Dynamics of Axially Coupled Rotor Pairs

Yuelong Li

October 2022

Abstract

In this paper I examine through a combination of theoretical derivations and numerical simulations the dynamics of an axially coupled rotor pair, with a structure quite analogous to a localized pair of particles with spins. In the section of preliminary analysis, I used the assumption of parallel motions between the two rotors, which is sufficient for arriving at dynamics of rotor pairs with the same "spin". In the section of rigid body simulation, I go over the underlying schemes of a generic rigid body simulator I created in Matlab. **In the last two sections**, I provide a detailed analysis of the behaviors seen in the simulations of the rotor pairs, and manage to derive an explicit formula describing the rotor pair dynamics when the spins are opposite, using Lagrangian dynamics, and verified the theory with the experimental results. Generalization of this theory may illuminate the dynamics in neutron scattering or phonon propagation through spin coupling.

Contents

1 Introduction

While the idea of using gyroscopic effects for stabilization is not new, which we see in navigation instruments and in bikes, it is not very commonly used in large-scale structural stabilization, such as in architectures. More often tuned mass dampers that rely on principles of harmonic oscillators are used. In this paper, I hope to investigate the potential of using gyroscopic rotors for structural stabilization, by considering axially coupled rotor pairs.

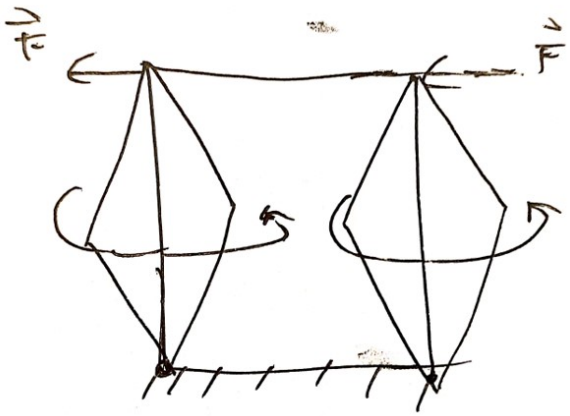


Figure 1: Axially coupled rotor

The key is to replace vertical supports of a structure with vertical rotors. Then the system should display a resistance in response to external shocks in that direction. To begin with, we will consider the case of two rotors grounded at the bottom and joined together on top by a horizontal axis. When subjected to external torque, if the rotors have the same rotational speed, the precession induced will be in the same direction, and there will be a permanent deformation of the structure after the shock. Thus this paper will put a particular focus on rotor pairs with opposing angular momenta, which turns out to display some very non-trivial dynamics.

The dynamics of systems with many degrees of freedom like the axially coupled rotor pairs can get quite complicated. Thus this paper will utilize a combination of rigid body simulation and physics derivations to perform the analysis. A general purpose rigid body simulator is built in MATLAB with key ideas taken from Special Topics: Modeling and Simulation in Science, Engineering, and Economics [2]. The code for rigid body rotational matrix is taken directly from programs written by Professor Charles S. Peskin, as well as the code for generating network of vertices representing wheels of various geometries.

2 Preliminary analysis

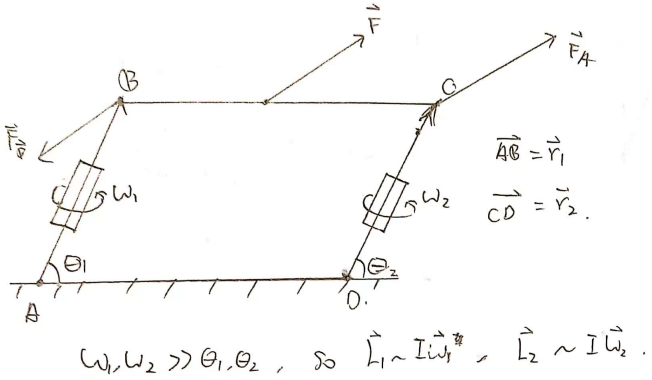


Figure 2: Rotor forces

As the diagram is showing, rotor 1 is on the left spinning at rate ω_1, rotor 2 is spinning at ω_2. From the setup diagram, we can see that $\vec{r}_1$ and $\vec{r}_2$ stays mostly parallel, so we will make this approximation: $\vec{r} = \vec{r}_1 = \vec{r}_2$ (**this assumption turns out to be problematic, more on this later**). The total external force acting on the system is $\vec{F}$. Then assuming the link in the middle is massless, the force acting on rotor 1 $\vec{F}_A$ and rotor 2 $\vec{F}_B$ should satisfy:

$$\vec{F}_A + \vec{F}_B = \vec{F}.$$

There are no other forces than $\vec{F}_A$ acting on rotor 1 and no other forces other than $\vec{F}_B$ acting on rotor 2, so we can consider their dynamics in separation from one other in terms of only $\vec{F}_A$ and $\vec{F}_B$. In high rotational speed limit, the angular momentum due to the rotation of the rotor axis, namely $\omega_1, \omega_2 >> \theta_1, \theta_2$. So the angular momentum of the left and right rotor are

$$\vec{L}_1 = I\vec{\omega}_1, \tag{1}$$

$$\vec{L}_2 = I\vec{\omega}_2. \tag{2}$$

Then:

$$\vec{r} \times \vec{F}_A = \frac{d\vec{L}_1}{dt} = \frac{d}{dt}(\omega_1 I\hat{r}) = \omega_1 I\dot{\hat{r}}, \tag{3}$$

$$\vec{r} \times \vec{F}_B = \frac{d\vec{L}_2}{dt} = \frac{d}{dt}(\omega_2 I\hat{r}) = \omega_2 I\dot{\hat{r}}. \tag{4}$$

To use the symmetry between $\vec{F}_A$ and $\vec{F}_B$, we have:

$\vec{F} = \vec{F}_1 + \vec{F}_2$.

Define $\vec{f} = \frac{\vec{F}}{2} - \vec{F}_B$. then:

$\vec{f} = \frac{\vec{F} - \vec{F}_B}{2} - \frac{\vec{F}_B}{2} = \frac{\vec{F}_A}{2} - \frac{\vec{F}_B}{2} = \vec{F}_A - \frac{\vec{F}}{2}$.

So $\vec{F}_A = \frac{\vec{F}}{2} + \vec{f}$, $\vec{F}_B = \frac{\vec{F}}{2} - \vec{f}$.

So we can get:

$$\omega_1 I\dot{\hat{r}} = \vec{r} \times \left(\frac{\vec{F}}{2} + \vec{f}\right) = \frac{\omega_1}{\omega_2}(\omega_2 I\dot{\hat{r}}) = \frac{\omega_1}{\omega_2}\vec{r} \times \left(\frac{\vec{F}}{2} - \vec{f}\right).$$

Giving:

$$\vec{r} \times \frac{\vec{F}}{2} - \frac{\omega_1}{\omega_2}\vec{r} \times \frac{\vec{F}}{2} = \vec{r} \times \vec{f} + \frac{\omega_1}{\omega_2}\vec{r} \times \vec{f}.$$

Then we can re-express $\vec{r} \times \vec{f}$ in terms of $\vec{r} \times \vec{F}$:

$$\vec{r} \times \vec{f} = \frac{1 - \frac{\omega_1}{\omega_2}}{2\left(1 + \frac{\omega_1}{\omega_2}\right)}\vec{r} \times \vec{F}. \tag{5}$$

Plugging this result back to (3) gives us:

$$\omega_1 I\dot{\hat{r}} = \vec{r} \times \left(\frac{\vec{F}}{2} + \vec{f}\right) = \vec{r} \times \frac{\vec{F}}{2} + \frac{1 - \frac{\omega_1}{\omega_2}}{2\left(1 + \frac{\omega_1}{\omega_2}\right)}\vec{r} \times \vec{F}.$$

Then

$$\dot{\hat{r}} = \frac{1}{\omega_1 I}\vec{r} \times \left(\frac{1 + \frac{\omega_1}{\omega_2} - \frac{\omega_1}{\omega_2} + 1}{1 + \frac{\omega_1}{\omega_2}}\right)\frac{\vec{F}}{2} \tag{6}$$

$$= \frac{\vec{r} \times \vec{F}}{(\omega_1 + \omega_2)\, I}. \tag{7}$$

This is exactly the same equation for the case of a single gyroscope rotating at velocity $\omega_1 + \omega_2$, and it makes sense. When $\omega_1 = \omega_2$, the system responds by distributing forces evenly between the gyroscopes, so each precesses at the rate of $\frac{\vec{r} \times \vec{F}/2}{\omega I} = \frac{\vec{r} \times \vec{F}}{(\omega_1 + \omega_2) I}$. When $\omega_1 = -\omega_2$, the system will behave like the rotors are not rotating at all [1].

3 Rigid Body Simulation

For simulating various setups of the coupled rotor, I wrote general purpose, object oriented rigid body simulator in MATLAB.

[1] we can not draw this conclusion directly from equation (7), since it is derived from (1) and (2) which are approximations and correct only if the gyroscopes are massless (and somehow with non-zero rotational inertia). But it should tell us that with little force the rotor axis will rotate really fast when there is no mass, and with mass we can expect to move according Newtonian acceleration $\vec{a} = \frac{\vec{F}}{m}$.

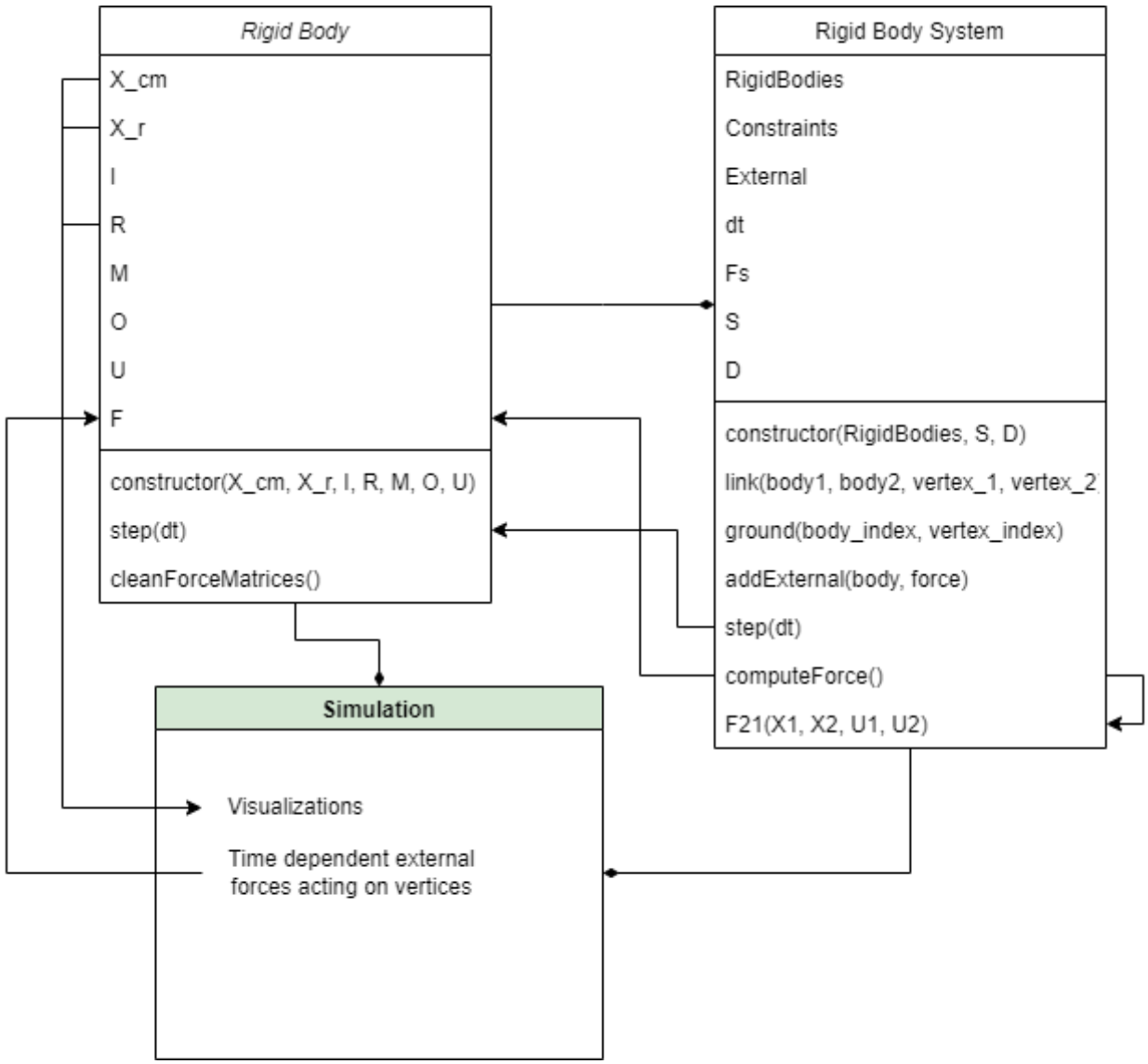


Figure 3: OOP design of rigid body simulator

On an abstract level, this simulator allows for the creation, orientation of rigid bodies to form a rigid body system. The Rigid Body System Object then allows one to specify certain interactions between these rigid bodies, such as link, where vertices of two rigid bodies are joined together, ground, where a point of the rigid body is fixed spatially, and add external force, where an external force is placed at the center of mass at a specified rigid body.

