The University of Melbourne
Department of Mechanical Engineering

MCEN90038 Dynamics - Lab Report

Gyroscope Motion Simulation

Group 44

Tuan Khoi Nguyen (1025294)
Nguyen Hong Phu Trinh (1086263)
Ze Kai Cheong (1182444)

Introduction & Context

This report will describe the processes and outcome of an experiment aiming to model a toy gyroscope's operation process, representing a dynamic system that demonstrates the gyroscopic effect. First, the report will demonstrate a brief introduction to the gyroscopic concepts and the system representation of interest, which then leads to identifying the system behaviour. Following that, equations of motion will be constructed to put into the modelling stage, where suitable initial conditions and parameters will be chosen for simulation setup. After the simulation is done, an analysis on the final result given will also be done.

The Gyroscopic Effect

Overview

The Gyroscopic Effect is a common effect found in rotary motions. For a rotating body, its rotation axis will have the tendency to keep a steady orientation [1], and when a constant torque is applied externally, that axis itself undergoes a circular movement around the torque's normal with constant angular velocity, commonly known as precession [2, 3]. Therefore, when putting a rotor's rotation axis on a ball joint, provided that the initial angular velocity is large enough, that rotor will experience a circular movement rather than tipping over.

This effect happens due to the conservation of angular momentum and the 2 existing forces of the gyroscope: the normal that the joint exerts upwards, and the downward gravity force at the rotor's center of gravity. When the rotation axis's orientation is perfectly upward, no torque is created, and the axis remains in the same direction. Otherwise, when 2 forces are not perfectly aligned, they are still in parallel, creating a torque around the normal as a result. This torque adds up to the current angular momentum, and changed the latter's direction as a result. This causes the rotor axis to move towards the new angular momentum, which also changes its direction circularly with the torque, and created a cyclic movement as a result.

Applications

As the Gyroscopic Effect involves orientations and stability, it is commonly used in tasks that require navigational directions or self-balancing in real life.

Navigation With the rotation axis having steady orientation, attaching a rotating body to a frame will show what orientation of the latter is relative to the body's rotation axis. Therefore, as long as the rotation axis is known initially, directions of the attached frame of interest can be worked out. This application is commonly found in detecting heading on vehicles such as planes, ships and missiles [4], where it can negate the iron distorting effects [5] of the magnetometer, another common heading estimator that relies on Earth's magnetic field.

Stabilization Due to the self-balancing nature of the effect and its ability to turn by applying downward forces, the gyroscope is also applied to stabilizing using wheels, which is now seen most common in 2-wheeled means of transport such as bicycles or motorcycles [1]. This is also applied in more modern applications, such as preventing boats from tipping over [6], or self-balancing camera gimbals [7].

Experiment System

The simulation for the gyroscopic effect will be done on a toy gyroscope, which consists of 2 parts: the frame and the rotor. The rotor will be the rotating object, with the frame attaching the rotor's rotating axis to its vertical axis.

Rotor

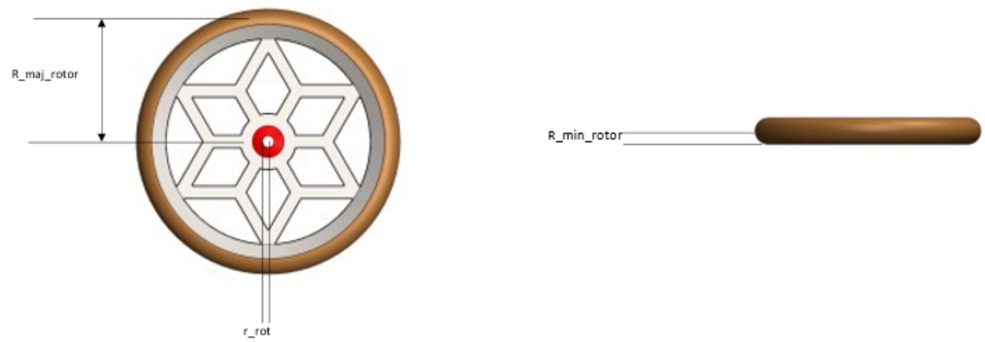


Figure 1: Rotor and its defined variables

The rotor in this assignment is assumed to be a solid torus that connects to the rotating axis and frame with massless links. This will be the rotary body of interest, which will be the source of precessions when gyroscopic effect happens.

