Enhancing Control Engineering Education through Simulation-Based Lab Work Using Open-Source Multibody Dynamics Software

Young Suk KIM[1*] and Odbileg NOROVRYENCHIN[1]

[1]Faculty of Mechanical and Electrical Engineering, German-Mongolian Institute for Resources and Technology, Ulaanbaatar, Mongolia
yskim@gmit.edu.mn

Abstract. Control engineering is inherently rich in mathematics and thus through conventional instruction-based learning only, students often have a hard time to learn the concepts of control engineering. To address this issue, this study proposes the integration of simulation-based laboratory practices in control engineering education. Specifically, we demonstrate the use of an open-source multibody dynamics software to be used as support for control engineering courses. As a case study, the classic inverted pendulum control problem—widely recognized in control engineering curricula—is selected. The inverted pendulum is physically modeled within the open-source multibody dynamics software. A PID (Proportional-Integral-Derivative) controller is then implemented to stabilize the system and various studies are performed with the model, showing the usefulness of the method for control education. This method not only enhances student engagement but also fosters deeper comprehension of the concepts of control engineering. Integrating open-source multibody dynamics software into control engineering courses can significantly improve the learning experience and educational outcomes.

Keywords: Control Engineering Education, Simulation-Based Lab, Inverted Pendulum, Multibody Dynamics Software, PID controller

1 Introduction

Control engineering is concerned with how to adjust the input signal to a system in order to obtain the desired condition or output. The primary aim of control system design is stability, because a system cannot achieve its desired objectives if its output diverges. The stability of control systems can be assessed through the mathematical model of the system, and mathematical modeling and analysis are core parts of control engineering. Therefore, control engineering, by its nature, is rich in mathematics, and students often find it difficult to grasp the basic concepts through conventional instruction-based learning alone. As a result, they may become unmotivated. For this reason, many control engineering educators have advocated for making lab projects an integral part of control engineering courses, based on their teaching experiences [1], [2].

With the emergence of new simulation technologies, simulation-based labs have become feasible at the classroom level [3]. Several studies have investigated the

A. Lkhamsuren et al. (eds.), *Proceedings of the 3rd International Conference Resources and Technology (RESAT 2025)*, Advances in Engineering Research 284,
https://doi.org/10.2991/978-94-6463-928-5_25

relative effectiveness of simulation-based labs compared to physical labs in education. The results suggest that simulation-based labs are particularly suitable when students need to develop theoretical knowledge and conceptual understanding [4].

In this paper, multibody dynamics simulation tools are proposed for simulation-based lab practice in a control engineering course. As a case study, the design and analysis of an inverted pendulum control system is performed using an open-source multibody dynamics simulation tool (CoppeliaSim). The procedures are described in detail so that the case study can be adopted as an integral part of the control engineering curriculum.

2 Suggested Simulation-Based Lab Case

2.1 Inverted Pendulum System

An inverted pendulum system is a very popular lab experiment that is used to explain control theories. Figure 1 shows the inverted pendulum system, which was presented as a lab experiment in the lecture by Prof. Alan V. Oppenheim [5]. In the inverted pendulum system, a thin rod with a weight at the top is pivot connected to a cart at the bottom. The cart can move in either direction along a track. The system is unstable and without additional control the pendulum will fall over. Prof. Oppenheim demonstrated that the pendulum can be suspended stably in this inverted position by using a PID control system. In the control process, the angle of the pendulum was monitored and the cart was moved horizontally, keeping it balanced. At the later part of the demonstration, a cup filled with liquid was placed on top of the pendulum as shown in Figure 2 and the system was able to maintain the pendulum stably in the inverted position. In this paper, the inverted pendulum system by prof. Oppenheim is selected as a simulation-based lab case.

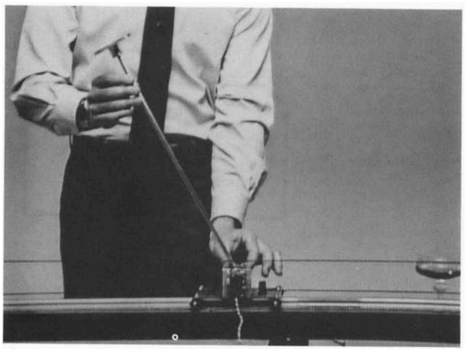


Fig. 1. Inverted Pendulum System by Prof. Alan V. Oppenheim (MIT OpenCourseWare).