3.1 Rigid body dynamics

Due to the restriction of space, I will not give the full derivation, nor the precise definition of many of the quantities that I define below. More detailed derivations can be found at [2]. Let's first consider a system of mass particles m_i, each located at $\vec{X}_i$, with forces acting on each other satisfying $\vec{f}_{ij} = -\vec{f}_{ji}$, and subjected to external forces $\vec{F}_i$ acting on the i-th

particle. We shall define the following quantities:

$$M = \sum_i m_i.$$

$$\vec{F} = \sum_i \vec{F}_i.$$

$$\vec{X}_{cm} = \frac{1}{M} \sum_i m_i \vec{X}_i.$$

$$\vec{U}_{cm} = \dot{\vec{X}}_{cm} = \sum_i \vec{U}_i.$$

$$\vec{L} = \sum_i m_i (\vec{X}_i - \vec{X}_{cm}) \times \vec{U}_i.$$

$$\vec{\tau} = \sum_i (\vec{X}_i - \vec{X}_{cm}) \times \vec{F}_i.$$

Then, at any given moment, the following equations will be satisfied:

$$\vec{F} = M\dot{\vec{U}}_{cm},$$

$$\tau = \dot{\vec{L}}.$$

This holds in fact for any system of particles. In particular, when the geometry of the system of particles is fixed, i.e. rigid body, we shall characterize its motion by measuring the rate of rotation around its center of mass $\vec{\Omega}$, and velocity of its center of mass $\vec{U}_{cm}$. Then for the i-th particle, we can compute its velocity by:

$$\vec{U}_i = \vec{U}_{cm} + \Omega \times (\vec{X}_i - \vec{X}_{cm}).$$

Plugging this equation back into the definition of $\vec{L}$ gives:

$$\vec{L} = \left(\sum_i m_i (\mathbb{1}(\vec{X}_i - \vec{X}_{cm})^2 - (\vec{X}_i - \vec{X}_{cm})(\vec{X}_i - \vec{X}_{cm})^T) \right) \vec{\Omega}.$$

We define the matrix term in the above equation by I and call it the inertial tensor:

$$I = \left(\sum_i m_i (\mathbb{1}(\vec{X}_i - \vec{X}_{cm})^2 - (\vec{X}_i - \vec{X}_{cm})(\vec{X}_i - \vec{X}_{cm})^T) \right).$$

It is a 3-by-3 symmetric matrix, and depends on the geometry of the rigid body, as well as its orientation. Since in rigid body dynamics, the relative positions $(\vec{X}_i - \vec{X}_{cm})$ will not change other than undergo unitary transformation from its original positions, we shall denote this unitary transformation that changes with time $R(t)$. As one might expect, the inertial tensor depends on time in the following way:

$$I(t) = R^{-1}(t) I_0 R(t) = R^T(t) I_0 R(t),$$

where I_0 is the initial inertial tensor in the original basis, namely when $R = \mathbb{1}$. We get to replace the matrix inversion of R by matrix transposition without any loss of precision because it is unitary. This will have great computational advantages.

3.2 Applying to the rigid body simulation

For each of the rigid bodies, one should recognize that the set of variables: $(\vec{X}_{cm}, \{\vec{X}_i - \vec{X}_{cm}\}_{i\in[n]}, I, R, M, \vec{\Omega}, \vec{U}_{cm})$ will fully specify its state of motion at any instant. Additionally, the positions of its particles, $\{\vec{X}_i - \vec{X}_{cm}\}_{i\in[n]}$, is only relevant for the calculation of torques $\vec{\tau}$. All the other dynamical variables change in a way that don't depend on the specific positions of its particles. So we shall only include the subset of particles, which we call vertices, that are subjected to external forces. We denote the positions of this subset of particles relative to the center of mass by $\{\vec{X}_r(i)\}_{i\in S}$. So for each of the rigid body, we represent it by an object, and keep as its properties: $(\vec{X}_{cm}, \{\vec{X}_r(i)\}_{i\in S}, I, R, M, \vec{\Omega}, \vec{U}_{cm})$.

3.2.1 Euler's method

At time t, suppose that $\vec{F}_i$ is applied to the $i-th$ vertex of the rigid body. Using Euler's method, we perform the following in iterative steps of Δt:
compute $\vec{F} = \sum_i \vec{F}_i$;
compute $\vec{\tau} = \sum_i \vec{X}_r(i) \times \vec{F}_i$;
Increment first order quantities:
$\vec{L}(t+\Delta t) = \vec{L}(t) + \tau \Delta t$;
$\vec{U}(t+\Delta t) = \vec{U}(t) + \vec{F}\Delta t/M$;
$\Omega(t+\Delta t) = I^{-1}(t)\vec{L}(t+\Delta t)$;
Increment the zeroth order quantities:
$\vec{X}_{cm}(t+\Delta t) = \vec{X}_c m(t) + \vec{U}\Delta t$.
Now rate of change of the relative position of vertices follow rotation:

$$\frac{d}{dt}\vec{X}_r = \vec{X}_r \times \vec{\Omega}$$

So we may expect to write here $\vec{X}_r(t+\Delta t) = \vec{X}_r(t) + \vec{X}_r \times \vec{\Omega}\Delta t = (\mathbb{1} + \Delta t \Omega_\times)\vec{X}_r(t)$. But as it turns out, when Δt is not approaching 0, $\mathbb{1} + \Delta t\Omega_\times$ is not unitary. The precise matrix of rotation in 3D is:

$$dR = \Re(\vec{\Omega}dt) = P(\vec{\Omega}) + \cos(\|\Omega\|\Delta t)(\mathbb{1} - P(\Omega)) + \sin(\|\Delta t\|\Omega)(\hat{\Omega}_\times),$$

where $P(\vec{\Omega}) = \Omega^T\Omega$ is the projection matrix, and $\hat{\Omega}_\times$ is the cross product matrix of $\hat{\Omega}$. Then:
$\vec{X}_r(t+\Delta t) = dR\vec{X}_r(t)$;
$R(t+\Delta t) = dR \cdot R(t)$.

- In terms of implementation, the additional nuance is that vectors of a particular vertex is best stored as a row vector in MATLAB. This way, array access of indexed quantities feels more natural.
- This means in each case where a vector is involved, such as $\vec{F}_i$, it is taken to the transpose of itself when implemented. So programmatically, F(i) = $(\vec{F}_i)'$, L = $(\vec{L})'$, etc.
- All matrix quantities are still stored as is.

- In response, the matrix multiplication statements are all transposed such that:

$$\Omega = R'(t)I_0^{-1}R(t)L(t)$$

becomes

$$\mathrm{O} = (R'(t)I_0^{-1}R(t)L(t))' = L(t)'R'(t)I_0^{-1}R(t) = \text{ L*R'I*R;}$$
$$\mathrm{X\_r}(t+\Delta t) = \mathrm{X\_r}(t)\text{*dR'.}$$

- By utilizing transpose and storing in advance the inverse of the inertial matrix, we can **completely bypass matrix inversions** in the iterative steps. This is great for the program efficiency.

3.3 Rigid body system (Class)

The design of the simulator purports to hide (abstract away) any details of the rigid bodies and to allow for manipulation of the rigid bodies in an intuitive way, like we would with them in reality. Four important methods of the Rigid Body System accessible to public are:

- constructor(RigidBodies, S, D)
- addExternal(body, force)
- link(body1, body2, vertex1, vertex2)
- ground(body, vertex)

External forces are extra forces that don't generate torque but are added to the total force acting on rigid bodies. We call it $\vec{F}_c$ because it effectively correspond to forces acting on center of mass of rigid bodies. Gravity is of this kind.

The simulation of links and grounds involve the following process:

- When constraining two vertices of two rigid bodies together, we call the **link** method. The rigid body system then simulates a spring of rest length 0 that "glues" these two points together spatially:

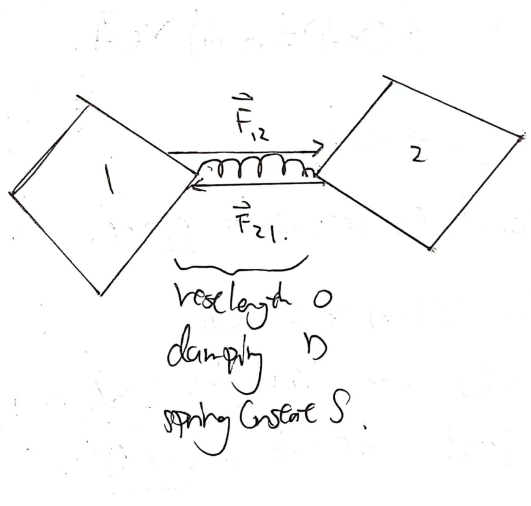


Figure 4: Spring created between vertex 1 and vertex 2

S and D are damping constant and spring constant respectively, and remains the same for all links in the rigid body system. The equations of motion follow:

$$\vec{F}_{12} = S\left(\left\|\vec{X}_1 - \vec{X}_2\right\|\right) + D(\vec{U}_1 - \vec{U}_2) \cdot \hat{e}_{21}$$

$$\vec{F}_{21} = -\vec{F}_{12}$$

These forces are then incremented into the force matrices of each of the rigid bodies, iterating over all the links of a rigid body system ensures the constraint conditions.

- When grounding a point of an object, say X_{r1}, we create a special rigid body called "ground" (G), which never undergoes any time-evolution in the step method so its vertices are fixed from creation. Then the point X_{r1} is added to ground as a vertex $G_r(1)$, and a link is created between $G_r(1)$ and $X_r(1)$:

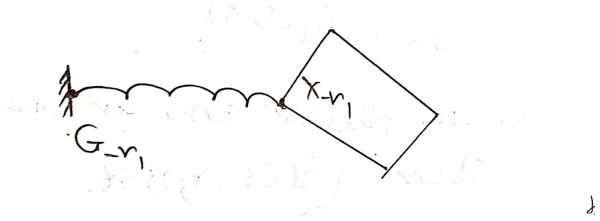


Figure 5: Link created between ground vertex 1 and body vertex 1

3.4 Visualization

Visualization of the rigid body system can be created separately. In particular, calling
[kmax,lmax,X,jj,kk,Ss,Ds,Rzero,Ms] = ...
wheel(r,a,n,M_rim,M_axle,S_rim,D_rim,S_spoke,D_spoke,S_axle,D_axle);
gives the wire frame of a wheel stored in X. Then the program uses the rotational transform R and the translational transform X_cm stored in each rigid body to transform this wireframe into the corresponding location for visualization:
Xn = X*Sys.RigidBodies(1).R'+Sys.RigidBodies(1).X_cm;

4 Modeling and Validation

4.1 Setup

Once the generic rigid body system class is built, the setting up the configuration becomes somewhat trivial. Note that to simulate the linkage between rotor 1 and rotor 2, a third rigid body is used, which is closer to what would happen when building a system like this in real life. It also has rotational inertia and mass, but has no rotation in the horizontal direction. The vertices of each of the rigid bodies can be reduced to 2, where they connect to the ground and to the link. Rigid body 1 and rigid body 2 are very similar, except center of mass of R1 is located at [0 0 -0.5] at initialization, and center of mass of R2 is located at [0 0 0.5]. The axis length (distance between two vertices) of R1, R2, and R3 are all 1. So together they form a square frame. All rigid bodies are given the identity as their inertial tensor.