Frame

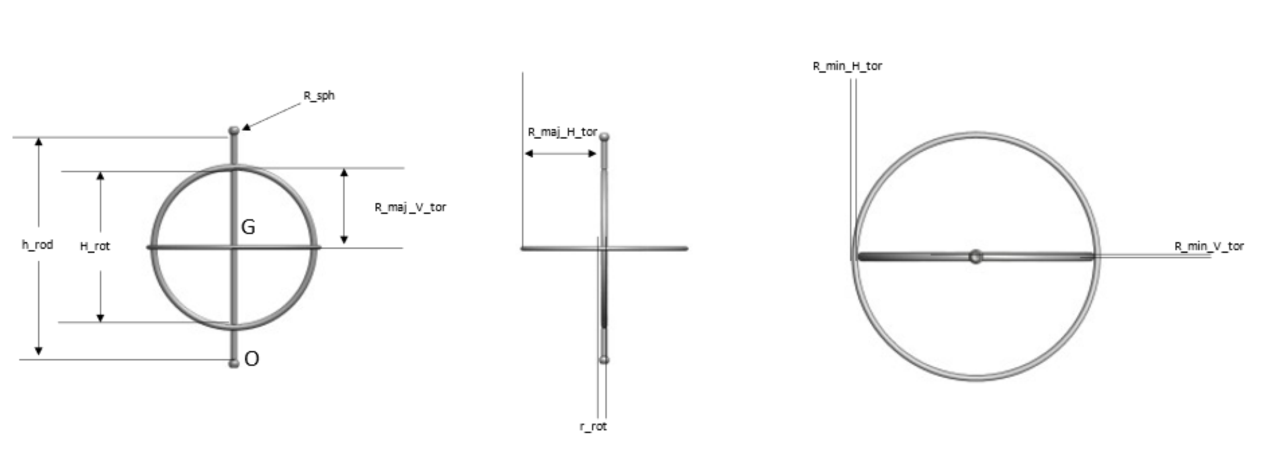


Figure 2: Frame and its defined variables

The frame is where the rotor's rotary axis will be linked to. It encloses the 2 heads of the axis like a revolute joint, allowing the rotor to rotate freely and independent from the frame's rotation, assuming no friction at the attachment.

Operation

The gyroscope will have its rotor, which is attached to the frame, spun with a high angular velocity and put on the support holder, which will act as a ball joint between the bottom tip of the frame and the holder's top.

Measurements

The measurement for the dimension variables of interest in Figures 1 and 2 are measured with a straight ruler with 1-millimeter (mm) unit. Other than that, mass of each part is also needed for this measurement, and they are measured using an electronic scale with 1-gram (g) unit. For each variable, measurements are done twice, and the average value is calculated to get the final estimation.

The result for this measurement process is listed in Table 1 below:

Variable	Unit	Value
`m_rotor`	g	45
`r_rot`	mm	2
`R_maj_rotor`	mm	26.5
`R_min_rotor`	mm	3.25
`m_frame`	g	23
`h_rod`	mm	88
`H_rot`	mm	60
`R_sph`	mm	2.25
`R_maj_v_tor`	mm	31.75
`R_maj_H_tor`	mm	34.5
`R_min_v_tor`	mm	1.75
`R_min_H_tor`	mm	1

Table 1: Measurement results on sections of the gyroscope

Frame Assignment

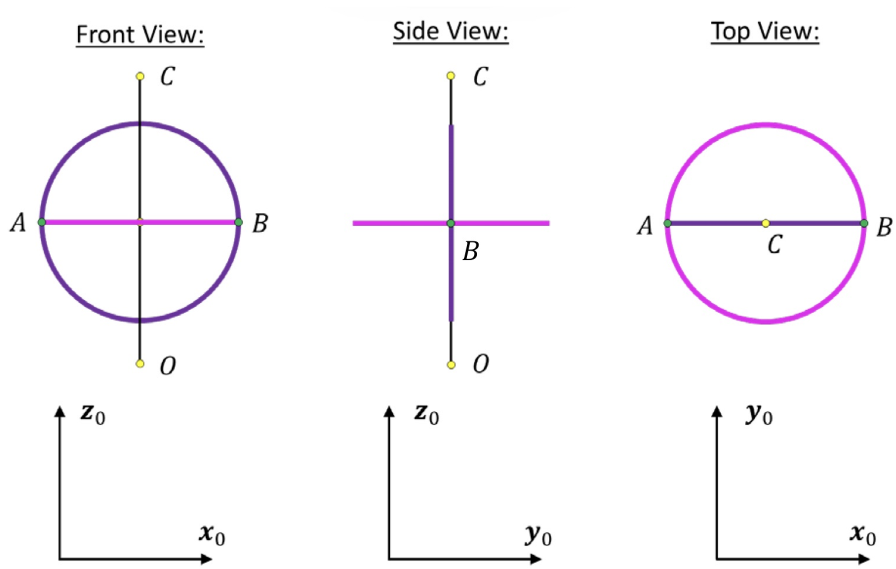


Figure 3: Initial frame allocation of the gyroscope frame

As seen in Figure 3, the frame assignment is done so that the z-axis goes up the gyroscope's vertical axis OC, and the x-axis lying parallel to the horizontal ring's span (axis AB). This it set to be initial frame, or frame $\{0\}$.

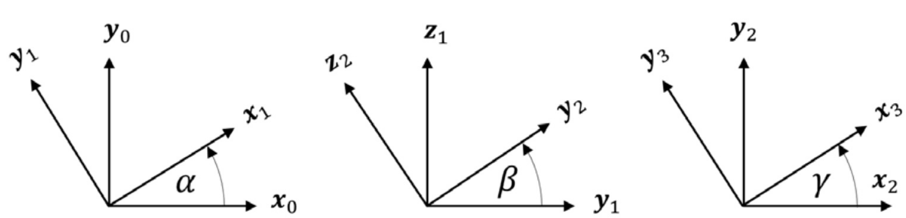


Figure 4: Frame assignment for different rotations of the gyroscope

In the frame assignment, the first rotation leading to frame $\{1\}$ will be α - the whole gyroscope's spinning around OC - the vertical axis. The second rotation states how much this vertical axis is tipping over, denoted as β, and leads to frame $\{2\}$. And as the rotor is rotating independently of the frame, that rotation would also be assigned to the next frame $\{3\}$, with the rotation labelled as γ.

