Design and Control of a Ball and Plate Didactic Device

Jihad Alsamarji

Master thesis submitted under the supervision of

Prof. Dr. Ir. Micheal Kinnaert

Ir. Laurent Catoire

The co-supervision of

Dr. Ir. Christoph Mertens

In order to be awarded the Master's Degree in

Electromechanical engineering:

Mechatronics-Construction

Academic year

2018-2019

Acknowledgement

[Acknowledgement]

Abstract

This project tackles the design and control problem of a two DOF ball and plate system to build a commercial didactic device that helps young engineers to practice their control theory knowledge. Two sensing techniques are proposed to measure the displacement of the ball; a resistive touch screen or a camera with some image processing. The plant is designed with CAD and operated using an arduino. A computer is also connected to the plant for image processing using a GUI developed with python. The plant is modelled and simulated using Matlab and Simulink. Multiple controllers are compared and validated with the real system.

Contents

1 Introduction

Control system design is one of the toughest courses taught for young engineers where one combines abstract mathematical knowledge and analysis with practical engineering be it in mechanical, electrical, biomedical or other applications. An Engineering student often finds his ways into solving dynamic equations (differential equations) without a deep understanding of these equations and how they are coupled together. A great idea to understand control theory is by working on a physical plant and observe how do these equations manifest in the real world. However, this idea of learning by working on a real plant is often not feasible since it requires that the plant is affordable, available for the student to use, portable so that the student can carry it home to experiment on and safe. So the idea behind this thesis is to design and control a small physical plant that will act as a didactic device for students in pursuit of learning control theory.

1.1 Plant Requirements

The requirements of such a plant are listed as follows:

- Portable : The ability to easily deploy or pack the plant, which means that the plant needs to be lightweight, non fragile, easily assembled and disassembled.
- Safe : The ability to be operated without any risk of harm nor damaging the equipment by choosing extreme control parameters.
- Affordable : The plant should be affordable for students and therefore the hardware selection and manufacturing should be optimal for its usage.
- Controllable : The plant should be designed in a way that allows an external

input to move the internal states of the system from any initial state to any other final state in a finite time interval. This is valid when the controllability matrix $R = [\ B\ AB\ A^2B\ ...\ A^{n-1}B]$ is full rank (rank(R) = n) where n is the number of the states. Theoretically, With such an ability, one can arbitrary alter the system's eigenvalues[1] by an input $u = -Kx$ which gives $\dot{x} = (A - BK)x$, where the control gain matrix K can be chosen arbitrarily for pole placement. A controllable system in continuous time is equivalent to a system with a full reachability thus with a reachable set $\Gamma = \mathbb{R}^n$. However, in practice, the actuators should have a sufficient actuating range to ensure controllability. [example : if one linearizes the system of an automobile and find out that it is controllable, it doesn't mean that the automobile can reach a far destination with a blink of eye by using a controller with an enormous gain, because in reality, the actuators are saturated with certain limits and hence a non linear system.]

- Robust: The plant should be stabilizable with a reasonable gain and phase margin, meaning that with any small change to the transfer function of the system, it can still achieve stability by the same controller. In other words, using the Nyquist stability criterion, the further away our system's Nyquist plot is from the point (-1,0.j) the more it is robust to model uncertainty, delays, higher control gains, noise and disturbance.
- Scalable : the plant could be targeted to an educational institute as well where the required size of the plant is to be bigger. Therefore, the plant should be able to be scaled mechanically and electrically with no major changes to the plant's dynamics and therefore to its control schemes. Fur-

[1] there is a link between the eigenvalue of state matrix A and the poles of the corresponding transfer function. All the poles are eigenvalues of A, but the vice versa is not true since cancellations with the numerator can happen. [1]

thermore, the plant should support a wide range of controllers from classical to advanced, modern and optimal control based on user experience.

- Convenient : The main purpose of the plant is to be didactic and targeted for young engineers, therefore, it should be simple and self explanatory with no abstractions. Usually mechanical systems tend to have less abstractions since you directly feel and see the change of any control parameter and you witness the manifestation of the governing equations in real life.

1.2 Plant Selection

Now that the plant's constraints are determined, several choices can be made. In fact, the Ball and Plate is a perfect candidate since it's unstable by default and tricky to control. This plant gives space to apply many control methods and results in different behaviors which are interesting to be analyzed and compared.

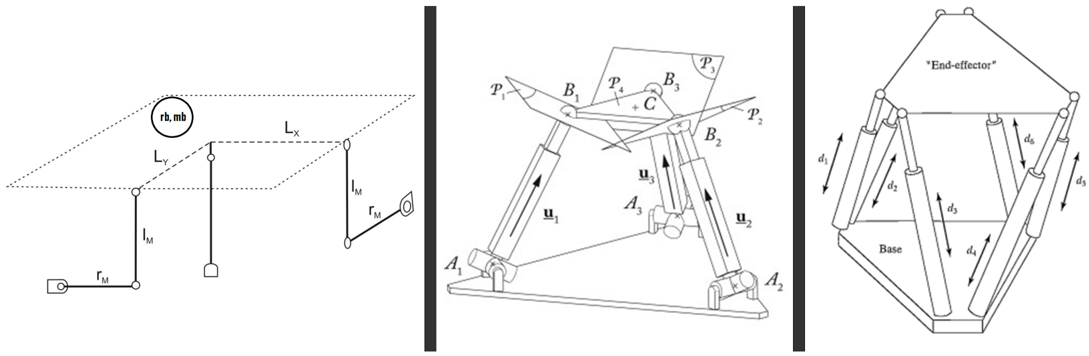

Figure 1: A demonstration of different Ball-Plate plants. From left to right: 2DOF, 3DOF and 6DOF

There exist many forms of the Ball-Plate plant which differ by the number of degrees of freedom (Some of these Plants are shown in fig.1). The objective is always to control the position of the ball at least in the X and Y directions. The system is MIMO[2] since it requires two states to be measured and two actuators to be controlled.

In a 3 DOF Ball-Plate, at least 3 actuators are needed. Each combination of these 3 driving angles will result in an inclination around the X and Y axis and a displacement in the Z direction which is often ignored. Since the resulting inclination is a function of all actuator angles, the outputs are coupled and this can be seen in the system's governing equations.[6]

Remark: *In the 2 DOF Ball-Plate, each actuator is responsible for one degree of freedom because the ball motion in the X and Y directions are decoupled. Therefore, decentralized control can be used by treating this MIMO system as two separated SISO systems where each direction is actuated, measured and controlled separately. (more on that later)*

For the sake of simplicity and affordability, the 2 DOF plant is chosen for this study.

