Vehicle Control Applications
Assignment Report

Maxime Sabbadini
maxime.sabbadini7@gmail.com
Advanced Vehicle Engineering Center
Cranfield University
Fall, 2022

March 6, 2023

---

1 Task 1

1.1 Modeling of the plant

For this assignment, we will be considering the following system (figure 1) :

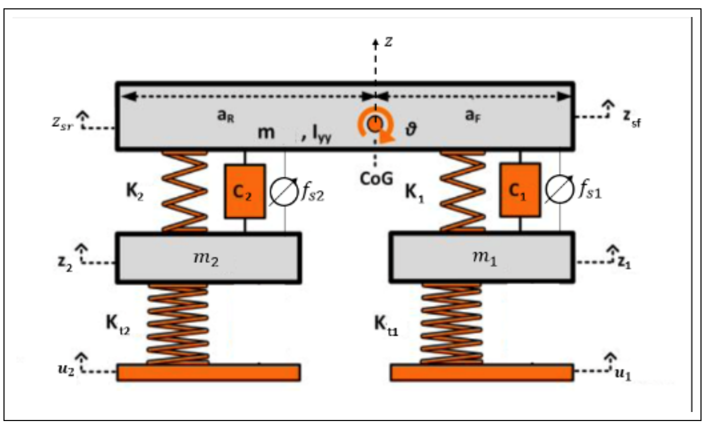

Figure 1: Half car suspension model

This system is a half car suspension model, which we will use to implement active suspension. To model it, we will start from first principle by applying Newton's second law to each mass but we will also be considering the pitch motion of the vehicle with the angle ν. We get the following system :

$$\begin{cases} m_1\ddot{z_1} = -k_{t_1}(z_1 - u_1) + k_1(z - z_1 - a_f\nu) + c_1(\dot{z} - \dot{z_1} - a_f\dot{\nu}) - f_{s1} \\ m_2\ddot{z_2} = -k_{t_2}(z_2 - u_2) + k_2(z - z_2 + a_r\nu) + c_2(\dot{z} - \dot{z_2} + a_r\dot{\nu}) - f_{s2} \\ m\ddot{z} = -k_1(z - z_1 - a_f\nu) - k_2(z - z_2 + a_r\nu) - c_1(\dot{z} - \dot{z_1} - a_f\dot{\nu}) - c_2(\dot{z} - \dot{z_2} + a_r\dot{\nu}) + f_{s1} + f_{s2} \\ I_{yy}\ddot{\nu} = a_f k_1(z - z_1 - a_f\nu) - a_r k_2(z - z_2 + a_r\nu) + a_f c_1(\dot{z} - \dot{z_1} - a_f\dot{\nu}) - a_r c_2(\dot{z} - \dot{z_2} + a_r\dot{\nu}) + a_f f_{s1} + a_r f_{s2} \end{cases}$$

These equations were expressed for the Mechatronics modeling assignment however all that follows was specially made for this module's assignment.

With these equations, we can simulate the behavior of our system. However, in our case we are going to proceed with a change of variables. In the end, we want our system to compensate the elevation induced by irregularities on the road, to improve passenger comfort. In other words, we want to have z_{sr} and z_{sf} as close to 0 as possible at anytime, by acting on the force actuators we have.

We will then reformulate this set of equations with the following change of variables ([1]) :

$$
\begin{cases}
z_{sf} = z - a_f \nu \\
z_{sr} = z + a_r \nu \\
\nu = \dfrac{z_{sr} - z_{sf}}{L} \\
z = \dfrac{a_f z_{sr} + a_r z_{sf}}{L}
\end{cases}
\tag{1}
$$

We now have the following set of equations :

$$
\begin{cases}
m_1 \ddot{z_1} = -k_{t_1}(z_1 - u_1) + k_1(z_{sf} - z_1) + c_1(\dot{z_{sf}} - \dot{z_1}) - f_{s1} \\
m_2 \ddot{z_2} = -k_{t_2}(z_2 - u_2) + k_2(z_{sr} - z_2) + c_2(\dot{z_{sr}} - \dot{z_2}) - f_{s2} \\
m \left(\dfrac{a_f \ddot{z_{sr}} + a_r \ddot{z_{sf}}}{L} \right) = -k_1(z_{sf} - z_1) - k_2(z_{sr} - z_2) - c_1(\dot{z_{sf}} - \dot{z_1}) - c_2(\dot{z_{sr}} - \dot{z_2}) + f_{s1} + f_{s2} \\
I_{yy} \left(\dfrac{\ddot{z_{sr}} - \ddot{z_{sf}}}{L} \right) = a_f k_1(z_{sf} - z_1) - a_r k_2(z_{sr} - z_2) + a_f c_1(\dot{z_{sf}} - \dot{z_1}) - a_r c_2(\dot{z_{sr}} - \dot{z_2}) + a_f f_{s1} + a_r f_{s2}
\end{cases}
$$

As we can see we have two equations combining two different second order derivatives. We cannot deduce a state space model from these equations so we have to solve this system before.

