle is derived as follows:

$$T_W - R_r - F_D - F_g \cos(\theta_{\text{inc}}) = ma$$

Where T_W is the torque of the wheel, F_r is the rolling resistance, F_D is the drag force, F_g is the gravitational force, and m is the mass of the vehicle.

$$T_W = \frac{T_{\text{eng}}(t) \cdot R_{\text{cvt}}(t)}{r_{\text{gear}} + r_{\text{wheel}}} \quad (\text{see TM:GR})$$

,

$$F_r = C_{\text{rr}} F_g \sin(\theta_{\text{inc}}) \quad (\text{see TM:FR})$$

$$F_D = \frac{1}{2} \rho v^2 C_D A \quad (\text{see TM:AR})$$

$$m = m_v + m_d$$

Substituting the above equations in and solving for a gives us the final equation:

$$a(t) = \frac{\left(\frac{T_{\text{eng}}(t) \cdot R_{\text{cvt}}(t)}{r_{\text{gear}} + r_{\text{wheel}}} \right) - (C_{\text{rr}} F_g \sin(\theta_{\text{inc}})) - \left(\frac{1}{2} \rho v^2 C_D A \right) - (F_g \cos(\theta))}{m_v + m_d}$$

Number	IM2
Label	Velocity
Input	$a(t)$
Equation	$v(t) = \int a(t) dt$
Output	$v(t)$
Description	$a(t)$ is the function of acceleration over time
Sources	-
Ref. By	
Number	IM3
Label	Distance
Input	$v(t)$
Equation	$d(t) = \int v(t) dt$
Output	$d(t)$
Description	$v(t)$ is the function of velocity over time
Sources	-
Ref. By	

Number	IM4
Label	Primary Clamping Force
Input	$m_{\text{fly}}, r_{\text{fly}}, f_{\text{prim_height}}(d_{\text{shift}}), f_{\text{prim_angle}}(d_{\text{shift}}(t)), k_{\text{prim}}, d_{\text{prim}}, d_{\text{shift}}$
Equation	$F_{\text{prim_clamp}}(t) = m_{\text{fly}}(r_{\text{fly}} + f_{\text{prim_height}}(d_{\text{shift}}(t)))(\omega_{\text{eng}}(t))^2 \tan(f_{\text{prim_angle}}(d_{\text{shift}}(t))) + k_{\text{prim}}(d_{\text{prim}} + d_{\text{shift}}(t))$
Output	The primary clamping force $F_{\text{prim_clamp}}$
Description	<p>m_{fly}: Mass of the flyweight system. (kg)</p> <p>r_{fly}: Initial radius of the flyweights. (m)</p> <p>$f_{\text{prim_height}}(d_{\text{shift}})$: Function that represents the height of the flyweights based on the shift distance. (rad)</p> <p>$f_{\text{prim_angle}}(d_{\text{shift}})$: Function that returns the instantaneous angle of the ramps based on the shifting distance. (rad)</p> <p>k_{prim}: Primary spring coefficient. (unitless)</p> <p>$d_{\text{shift}}(t)$: Displacement of the sheave as a function of time. (m)</p> <p>$\omega_{\text{eng}}(t)$: Angular velocity of the engine as a function of time. (rad/s)</p> <p>d_{prim}: Initial compression distance of the primary spring. (m)</p>
Sources	
Ref. By	IM6

Derivation of Primary Clamping Force

The flyweight experiences a [centrifugal force](#) $F_c = mr\omega^2$ as it rotates. The ramp angle $f_{\text{prim_angle}}(d_{\text{shift}})$ is where the force applied by the flyweight meets the inclined surface. The force can be broken into the radial force, which is perpendicular to the ramp, and the axial force which acts in the direction of the ramp. The axial force is what is responsible for moving the mechanism causing the flyweights to press against a ramp or surface, therefore the force we are interested in. We can ignore the radial force as it acts perpendicular to the ramp and does not contribute to the axial force. (see [A10](#)) To find the flywheel force in this axial direction the tangent of the angle $f_{\text{prim_angle}}(d_{\text{shift}})$ is used. The flywheel force is derived as follows:

$$F_{FW} = mr\omega^2 \tan(f_{\text{prim_angle}}(d_{\text{shift}})) \quad (\text{see } \text{TM:CF})$$

