


RESEARCH ARTICLE

Design and stability analysis of semi-implicit cascaded proportional-derivative controller for underactuated cart-pole inverted pendulum system

Changyi Lei¹ , Ruobing Li² and Quanmin Zhu²

¹Department of Informatics, King's College London, London, UK and ²Department of Engineering, Design and Mathematics, University of the West of England, Bristol, UK

Corresponding author: Quanmin Zhu; Email: quan.zhu@uwe.ac.uk

Received: 18 January 2023; **Revised:** 3 August 2023; **Accepted:** 12 September 2023; **First published online:** 16 October 2023

Keywords: semi-implicit controller; underactuated system; cascaded PD controller; nonlinear control; cartpole system

Abstract

This article proposes a control method for underactuated cartpole systems using semi-implicit cascaded proportional-derivative (PD) controller. The proposed controller is composed of two conventional PD controllers, which stabilizes the pole and the cart second-order dynamics respectively. The first PD controller is realized by transforming the pole dynamics into a virtual PD controller, with the coupling term exploited as the internal tracking target for the cart dynamics. Then, the second PD controller manipulates the cart dynamics to track that internal target. The solution to the internal tracking target relies on an equation set and features a semi-implicit process, which exploits the internal dynamics of the system. Besides, the design of second PD controller relies on the parameters of the first PD controller in a cascaded manner. A stability analysis approach based on Jacobian matrix is proposed and implemented for this fourth-order system. The proposed method is simple in design and intuitive to comprehend. The simulation results illustrate the superiority of proposed method compared with conventional double-loop PD controller in terms of convergence, with the theoretical conclusion of at least locally asymptotic stability.

1. Introduction

Underactuated systems feature a class of systems whose state Degree-of-Freedom (DoF) is greater than its number of control inputs. This kind of systems are easily witnessed in a wide range in practice, for example, wheeled robots [1], underwater vehicles [2], flexible robot systems [3], etc. One critical advantage of underactuated systems over fully and over-actuated systems is that they require less cost and have less complexity due to lack of control inputs. Nonetheless, due to the exact same reasons, the control problem of underactuated systems has been a heated research direction. Among all underactuated systems, cartpole system has been a classic benchmark model that absorbs uncertainty, coupling, nonlinearity, non-minimum phase, multivariable and instability, which encompasses a majority of other underactuated systems. Therefore, the research into cartpole system has a fundamental significance to gain insight into other system dynamics [4].

Over the decades, various methods have been constantly proposed for the stabilization of cartpole system. Some early work tried to linearize the nonlinear model of cartpole near the equilibrium location and then implemented linear controllers. This simplification usually ensures stability near the equilibrium. Well-known examples include Proportional-Integral-Derivative (PID) controller and Linear-Quadratic-Regulator [5–7]. Nonetheless, the linearization procedure impairs the accuracy of the dynamics and therefore cannot achieve large-scale stability [8]. Backstepping has been one of the most

widely researched method for cartpole system control problem. Shao et al. adopted a state-feedback-based backstepping controller for the tracking and switching control of cartpole systems [9]. Targeting at underactuated systems, Jiang et al. proposed an underactuated backstepping method for a class of underactuated systems [10]. Compared with conventional backstepping, this method has a systematic solution to a class of systems. However, the tuning and selection of those control matrices still remain an open question. Adaptive and robust control methods also received public attention. An adaptive optimal fuzzy controller based on feedback linearization and sliding mode control was proposed in ref. [11] for cartpole systems. Fuzzy logic system and gradient descent were combined to tune the parameters and a multi-object optimization algorithm was used to adjust sliding mode control gain. In ref. [12], an adaptive output-feedback optimal control was combined with integral sliding mode control for wheeled inverted pendulum under disturbance. The integral sliding mode controller was responsible for finite-time convergence, and adaptive dynamic programming was to deal with coupled uncertainties. In ref. [13], an adaptive control scheme is proposed based on Adaptive Ellipsoid Method (AEM) to tune the gain matrices of the observer and controller. The experiments on an underactuated vertical double pendulum with uncertainties illustrate the superiority over conventional AEM controller. There are many other control methods implemented case-by-case to cartpole systems, including energy-based [14], state-feedback controller [15], neuro network [16], and so on. Besides, more other methods were targeting at other underactuated systems which could easily be extended to cartpole system, for example, event-triggered dynamic surface control [17], fast terminal sliding mode control [18], etc. Nonetheless, most of the above-mentioned methods are tedious in design, which hinders their application in industry.