Below are the diagram for the rigid body system of the axially coupled rotor pairs and the corresponding code used for initialization:

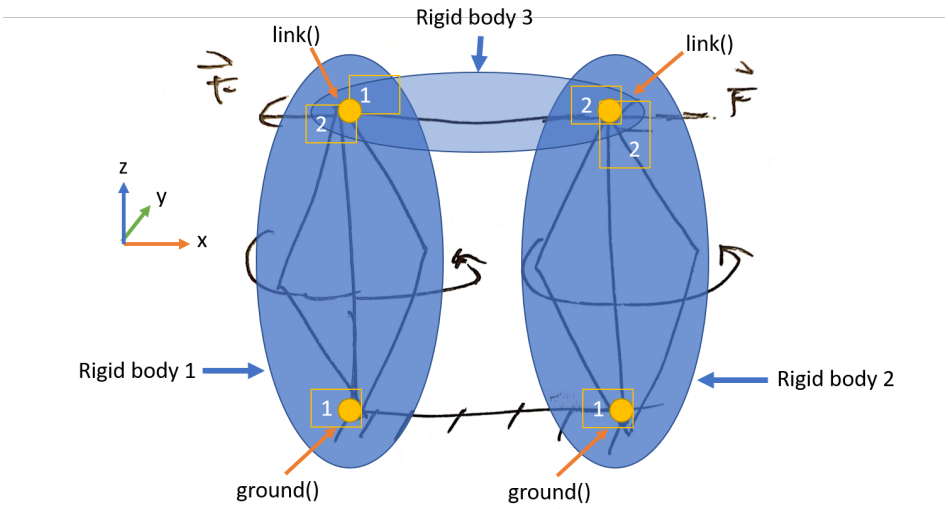


Figure 6: Axial coupled rotor pair can be modeled using 3 rigid bodies

```
S = 2000*(M/n)*omega_s^2;  %Stiffness of edges
D = 500*(M/n)*omega_s;  %Damping of edges (kg/s)

X_cm = [0 0 0];
X_r=[0 0 -0.5; 0 0 0.5];
I = [1 0 0; 0 1 0; 0 0 1];
%R = [1/2^0.5 0 -1/2^0.5; 0 1 0; 1/2^0.5 0 1/2^0.5];
R = eye(3);
M = 1;
O = [0 0 10];
U = [0 0 0];
R1 = RigidBody([-0.5 0 0], X_r, I, R, M, [0 0 10], U);
R2 = RigidBody([0.5 0 0], X_r, I, R, M, [0 0 10], U);

X_r = [-0.5 0 0; 0.5 0 0];
O = [0 0 0];
U = [0 0 0];
R3 = RigidBody([0 0 1], X_r, I, R, M, O, U);

Sys = RigidBodySystem([R1, R2, R3], S, D);
Sys.ground(2, 1);
Sys.ground(1, 1);
%Link vertex 2 of rigid body 1 to vertex 1 of rigid body 3
Sys.link(1, 3, 2, 1);
%Link vertex 2 of rigid body 2 to vertex 2 of rigid body 3
Sys.link(2, 3, 2, 2);
```

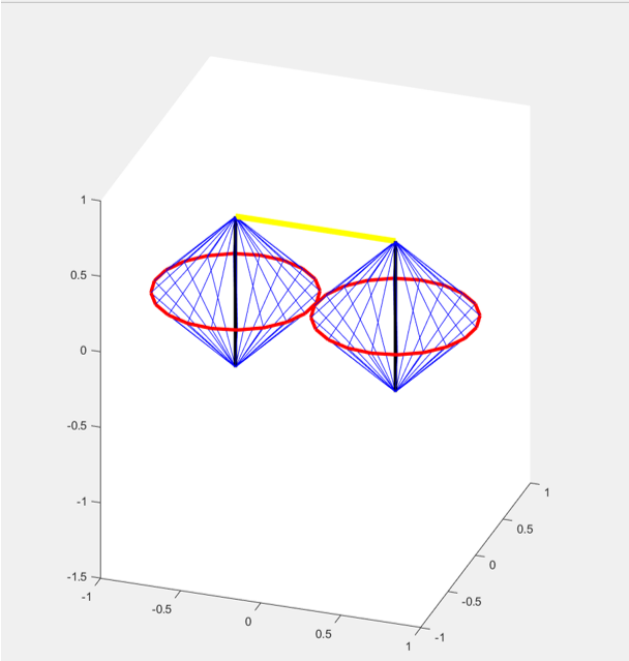


Figure 7: Simulation of coaxial rotor with same angular velocities

4.2 Simulation and Validation

We run this set up for 10 seconds using the following code:

```
clock = 0;
t=0;
dt = 0.001;

while t<10
    clock=clock+1;
    Sys.computeForces();
    Sys.step(dt);
    t=t+dt;
    %visualization...
end
```

The result of this simulation: simulation result. We shall also record the energy of the system:

$$E = K(1) + K(2) + K(3) \tag{8}$$

$$= \frac{1}{2}\left(M_1 U_1^2 + (I_1\Omega_1)\cdot\Omega_1 + M_2 U_2^2 + (I_2\Omega_2)\cdot\Omega_2 + M_3 U_3^2 + (I_3\Omega_3)\cdot\Omega_3\right). \tag{9}$$

Then in while loop we add:

```
    T_save(clock) = t;
    R1 = Sys.RigidBodies(1);
    R2 =Sys.RigidBodies(2);
    R3 = Sys.RigidBodies(3);
    E = 0.5*(R1.M*dot(R1.U, R1.U)+dot(R1.O*R1.I, R1.O)+...
```

```
        R2.M*dot(R2.U, R2.U)+dot(R2.O*R2.I, R2.O)+...
        R2.M*dot(R3.U, R3.U)+dot(R3.O*R3.I, R3.O));
    E_save(clock) = E;
```

As a result, we shall plot the energy:

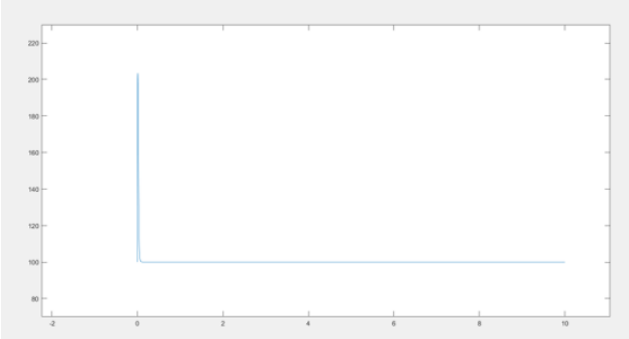


Figure 8: Energy vs time

Something is wrong here though, energy spikes in the beginning and go back to usual. What is happening here? Because of the exponential decay, we may predict that this is due to the corrective damping on one of the links. Then, were there any links that were incorrectly added? As it turn out, in the setup, center of mass of the rigid body 3, the axial link, is set at [0 0 1], while the z coordinate of the two top vertices of the rotors are at 0.5. After fixing the axial link initial center of mass to be at [0 0 0.5], we have the conservation of energy:

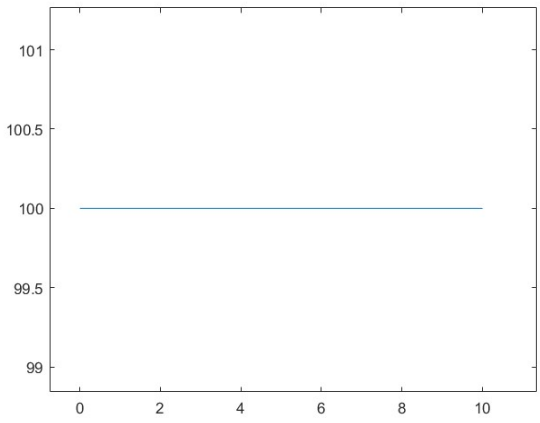


Figure 9: Energy vs time (corrected model)

As expected, the energy is $E = \frac{1}{2}\left(1 \cdot 10^2 + 1 \cdot 10^2\right) = 100$.

Now let's investigate if the simulation is accurate enough for when there are more complex interactions. We let a gaussian function that spikes to 50 at $t = 2$ be our shock function, and apply this as a force acting on the top of rigid body 1 in the x-direction. In simulation loop we add:

```
while t<10:
    %...
    Sys.RigidBodies(1).F(2, :) = Sys.RigidBodies(1).F(2, :)+[50*exp(-(50*(t-2))^2) 0 0];
    %...
end
```

This time somehow the system completely takes on NaN values after some time. After extensive debugging, it was discovered that the projection matrix calculation in RigidBody yields NaN matrices when $\|\Omega\|$ is not zero but $\|\Omega\|^2$ gets rounded to 0. Fixed code in RigidBody:

```
function [P,omega_norm] = P_omega(~,omega)
        % output: projection matrix that projects a vector onto omega
        omega_norm = norm(omega,2);
        if(omega_norm^2 == 0) % originally: omega_norm == 0
            P = eye(3);
            return
        end
        P = zeros(3,3);

        for i = 1:3
            for j = 1:3
                P(i,j) = omega(i)*omega(j)/(omega_norm^2);
            end
        end
    end
```

Simulation result: Same spin with shock response. The energy diagram:

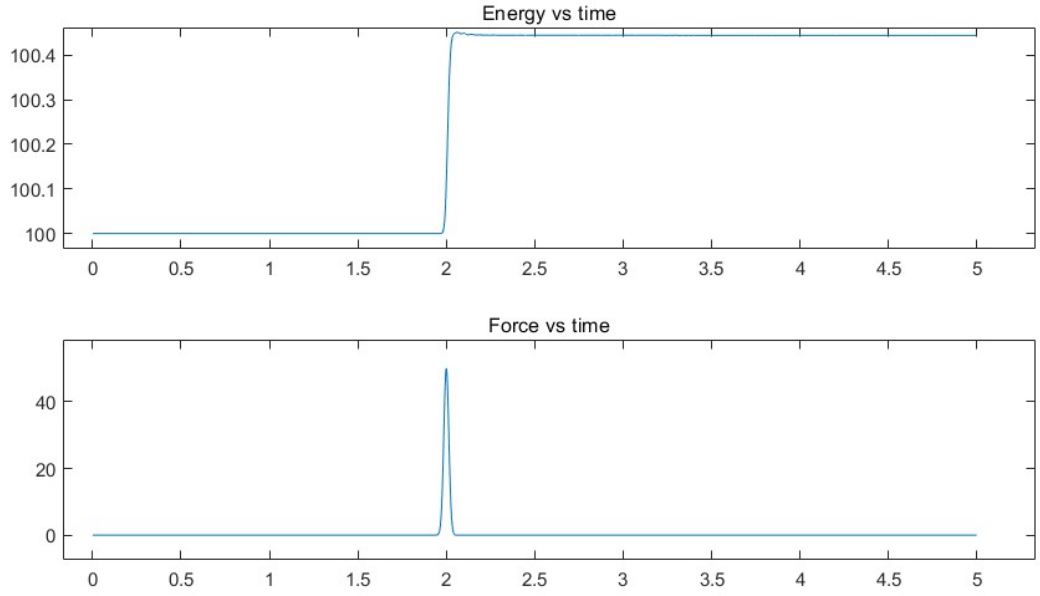


Figure 10: Shock Response with E,F vs T

We can see from the graphs that there is a jump in the energy after the shock. But this is completely expected because the force has done work by causing displacement. Energy goes up from 100 to about 100.45. What we want to see is that energy is conserved in the absence of external force. And indeed, after the spike, energy stays constant when there are no external force.