Where r can be represented by $r_{\text{fly}} + f_{\text{prim_height}}(d_{\text{shift}})$. Then F_{FW} becomes

$$F_{FW} = m_{\text{fly}}(r_{\text{fly}} + f_{\text{prim_height}}(d_{\text{shift}}))\omega^2 \tan(f_{\text{prim_angle}}(d_{\text{shift}}))$$

The spring force is derived as follows:

$$-F_S = +k_{\text{prim}}(d_{\text{prim}} + d_{\text{shift}}) \quad (\text{see } \text{TM:HL})$$

Combining these two forces gives the total primary clamping force:

$$F_{FW} - F_S = m_{\text{fly}}(r_{\text{fly}} + f_{\text{prim_height}}(d_{\text{shift}}))\omega^2 \tan(f_{\text{prim_angle}}(d_{\text{shift}})) + k_{\text{prim}}(d_{\text{prim}} + d_{\text{shift}})$$

The final equation can be written as:

$$F_{\text{prim_clamp}}(t) = m_{\text{fly}}(r_{\text{fly}} + f_{\text{prim_height}}(d_{\text{shift}}(t)))(\omega_{\text{eng}}(t))^2 \tan(f_{\text{prim_angle}}(d_{\text{shift}}(t))) + k_{\text{prim}}(d_{\text{prim}} + d_{\text{shift}}(t))$$

Number	IM5
Label	Secondary Clamping Force
Input	$k_{\text{sec_tor}}, \theta_{\text{sec}}, r_{\text{sec_ramp}}, f_{\text{sec_ramp}}(d_{\text{shift}}), f_{\text{sec_shift}}(d_{\text{shift}}), k_{\text{sec_comp}}, d_{\text{sec}}, d_{\text{shift}}(t), T_{\text{eng}}(t), R_{\text{cvt}}(t)$
Equation	$F_{\text{sec_clamp}}(t) = \frac{k_{\text{sec_tor}}(\theta_{\text{sec}} + f_{\text{sec_shift}}(d_{\text{shift}}(t))) + T_{\text{eng}}(t)R_{\text{cvt}}(t)}{2r_{\text{sec_ramp}} \tan(f_{\text{sec_ramp}}(d_{\text{shift}}(t)))} + k_{\text{sec_comp}}(d_{\text{sec}} + d_{\text{shift}}(t))$
Output	Secondary clamping force $F_{\text{sec_clamp}}$
Description	<p>$k_{\text{sec_tor}}$: Torsional spring rate. (Nm/rad)</p> <p>θ_{sec}: Initial torsional rotation. (rad)</p> <p>$f_{\text{sec}}(d_{\text{shift}})$: Amount of torsional rotation based on the distance shifted. (rad)</p> <p>$r_{\text{sec_ramp}}$: Radius of the helix from the shaft. (m)</p> <p>$k_{\text{sec_comp}}$: Compression spring rate. (N/m)</p> <p>d_{sec}: Initial compression distance. (m)</p> <p>d_{shift}: Displacement of the sheave as a function of time. (m)</p> <p>$T_{\text{eng}}(t)$: Torque seen at the engine as a function of time. (Nm)</p> <p>$R_{\text{cvt}}(t)$: Ratio of the CVT as a function of time. (unitless)</p>
Sources	
Ref. By	IM6

Derivation of Secondary Clamping Force

The secondary clamping force is derived as follows:

$$F_{\text{sec.clamp}} = F_H + F_S$$

The helix force is derived as follows:

$$F_H = \frac{T_{eng}(t)R_{cvt}(t)}{2r \tan(\beta)} \quad (\text{see GD2})$$

Substituting in our variables gives us the following equation:

$$F_H = \frac{T_{eng}(t)R_{cvt}(t)}{2r_{\text{sec.ramp}} \tan(f_{\text{sec.ramp}}(d_{\text{shift}}(t)))}$$

