Albasua2-DesignProblem01

Nicolas Alba

February 2017

1 Design Goals

The system will begin spinning randomly at some angular velocity $\dot{\vec{\theta}}$. I want to eventually achieve $\dot{\vec{\theta}} = 0$. In order to do that I will start by by changing the initial angular velocities to some arbitrary constants just to see how the system will behave. I will also look at the angular momentum J and see how I can change that to get a better understanding of the system.

Now, due to the way Professor made it sound, I may not actually be able to get the satellite to stop spinning completely. So I will try to make w_1 and w_2 converge to 0. I will attempt to get w_3 to converge to a steady number, but not necessarily to 0.

If I want w_3 to converge to 0, I will need my A matrix to include terms that will allow the input to regulate w_3 based on the state.

2 Linearize

We start with:

$$\begin{aligned}
\tau_1 &= J_1\dot{w}_1 - (J_2 - J_3)w_2w_3 \\
\tau_2 &= J_2\dot{w}_2 - (J_3 - J_1)w_3w_1 \\
0 &= J_3\dot{w}_3 - (J_1 - J_2)w_1w_2,
\end{aligned}$$

Solving for $\dot{w}$, we get:

$$\begin{aligned}
\dot{w}_1 &= [(J_2 - J_3)w_2w_3 - \tau_1]/J_1 \\
\dot{w}_2 &= [(J_3 - J_1)w_3w_1 - \tau_2]/J_2 \\
\dot{w}_3 &= [(J_1 - J_2)w_1w_2 - \tau_3]/J_3
\end{aligned}$$

Then we linearize at an equilibrium point. Say $\tau_1 = \tau_2 = 0$ and $w_1 = w_2 = 0$ and $w_3 = 1$
In matrix form this becomes:

$$\begin{bmatrix}\dot{w}_1\\ \dot{w}_2\\ \dot{w}_3\end{bmatrix} = \begin{bmatrix}[(J_2 - J_3)w_2w_3 + \tau_1]/J_1\\ [(J_3 - J_1)w_3w_1 + \tau_2]/J_2\\ [(J_1 - J_2)w_1w_2]/J_3\end{bmatrix} = \begin{bmatrix}0\\0\\0\end{bmatrix}$$

With:

$$\dot{x} = \begin{bmatrix}\dot{w}_1\\ \dot{w}_2\\ \dot{w}_3\end{bmatrix}; x = \begin{bmatrix}w_1 - w_{1e}\\ w_2 - w_{2e}\\ w_3 - w_{3e}\end{bmatrix}; u = \begin{bmatrix}\tau_1 - \tau_{1e}\\ \tau_2 - \tau_{2e}\end{bmatrix}$$

A and B are:

$$A = \begin{bmatrix}0 & [(J_2 - J_3)w_{3e}]/J_1 & [(J_2 - J_3)w_{2e}]/J_1\\ [(J_3 - J_1)w_{3e}]/J_2 & 0 & [(J_3 - J_1)w_{1e}]/J_2\\ [(J_1 - J_2)w_{2e}]/J_3 & [(J_1 - J_2)w_{1e}]/J_3 & 0\end{bmatrix} = \begin{bmatrix}0 & (J_2 - J_3)w_{3e}/J_1 & 0\\ (J_3 - J_1)w_{3e}/J_2 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}$$

