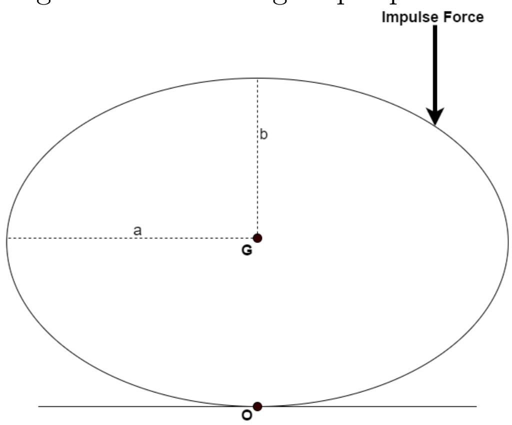
Contents

1 Introduction

This report is intended to address the problem of an ellipse "rolling" on a flat surface without slipping due to an initial applied impulse force shown in Figure 1. This report will first introduce the coordinate system, system constraints, and system's degree of freedom. The Lagrangian can be found from the potential and kinetic energy and the equations of motion can be obtained using the Lagrangian. The last section of this report explains the process and MATLAB codes used to animate the system.

Figure 1: The rolling ellipse problem

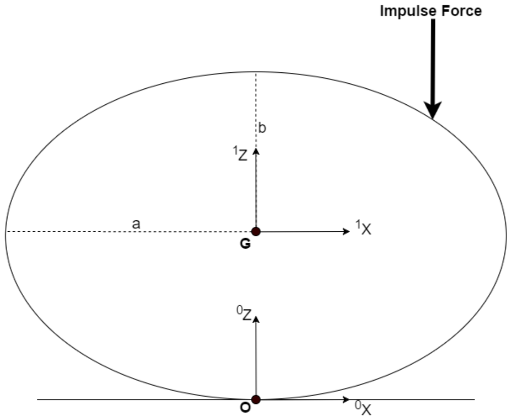


2 System Kinematics

2.1 Defining the coordinate systems

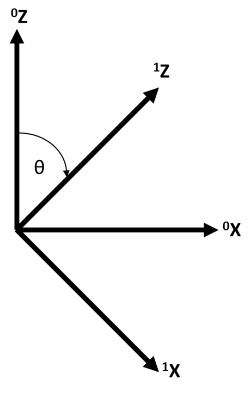
The inertial frame {0} has the origin at point **O** and frame {1} is rigidly attached to the ellipse (Figure 2).

Figure 2: Coordinate systems



The angle of rotation θ is defined as angle between frame $\{0\}$ and frame $\{1\}$ rotated in clockwise-direction as shown in Figure 3.

Figure 3: Coordinate system (show angle of rotation)



The generalized coordinates for the problem are denoted by $\mathbf{Q} = \{\mathbf{x},\ \mathbf{z},\ \theta,\ \psi,\ \mathbf{r}\}$. $\mathbf{x}$ and $\mathbf{z}$ describe the position of the center of gravity for the ellipse with respect to time. θ describes the orientation of the ellipse with respect to time. ψ describes the polar angle between vector $\mathbf{r}_{GC}$ and negative $^1\mathbf{Z}$ axis. $\mathbf{r}$ describes the magnitude of vector $\mathbf{r}_{GC}$.

2.2 Defining the constraints

In order to obtain the constraints, the position of the contact point **C** relative to the ellipse's center of gravity must be determined. The problem of "a rolling ellipse with no-slip condition" can thought of from the point of view of an observer in the reference frame attached to the ellipse. From the point of view of this observer the ground rolls around the stationary ellipse, shown in figure 4. This rotating ground is the tangent line to the contact point **C**. A polar angle (ψ) is used to represent the angle between the negative Z-axis of frame $\{1\}$ to the vector from the ellipse's center of gravity to the contact point $({}^{1}\mathbf{r}_{GC})$. This rotation as seen from the inertial frame can be seen in 5.

Figure 4: Rotation from the point of view of an observer on the ellipse

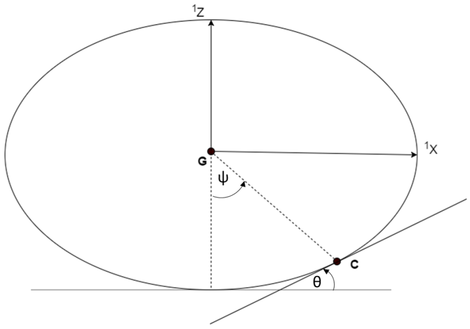
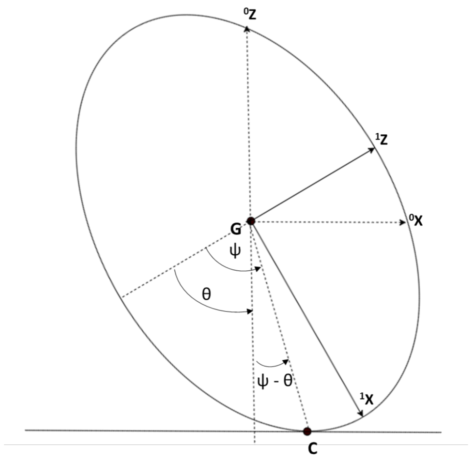


Figure 5: Rotation as seen by an observer in the inertial frame



As point **C** can be seen as rotating around the ellipse, the point **C** can be described using the parameterized equation (1) with its derivative seen in (2).

$$\begin{aligned} x'(t) &= a \sin t \\ z'(t) &= -b \cos t \end{aligned} \tag{1}$$

$$\begin{aligned} \frac{dx'}{dt} &= a \cos t \\ \frac{dz'}{dt} &= b \sin t \end{aligned} \tag{2}$$

where $x'(t)$ and $z'(t)$ represent the x and z position of the contact point on the ellipse in frame $\{1\}$.