[2] Multiple-input and multiple-output

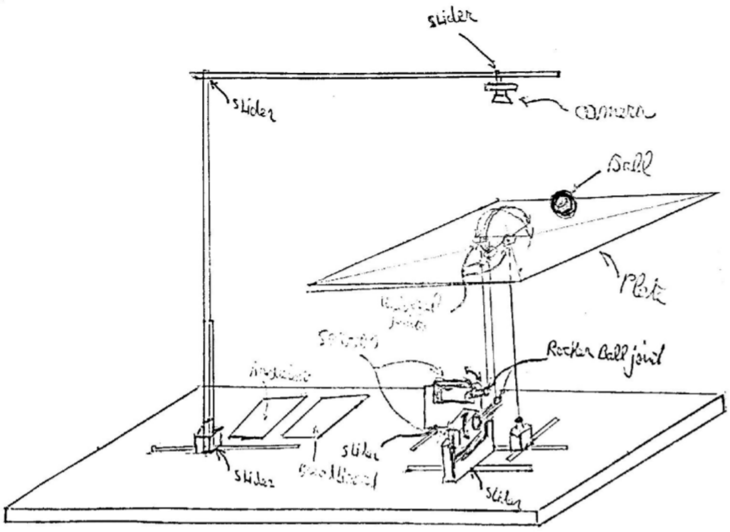

Figure 2: A sketch of the 2DOF Ball-Plate prototype done prior to the design stage

1.3 Plant Description

Ball-Plate is a system that holds balance of a ball on a plate or board. The actuators allow the plate to be oriented with a certain angle of inclination to counterbalance the motion of the ball. A sensing device should be able to extract the position of the ball in the two axes and feed it to the controller which will determines the position, speed and acceleration of the ball and therefor process the right command for the actuators. The controller will also counter act the destabilization of the ball by any disturbance be it an external force or noise in the system. A first sketch of the prototype is drawn in fig.2.

2 Design

2.1 Hardware Selection

Before proceeding to the mechanical design, one should plan ahead on which hardware is suitable for such a device and then design accordingly.

2.1.1 Actuators

Knowing that our plant is 2 DOF, the Plate is going to be tilting around it's pivoting point in the center. There are two actuators to be placed perpendicularly to tilt the plate around the two axes (X and Y). Based on their distance from the center, the ratio between the angle of the plate and the angle of the actuators can be determined. However, to fully utilize the actuators, only a range of 180 degrees is needed. To achieve zero steady state error, it is crucial that the actuators are accurate. Any error in the angle of the actuator will lead to an error in the angle of the plate. The ball is sensitive to roll over uneven surfaces, meaning that with an inaccurate actuator, a steady state is hard to be achieved.

The two most suitable actuators for such a plant are the Servo motor and the stepper motor.

- Stepper motor : often comes without an encoder nor a drive and therefore they are needed to be bought separately, installed and controlled. Steppers exist in 4,6 or 8 wire configurations. All of the stepper wires are to power the different coil pairs. Different powering of the coils will result in different rotation direction and step accuracy. The frequency of powering these coils determines the step speed. Manually managing the powering of the coils is troublesome and requires complex circuitry, that is why a stepper drive is necessary for such purpose. Even a stepper motor can miss some steps and that is why an encoder is used to feedback and eliminate error. The torque-speed curve of the stepper motor shows that the torque decreases with an increase of speed. To add things up, steppers consume current continuously even when idle unless it is controlled not to do so.

- Servo motor: is nothing but a DC motor attached to a gear box for speed reduction and higher torque. It comes packed with an internal circuitry that contains an encoder and a controller. It is only powered by two cables and commanded via a third cable in the form of PWM (pulse width modulation). Servos have almost a flat relationship between torque and speed. Servos only consume current when needed which enables it to run much cooler.

To summarize, Servo control systems are best suited to high speed, high torque applications that involve dynamic load changes. Stepper control systems are less expensive and are optimal for applications that require low-to-medium acceleration, high holding torque and the flexibility of open or closed loop operation.[4]

For the Ball-Plate, high speed actuation is needed to cope with the reference tracking or else there will be delays in the system which makes the plant uncontrollable in practice. Since the actuators are going to be varying their rotating speed continuously, a flat torque-speed behavior is needed. Using Steppers will complicate the circuitry, take much more space, consume more power and cost more in case of closed loop operations. With that being said, servo motors are chosen as actuators for this study.

Due to the need of a High speed and moderate torque, a standard sized Servo motor will be chosen in the next section after revealing some additional mechanical constraints.

2.1.2 Sensors

Only one displacement sensor is needed for a full state estimation (proven in the observability study done in section 3.3). A reliable sensor measurement is of utmost importance since any noise, inaccuracy or processing time delay, will induce instability. Two ways to detect the displacement of the ball are proposed.

- Camera : a camera can be placed on top of the plate to monitor the upper side of the plate. Of course, a bare image will not carry any useful information of the ball and therefore, the image should be processed either by movement detection or by color filtering. Even though this will require a back-end for image processing and hence delay, the readings will be extremely accurate and noise free.
- Resistive touch screen : another way of measuring the displacement of the ball is to place a resistive touch screen on top of the plate that measures the displacement of the ball due to the ball's touch pressure. This solution is definitely simpler since it doesn't require any processing. However, the touch screen has a certain touch pressure threshold to start reading inputs. This threshold is something not to be underestimated since the touch pressure of the ball will decrease when the ball is moving or even worse if the plate is vibrating. Vibration causes the ball to be lifted from the screen's surface which makes it undetectable anymore. On the other hand, the touch screen is going to have noisy readings that need to be filtered since it cause additional vibrations specially from the derivative gain in the controller. (more on that later)

Even with the difficulties of the resistive touch screen option, both of these sensing methods will be used and compared later in the study.

2.1.3 Processing Unit

Now that the actuators and sensors are known, a suitable processing unit can be chosen to act as the controller. For the sake of simplicity and since this study focuses only on the design of the prototype and not the final product, a commercial controlling unit will be used instead of a custom built PCB. For this purpose, the Arduino Uno is enough for the task.

The Arduino boards are mainly composed of a microcontroller around which there are many input and output pins that allow the board to interact with external electronics. Several models exist, the major difference that separates them is the number of i/o pins. Another difference is in the memory size of the microcontroller. For example a microcontroller contained in an Arduino Mega will contain an EEPROM with a capacity of 4kB while an Arduino Uno will have an EEPROM limited to 1 kB. These cards allow anyone to design a project at a lower cost. An Arduino Uno costs around 20 euros which makes these cards accessible. Another advantage is the huge open-source community, where people post tutorials and help with projects on the internet. Furthermore, Arduino also provides free programming software that allows to easily program any Arduino or other cards with a similar microcontroller. Indeed the Arduino software can also be used to program a custom made card with the same bootloader. This is an example of one of the many benefits that open-source software offer.

Knowing that the arduino is not suitable for image processing, a computer will be attached to the plant only in case of using the camera. However, the Arduino is always going to be used to send PWM signals to the servos.

2.2 Mechanical Design

There is no perfect design. Designing is an iterative procedure to reach a better product, It begins by fixing assumptions and determining the unknowns. Based on the availability in the market, ease of manufacturing, cost and efficiency, the following assumptions are taken :

- Size of the actuated plate (30x30cm is a reasonable space to experiment different ball movement patterns and tolerate a bit of overshoot)
- Height of the actuated plate (27cm is comfortable to reach and doesn't require a tall camera pole).
- Size of the supporting bench (50x30cm is enough to contain all of the plant).
- Size of the servo frame (Standard size is used to match the selected servo motor in the previous section) .
- Material and thickness of the plate and bench (4mm Ply-Wood is used since it is available and easy to be cut).
- Carbon fibre is used for the actuating rods and the supporting structure of the camera.
- PLA is to be used for 3d printing of additional supporting structure.

2.2.1 Reasoning

- The joints between the servo arms and rods need to convert the rotation motion into a linear motion while keeping the rods free to rotate. This can be achieved by a rod end bearing joint or even better handled with the rocker ball joint to compensate for any misalignment during motion.

- Another special joint needs to be placed between the rods and the plate to allow the latter to be pivoting around 2 axis while connected in 3 places. This can be done by a universal joint or a magnetic ball joint.
- The distance d_c and d_m shown in fig.7 are very important since the ratio $\kappa = \frac{d_c}{d_m}$ defines the relationship between the angle θ of the servos and the angle α. The bigger κ is, the bigger the range of θ is needed
- To leave some flexibility in the assembly phase, some slide tracks are cut in the bench that allow for adjustments by sliding of the camera pole, servo supports and central plate pole.