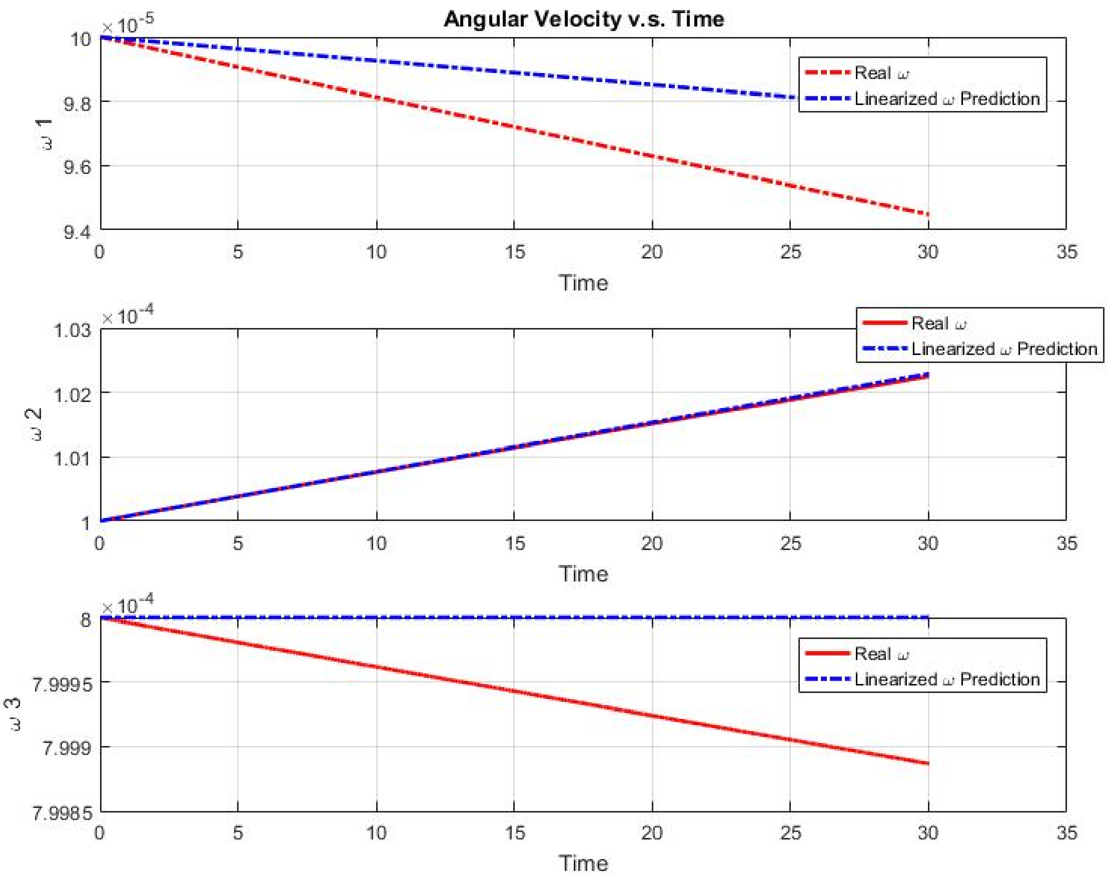
$$B = \begin{bmatrix} 1/J_1 & 0 \\ 0 & 1/J_2 \\ 0 & 0 \end{bmatrix}$$

3 Application of Zero Input

Suppose we choose to apply the input $u = 0$. The resulting system is

$$\dot{x} = Ax$$

In this case, $x = \vec{w}$. So We can linearize the system with the A matrix found in the previous section.



You can see how for the first 2 plots, the linearized prediction somewhat follows the actual angular velocities. The linear model does not take into account any disturbance and really only works best for small values.

4 Application of State Feedback

Suppose we choose to apply an input in the form $u = -Kx$. The resulting system is

$$\dot{x} = (A - BK)$$

Suppose, in particular, we choose

$$K = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.2 & 0 \end{bmatrix}$$

The Eigenvalues of $A - BK$ are $s_1 = -11.5385$, $s_2 = -1.048$ and $s_3 = 0.$. Only the first 2 eigenvalues are negative and real. This shows how you will only be able to stabilize about the first 2 axes. This means that they are asymptotically stable. We verified this prediction by implementing the controller that corresponds to this choice of input.

```
function [actuators,data] = initControlSystem(sensors,references,parameters,data)
actuators.tau1 = 0;
actuators.tau2 = 0;
data.K= [1 0 0; 0 .2 0];
end

%
% STEP #1: Modify, but do NOT rename, this function. It is called every
% time through the simulation loop.
%

function [actuators,data] = runControlSystem(sensors,references,parameters,data)
x=[sensors.w1;sensors.w2; sensors.w3];
u= -data.K*x;
actuators.tau1 = u(1);
actuators.tau2 = u(2);
end
```