The spring force is derived as follows:

$$F_S = \frac{k_{\text{tor}}(\theta_0 + \theta)}{2r \tan(\beta)} + k_{\text{comp}}(x_0 + x) \quad (\text{see GD1})$$

Substituting in our variables gives us the following equation:

$$F_S = \frac{k_{\text{sec.tor}}(\theta_{\text{sec}} + f_{\text{sec.shift}}(d_{\text{shift}}(t)))}{2r_{\text{sec.ramp}} \tan(f_{\text{sec.ramp}}(d_{\text{shift}}(t)))} + k_{\text{sec.comp}}(d_{\text{sec}} + d_{\text{shift}}(t))$$

Combining these two forces gives the total secondary clamping force:

$$F_H + F_S = \frac{T_{eng}(t)R_{cvt}(t)}{2r_{\text{sec.ramp}} \tan(f_{\text{sec.ramp}}(d_{\text{shift}}(t)))} + \frac{k_{\text{sec.tor}}(\theta_{\text{sec}} + f_{\text{sec.shift}}(d_{\text{shift}}(t)))}{2r_{\text{sec.ramp}} \tan(f_{\text{sec.ramp}}(d_{\text{shift}}(t)))} + k_{\text{sec.comp}}(d_{\text{sec}} + d_{\text{shift}}(t))$$

The final equation can be reached through simplifying the above equation by combining like terms:

$$F_{\text{sec.clamp}} = \frac{k_{\text{sec.tor}}(\theta_{\text{sec}} + f_{\text{sec.shift}}(d_{\text{shift}}(t))) + T_{eng}(t)R_{cvt}(t)}{2r_{\text{sec.ramp}} \tan(f_{\text{sec.ramp}}(d_{\text{shift}}(t)))} + k_{\text{sec.comp}}(d_{\text{sec}} + d_{\text{shift}}(t))$$

Number	IM6
Label	Sheave Acceleration
Input	$F_{\text{clamp_prim}}, F_{\text{clamp_sec}}, m_{\text{sheaves}}$
Equation	$a(t) = \frac{F_{\text{clamp_prim}}(t) - F_{\text{clamp_sec}}(t)}{m_{\text{sheaves}}}$
Output	Sheave acceleration a
Description	$F_{\text{clamp_prim}}$: Primary clamping force $F_{\text{clamp_sec}}$: Secondary clamping force m_{sheaves} : Mass of the movable sheaves, combined
Sources	
Ref. By	-

Derivation of Sheave Acceleration

One can derive the acceleration of the sheaves using [Newton's Second Law](#). The forces acting on the built include only the primary and secondary clamping forces, which act in opposite directions as they act out radially. The acceleration of the sheave is derived as follows:

$$ma(t) = F_{\text{clamp_prim}}(t) - F_{\text{clamp_sec}}(t)$$

Rearranging the equation for acceleration gives the final equation:

$$a(t) = \frac{F_{\text{clamp_prim}}(t) - F_{\text{clamp_sec}}(t)}{m}$$

Number	IM7
Label	Belt Acceleration
Input	$T_{\text{eng}}, r_{\text{prim}}, m_{\text{belt}}$
Equation	$a_{\text{belt}} = \frac{T_{\text{eng}}}{mr_{\text{prim}}}$
Output	Belt Acceleration a_{belt}
Description	T_{eng} : Engine torque r_{prim} : Primary radiu m_{belt} : Mass of the belt.
Sources	
Ref. By	IM6

Derivation of Belt Acceleration

The torque of the engine equation in terms of belt tension is derived as such:

$$T_{\text{eng}} = (T_T - T_s)r_{\text{prim}} \quad \Rightarrow \quad T_T - T_s = \frac{T_{\text{eng}}}{r_{\text{prim}}}$$