One big class of methods that are emergent recently is learning-based method. The common feature of this class of approaches lies in using huge amount of data to train a specific controller and optimizes a designed objective function. A milestone research in this direction was conducted by Google DeepMind, which used deep Q-learning to complete the control task of cartpole [19]. A forward neural network was implemented to approximate the Q-values of state-action pairs. Shi et al. combined type-1 Fuzzy Logic System (FLS) with Reinforcement Learning to achieve robust cartpole control [20]. The FLS was implemented as an encoder to cope with the uncertainty of the system, and RL was to find optimal policy that minimizes the tracking error. Hiremath et al. applied a deep neural network-based gated-recurrent-units (GRUs) method for the stabilization and tracking problem of constrained stochastic cartpole system [21]. Nevertheless, this class of methods usually require too much data to train the model. Besides, the black-box models trained make the inner dynamics intractable.

Therefore, borrowing ideas from conventional PID controller, this paper proposes a control method for underactuated cartpole system. The advantages of conventional PID controller are simple to design and intuitive to understand. Borrowing ideas from backstepping method, the work here extends the conventional PID controller to a cascaded version, which helps establish internal control targets. This manipulation increases the order of controller. In this way, the merits of PID controller can be maintained, while it can be implemented directly without linearization procedures. Contrary to widely accepted backstepping approach, the proposed method does not suffer from exploding terms or complex coordinate transformation technique. What is more, while most of the analysis of previous PID research lies in using linearization and transfer function, this paper proposes to implement Jacobian matrix-based stability analysis, which is applicable to any differentiable nonlinear system dynamics. The contributions of this article are summarized as follows:

- Propose a unique cascaded PD controller, which transforms the pole dynamics into a virtual PD controller and using the coupling term as the design variable for the second PD controller design. This model-based method absorbs the simplicity and intuitiveness of a conventional PID controller, but exploits the system dynamics in the meantime.
- Introduce a stability analysis method for the fourth-order cascaded PD controller using the Jacobian matrix of the residual system, although it concludes only locally asymptotic stability.

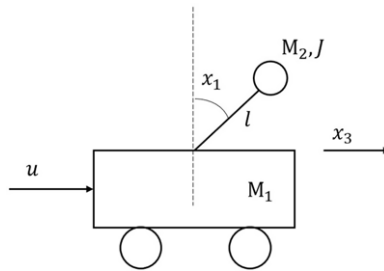


Figure 1. Conceptual structure of cartpole system.

This presents a novel way to approach stability analysis in this context, with the potential to be used for parameter design.

- Achieve automation of complex derivations and design processes through symbolic calculations on a PC, allowing for more efficient design and validation.
- This paper includes a comprehensive analysis on both the linearized and original nonlinear models, providing a thorough examination of the proposed method's applicability.
- Simulation results reveal the proposed method's advantages in stabilizing both the cart and the pole simultaneously, showing superior performance over widely used double-loop PD controllers. The robustness against Coulomb friction and random noise is also demonstrated.

The rest of the paper is organized as follows. In Section 2, some background rationale is introduced. Firstly, the dynamic model of the cartpole system is given, both linear and nonlinear. Secondly, conventional PD controller is presented, which serves as the basis of the proposed method. Section 3 articulates the design process of the proposed method. The overall framework and workflow are foremost described. Then, the design process for linear and nonlinear dynamics is presented, followed with stability analysis procedures. Section 4 illustrates the results in simulation. The system responses are depicted, and further analysis is carried out using Jacobian matrix. Section 5 concludes the article and points out many potent further research directions.

2. Preliminaries

2.1. Dynamic model description

This section introduces the structure as well as dynamics of the cartpole system to be investigated later in this research. Both nonlinear and linear version of the dynamics will be presented, and the controller design is to be carried out on both. The inclusion of linear version model is to present the method more clearly, as the nonlinear model of cartpole system is so complicated that the readers may be distracted from the mathematics instead of the workflow of the proposed control method.

Figure 1 illustrates the conceptual structure of the cartpole system. The system, as the name suggests, is composed of a cart, to which a pole is connected on top of it. The goal of control in this regard is to keep the pole upstraight for as long as possible. In the meantime, it is reasonably required to reduce the movement of the cart during the process. The (nonlinear) dynamics of the system can be expressed as follows [22]. In Eq. (1) and Fig. 1, u is the control force (N), x_1, x_2 are the angular position and angular velocity of the pole (rad, rad/s), x_3, x_4 are the position and linear velocity of the cart (m, m/s). l is the length of the pole (m), M_1 is the mass of the cart (kg), J is the moment of inertia (kg m^2), and M_2 is the mass of the pole (kg). Besides, F_0, F_1 are the friction factor of the cart and the pole respectively (N/m/s). g is the gravity coefficient.