Fig. 2. Inverted Pendulum Demonstration with a Cup on Top of Pendulum(MIT OpenCourseWare).

2.2 Problem Setup

Because the physical quantities for inverted pendulum system were not given at the prof. Oppenheim's demonstration, the following data from the reference [6] were assumed for the current system as shown in Table 1 and Figure 3. When the case is given to the students as a lab project, the data can be modified and different values can be assigned to each student.

Table 1. Physical quantities for inverted pendulum.

Variable	Name	Value
(M)	mass of the cart	0.5 kg
(m)	mass of the pendulum	0.2 kg
(b)	coefficient of friction for cart	0.1 N·s/m
(l)	length to pendulum center of mass	0.3 m
(I)	mass moment of inertia of the pendulum	0.006 kg·m^2
(F)	disturbance force applied to the cart	1 N·s
(x)	cart position	
(ϕ)	pendulum angle from vertical (up)	

3 Modelling and Analysis of System without Control

3.1 Mathematical Modelling

A linearized mathematical model for the inverted pendulum system of Figure 3 is derived by the procedures in [6] and is shown in Eq. (1). The transfer function of the pendulum angle to the force applied to the cart is obtained by Laplace transforming Eq. (1) as in Eq. (2).

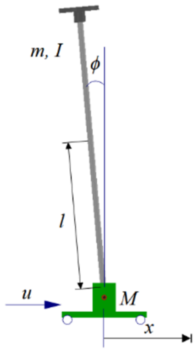


Fig. 3. Schematic Diagram of Inverted Pendulum.

$$\left(I + ml^2\right)\ddot{\phi} - mgl\phi = ml\ddot{x}\ (M + m)\ddot{x} + b\dot{x} - ml\ddot{\phi} = u \tag{1}$$

$$G(s) = \frac{\Phi(s)}{U(s)} = \frac{\frac{ml}{q}s}{s^3 + \frac{b\left(I+ml^2\right)}{q}s^2 - \frac{(M+m)mgl}{q}s - \frac{bmgl}{q}},\ where\ q = \left[(M + m)\left(I + ml^2\right) - (ml)^2\right] \tag{2}$$

3.2 Modelling Using Multibody Dynamics Simulation Tool

The inverted pendulum system is also modelled using CoppeliaSim (with Bullet 2.78 Physics Engine) as shown in Figure 4. A pendulum and a cart are modelled as solid bodies applying the size and mass quantities given in Table 1. Mass quantities are applied directly using user input method and the center of mass location of pendulum is artificially kept in the half-length position of the rod. A revolute joint is modelled between the pendulum and the cart and a translational joint is modelled between the cart and the ground that is equivalent to a fixed track. An impulse force of 1 N·s is applied to the cart approximated as a force of 200 N for 0.005 s duration.

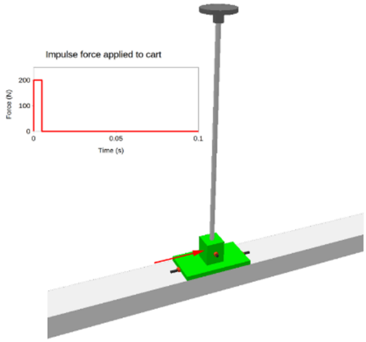


Fig. 4. CoppeliaSim Model of Inverted Pendulum.

3.3 Stability Analysis of Inverted Pendulum without Control

Eq. (3) is the transfer function for the system with actual numbers obtained by putting physical quantities of Table 1 into Eq. (2).

$$G(s) = \frac{4.545s}{s^3 + 0.1818s^2 - 31.18s - 4.455} \tag{3}$$

Absolute stability of the system can be judged from the pole locations of the transfer function, which are the roots of the denominator of Eq. (3). As can be seen from Figure 5, one pole of the system is in the right half plane of the graph and it can be judged that the system is unstable.

