Team No. 190

Problem B

DESIGN A ROLLER COASTER

Abstract

In this problem, we are required to design a safe and exciting roller coaster. To solve the problem, we set up a design model of the trajectory with Solid-works and Planet Coaster. Then we derive the equations of motion based on basic dynamic laws. Next with Euler's method, we estimate the solutions and simulate of the roller coaster's motion using MATLAB. For safety and excitement judgement, we define several parameters related to position, velocity, and acceleration to represent the degree of safety and excitement. The value of these parameters and their simulation with time are also obtained with MATLAB. Finally we come to the conclusion that the designed roller coaster is safe and exciting.

Contents

1 Introduction

Roller coaster is one of the most exciting recreation facilities in amusement park. In this article, we are going to design a roller coaster which is safe and exciting. We will first give a concept diagram of our roller coaster's whole trajectory, then divide it into five parts and analyse them respectively. We are going to use the basic kinematics laws to construct second order ODEs for the motion of the roller coaster car, and apply the Euler Method to obtain an approximate solution for the ODEs, from which we could obtain the velocity and acceleration at any instant of time.

In Model Section, we will first give our method of judgement of safety and excitement, then introduce the basic laws and methods we will use in this project. Moreover, for each part of our trajectory, we will give a simple model with parameters.

In Result Section, we will first give the overall results of our whole model, and give the proof of the safety and excitement. Then we will explain in details about how we obtain the results and the motion of the car in each part of the trajectory.

Finally we will draw a conclusion and discuss the limitations and advantages of our model and give some suggestions about how to improve the model.

2 Model

2.1 Problem Overview

In this article, we are going to design a roller coaster that is *safe* and *exciting*. We will design the trajectory of the roller coaster and clearly give the visualization of our trajectory first. Based on the trajectory and the initial conditions, we derive motions of equations and then use the Euler Method to find the distance, velocity, acceleration with respect to t. Using the result of acceleration, velocity, and height, we then find the value of Safety and Excitement referring to our definition below, which prove that our goal of safety and excitement is reached.

2.2 Definition

2.2.1 Parameter Definition

First we assume that passengers along with the car are considered as particles. Then we define the G-axes with three types of accelerations for human body as Figure 2.1 shows[1]:

- Forward acceleration a_x along G-x axes in the direction of the velocity,
- Lateral acceleration a_y along G-y axes in the direction of the radius of curvature,
- Upward acceleration a_z along G-z axes in the direction of the normal vector.

Based on the above definition, the related parameters representing the value of accelerations are defined as

$$G_i = a_i/g \tag{1}$$

for i=x,y,z, where $g = 9.78m/s^2$ is the gravity near earth's surface.

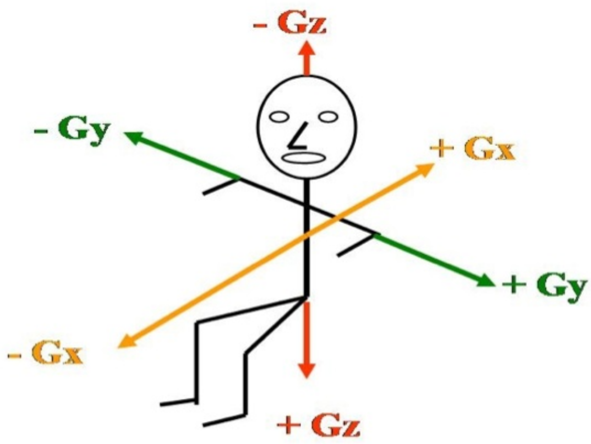


Figure 2.1: The definition of G-axes[4].

2.2.2 Safety Judgement Definition

1. According to Kumar and Norfleet's study[2], the influence of acceleration differs with different types of accelerations. For G_z, it is shown that if G_z=4.8 g, safety is satisfied when the maintaining time is less than 2.5 s. For G_y=5.5 g, safety is satisfied when the maintaining time is less than 2.5 s. For G_x, if G_x=6.0 g with passengers sitting straight 90 degree, safety is satisfied when the maintain time is less than 30 s.

   To simplify the model, in this project, it is assumed that if:

$$G_z < 4.8g, \tag{2}$$

$$G_y < 5.5g, \tag{3}$$

$$G_x < 6.0g, \tag{4}$$

   with the maintaining time < 2.5 s, the safety requirement is satisfied.

   Since in this project, there's no other forward accelerating force except gravity, so

$$G_x \leq g + \frac{a_{friction}}{g} < 5g$$

   is always satisfied since friction is relatively small with respect to gravity. In this article, we will calculate the concrete value of G_z and G_y to indicate they are under a safe range.

2. Moreover, we should consider the weight capacity of the track. In order to ensure safety, we should make sure that the force applied on the track should not larger than certain value. According to Newton's Laws, we have

$$F = Ma_z = MG_zg$$

Since the weight capacity of different materials varies a lot. Typically most of the materials could stand weight larger than 20000N. If we choose $M = 1000kg$, the corresponding G_z that create such weight should be 20, which is much larger than the largest G_Z that human body could bear. Hence when the acceleration is less than the limit of human endurance, both the safety of human body and the track structure could be ensured.

3. Finally, we should ensure that at every point of the trajectory, the speed of the car is larger than 0m/s.

2.2.3 Excitement judgement definition

It is the acceleration of the human body that can be sensed and converts to excitement. Based on the definition of the three types of acceleration, it is noticed that they have different effect on excitement.
When the values G_x, G_y, G_z are the same, the lateral acceleration a_x representing the orientation excites human bodies most, followed by the forward acceleration, while the upward acceleration has the least effect. Therefore, a parameter $S_{excitement}$ is defined to quantify the degree of excitement:

$$S = k_1(4G_x + 5G_y + 3G_z) \tag{5}$$

Moreover, we notice that the degree of excitement also has some relation to the speed of human body. Generally speaking, when human body is moving at a higher speed, the human would feel much more sense of excitement. Hence another item relate to speed should be added.

$$S = k_1(4G_x + 5G_y + 3G_z + k_2 v) \tag{6}$$

Finally, when human body is in a high altitude, they will feel much more exciting. Hence we add the third item relate to altitude into this equation

$$S = k_1(4G_x + 5G_y + 3G_z + k_2 v + k_3 H) \tag{7}$$

Among all the factors, the acceleration influence the degree of the excitement most, while the speed influence the least. Since when passengers are moving at a high speed, they could hardly feel how fast they are. Consequently, finally we choose the coefficients $k_1 = 1$, $k_2 = 0.05$, $k_3 = 0.1$. The unit for H and v is in SI.

$$S = 4G_x + 5G_y + 3G_z + 0.05v + 0.1H \tag{8}$$

The larger S represents higher degree of excitement. When passenger are resting on the reference point, S equals 0. And for a car being suddenly slammed, its acceleration is around $G_x = 30m/s^2 \approx 3g$ [5] and $S = 12$ which already has relatively strong excitement on human body.
Therefore we assume the excitement requirement is satisfied when:

$$S > 12 \tag{9}$$

2.3 Assumptions and Laws

2.3.1 Natural axes –The G-axes

In this article, the main focus is the passengers' acceleration on the roller coaster. As introduced in section 2.2.1, our start point is to set up a natural axes referring to passengers' motion, which is defined as the G-axes. Our central point of solving this problem is to calculate and analysing the value of velocity and G_i along each axes.

2.3.2 Euler's Method[3]

To calculate and analysing the distance, velocity and acceleration, we need to solve many complex second-order ordinary differential equations. Mathematical methods might not working so well when solving the complex ODEs. Euler's method is a numerical method helping find the unique solution of the second-order ordinary differential equation (ODE) in our project:

$$\frac{d^2r}{dt^2} = \frac{F(v,r,t)}{m}$$

through the initial conditions v(0)=v_0, r(0)=r_0, we will find the position and velocity of our particle at any instant of time t.