The parameter t can be related to the polar angle ψ with equation (3).

$$\tan \psi = \frac{-x'}{z'} = \frac{a}{b} \tan t \tag{3}$$

Since the tangent line makes angle θ to the horizontal, the slope of the tangent line is given by equation (4).

$$\begin{aligned} \tan \theta &= \frac{dz'}{dx'} = \frac{b}{a} \tan t \\ \tan t &= \frac{a}{b} \tan \theta \end{aligned} \tag{4}$$

Combining equations (3) and (4), the relationship between θ and ψ is seen in equation (5).

$$\begin{aligned} \tan \psi &= \frac{a^2}{b^2} \tan \theta \\ \psi &= \arctan \left(\frac{a^2}{b^2} \tan \theta \right) \end{aligned} \tag{5}$$

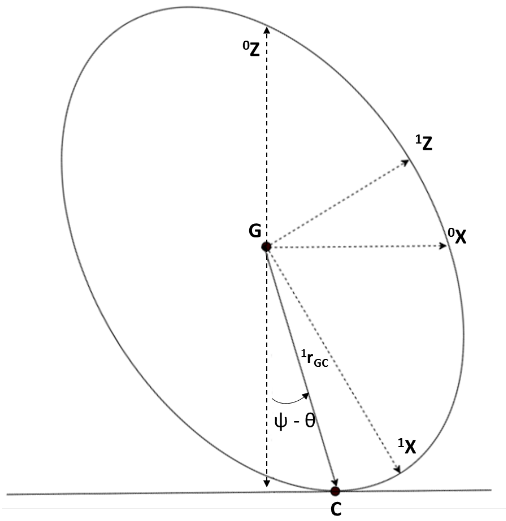
The magnitude of the vector ${}^{1}\mathbf{r}_{OC}$ is given by $\mathbf{r}$, where $\mathbf{r}$ is the radius of the ellipse at ψ,

$$r = \frac{ab}{\sqrt{b^2 \sin^2 \psi + a^2 \cos^2 \psi}} \tag{6}$$

Using the angle $(\psi - \theta)$ (see figure 6), the relative position of point **C** to the center of gravity can be described as in equation (7).

$$^0\mathbf{r}_{GC} = \begin{bmatrix} \pm rsin(\psi - \theta) \\ 0 \\ -rcos(\psi - \theta) \end{bmatrix} \tag{7}$$

Figure 6: Position of the center of gravity relative to point **C**



Using (7), the vector $^0\mathbf{r}_{OC}$ is

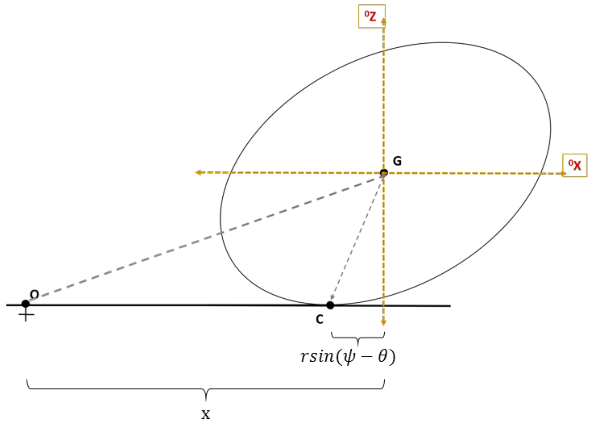
$$\begin{aligned} ^0\mathbf{r}_{OC} &=^0 \mathbf{r}_{OG} +^0 \mathbf{r}_{GC} = \begin{bmatrix} x \\ 0 \\ z \end{bmatrix} + \begin{bmatrix} \pm rsin(\psi - \theta) \\ 0 \\ -rcos(\psi - \theta) \end{bmatrix} \\ ^0\mathbf{r}_{OC} &= \begin{bmatrix} x \pm rsin(\psi - \theta) \\ 0 \\ z - rcos(\psi - \theta) \end{bmatrix} \end{aligned} \tag{8}$$

From equation (8), the term ${}^0\mathbf{r}_{GC,x}$ can be either positive or negative. The sign of ${}^0\mathbf{r}_{GC,x}$ is dependent on the angle of rotation θ, if θ is in the second quadrant the sign will be negative and in the other quadrants, the term ${}^0\mathbf{r}_{GC,x}$ will be positive. This is explained below:

1. **Case 1**: ${}^0\mathbf{r}_{GC,x} = -rsin(\psi - \theta)$

   This case will occur when $\frac{\pi}{2} \leq \theta \leq \pi$ (which indicates that θ is in the second quadrant). During this stage, the point of contact (point **C**) is located to the left of the ellipse's center of gravity as shown in figure 7.

Figure 7: Point **C** is located to the left of the center of gravity when θ is in the second quadrant

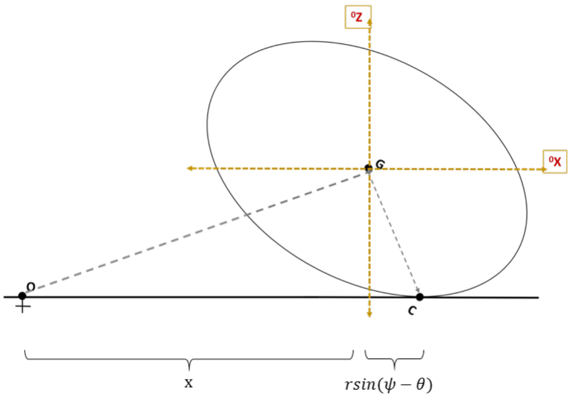


   From the above figure, it can be seen that ${}^0\mathbf{r}_{OC,x}$ is defined as $\mathbf{x} - rsin(\psi - \theta)$

2. **Case 2**: ${}^0\mathbf{r}_{GC,x} = rsin(\psi - \theta)$

   This case will occur when θ is in the first, third and fourth quadrant. During this stage, the point of contact (point **C**) is located to the right of the ellipse's center of gravity as shown in figure 8.