Now that the mechanical design is more detailed, the plate is going to be supported in 3 places. The plate is going to be made out of ply wood. Assuming an iron solid ball of 0.1 kg to be rolling on the surface. Assuming a critical situation where the total weight is handled by one servo arm. Roughly speaking, a torque of $8N.cm$ is needed. The "Turnigy™ TGY-4409MD" was chosen as the actuator since it packs the torque needed, fast enough and relatively cheap.

TGY-4409MD	4.8 V	6.0 V
Speed (in second / 60°)	0.13	0.11
Torque (in Kg.cm)	8.65kg.cm	9.45kg.cm

Table 1: TGY-4409MD specs

Remark: *As a matter of fact, servo motors shouldn't be chosen only by their rated specifications. For this kind of application, their control circuitry is of much importance since a high speed high torque servo is useless if it takes time to respond. Not just response time but also the accuracy plays a big role in stabilizing such a plant.*

2.3 Electrical Design

The circuits used for each of the sensing methods are rather simple as discussed in the hardware selection section.

2.3.1 Resistive touch screen

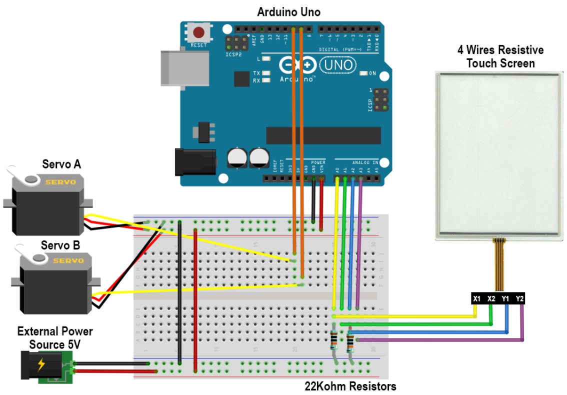

Figure 3: Electrical wiring of the touch screen sensor

As shown in fig.3, the arduino is fed from the same power source of the servos(given that the power source can supply the maximum current drawn from all of the component or else the circuit will not work). The touch screen used in this project contains 4 wires (X1 X2 Y1 Y2), X1 and X2 are connected inside the touch screen and the same goes for Y1 and Y2. The connectivity inside the screen is a resistive one as shown in fig.4.

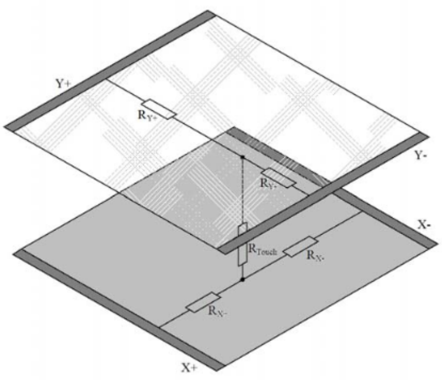

Figure 4: Touch screen inner circuit taken from [7]

The 2 connections are shorted whenever pressure is applied, this will create a voltage divider. The horizontal and vertical coordinates of a touch can be read in two steps. First, Y1 is driven high, Y2 is driven to ground, and the voltage at X1 is measured. The ratio of this measured voltage to the drive voltage applied is equal to the ratio of the vertical coordinate to the height of the touch screen. Horizontal coordinate can be similarly obtained in the opposite manner. Both measurements can be shown in fig.5.

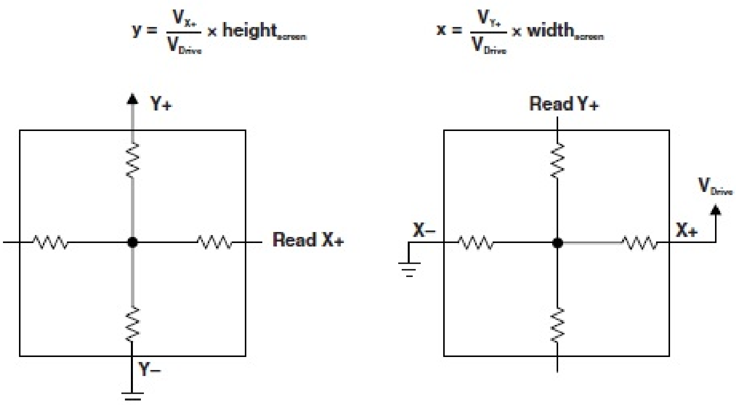

Figure 5: Pin states for reading each coordinate.

Therefore both measurements cannot be taken in the same time. The Arduino will flip the pins whenever a reading command is signaled. Lastly, the resistors hooked in 3 are pull-down resistors to pull the voltage down to ground if there was no touch.

2.3.2 Camera

The circuit used with the image processing is shown in fig.6.

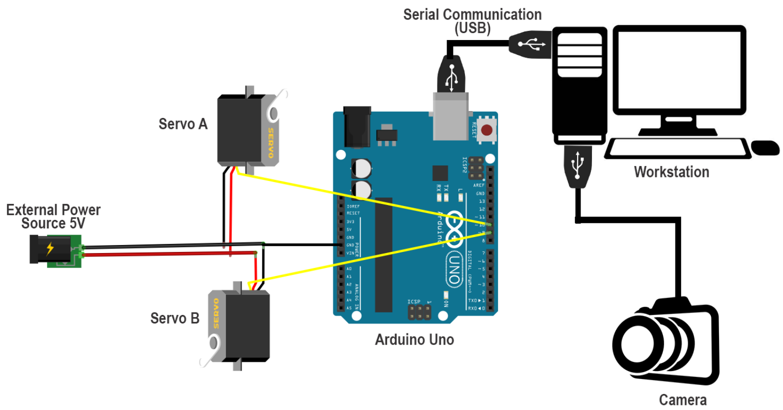

Figure 6: Electrical wiring of the camera sensor

In this setup, the serial connection between the Arduino and the computer is to receive command signals and power the Arduino in the same time. The camera is a digital sensor that sends a picture every sampling time $t_s = 0.033s$. The picture is then processed to find the position of the ball in pixels. Based on the chosen control law, the computer will give a command signal in the form of an angle θ for each servo. The arduino will translate the angle into a PWM[3] signal for each of these servos.

[3]Pulse width modulation

2.4 Software

In this section, a brief methadology of the software design is going to be explained. However, The full software implementation is shown in the appendix.

2.4.1 Software for the touch screen

By using the touch screen, a simple C code can do the job using the following alogrithm:

3 Simulation and Control

Before doing any control on the real plant, the system should be modelled and simulated on Matlab to fully understand the dynamics evolving around it to have an idea of what to expect in reality and what control method should we use to achieve our requirements. Modeling and simulation are important to solve problems safely and efficiently, It allows us to analyze and easily validate the theory with the real plant. Even though the Ball-Plate is safe to be operated even with false parameters, as an engineer, one should always model the problem at hand and avoid the trial and error approach. In other words, modelling could save money and time, avoid risks, gives better visualization of the dynamics, solves directly for the optimal result, handles uncertainty. For this purpose, Matlab and simulink are going to be used.

3.1 System Dynamics Modelling

For the sake of simplicity and since our problem can be decoupled by separating the X from the Y direction, only one of these direction will be studied (fig.7 shows the model in the X direction).