Gyroscope's Gyroscopic Effect

As explained, gyroscopic effect is caused by a torque between the normal and the gravitational force. With the normal being the pivot and gravity pointing downwards, the resultant torque will always head anti-clockwise around the normal according to the right-hand rule.

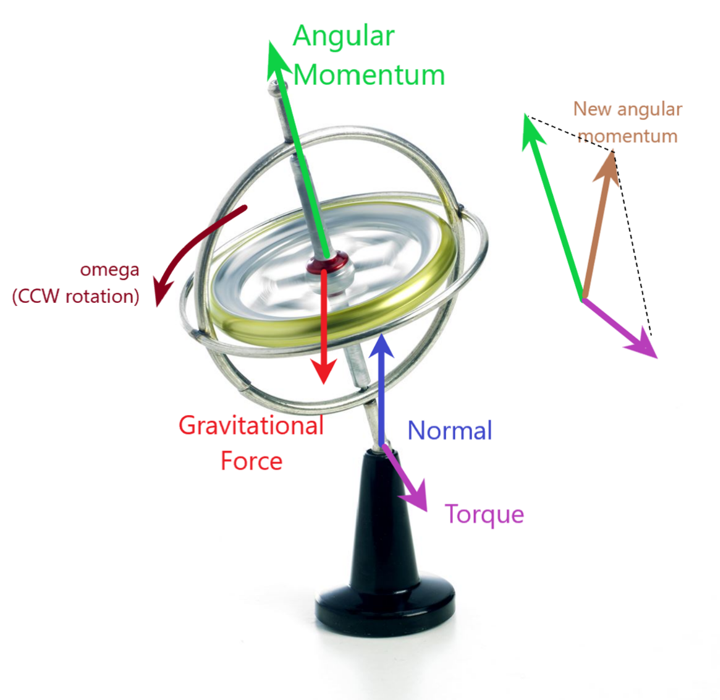


Figure 5: Demonstration of how the new angular momentum is created for a counter-clockwise rotor. Clockwise rotor will be the same, with angular momentum pointing downwards instead.

When the rotor spins, the spinning direction will decide if the angular momentum will be directing up or down along the rotation axis. This momentum, adding up with the additional angular momentum created by the torque, will create a new momentum which the axis will try to catch up. As the torque direction always heading anti-clockwise, we will know that the new momentum will be away from the old one with the same direction as the rotor's spinning direction. Therefore, the precession direction of the frame will be the same to that of rotor's.

Objectives

With the system defined, this experiment will attempt to achieve the following:

- Work out the 4 Newton-Euler equations of motion that defines the gyroscope dynamics.
- Define the initial conditions of this system.
- Using MATLAB to input the equations with initial conditions, and turn it to a simulation that resembles the real-life video.
- Evaluate the efficiency of this simulation model, and work out potentials of this experiment process.

Stage 1

Set Up & Modelling

1.1 Assumptions of System & Behaviour

The system is assumed to have no friction. We model the gyroscope as a system of two rigid bodies (frame, rotor). Only simple torus and cylinder shapes are used (Figure 1.1) with the measurements as shown in Table 1. The frame (as shown in Figure 2) is composed of three welded parts- a vertical inner torus, a horizontal outer torus and vertical cylinder. In modelling, we omit the spherical sections on the top and bottom of the cylinder. The rotor rod is assumed to be part of the frame. This means the rotor rod mass (rotor density multiplied by rotor rod volume) has to be added to the original measured frame mass to get the new `m_frame`. In the real gyroscope, the rotor rod is actually part of the rotor and has height `H_rot` so there are two revolute joints. However, we assume the rotor is simply a 'floating' torus (connected to its joints with massless links) with a single revolute joint at the centre of the cylinder G, and there is no rotor 'webbing'. We multiply the rotor density with the rotor torus volume to get the new mass of the rotor `m_rotor`. Figure 1.1 shows the gyroscope model where the frame consists of the red, blue and gold parts, and the rotor is the separate green part. Notice the blue torus and gold cylinder intersect at two locations so there is some 'doubling up' of the mass and volume. The bottom of the cylinder O is assumed to be a 3 degree-of-freedom joint (α, β, γ) and the Rotor adds 1 degree-of-freedom (δ).

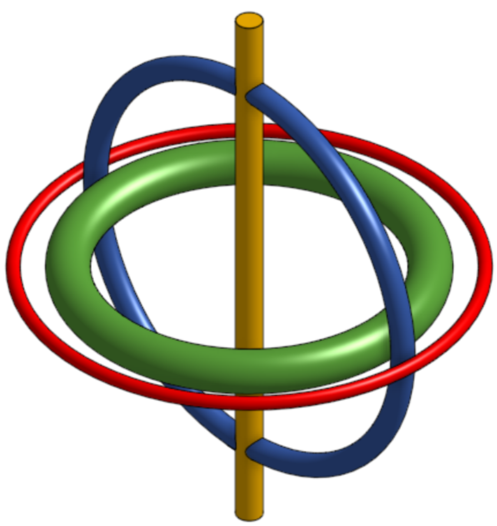


Figure 1.1: Model of the gyroscope

1.2 Equations of Motion

1.2.1 Inertia tensors

We get the inertia tensors from modifying the standard forms usually found in textbooks. The inertia tensor of the horizontal torus about its centre of mass G in frame 3 is