Newton's equation of motion can be rewritten as a pair of coupled first-order ODEs, like this:

$$\frac{dv}{dt} = \frac{F(v(t),r(t),t)}{m}$$

$$rt = v(t)$$

write $v(t) = f(t) = rt$, and further we transform

$$\frac{dv}{dt} = \frac{df}{dt} = G(f(t),t)$$

with the initial condition is given as f(0)=f_0.

Consider the Taylor expansion of a function f at time (t+Δt) and ignore the terms with order of Δt higher than 1, we can get this formula:

$$f(t+\Delta t) \approx f(t) + \frac{f}{t}\Delta t = f(t) + G(f(t),t)\Delta t$$

Using this formula, the approximate value f^*(t+Δt) can be found if the value of f (t) is known. That means, through Euler's method, we start at t_0=0 where f(0)=f_0 is known exactly, using the slope k of the curve at the left end of the interval (t_0, t_1), to get the approximate value f^* of the other end, as shown in Figure 2.2 below

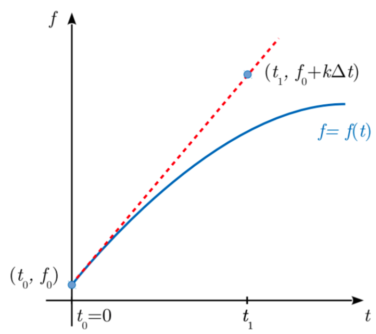


Figure 2.2: Figure of Euler method

In subsequent steps, the method is used continuously to obtain an approximation of the function f. Since the computer is a discrete machine, we need a discrete time step Δt to represent the discrete set of points at the time t_i= iΔt, and then the algorithm is as follows

$$f_0^* = f_0$$
$$k = G(f^*(t_i), t_i)$$
$$f^*(t_{i+1}) = f^*(t_i) + k\Delta t$$

Given the initial value f(0)=f_0, the equation can help us to get the approximate values in the required time interval. that is to numerically solve the first-order ODE with that initial condition.

2.3.3 Law of Conservation of Energy

In this article, we assume that all the motions are under the category of classical mechanics. Hence we could use the following basic laws from kinematics and dynamics.

The energy is conserved during the whole process of motion. The total amount of energy is the gravitational potential energy the roller coaster hold first. We assume that work against resistance is the only dissipation of energy during the process. Hence the equation for conservation of energy could be written as

$$E_{p0} = E_{p(t)} + E_k + W_f \tag{10}$$

2.3.4 Newton's Laws of Motion

The most basic and important laws we will used in this article is Newton's Laws of Motion. In this article, we could consider problems under classical mechanics, where the mass m could be regarded as a constant. Hence we could write the Newton's equation as follow:

$$\sum \vec{F} = \frac{d(m\vec{v})}{dt} = m\frac{d\vec{v}}{dt} = m\frac{d^2\vec{r}}{dt^2} \tag{11}$$

$$\vec{F_{12}} = -\vec{F_{21}} \tag{12}$$

2.3.5 Motion with Resistance

In this article, we consider two types of resistance. The first is the friction between the roller coaster car and the track, which we note it as f_1

$$\vec{f_1} = -\mu N(\frac{\vec{v}}{v}) \tag{13}$$

where μ is the friction coefficient, which is usually between 0 and 1, N is the force that car applied perpendicularly on the track.

The second is the *quadratic drag* between the roller coaster car and the air. Since the car is enormous with large mass and high speed, we consider the drag to be quadratic instead of linear, which we note it as f_2

$$\vec{f_2} = -\beta v^2(\frac{\vec{v}}{v}) \tag{14}$$

with the coefficient β

$$\beta = \frac{1}{2}\rho C_d A \tag{15}$$

where ρ is the fluid density, C_d is the drag coefficient, which is usually between 0.25 and 0.5 for cars, A is the cross-sectional area perpendicular to the direction of motion.

Consequently, the total resistance equals the sum of f_1 and f_2.

2.3.6 Dynamics of Circular Motion

Roller coaster usually follows a trajectory that has a circular shape. When choosing the roller coaster car as the frame of reference, we should consider the *centrifugal 'force'*, F_c

$$F_c = -m\vec{\omega} \times (\vec{\omega} \times \vec{r}) \tag{16}$$

2.4 Basic Model

The overall concept diagram of our roller coaster is shown below in Figure 4.1 and 2.4. To make it easier to analyse the equations of motion, we divided the whole trajectory into five parts and analyze them respectively.

Figure 2.3: Concept Diagram of Roller Coaster

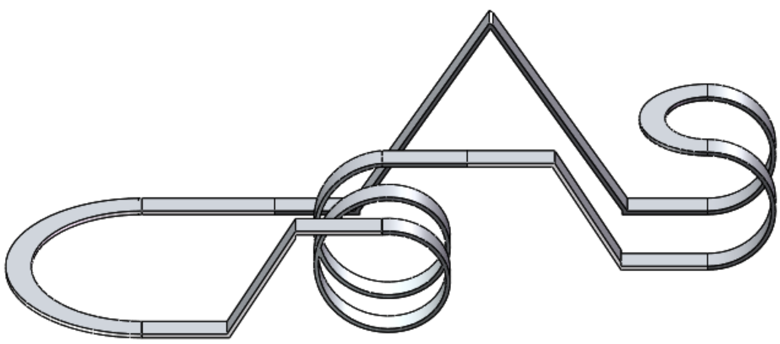
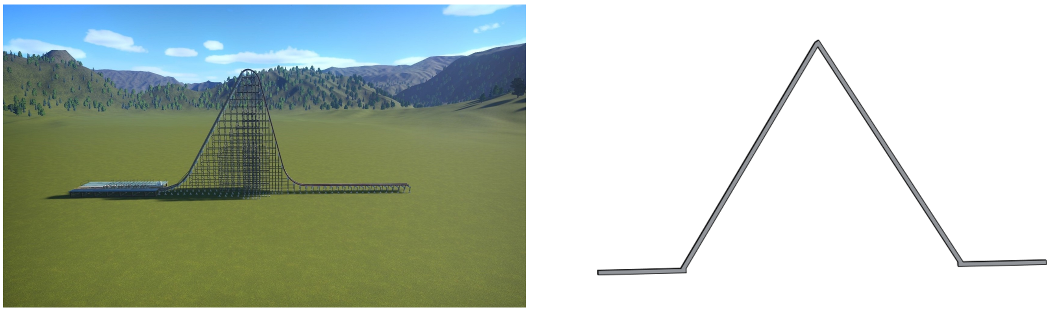


Figure 2.4: Solidworks diagram of Roller Coaster

Using solid works, we could simplify the concept diagram of our roller coaster, as shown above in Figure 2.4. In order to analyze the motion of the whole trajectory, we divided the track into five parts, as shown below in Figure 2.5

2.4.1 The First Cant Track



(a) The First Cant Track

As shown in Figure (a), the first cant track are formed of four straight line parts:

- 20-meter long horizontal track
- 30-meter height rising slope with 75^o inclining angle
- 30-meter height falling slope with 75^o inclining angle
- 20-meter long horizontal track

2.4.2 The Tri-semicircle Circle Track

(b) The Tri-semicircle Circle Track

As shown in Figure (b), the tri-semicircle track are formed of three half-circle parts:

- 16-meter height up-circle with 8-meter radius
- 16-meter height mid-circle with 8-meter radius
- 16-meter height down-circle with 8-meter radius

2.4.3 The Upward Cant Track

(c) The Upward Cant Track

As shown in Figure (c), the upward cant track are formed of three straight line parts:

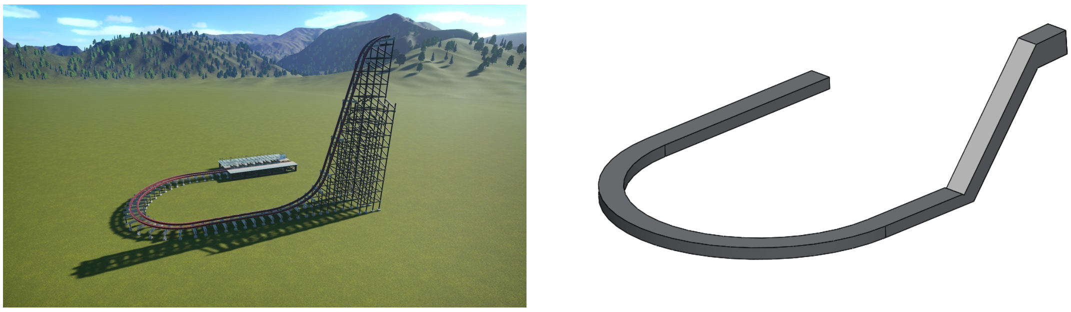
- 20-meter long horizontal track
- 22-meter height rising slope with 60^o inclining angle
- 10-meter long horizontal track

2.4.4 The Double Helix Track

(d) The Double Helix Track

As shown in Figure (d), the double helix track looks like a flat, twice-wound spring. The radius of the circle is 8 meters, with a screw pitch of 2π meters each.