Note that total momentum is not conserved, as the two grounds are capable of exerting a non-zero net force on the system. It can also exert torque, regardless of which point we choose to be the ground. So angular momentum is not conserved. So we will not be using these two quantities as metrics for stability.

5 Investigation of Coaxial Rotor Dynamics

Now that the simulator has been validated, it would be interesting to see its effect on rotor pairs that have different angular velocities, namely $\omega_1 \neq \omega_2$. To start off, we want to verify the preliminary analysis, by looking at the dynamics of the system when $\omega_1 = -\omega_2$. Based on our predictions, its response to shock should behave as though the rotors are not spinning at all. Let's first see what it looks like when the rotors are not spinning: No Spin Shock.

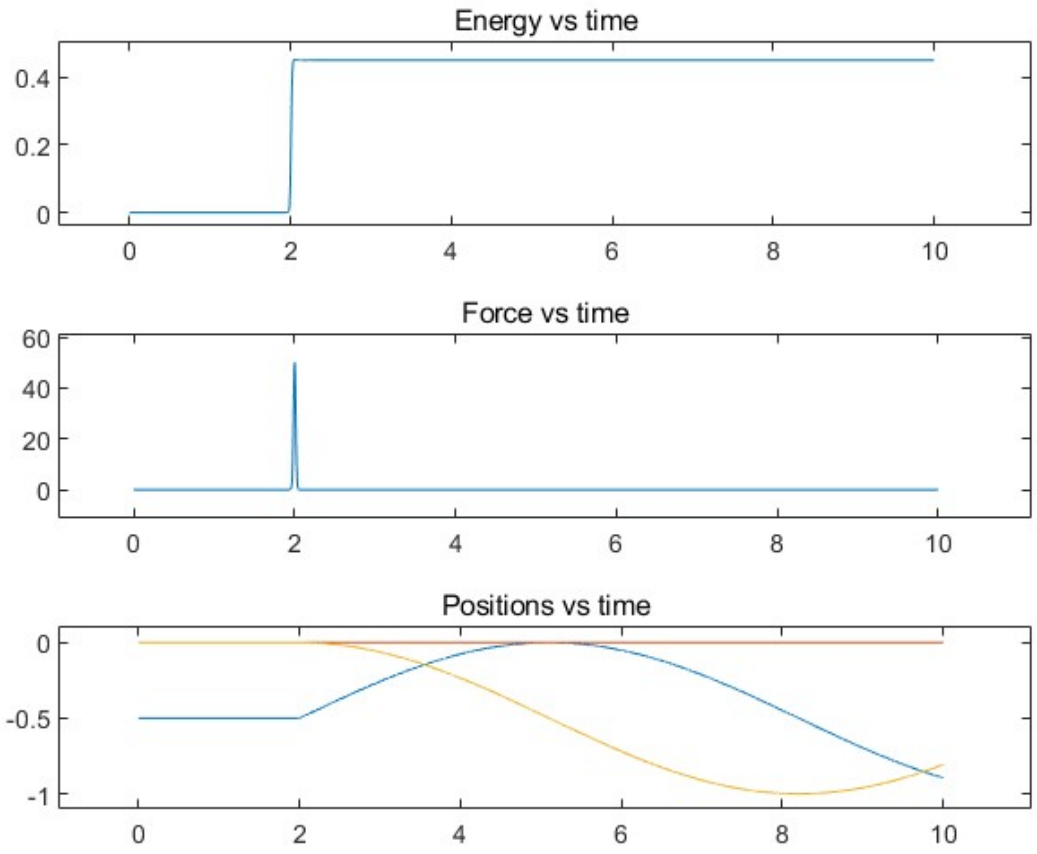


Figure 11: E, F, and R1.X_cm vs time for no spin

As one might guess, the rotor pair topples over after the shock, and starts the slow and continued the cyclic motion around the two grounds.

Now let's move on to anti spinning rotor pairs and see if that's also the case. We set angular velocities:

```
R1 = RigidBody([-0.5 0 0], X_r, I, R, M, [0 0 10], U);
R2 = RigidBody([0.5 0 0], X_r, I, R, M, [0 0 -10], U);
```

Simulation result and the graph: Anti-spin shock

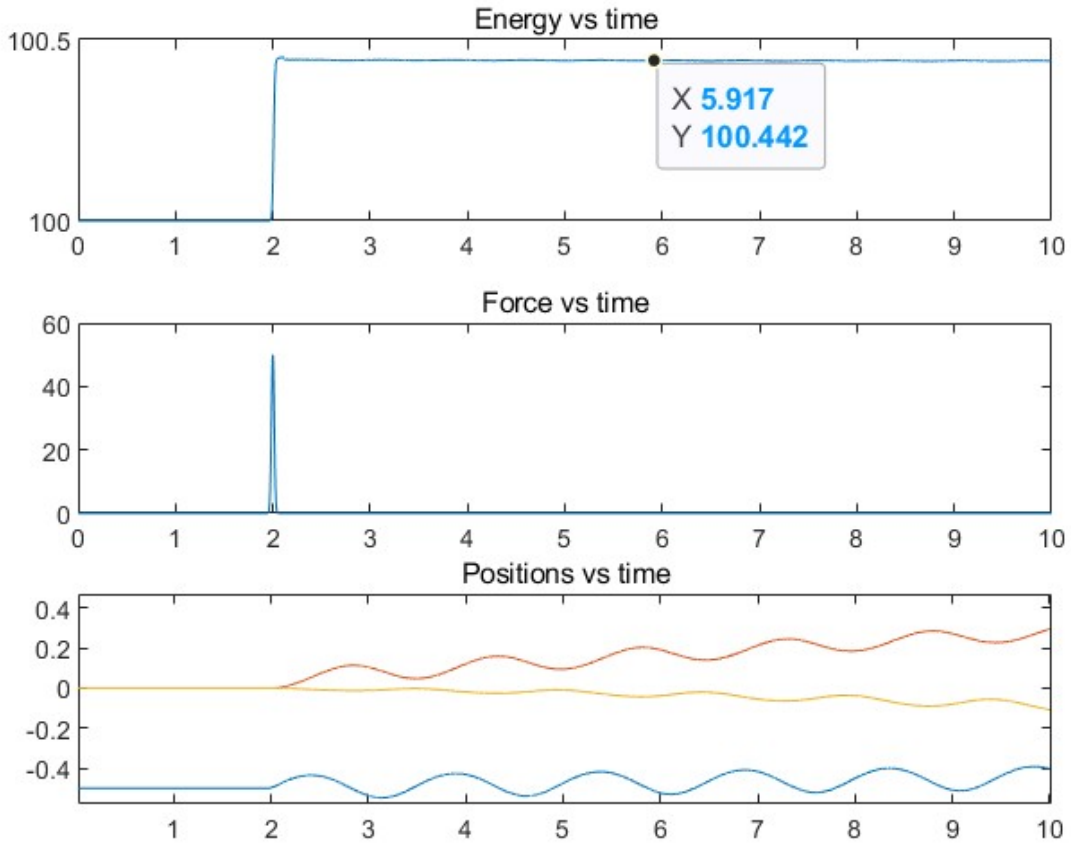


Figure 12: E, F, and R1.X_cm vs time for antispin. In the bottom graph, blue, red and yellow line correspond to x, y, and z position of center of mass or rigid body 1 respectively.

This new result clearly shows something different. While the system starts slowly toppling in the Y-Z plane, which agrees with its instability, the system **displays stability in the X direction** (seen in blue).

We can again verify this property by applying a shock in the y-direction: Anti-spin shock

```
Sys.RigidBodies(1).F(2, :) = Sys.RigidBodies(1).F(2, :)+[0 50*exp(-(50*(t-2))^2) 0];
```

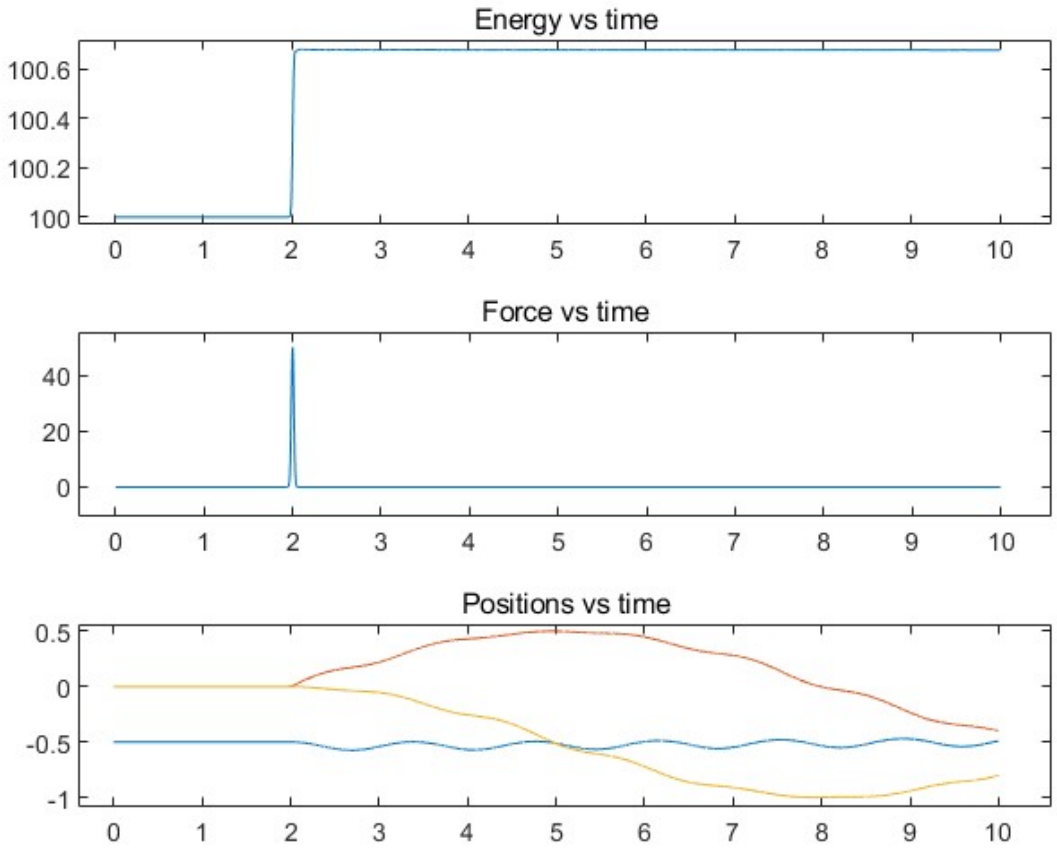


Figure 13: E, F, and R1.X_cm vs time for antispin. In the bottom graph, blue, red and yellow line correspond to x, y, and z position of center of mass or rigid body 1 respectively.