$$ {}^3\mathbf{I}^G_{tor,H} = m_{tor,H} \begin{bmatrix} \frac{5}{8}R^2_{min,tor,H} + \frac{1}{2}R^2_{maj,tor,H} & 0 & 0 \\ 0 & \frac{5}{8}R^2_{min,tor,H} + \frac{1}{2}R^2_{maj,tor,H} & 0 \\ 0 & 0 & \frac{3}{4}R^2_{min,tor,H} + R^2_{maj,tor,H} \end{bmatrix}. $$

Similarly, the inertia tensor of the vertical torus is

$$ {}^3\mathbf{I}^G_{tor,V} = m_{tor,V} \begin{bmatrix} \frac{5}{8}R^2_{min,tor,V} + \frac{1}{2}R^2_{maj,tor,V} & 0 & 0 \\ 0 & \frac{3}{4}R^2_{min,tor,V} + R^2_{maj,tor,V} & 0 \\ 0 & 0 & \frac{5}{8}R^2_{min,tor,V} + \frac{1}{2}R^2_{maj,tor,V} \end{bmatrix}. $$

The inertia tensor of the vertical cylinder rod is

$$^3\mathbf{I}^G_{rod} = m_{rod} \begin{bmatrix} \frac{3r_{rot}^2 + h_{rod}^2}{12} & 0 & 0 \\ 0 & \frac{3r_{rot}^2 + h_{rod}^2}{12} & 0 \\ 0 & 0 & \frac{r_{rot}^2}{2} \end{bmatrix}.$$

Using the superposition principle, we can sum these up to get the total inertia tensor of the frame about G in frame 3

$$^3\mathbf{I}^G_{frame} = {}^3\mathbf{I}^G_{tor,H} + {}^3\mathbf{I}^G_{tor,V} + {}^3\mathbf{I}^G_{rod}$$

The inertia tensor of the rotor about G in frame 4 is

$$^4\mathbf{I}^G_{rotor} = m_{rotor} \begin{bmatrix} \frac{5}{8}R^2_{min,rotor} + \frac{1}{2}R^2_{maj,rotor} & 0 & 0 \\ 0 & \frac{5}{8}R^2_{min,rotor} + \frac{1}{2}R^2_{maj,rotor} & 0 \\ 0 & 0 & \frac{3}{4}R^2_{min,rotor} + R^2_{maj,rotor} \end{bmatrix}.$$

Note that the masses $m_{tor,H}$, $m_{tor,V}$, m_{rod}, m_{rotor} are calculated from multiplying the corresponding density with the corresponding volume of the object (e.g. m_{rotor} = volume of the rotor torus $\times$ density of the rotor).

1.2.2 Rotation matrices

Using Figure 4, we obtain the rotation matrices

$${}^1_0\mathbf{R} = \begin{bmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^2_1\mathbf{R} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\beta & \sin\beta \\ 0 & -\sin\beta & \cos\beta \end{bmatrix}$$

$${}^3_2\mathbf{R} = \begin{bmatrix} \cos\gamma & \sin\gamma & 0 \\ -\sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$${}^4_3\mathbf{R} = \begin{bmatrix} \cos\delta & \sin\delta & 0 \\ -\sin\delta & \cos\delta & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

So the rotation matrix from 4 to 0 is

$${}^0_4\mathbf{R} = {}^1_0\mathbf{R}^{\mathrm{T}}\,{}^2_1\mathbf{R}^{\mathrm{T}}\,{}^3_2\mathbf{R}^{\mathrm{T}}\,{}^4_3\mathbf{R}^{\mathrm{T}}.$$

1.2.3 Kinematic quantities

The kinematic quantities for the frame are:

$$^3\mathbf{r}_{OG} = \begin{bmatrix} 0 \\ 0 \\ L \end{bmatrix}, {}^4\mathbf{r}_{OG} = {}^4_3\mathbf{R}\,{}^3\mathbf{r}_{OG}, {}^1\omega_1 = \begin{bmatrix} 0 \\ 0 \\ \dot{\alpha} \end{bmatrix}, {}^2\omega_{21} = \begin{bmatrix} \dot{\beta} \\ 0 \\ 0 \end{bmatrix}, {}^3\omega_{32} = \begin{bmatrix} 0 \\ 0 \\ \dot{\gamma} \end{bmatrix}, {}^4\omega_{43} = \begin{bmatrix} 0 \\ 0 \\ \dot{\delta} \end{bmatrix}$$

$$^3\omega_3 = {}^3_2\mathbf{R}\,{}^2_1\mathbf{R}\,{}^1\omega_1 + {}^3_2\mathbf{R}\,{}^2\omega_{21} + {}^3\omega_{32}$$