Figure 8: Point **C** is located to the right of the center of gravity when θ is in the first, third, fourth quadrant



From the above figure, it can be seen that ${}^0\mathbf{r}_{OC,x}$ is defined as $\mathbf{x} + rsin(\psi - \theta)$

From (8), the absolute velocity of point **C** can be computed using

$$\begin{aligned} {}^0\dot{\mathbf{r}}_{OC,x} &= {}^0\mathbf{r}'_{OC,x} \\ &= \dot{x} \pm \frac{d}{dt}\Big(r(\psi)sin(\psi-\theta)\Big) \end{aligned} \tag{9}$$

Using (9) and the no-slip condition the constraints can be calculated. According to the no-slip condition we can say that the velocity of the contact point is always 0, and as the ellipse is rolling on the ground we can also say that the displacement of the contact point in the z direction is always zero. Equation (9) was derived using MATLAB (see Appendix B) to give f_1, and f_2 was taken from (8) as ${}^0\mathbf{r}_{OC,z} = 0$ (to keep the ellipse on the ground). The third constraint f_3 refers to the relationship provided in (6). The fourth constraint f_4 refers to the relationship provided in (5). The constraint equations can be seen in equation (10).

$$
\begin{aligned}
&f_1(\dot{x},\dot{\theta},\theta,\dot{\psi},\psi,\dot{r},r) = 0 = \; \dot{x} \pm rcos(\psi-\theta)(\dot{\psi}-\dot{\theta}) \pm \dot{r}sin(\psi-\theta) \\
&f_2(z,\theta,\psi) = 0 = z - rcos(\psi-\theta) \\
&f_3(r,\psi) = 0 = r - \frac{ab}{\sqrt{b^2\sin^2\psi + a^2\cos^2\psi}} \\
&f_4(\theta,\psi) = 0 = \psi - \arctan\left(\frac{a^2}{b^2}\tan\theta\right)
\end{aligned} \tag{10}
$$

In this case, the constraint f_1 is a non-holonomic constraint and the constraint f_2, f_3 and f_4 is a holonomic constraint.

3 Finding the Equations of Motion

3.1 The Lagrangian

In this particular case, the ellipse rotates as the contact point traverses along the x axis when an impulse force is applied on it. As the ellipse moves forward the height of the center of mass changes simultaneously with it according to the constraint f_2. The ellipse will have kinetic energy from its motion, and potential energy due to gravity only.

3.1.1 Kinetic Energy

The ellipse will have both rotational as well as linear motion in this case which signifies that it will have both rotational as well as linear kinetic energy.

The kinetic energy due to the linear motion of the ellipse is seen in equation (11).

$$\begin{aligned} T_{linear} &= \frac{1}{2}\left({}^{0}\dot{\mathbf{r}}_{OG}^{T} m {}^{0}\dot{\mathbf{r}}_{OG}\right) \\ &= \frac{1}{2}\left(m\dot{x}^2 + m\dot{z}^2\right) \end{aligned} \tag{11}$$

The rotational kinetic energy of the ellipse of mass m and angular velocity $\dot{\theta}$ is given in equation (12).

$$\begin{aligned} T_{rot} &= \frac{1}{2} {}^{1}\omega_1^{T} {}^{1}\mathbf{I}^{G} {}^{1}\omega_1 \\ &= \frac{1}{2} I_{yy}^{G} \dot{\theta}^2 \\ &= \frac{1}{2} \cdot \frac{m(a^2+b^2)}{4}\dot{\theta}^2 \end{aligned} \tag{12}$$

Hence, the total kinetic energy of the ellipse can be seen in equation (13).

$$T = \frac{1}{2}\left(m\dot{x}^2 + m\dot{z}^2 + \frac{m(a^2+b^2)}{4}\dot{\theta}^2\right) \tag{13}$$

3.1.2 Potential Energy

Potential energy in this body will be produced due to the change in height of the center of mass of the ellipse as it rolls. Hence, the potential energy of the ellipse of mass m is seen in (14) with the ground plane through the origin.

$$V = mgz \tag{14}$$

3.1.3 Lagrangian

The Lagrangian of the body is given in (15).

$$\begin{aligned}\mathcal{L} &= T - V \\ \mathcal{L} &= \frac{1}{2}\left(m\dot{x}^2 + m\dot{z}^2 + \frac{m(a^2+b^2)}{4}\dot{\theta}^2\right) - mgz\end{aligned} \tag{15}$$

3.2 Lagrange Multiplier Method

In order to calculate the Euler-Lagrange equation the partial derivatives of the Lagrangian with respect to each generalised coordinate and its first derivative were taken. The results of this can be seen in equations (16) to (20).

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) = m\ddot{x} \qquad \frac{\partial L}{\partial x} = 0 \tag{16}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{z}}\right) = m\ddot{z} \qquad \frac{\partial L}{\partial z} = -mg \tag{17}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) = \frac{\ddot{\theta}m(a^2+b^2)}{4} \qquad \frac{\partial L}{\partial \theta} = 0 \tag{18}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\psi}}\right) = 0 \qquad \frac{\partial L}{\partial \psi} = 0 \tag{19}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{r}}\right) = 0 \qquad \frac{\partial L}{\partial r} = 0 \tag{20}$$

For a system without constraints this would return the equation of motion seen in equation (21) for each generalised coordinate.

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} = 0 \tag{21}$$

This must be augmented with the constraint equations and their corresponding lagrange multipliers. As we have four constraints and five generalised coordinates, this results in the equations (22) to (26).

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) - \frac{\partial L}{\partial x} = a_{1x}\lambda_1 + a_{2x}\lambda_2 + a_{3x}\lambda_3 + a_{4x}\lambda_4 \tag{22}$$
$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{z}}\right) - \frac{\partial L}{\partial z} = a_{1z}\lambda_1 + a_{2z}\lambda_2 + a_{3z}\lambda_3 + a_{4z}\lambda_4 \tag{23}$$
$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \frac{\partial L}{\partial \theta} = a_{1\theta}\lambda_1 + a_{2\theta}\lambda_2 + a_{3\theta}\lambda_3 + a_{4\theta}\lambda_4 \tag{24}$$
$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\psi}}\right) - \frac{\partial L}{\partial \psi} = a_{1\psi}\lambda_1 + a_{2\psi}\lambda_2 + a_{3\psi}\lambda_3 + a_{4\psi}\lambda_4 \tag{25}$$
$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{r}}\right) - \frac{\partial L}{\partial r} = a_{1r}\lambda_1 + a_{2r}\lambda_2 + a_{3r}\lambda_3 + a_{4r}\lambda_4 \tag{26}$$

Using the values calculated in equations (16) to (20) these equations can be seen in equations (27) to (31).