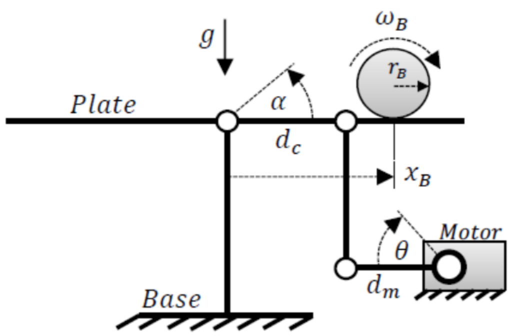

Figure 7: system dynamics

The mathematical formulation will only be conducted on the X direction since the Y direction is the same. Equation of motions can be obtained using the classical approach by working on the free body diagram or by using the Hamiltonian, Virtual work or Langrangian formulation. For the sake of simplicity, the Lagrangian formulation was used in this study:

$$\frac{\mathrm{d}}{\mathrm{d}t}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = Q_i \tag{1}$$

Where L is the langrangian and Q_i is the external forces acting on the system generalized coordinate q_i, in this case q_i will be the x coordinate.

The lagrangian L can be written as :

$$L(\dot{q}_i, q_i) = T(\dot{q}_i, q_i) - V(q_i) \tag{2}$$

Where T is the system total kinectic energy and V the total potential energy. Both of them are in function of the generalized coordinate q_i which is x.

Now, we can seperate T into the linear component T_l and the rotational component T_r Component:

$$T_l = \frac{1}{2} m_b \dot{x_b}^2 \tag{3}$$

$$T_r = \frac{1}{2} I \omega_b^2 \tag{4}$$

Where m_b is the mass of the ball, x_b is the ball's displacement, , I is the ball's moment of inertia and ω_b is the angular velocity of the ball. Note that I depends on the shape of the ball where $I = \frac{2}{5} m_b r_b^2$ in case of a solid sphere and $I = \frac{2}{3} m_b r_b^2$ in case of a hollow sphere.

by replacing ω_b with $\frac{\dot{x_b}}{r_b}$, the resulting total kinetic energy will be :

$$T = \frac{7}{10} m_b \dot{x_b}^2 \tag{5}$$

Solid sphere

$$T = \frac{5}{6} m_b \dot{x_b}^2 \tag{6}$$

Hollow sphere

Now for the potential energy:

$$V = m_b g . x_b sin(\alpha) \tag{7}$$

Where g is the gravitational acceleration and $x_b.sin(\alpha)$ is the x component of the gravitational force (weight).

To get back into solving the Lagrangian equation (1), the only composite force actuating on the q_i is the friction F_x which can be written as $Q_i = F_x = -f_c\dot{x}_b$ and hence:

$$\frac{7}{5}\ddot{x}_b + \frac{f_c\dot{x}_b}{m_b} + g\sin\alpha = 0 \tag{8}$$

Equation of motion of the ball in case of Solid sphere

$$\frac{5}{3}\ddot{x}_b + \frac{f_c\dot{x}_b}{m_b} + g\sin\alpha = 0 \tag{9}$$

Equation of motion of the ball in case of Hollow sphere

The obtained equation will be nonlinear (includes a trigonometric term). Therefore, it will have to be linearized around the equilibrium[4] point.

Bear in mind that the input of our system is not the angle α, it is the reference angle of the servo θ_r. A generic second order relation is assumed for the servo-motor dynamics in:

$$\ddot{\theta} + 2\zeta\omega_n\dot{\theta} + \omega_n^2\theta = \omega_n^2\theta_r \tag{10}$$

Where ω_n is the motor's natural frequency ζ is motor's damping ratio and ω_r is the reference servo angle.

Remark: *assuming second order dynamics to the servo motor and assigning the parameters ζ and ω_n arbitrarily based on the rated specifications is a big step toward system uncertainty. Therefore, system identification is a great method to capture the real dynamics of the system. This will be done in the next section.*

To connect the equation of the servo motor with the equation of motion of the

[4] An equilibrium point is a point such that, in the ideal case, maintaining constant the input, the state "does not move". Note that in real nonlinear systems there may be several points of equilibrium.[1]

ball, there is a relation between the angle α and θ:

$$d_m \sin\theta = d_c \sin\alpha \tag{11}$$

where d_m is the length of the servo arm and d_c the distance between the center of the plate and the linking arm. this can be reduce by using $\kappa = \frac{d_c}{d_m}$.

$$\sin\theta = \kappa \sin\alpha \tag{12}$$

Remark: *the latter relation between α and θ is obtained assuming that the linking rod is quasi vertical, meaning that the angle ψ in fig.8 is zero. This is not the case in real life, therefore a position kinematic analysis will be done.*

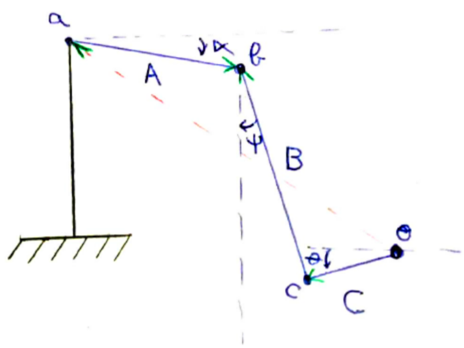

Figure 8: structural constraints sketch

Looking at fig.8, the structure is a 4 bar mechanism, so the constraint equation is obtained from the requirement that the distance B between the moving pivots a and b is constant throughout the movement of the linkage. The symbols A,B,C are used as lengths in this section and they will be used later for other purposes in the state space matrix. The constraint equations are :

$$\vec{oc} + \vec{cb} = \vec{oa} + \vec{ab} \tag{13}$$

projecting this equation on the horizontal and vertical planes gives:

$$\begin{bmatrix} -C.\sin\theta \\ -C.\cos\theta \end{bmatrix} + \begin{bmatrix} B.\cos\psi \\ -B.\sin\psi \end{bmatrix} = \begin{bmatrix} B \\ A+C \end{bmatrix} + \begin{bmatrix} A.\sin\alpha \\ A.\cos\alpha \end{bmatrix} \tag{14}$$

For a given angle θ, solving for α by hand can be troublesome. therefore, an analytical solution is hard to reach. By using the Simscape library in Simulink, we can build the geometrical structure and plot the numerical relationship between α and θ.

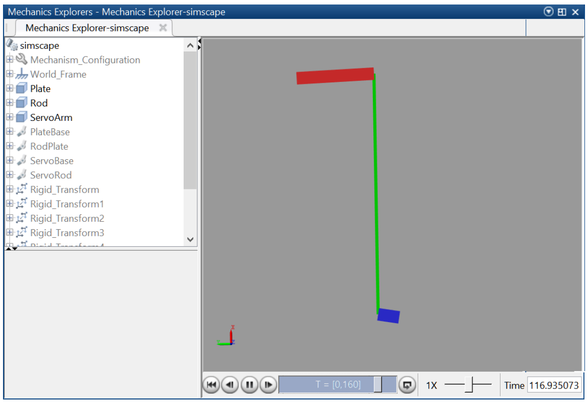

Figure 9: Motion animation of the model using Simscape

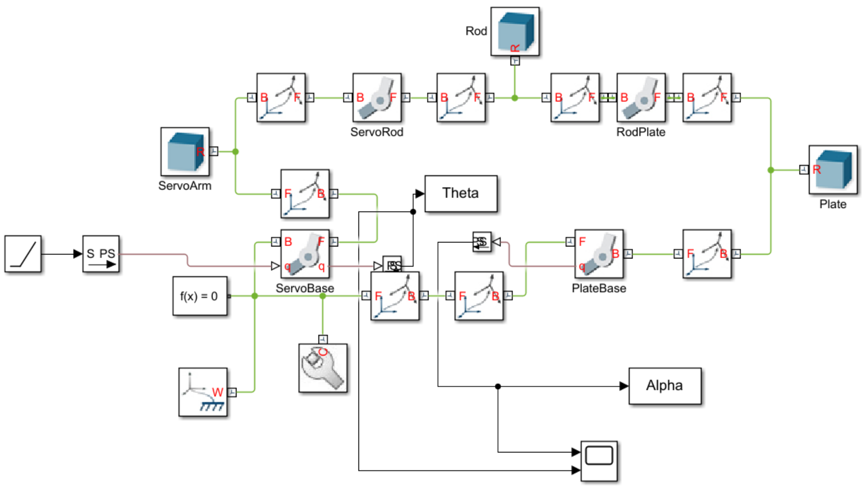

Figure 10: Simscape Block diagrams of the model in Simulink.