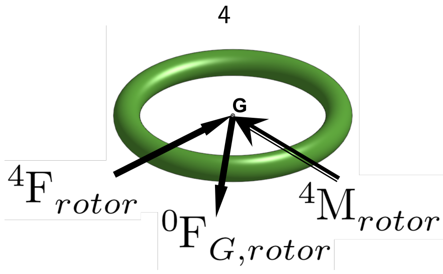
$$^3\dot{\mathbf{r}}_{OG} = {}^3\mathbf{r}'_{OG} + {}^3\omega_3 \times {}^3\mathbf{r}_{OG}$$

The kinematic quantities for the rotor are:

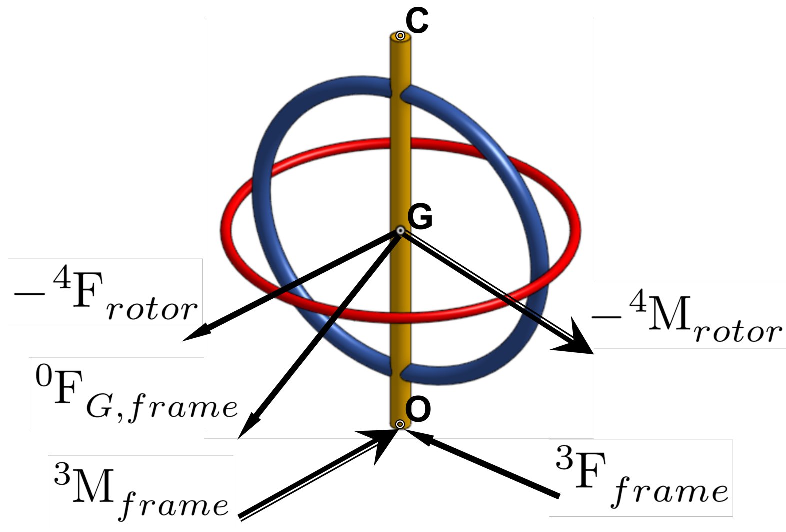
$$^4\omega_4 = {}^4_3\mathbf{R}\,{}^3\omega_3 + {}^4\omega_{43}$$

$$^4\dot{\mathbf{r}}_{OG} = {}^4\mathbf{r}'_{OG} + {}^4\omega_4 \times {}^4\mathbf{r}_{OG}$$

$$^4\ddot{\mathbf{r}}_{OG} = {}^4\dot{\mathbf{r}}'_{OG} + {}^4\omega_4 \times {}^4\dot{\mathbf{r}}_{OG}$$



(a) FBD of Rotor



(b) FBD of Frame

Figure 1.2: Free Body Diagrams (FBDs) of Rotor & Frame

1.2.4 Newton-Euler Equations

Finally, we derive the Newton-Euler (NE) equations from the Free Body Diagrams (FBDs) of the rotor and frame. Note that the Frame is attached to frame 3 and the Rotor is attached to Frame 4. ${}^{0}\mathbf{F}_{G,rotor}$ & ${}^{0}\mathbf{F}_{G,frame}$ are the weight forces in frame 0 of the Rotor & Frame respectively.
Starting from the FBD of the Rotor in Figure 1.2a, the 'linear' NE equation is

$$
\begin{aligned}
{}^{4}\sum \mathbf{F}_{rotor} &= {}^{4}\mathbf{F}_{rotor} + {}^{4}_{0}\mathbf{R}\,{}^{0}\mathbf{F}_{G,rotor} = m_{rotor}\,{}^{4}\ddot{\mathbf{r}}_{\mathbf{OG}} \\
\longrightarrow {}^{4}\mathbf{F}_{rotor} &= \begin{bmatrix} F_{G,x} \\ F_{G,y} \\ F_{G,z} \end{bmatrix} = m_{rotor}\,{}^{4}\ddot{\mathbf{r}}_{\mathbf{OG}} - {}^{4}_{0}\mathbf{R}\,{}^{0}\mathbf{F}_{G,rotor}
\end{aligned} \tag{1}
$$

where ${}^{4}\mathbf{F}_{rotor}$ is the force applied by the Frame on the Rotor at point G, expressed in frame 4.

Similarly, the 'angular' NE equation of the Rotor is

$$
{}^{4}\mathbf{M}_{rotor} = {}^{4}\dot{\mathbf{h}}^{G}_{rotor} = {}^{4}\dot{\mathbf{h}}^{G}_{rotor}{}' + {}^{4}\omega_{4} \times {}^{4}\mathbf{h}^{G}_{rotor} = ({}^{4}\mathbf{I}^{G}_{rotor}\,{}^{4}\omega_{4})' + {}^{4}\omega_{4} \times ({}^{4}\mathbf{I}^{G}_{rotor}\,{}^{4}\omega_{4}) \tag{2}
$$

where ${}^{4}\mathbf{M}_{rotor} = \begin{bmatrix} M_{G,x} \\ M_{G,y} \\ 0 \end{bmatrix}$ is the moment applied by the Frame on the Rotor at point G expressed in frame 4, since $M_{G,z} = 0$ due to no applied torque about the z rotational degree of freedom. **The third component will give us our first equation of motion.**

Now moving on to the FBD of the Frame in Figure 1.2b, the 'linear' NE equation is

$$
{}^{3}\mathbf{F}_{frame} = \begin{bmatrix} F_{O,x} \\ F_{O,y} \\ F_{O,z} \end{bmatrix} = m_{frame}\,{}^{3}\ddot{\mathbf{r}}_{\mathbf{OG}} - {}^{3}_{0}\mathbf{R}\,{}^{0}\mathbf{F}_{G,frame} + {}^{3}_{4}\mathbf{R}\,{}^{4}\mathbf{F}_{rotor} \tag{3}
$$

where ${}^{3}\mathbf{F}_{frame}$ is the force applied by the ground on the Frame at point O, expressed in frame 3.