$$m\ddot{x} = a_{1x}\lambda_1 + a_{2x}\lambda_2 + a_{3x}\lambda_3 + a_{4x}\lambda_4 \tag{27}$$
$$m\ddot{z} + mg = a_{1z}\lambda_1 + a_{2z}\lambda_2 + a_{3z}\lambda_3 + a_{4z}\lambda_4 \tag{28}$$
$$\frac{\ddot{\theta}m(a^2 + b^2)}{4} = a_{1\theta}\lambda_1 + a_{2\theta}\lambda_2 + a_{3\theta}\lambda_3 + a_{4\theta}\lambda_4 \tag{29}$$
$$0 = a_{1\psi}\lambda_1 + a_{2\psi}\lambda_2 + a_{3\psi}\lambda_3 + a_{4\psi}\lambda_4 \tag{30}$$
$$0 = a_{1r}\lambda_1 + a_{2r}\lambda_2 + a_{3r}\lambda_3 + a_{4r}\lambda_4 \tag{31}$$

The four constraint equations can be written in Pfaffian form in order to derive the constraint matrix for the Lagrange Multiplier method. These can be seen in equation (32).

$$
\begin{aligned}
df_1 &= 0 = \underline{1} \cdot dx + \underline{\sin(\psi-\theta)} \cdot dr + \underline{r\cos(\psi-\theta)} \cdot d\psi \ \underline{-r\cos(\psi-\theta)} \cdot d\theta + \underline{0} \cdot dz \\
df_2 &= 0 = \underline{0} \cdot dx \ \underline{-\cos(\psi-\theta)} \cdot dr \ \underline{-r\sin(\psi-\theta)} \cdot d\psi + \underline{r\sin(\psi-\theta)} \cdot d\theta + \underline{1} \cdot dz \\
df_3 &= 0 = \underline{0} \cdot dx + \underline{2\left(b^2\sin^2(\psi) + a^2\cos^2(\psi)\right)} \cdot dr \ \underline{-ab(a^2-b^2)\sin^2(\psi)} \cdot d\psi + \underline{0} \cdot d\theta + \underline{0} \cdot dz \\
df_4 &= 0 = \underline{0} \cdot dx + \underline{0} \cdot dr + \underline{(\frac{a^4\tan^2(\theta)}{b^2} + b^2)} \cdot d\psi \underline{-a^2(\tan^2(\theta)+1)} \cdot d\theta + \underline{0} \cdot dz
\end{aligned} \tag{32}
$$

From the Pfaffian form the coefficients attached to each of the generalised coordinates are extracted and placed in to a matrix.

The underlined coefficients from equation (32) are placed in the matrix below. These are used to augment the Euler-Lagrange equation calculated in equations (27) to (31).

$$
\begin{bmatrix}
1 & \sin(\psi-\theta) & r\cos(\psi-\theta) & -r\cos(\psi-\theta) & 0 \\
0 & -\cos(\psi-\theta) & -r\sin(\psi-\theta) & r\sin(\psi-\theta) & 1 \\
0 & 2\left(b^2\sin^2(\psi) + a^2\cos^2(\psi)\right)^{3/2} & -ab(a^2-b^2)\sin(2\psi) & 0 & 0 \\
0 & 0 & (\frac{a^4\tan^2(\theta)}{b^2} + b^2) & -a^2(\tan^2(\theta)+1) & 0
\end{bmatrix}
$$

By substituting these constraint coefficients into equations (27) through (31) above, the equations of motion can be found. The equations of motion can be seen in equations (33) to (37).

$$
\begin{aligned}
0 &= m\ddot{x} - \lambda_1 && (33) \\
0 &= m\ddot{z} - \lambda_2 + mg && (34) \\
0 &= \frac{\ddot{\theta}m(a^2+b^2)}{4} + \lambda_1 r\cos(\psi-\theta) - \lambda_2 r\sin(\psi-\theta) - \lambda_4 a^2\sec^2\theta && (35) \\
0 &= \lambda_1\sin(\psi-\theta) - \lambda_2\cos(\psi-\theta) + 2\lambda_3\left(b^2\sin^2(\psi) + a^2\cos^2(\psi)\right)^{3/2} && (36) \\
0 &= \lambda_1 r\cos(\psi-\theta) - \lambda_2 r\sin(\psi-\theta) + \\
&\qquad \lambda_3 ab(a^2-b^2)\sin 2\psi \ + \lambda_4\left(\frac{a^4\tan^2(\theta)}{b^2} + b^2\right) && (37)
\end{aligned}
$$

In order to solve these equations of motion the velocity form of the constraints must also be found, these follow from the Pfaffian form of the constraints seen in equation (32) and can be seen in equation (38).

$$
\begin{aligned}
\frac{df_1}{dt} &= 0 = \dot{x} - \dot{r}\cos(\psi-\theta) - (\dot{\psi}-\dot{\theta})r\sin(\psi-\theta) \\
\frac{df_2}{dt} &= 0 = \dot{z} - \dot{r}\sin(\psi-\theta) - \dot{\psi}\cdot r\sin(\psi-\theta) + \dot{\theta}\cdot r\cos(\psi-\theta) \\
\frac{df_3}{dt} &= 0 = 2\dot{r}\left(b^2\sin^2(\psi)+a^2\cos^2(\psi)\right)^{3/2} - \dot{\psi}\cdot ab(a^2-b^2)\sin(2\psi) \\
\frac{df_4}{dt} &= 0 = \dot{\psi}\left(\frac{a^4\tan^2(\theta)}{b^2}+b^2\right) - \dot{\theta}\left(a^2(\tan^2(\theta)+1)\right)
\end{aligned} \tag{38}
$$

3.3 Solving the Euler-Lagrange equation and the Equation of Constraint

In this section, the Euler-Lagrange equations and Equations of Constraint (in Velocity form) obtained in Section 3.2 are used to solve for $\{\ddot{x}, \ddot{z}, \ddot{r}, \ddot{\theta}, \ddot{\psi}\}$ in terms of the generalized coordinates and their derivatives $(x, \dot{x}, z, \dot{z}, r, \dot{r}, \theta, \dot{\theta}, \psi, \dot{\psi})$.