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{\left(-F_1(M_1 + M_2)x_2 - M_2^2 l^2 x_2^2 \sin(x_1) \cos(x_1) + F_0 M_2 l x_4 \cos(x_1) \right. \\ \qquad \qquad \qquad \left. + (M_1 + M_2) M_2 g l \sin(x_1) - M_2 l \cos(x_1) u \right)}{p_2} \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = \frac{\left(F_1 M_2 l x_2 \cos(x_1) + (J + M_2 l^2) M_2 l x_2^2 \sin x_1 - F_0 (J + M_2 l^2) x_4 \right. \\ \qquad \qquad \qquad \left. - M_2^2 g l^2 \sin(x_1) \cos(x_1) + (J + M_2 l^2) u \right)}{p_2} \end{cases} \quad (1)$$

To simplify the controller design, the model is frequently linearized near the equilibrium location, namely when $x_1 \approx 0, x_3 \approx 0$. Therefore, the following approximations hold:

$$\sin(x_1) \approx x_1 \quad (2)$$

$$\cos(x_1) \approx 1 \quad (3)$$

$$x_2^2 \approx x_4^2 \approx 0 \quad (4)$$

Integrating with Eq. (1), a linearized model of cartpole is derived:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = Ax_1 + Bx_2 + Cx_4 + Du \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = Ex_1 + Fx_2 + Gx_4 + Hu \end{cases} \quad (5)$$

where

$$A = \frac{(M_1 + M_2)M_2gl}{J(M_1 + M_2) + M_1M_2l^2} \quad (6)$$

$$B = \frac{-F_1(M_1 + M_2)}{J(M_1 + M_2) + M_1M_2l^2} \quad (7)$$

$$C = \frac{F_0M_2l}{J(M_1 + M_2) + M_1M_2l^2} \quad (8)$$

$$D = -\frac{M_2l}{J(M_1 + M_2) + M_1M_2l^2} \quad (9)$$

$$E = \frac{-M_2^2gl^2}{J(M_1 + M_2) + M_1M_2l^2} \quad (10)$$

$$F = \frac{F_1M_2l}{J(M_1 + M_2) + M_1M_2l^2} \quad (11)$$

$$G = \frac{-F_0(J + M_2l^2)}{J(M_1 + M_2) + M_1M_2l^2} \quad (12)$$

$$H = \frac{J + M_2l^2}{J(M_1 + M_2) + M_1M_2l^2} \quad (13)$$

It is conceivable that the cartpole system is a highly nonlinear system. High order of trigonometric functions appear both in the denominators and the numerators. Besides, the coupling effect is remarkable. x_1, x_2, x_4 and u pose influence on the dynamics of both the cart and the pole. Therefore, the control problem of cartpole system is challenging and has some fundamental influence in control realm.

2.2. Conventional PD controller

Figure 2 is the conceptual structure of a conventional PD controller. R is the reference signal, $e(t)$ is the error in time t , U is the control signal and Y is the output. The core modules of PD controller are

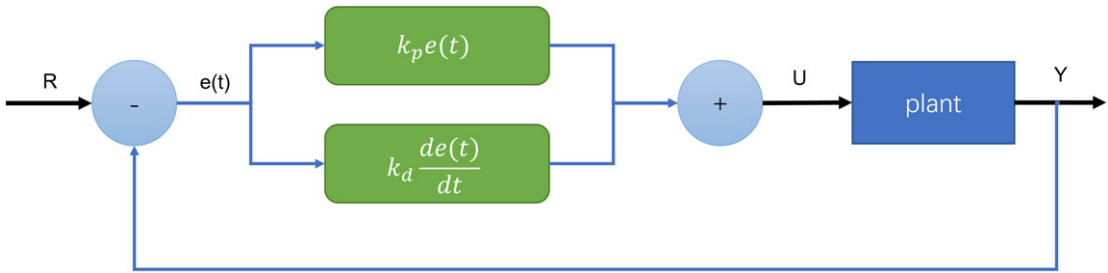


Figure 2. Conceptual structure of conventional PD controller.

proportional and derivative module, and the mathematical expression is [23]

$$U = K_p e(t) + k_d \frac{de(t)}{dt}, \tag{14}$$

where k_p, k_d are the proportional gain and derivative gain. PD controller is a simplified version of PID controller, which is widely adopted in the industry [24].