The 'angular' equation for the Frame is

$$\begin{aligned}{}^3\mathbf{M}_{frame} &= {}^3\dot{\mathbf{h}}^G_{frame} + {}^3_4\mathbf{R}\,{}^4\mathbf{M}_{rotor} + {}^3\mathbf{r}_{OG} \times {}^3\mathbf{F}_{frame} \\ &= ({}^3\mathbf{I}^G_{frame}\,{}^3\omega_3)' + {}^3\omega_3 \times ({}^3\mathbf{I}^G_{frame}\,{}^3\omega_3) + {}^3_4\mathbf{R}\,{}^4\mathbf{M}_{rotor} + {}^3\mathbf{r}_{OG} \times {}^3\mathbf{F}_{frame}\end{aligned} \tag{4}$$

where ${}^3\mathbf{M}_{frame} = \begin{bmatrix} M_{O,x} \\ M_{O,y} \\ M_{O,z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$ is the moment applied by the ground on the Frame at point O expressed in frame 3 as there is no applied torque about each of the three rotational degrees of freedom. **All three components give us the final three equations of motion.**

These equations of motion will be in the following format:

$$\begin{aligned} 0 &= g_1(\alpha, \beta, \gamma, \delta, \dot{\alpha}, \dot{\beta}, \dot{\gamma}, \dot{\delta}, \ddot{\alpha}, \ddot{\beta}, \ddot{\gamma}, \ddot{\delta}) \\ 0 &= g_2(\alpha, \beta, \gamma, \delta, \dot{\alpha}, \dot{\beta}, \dot{\gamma}, \dot{\delta}, \ddot{\alpha}, \ddot{\beta}, \ddot{\gamma}, \ddot{\delta}) \\ 0 &= g_3(\alpha, \beta, \gamma, \delta, \dot{\alpha}, \dot{\beta}, \dot{\gamma}, \dot{\delta}, \ddot{\alpha}, \ddot{\beta}, \ddot{\gamma}, \ddot{\delta}) \\ 0 &= g_4(\alpha, \beta, \gamma, \delta, \dot{\alpha}, \dot{\beta}, \dot{\gamma}, \dot{\delta}, \ddot{\alpha}, \ddot{\beta}, \ddot{\gamma}, \ddot{\delta}) \end{aligned}$$

We can use the Matlab `equationsToMatrix` function to decouple the equations of motion from $\ddot{\alpha}, \ddot{\beta}, \ddot{\gamma}, \ddot{\delta}$, making sure to eliminate the unknown components of constraint forces and moments.

$$\begin{aligned} \ddot{\alpha} &= h_1(\alpha, \beta, \gamma, \delta, \dot{\alpha}, \dot{\beta}, \dot{\gamma}, \dot{\delta}) \\ \ddot{\beta} &= h_2(\alpha, \beta, \gamma, \delta, \dot{\alpha}, \dot{\beta}, \dot{\gamma}, \dot{\delta}) \\ \ddot{\gamma} &= h_3(\alpha, \beta, \gamma, \delta, \dot{\alpha}, \dot{\beta}, \dot{\gamma}, \dot{\delta}) \\ \ddot{\delta} &= h_4(\alpha, \beta, \gamma, \delta, \dot{\alpha}, \dot{\beta}, \dot{\gamma}, \dot{\delta}) \end{aligned}$$

This allows us to prepare a set of 8 first order ODE in $\dot{X}$ for numerical integration & simulation.

$$\mathbf{X} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta \\ \gamma \\ \delta \\ \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \\ \dot{\delta} \end{bmatrix}$$

$$\dot{\mathbf{X}} = \begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \\ \dot{x_5} \\ \dot{x_6} \\ \dot{x_7} \\ \dot{x_8} \end{bmatrix} = \begin{bmatrix} \dot{\alpha} \\ \dot{\beta} \\ \dot{\gamma} \\ \dot{\delta} \\ h_1(x_1, x_2, ..., x_8) \\ h_2(x_1, x_2, ..., x_8) \\ h_3(x_1, x_2, ..., x_8) \\ h_4(x_1, x_2, ..., x_8) \end{bmatrix}$$

Stage 2

Conditions tuning

2.1 Initial Conditions Tuning

The tuning parameters were through visual observation from the videos where some of the initial conditions can be identified. However, an initial understanding of each variable effects is needed. From then a sets of constraint were developed from what we've understand about the effect of each individual variables on the modelled system. Therefore, through trial and error of the remaining variables ensuring that all constraints are satisfied, we were able to obtain a set of initial conditions for the simulation shown in the Table 2.1

2.2 Investigate Initial Conditions effects

The state space equation were acquired using Matlab `ode45` solver for every time step of 0.03 seconds over the span of 7 seconds. In order, to choose suitable initial conditions, it is important to understand the effects of each parameters on the system. Through, experimentation of Gyroscope animation, it was determined that the change of initial angles values will result in change of gyroscope starting position in the plane of the inertia frame (x_0, y_0, z_0).