We have the following expression :

$$
\ddot{z_{sf}} = z_{sf} \left(-\frac{k_1}{m} - \frac{a_f^2 k_1}{I_{yy}} \right) + \dot{z_{sf}} \left(-\frac{c_1}{m} - \frac{a_f^2 c_1}{I_{yy}} \right) + z_{sr} \left(-\frac{k_2}{m} + \frac{a_f a_r k_2}{I_{yy}} \right) + \dot{z_{sr}} \left(-\frac{c_2}{m} + \frac{a_f a_r c_2}{I_{yy}} \right) + z_1 \left(\frac{k_1}{m} - \frac{a_f^2 k_1}{I_{yy}} \right) + \dot{z_1} \left(\frac{c_1}{m} + \frac{a_f a_r c_1}{I_{yy}} \right) + z_2 \left(\frac{k_2}{m} - \frac{a_f a_r k_2}{I_{yy}} \right) + \dot{z_2} \left(\frac{c_2}{m} - \frac{a_r^2 c_2}{I_{yy}} \right) + f_{s1} \left(\frac{1}{m} + \frac{a_f^2}{I_{yy}} \right) + f_{s2} \left(\frac{1}{m} - \frac{a_f a_r}{I_{yy}} \right)
$$

With the same method, we can deduce an expression of $\dot{z_{sr}}$:

$$
\ddot{z_{sr}} = z_{sf} \left(-\frac{k_1}{m} - \frac{a_f a_r k_1}{I_{yy}} \right) + \dot{z_{sf}} \left(-\frac{c_1}{m} + \frac{a_f a_r c_1}{I_{yy}} \right) + z_{sr} \left(-\frac{k_2}{m} - \frac{a_r^2 k_2}{I_{yy}} \right) + \dot{z_{sr}} \left(-\frac{c_2}{m} - \frac{a_r^2 c_2}{I_{yy}} \right) + z_1 \left(\frac{k_1}{m} - \frac{a_f a_r k_1}{I_{yy}} \right) + \dot{z_1} \left(\frac{c_1}{m} - \frac{a_f a_r c_1}{I_{yy}} \right) + z_2 \left(\frac{k_2}{m} + \frac{a_r^2 k_2}{I_{yy}} \right) + \dot{z_2} \left(\frac{c_2}{m} + \frac{a_r^2 c_2}{I_{yy}} \right) + f_{s1} \left(\frac{1}{m} - \frac{a_f a_r}{I_{yy}} \right) + f_{s2} \left(\frac{1}{m} + \frac{a_r^2}{I_{yy}} \right)
$$

We are now going to express our system into the state space form, with a slight deviation from the original one. As we can see figure 1, we have in total four inputs to the system, which are : u_1, u_2, f_{s1}, f_{s2}. We are going to divide these inputs into two categories :

- The control inputs which we can control (the force in the actuators)
- The external inputs which we do not have control over (road input)

Hence, we will be using the following state space representation (equation 2) :

$$
\dot{q} = Aq + Bu + Kw \tag{2}
$$

With $q = \begin{pmatrix} \dot{z_{sf}} & \dot{z_1} & \dot{z_2} & \dot{z_{sr}} & z_{sf} & z_1 & z_2 & z_{sr} \end{pmatrix}^T$, $u = \begin{pmatrix} f_{s1} & f_{s2} \end{pmatrix}^T$ and $w = \begin{pmatrix} u_1 & u_2 \end{pmatrix}^T$

The state space matrices can be found in the Matlab code.

We will be using the following values as parameters for the system (table 1.1) :

1.2 Test cases

We will now be looking at the different test cases we will be implementing to test the validity of our system. We will be considering two test cases, the first one will be a speed bump, the second one will be a random road profile.

Parameter	Value	Unit
$m_{1,2}$	40	kg
m	400	kg
$c_{1,2}$	1500	Ns/m
$k_{t_{1,2}}$	150 000	N/m
$k_{1,2}$	21 000	N/m
I_{yy}	600	$kg.m^2$
a_r	1.45	m
a_f	0.8	m

Table 1: Values of the parameters used in the modelization

1.2.1 Speed Bump

As we have seen before, we have in our system two road inputs which correspond to the front and the rear. All along this assignment, we will be considering that our vehicle is traveling at a constant speed V. Also, we will be considering the same inputs at both front and rear, but the input at the rear will be delayed depending on the speed of the vehicle. We can estimate this delay to be : $dt = \frac{L}{V}$ with $L = a_r + a_f$ the distance between the front and rear axle.

The profile of the speed bump we will be considering throughout this assignment is shown figure 2 :

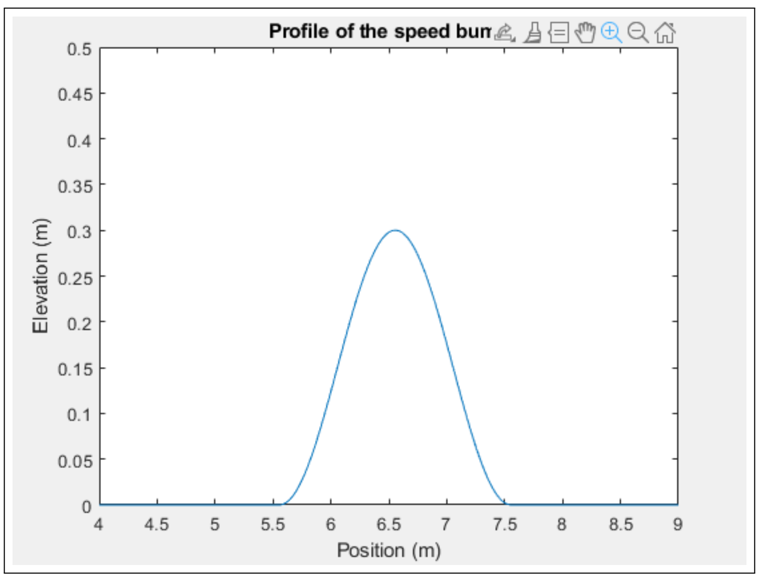

Figure 2: Profile of the speed bump

To design it, we use the following equation from **[2]** :

$$z(x) = \begin{cases} -\frac{1}{2}H\left(\cos\left(2\pi\frac{vt}{L}\right) - 1\right) \text{ if } t_0 < t < t_0 + \frac{L}{V} \\ 0 \text{ else} \end{cases}$$

With H and L the height and length of the bump.