Using the first four equations of motion, the Lagrange multipliers can be solved in terms of $\{x, \dot{x}, \ddot{x}, z, \dot{z}, \ddot{z}, r, \dot{r}, \ddot{r}, \theta, \dot{\theta}, \ddot{\theta}, \psi, \dot{\psi}, \ddot{\psi}\}$.

$$
\begin{aligned}
\lambda_1 &= m\ddot{x} \\
\lambda_2 &= m(\ddot{z}-g) \\
\lambda_3 &= \frac{F}{4a^3b^3sin(2\psi)(a^2-b^2)}, \quad \text{where F is obtained from (40)} \\
\lambda_4 &= \frac{-m\cos^2(\theta)((a^2+b^2)\ddot{\theta}+4r\ddot{x}\cos(\psi-\theta)-4r\ddot{z}\sin(\psi-\theta)+4rg\sin(\psi-\theta))}{4a^2b^2}
\end{aligned} \tag{39}
$$

$$
\begin{aligned}
F = {} & b^6\ddot{\theta}\cdot\cos^2(\theta) + a^6\ddot{\theta}\cdot\cos^2(\theta) + a^4b^2\tan^2(\theta) + a^2b^4\ddot{\theta}\cdot\cos^2(\theta) - 4a^2b^2r\cos(\psi-\theta)\ddot{x} \\
& + a^4b^2\ddot{\theta}\cdot\cos^2(\theta)\cdot\tan^2(\theta) + 4a^2b^2\ddot{z}r\sin(\psi-\theta) - 4a^2b^2gmr\sin(\psi-\theta) \\
& + 4b^4\ddot{x}mr\cos(\psi-\theta) + 4a^4\ddot{x}mr\cos(\psi-\theta)\cos^2(\theta)\tan^2(\theta) - 4a^4\ddot{z}mr\sin(\psi)\cos^2(\theta) \\
& - 4b^4\ddot{z}mr\sin(\psi-\theta)\cos^2(\theta) + 4b^4gmr\sin\sin(\psi-\theta)\cos^2(\theta) \\
& + 4a^4\ddot{x}mr\cos(\psi-\theta)\cos^2(\theta)\tan^2(\theta) - 4a^4\ddot{z}mr\sin(\psi-\theta)\cos^2(\theta)\tan^2(\theta) \\
& + 4a^4gmr\sin(\psi-\theta)\cos^2(\theta)\tan^2(\theta)
\end{aligned} \tag{40}
$$

Using results obtained in (39), equation (37) can be rewritten in terms of $\{x, \dot{x}, \ddot{x}, z, \dot{z}, \ddot{z}, r, \dot{r}, \ddot{r}, \theta, \dot{\theta}, \ddot{\theta}, \psi, \dot{\psi}, \ddot{\psi}\}$ only,

$$0 = \cos(\psi - \theta)(\ddot{z}m - gm) - \ddot{x}m \cdot \sin(\psi - \theta) - \frac{F1}{4a^3b^3 \sin(2\psi)(a^2 - b^2)} \tag{41}$$

where F1 is,

$$\begin{aligned} F1 = {} & \cos(\psi - \theta)(\ddot{z}m - gm) - \ddot{x}m \cdot \sin(\psi - \theta) - (2a^2 \cos^2(\psi) + 2b^2 \sin(\psi))b^6\ddot{\theta}m \cos^2(\theta) \\ & + a^6\ddot{\theta}m \cos^2(\theta) \tan^2(\theta) + a^2b^4\ddot{\theta}m \cos^2(\theta) - 4a^2b^2\ddot{x}mr \cos(\psi - \theta) + a^4b^2\ddot{\theta}m \cos^2(\theta) \tan^2(\theta) \\ & + 4a^2b^2\ddot{z}mr \sin(\psi - \theta) - 4a^2b^2gmr \sin(\psi - \theta) + 4b^4\ddot{x}mr \cos(\psi - \theta) \cos^2(\theta) \\ & - 4b^4\ddot{z}mr \sin(\psi - \theta) \cos^2(\theta) + 4b^4gmr \sin(\psi - \theta) \cos^2(\theta) + 4a^4\ddot{x}mr \cos(\psi - \theta) \cos^2(\theta) \tan^2(\theta) \\ & - 4a^4\ddot{z}mr \sin(\psi - \theta) \cos^2(\theta) \tan^2(\theta) + 4a^4gmr \sin(\psi - \theta) \cos^2(\theta) \tan^2(\theta)) \end{aligned} \tag{42}$$

Now with equation (41), and the second derivatives of the constraint equations, the solution for $\{\ddot{x}, \ddot{z}, \ddot{r}, \ddot{\theta}, \ddot{\psi}\}$ can be found by solving these 5 equations simultaneously in MATLAB. The code used to compute these solutions is provided in Appendix C.

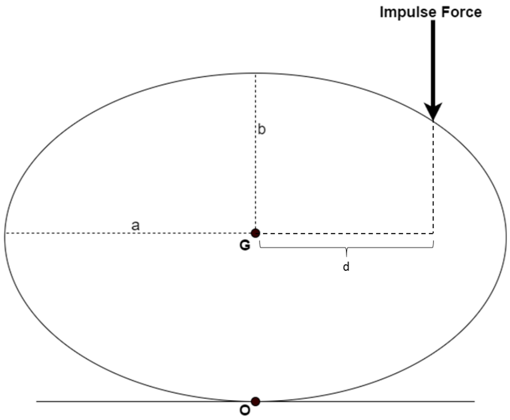
3.4 Impulsive Force

The generalised momentum corresponding to a generalised coordinate is seen in equation (43).

$$p = \frac{\partial L}{\partial \dot{q}_i} \tag{43}$$

The impulse momentum equation can be used to find the effect of the applied impulse applied at horizontal distance d from the centre of gravity, seen in Figure 9.