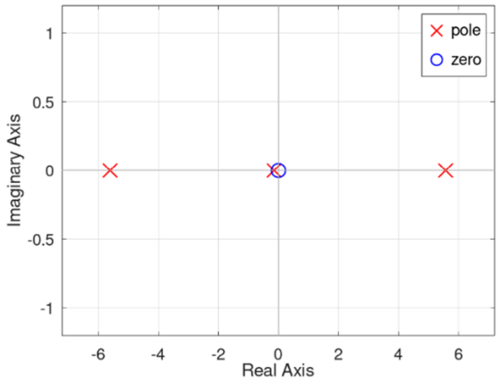


Fig. 5. Pole-zero plot of the system without control.

To confirm the stability analysis result above, the response of the system to the impulse force is studied using the mathematical model and the CoppeliaSim model. Response of mathematical model is calculated from the transfer function using GNU

Octave command 'impulse' as the procedures shown in [6]. The multibody dynamics model response is obtained from CoppeliaSim by measuring the pendulum angle changes from simulation. As can be seen from the result in Figure 6, the system is unstable and the pendulum angle increases unlimitedly. The difference between two results at later time is due to the linearization of the mathematical model on the assumption that the pendulum angle is small. Therefore, the result from the mathematical model matches well with the result from the CoppeliaSim model when the pendulum angle is less than 30°. Figure 7 is the snapshots of CoppeliaSim simulation.

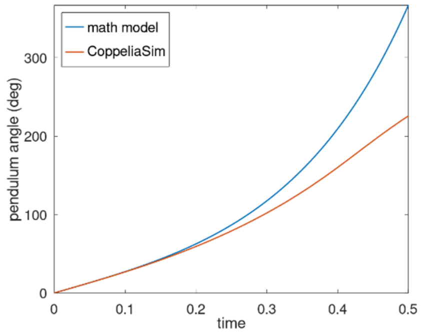


Fig. 6. Pendulum angle response for both models.

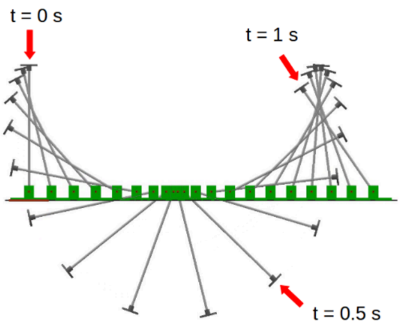


Fig. 7. Snapshots of CoppeliaSim simulation.

4 Modelling and Analysis of System with Feedback Control

4.1 Modelling of Control System into Mathematical Model

The aim of the control design is to keep the inverted pendulum in vertical position by adjusting an actuator force to the cart. The schematic of the whole feedback control system is shown in Figure 8. The resulting transfer function for pendulum angle to disturbance force is shown in Eq. (4).

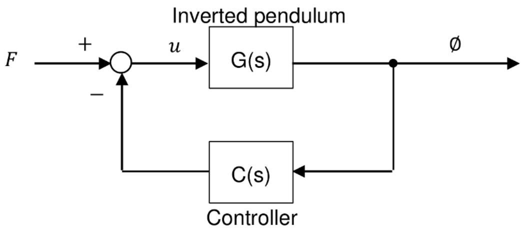


Fig. 8. Schematic of feedback control system.

$$\frac{\Phi(s)}{F(s)} = \frac{G(s)}{1+C(s)G(s)} \tag{4}$$

In this study, PID (Proportional plus Integral plus Differential) controller, which has general applicability to most control systems, is applied to the controller design. A mathematical model of the system with PID controller can be obtained by adding a few GNU Octave command lines to the model for the system without controller as shown in [6]. In the PID control, a set of gain values of Kp, Ki and Kd are adjusted for a stable operation of the system. Figure 9 is the response of the system with gain values of Kp=100, Ki=1 and Kd=1 to impulse disturbance force to the cart.

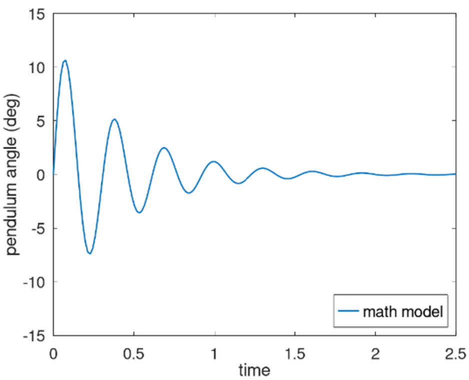


Fig. 9. Response of system with gain values of Kp=100, Ki=1, Kd=1.