1.2.2 Random road profile

For testing of the system, we will also be using a random road profile, generated with the Simulink schematic shown figure 3 and that can be seen figure 4 :

1.3 Performance targets

v Before starting to design the controller, we will be defining the performance targets we would like to achieve with such system. First, with obstacles such as speed bumps, we want to reduce the vertical body

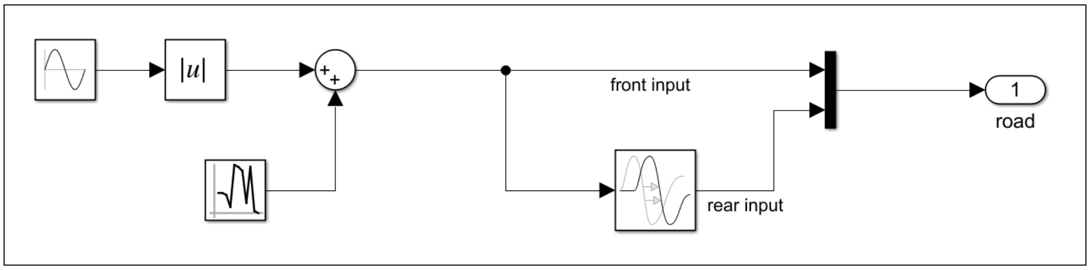

Figure 3: Simulink Schematic to generate the random road profile

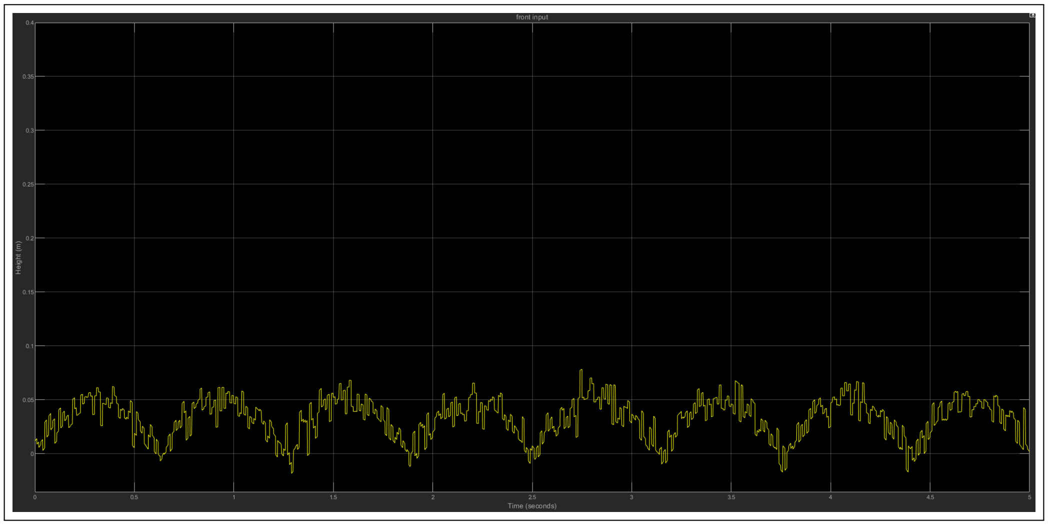

Figure 4: Visualization of the random road profile

displacement induced by this obstacle considerably, aiming for at least 1/3 of the peak value. The settling time of our system should be quick to allow the passenger maximum comfort. We will aim for about one second of settling time as a criteria for the design. One other criteria will be on the control effort we sent to the actuator. This control effort shall not be noisy because it would risk damaging the actuator which is not desirable at all. Body acceleration will also be a critical design criteria as this is what makes the passenger feel discomfort if there is too much change.

1.4 System validation

To assess the performance of the controller we designed, we will be using it with a Simscape model we developed that should represent the real system. We have decided to use this method to follow the real path of design of a control system in the industry :

- First we derive equations for our plant
- Then we linearize our plant to design the controller
- We test the controller on the linearized and non-linear plant
- We can then start implementing the controller on the real system

The system we designed in Simscape is shown figure 5 :

2 Task 2

We will now test the plant we have developed in the previous part with the different test cases.