Similarly, for the load torque:

$$T_{\text{load}} = (T_T - T_s)r_{\text{sec}} \quad \Rightarrow \quad T_T - T_s = \frac{T_{\text{load}}}{r_{\text{sec}}}$$

Equating the two torque equations:

$$\frac{T_{\text{eng}}}{r_{\text{prim}}} = \frac{T_{\text{load}}}{r_{\text{sec}}} \quad (\text{see A9})$$

The equation for the torque of the taut side is derived using the [capstan equation](#):

$$T_T = T_s e^{\mu\theta} \quad (\text{TM:CE})$$

Substituting our engine torque into this:

$$\frac{T_{\text{eng}}}{r_{\text{prim}}} + T_s = T_s e^{\mu\theta}$$

Solving for T_s (slack side):

$$T_s = \frac{T_{\text{eng}}}{r_{\text{prim}}(e^{\mu\theta} - 1)}$$

Using [Newton's 2nd law](#) the final equation for belt acceleration is derived as:

$$m_{\text{belt}} a_{\text{belt}} = \frac{T_{\text{eng}} e^{\mu\theta}}{r_{\text{prim}}(e^{\mu\theta} - 1)} - \frac{T_{\text{eng}}}{r_{\text{prim}}(e^{\mu\theta} - 1)} \quad (\text{TM:N2})$$

Solving for acceleration:

$$a_{\text{belt}} = \frac{T_{\text{eng}}}{m_{\text{belt}} r_{\text{prim}}(e^{\mu\theta} - 1)}(e^{\mu\theta} - 1)$$

Finally, simplifying to get:

$$a_{\text{belt}} = \frac{T_{\text{eng}}}{m_{\text{belt}} r_{\text{prim}}}$$

Number	IM8
Label	RPM and Torque of Engine
Input	t, T_{load}
Equation	$T_{\text{eng}}, \omega_{\text{eng}} = F(t, T_{\text{load}})$
Output	$T_{\text{eng}}, \omega_{\text{eng}}$
Description	<p>T_{eng}: Torque seen at the engine</p> <p>ω_{eng}: Angular velocity of the engine</p> <p>These are functions that are defined in a library. It will be passed the torque curve of the engine at 100% load and calculate the torque and RPM at based on the time and the load of the system.</p>
Sources	
Ref. By	

4.2.10 Input Data Constraints

Table 2 shows the data constraints on the input output variables. The column for physical constraints gives the physical limitations on the range of values that can be taken by the variable. The column for software constraints restricts the range of inputs to reasonable values. The software constraints will be helpful in the design stage for picking suitable algorithms. 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. 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 specification parameters in Table 2 are listed in Table 4.

Table 2: Input Variables

Var	Physical Constraints	Software Constraints	Typical Value	Uncertainty
m_{fly}	$0 \leq m_{fly}$	$m_{fly} \leq 1$	0.2 kg	
ω_{engine}	$0 \leq \omega_{engine}$	$\omega_{engine} \leq \omega_{\max}$	400 rad/s	
d_{shift}	$0 \leq d_{shift} \leq w_{belt}$		0.1 m	
θ_{prim_out}	$0 \leq \theta_{prim_out} \leq \frac{\pi}{2}$		$\frac{\pi}{4}$ rad	
d_{prim}	$0 \leq d_{prim} \leq d_{\max_prim_pre}$		0.1 m	
d_{sec}	$0 \leq d_{sec} \leq d_{\max_sec_pre}$		0.1 rad	
m_{driver}	$0 \leq m_{driver}$	$m_{driver} \leq m_{\max_human}$	75 kg	
m_{car}	$0 \leq m_{car}$	$m_{car} \leq m_{\max_car}$	225 kg	
k_{prim}	$0 \leq k_{prim}$	$k_{prim} \leq k_{\max_comp_spring}$	100 N/m	
k_{sec_comp}	$0 \leq k_{sec_comp}$	$k_{sec_comp} \leq k_{\max_comp_spring}$	100 N/m	
k_{sec_tor}	$0 \leq k_{sec_tor}$	$k_{sec_tor} \leq k_{\max_tor_spring}$	50 Nm/rad	