3. Controller design and analysis

3.1. Controllability analysis around equilibrium point

In this section, the controllability of the system near the unstable equilibrium point is analysed, which serves as the foundation for controller design. Consider system dynamics (5) and write down the system matrices as follows:

$$Y = \begin{bmatrix} 0 & 1 & 0 & 0 \\ A & B & 0 & C \\ 0 & 0 & 0 & 1 \\ E & F & 0 & G \end{bmatrix} \tag{15}$$

$$Z = [0, D, 0, H]^T \tag{16}$$

According to controllability theorem, if matrix $[Z, YZ, Y^2Z, Y^3Z]$ has full rank, then the system is controllable in the equilibrium point. Integrating system parameters in Table II, the following controllability matrix is verified to be full rank.

$$\begin{bmatrix} 0 & -4.55 & 0 & 1.82 \\ -4.55 & 24.93 & 1.82 & -2.40 \\ 24.93 & -275.04 & -2.40 & 23.92 \\ -275.04 & 2246.79 & 23.92 & -196.00 \end{bmatrix} \tag{17}$$

3.2. Framework overview

Figure 3 is the conceptual framework of proposed semi-implicit cascaded PD controller design. The original fourth-order system of cartpole is considered as two coupled second-order systems. This paper uses “subplant1” to denote the pole dynamics, namely x_1, x_2 , and “subplant2” to represent the cart dynamics, which is x_3, x_4 . One direct comprehension of the proposed method is to use one PD controller each for two subplants, respectively. It is expected that if both subplants can be stabilized separately, the overall system can be stable. Nonetheless, the coupling effect inside the model determines that a direct realization of such idea will yield unsatisfying performance. Besides, while the reference signal for the first PD controller, namely “PD1,” is given, the tracking target for the second PD controller, “PD2,” is not available and should be determined in some way. The proposed method solves the above-mentioned

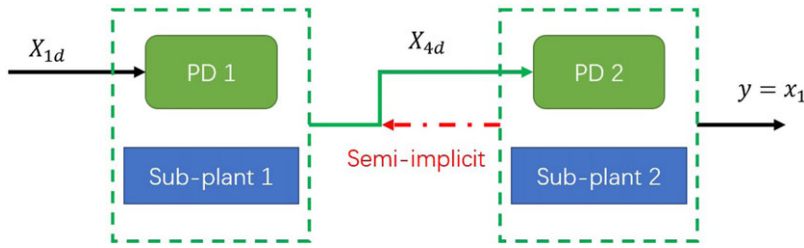


Figure 3. Framework of semi-implicit cascaded PD controller design.

problems and makes cascaded PD controller feasible in this situation. It is achieved by (1) transforming the subplant1 dynamics into an equivalent virtual PD controller, considering the coupling term of subplant2 as a design variable, and then (2) the desired tracking target for subplant2 is derived in a semi-implicit manner through the coupling term, as well as feedback linearization design of PD2. Finally, a cascaded PD controller can be implemented with the coupling effect exploited and solved.

The workflow of proposed controller is specified in the following. Firstly, the reference signal X_{1d} is input into PD1, where the desired virtual torque for subplant1 is computed. The coupling term from subplant2 is transformed to a design variable. In this way, the coupling term represents the desired position for subplant2 X_{4d} , which helps the dynamics of subplant1 to approximate a PD controller to stabilize subplant1. Up to now, the control u is not calculated, so the desired position for subplant2 cannot be calculated explicitly, and require further information from subplant2. Focusing on subplant2 only, a PD controller with feedback linearization can be easily designed. Combining the expression of the controller for subplant2 and that from subplant1, an equation set should be solved, and the expressions for u as well as X_{4d} are derived. The X_{4d} is then fed into PD2 to complete the control for subplant2. Notice that using Gaussian elimination method [25], X_{4d} appears on both sides of the equations, thus representing a semi-implicit process, resembling that of semi-implicit Euler integration method [26]. Besides, PD1 is designed and utilized on top of PD2 during the control process, therefore forming a cascaded relationship.

3.3. Semi-explicit cascaded PD controller design for linear approximated model

This section illustrates the design process of the proposed method using the linearized version of dynamic model (5). The purpose of using a linearized simple model is to make the derivation process tractable, thus enabling a clearer presentation of the idea and rationale. The design process based on original nonlinear model (1) will be given in Section 3.5.