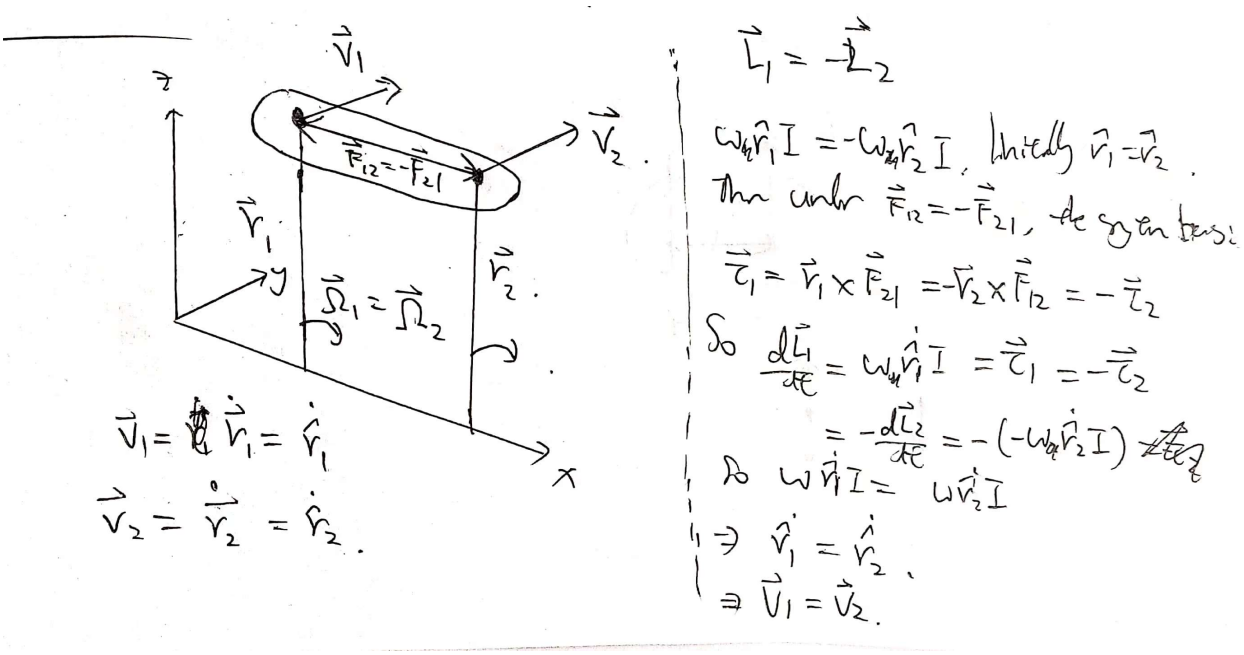
We can see that while the rotor pair starts to topple in the Y-Z plane, its position in the X-direction (blue) remains relatively fixed. So what is happening? The answer is that the assumption that $\vec{r}_1 = \vec{r}_2$ in our preliminary analysis is incorrect. The linkage in the middle can twist out of its original plane, causing $\vec{r}_1 \neq \vec{r}_2$. Hence we must use a completely different approach to predict its motion when the angular velocities are opposite.

For rotor pair with the same spin, the motion is as shown here. Its motion is "trivially" solved since the rotational axis of the two rotors are necessarily aligned at all times. Then analysis made in the preliminary analysis will all remain valid.

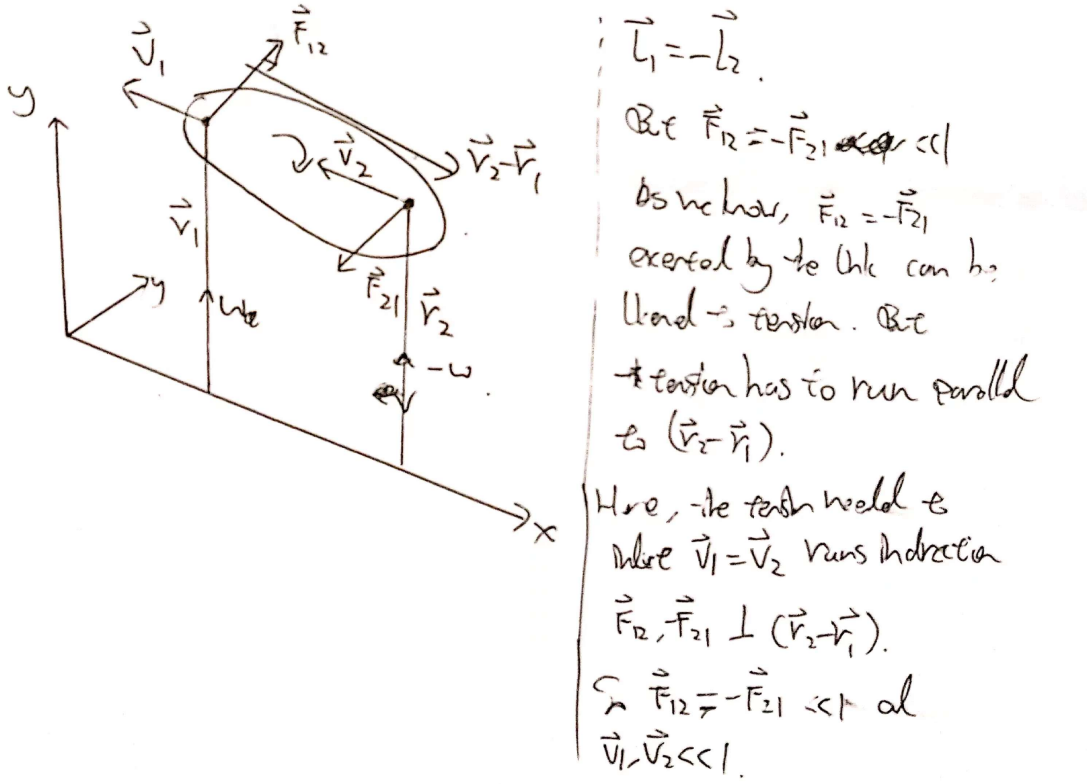
5.1 Instabilities of anti-spin rotor pair

In this section I will focus on investigating the motion of a pair of coupled rotor with opposite angular velocities. To begin with, I will address why the the rotor pair displays instability in the Y-Z plane but not in the X-Z plane. In the next section, I will use a perturbative approach to derive the natural oscillation frequency and approximate motion of the anti-spin rotor pair for small displacements in the X-direction (like the ones seen thus far).

For displacements in the Y-Z plane, we have the following force analysis:



For displacements in the X-Z plane, we have the following analysis:



From the above analysis we can draw the conclusion that $\vec{r}_1$ and $\vec{r}_2$ have on average no displacements in the x-direction.

5.2 Free precessions of the anti-spin rotor pair

But this doesn't mean that the system is not oscillating in the X-direction. If we think in terms of oscillations, the rotational motion of the rotor pair around the X-axis can be thought of as one **mode**, and the oscillation of the rotor pair in the X-direction, combined with its corresponding precessions, can be thought of as the other **mode**. For investigating the second mode, I will attempt to generalize conclusions from the free precession of a single rotor.

For a single rotor, let its coordinate system be the Euler Angles:

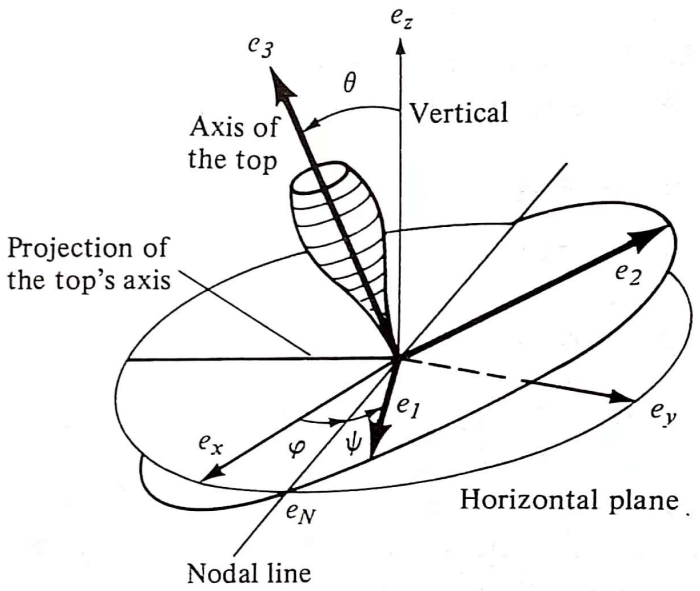


Figure 126 Euler angles

Figure 14: Euler angles as illustrated in Mathematical Methods [1]

Let its rotational inertia be $[I_1, 0, 0; 0, I_2, 0; 0, 0, I_3]$, $I_1 = I_2$ is the rotational inertia perpendicular to the gyroscope axis. I_3 is the rotational inertia along the gyroscope axis. Then its Lagrangian is given by:

$$\mathcal{L} = \frac{I_1}{2}(\dot{\theta}^2 + \dot{\varphi}^2 \sin^2\theta) + \frac{I_3}{2}(\dot{\psi} + \dot{\varphi}\cos\theta)^2.$$

Solving this Lagrangian yields [1]:

$$\dot{\varphi} = \frac{M_z - M_3 \cos\theta}{I_1 \sin^2\theta}.$$

Where

$$M_z = \dot{\varphi}(I_1 \sin^2\theta + I_3 \cos^2\theta) + \dot{\psi} I_3 \cos\theta$$
$$M_3 = \dot{\varphi} I_3 \cos\theta + \dot{\psi} I_3$$

are the constants of motion. In our case a rotor has uniform rotational inertia I and angular velocity (along the axial direction) $\dot{\psi} + \dot{\varphi}\cos\theta = \omega$ (and $-\omega$), so the constants of motion

reduces to:

$$M_z = \dot{\varphi} I + \dot{\psi} I \cos\theta$$
$$M_3 = \dot{\varphi} I \cos\theta + \dot{\psi} I = I\omega.$$

Rate of precession becomes:

$$\dot{\varphi} = \frac{M_z - I\omega\cos\theta}{I\sin^2\theta} = \frac{a - \omega\cos\theta}{\sin^2\theta},$$

where

$$\begin{aligned} a &= \dot{\varphi} + \dot{\psi}\cos\theta \\ &= \dot{\varphi} + (\omega - \dot{\varphi}\cos\theta)\cos\theta \\ &= \dot{\varphi}(1 - \cos^2\theta) + \omega\cos\theta \\ &= \dot{\varphi}_0 \sin^2\theta_0 + \omega\cos\theta_0 \end{aligned}$$

is a constant of motion that can be programmatically measured.

Now we shall continue to investigate the dynamics of of **axially coupled gyroscopes (rotors)**. We may start by considering the following generalized coordinates of the coupled gyroscopic system:

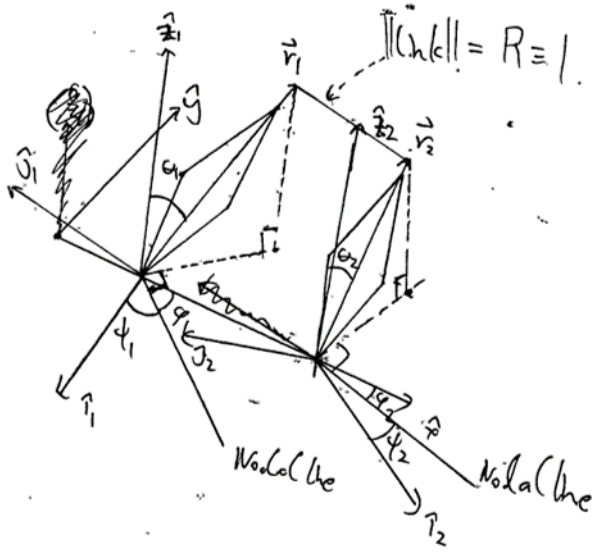


Figure 15: Axially-coupled rotor generalized coordinates

In particular we have two separate sets of three Euler angles $\theta_1, \varphi_1, \psi_1, \theta_2, \varphi_2, \psi_2$ characterizing a total of 6 degrees of freedom of the system. On the tip of the two rotors, the link between $\vec{r}_1$ and $\vec{r}_2$ must be of constant length R.