After obtaining the analytical solution for the relation between α and θ, a comparison with the assumption in equation(11) is shown in fig.11

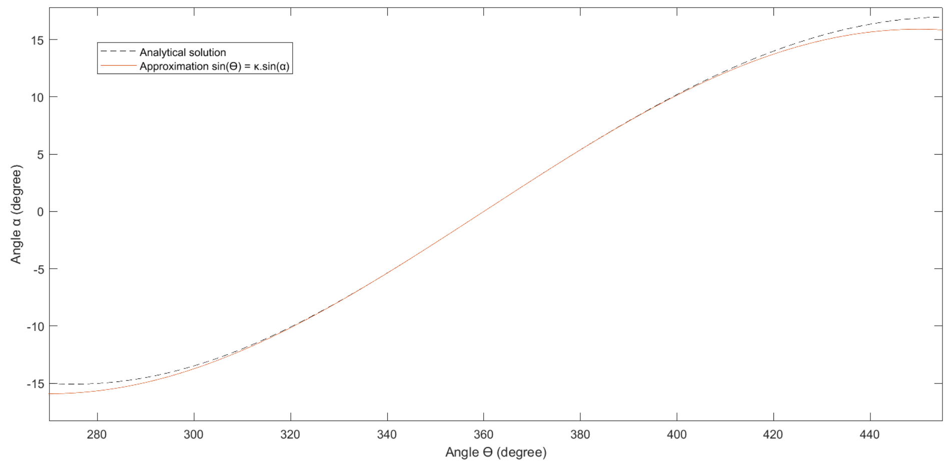

Figure 11: Comparison between the analyitical solution and the vertical rod assumption

It can be said that the assumption is a very close approximation throughout the operating region. Therefore, the relation $d_m \sin\theta = d_c \sin\alpha$ will be adopted in this study.

As discussed earlier, the trigonometric terms needs to be linearized around the state of equilibrium in order to come up with a control design, however, we can use a look up table to map the relation between α and θ upon using the real plant. The angle α is designed to be contained between -15° and $+$15° which means that $\sin\alpha$ can always be linearized around the point of equilibrium 0° which further means that $\sin\alpha \approx \alpha$. The same linezarization goes for the servo motor angle θ around 0° where $\sin\theta \approx \theta$, however this relation only holds between -15° and $+$15° and other linearizations need to be done to cover the whole range of the servo's operating angle for a more accurate modeling.

$$\theta = \kappa\alpha \tag{15}$$

this will give us the equation that will be used to connect the system's input α_r to the system's output x_b.

$$\ddot{\alpha} + 2\zeta\omega_n\dot{\alpha} + \omega_n^2\alpha = \omega_n^2\alpha_r \tag{16}$$

Remark: *notice that the plate's equation of motion copied the servo motor dynamics without adding any kind of friction due to the joints or vibrations due to the stiffness of the rods. This is also an uncertainty that will be validated in the next section.*

Now that the fundamental equations and relations are all at hand, the state space[5] of the continuous linear time invariant[6] system can be constructed using the general representation :

$$\dot{q}(t) = A.q(t) + B.u(t) \tag{17}$$

$$y(t) = C.q(t) + D.u(t) \tag{18}$$

where $q = [\ x_b\ \dot{x}_b\ \alpha\ \dot{\alpha}]^T$ are the system's states and $u = \alpha_r$ is the system's input and $y = x_b$ is the output. the matrices A, B, ,C and D are shown in the

[5]Local input/state/output representation

[6]The system is time invariant to any temporal translation. (or equivalently the output doesn't depend explicitly on the time)

following:

$$\begin{bmatrix} \dot{x}_b \\ \ddot{x}_b \\ \dot{\alpha} \\ \ddot{\alpha} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-5}{7}\frac{f_c}{m_b} & \frac{-5}{7}g & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\omega_n^2 & -2\zeta\omega_n \end{bmatrix} \cdot \begin{bmatrix} x_b \\ \dot{x}_b \\ \alpha \\ \dot{\alpha} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & \omega_n^2 \end{bmatrix} \cdot \begin{bmatrix} \alpha_r \end{bmatrix} \tag{19}$$

$$\begin{bmatrix} x_r \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_b \\ \dot{x}_b \\ \alpha \\ \dot{\alpha} \end{bmatrix} + \begin{bmatrix} 0 \end{bmatrix} \cdot \begin{bmatrix} \alpha_r \end{bmatrix} \tag{20}$$

3.2 System parameter identification

Bear in mind that this section is only purposed to choose and validate the system's parameters and not to totally identify the system as a black box. It's more like a grey box approach where we already know the theoretical model of the system and we're interested to notice the differences with the real plant.

Without this section, the parameters that should have been used are shown in table.2 and table.3.

The shown parameters, are all measured except for ω_n and ζ were assumed based on the servo's rated specifications. Since the servos are going to be loaded and attached to the plate, the assumed parameters cannot be trusted and therefore needs validation.

Since the point of interest lies in validating the equation governing the dynamics of the plate due to a reference angle α_r as an input, a good idea is to apply system identification to get the transfer function regarding equation (14). The procedure requires a sensor to measure the angle α resulting from an input α_r.

Symbol	Value
ω_n	$19rad/s$
ζ	1
f_c	$0.001Ns/m$
g	$9.81m/s^2$
d_m	$0.017m$
d_c	$0.062m$

Table 2: Theoretical system parameters

Configuration	Description	r_b (m)	m_b (kg)
1	Hollow steel	0.016	0.013
2	Solid Iron	0.01	0.046
3	Hollow Ping Pong	0.02	0.001
4	Solid Marble	0.007	0.0049

Table 3: Ball configurations

Measuring the angle α requires a potentiometer to be attached on the central pivoting point of the plate. This measurement method is time consuming and requires a clock module to be connected on the arduino to know the absolute time of the measurement and sync it with the input. A simpler way is to place a smart phone on top of the plate and connect it to MATLAB by wifi and capture the readings of the phone's gyroscope to get the angle α. To sync the input signal with the measurements, the absolute time of both signals is known and can be matched using simple MATLAB matrix manipulations. A sampling frequency of $90Hz$ was used.