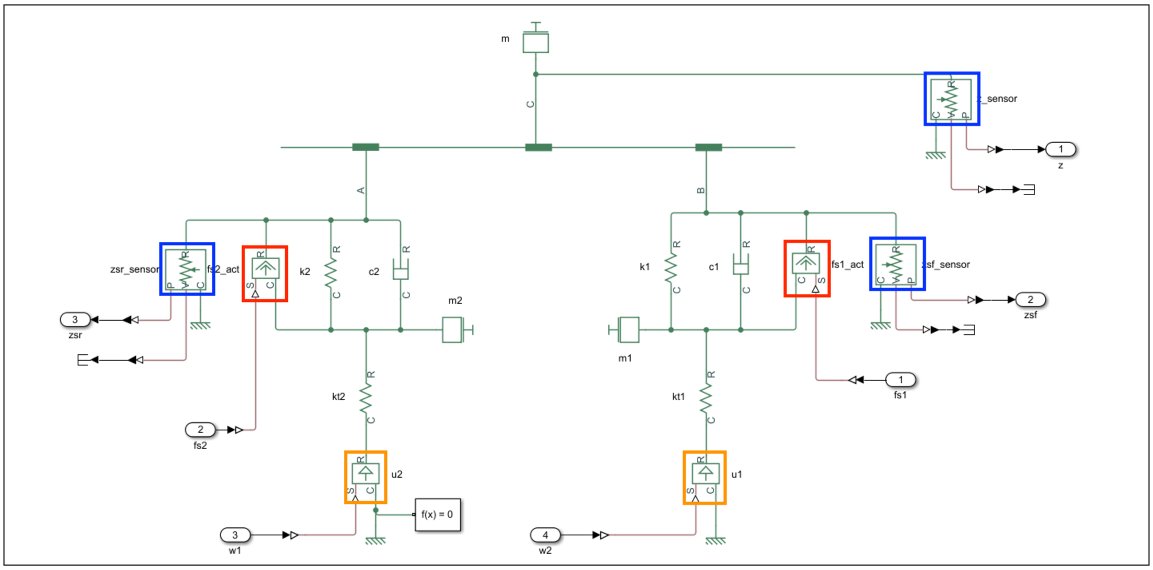

Figure 5: System designed under Simscape with in red the force actuators, in blue the position sensors and in orange the velocity sources which represents the input from the road

2.1 Response of the system to a bump input

For better readability of the graphs, we will only be considering the body elevation z when studying the performances of our system. Moreover, if not precised the speed used for the simulations will be $20km/h$. We can study the response of our passive suspension system to a bump input (figure 6) :

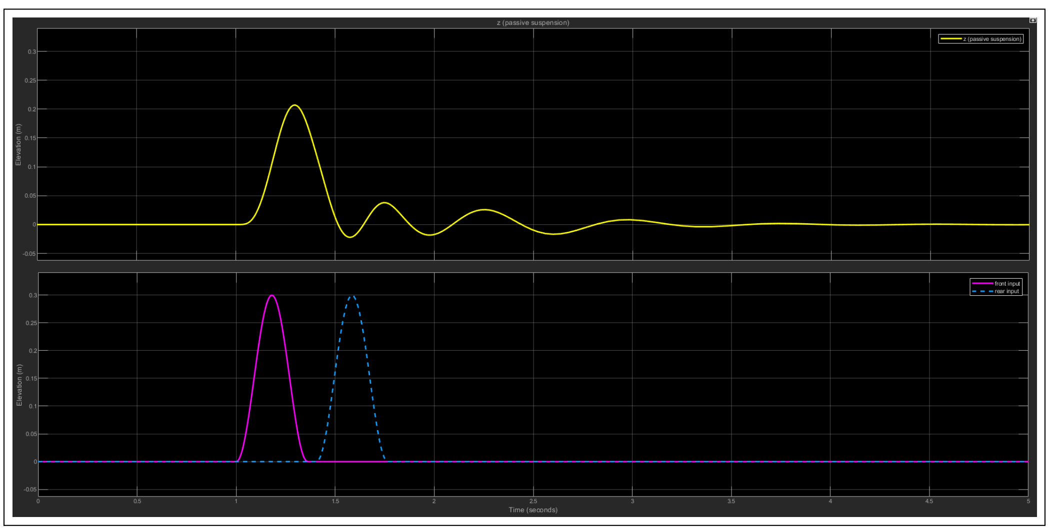

Figure 6: Response of the system to a speed bump at $20km/h$

As we could expect, as soon as we hit the bump the body starts oscillating until it reaches its equilibrium point which is zero deflection. We can compare the system we derived from the equations with the actual system designed in simscape together to judge on the accuracy of our model (figure 7) :

We can see that both responses are quite similar but still they are not perfectly equal. This is due to the fact that we had to make assumptions when we were deriving the system from first principles. In effect, the system is originally non-linear with some terms in $\cos\nu$ and $\sin\nu$, so to linearize it we made the assumption that $\nu \approx 0$ so we can say that $\cos\nu \approx 1$ and $\sin\nu \approx \nu$. But, we still have good performances and our linearized model will be very helpful in the design of the control system.

2.2 Response of the system to a random road profile input

We will now study the response of the passive system to the random road profile. We get the following figure :

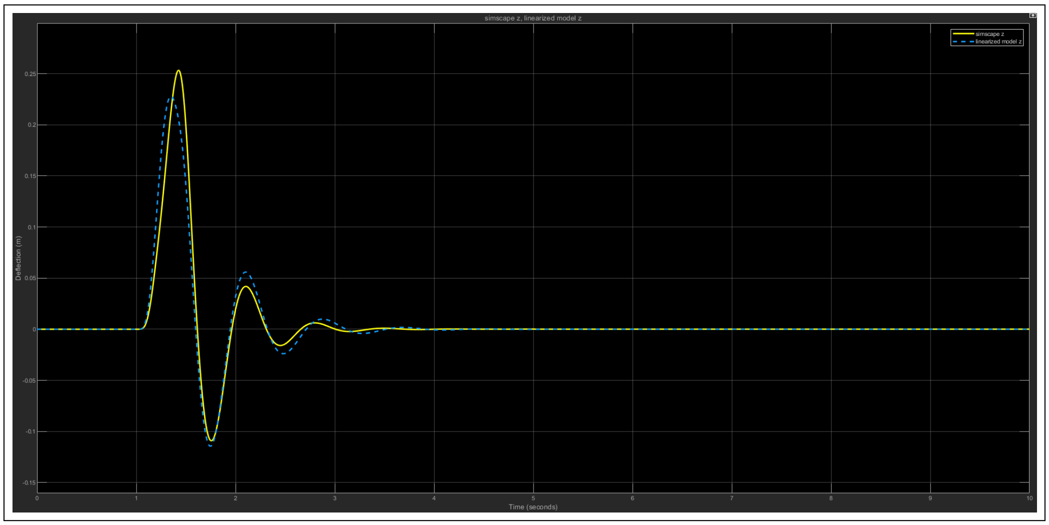

Figure 7: Comparison between the linearized model (dashed blue) and simscape model (solid yellow) for the speed bump input