2.4.5 The Downward Cant and Circular Track



(e) The Downward Cant and Circular Track

As shown in Figure (e), The downward cant and circular track are formed of four parts:

- 16-meter height falling slope with 75^o inclining angle
- 26-meter long horizontal track
- horizontal half-circle track with $(8+2\pi)$-meter radius
- 20-meter long horizontal track leading it back to the origin

3 Results

3.1 The Whole Model

According to our calculation, we obtain the detailed values of distance, speed, and time in each part of the model. The model is in Figure 3.1 and the data is recorded in Table 1. The direction of the roller coaster is always tangential to the trajectory. Based on the calculated values, we find the total length of the trajectory L and the duration T

$$L = 394.7507 + 33.3597 = 428.1104[m]$$
$$T = 32.3094 + 16.6798 = 48.9892[s]$$

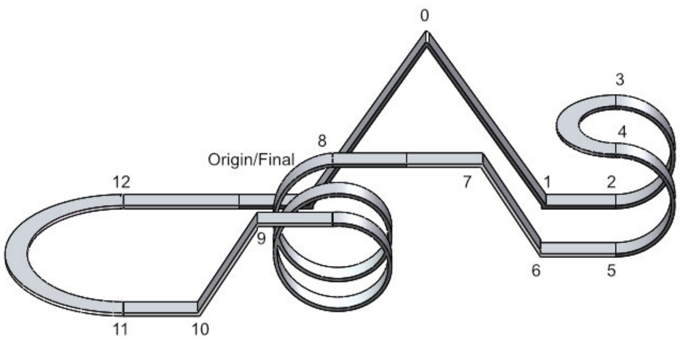


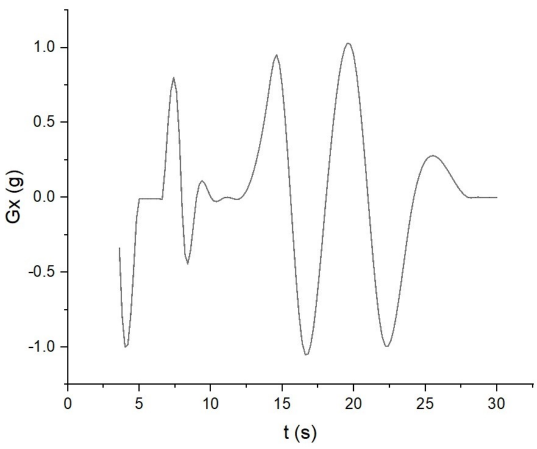
Figure 3.1: Figure of Each Point's Position

	s[m]	v[m/s]	t[s]
Origin	-33.3597	2.0000	-16.6798
0	0	0	0
1	31.0583	23.5084	2.6231
2	51.0583	22.9152	3.4938
3	76.1910	14.2516	4.8664
4	101.3238	14.0754	6.6409
5	126.4565	22.4271	8.0352
6	146.4565	21.8213	8.9393
7	171.8599	5.3568	10.8123
8	181.8599	4.3289	12.8775
9	283.1732	1.5490	24.4635
10	303.8787	20.2203	26.5508
11	329.8787	19.3877	27.8638
12	374.7507	18.9860	30.2026
Final	394.7507	0	32.3094

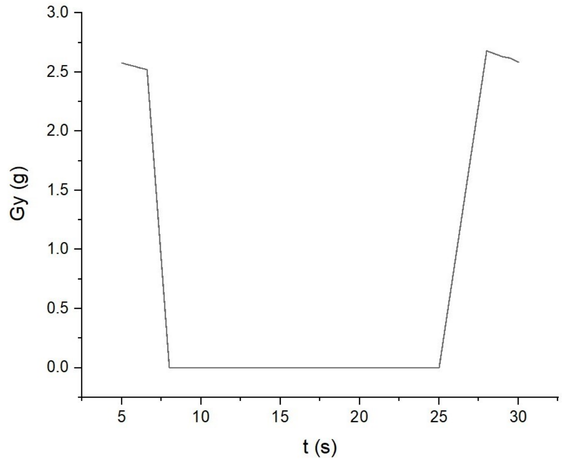
[1,2]

Table 1: The Whole Trajectory Values

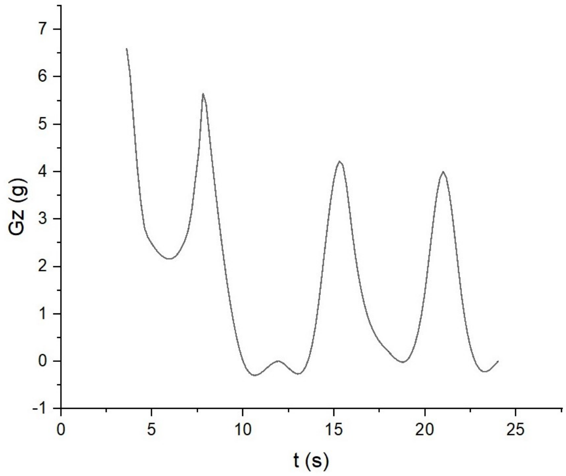
The followings are the graphs of G_x, G_y, G_z, S depending on time t for the whole process. These four graphs of G_i and S are generalized from plots in MATLAB for each five parts as we have divided into. All the single plots and codes in MATLAB is attached in the Appendix



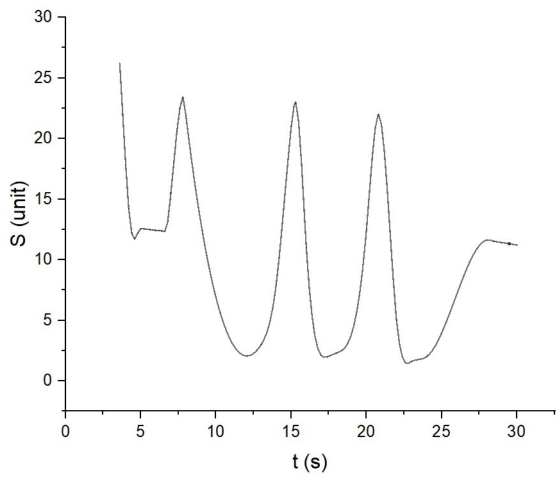
(a) Gx vs t in the whole time interval



(b) Gy vs t in the whole time interval



(c) Gz vs t in the whole time interval



(d) S vs t in the whole time interval

Figure 3.2: Relation with Respect to t in the Whole Time Interval

1. **The Safety Judgement**

   According to Figure 3.2(a) and Figure 3.2(b), we notice the maximal G_x is 1, while the maximal G_y is 2.5, which are below the safe range according to Equation (3) and (4).

   As for G_z, the maximal value is $6.5 > 4.8$ which is the limit in Equation (2). But value of G_z larger than 4.8 only lasting for 1 second, much smaller than the maximal lasting time 2.5s, hence it is also safe according to NASA's study[2].