4.2 Modelling of Control System into CoppeliaSim Model

CoppeliaSim uses the Lua programming language as its built-in scripting environment to control the behavior of simulated robots, sensors, and other multibody dynamic objects. A PID control system can be added to the multibody physics model of the inverted pendulum model through this Lua script as conceptually demonstrated in Figure 10. In this case, the reference input is the pendulum angle of 0°, and the disturbance is the impulse shown in Figure 4. Table 2 summarizes the main Lua script for the added control system.

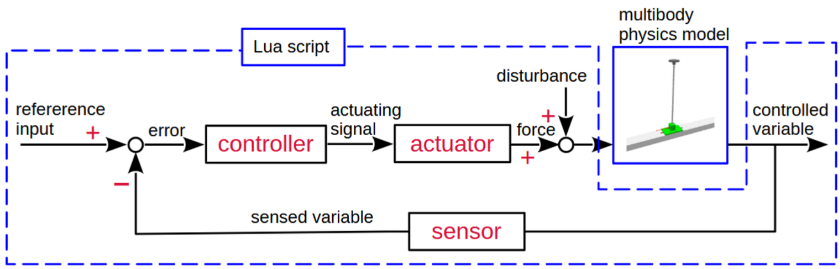
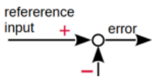


Fig. 10. Conceptual demonstration of control system added to the physics model though Lua script.

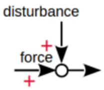
Table 2. Lua script to add a PID control system.

initialization	`-- get handles for joint and cart`
	`jointR=sim.getObject('/Revolute_joint')`
	`cart=sim.getObject('/cart')`
	`errorSum=0`
	`errorOld=0`

sensor	`-- measure current pendulum angle` `jPos=sim.getJointPosition(jointR)`
	`refAngle=0` `error=(refAngle-jPos)`

controller	`-- PID controller`
	`dt=sim.getSimulationTimeStep()`
	`Kp=100`
	`Ki=1`
	`Kd=1`
	`errorSum=errorSum+error*dt`
	`errorRate=(error-errorOld)/dt`
	`errorOld=error`
	`u=Kp*error+Ki*errorSum+Kd*errorRate`

actuator	`-- force` `sim.addForce(cart,{0,0,0},{0,u,0})`

	`-- disturbance`
	`t=sim.getSimulationTime()`
	`if t<0.005 then`
	`   sim.addForce(cart,{0,0,0},{0,200,0})`
	`end`

The CoppeliaSim model with the PID control is run using the gain values of Kp=100, Ki=1 and Kd=1, and the result is compared to the result of mathematical model, as shown in Figure 11. Two results match well, and the CoppeliaSim model for inverted pendulum system is now ready for students to perform various virtual experiments to gain deeper insight and conceptual understanding of control engineering.

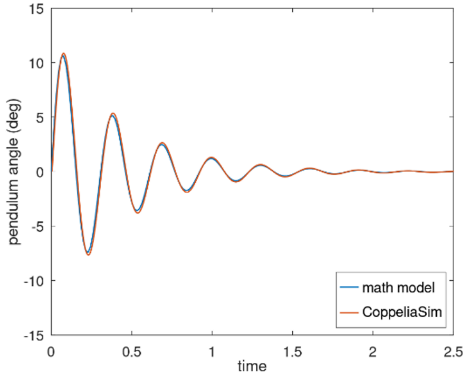


Fig. 11. Response of the system with gain values Kp=100, Ki=1, Kd=1 (math model. vs. CoppeliaSim model).

4.3 Stability Analysis of Inverted Pendulum with Feedback Control

To help students to understand the relation between a stability and pole locations of the system, cases with various gain values of Kp, Ki and Kd can be run with the CoppeliaSim model and the results can be compared with the pole locations of the cases (Fig. 12).