Figure 2.1 shows the effects of each angles variable initial condition on the gyroscope. α angle shown in figure 2.1a describe gyroscope rotation around inertia frame z_0 in x_0 and y_0 plane. β angles determine the gyroscope starting position relative to the x_0 and y_0 plane (out of plane rotation). The figure 2.1c shows that γ determines the gyroscope about its own central z_3 axis. Lastly, the angle δ describe the rotor starting position and does create observable effects on the system dynamics.

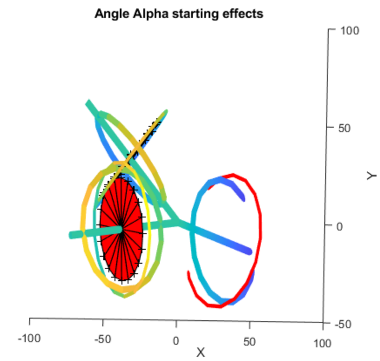
The effects of the angular velocities were also investigated to provide a better understanding of the system. The values $\dot{\alpha}$ affects the gyroscope rotation in the x_0 and y_0 plane whilst, $\dot{\beta}$ affects the out of plane rotation around the gyroscope, not tuning $\dot{\beta}$ properly caused the system to become unstable and gyroscope frame oscillates in and out of $x_0 - y_0$ plane. The angular velocity $\dot{\gamma}$ determines the the velocity of the frame around the central axis. The rotor angular velocity $\dot{\delta}$ determines the speed in which the rotor rotate, this was observed to affects gyroscope in various ways. If the values, is low than the gyroscope would be unstable and would fall; however, if the values is too high the system would be too stable hence the models would not match the recorded videos in the time span. The rotation of the gyroscope simulation (Clockwise and Counter - Clockwise) will be determined by the sign values of both $\dot{\gamma}$ and $\dot{\delta}$, same sign values of $\dot{\gamma}$ and $\dot{\delta}$ will have opposite rotation direction.

2.2.1 Conditions Constraints

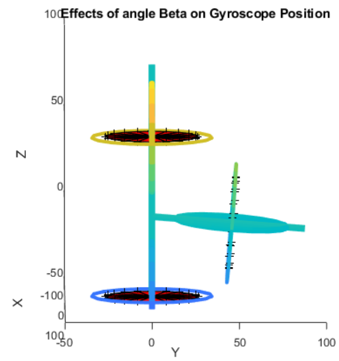
After, initial conditions effects have been investigated conditions constraint was developed as a initial range to assist in the tuning process of the variable for a relative stable gyroscope. Understanding of the models behaviours and observable behaviours from videos were used to developed the constraint as follows:

- $\alpha \approx \frac{\pi}{2}$ (1)
- $\beta \approx 0$ (2)
- $\delta = 0$ (3)
- $\dot{\beta} = 0$ (4)
- CW: $\dot{\delta} > 100$ and for CCW: $\dot{\delta} < -100$ (5)
- $\dot{\delta}$ and $\dot{\gamma}$ has to be the same sign (6)

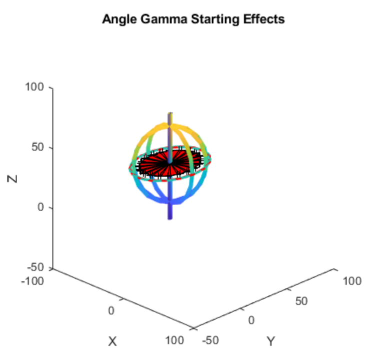
The first 3 initial conditions constraints depends on the position of the gyroscope based on the initial condition in the videos recorded. Both α and β constraint has taken account that the gyroscope initial position is approximately perpendicular to the base. The 4th initial constraints was derived from the investigation of the $\dot{\beta}$ in the previous section, setting it to zero would eliminate the 'wobbling' effect of gyroscope rotating in and out of $x_0 - y_0$ plane.



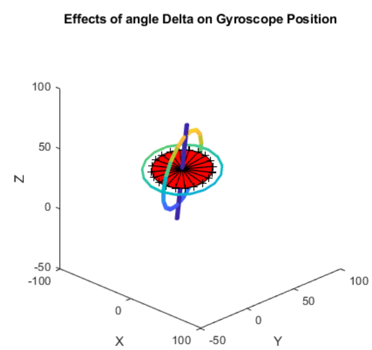
(a) Alpha Initial condition



(b) Beta Initial condition



(c) Gamma Initial condition



(d) Delta Initial condition

Figure 2.1: Angle Initial condition

During Matlab experimentation, it was observed that for rotor to rotate it must satisfied the constraint (5) as any values outside any range would cause the system to fail. Furthermore, from the experimentation it is noted that $\dot{\delta}$ and $\dot{\gamma}$ has to be the same sign.