   From our calculation, the safety of our design is confirmed. And we find that the most significant

[1] When the car is at the highest point with altitude equals 30 meters, the travelling distance and the time is 0.

[2] All the codes needed to obtain the results in MATLAB is attached in the appendix.

limiting factor of safety is the upward acceleration, while the forward acceleration and the lateral acceleration can hardly exceed the limitation.

2. **The Excitement Judgement**

   The degree of excitement S with respect to time interval t is shown in Figure 3.2(d). When $S \geq 12$ we regard that the amusement is very exciting. We find that the largest S is even larger than 20.

   Also we noticed many sudden change of S. Passengers would experience a period of high level of S while suddenly rise into a very high level, which is really exciting.

   Consequently, the excitement requirement of our roller coaster could be regarded as satisfied.

3.2 Detailed Model Analysis

Before analyze the model in details, we define the following

- The gravitational acceleration: $g = 9.78[m/s^2]$
- The mass of the roller coaster: $M = 1000[kg]$
- The friction coefficient: $\mu = 0.05$ for strait track, $\mu = 0.001$ for circular track
- The damping coefficient: $\beta = \dfrac{1}{2}\rho C_d A = \dfrac{1}{2} \times 1.293 \times 0.25 \times 10 = 0.37[kg/m]$
- We will choose the 20 meter horizontal segment as the potential reference.

3.2.1 The First Cant Track

1. **The Start Point**

   The roller coaster is first taken to the highest point which is $H = 30[m]$ above the starting location. The slope has an angel of 64^o, with a length of 33.3597m. Assume the electromotor drive the roller coaster at a constant speed of 2m/s, then it takes the car 16.6798s to arrive at the top. Choose this time as the original time $t = 0s$. Then choosing the starting location as the potential reference point, the roller coaster would possess a high potential energy, and we note it as E_{p0}.

$$E_{p0} = MgH = 293400[J] \tag{17}$$

2. **The Downward Cant**

   Now we consider the downward cant track with the dip angle $\theta = 75^o$. We use the v_x, v_y, v_z as parametrization, which is the forward speed in the direction of G_x, G_y, G_z. According to Newton's law we obtain Using Equation (11) and (12), we could write the Newton equations

$$\begin{cases} M\dot{v_x} = F_x = f \\ M\dot{v_y} = F_y = 0 \\ M\dot{v_z} = F_z = 0 \end{cases} \tag{18}$$

   where

$$f = -\mu N - \beta v_x^2 = -\mu N - \beta v_x^2$$

$$N = Mgcos(\theta)$$

Using the above equations, we could get the ODE equation for v_x below:

$$Mgsin(\theta)cos(\theta) - (\mu N + \beta v_x^2)cos(\theta) = Mcos(\theta)\dot{v_x} \tag{19}$$

Insert the concrete value into this equation, we have

$$\dot{v_x} + 3.7 \times 10^{-4} v_x^2 - 9 = 0 \tag{20}$$

Using Matlab, we obtain the following data for this motion:

x[R]	v_x[m/s]	$\dot{v_x}$[m/s^2]	t[s]
0	0	0	0
4.9994	9.4770	8.9668	1.0543
9.9989	13.3905	8.9337	1.4915
19.9993	18.9029	8.8678	2.1107
31.0578	23.5084	8.7955	2.6231

3. **The Horizontal Track**

   Now we comes the first horizontal track. Assume the total length of this track is 20m. Using Equation (3.2.4), and apply $\theta = 0^o$, we obtain

$$\dot{v_x} + 3.7 \times 10^{-4} v_x^2 + 0.489 = 0 \tag{21}$$

   Using Matlab, we obtain the following data for this motion:

x[R]	v_x[m/s]	$\dot{v_x}$[m/s^2]	t[s]
0	0	0	0
4.9986	23.3605	-0.6919	2.8364
9.9981	23.2123	-0.6894	3.0511
19.9983	22.9152	-0.6843	3.4948

Using Matlab to apply Euler's method, we approximately solve the ODE Equation (19) and (20) and further get G_x, G_y, G_z as well as $S_{excitement}$ related to time t based on Equation (1) $\sim$ (4). The code and figures are attached in Appendix. [3]

3.2.2 The Tri-semicircle Track

1. **The Upward Circle Track**

   Then the upward half-circle track follows. Assume the angle for the car to move up is θ and the radius of the circle is R. We assume R is 8m. Then

$$\theta = \int_0^t \frac{v_x}{R} dt \tag{22}$$

[3] Using cant.m, horizontal.m, climb.m

Using Equation (11) and (12), we could write the Newton equations

$$\begin{cases} M\dot{v_x} = F_x = -Mgsin(\int_0^t \frac{v_x}{R}dt) - \mu\left[M\frac{v_x^2}{R} + Mgcos(\int_0^t \frac{v_x}{R}dt)\right] - \beta v_x^2 \\ M\dot{v_y} = F_y = 0 \\ M\dot{v_z} = F_z = \frac{Mv_x^2}{R} \end{cases} \tag{23}$$

Using the above equations, we could get the ODE equation for v_x below:

$$\dot{v_x} + \frac{\beta v_x^2}{M}\dot{v}^2 + gsin(\int_0^t \frac{v_x}{R}dt) + \mu\left[\frac{v_x^2}{R} + gcos(\int_0^t \frac{v_x}{R}dt)\right] = 0 \tag{24}$$

Using Matlab with Euler's law, with the estimation of θ as:

$$\theta = \int_0^t \frac{v_x}{R}dt = \Sigma\frac{v_x}{R}dt \tag{25}$$

together with

$$\begin{cases} G_x = \frac{\dot{v_x}}{g} \\ G_y = 0 \\ G_z = \frac{\dot{v_z}}{g} = \frac{v_x^2}{gR} \end{cases} \tag{26}$$

Using Matlab, we obtain the following data for this motion:

θ[R]	v_x[m/s]	$\dot{v_x}$[m/s^2]	an[m/s^2]	t[s]
0	22.9152	0	0	3.4938
$\pi/4$	21.8158	-7.1701	59.4912	3.7727
$\pi/2$	19.0369	-9.9794	45.3004	4.0792
π	14.2516	-0.0937	25.3096	4.8673

2. **The Middle Circle Track**

Then the horizontal half-circle track follows. Assume the radius of the circle is R. We assume R is 8m and the angle for the car to move forward is θ. We write the Newton's equations

$$\begin{cases} M\dot{v_x} = F_x = -\mu(M\frac{v_x^2}{R} + Mg) - \beta v_x^2 \\ M\dot{v_y} = F_y = \frac{Mv_x^2}{R} \\ M\dot{v_z} = F_z = 0 \end{cases} \tag{27}$$

Using the above equations, we could get the ODE equation for v below:

$$\dot{v_x} + \frac{\beta v_x^2}{M} + \mu(M\frac{v_x^2}{R} + Mg) = 0 \tag{28}$$

together with

$$\begin{cases} G_x = \dfrac{\dot{v_x}}{g} \\ G_y = \dfrac{\dot{v_y}}{g} = \dfrac{v_x^2}{gR} \\ G_z = 0 \end{cases} \tag{29}$$

Using Matlab, we obtain the following data for this motion:

θ[R]	v_x[m/s]	$\dot{v_x}$[m/s^2]	an[m/s^2]	t[s]
0	14.2516	0	0	4.8673
$\pi/4$	14.2074	-0.0999	25.2311	5.3079
$\pi/2$	14.1632	-0.0993	25.0747	5.7508
π	14.0754	-0.0981	24.7646	6.6409