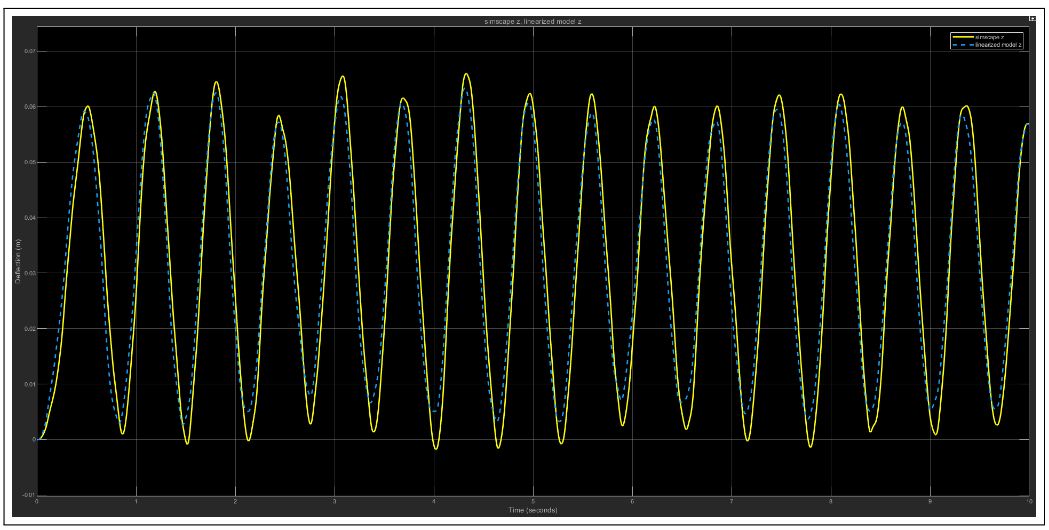

Figure 8: Comparison between the linearized model (dashed blue) and simscape model (solid yellow) for the random road profile input

Once again both our models are very close in terms of behavior which is what we would expect.

3 Task 3

As explained in Task 1, we will be designing a controller that will be able to handle the task of controlling our active suspension system. The main goal of the controller will be to maintain the ride height at the desired reference point set in advance. This will have as effect to improve riding comfort when traveling on uneven roads, mostly used in high end cars. In order to control this, we will be using two force actuators that will be used to counteract the road disturbances by applying force in a specific direction. We will monitor ride height at both end of the vehicle (which corresponds to z_{sf} and z_{sr}). It is critical that our sensors and actuators have high sampling rates in order to actuate the system more often which is needed for active suspension. The sensor shown figure 9 could be used for our system because it can have a very high sampling rate (up to 4kHz) but has also a very good resolution (0.02mm)

The force actuators can be of various nature (hydraulic, electro-magnetic, pneumatic, ...) each system comes with their pros and cons. In our particular case, we want a system that could be lightweight but that needs to be reliable. We can use the CAN Bus of the vehicle to send the informations from the sensors to the control unit to the actuators.

We will now start to design the controller for the linearized system. We have decided to go with an

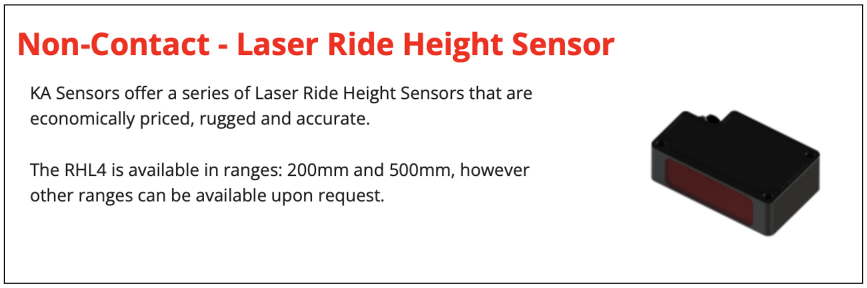

Figure 9: KA Sensor Non contact ride height Laser sensor

optimal control design technique which is the H-infinity loop shaping technique.

From the state space equation, we can plot the singular values of the system. From this plot, we will be able to add weights to the plant to have the desired singular value plot. We have decided to go with the following design choices :

- We will be aiming for a minimum bandwidth that ensures good performance of the system without asking for too much computing power from the control unit
- Steady-state error will not be tolerated, hence we will have to implement a PI weight which will also improve disturbance rejection
- We will implement a high-frequency filter weight to improve noise rejection

To start, we have decided to choose a target bandwidth of $30 rad/s$ which we could tune later on if we need it. We will not go into too much detail on the design process but if needed the reader can refer to the Matlab code where the design will be explained more in detail. We end up with the following singular values (figure 10) :