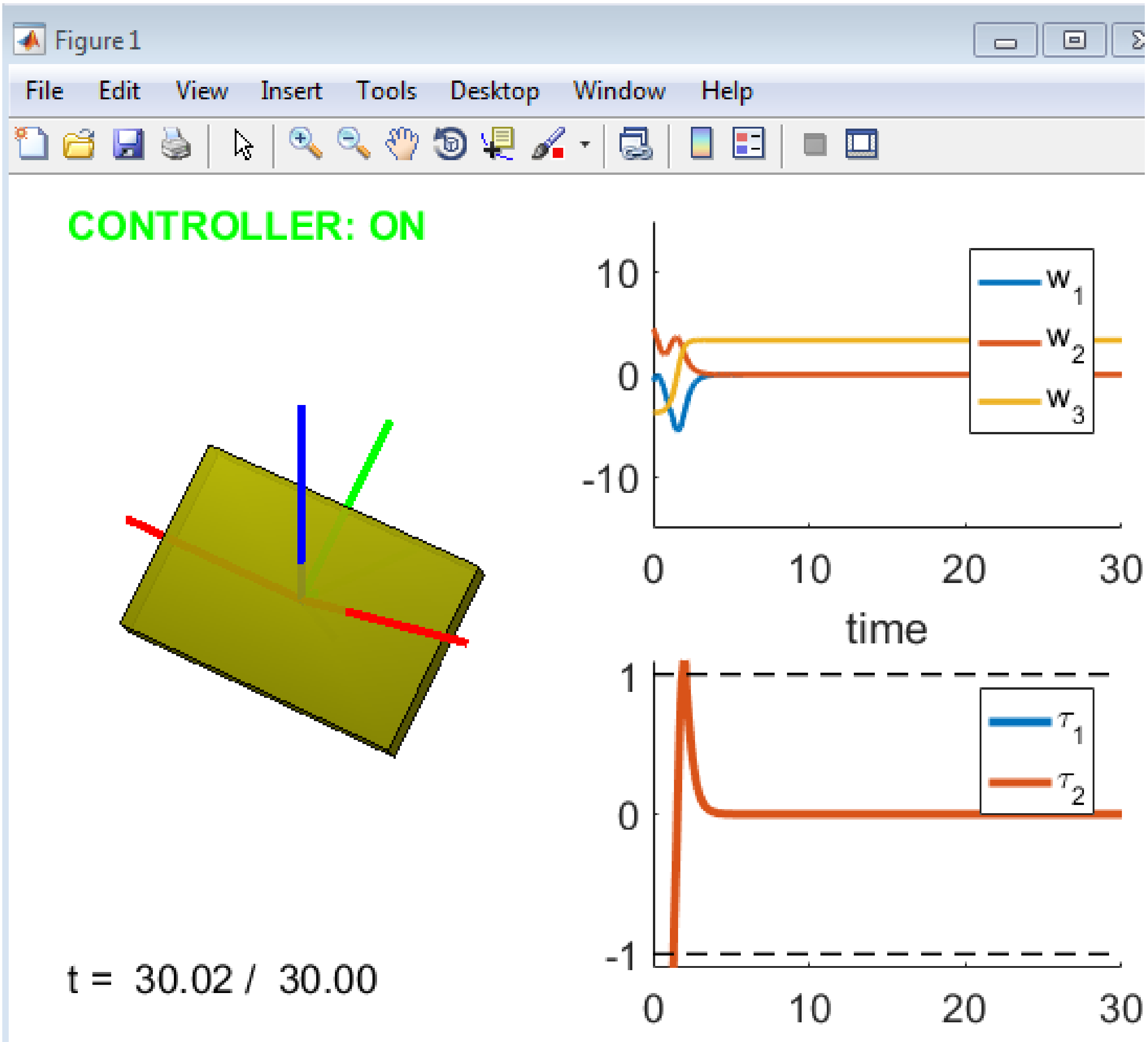


Figure 1
File Edit View Insert Tools Desktop Window Help
CONTROLLER: ON
t = 30.02 / 30.00
10
0
-10
0 10 20 30
time
w_1
w_2
w_3
1
0
-1
0 10 20 30
τ_1
τ_2