Case 1: Kp=100, Ki=0, Kd=0

Closed loop transfer function

$$T(s) = \frac{4.545s}{s^3+0.1818s^2+423.4s-4.455}$$

pole: $s_1 = -0.0962 + j20.6, s_2 = -0.0962 - j20.6, s_3 = 0.0105$

zero: $z_1 = 0.0$

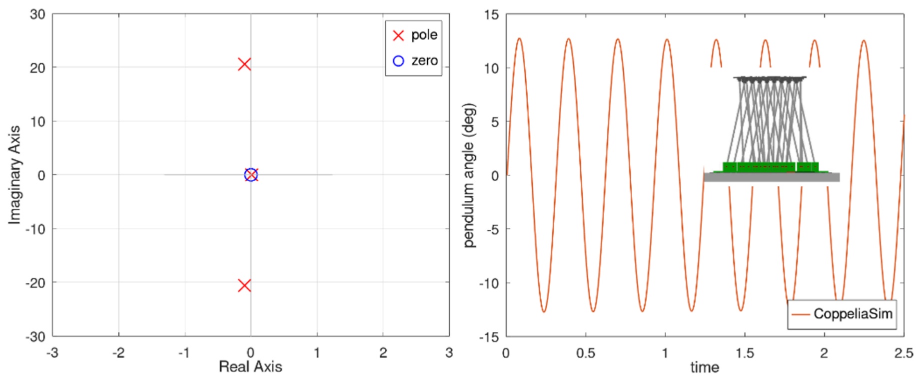


Fig. 12. Pole-zero plot (left) and response of the system (right) with gain values Kp=100, Ki=0 and Kd=0.

Analysis: Two complex conjugate poles are in the left-half plane and one real pole is in the right-half plane near the imaginary axis. The system is stable, even though the pole is located in the right-half plane, because its effect is canceled by a zero located at the origin. However, the response oscillates with slow convergence because the complex conjugate poles in the left-half plane are close to the imaginary axis.

Case 2: Kp=100, Ki=1, Kd=1

Closed loop transfer function

$$T(s) = \frac{4.545s}{s^3+4.727s^2+423.4s+0.09091}$$

pole: $s_1 =- 2.36 + j20.4, s_2 =- 2.36 - j20.4, s_3 =- 0.00021$

zero: $z_1 = 0.0$

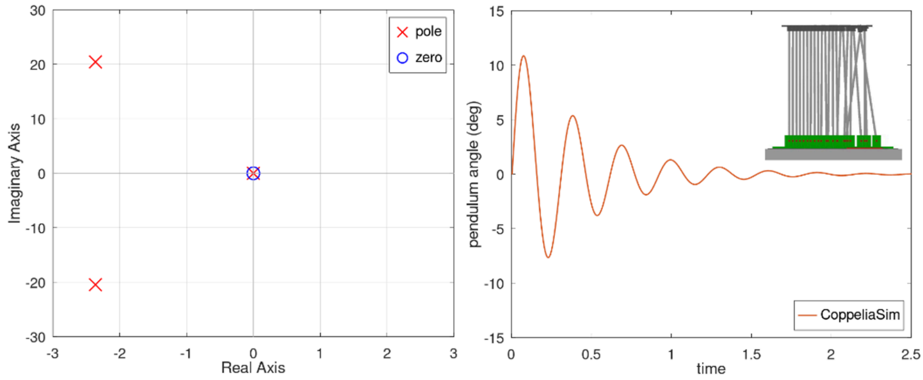


Fig. 13. Pole-zero plot (left) and response of the system (right) with gain values Kp=100, Ki=1 and Kd=1.

Analysis: All three poles of the system are in the left-half plane, and one pole near the origin is canceled by a zero at the origin. The two complex conjugate poles are far from the imaginary axis and thus the response converges fast with oscillation. The position of the cart is not a controlled variable, therefore the cart moves to the left.

4.4 PID Controller Design using Root Locus

At this stage, students have their own inverted pendulum models with a PID controller and attempt to find the optimum Kp, Ki and Kd gain values by trial-and-error. Through the practice, they learn the effect of increasing the PID gains, such as increasing Kp will reduce the rise time while increasing the overshoot, and increasing Kd will decrease the overshoot but increase the rise time. This is the right time to introduce the root locus method for PID controller design, so that optimum gain values can be determined using the root locus. The root locus is the trace of poles in the s-plane as a gain K varies from zero to infinity for a closed-loop system such as the one shown in Figure 13.