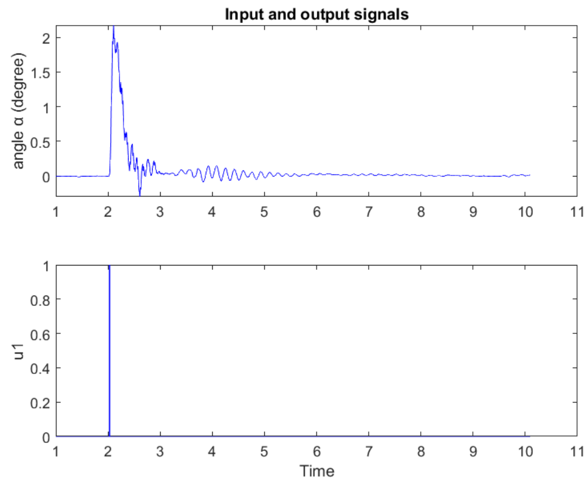

Figure 12: Impulse response

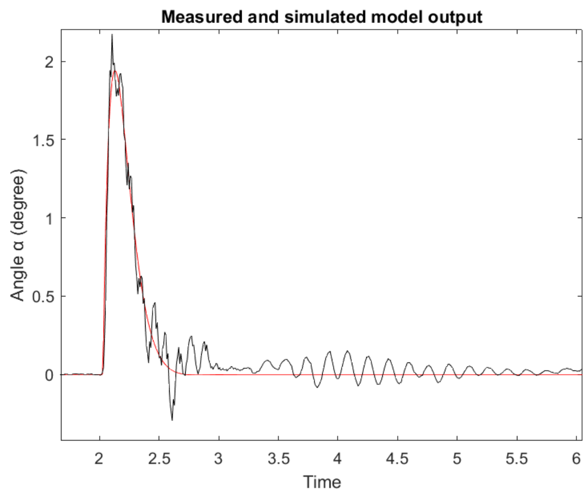

Figure 13: System Identification using impulse response.

As a first attempt, the impulse response was captured in fig.12 and then identified as a second order system in fig.13.

Remark: *The latter observation gives us a possible way to identify the transfer function of an unknown linear system in the case we are able to measure re-*

liably the impulse response ! It must be remarked that this method is usually quite fragile[1]. Indeed it is very hard to generate an infinitesimally short signal and capture a very small measurement where the noise of the sensor can be dominating.

Due to the fragility of the first attempt, a second try was done using the step response as shown in fig.14

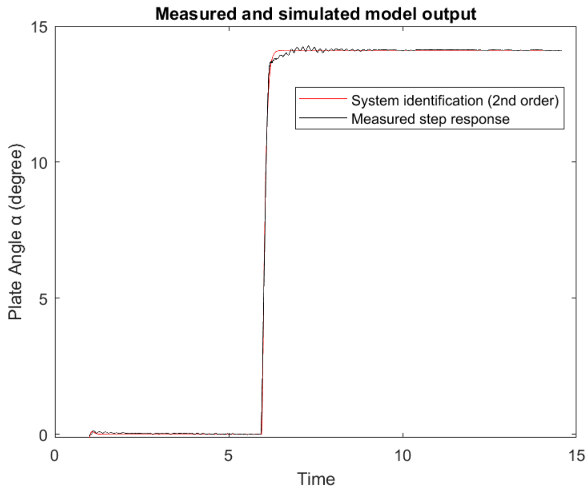

Figure 14: System identification using step response (a step of 80 degree is used here to reach the maximum actuator angle).

for the sake of completeness, a third attempt of identification was done with the unitary step response.

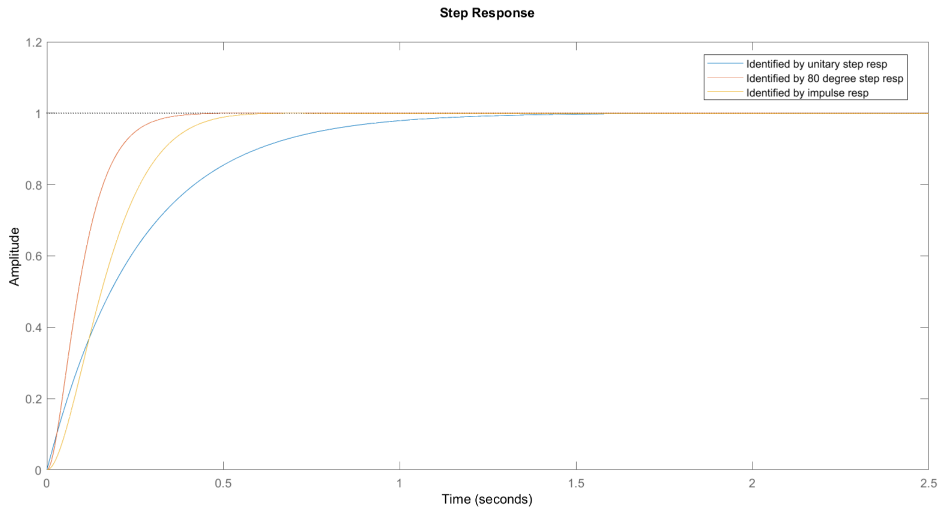

Figure 15: Unitary Step response of the three identifications.

<u>Remark:</u> *Notice in fig.15 that each of the attempts identified a different transfer function for the same system. If other types of input signals like a triangular or sinusoidal signal was used, other aspect of the non linear plant will show. This difference is not just due to the unreliable measurement and sensor noise, it is also because the behavior around the origin is different than around a high operating point. This latter observation is not only because of the structural non linearities but also the servo's internal controller that is more sensible to big references than a small one.*

The system can be trimmed into different regions of linearization and apply different control laws in each region. However, for the sake of simplicity, the identification used for the 80 degree step response is going to be used.

Obtaining the transfer function in the form of $\frac{C.\omega_n^2}{s^2+2.\zeta.\omega_n.s+\omega_n^2}$, one can directly determines the parameters ω_n and ζ to be successively $0.18rad/s$ and 1.

Remark: *Note that the retrieved parameters are almost the same as the parameters assumed without measurements, this is a good indication that the real plant is built correctly.* ***However****, all of the perturbation in the signal was treated as a sensor noise and smoothed out, but in reality, a big portion of it is due to the vibration. Not including these vibrations in the equation of motion is like ignoring some of the system's dynamics and might cause problems later on in the final validation test.*

3.3 Control system design

3.3.1 States of equilibrium:

Reminder: *A state of equilibrium with respect to a constant input* $\bar{u}$ *is a configuration* x_e *of the system such that:*

- $\bar{f}(x_e, \bar{u}) = 0$ *in continuous time*
- $\bar{f}(x_e, \bar{u}) = x_e$ *in discrete time*

where:

- $\dot{x} = \bar{f}(x, u)$ *in continuous time*
- $x(t+1) = \bar{f}(x, u)$ *in discrete time [1]*

In order to determine the states of equilibrium of our system :

$$\begin{bmatrix} \dot{x_b} \\ \ddot{x_b} \\ \dot{\alpha} \\ \ddot{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{21}$$

$$\begin{bmatrix} \dot{x}_b \\ \frac{5}{7}(\frac{-f_c}{m_b}\dot{x}_b - g\sin\alpha) \\ \dot{\alpha} \\ -2\zeta\omega_n\dot{\alpha} - \omega_n^2\alpha \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{22}$$

by solving this equality, we will find that there is an infinite states of equilibrium where x_b can be any real number while all the other states are equal to zero. In other words, $\bar{x}$ is a state of equilibrium if the following is true: $(x_b \in \mathbb{R}^n) \cap (\dot{x}_b = 0) \cap (\alpha = 0) \cap (\dot{\alpha} = 0)$. ***Remark:*** *The latter outcome physically means that any position on the plate is an equilibrium point as long as the velocity of the ball, the angle of the plate and the angular velocity of the plate are zero.* However, *in reality, the plate has boundaries which means that only the equilibrium states bounded by the plate's area are valid.*

Since there is an infinite number of equilibrium points, it is impossible for the equilibrium points to be globally attractive. And for the local attractivity, we know that a non-empty neighborhood of x_e for which the state converges to x_e does not exist. Therefore, non of the equilibrium points are locally attractive.