3. **The Downward Circle Track**

Then the upward half-circle track follows. Assume the angle for the car to move up is θ and the radius of the circle is R. We assume R is 8m. Then

$$\theta = \int_0^t \frac{v_x}{R} dt \tag{30}$$

Using Equation (11) and (12), we could write the Newton's equations

$$\begin{cases} M\dot{v_x} = F_x = -Mgsin(\pi + \int_0^t \frac{v_x}{R}dt) - \mu \left[M\frac{v_x^2}{R} + Mgcos(\pi + \int_0^t \frac{v_x}{R}dt) \right] - \beta v_x^2 \\ M\dot{v_y} = F_y = 0 \\ M\dot{v_z} = F_z = \dfrac{Mv_x^2}{R} \end{cases} \tag{31}$$

Using the above equations, we could get the ODE equation for v below:

$$\dot{v_x} + \frac{\beta v_x^2}{M}\dot{v}^2 + gsin(\pi + \int_0^t \frac{v_x}{R}dt) + \mu \left[\frac{v_x^2}{R} + gcos(\pi + \int_0^t \frac{v_x}{R}dt) \right] = 0 \tag{32}$$

Using Matlab with Euler's law, with the estimation of θ as:

$$\theta = \int_0^t \frac{v_x}{R} dt = \Sigma \frac{v_x}{R} dt \tag{33}$$

together with

$$\begin{cases} G_x = \dfrac{\dot{v_x}}{g} \\ G_y = 0 \\ G_z = \dfrac{\dot{v_z}}{g} = \dfrac{v_x^2}{gR} \end{cases} \tag{34}$$

Using Matlab, we obtain the following data for this motion:

θ[R]	v_x[m/s]	$\dot{v_x}$[m/s^2]	an[m/s^2]	t[s]
0	14.0754	0	0	6.6409
$\pi/4$	15.5821	6.8140	30.3501	7.0718
$\pi/2$	18.7581	9.6258	43.9835	7.4396
π	22.4271	-0.2549	62.8718	8.0352

Using Matlab to apply Euler's method, we approximately solve the ODE Equation (24), (28) and (32). Then further get G_x, G_y, G_z as well as $S_{excitement}$ related to time t based on Equation (26), (29) and (34). The code and figures are attached in Appendix. [4]

3.2.3 The Upward Cant Track

1. **The Horizontal Track**

   The length of this track is 20 meters. The procedures are similar to what we have done in Section 3.2.1, hence we directly list the results here. Using Matlab, we obtain the following data for this motion:

x[R]	v_x[m/s]	$\dot{v_x}$[m/s^2]	t[s]
0	22.4271	0	8.0352
4.9978	22.2762	-0.6736	8.2588
9.9996	22.1247	-0.6711	8.4841
14.9980	21.9729	-0.6686	8.7108
19.9992	21.8213	-0.6662	8.9393

2. **The Upward Track**

   The height of the cant is 22m, with a dip angle θ equals 60^o. Similarly, we could obtain the equation of motion as follows

$$-Mgsin(\theta) - \mu Mgcos(\theta) - \beta v_x^2 = M\dot{v_x} \tag{35}$$

   Apply the constant quantities, we obtain the following equation

$$\dot{v_x} + 3.7 \times 10^{-4} v_x^2 + 8.714 = 0 \tag{36}$$

   Similarly to section 3.1.2, we obtain the ODE' s solution as well as G_x, G_y, G_z and $S_{excitement}$ related to time t. The terminal state is shown in the Table below. [5]

   Using Matlab, we obtain the following data for this motion:

[4] Using upcircle.m, midcircle.m, downcircle.m

[5] Using cant.m, horizontal.m, climb.m

x[R]	v_x[m/s]	$\dot{v_x}$[m/s^2]	t[s]
0	21.8213	0	8.9393
9.9907	17.2958	-8.8248	9.4503
19.9941	11.0990	-8.7597	10.1553
25.4010	5.3568	-8.7246	10.8123

3. **The Horizontal Track**

   After the upward track is a horizontal track of altitude equals 22m. Assume the total length of this track is 10m. Similarly, using Equation (28) again,

$$\dot{v_x} + 3.7 \times 10^{-4} v_x^2 + 0.489 = 0 \tag{37}$$

   Using Matlab, we obtain the following data for this motion:

x[R]	v_x[m/s]	$\dot{v_x}$[m/s^2]	t[s]
0	5.3568	0	10.8123
4.9998	4.8681	-0.4988	11.7903
9.9997	4.3289	-0.4969	12.8775

3.2.4 The Double Helix Track

After the horizontal track is the double helix track. Assume the angle for the car to move up is θ, then the car passes the whole double helix track when $\theta = 3\pi$. The radius of the helix is ρ, and the screw pitch is $2\pi k$. We assume ρ is 8m, k is 1m. Then

$$\theta = \int_0^t \frac{v_x}{\rho} dt \tag{38}$$

Using Equation (11) and (12), we could write the Newton equations

$$\begin{cases} M\dot{v_x} = F_x = -M\frac{\rho}{\sqrt{\rho^2+k^2}} gsin(\pi + \int_0^t \frac{v_x}{\rho}dt) - \mu\left[M\frac{v_x^2}{\rho} + M\frac{\rho}{\sqrt{\rho^2+k^2}} gcos(\pi + \int_0^t \frac{v_x}{\rho}dt)\right] - \beta v_x^2 \\ M\dot{v_y} = F_y = 0 \\ M\dot{v_z} = F_z = \frac{Mv_x^2}{\rho} \end{cases} \tag{39}$$

Using the above equations , with the estimation of θ as:

$$\theta = \int_0^t \frac{v_x}{R} dt = \Sigma \frac{v_x}{R} dt \tag{40}$$

we could get the ODE equation for v below:

$$\dot{v_x} + \frac{\beta v_x^2}{M}\dot{v}^2 + \frac{\rho}{\sqrt{\rho^2+k^2}} gsin(\pi + \int_0^t \frac{v_x}{\rho}dt) + \mu\left[\frac{v_x^2}{\rho} + \frac{\rho}{\sqrt{\rho^2+k^2}} gcos(\pi + \int_0^t \frac{v_x}{\rho}dt)\right] = 0 \tag{41}$$

together with

$$\begin{cases} G_x = \frac{\dot{v_x}}{g} \\ G_y = 0 \\ G_z = \frac{\dot{v_z}}{g} = \frac{v_x^2}{gR} \end{cases} \tag{42}$$

Using Matlab, we obtain the following data for this motion:

θ[R]	v_x[m/s]	$\dot{v_x}$[m/s^2]	an[m/s^2]	t[s]
0	4.3289	0	0	12.8775
$\pi/2$	13.0976	9.5249	21.4434	14.6215
π	17.9443	-0.1379	40.2495	15.3985
2π	3.2196	-0.0017	1.2958	18.1905
3π	17.7153	-0.1274	39.2288	20.9895
4π	1.5490	0.0063	0.2999	24.4635

Using Matlab to apply Euler's method, we approximately solve the ODE Equation (41). Then further get G_x, G_y, G_z as well as $S_{excitement}$ related to time t based on Equation (42). The code and figures are attached in Appendix. [6]

3.2.5 The Downward Cant and Circular Track

1. **The Downward Cant**

   The calculation for the downward cant is similar to the first downward track in 3.2.1, so we directly list results here in the table below, where x is the total displacement along G-x axis. [7]

x[m]	v_x[m/s]	$\dot{v_x}$[m/s^2]	t[s]
0	1.5490	8.9991	24.4625
4.9997	9.6027	8.9659	25.3597
9.9988	13.4790	8.9328	25.7928
14.9986	16.4573	8.8998	26.1268
19.9988	18.9651	8.8669	26.4091
22.7752	20.2203	8.8487	26.5508

2. **The Horizontal Track**

   Followed by the downward cant is a horizontal track whose length equals 26 meters. The calculation is similar to the middle circle track in 3.2.1, so we directly list our calculation results here in the table below, where x is the total displacement along G-x axis. [8]