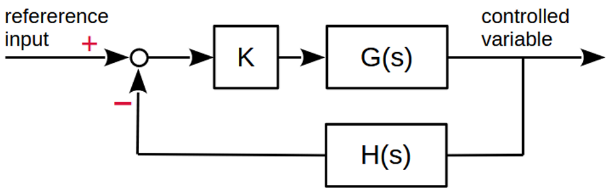


Fig. 14. Closed-loop control system with a gain K.

To draw a root locus, our system with PID controller can be represented by the block diagram shown in Figure 14, and it can be seen that an initial choice of the PID gain values Kp, Ki and Kd is needed.

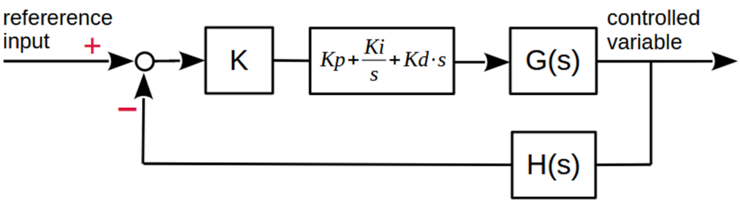


Fig. 15. Closed-loop PID control system with a gain K and PID controller.

The educated guess of them can be achieved using the Ziegler-Nichols PID tuning method shown in Table 3, where K_U is the ultimate value of the gain Kp at which output shows sustained oscillation, and T_U is the period of the sustained oscillation.

Table 3. Ziegler-Nichols PID Tuning [7].

Controller Type	K_P	K_I	K_D
$G_c(s) = K_P + \frac{K_I}{s} + K_D s$	$0.6K_U$	$\frac{1.2K_U}{T_U}$	$\frac{0.6K_U T_U}{8}$

Because the Case 1, shown in Figure 11, is in near sustained oscillation, the values of $K_U = 100$ and $T_U = 0.3\ s$ can be obtained from the result of Case 1. By applying the obtained values to Table 3, the educated guess of the PID gain values (Kp=60, Ki=400 and Kd=2.25) is decided. Figure 15 shows the response of the system when these values are applied to the inverted pendulum model with the PID controller. The response is stable with slow convergence because the dominant poles are located near the imaginary axis.

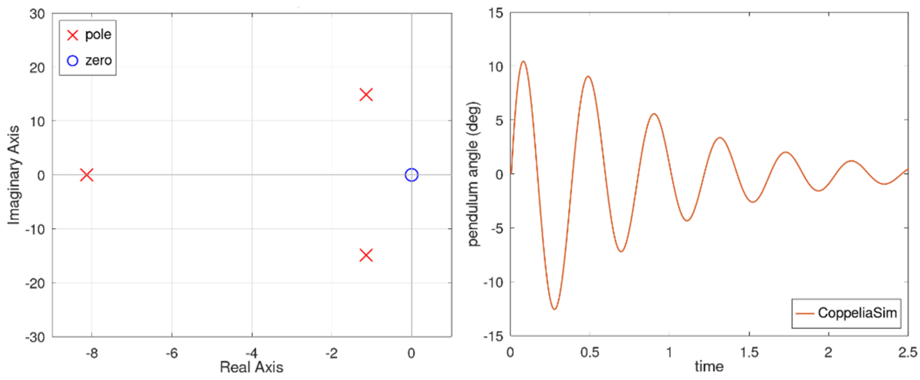


Fig. 16. Pole-zero plot (left) and response of the system (right) with gain values Kp=60, Ki=400 and Kd=2.25.

Now, by applying these values to the PID gain block of Figure 14, the root locus can be plotted using gnu Octave, where the K value varies from 0 to infinity, as shown in Figure 16. If the design requirements of maximum percent overshoot $M_p < 10\%$ and settling time $t_s < 1\ s$ is specified, the region satisfying these requirements is decided and marked in yellow using the method described in reference [8].