Based on Eq. (5), the dynamics for two subplants can be written explicitly as:

$$\text{subplant1: } \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = Ax_1 + Bx_2 + Cx_4 + Du \end{cases} \quad (18)$$

$$\text{subplant2: } \begin{cases} \dot{x}_3 = x_4 \\ \dot{x}_4 = Ex_1 + Fx_2 + Gx_4 + Hu \end{cases} \quad (19)$$

In Eq. (18), the coupling term from subplant2 is Cx_4 , which will be used as design variable. Borrowing ideas from a serial integrator under the control of a PD controller, if the following equation always holds,

$$Ax_1 + Bx_2 + Cx_4 + Du = k_{p1}e_1 + k_{d1}e_2, \quad (20)$$

then there must exist certain parameters k_{p1}, k_{d1} that makes subplant1 stable. Here, k_{p1}, k_{d1} are the proportional and derivative gains of PD1 controller, and $e_1 = -x_1, e_2 = -x_2$ are the angular and angular velocity errors of subplant1. With this assumption satisfied, Eq. (18) becomes a normal second-order

system controller by a PD controller as follows:

$$\text{subplant1: } \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = k_{p1}e_1 + k_{d1}e_2 \end{cases} \quad (21)$$

Noticing the coupling term x_4 can be controlled by a first-order system in Eq. (19), Eq. (20) is rewritten as

$$Ax_1 + Bx_2 + Cx_{4d} + Du = k_{p1}e_1 + k_{d1}e_2, \quad (22)$$

and that

$$x_{4d} = \frac{k_{p1}e_1 + k_{d1}e_2 - Ax_1 - Bx_2 - Du}{C}, \quad (23)$$

where x_{4d} defines the desired tracking target for subplant2. With the tracking target calculated, the attention shall be shifted to subplant2, where a PD2 controller with feedback linearization is designed as

$$u = \frac{1}{H}(-Ex_1 - Fx_2 - Gx_4 + k_{p2}e_3 + k_{d2}e_4), \quad (24)$$

where k_{p2}, k_{d2} are the proportional and derivative gains of PD2 controller, and e_3, e_4 are the angular and angular velocity errors of subplant2 delicately chosen as

$$e_3 = x_{3d} - x_3 = (x_3 + \Delta t \cdot x_{4d}) - x_3 + x_r - x_3 = \Delta t \cdot x_{4d} + x_r - x_3 \quad (25)$$

$$e_4 = -x_4 \quad (26)$$

in which x_r is user-defined constant target for the cart position. Notice that x_{4d} is a velocity signal. In order to implement PD controller, it should be converted to position signal, hence the manipulation in Eq. (25), where Δt represents sampling time. With that being done, the information of x_{4d} is fully reflected in e_3 , enabling the assignment of Eq. (26). Frankly speaking, x_{4d} is used to construct discrete position targets with reference velocity of the cart being 0. The subplant2 dynamics now becomes a second-order system controlled by PD2 controller with a reference signal related to x_1, x_2 :

$$\text{subplant2: } \begin{cases} \dot{x}_3 = x_4 \\ \dot{x}_4 = k_{p2}(\Delta t \cdot x_{4d} + x_r - x_3) - k_{d2}x_4 \end{cases} \quad (27)$$

Up to now, neither u nor x_{4d} are explicitly expressed. Combining Eqs. (23) and (24) to eliminate u arriving at the following

$$x_{4d} = \frac{-k_{p1}x_1 - k_{d1}x_2 - Ax_1 - Bx_2 - \frac{D}{H}(-Ex_1 - Fx_2 - Gx_4 + k_{p2}\Delta t \cdot x_{4d} + k_{p2}(x_r - x_3) - k_{d2}x_4)}{C}. \quad (28)$$

Consequently, the expression for x_{4d} is derived:

$$x_{4d} = \frac{-(k_{p1} - \frac{DE}{H} + A)x_1 - (k_{d1} - \frac{DF}{H} + B)x_2 - \frac{D}{H}k_{p2}(x_r - x_3) + \frac{D(G+k_{d2})}{H}x_4}{C + \frac{D}{H}k_{p2}\Delta t}. \quad (29)$$

Similarly, the full expression for u is available by substituting Eq. (29) into Eq. (24).