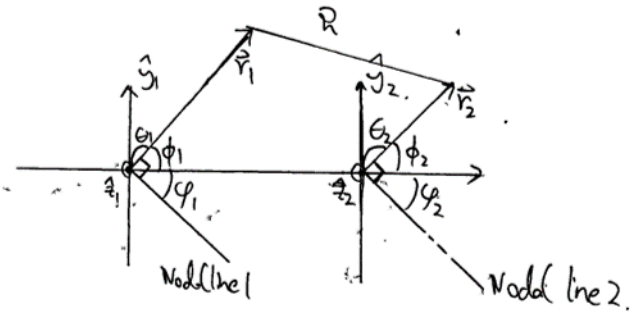


Figure 16: Top view of the coordinates

Let's again write the Lagrangian of the system:

$$\mathcal{L} = \frac{I_1}{2}(\dot{\theta}_1^2 + \dot{\varphi}_1^2 \sin^2\theta_1) + \frac{I_3}{2}(\dot{\psi}_1 + \dot{\varphi}_1 \cos\theta_1)^2 + \frac{I_1}{2}(\dot{\theta}_2^2 + \dot{\varphi}_2^2 \sin^2\theta_2) + \frac{I_3}{2}(\dot{\psi}_2 + \dot{\varphi}_2 \cos\theta_2)^2.$$

Notice first that the Lagrangian only depends on $\dot{\varphi}$, so the Lagrangian will be invariant if we shifted φ's definition by a constant. Thus we shall define $\phi_1 = \varphi_1 + \frac{\pi}{2}, \phi_2 = \varphi_2 + \frac{\pi}{2}$, so that it is in agreement with our usual definition of the spherical angle of $\vec{r}_1$ and $\vec{r}_2$. The new Lagrangian is then:

$$\mathcal{L} = \frac{I_1}{2}(\dot{\theta}_1^2 + \dot{\phi}_1^2 \sin^2\theta_1) + \frac{I_3}{2}(\dot{\psi}_1 + \dot{\phi}_1 \cos\theta_1)^2 + \frac{I_1}{2}(\dot{\theta}_2^2 + \dot{\phi}_2^2 \sin^2\theta_2) + \frac{I_3}{2}(\dot{\psi}_2 + \dot{\phi}_2 \cos\theta_2)^2.$$

Notice now that due to the constraint that the length of the link is fixed: $\|\vec{r}_2 - \vec{r}_1\| = R = 1$, the real degree of this freedom is 5 instead of 6. So it would be incorrect to express the Lagrangian in all of $\theta_1, \phi_1, \psi_1, \theta_2, \varphi_2, \psi_2$. In particular, among $\theta_1, \phi_1, \theta_2, \varphi_2$, one of these coordinates needs to be re-expressed in terms of all the remaining coordinates. However, this discrepancy will be small for $\theta_1, \theta_2 << 1$.

For anti-spinning rotor pairs, namely $\dot{\psi}_1 = -\dot{\psi}_2 = \dot{\psi}$, we have the Lagrangian

$$\mathcal{L} = \frac{I_1}{2}(\dot{\theta}_1^2 + \dot{\phi}_1^2 \sin^2\theta_1) + \frac{I_3}{2}(\dot{\psi} + \dot{\phi}_1 \cos\theta_1)^2 + \frac{I_1}{2}(\dot{\theta}_2^2 + \dot{\phi}_2^2 \sin^2\theta_2) + \frac{I_3}{2}(-\dot{\psi} + \dot{\phi}_2 \cos\theta_2)^2$$

We shall use the **ansatz** that $\dot{\phi} = \dot{\phi}_1 = -\dot{\phi}_2$, which will be in agreement with what we **observed**. Then the Lagrangian becomes:

$$\begin{aligned}\mathcal{L} &= \frac{I_1}{2}(\dot{\theta}_1^2 + \dot{\phi}^2 \sin^2\theta_1) + \frac{I_3}{2}(\dot{\psi} + \dot{\phi} \cos\theta_1)^2 + \frac{I_1}{2}(\dot{\theta}_2^2 + \dot{\phi}^2 \sin^2\theta_2) + \frac{I_3}{2}(-\dot{\psi} - \dot{\phi} \cos\theta_2)^2 \\ &= \frac{I_1}{2}(\dot{\theta}_1^2 + \dot{\phi}^2 \sin^2\theta_1) + \frac{I_3}{2}(\dot{\psi} + \dot{\phi} \cos\theta_1)^2 + \frac{I_1}{2}(\dot{\theta}_2^2 + \dot{\phi}^2 \sin^2\theta_2) + \frac{I_3}{2}(\dot{\psi} + \dot{\phi} \cos\theta_2)^2\end{aligned}$$

Now because ϕ and ψ show up in the Lagrangian only in their time derivatives, they are cyclic, and we get to obtain first integrals (by Noether's theorem):

$$\begin{aligned}A = \frac{\partial \mathcal{L}}{\partial \dot{\phi}} &= \dot{\phi}(I_1 \sin^2\theta_1 + I_3 \cos^2\theta_1) + \dot{\psi} I_3 \cos\theta_1 + \dot{\phi}(I_1 \sin^2\theta_2 + I_3 \cos^2\theta_2) + \dot{\psi} I_3 \cos\theta_2 \\ &= 2\dot{\phi} I + \dot{\psi} I(\cos\theta_1 + \cos\theta_2) \\ B = \frac{\partial \mathcal{L}}{\partial \dot{\psi}} &= \dot{\phi} I \cos\theta_1 + \dot{\psi} I + \dot{\phi} I \cos\theta_2 + \dot{\psi} I \\ &= \dot{\phi} I(\cos\theta_1 + \cos\theta_2) + 2\dot{\psi} I.\end{aligned}$$

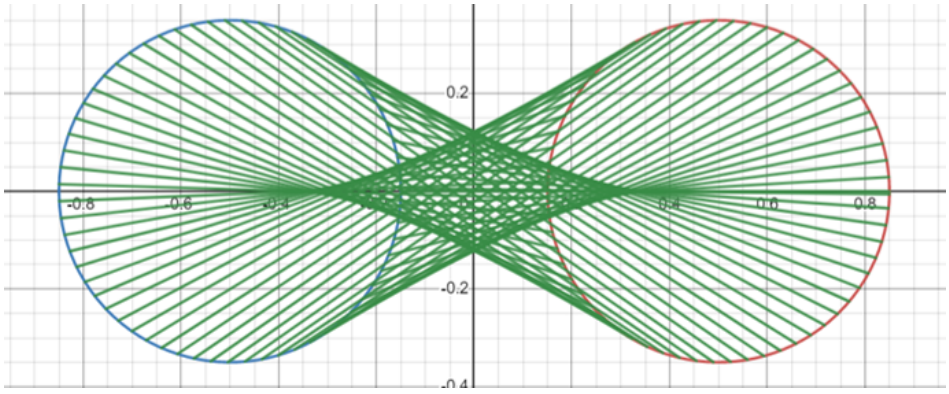


Figure 17: Trajectories of the coaxial rotor pair based on the ansatz

Using these two conserved quantities, we can eliminate $\dot{\psi}$ with:

$$\frac{A - 2\dot{\phi}I}{\cos\theta_1 + \cos\theta_2} = I\dot{\psi} = \frac{B - \dot{\phi}I(\cos\theta_1 + \cos\theta_2)}{2}$$

and obtain:

$$\dot{\phi} = \frac{2A - B(\cos\theta_1 + \cos\theta_2)}{I(4 - (\cos\theta_1 + \cos\theta_2)^2)},$$

where $B \sim 2\omega$ and $A \sim 2I(\dot{\phi}_0 \sin^2\bar{\theta_0} + \omega\cos\bar{\theta_0})$. $\bar{\theta}$ is the average inclination of θ_1 and θ_2. Value of A is based on derivation of a.

Now the dynamics of the rotor-pair with opposite angular velocity would be solved if θ_1 and θ_2 can just stay constant. However, we must not forget to check the constraint condition:

$$R = \|\vec{r}_2 - \vec{r}_1\| \sim \|(\sin\theta_1\cos\phi_1 - 0.5, \sin\theta_1\sin\phi_1) - (\sin\theta_2\cos\phi_2 + 0.5, \sin\theta_2\sin\phi_2)\|.$$

The above equation is taken from the projection of the tip of the rotors down to the X-Y plane. Based on our ansatz $\phi_1 = -\phi_2 = \phi$, we then obtain:

$$\begin{aligned}&\|(\sin\theta_1\cos\phi - 0.5, \sin\theta_1\sin\phi) - (\sin\theta_2\cos\phi + 0.5, -\sin\theta_2\sin\phi)\| \\ =&\sqrt{(1 + \cos\phi(\sin\theta_2 - \sin\theta_1))^2 + \sin^2\phi(\sin\theta_2 + \sin\theta_1)^2} \equiv C\end{aligned}$$

We can see that if we fixed $\theta_1 = \theta_2 \equiv \theta_0$ constant, value of C fluctuates between 1 and $\sqrt{1 + (2\sin\theta_0)^2}$ as ϕ goes from 0 to $\frac{\pi}{2}$ (see figure 18). Now for $\theta_0 << 1$, $\sqrt{1 + (2\sin\theta_0)^2} \sim 1 + \frac{1}{2}(2\sin\theta_0)^2$, so the amplitude of the fluctuation is approximately $\frac{1}{2}(2\sin\theta_0)^2$. The period of the fluctuation is π, since when ϕ goes from 0 to $\frac{\pi}{2}$ to π, R starts from its minimum and goes to maximum and back. So we can construct the amplitude-based correction:

$$\sin\theta_1 = \sin\theta_0 - \sin^2\theta_0\cos(\phi)$$

$$\sin\theta_2 = \sin\theta_0 + \sin^2\theta_0\cos(\phi).$$

We can express the corrected precession path of the coaxial rotor pair using parametric functions: analytical animation.

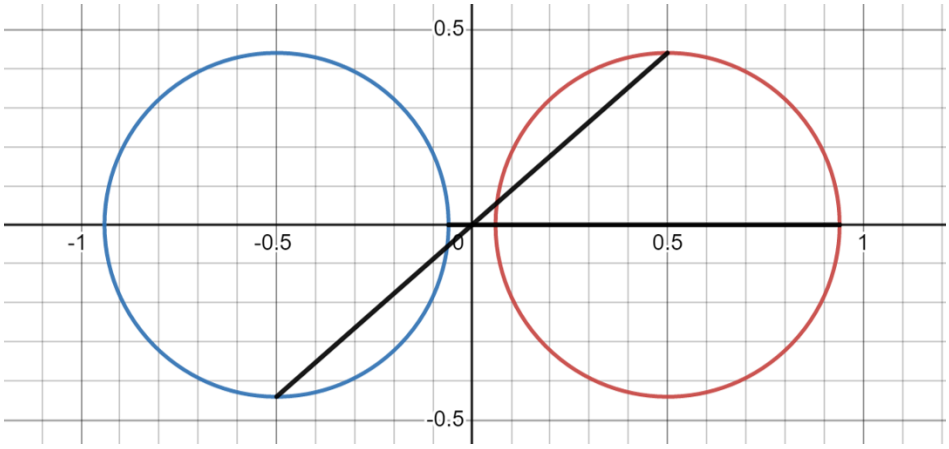


Figure 18: Fluctuation of link length R under constant θ assumption seen in the difference between length of the two black lines

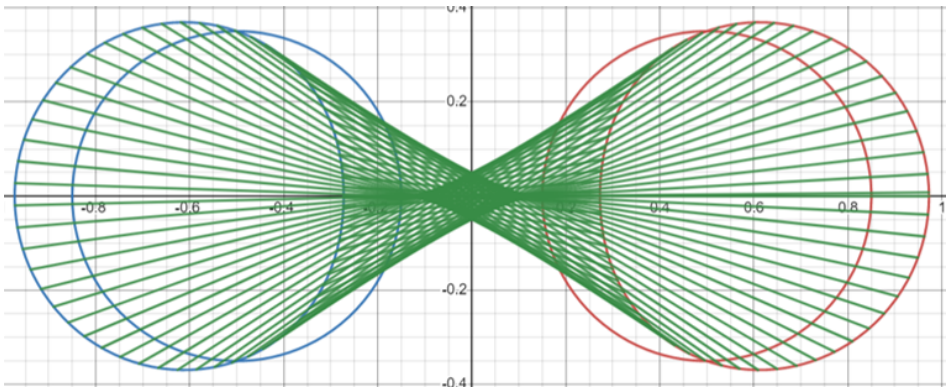


Figure 19: Corrected trajectories of the coaxial rotor pair

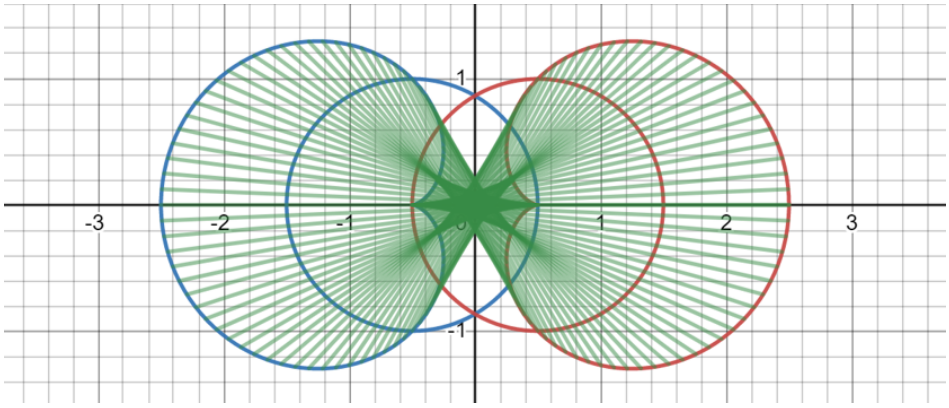


Figure 20: Corrected trajectories of the coaxial rotor pair with large oscillations

Interactive model available here. Then the precession rate after the correction is:

$$\dot{\phi} \sim \frac{2A - B(\cos\theta_0 + \cos\theta_0)}{I(4 - (\cos\theta_0 + \cos\theta_0)^2)} + \frac{\partial \frac{2A - B(\cos\theta_0 + \cos\theta_0)}{I(4-(\cos\theta_0+\cos\theta_0)^2)}}{\partial(\cos\theta_1 + \cos\theta_2)} \left(\frac{\partial(\cos\theta_1 + \cos\theta_2)}{\partial \sin\theta_1} - \frac{\partial(\cos\theta_1 + \cos\theta_2)}{\partial \sin\theta_2} \right) \sin^2\theta_0 \cos\phi \to 0$$

$$= \frac{2A - B(\cos\theta_0 + \cos\theta_0)}{I(4 - (\cos\theta_0 + \cos\theta_0)^2)}$$

unchanged.