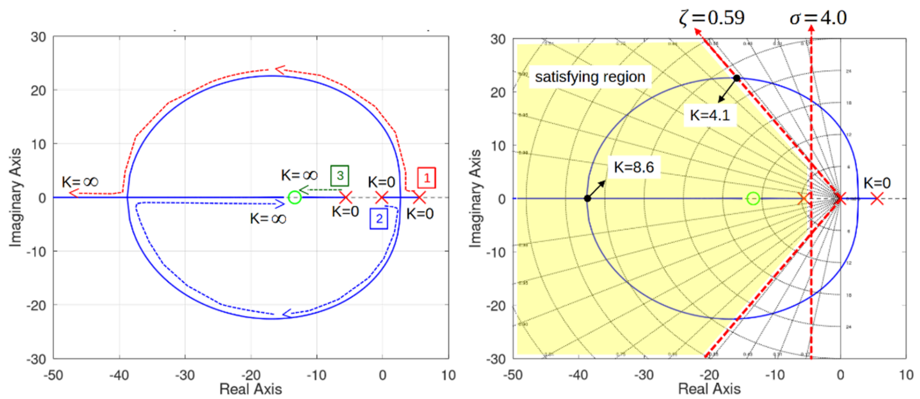


Fig. 17. Root locus of the system, pole traces (left), region satisfying design requirements (right).

Considering that a high gain value can increase the cost of an actuator, K=5 is selected as an optimum choice. The new PID gains, which will be applied to the inverted pendulum model, can be obtained by multiplying this K value with the PID gains previously determined using the Ziegler-Nichols tuning method: Kp=5×60=300, Ki=5×400=2000, Kd=5×2.25=11. Figure 17 shows the pole locations and the response of the system, which meets the design requirements of $M_p < 10\%$ and $t_s < 1\,s$.

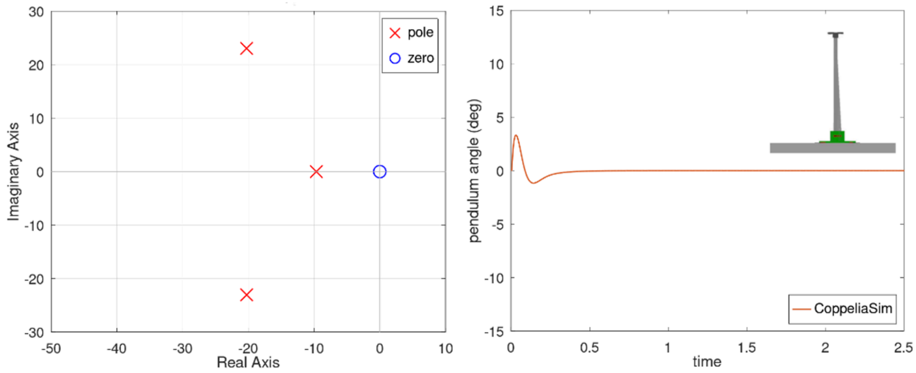


Fig. 18. Pole-zero plot (left) and response of the system (right) with gain values Kp=300, Ki=2000 and Kd=11.

5 Simulation of the System with a Cup on Top of the Pendulum

In the later part of Prof. Oppenheim's lab experiment [5], a cup was placed on top of the pendulum and the robustness of the designed control system was demonstrated. To give students more realistic experience to the lab experiment, the cup on top of the pendulum can be included in the model for the student's lab project, and stable holding of the cup can be added to the design requirements.

A cup of 0.35 kg is added to the CoppeliaSim model, as shown in Figure. 18, and a solid-to-solid contact is modeled between the cup and the pendulum. The two cases in Section 4.3, along with the optimum case determined using the root locus method, are simulated, and the results are shown in Figures 18, 19, 20. The optimum case with gain values of Kp=600, Ki=2000 and Kd=11 stably holds the cup in spite of the disturbance impulse force.

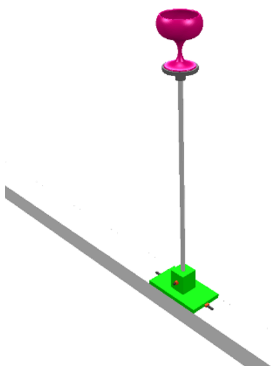


Fig. 19. Inverted pendulum system with a cup on top of pendulum.

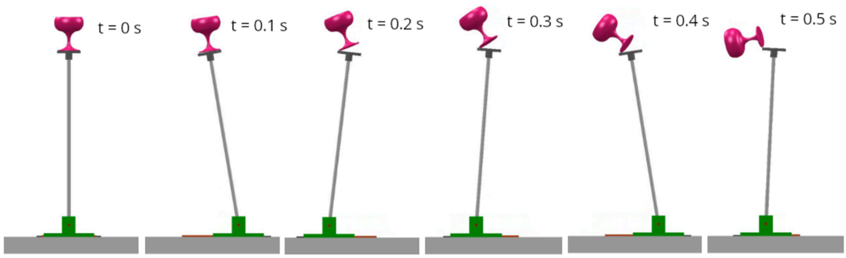


Fig. 20. Case 1: Kp=100, Ki=0 and Kd=0.