Table 4: Specification Parameter Values

Var	Value
L_{\min}	0.1 m
ω_{\max}	600 rad/s
w_{belt}	UNKNOWN, see belt spec m
$d_{\text{max_prim_pre}}$	UNKNOWN, see CAD m
$d_{\text{max_sec_pre}}$	UNKNOWN, see CAD rad
$m_{\text{max_human}}$	200 kg
$m_{\text{max_car}}$	350 kg
$k_{\text{max_comp_spring}}$	1000 N/m
$k_{\text{max_tor_spring}}$	750 Nm/rad
v_{\max}	UNKNOWN, do gear calcs m/s
$f_{\text{belt_max}}$	UNKNOWN, see belt spec N

4.2.11 Properties of a Correct Solution

Table 6: Output Variables

Var	Physical Constraints	unit
T_W	$T_{\text{init}} \leq T_W \leq T_C$ (by A??)	unitless
$v(t)$	$0 \leq v(t) \leq v_{\max}$	m/s
Sum of clamping forces	$\text{Clamp} \leq f_{\text{belt_max}}$	N

5 Requirements

This section provides the functional requirements, the business tasks that the software is expected to complete, and the nonfunctional requirements, the qualities that the software is expected to exhibit.

5.1 Functional Requirements

- R1: The system shall simulate the behavior of the McMaster Baja Team's Continuous Variable Transmission(CVT) using mathematical models.
- R2: The system shall calculate the acceleration of the vehicle as a function of time using inputs: gear radius, wheel radius, density of the fluid, drag coefficient, cross sectional area, angle of incline, torque transferred to the wheel, coefficient of rolling resistance, mass of the vehicle and driver.
- R3: The system shall calculate the velocity of the vehicle as a function of time by taking the integral of acceleration with respect to time.
- R4: The system shall calculate the distance as a function of time by taking the integral of the velocity with respect to time.
- R5: The system shall calculate the primary clamping force using inputs: mass of the fly-weight, radius related to the flywheel, height of the primary system, force dependent of the shifting distance, stiffness of primary system and displacement or shift in the system.
- R6: The system shall calculate the secondary clamping force using inputs: torsional spring rate, initial torsional rotation, amount of torsional rotation based on the distance shifted, radius of the ramps from the shaft, compression spring rate, initial compression distance, displacement in system, torque of the engine and ratio of the CVT system.
- R7: The system shall calculate the sheave acceleration using inputs: primary clamping force, secondary clamping force, coefficient of friction, torque of the taut side of the belt, torque of the slack side of the belt and mass of the system.
- R8: The system shall calculate the belt acceleration using inputs engine torque, primary radius, friction force and mass of the system.
- R9: The system shall calculate RPM and Torque of the Engine.
- R10: The system shall allow users to adjust the following CVT tuning parameters: primary weight, primary ramp geometry, primary spring rate, primary spring pretension, secondary helix geometry, secondary spring rate, secondary spring pretension.
- R11: The system shall allow users to adjust vehicle and driver weight, traction and angle of incline in addition to the CVT tuning parameters.
- R12: The system shall provide users with an interface to input tuning parameters.
- R13: The system shall display graphs of simulation output based on the user inputted tuning parameters.
- R14: The system shall allow users to compare results of simulations with different CVT tuning parameters.

- R15: The system shall allow users to export outputted simulated graphical data.
- R16: The system shall allow users to access the application through the use of a personal computer or laptop.
- R17: The system shall be compatible with Windows, MacOS and Linux.