3.3.2 Stability analysis:

Reminder: *Based on the notion of stability and asymptotical stability (introduced in the reference book [1]), in order for an equilibrium state to be stable, any ball of radius $\epsilon > 0$ in which we want to confine the whole state trajectory we can find a ball of initial states around the equilibrium such that the trajectory will not go outside a ball of radius $\delta > 0$.*

Based on the stated definition of stability, non of the equilibrium states is stable.

Based on the BIBO stability definition, if a system gives a bounded output for any bounded input then it is BIBO stable. In our system, any bounded input angle α will cause the state x_b to go to infinity, Hence the system is not BIBO stable. This can be proven by looking at the transfer function obtained by the laplace transformation:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{X_b(s)}{\alpha(s)} = \frac{2507}{s^4 + 38.47s^3 + 369.6s^2 + 111.1s}$$

The roots of the corresponding characteristic polynomial are:

$$s_1 = 0$$
$$s_2 = -21.5800$$
$$s_3 = -16.5791$$
$$s_4 = -0.3106$$

Note that, s_1 is zero and therefore, not all the poles of the system have a negative real part. This means that the system is not BIBO stable.

After replacing all of the parameters in the state space with their real numbers and choosing the 4th ball configuration we get:

$$\begin{bmatrix} \dot{x}_b \\ \ddot{x}_b \\ \dot{\alpha} \\ \ddot{\alpha} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -0.3106 & 7.007 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -357.8 & -38.16 \end{bmatrix} \cdot \begin{bmatrix} x_b \\ \dot{x}_b \\ \alpha \\ \dot{\alpha} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 357.8 \end{bmatrix} \cdot \begin{bmatrix} \alpha_r \end{bmatrix} \tag{23}$$

As an LTI system, the stability depends only on the eigen vector of the matrix

A:

$$eig(A) = \begin{bmatrix} 0 \\ -0.3106 \\ -16.5791 \\ -21.5800 \end{bmatrix} \tag{24}$$

Now that we know the system is unstable, the question is *"can we stabilize the system ?"*. To answer this question, we should first check if the system is controllable or not.

3.3.3 Controllability analysis:

As previously defined, the controllability matrix corresponding to our system is:

$$R = \begin{bmatrix} B & AB & A^2B & A^3B \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 2506.99 \\ 0 & 0 & 2506.99 & -96443.34 \\ 0 & 357.77 & -13652.46 & 392961.5 \\ 357.77 & -13652.46 & 392961.5 & -10110524.55 \end{bmatrix}$$

This controllability matrix is full rank ($rank(R) = 4$), this means that the system is controllable and therefore, *theoretically*, any eigen value can be manipulated arbitrarily to reach the desired behavior.

3.3.4 Observability analysis:

Reminder: *The observability is the ability to fully estimate the states given a certain configuration of sensor measurement. Just like in controllability, if the observability matrix is full rank then the system is observable. Note that this definition is precisely for the critical states and not useless states in the state* *vector, therefore, this definition is only valid for the minimal realization of the state space.*

Our system has the following observability matrix:

$$O = \begin{bmatrix} C \\ AC \\ A^2C \\ A^3C \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -0.31 & 7 & 0 \\ 0 & 0.09 & -2.17 & 7 \end{bmatrix}$$

The system is observable since the matrix O is full rank ($rank(O) = 4$). This means that we can estimate all the states from the displacement measurement. Bear in mind that if we chose to measure exclusively the velocity of the ball, the angle of the plate or the angular velocity of the plate, the new C matrix will render the observability matrix O non full rank which means that the system would not be observable. (this proves that choosing only one sensor for the ball displacement is enough.)

3.3.5 Control Requirements:

Now that we know the system needs to be stabilized, the first task is to indicate the control requirements. The requirements will put the foundations and constraints needed to come up with a good controller. A reasonable requirement list for a step response is as follows:

- settling time : 4% $t_s \leq 3s$
- overshoot : $\leq 20\%$
- steady state error : $\leq 5mm$

3.3.6 Control Design:

To design a controller that fulfill the stated requirements, the most common ways do so is by:

- Pole placement (using the root locus)
- Loop shaping (using bode plot)
- Automatic tuning by simulink

As a first attempt, we will be using the robust feedback scheme with a PID controller. The block diagram is arranged as shown in fig.16.

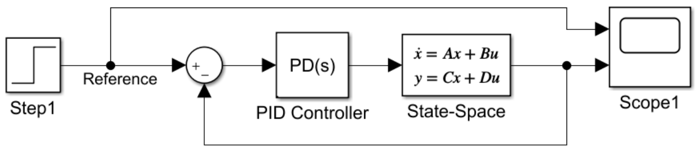

Figure 16: Simple closed feedback loop

With such requirements, root locus can be the simplest tool to design the controller.

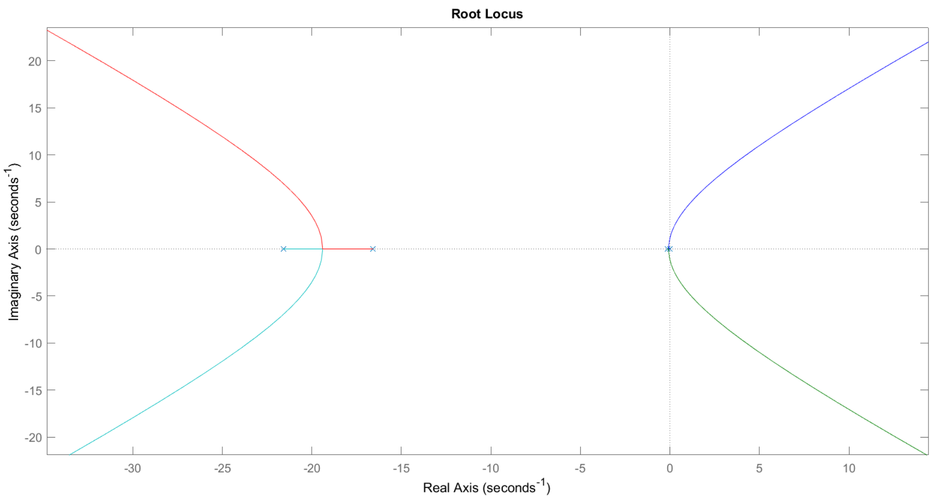

Figure 17: Corresponding root locus with no controller

From the root locus plot shown in fig.17, it is clear that the system is not BIBO stable and that there is 2 dominant poles that are very close from the real axis that needs to be damped. By closing the loop and applying a small proportional gain, the system is stabilized, however, it is far from fulfilling the control requirements and it has poor stability margins.

Since the plant is a 4th order system, the 2nd order formulas for placing the poles in a region where the requirements can be fulfilled doesn't work. However, the system can be treated as a second order as long as the 2 damped poles are reasonably far and don't contribute much to the dynamics of the system.

By using the 2nd order overshoot formula [2]:

$$M_p = e^{-\pi\zeta/\sqrt{1-\zeta^2}}$$

Where M_p is the overshoot and ζ is the damping ratio of the second order system. For an overshoot $M_p \leq 10\%$, $\zeta \geq 0.6$. A bounded ζ region can be constrained on the root locus where the angle of the dominant poles shouldn't exceed $\sin^{-1}\zeta$.