5.2.1 Confirmation of theory

According to this theory about the rotor trajectory, we can obtain the x-coordinate of the left and right rotor top vertices expressed in terms of time as:

$$x_1 = \cos\left(\dot{\phi}t\right)\left(r_0 - r_0^2 \cos\left(\dot{\phi}t\right)\right) - 0.5,$$

$$x_2 = \cos\left(-\dot{\phi}t\right)\left(r_0 + r_0^2 \cos\left(-\dot{\phi}t\right)\right) + 0.5;$$

where $r_0 = \sin\theta_0$ and $\dot{\phi}$ appear to be constants of motion. The graph of the expression looks like:

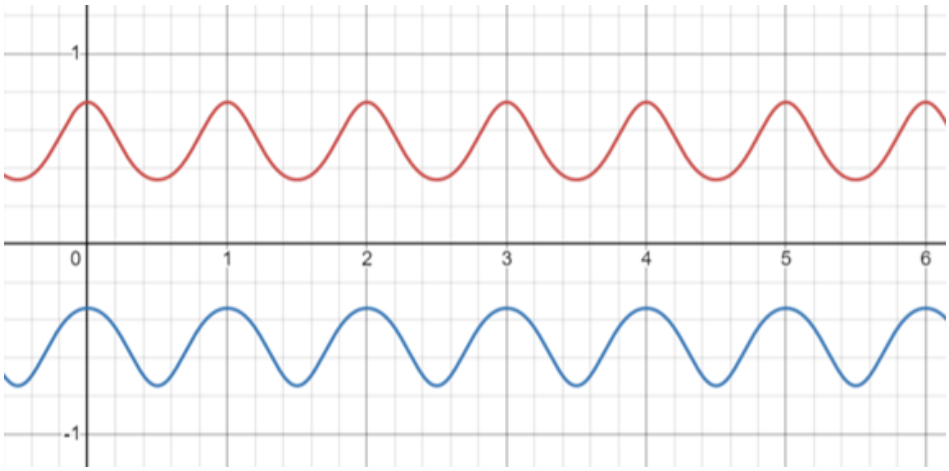


Figure 21: Analytical solution to the x-coordinate of the precession trajectories under conditions $\dot{\phi} = 1$ and $r_0 = 0.204$

compared to the simulated result of the x coordinates:

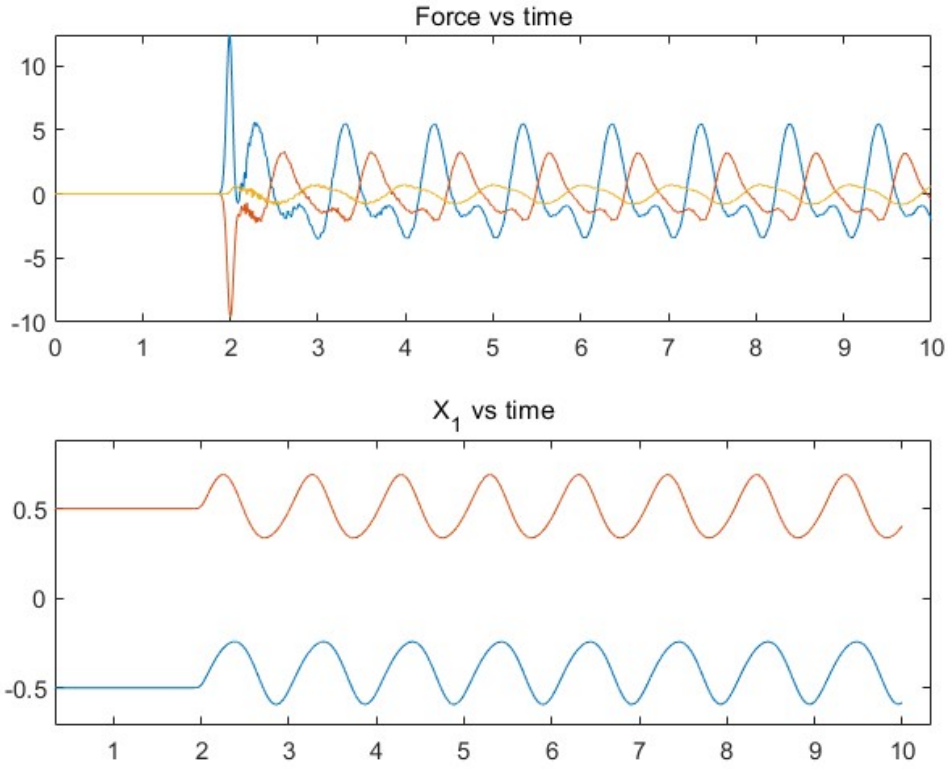


Figure 22: Simulated precession trajectories in the x-direction, here the "skewness" of the trajectories are due to a small angle between the precession centers and the x-axis, so the measured results are approximately correct with a little mixture of the y components.

Notably the "pinching" of the valleys of the left rotor x-trajectory and the peaks of the right rotor x-trajectory is visibly shared between the analytical and simulated solutions. Up to a good approximation, rotating the analytical solution to align with the skew axis should create agreeing results.

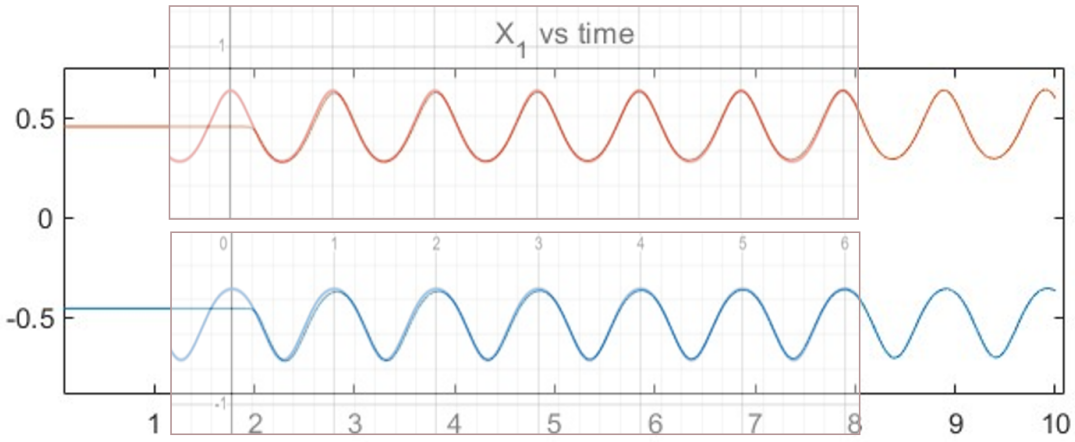


Figure 24: After taking the component of lateral oscillation along the skew axis, the results accurately agree with the predictions.

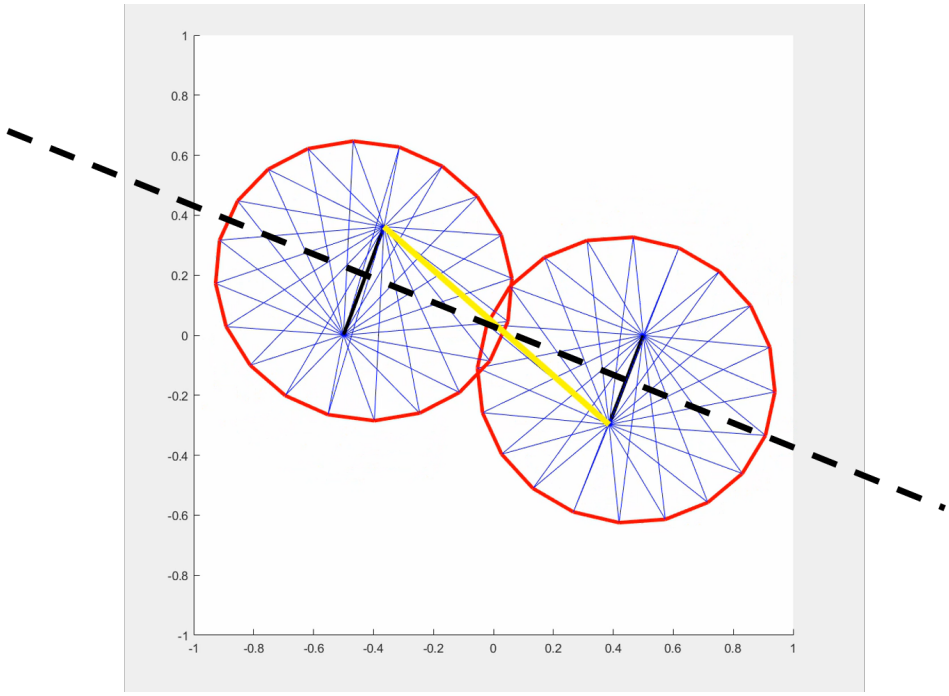


Figure 23: Skewness of the precession centers

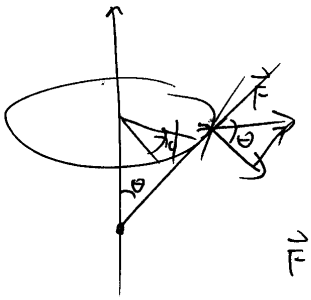
Initialization conditions:

```
X_r=[0 0 -0.5; 0 0 0.5];
I = [1 0 0; 0 1 0; 0 0 1];
%R = [1/2^0.5 0 -1/2^0.5; 0 1 0; 1/2^0.5 0 1/2^0.5];
R = eye(3);
M = 1;
O = [0 0 10];
U = [0 0 0];
R1 = RigidBody([-0.5 0 0], X_r, I, R, M, [0 0 10], U);
R2 = RigidBody([0.5 0 0], X_r, I, R, M, [0 0 -10], U);
X_r = [-0.5 0 0; 0.5 0 0];
O = [0 0 0];
U = [0 0 0];
R3 = RigidBody([0 0 0.5], X_r, 0.1*I, R, 0.01, O, U);
%...
while t0<T
%...

    Sys.RigidBodies(1).F(2, :) = Sys.RigidBodies(1).F(2, :)+...
        [40*exp(-(20*(t-2))^2) -11*exp(-(20*(t-2))^2) 0];
%...
end
```

5.3 Natural precession rate of the coupled oscillators

We have below the derivation of torque free precession (nutation) of a single gyroscope:



In the gyroscopic frame, centripetal force is given by

$$\vec{F} = \dot{\phi}^2 m r \sin\theta .$$

Then total torque exerted on the gyroscope is:

$$\vec{\tau} = \int_0^R \vec{r}\times\vec{F}\, dr = \hat{\theta}\int_0^R \dot{\phi}^2 \rho_r r^2 \sin\theta\cos\theta\, dr$$

$$= \hat{\theta}\dot{\phi}^2 \sin\theta\cos\theta \int_0^R \rho_r r^2 dr$$

$$= \hat{\theta}\dot{\phi}^2 \sin\theta\cos\theta \left(I_1 + \frac{1}{4}R^2 M\right) \quad \text{(parallel axis)}$$

Simultaneously,

$$\vec{L} = \omega I_3 \hat{r} \quad \text{so}$$

$$\frac{d\vec{L}}{dt} = \omega I_3 \dot{\hat{r}} = \omega I_3 \sin\theta \cdot \hat{\theta}\dot{\phi}$$

So $\frac{d\vec{L}}{dt} = \vec{\tau} \Rightarrow \omega I_3 \sin\theta\, \dot{\phi}\hat{\theta} = \hat{\theta}\dot{\phi}^2 \sin\theta\cos\theta\left(I_1 + \frac{1}{4}R^2 M\right)$.

$$\Rightarrow \dot{\phi} = \frac{\omega I_3}{\cos\theta\left(I_1 + \frac{1}{4}R^2 M\right)}$$

In our case where $M = 1$, $I_3 = I_1 = 1$,

$$\dot{\phi} = \frac{\omega}{\cos\theta}\cdot\frac{4}{5}.$$

For $\theta \sim 0$, $\dot{\phi} = \frac{4}{5}\omega$.