5.2 Nonfunctional Requirements

- NFR1: **Accuracy** The system shall achieve an accuracy of at least 95 % in correctly predicting CVT outputs.
- NFR2: **Usability** 85 % of a representative user group shall be able to successfully input CVT parameters and receive CVT outputs on their first use.
- NFR3: **Maintainability** If a likely change is made to the finished software, it will take at most 10 % of the original development time, assuming the same development resources are available.
- NFR4: **Verifiability** The system shall output the resulting tuned CVT outputs which must be able to be cross-verified against data available through the McMaster Baja Team with a consistency rate of 95 %.
- NFR5: **Understandability** The system shall be understood by at least 90 % of a representative user group with no more than 15 minutes of initial training, as measured by assessing ability to generate and export tuned CVT output where users must score at least 90 %
- NFR6: **Reusability** The system's components shall be designed for reuse in the case the McMaster Baja team acquires a new CVT. These modifications should be made with minimal modifications, defined as requiring modification to less than 30 % of the code to configure these changes.

5.3 Rationale

The assumptions made for the CVT (Continuously Variable Transmission) system model aim to simplify its dynamics, facilitating the development of mathematical models to simulate the behavior of the CVT(R1). Many key assumptions are made to meet the requirement of developing these mathematical models. The following assumptions are also factored in to the calculations of the CVT ratio, clamping force, RPM, torque, engine input, distance, speed and acceleration(R2).

The systems excludes the effects of temperature on material properties, aside from the belt, which allows for a more straightforward focus on the mechanics of the interactions (A1). The

influence of gravitational forces is negligible and the elasticity of belts is considered insignificant(A2, A3). Vibrational effects from external sources are also assumed to be minimal, avoiding the complexity of vibration-induced disturbances in transmission behavior (A4). Wear and tear of components, including the belt and pulley are disregarded and frictional losses in non-critical components, are deemed negligible (A5, A6). The engine is assumed to operate at full throttle with consistent power delivery, simplifying the model by maintaining uniform engine behavior (A4). Finally, the CVT axis are assumed to be in line and the belt slippage of the CVT is assumed as negligible (A8, A9).

These simplifications affect the process of creating our mathematical models and formulating equations governing velocity, distance, primary clamping force, secondary clamping force, CVT Ratio, Torque transfer and RPM and Torque of the Engine(IM1-IM8)

6 Likely Changes

LC1: The effect of temperature on material properties might be included in the future, (A1)

LC2: The elasticity of the belt might be considered in the future, (A3)

LC3: Frictional losses in non-critical components might be considered in the future, (A6)

LC4: Some of the math models will most likely change as the design is refined.

7 Unlikely Changes

None

8 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” may have to be modified as well. Table 11 shows the dependencies of theoretical models, general definitions, data definitions, and instance models with each other. Table 13 shows the dependencies of instance models, requirements, and data constraints on each other. Table 10 shows the dependencies of theoretical models, general definitions, data definitions, instance models, and likely changes on the assumptions.

[You will have to modify these tables for your problem. —TPLT]

[The traceability matrix is not generally symmetric. If GD1 uses A1, that means that GD1’s derivation or presentation requires invocation of A1. A1 does not use GD1. A1 is “used by” GD1. —TPLT]

	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??	A??
TM??	X																		
TM??																			
TM??																			
GD??		X																	
GD??			X	X	X	X													
DD??							X	X	X										
DD??			X	X						X									
DD??																			
DD??																			
IM??											X	X		X	X	X			X
IM??												X	X			X	X	X	
IM??														X					X
IM??													X					X	
LC??				X															
LC??								X											
LC??									X										
LC??											X								
LC??												X							
LC??															X				

Table 9: Traceability Matrix Showing the Connections Between Assumptions and Other Items

	A1	A2	A3	A4	A5	A6	A7
TM1							
TM2							
TM3							
TM4							
TM5							
TM6							
TM7							
TM8							
TM9							
IM1							
IM2							
IM3							
IM4							
IM5							
IM6							
IM7							
IM8							
LC1							
LC2							
LC3							
LC4							

Table 10: Traceability Matrix Showing the Connections Between Assumptions and Other Items

[The traceability matrix is challenging to maintain manually. Please do your best. In the future tools (like Drasil) will make this much easier. —TPLT]