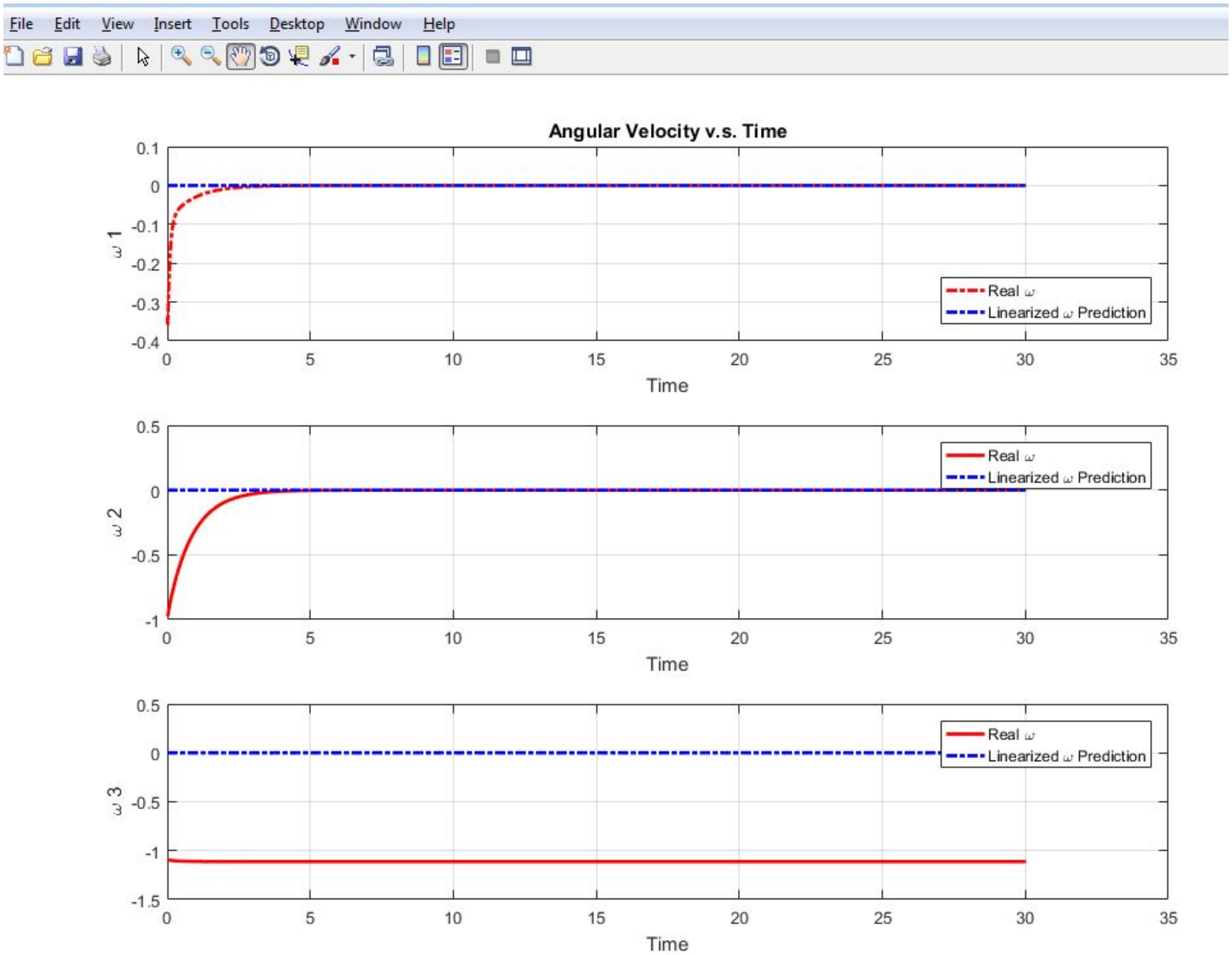


Figure 1: As you can see, the results of applying the state feedback to the rigid body for a particular initial condition.