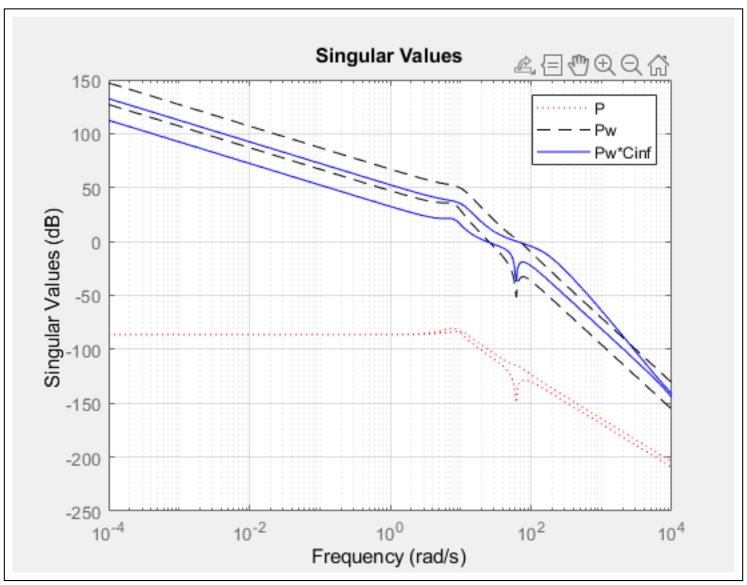

Figure 10: Singular values of the original (dotted red), weighted (dashed black) and final controlled system (solid blue)

From this graph, we can see that we significantly improved the response of the system by adding a PI weight which has an effect of raising the gain in the low frequency zone which will avoid steady-state error and improve disturbance rejection. Moreover, We improved noise rejection in the high frequencies. We end up with a system with a bandwidth of $23 rad/s$ which is pretty close to the design we were targeting.

4 Task 4

We have designed our controller, we now need to test the controller on the linearized system first, then on the real system. To assess the performances of the controller, we will be comparing the active suspension with the passive system, with the following simulink model (figure 11) :

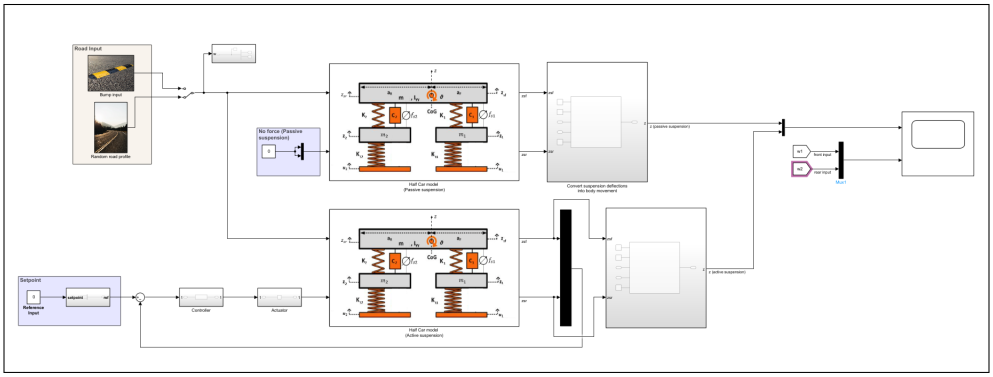

Figure 11: Simulink model used to assess the performances of the system

First, we will look at the response of both systems to the bump input. We get the following graph (figure 12) :

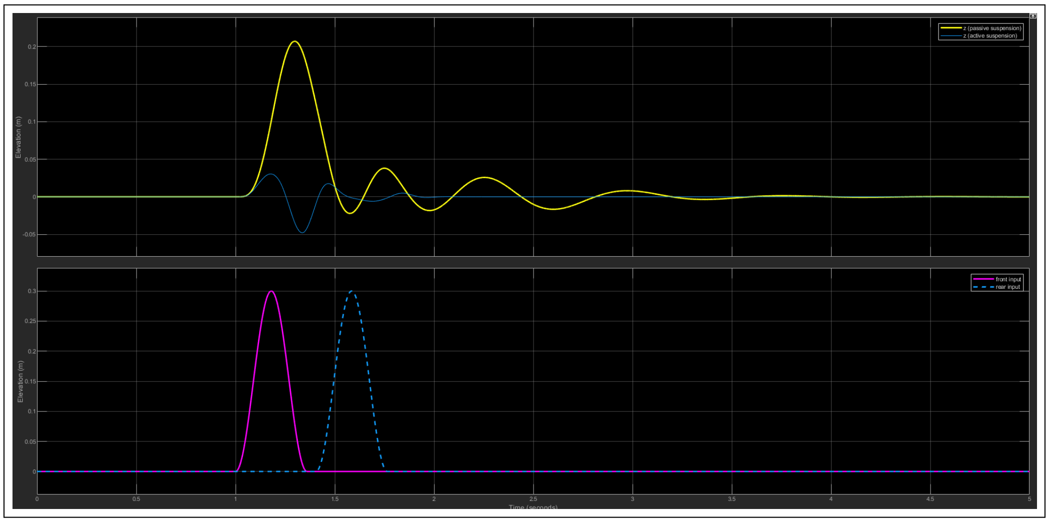

Figure 12: Response of both systems to a bump input

We can see the significant difference between each system but we can also quantify it :

- The peak value has been reduced by 77%
- The active system takes 1 second to settle whereas the passive takes 3 seconds

Both indicators we just saw which meet the performance criteria we gave before.

Even though we get very good performances with this system, we need to have a look to the control effort we are giving to the actuator (figure 13).