Using this theory, let's compare the oscillation period between a single gyroscope and our predicted precession: Setup:

```
X_r=[0 0 -0.5; 0 0 0.5];
I = [1 0 0; 0 1 0; 0 0 1];
%R = [1/2^0.5 0 -1/2^0.5; 0 1 0; 1/2^0.5 0 1/2^0.5];
R = eye(3);
M = 1;
U = [0 0 0];
R1 = RigidBody([-0.5 0 0], X_r, I, R, M, [0 0 20], U);
...
while t<T
...
    Sys.RigidBodies(1).F(2, :) = Sys.RigidBodies(1).F(2, :)+[40*exp(-(20*(t-2))^2) 0 0];
...
end
```

The result is shown below:

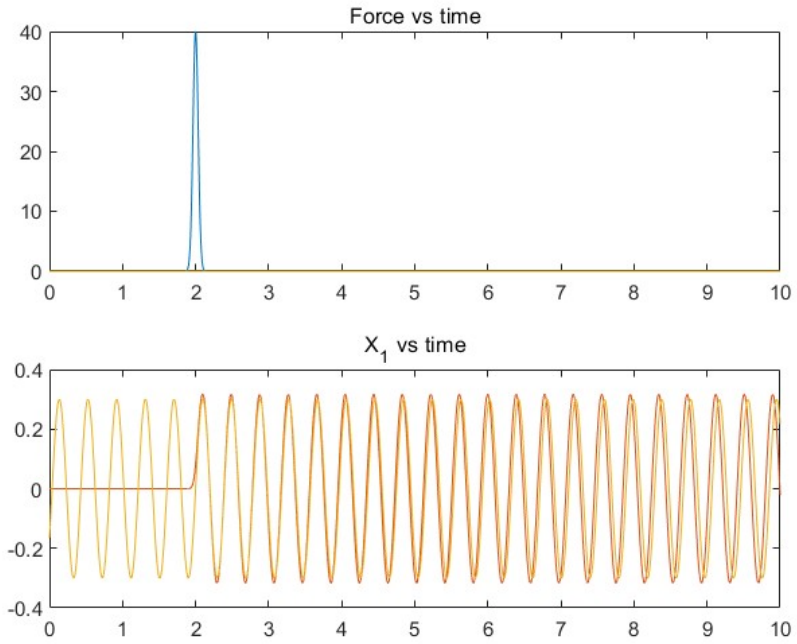


Figure 25: In the lower graph, orange curve is the simulated oscillation of the X_1 coordinate of the tip of the gyroscope, the yellow line is predicted according to the oscillation frequency

According to theory, $\dot{\phi} = \frac{4}{5}\omega$, in this set up $\omega = 20$, then $\dot{\phi} = 16$. As we plot $\sin 16(t-2)$ (shock located at t = 2), the resultant precession is almost an exact match.

For coupled rotors with rotational rate 20 and 20:

```
X_r=[0 0 -0.5; 0 0 0.5];
I = [1 0 0; 0 1 0; 0 0 1];
%R = [1/2^0.5 0 -1/2^0.5; 0 1 0; 1/2^0.5 0 1/2^0.5];
```

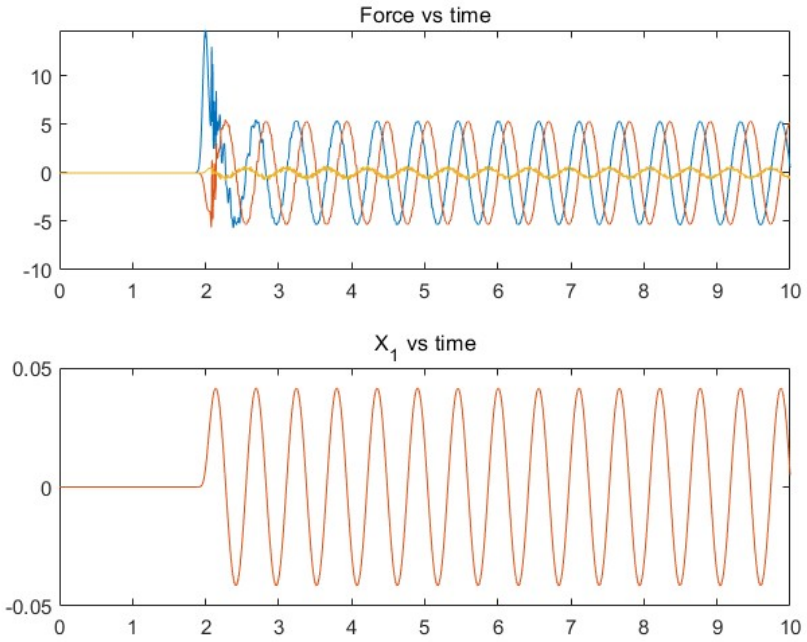


Figure 26: In the top graph, forces exerted on the top vertex of the left rotor are shown. The bottom graph shows the X_1 of the tip of the gyroscope.

```
R = eye(3);
M = 1;
U = [0 0 0];
R1 = RigidBody([-0.5 0 0], X_r, I, R, M, [0 0 20], U);
R2 = RigidBody([0.5 0 0], X_r, I, R, M, [0 0 20], U);
```

It seems like the angular frequency of $\dot{\phi}$ is somewhere between 16 and 8, namely it is a value between half of what it should be have been in the uncoupled gyroscope case. After some trial and error, the precession rate $\dot{\phi}$ is determined to be 11.4:

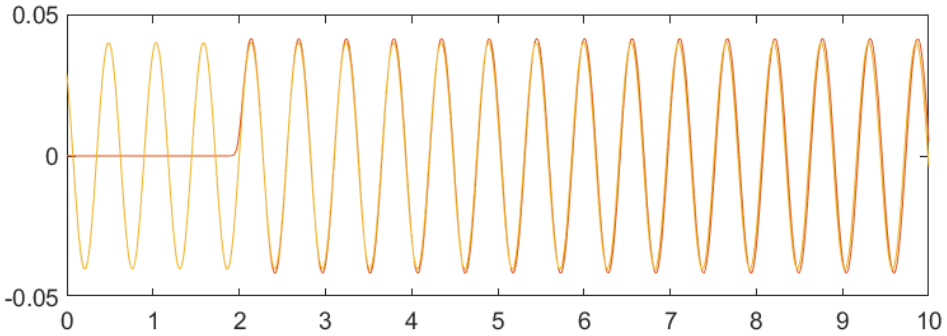


Figure 27: The X_1 of the tip of the gyroscope matched to oscillation frequency of 11.4.

So we hypothesize the oscillation frequency of coupled rotors and regular gyroscopes to differ by a constant:

$$\dot{\phi}_{coupled} = 0.7125 \cdot \frac{4}{5}\omega.$$

We shall see if this is also the case for anti-spinning rotors:

```
X_r=[0 0 -0.5; 0 0 0.5];
I = [1 0 0; 0 1 0; 0 0 1];
%R = [1/2^0.5 0 -1/2^0.5; 0 1 0; 1/2^0.5 0 1/2^0.5];
R = eye(3);
M = 1;
O = [0 0 10];
U = [0 0 0];
R1 = RigidBody([-0.5 0 0], X_r, I, R, M, [0 0 10], U);
R2 = RigidBody([0.5 0 0], X_r, I, R, M, [0 0 -10], U);
```

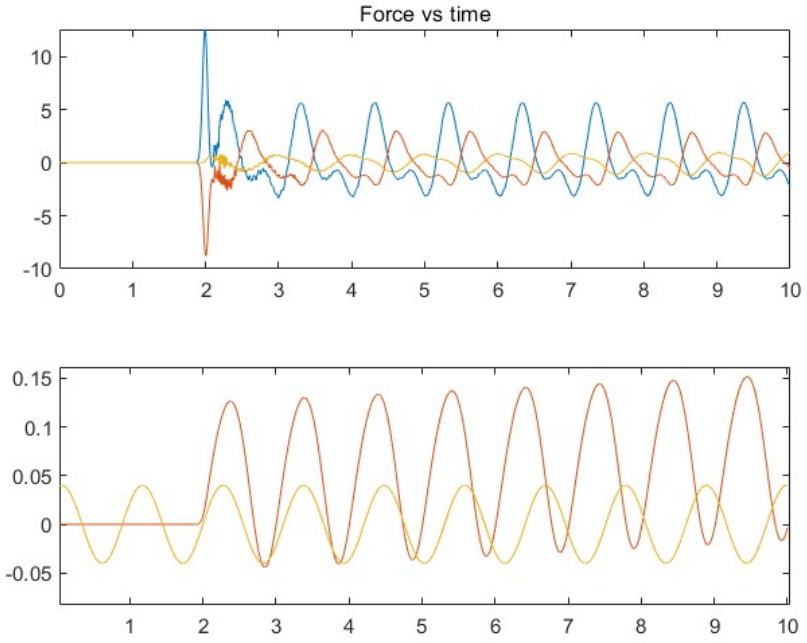


Figure 28: Simulation of a conversely spinning pair in the 2nd mode

Here we matched it to a frequency of $\dot{\phi} = 10 \cdot \frac{4}{5} \cdot 0.7125 = 0.57$, which does seem to approximately hold at least for the first few cycles. With the factor of $c = 0.7125$ further verified, we can write the oscillation frequency of axially coupled rotors at average inclination θ as:

$$\dot{\phi} = 0.7125 \cdot \frac{\omega I_3}{\cos\theta(I_1 + \frac{1}{4}R^2M)}.$$

When $\theta = 0$, we obtain what I define to be the "natural" oscillation frequency of the coupled rotor pair:

$$\dot{\phi} = 0.7125 \cdot \frac{\omega I_3}{I_1 + \frac{1}{4}R^2M}.$$

6 Conclusion

Through a combination of derivations and experiments, we shall conclude for a axially coupled rotor with opposite spins that there are 5 degrees of freedoms, 3 excluding the rotation of the

rotor themselves; there are 2 modes of oscillations, 1 is transversal around the $Y - Z$ plane, and the other is a special paired precession motion described in section 5.2. The natural frequency of precession of axially symmetric rotor pairs can both be approximated to be:

$$\dot{\phi} = 0.7125 \cdot \frac{\omega I_3}{\cos\theta(I_1 + \frac{1}{4}R^2M)}.$$

The collage of simulations referenced in the paper are available here.

References

[1] V.I. Arnold (1978), *Mathematical Methods of Classical Mechanics, Chapter 30 Lagrange's top*, Springer-Verlag New York.

[2] Charles S. Peskin, *Lecture Note from Special Topics: MODELING AND SIMULATION IN SCIENCE, ENGINEERING, AND ECONOMICS*, `https://www.math.nyu.edu/~peskin/modsim_lecture_notes/rigid_body_motion.pdf`