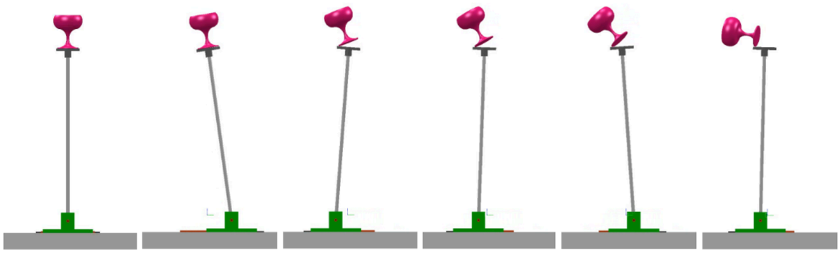


Fig. 21. Case 2: Kp=100, Ki=1 and Kd=1.

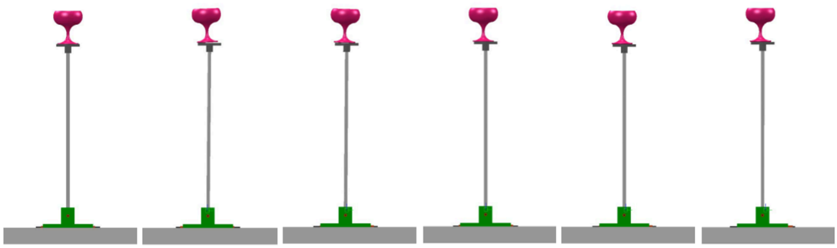


Fig. 22. Optimum Case: Kp=600, Ki=2000 and Kd=11.

6 Conclusions

The author's own experience in taking a control engineering course was full of equations and plots, and it was hard to understand them in relation to physical systems. In this paper, simulation-based lab experiments using open-source multibody dynamics software are proposed as an integral part of a control engineering course. As a lab case, the design and analysis of an inverted pendulum control system is carried out using a multibody dynamics simulation tool. The procedures are summarized in detail, so that this case study can be effectively used as an integral part of the control engineering course.

7 References

1. Antsaklis, P., Başar, T., DeCarlo, R., McClamroch, N.H., Spong, M., Yurkovich, S.: Report on the NSF/CSS Workshop on New Directions in Control Engineering Education. IEEE Control Systems 19, 53–58 (1999)
2. Nickerson, J.V., Corter, J.E., Esche, S.K., Chassapis, C.: A model for evaluating the effectiveness of remote engineering laboratories and simulations in education. Computers & Education 49, 708–725 (2007)
3. Meza, C., Andrade-Romero, J.A., Bucher, R., Balemi, S.: Free open source software in control engineering education: A case study in the analysis and control design of a rotary inverted pendulum. In: 2009 IEEE Conference on Emerging Technologies & Factory Automation, pp. 1–6. IEEE, Mallorca, Spain (2009)
4. Goodwin, G.C., Medioli, A.M., Sher, W., Vlacic, L.B., Welsh, J.S.: Emulation-Based Virtual Laboratories: A Low-Cost Alternative to Physical Experiments in Control Engineering Education. IEEE Transactions on Education 54, 48–55 (2011)
5. Oppenheim, A.V.: Signals and Systems. MIT OpenCourseWare, https://ocw.mit.edu/resources/res-6-007-signals-and-systems-spring-2011/, last accessed 2025/06/11
6. University of Michigan: Control Tutorials for MATLAB and Simulink. http://ctms.engin.umich.edu/CTMS/index.php?example=InvertedPendulum&section=SystemAnalysis, last accessed 2025/06/11
7. Dorf, R.C., Bishop, R.H.: Modern Control Systems. 13th edn. Pearson, Boston (2017)
8. Ogata, K.: Modern Control Engineering. 5th edn. Prentice Hall, Upper Saddle River (2010)