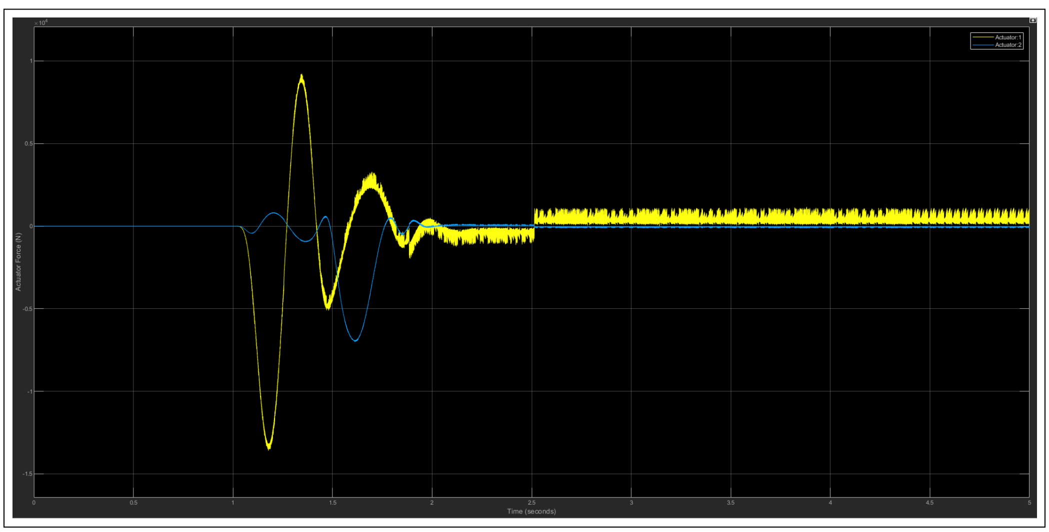

Figure 13: Actuator effort (at the front in yellow and at the rear in blue)

We can see with the graph (figure 13) that the signal is very noisy, which means that this could damage the actuator hence this is not acceptable. This can be solved by fine tuning the controller and reducing its bandwidth for instance. In our case, the actuator model is 1 we did not implement it because we did not have the time to model it. However, in the real simulation on simscape we will implement a saturation block which will represent the maximum output it can provide to avoid having unreal values.

Now we will be comparing both passive and active outputs with the random road profile input to make sure that our system works well (figure 14).

Once again we see that our system is very effective. With the passive system we have 0.03 peak-to-peak amplitude and with the active system we have a 0.013 peak-to-peak amplitude which represents an

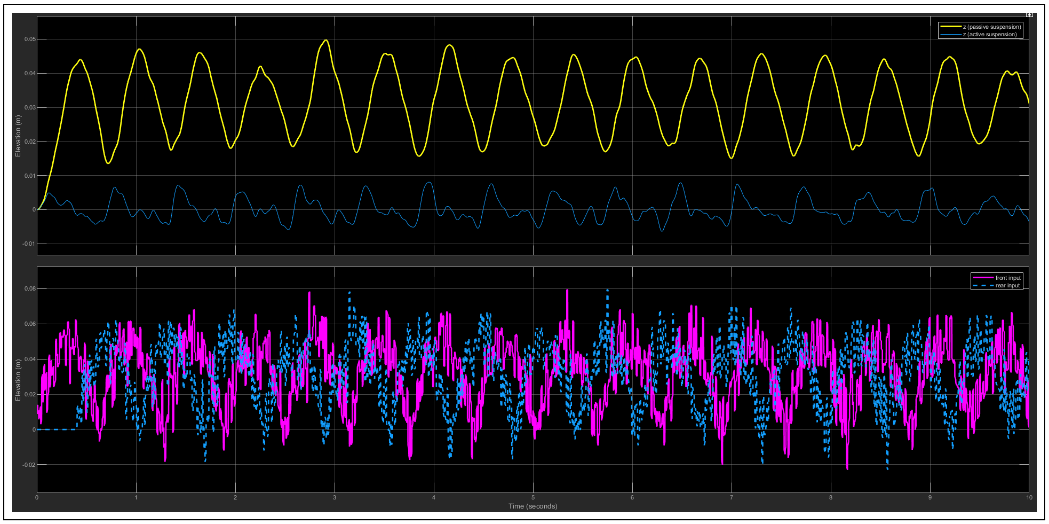

Figure 14: Response of both systems to random road profile input

improvement of almost 3times. However, we can see that our position has some undesired high frequency components which could be problematic because we also want to have the less acceleration as we could to avoid the passenger feeling the changes and hence express discomfort. There is a few action we can do to solve this issue.

- The first action is to lower the bandwidth of the system. This will mean that the system will be slower but will also have less variations at the output.
- The second action is to lower the frequency at which the high-frequency filter is applied. The noise will be attenuated more and we will get less variations hence less acceleration.

Now that we have designed a good-enough controller, we will implement it on the real simscape system to judge about its performances. Before simulating, we will be modeling the actuator as follow (figure 15) :

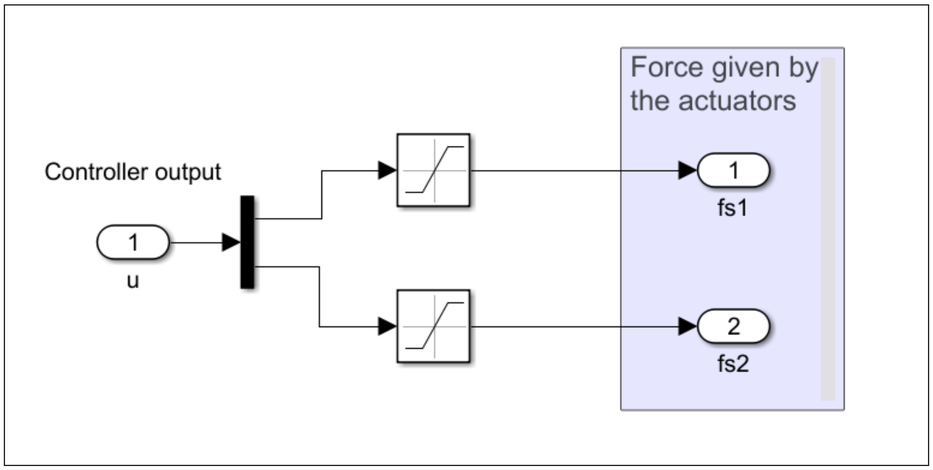

Figure 15: Actuator modeling

In this case, we have decided that the max output from the actuator would be $\pm 10kN$.