Figure 9: The rolling ellipse problem



The impulse force pictured in Figure 9 creates a moment around point G. There is a reaction impulse at point O due to a constraint, but only $\dot{\theta}$ will change as the reaction force will do no work. The effect of the impulse can be found using the impulse momentum equation seen in equation (44).

$$\Delta p = \int_{t^-}^{t^+} Q_\theta \cdot dt \tag{44}$$

Because the ellipse starts at rest and the impulse is applied at time zero, the impulse force has the effect of setting the initial angular velocity $\dot{\theta}$. This can be seen in equation (45).

$$I_{yy}\dot{\theta}(0^+) = \int_{0^-}^{0^+} F\delta(t) \cdot d \cdot dt \tag{45}$$

$$I_{yy}\dot{\theta}(0^+) = F \cdot d \tag{46}$$

$$\dot{\theta}(0^+) = \frac{F \cdot d}{I_{yy}} \tag{47}$$

$$\dot{\theta}(0^+) = \frac{4F \cdot d}{m(a^2 + b^2)} \tag{48}$$

4 Simulation of the Ellipse

4.1 State-space equations

The state variables of the system can be defined as followed:

$$
\begin{aligned}
X_1 &= x & X_6 &= \dot{x} \\
X_2 &= z & X_7 &= \dot{z} \\
X_3 &= r & X_8 &= \dot{r} \\
X_4 &= \theta & X_9 &= \dot{\theta} \\
X_5 &= \psi & X_{10} &= \dot{\psi}
\end{aligned}
$$

Thus the state dynamics can be defined as,

$$
\begin{aligned}
\dot{X}_1 &= \dot{x} = X_6 & \dot{X}_6 &= \ddot{x} = f_1(x, \dot{x}, z, \dot{z}, r, \dot{r}, \theta, \dot{\theta}, \psi, \dot{\psi}) \\
\dot{X}_2 &= \dot{z} = X_7 & \dot{X}_7 &= \ddot{z} = f_2(x, \dot{x}, z, \dot{z}, r, \dot{r}, \theta, \dot{\theta}, \psi, \dot{\psi}) \\
\dot{X}_3 &= \dot{r} = X_8 & \dot{X}_8 &= \ddot{r} = f_3(x, \dot{x}, z, \dot{z}, r, \dot{r}, \theta, \dot{\theta}, \psi, \dot{\psi}) \\
\dot{X}_4 &= \dot{\theta} = X_9 & \dot{X}_9 &= \ddot{\theta} = f_4(x, \dot{x}, z, \dot{z}, r, \dot{r}, \theta, \dot{\theta}, \psi, \dot{\psi}) \\
\dot{X}_5 &= \dot{\psi} = X_{10} & \dot{X}_{10} &= \ddot{\psi} = f_5(x, \dot{x}, z, \dot{z}, r, \dot{r}, \theta, \dot{\theta}, \psi, \dot{\psi})
\end{aligned}
$$

where f_1, f_2, f_3, f_4, and f_5 are the solutions for $\{\ddot{x}, \ddot{z}, \ddot{r}, \ddot{\theta}, \ddot{\psi}\}$ obtained in Appendix C. Now that the state dynamic vector $\dot{X}$ is expressed only in the terms of $\{x, \dot{x}, z, \dot{z}, r, \dot{r}, \theta, \dot{\theta}, \psi, \dot{\psi}\}$, we can use MATLAB's numerical solver tool to solve for $\dot{X}$ given a specific set of initial conditions.

4.2 Simulation

The ellipse was simulated using the above state-space model and MATLAB's numerical solver. As the equations of motion found in Appendix C were difficult to read due to their length, **eom2.m** was written to solve for these numerically at each timestep. This resulted in a much longer simulation time, but made debugging possible.

4.2.1 Description of the simulation files

The simulation runs using 3 files: eom2.m, Ellipse_Animation.m, and Approach_1_ODE.m

eom2.m:
: Numerically compute and solve the equations of motion when called by the ODE solver

Ellipse_Animation.m:
: Plot the animation using the constraint equations and the solutions from the ODE solver

Approach_1_ODE.m:
: Run the ODE solver using a set of initial conditions (obtained from the impulse force applied and constraint equations)

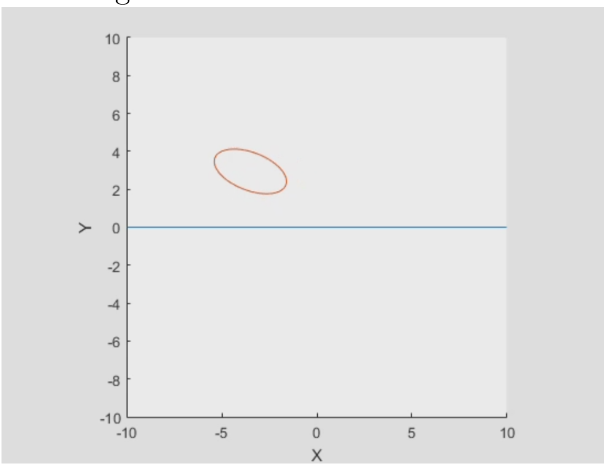
rotation.m:
: Rotate the input coordinates set with input rotation matrix and return the rotated set of coordinates

These files can be seen in Appendices D, E, F and G respectively.

4.2.2 Result

There were some issues with the simulation when solving for all generalised coordinates, this can be seen in the attached video, shown also in Figure 10.

Figure 10: Simulation Problems



It can be seen that the ellipse floats off the ground which is a clear violation of one of the constraints.

In order to discover the mistakes the results for the simulation were substituted back into the EoMs and EoCs. The results were all zero which indicates that the set of equations were being solved correctly.

Figure 11: Back Substitution into EoMs and EoCs

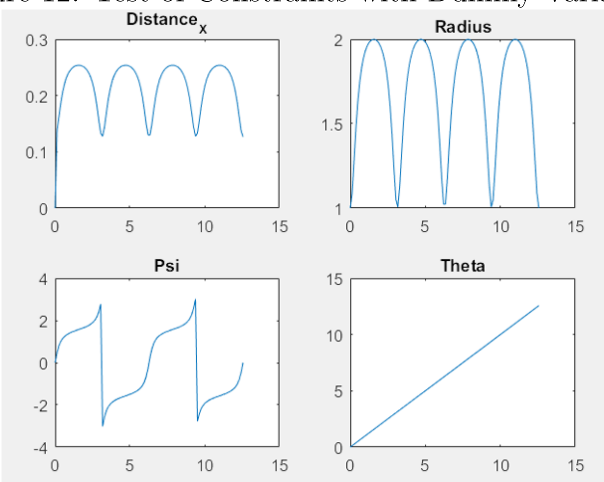
```
>> eq1_sub

eq1_sub =

0.0

>> eq2_sub

eq2_sub =

4.466878533124692530281198841444 5e-37

>> eq3_sub

eq3_sub =

-1.203706215242022408159986214115 6e-35

>> eq4_sub

eq4_sub =

2.586087571809032517531220381888 9e-37

>> eq5_sub

eq5_sub =

0
```

```
>> eq6_sub

eq6_sub =

0

>> eq7_sub

eq7_sub =

0

>> eq8_sub

eq8_sub =

0

>> eq9_sub

eq9_sub =

0
```

The constraints were also tested by substituting a dummy variable $\theta = t$. The results of this can be seen in Figure 12.