[6] Using orbit.m
[7] Using cant.m
[8] Using horizontal.m

x[m]	v_x[m/s]	$\dot{v_x}$[m/s^2]	t[s]
0	20.2203	-0.6403	26.5508
5.0010	20.0616	-0.6379	26.7991
10.0004	19.9023	-0.6356	27.0493
15.0016	19.7422	-0.6332	27.3016
20.0016	19.5815	-0.6309	27.5559
26.0009	19.3877	-0.6281	27.8638

3. **The Circular Track**

   Followed by the horizontal track is a circular track with radius equals 8+2π meters. The calculation is similar to the middle circle track in 3.2.2, so we directly list our calculation results here in the table below, where θ is the total displacement of rotation angle. [9]

θ[R]	v_x[m/s]	$\dot{v_x}$[m/s^2]	an[m/s^2]	t[s]
0	19.3877	-0.6281	26.3165	27.8638
$\pi/4$	19.2866	-0.1735	26.0426	28.4439
$\pi/2$	19.1859	-0.1718	25.7715	29.0271
$3\pi/4$	19.0857	-0.1701	25.5030	29.6133
π	18.9860	-0.1684	25.2372	30.2026

4. **The Last Horizontal Segment**

   The last part of the track is also the original 20 meter horizontal segment. In order to slow down the high-speed roller coaster, we need to apply a large damping coefficient μ_0. Since this μ_0 is large enough, we could ignore the influence of quadric drag. Hence this motion could be regarded as a uniformly retarded motion. We have the following equation

$$v^2 = 2a(x - x_0) \tag{43}$$

$$t = t_0 + \frac{v}{a} \tag{44}$$

   Hence we calculate the final state of motion as [10]

s[m]	v[m/s]	a[m/s^2]	t[s]
394.7507	0	9.3971	32.3094

4 Discussion

4.1 Conclusion

During the whole process, we draw a concept model for the ideal roller coaster and then simplify the complex model using the assumptions that roller coaster could be regarded as a particle. Also, we ignore

[9] Using horicircle.m
[10] Using horizontal.m

the rotation of tracks and the roller coaster to simplify the calculation.

Using Euler's Method, we could solve the complex second order ODEs of motion, then Matlab is applied to find the approximate solution of the equation. Calculate the five models respectively, we finally obtain the total duration of the ride equals 48.9892[s], covering a track lengthen out to 428.1104[m].

Before our calculation, we set the mass of roller coaster M, the friction coefficient μ, and the damping coefficient of air drag. Furthermore, the shape and length of the track is determined first, such as the height and length of the track, the radius and screw pitch of the circle. We could obtain some different results by changing the value of this constants. In this article, we choose the parameters that can reach to the most safe and exciting roller coaster.

During the calculation, we find that the friction coefficient would lead to a huge dissipation of energy due to the large centrifugal 'force', which make it difficult for the car passing through the helix. Hence we replace the origin μ with a smaller one for circular track.

4.2 Limitations and Possible Improvement

4.2.1 Shape of the Car Consideration

In this article, we consider the car as a mass point without length and volume. However, actually it is not the case. If we want to examine the rotation of the roller coaster, we should take into consideration of its length and volume. However, in this article we only need to find out its Newton's equations, hence our model could work well.

To make our model much better, we could consider the shape of the roller coaster car as a long cuboid. Take into consideration of the rotation of the car and the reverse of the track. To do so, we could try to consider the normal vector of the track surface.

Planet Coaster[4] can be applied to formulate the structure of the roller coaster, and calculate the three kinds of accelerations and sense of excitement respectively. We build the concept diagram of the roller coaster here, and it could also be used to test our model.

4.2.2 Self-rotation Consideration

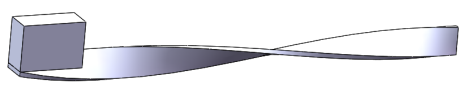


Figure 4.1: Graph of the self-rotating track

In the above model, we regard the track as a line, and the car as a particle. However,in actual world, the track of the roller coaster would have a width of three to five meters. So the car can be self-rotating along the trajectory, as Figure 4,1 shows.

One of our method to estimate the self-rotation effect is:

Let the initial angular momentum of the roller coaster be L, the projected radius of the track in the direction parallel to the track is R, and the rising height of the track is H.

Since the friction and air resistance is ignored, we assume the conservation of angular momentum is satisfied, thus we can simply calculate the angular velocity for the roller coaster to rotate along the axis is

$$\omega = \frac{4L^2}{md^2}$$

so the centripetal acceleration is

$$a_\omega = \omega^2 R$$

and the rising velocity along the axis as

$$v_{along} = \frac{2LH}{\pi md^2}$$

.

Then with the same method in 2.2.3, we define

$$G_\omega = \frac{a_\omega}{g}$$

We add a correction term to $S_{excitement}$ in Equation (7)

$$\Delta_w = k_1 \cdot k_4 G_\omega + k1 \cdot k_5 v_{along} \tag{45}$$

Then the corrected $S_{excitement}(new)$ is written as:

$$S_(new) = S + \Delta_w = S = k_1(4G_x + 5G_y + 3G_z + k_2 v + k_3 H + k_4 G_\omega + k_5 v_{along}) \tag{46}$$

4.2.3 Other Considerations

We could consider the air drag more precisely and apply a much more precise and convincing method to define safety and exciting.

4.3 Advantages

Firstly, our model is easy to formulate and calculate. We utilize some basic functions to represent the complex trajectory of the track. We also focus on analysing the state of motion of the roller coaster and write equations of motion.

Moreover, we build our model under many reasonable assumptions. We take into consideration of many important factors. Since the roller coaster run in a high speed, the factor of the quadratic drag should be considered. We assume the roller coaster to be a particle since we do not need to analyse the rotation of it. We also assume the friction between the car and the track is small, which could make the car to run much further. We set some reasonable initial conditions to increase the sense of excitement and

Thirdly, the model of the roller coaster track is very humanized. There are no sharp turns in the track of the roller coaster, which can avoid the huge acceleration caused by simplification. At the same time, it can make the roller coaster running more smoothly, and passengers can have a better experience.

Last but not least, we make a clear definition of the safety and excitement, which make it easier for us to analyse and fulfil our goal of constructing a roller coaster that is safe and exciting.

Above all, we tried our best to construct an analysable model of roller coaster and analyse its concrete motion in the whole process. We have searched many information and material to support our assumption, and apply Euler Method to find the approximate solution of velocity and the three kinds of accelerations at any instant time. Although the process is difficult, but we finally succeed to construct model and obtain all the results.

References

[1] Av Med, *G Force – What is?* [Online]. Available FTP: http://www.avmed.in/2012/06/g-force-what-is/

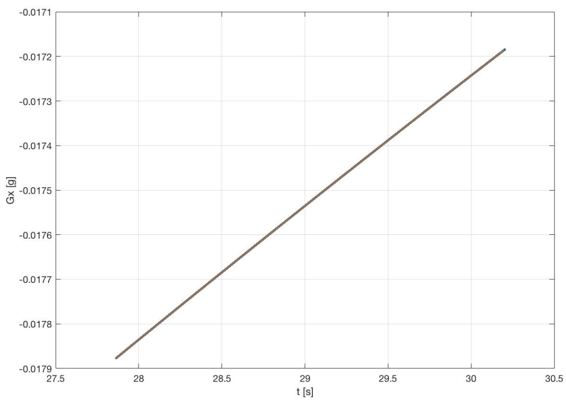
[2] K. Vasantha Kumar, William T. Norfleet, *Issues on Human Acceleration Tolerance After Long-Duration Space Flights*, NASA Technical Memorandum (1992), pages 2-9, 29-38

[3] Paul Dawkins, *Paul's Online Notes*, section 2-9 [Online].
Available FTP: http://tutorial.math.lamar.edu/Classes/DE/EulersMethod.aspx