	TM??	TM??	TM??	GD??	GD??	DD??	DD??	DD??	DD??	IM??	IM??	IM??
TM??												
TM??			X									
TM??												
GD??												
GD??	X											
DD??				X								
DD??				X								
DD??												
DD??								X				
IM??					X	X	X				X	
IM??					X		X		X	X		
IM??		X										
IM??		X	X				X	X	X		X	

Table 8: Traceability Matrix Showing the Connections Between Items of Different Sections

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. Figure ?? shows the dependencies of theoretical models, general definitions, data definitions, instance models, likely changes, and assumptions on each other. Figure ?? shows the dependencies of instance models, requirements, and data constraints on each other.

9 Development Plan

N/A

10 Values of Auxiliary Constants

[Show the values of the symbolic parameters introduced in the report. —TPLT]

[The definition of the requirements will likely call for SYMBOLIC_CONSTANTS. Their values are defined in this section for easy maintenance. —TPLT]

[The value of FRACTION, for the Maintainability NFR would be given here. —TPLT]

	TM1	TM2	TM3	TM4	TM5	TM6	TM7	TM8	TM9	IM1	IM2	IM3	IM4	IM5
TM1														
TM2														
TM3														
TM4														
TM5														
TM6														
TM7														
TM8														
TM9														
IM1														
IM2														
IM3														
IM4														
IM5														
IM6														
IM7														
IM8														

Table 11: Traceability Matrix Showing the Connections Between Items of Different Sections

References

	IM??	IM??	IM??	IM??	4.2.10	R??	R??
IM??		X				X	X
IM??	X			X		X	X
IM??						X	X
IM??		X				X	X
R??							
R??						X	
R??					X		
R??	X	X				X	X
R??	X						
R??		X					
R??			X				
R??				X			
R??			X	X			
R??		X					
R??		X					

Table 12: Traceability Matrix Showing the Connections Between Requirements and Instance Models

[The following is not part of the template, just some things to consider when filing in the template. —TPLT]

[Grammar, flow and L^AT_EX advice:

- For Mac users *.DS_Store should be in .gitignore
- L^AT_EX and formatting rules
 - Variables are italic, everything else not, includes subscripts (link to document)
 - * [Conventions](#)
 - * Watch out for implied multiplication
 - Use BibTeX
 - Use cross-referencing
- Grammar and writing rules
 - Acronyms expanded on first usage (not just in table of acronyms)
 - “In order to” should be “to”

	IM??	IM??	IM??	IM??	4.2.10	R??	R??
IM??		X				X	X
IM??	X			X		X	X
IM??						X	X
IM??		X				X	X
R??							
R??						X	
R??					X		
R??	X	X				X	X
R??	X						
R??		X					
R??			X				
R??				X			
R??			X	X			
R??		X					
R??		X					

Table 13: Traceability Matrix Showing the Connections Between Requirements and Instance Models

—TPLT]

[Advice on using the template:

- Difference between physical and software constraints
- Properties of a correct solution means *additional* properties, not a restating of the requirements (may be “not applicable” for your problem). If you have a table of output constraints, then these are properties of a correct solution.
- Assumptions have to be invoked somewhere
- “Referenced by” implies that there is an explicit reference
- Think of traceability matrix, list of assumption invocations and list of reference by fields as automatically generatable
- If you say the format of the output (plot, table etc), then your requirement could be more abstract

—TPLT]

Appendix — Reflection

[Not required for CAS 741 —SS]

The information in this section will be used to evaluate the team members on the graduate attribute of Lifelong Learning.

The purpose of reflection questions is to give you a chance to assess your own learning and that of your group as a whole, and to find ways to improve in the future. Reflection is an important part of the learning process. Reflection is also an essential component of a successful software development process.

Reflections are most interesting and useful when they're honest, even if the stories they tell are imperfect. You will be marked based on your depth of thought and analysis, and not based on the content of the reflections themselves. Thus, for full marks we encourage you to answer openly and honestly and to avoid simply writing "what you think the evaluator wants to hear."