The results shown in Figure 12 were expected and indicate that the set up of the constraints was also done correctly.

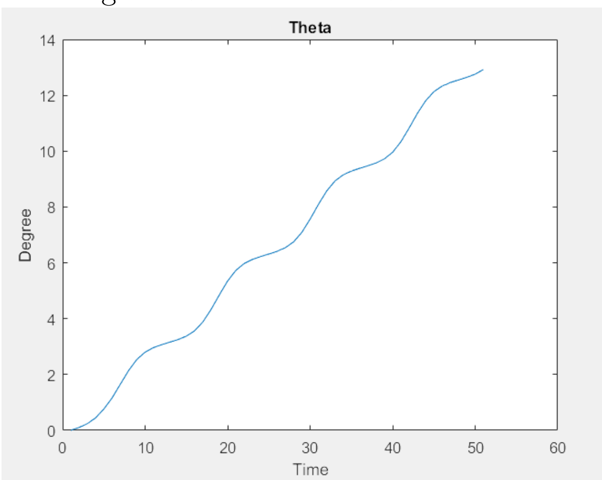
Figure 12: Test of Constraints with Dummy Variable



It can be seen in Figure 12 that ψ is discontinuous, which could mean the derivatives of ψ are not being handled well by the solver.

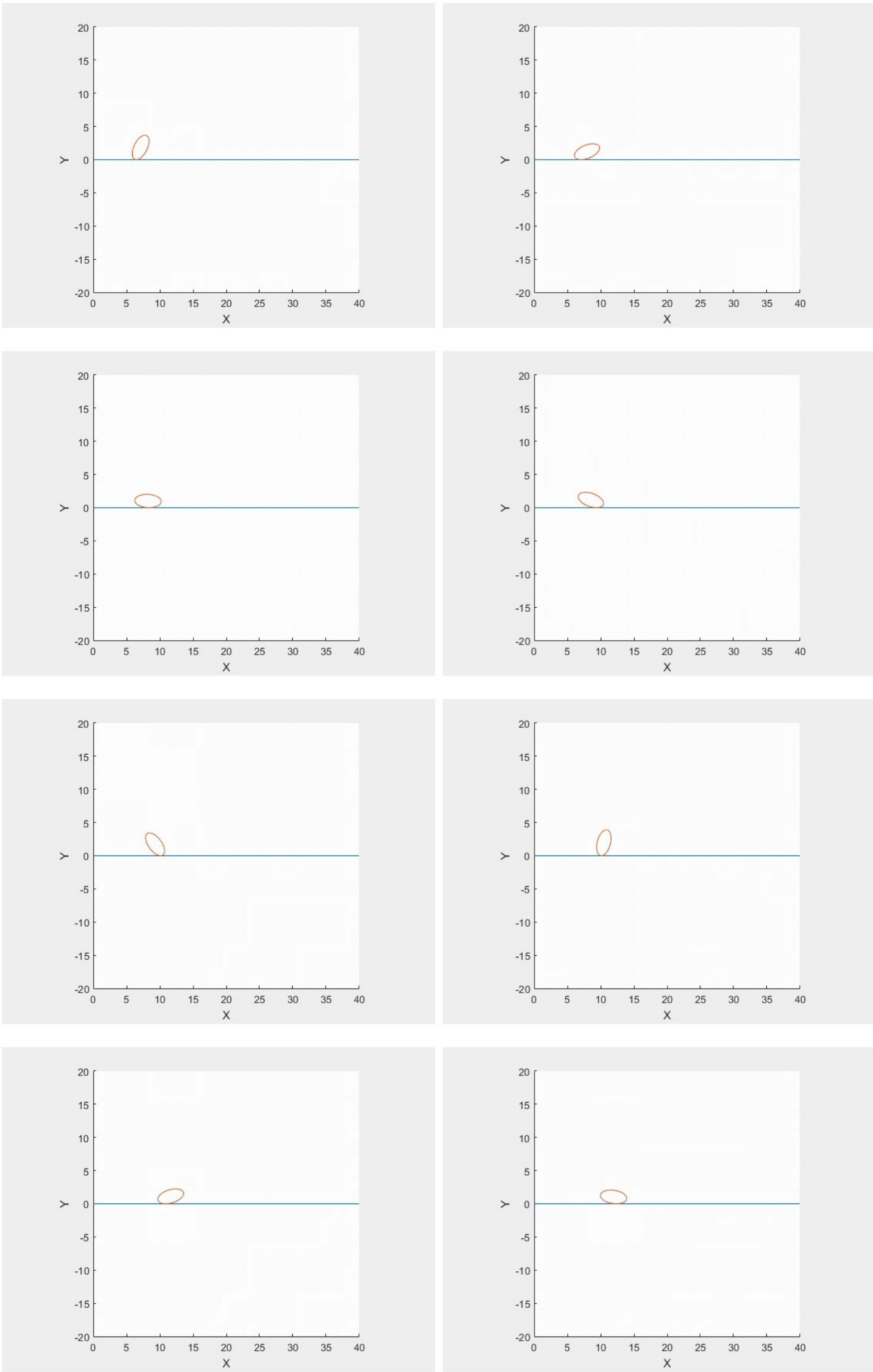
Looking closely at the video from the simulation, it can be seen that it actually solves θ correctly (seen in Figure 13). This result was used with the holonomic constraints to solve for four of the five generalised coordinates. The remaining generalised coordinate, $\dot{x}$, was then solved using numerical integration.

Figure 13: Theta from Simulation



The results of the simulation using the numerical integration can be seen in Figure 14.