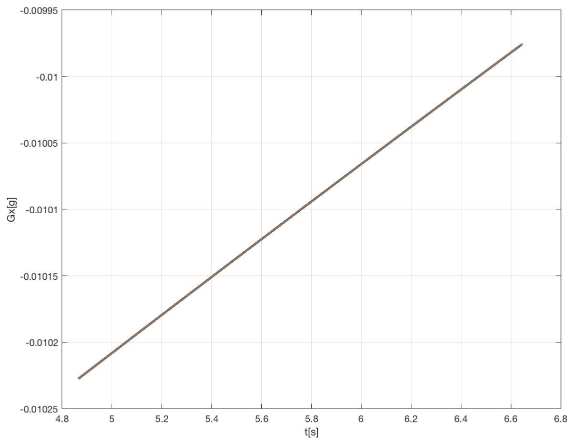
[4] Jody Macgregor, *PLANET COASTER REVIEW* (2016) [Online].
Available FTP: https://www.pcgamer.com/planet-coaster-review-1/

[5] Donggun Park, Sunghwan Park, *A comparative study on subjective feeling of engine acceleration sound by automobile types*, International Journal of Industrial Ergonomics (2019)

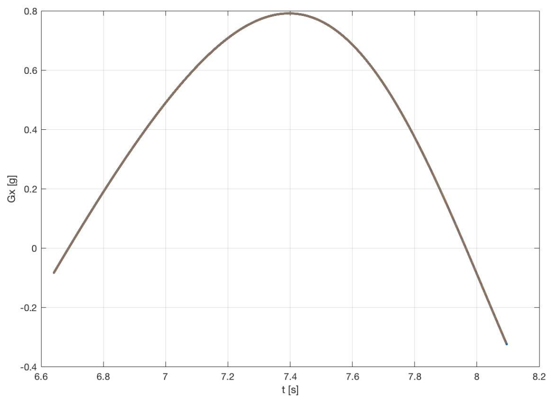
A Plotting Figures



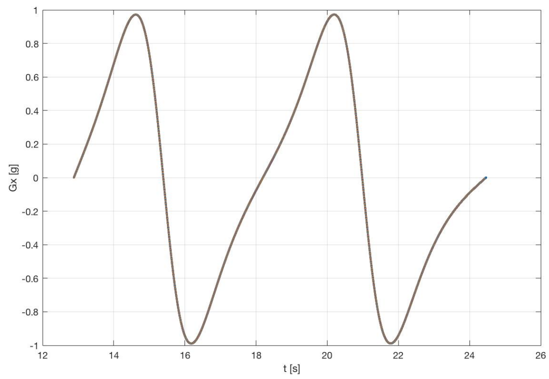
(a) Gx_horicircle



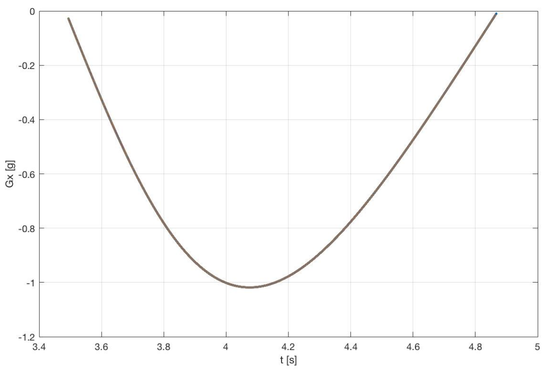
(b) Gx_midcircle



(c) Gx_downcircle

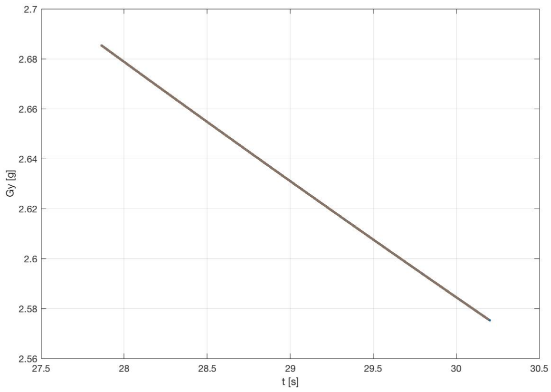


(d) Gx_orbit

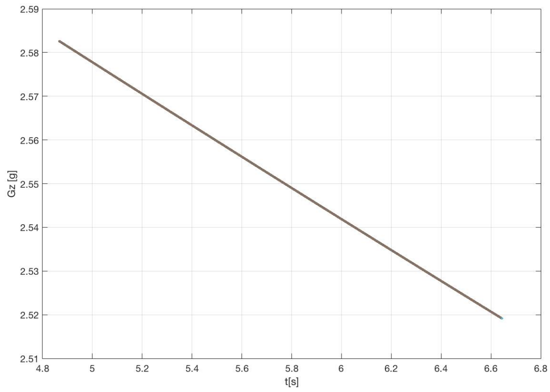


(e) Gx_upcicle

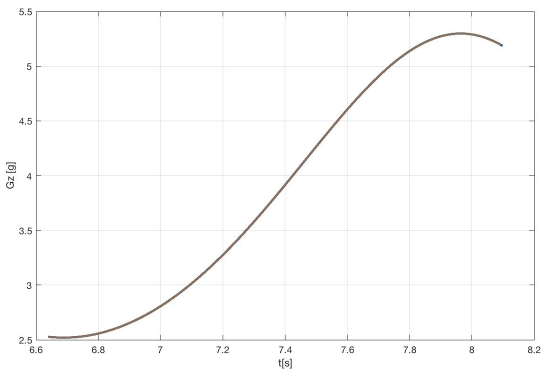
Figure A.1: G_x for the five circular track