We can look at the effect of the controller we designed on the real system in response to a bump (figure 16) :

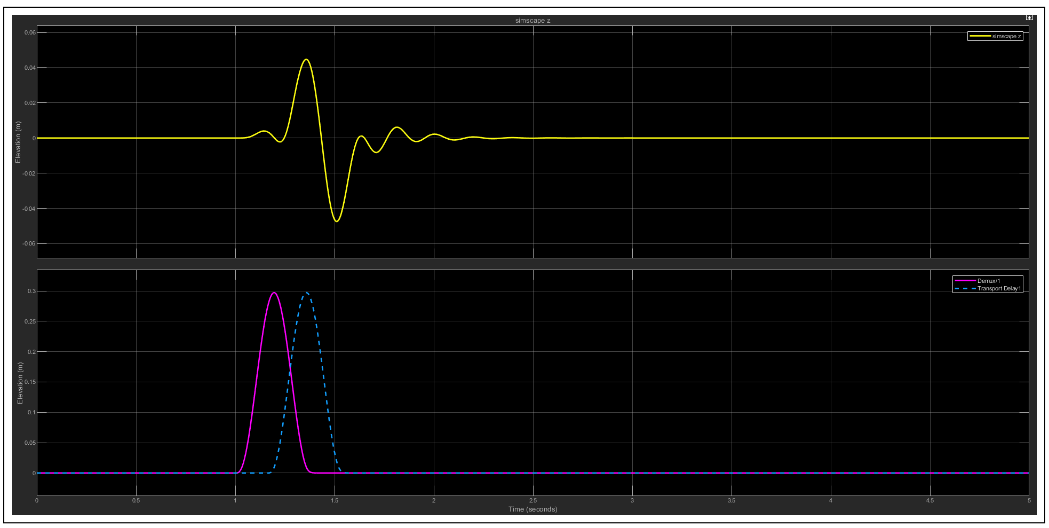

Figure 16: Response of the real system to a speed bump input

We can see that the amplitude variation is low, the body will travel only $10cm$ for a $30cm$ speed bump which is acceptable and fits the requirement. Now we will have a look at the random road profile input (figure 17).

There we have the same observation to make, the amplitude of the body displacement is very low compared to the dimensions of the obstacles on the road, we can hence say that our system works. However, we can notice a lot of variations in the position, which will result in big acceleration changes which is really not desirable. We will need to take care of this before implementing the system on a real car.

Even though our system has acceptable performances, it still needs more tuning and refining in the entire modeling. For tracks of improvements we have noted a few points :

- Model the sensor
- Model the actuator
- Fine tune the controller/change control technique

We have seen in the literature that a few controllers are often used for such active suspension systems, of which are fuzzy-logic based controllers, LQR, or even the basic PID.

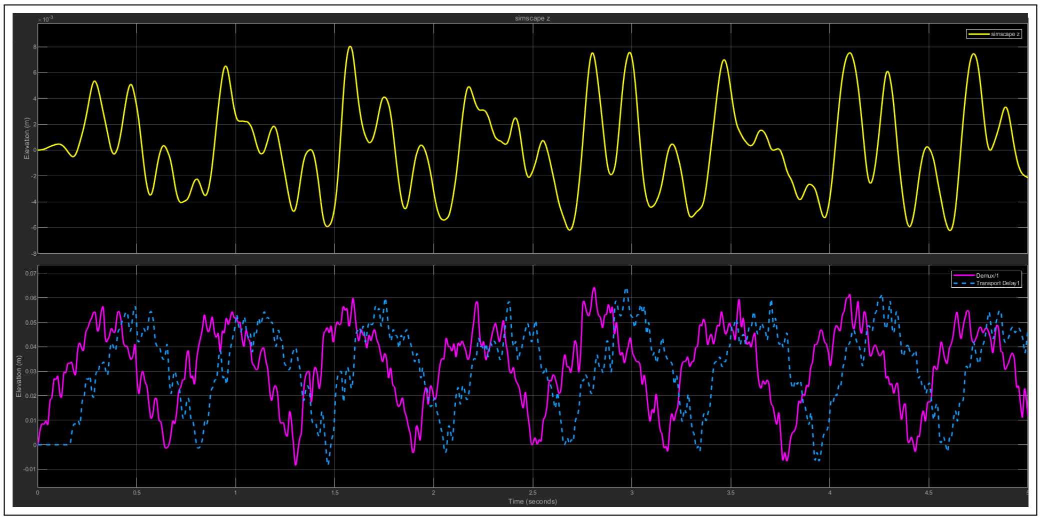

Figure 17: Response of the real system to the random road profile input

5 References

[1] Gandhi, Puneet & Sasidharan, Adarsh & Ramachandran, K.I.. (2017). Performance Analysis of Half Car Suspension Model with 4 DOF using PID, LQR, FUZZY and ANFIS Controllers. Procedia Computer Science. 115. 2-13. 10.1016/j.procs.2017.09.070.
[2] Kanjanavapastit, Apichan & Thitinaruemit, Aphirak. (2013). Estimation of a Speed Hump Profile Using Quarter Car Model. Procedia - Social and Behavioral Sciences. 88. 265-273. 10.1016/j.sbspro.2013.08.505.