Please answer the following questions. Some questions can be answered on the team level, but where appropriate, each team member should write their own response:

1. What went well while writing this deliverable?

We were successful in splitting up the work that could be done individually and then coming together to discuss and combine our work. We also did a good job in playing to each team members strengths and dividing the work accordingly. For example, we collaboratively worked on deriving our instance models by hand which required us as a team to also discuss our theoretical models, general definitions, data definitions and assumptions. This was crucial that each team member was on the same page at this stage as everything in this SRS document was centered around these concepts.

Once these concepts were discussed we were then able to split up writing up these instance models, theoretical models, general definitions, data definitions and assumptions. We also split up the remaining sections such as the functional/non-functional requirements, problem description, and general description sections. By splitting the work up this way it allowed our team to be time effective but also ensured that each team member had a solid grasp on the key concepts the SRS was centered around.

2. What pain points did you experience during this deliverable, and how did you resolve them?

The main pain points of this deliverable were trying to understand the mathematical models and physics behind the CVT system. We resolved this by meeting together as a team and having a group discussion on the topic. We went through the math and derived the equations together, which helped us understand the material better. We also reached out to our clients for clarifications and they provided us with help and resources to better understand the topic. As a result our team was more confident in deriving the mathematical models required for this project.

3. How many of your requirements were inspired by speaking to your client(s) or their proxies (e.g. your peers, stakeholders, potential users)?

One key requirement from our clients is the Verifiability requirement. During our first capstone meeting with Dr. Smith he stressed the importance of verifying our results. This resulted in us adding the requirement to our document. Additionally, the majority if not all of our functional requirements were inspired by the needs of the McMaster Baja team who we meet with regularly to gather feedback and insights.

4. Which of the courses you have taken, or are currently taking, will help your team to be successful with your capstone project.

Many of the courses we have taken so far will prove helpful in completing our capstone project. SFWRENG 4HC3, will help us with the design of the GUI. SFWRENG 2AA4 and SFWRENG 3A04 will help us architect our code base. PHYSICS 1D03, SFWRENG 3MX3 and SFWRENG 3DX34, will help us understand and model the physics and math behind the CVT system. SFWRENG 3XB3 and SFWRENG 4X03, will help with creating efficient and reliable implementations. SFWRENG 3S03, will help us verify and validate our software.

5. What knowledge and skills will the team collectively need to acquire to successfully complete this capstone project? Examples of possible knowledge to acquire include domain specific knowledge from the domain of your application, or software engineering knowledge, mechatronics knowledge or computer science knowledge. Skills may be related to technology, or writing, or presentation, or team management, etc. You should look to identify at least one item for each team member.

To successfully complete this capstone project our team will need to acquire several important skills. We will need to learn and understand Unity to create a user-friendly interface which will use our created mathematical models in Python. For Unity we need to learn how to make a GUI interface, animate 3D models, create and display graphs, and learn how to interface with a Python script to run the mathematical models. For Python we need to learn how to create and implement these complex mathematical models and learn how to make these Python models interface with Unity. It is important that each team member acquires a general knowledge of these skills as team members completing the Python code must be familiar with interfacing the code with Unity. As well as team members completing the models within Unity must understand the mathematical Python models to use Unity to simulate them.

6. For each of the knowledge areas and skills identified in the previous question, what are at least two approaches to acquiring the knowledge or mastering the skill? Of the identified approaches, which will each team member pursue, and why did they make this choice?

To familiarize ourselves with Unity projects, we will watch youtube tutorials and read guides provided by Unity. The team members who chose to do the videos did so because they are more visual learners and prefer to see the process in action. Those that chose to read the guides did so because they prefer to read and also prefer that it is from an official source. For learning the Python required, we will do research on relevant libraries and functions that will help us implement our mathematical models. The team members who chose to do research did so because they are more comfortable with researching and learning on their own.

This approach allows us to take into account the different learning styles of each team member so that everyone can learn in a way that suits them best. Additionally, this also allows our team to have a larger variety of resources and variation leading to a stronger final project.