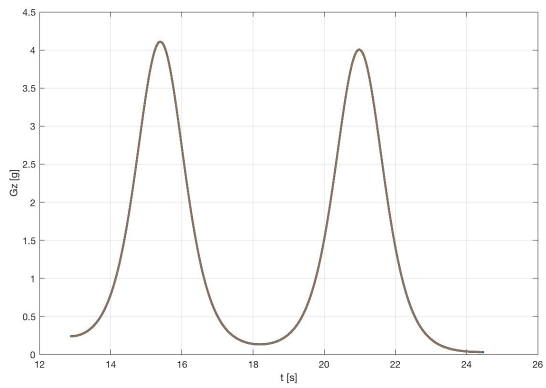
(a) Gy_horicircle



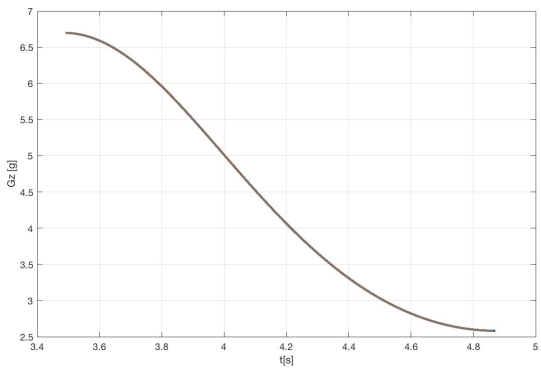
(b) Gy_midcircle



(c) Gz_downcircle

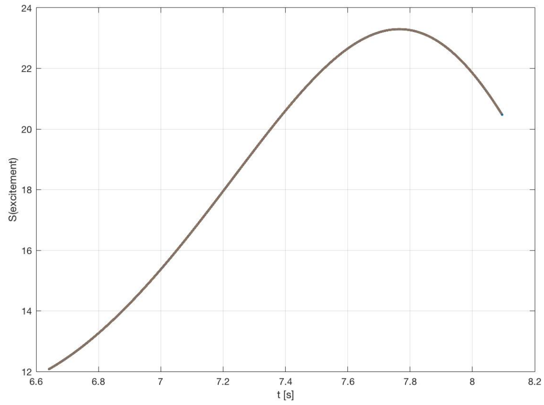


(d) Gz_orbit

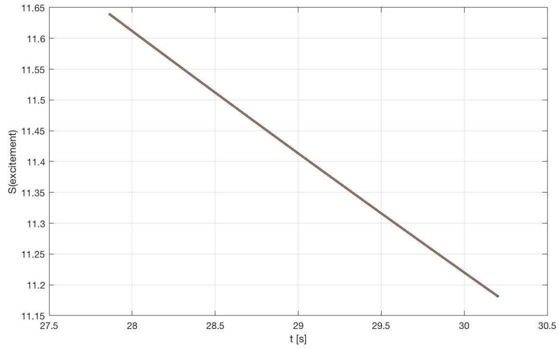


(e) Gz_upcicle

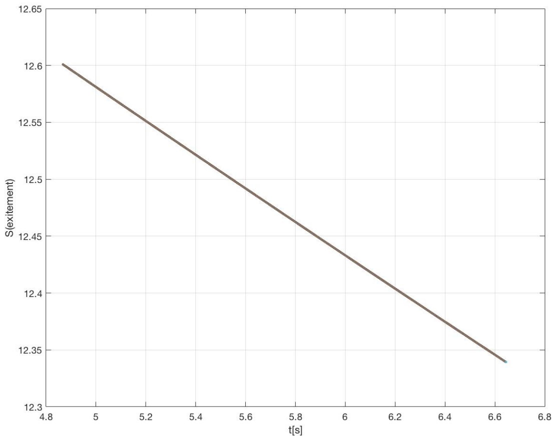
Figure A.2: G_y or G_z for the five circular track



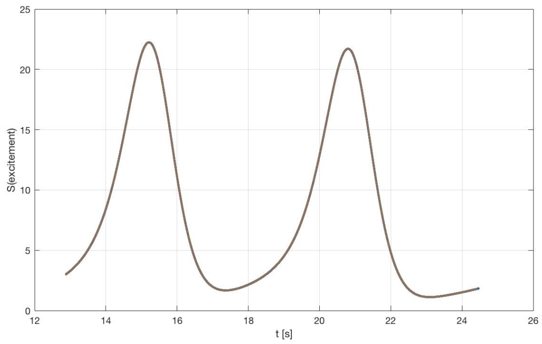
(a) S_downcircle



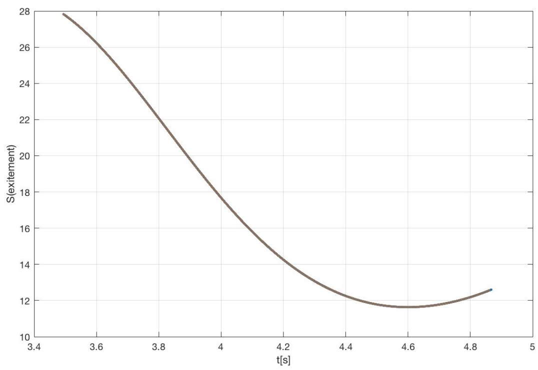
(b) S_horicircle



(c) S_midcircle



(d) S_orbit



(e) S_upcircle

Figure A.3: $S_{excitement}$ for the five circular track

B MATLAB Codes

[11]

```
%Euler Method for downward cant
function [xn,vn,an,tn] = cant(x0,v0,a0,t0)
a1 = -0.00037*v0^2+9;
v1 = v0 + a1*0.0001;
x1 = x0 + v1*0.0001;
t1 = t0 + 0.0001;
if x1<31.0583
    [xn,vn,an,tn]=cant(x1,v1,a1,t1);
else
    xn = x0;      vn = v0;
    an = a0;      tn = t0;
end
```

```
%Euler Method for horizontal track
function [xn,vn,an,tn] = horizontal(x0,v0,a0,t0)
a1 = -0.00037*v0^2-0.0098;
v1 = v0 + a1*0.0001;
x1 = x0 + v1*0.0001;
t1 = t0 + 0.0001;
if x1<10
    plotv(t1,v1);
    [xn,vn,an,tn]=horizontal(x1,v1,a1,t1);
else
    xn = x0;
    vn = v0;
    an = a0;
    tn = t0;
end
```

```
%Euler Method for upward circle track
function [xn,vn,an,bn,tn] = upcircle (x0,v0,a0,b0,t0)
a1 = -0.001*(v0^2/8+9.8*cos(x0)) - 0.00037*v0^2 - 9.8*sin(x0);
v1 = v0 + 0.0001*a1;
x1 = x0 + v1/8*0.0001;
b1 = v1^2/8;
t1 = t0 + 0.0001;
if x1<pi
    s = 4*a1/9.8+4*b1/9.8+0.05*v1+0.1*8*(1-cos(x1));
    plot(t1,s,'.');
    hold on;
    [xn,vn,an,bn,tn] = upcircle(x1,v1,a1,b1,t1);
else
    xn = x0;
    vn = v0;
    an = a0;
    bn = b0;
    tn = t0;
end
```

[11] We apply recursion to the Matlab Code. The function require initial conditions of the car as the input and can output the emotion condition at any instant of time.

```
%Euler Method for middle circle track
function [xn,vn,an,bn,tn] = midcircle(x0,v0,a0,b0,t0)
a1 = -0.001*(v0^2/8) - 0.00037*v0^2;
%assume surface orbits but self-rotation ignored
v1 = v0 + 0.0001*a1;
x1 = x0 + v1/8*0.0001;
b1 = v1^2/8;
t1 = t0 + 0.0001;
if x1<pi
    s = 4*a1/9.8+4*b1/9.8+0.05*v1+1.6;
    plot(t1,s,'.');    hold on;
    plotG(t1,b1);
    [xn,vn,an,bn,tn] = midcircle(x1,v1,a1,b1,t1);
else
    xn = x0;
    vn = v0;
    an = a0;
    bn = b0;
    tn = t0;
end
```

```
%Euler Method for downward circle track
function [xn,vn,an,bn,tn]=downcircle(x0,v0,a0,b0,t0)
a1 = -0.001*(v0^2/8+9.8*cos(x0+pi)) - 0.00037*v0^2 - 9.8*sin(x0+pi);
v1 = v0 + 0.0001*a1;
x1 = x0 + v1/8*0.0001;
b1 = v1^2/8;
t1 = t0 + 0.0001;
if x1<pi
    s = 4*a1/9.8+4*b1/9.8+0.05*v1+0.1*8*(1-cos(x1+pi));
    plot(t1,s,'.');    hold on;
    plotG(t1,b1);
    [xn,vn,an,bn,tn] = downcircle(x1,v1,a1,b1,t1);
else
    xn = x0;
    vn = v0;
    an = a0;
    bn = b0;
    tn = t0;
end
```

```
%Euler Method for climbing track
function [xn,vn,an,tn] = climb(x0,v0,a0,t0)
a1 = -0.00037*v0^2-8.714;
v1 = v0 + a1*0.001;
x1 = x0 + v1*0.001;
t1 = t0 + 0.001;
if x1<25
    [xn,vn,an,tn]=climb(x1,v1,a1,t1);
else
    xn = x0;    vn = v0;
    an = a0;    tn = t0;
end
```

```
%Euler Method for climbing track
function [xn,vn,an,bn,tn] = orbit(x0,v0,a0,b0,t0)
a1 = -0.001*(v0^2/8+5/26^0.5*9.8*cos(x0+pi))-5/26^0.5*9.8*sin(x0+pi)-0.00037*v0^2;
v1 = v0 + a1*0.001;
x1 = x0 + v1/8*0.001;
b1 = v1^2/8;
t1 = t0 + 0.001;

if x1<4*pi
    s = 4*a1/9.8+5*b1/9.8+0.05*v1+0.1*8*(1-cos(x1+pi));
    plot(t1,s,'.');
    hold on;
    %plotG(t1,b1);
    [xn,vn,an,bn,tn]=orbit(x1,v1,a1,b1,t1);
else
    xn = x0;
    vn = v0;
    an = a0;
    bn = b0;
    tn = t0;
end
```

```
%Function to Plot G
function plotG(t,a)
plot(t,a/9.8,'.');
hold on;
end
```

```
%Function to Plot G
function plotv(t,v)
plot(t,v,'.');
hold on;
end
```