Another region can be ploted for the settling time where $t_s = \frac{4.6}{\zeta\omega_n} = \frac{4.6}{\sigma} \leq 3$, thus the real part of the dominant poles shouldn't exceed σ.

By applying only a PD controller as shown in fig.18, the pole cancellation method can cancel one of the dominant poles and make the new dominant poles fit in the constrained region. The step response of the controlled plant is shown in fig.19

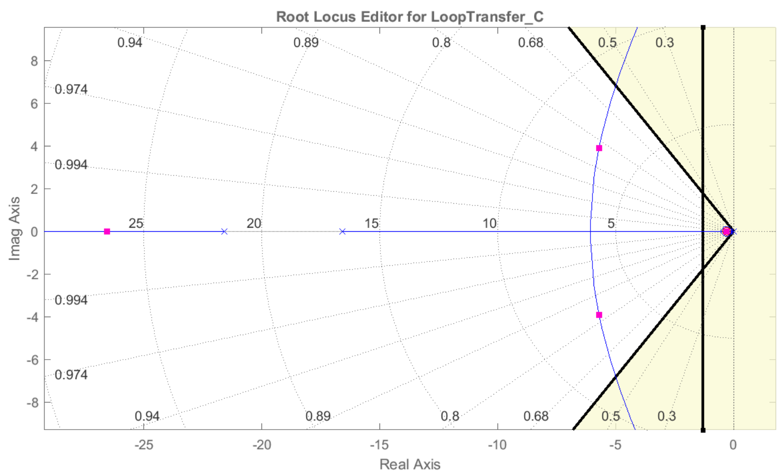

Figure 18: PD controller to fulfill the requirements

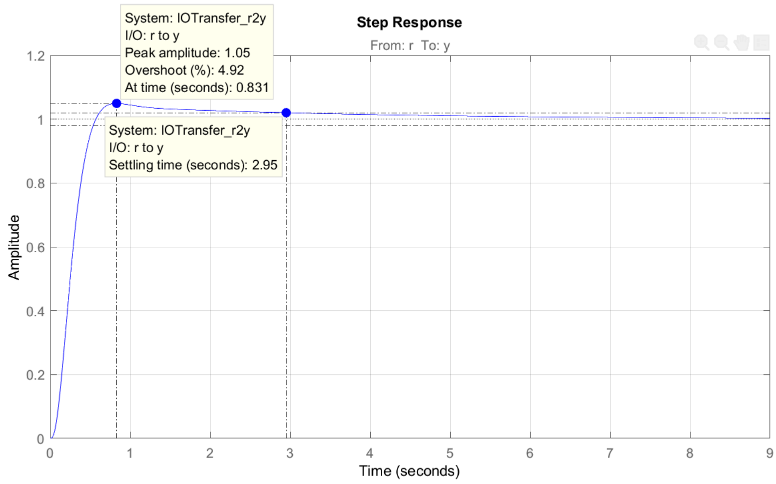

Figure 19: Step response of the plant by the aggressive PD controller.

Remark: *In practice, the locations of the lightly damped poles are not known precisely and exact cancellation is not really possible. However, placing the zero near the locations of the lightly damped poles may provide sufficient improvement in step response performance. Note that one must avoid unstable cancellations. For example, if the plant is unstable and therefore has a root in the RHP, we might cancel this pole by putting a zero at the same place. However, the unstable pole remains a pole of the system and this method will not work.[2]*

In reality, the latter controller won't work. One of the major reasons why is that the actuators are saturated which means that the plate can only be tilted to a limited angle of $\pm 15\,\mathrm{deg}$. Therefore, the model can be further developped as shown in fig.20.

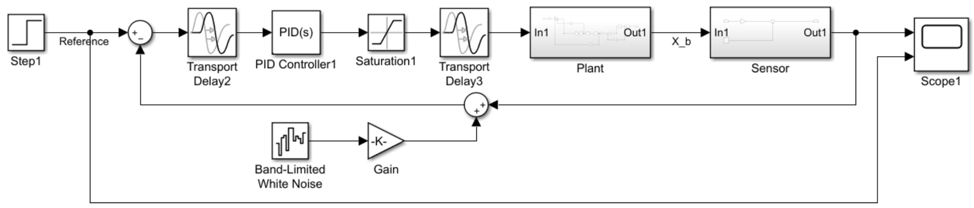

Figure 20: continuous time simulink model

Remark: *Notice the sensor block used after the plant, it is used to map the values of the displacement from the world frame to the sensor's reference frame which has pixels as coordinates. On the other hand, two delay blocks are used to imitate the image processing delay and the command transport delay from the computer to the arduino. A saturation block is used after the controller to saturate the actuator's angle. Furthermore, a sensor noise is also added to the model.*

In reality, the current model still doesn't hold true since the sampling time of the discrete controller can alter the behavior drastically. Therefore, another simulation model has been obtained by discretizing as shown in fig.21.

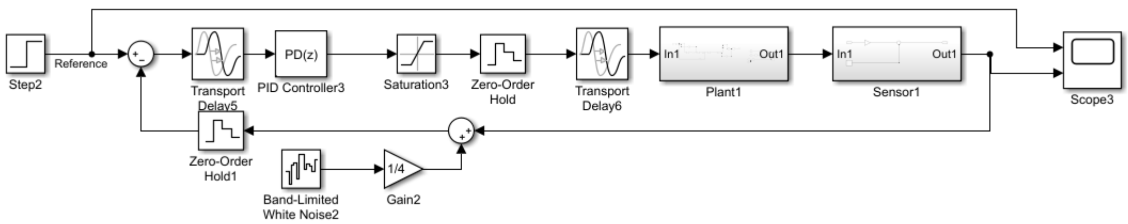

Figure 21: discrete time simulink model

Remark: *Notice the zero order hold blocks added to model the sampling time* *of the sensor (camera is capturing 30 FPS*[7]*, which gives a sampling time* $t_s = 0.033s$*). There is another ZOH*[8] *block to model the arduino's discrete output signal.*

Several approaches could be taken here like:

- Design a continuous controller and try it on the latest model.
- Design a discrete controller and try it on the latest model.
- Discretize the plant and design an appropriate discrete controller.

For the sake of simplicity, using the root locus again for pole placement we can get the following controller:

3.4 Simulation

4 Validation

In this chapter, the assumed dynamics are going to be validated on the real plant.

[7] Frames per second

[8] The zero-order hold is a mathematical model of the practical signal reconstruction done by a conventional digital-to-analog converter.

5 Results

6 Discussion and Conclusion

References

[1] Prof. Emanuele Garone, Assistant Professor Faculté des Sciences Appliquées/école Polytechnique Université Libre de Bruxelles, *Lecture Notes of Control System Design - Discrete Time System Theory*

[2] Gene F.Franklin, J.David Powell, Abbas Emami-Naeini - *Feedback Control of Dynamic Systems, Sixth Edition*

[3] Rafael da Silveira Castro, Group of Automation and Control Systems – PUCRS – Brazil, *BALL AND PLATE SYSTEM*

[4] Advanced Micro control INC., Stepper vs servo - *https://www.amci.com/industrial-automation-resources/plc-automation-tutorials/stepper-vs-servo/*

[5] Mohsin Mohammad Taufiq, Dr. Chan Ham, Kennesaw State University - 122nd ASEE annual conference and exposition - Paper ID #12313, *Development of a Ball and Plate System*

[6] Johan Link - Gymnase de Burier, *Système de stabilisation PID*

[7] Microchip - Atmel AVR341 document, *AVR341: Four and five-wire Touch screen Controller using tinyAVR and megaAVR devices*