View	α	β	γ	δ	$\dot{\alpha}$	$\dot{\beta}$	$\dot{\gamma}$	$\dot{\delta}$
CCW Top	$\frac{96\pi}{180}$	$\frac{4\pi}{180}$	$\frac{45\pi}{180}$	$\frac{0.01\pi}{180}$	0.01	0	4	400
CW Top	$\frac{90\pi}{180}$	$\frac{3\pi}{180}$	$\frac{45\pi}{180}$	$\frac{360\pi}{180}$	2	0	-7.13	-400
CCW Front	$\frac{96\pi}{180}$	$\frac{4\pi}{180}$	$\frac{45\pi}{180}$	$\frac{0.01\pi}{180}$	0.01	0	4	400
CCW Isometric	$\frac{96\pi}{180}$	$\frac{4\pi}{180}$	$\frac{45\pi}{180}$	$\frac{0.01\pi}{180}$	0.01	0	6.13	500

Table 2.1: Initial Conditions Table

Stage 3

Results & Analysis

3.1 Simulation Results

The resulting MATLAB simulation seems to perform well and closely describe the system behaviour. We can see that it manages to correctly show the rotor spinning, and the precession behaviour of the rotation axis, which goes around the normal vector: it goes counter-clockwise if the rotor is spinning counter-clockwise, and vice versa, it goes clockwise if the rotor goes clockwise.

3.2 Evaluation

The gyroscope was able to looks similar to the recorded videos is due to the experimentation process and initial condition tuning methods mentioned above. With some initial assumption of some of the initial condition based on what was observed in the videos. This has led to the animated behaviours has similar initial condition to the videos. The modeling process of abstracting and deriving the equations of motions was correct and all of the majors forces and moments that is acting on the gyroscope has been identified and it's effect has been taken into account when the equations of motions. Therefore, the animation was able to capture most of the real-life gyroscope model general characteristics and behaviour.

However, there exist some differences between the animated and real live models, especially as modelling time increases the error between the 2 models grow larger. Others factors that the animated model have not taken into account includes frictions in some of the components when the gyroscope spins (notably, friction between ball joints and base), air resistance. These factors, would led to energy loss in real live models as time goes on however, the animated models will spin forever as no energy are loss.

Another differences is the assumption of the some gyroscope geometry during the modelling process to simplified the problem. For example, as mentioned above the difference in the rod assumption between modelling and real-live model. The geometrical difference would impact the kinematics and dynamics calculation in the modelling process hence would cause different rotating behaviour between the rotor and the frame.

These discrepancies have resulted in different precession rates.

The gyroscopes used in real applications would be optimised to keep spinning longer and faster. It is common to use gyroscopes that are Micro Electro-Mechanical Sensors (MEMS). These do not have any moving parts.

3.3 Conclusion

Overall, the modelling process has a reasonable performance. It has effectively demonstrated the gyroscopic behavior, which consists of the rotation axis not tipping over, and its precession rotates the same direction as the rotor. Improvements for the accuracy are possible for this process and can be exploited if further time and resources allows, such as more precise initial speed measurement or more detailed calculations on separate parts rather than assuming them as a rigid body. Simulation results like these will be meaningful for real world applications such as sensor drift calibrating, turbines and flywheel energy storage.

Bibliography

[1] Tomáš Náhlík, , and Dana Smetanová. Applications of gyroscopic effect in transportation. *Naše more*, 65(4):293–296, October 2018.

[2] Vittorio M. N. Passaro, Antonello Cuccovillo, Lorenzo Vaiani, Martino De Carlo, and Carlo Edoardo Campanella. Gyroscope technology and applications: A review in the industrial perspective. *Sensors*, 17(10):2284, October 2017.

[3] David Halliday, Robert Resnick, and Jearl Walker. *Fundamentals of physics*. John Wiley & Sons, 2013.

[4] Nathaniel H Bowditch. *American practical navigator, the*. Paradise Cay Publications, Middletown, CA, November 2004.

[5] *The Anschutz gyro-compass and gyroscope engineering*. Wexford College Press, January 2003.

[6] Uğursal Demir. Use of anti rolling gyro (arg) optimization systems for planing boats, 2020.

[7] Nguyen Cong Danh. The stability of a two-axis gimbal system for the camera. *The Scientific World Journal*, 2021:1–8, May 2021.

Appendix A

Additional Resources

A.1 MATLAB Code Operation

The code is split into 4 different files. Noted that all files has to be in the same directory.

- $Labequation.mlx$
- $state\_space.m$
- $rotation.m$
- $Gyro\_animation.m$
- $Rotor\_Default\_sldprt\_new.STL$

The live script $Labequation.mlx$ will be use to obtain the Equation of Motion and the decouple equations for $\dot{\alpha}$, $\dot{\beta}$, $\dot{\gamma}$ the final variable `EOM` is a 4x1 matrix containing these decouple equations.

Copy and paste the output of variable `EOM` into $state\_space.m$ for the function to be able to calculate state_space model. Ensure that the variable `EOM` is a valid 4x1 matrix declared in the function.

Run the `Gyro_animation.m` file to plot the gyro animation, this videos is recorded for 10 seconds. To plot different view please uncomment the associated initial condition `X_init` and associated views.

Initial Conditions will be from Line 13 to 21 of the Matlab files; views condition will be from Line 149. All initial condition and views have been commented with its effects.

The current default view of the animation is `Isometric` with the default Initial Condition being `Counter_Clockwise` for `Isometric`. Please uncomment for new Initial Condition and comment out the old condition.

$Rotor\_Default\_sldprt\_new.STL$ file contain the rotor step file to plot the gyro animation.