Figure 14: Simulation of Ellipse



4.3 Assumptions

The simulation in the previous section is carried out with the implicit assumption that the impulse force is not too large that the ellipse "jumps." This would happen if $\dot{\theta}$ was large enough that the acceleration in the z direction from the rolling motion was able to overcome the gravitational force that keeps the ellipse on the ground.

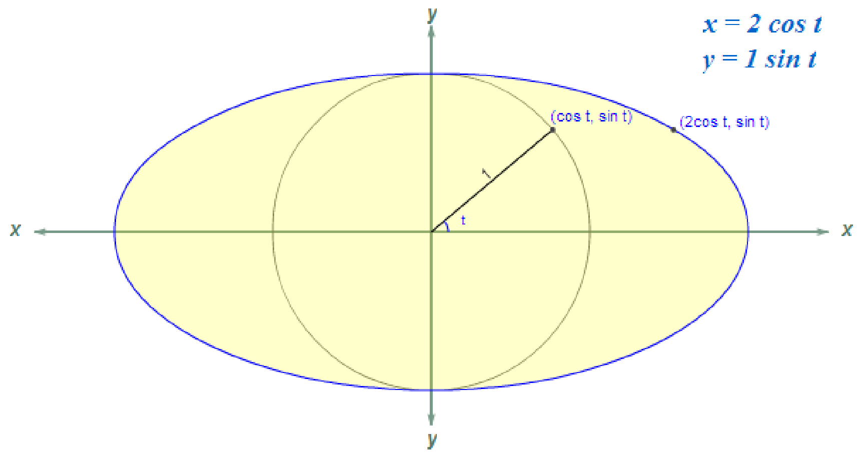
5 Appendices

A Parameterisation of an Ellipse

Reference: Math Open Reference

The parameter t

The parameter ***t*** can be a little confusing with ellipses. For any value of ***t***, there will be a corresponding point on the ellipse. But ***t*** is *not* the angle subtended by that point at the center. To see why this is so, consider an ellipse as a circle that has been stretched or squashed along each axis. In the figure below we start with a circle, and for simplicity give it a radius of one (a "unit circle").



The angle ***t*** defines a point on the circle which has the coordinates

$$x = \cos t$$
$$y = \sin t$$

The radius is one, so it is omitted. The blue ellipse is defined by the equations

$$x = 2\cos t$$
$$y = 1\sin t$$

So to get the corresponding point on the ellipse, the x coordinate is multiplied by two, thus moving it to the right. This causes the ellipse to be wider than the circle by a factor of two, whereas the height remains the same, as directed by the values 2 and 1 in the ellipse's equations.

So as you can see, the angle ***t*** is not the same as the angle that the point on the ellipse subtends at the center.

However, when you graph the ellipse using the parametric equations, simply allow t to range from 0 to 2π radians to find the (x, y) coordinates for each value of t.

B MATLAB Differentiation of ${}^{0}\mathbf{r}_{OC,x}$

```
syms x z t Th roc a b

th = sym('Th(t)');
X = sym('x(t)');
Z = sym('z(t)');


PSI=atan(a^2*tan(th)/b^2);
Th_p=atan(a*tan(th)/b);

%X component ROC
r_x =X+(((a*b)*sin(PSI-th))/(((b*b*sin(Th_p))^2)+((a*a*cos(Th_p))^2))^(1/2));
%Z component ROC
r_z = Z-(((a*b)*cos(PSI-th))/(((b*b*sin(PSI))^2)+((a*a*cos(PSI))^2))^(1/2));

A = diff(r_x);
pretty(A)
```

```
                      /             2  d                  \
                      |            a   -- Th(t) (#4 + 1)  |
                      |  d             dt                 |
          a b cos(#1) | -- Th(t) - -------------------- |
                      | dt               /  4        \    |
                      |                2 | a   #4     |    |
                      |              b   | ----- + 1 |    |
                      |                  |    4      |    |
 d                    \                  \   b       /    /
-- x(t) - -----------------------------------------------
dt                            sqrt(#2)

     /             /    4           3  d
     |             | 2 a  tan(Th(t))   -- Th(t) (#4 + 1)
     |             |                   dt
   - | a b sin(#1) | ----------------------------------
     |             |                     2
     \             \                   #3
```

```
        2  2             d
     2 a  b  tan(Th(t)) -- Th(t) (#4 + 1)
                        dt
   - ------------------------------------
                      #3

        6             d                \ \
     2 a  tan(Th(t)) -- Th(t) (#4 + 1) | |
                     dt                | |      3/2
   + --------------------------------- | |/(2 #2   )
                    2   2              | |
                   b  #3               / /
```

where

```
                     /  2            \
                     | a  tan(Th(t)) |
   #1 == Th(t) - atan| ------------- |
                     |       2       |
                     \      b        /

          4    2  2
         a    a  b  #4
   #2 == -- + --------
         #3      #3

          2
         a  #4
   #3 == ----- + 1
            2
           b

                       2
   #4 == tan(Th(t))
```

C MATLAB code for Double Derivatives

- Solving the EoM and EoC for all the double dot terms
- The coefficient from the equation of contraints in Pffafian form
- Setting up the EoM and EoC

C.1 Solving the EoM and EoC for all the double dot terms

```
%Filename: Approach_1_solving_generalized_coordinates.m
%% Solving the EoM and EoC for all the double dot terms
clear all

syms x dx ddx z dz ddz r dr ddr th dth ddth si dsi ddsi t a b m g

%% The coefficient from the equation of contraints in Pffafian form
a11 = 1;
a12 = sin(si-th);
a13 = r*cos(si-th);
a14 = -r*cos(si-th);
a15 = 0;
a21 = 0;
a22 = -cos(si-th);
a23 = -r*sin(si-th);
a24 = r*sin(si-th);
a25 = 1;
a31 = 0;
a32 = 2*(b^2*(sin(si))^2+a^2*(cos(si))^2)^(3/2);
a33 = -a*b*(sin(2*si))*(a^2-b^2);
a34 = 0;
a35 = 0;
a41 = 0;
a42 = 0;
a43 = a^4*(tan(th))^2/b^2+b^2;
a44 = a^2*((tan(th))^2+1);
a45 = 0;

%% Setting up the EoM and EoC
syms l1 l2 l3 l4
```