Remark 1 (Joint stability): If subplant1 and subplant2 can be stabilized separately, the whole system would be expected to be stable. Indeed, it can be proved that system (21) and (27) can be stabilized separately [27] under natural assumptions. However, a joint analysis is still required to ensure stability of the fourth-order system, which will be detailed in Sections 3.4 and 3.6.

Remark 2 (Cascaded PD controller): The cascaded PD controller in this paper is different from the conventional one. Conventional cascaded PD controller works in adjacent order of the system, for example, one PD controller to assign desired velocity and the other PD controller to control the acceleration [28]. In contrast, the PD1 controller in this paper serves as the acceleration controller for subplant1,

as well as the calculator of the reference signal for subplant2. And PD2 controller is the acceleration controller for subplant2.

Remark 3 (Semi-implicitness): The proposed cascaded PD controller is semi-implicit in two levels. The most shallow level is that in Eq. (29), x_{4d} appears in both sides of the equations, which means using an unknown term to calculate the result of the same unknown term. It is similar to the manipulation in semi-implicit Euler method for numerical integration. Nonetheless, the source of this semi-implicitness is generated from two sources of information utilized for the derivation of x_{4d} . The first is from transforming subplant1 into a virtual PD controller, and the second is from the PD controller design for subplant2. Both separate design processes finally point to the same target, x_{4d} .

Remark 4 (Transforming PI to PD): In Eq. (26), the authors transform velocity target for x_4 to position target for x_3 using discrete approximation $x_{4d}\Delta t$, so as to implement PD controller for subplant2. More frequently used controller for first-order system is Proportional-Integral (PI) controller. Nevertheless in this paper, for the consistency and simplicity of stability analysis, such a handling is chosen. The challenge for implementing PI controller instead is how to formulate a unified stability analysis together with PD controller.

3.4. Stability analysis using Jacobian matrix for linear approximated model

Following the controller design procedure of Section 3.3, this section presents the stability analysis method for the whole system. Although this is a linearized model, instead of using widely used transfer function, a Jacobian matrix-based method is implemented for wider applicability. The error vectors of the system are constructed, along with their dynamics. The Jacobian matrix is then calculated. By analysing the eigenvalues and eigenvectors of the Jacobian matrix, the stability of the system can be determined.

The error vectors for linear approximated model (5) is

$$\begin{cases} e_1 = -x_1 \\ e_2 = -x_2 \\ e_3 = \Delta t \cdot x_{4d} = \Delta t \cdot \frac{-(k_{p1} + \frac{DE}{H} + A)x_1 - (k_{d1} + \frac{DF}{H} + B)x_2 + \frac{D}{H}k_{p2}(x_r - x_3) - \frac{D(G+k_{d2})}{H}x_4}{C + \frac{D}{H}k_{p2}\Delta t} \\ e_4 = -x_4 \end{cases} \quad (30)$$

Differentiating (30) and replacing all state variables using $e_i, i = 1, 2, 3, 4$:

$$\begin{cases} \dot{e}_1 = e_2 \\ \dot{e}_2 = -\dot{x}_2 \\ = -Ax_1 - Bx_2 - Cx_4 - Du \\ = (A - \frac{DE}{H})e_1 + (B - \frac{DF}{H})e_2 - \frac{Dk_{p2}}{H}e_3 + (C - \frac{DG}{H} - \frac{Dk_{d2}}{H})e_4 \\ \dot{e}_3 = \Delta t \cdot \dot{x}_{4d} + e_4 \\ \dot{e}_4 = -\dot{x}_4 \\ = -Ex_1 - Fx_2 - Gx_4 - Hu \\ = -k_{p2}e_3 - k_{d2}e_4 \end{cases} \quad (31)$$

In the following, the term \dot{e}_3 is managed separately due to its complexity in calculation. Rewriting Eq. (29) using $e_i, i = 1, 2, 3, 4$:

$$x_{4d} = \frac{(k_{p1} - \frac{DE}{H} + A)e_1 + (k_{d1} - \frac{DF}{H} + B)e_2 - \frac{D}{H}k_{p2}e_3 - \frac{D(G+k_{d2})}{H}e_4}{C} \quad (32)$$

Differentiating x_{4d} and substituting Eq. (31):

$$\begin{cases} \dot{x}_{4d} &= [(k_{p1} - \frac{DE}{H} + A)e_2 + (k_{d1} - \frac{DF}{H} + B)\dot{e}_2 - \frac{D(G+k_{d2})}{H}\dot{e}_4]/C \\ &= [a_1e_1 + a_2e_2 + a_3e_3 + a_4e_4]/(C + \frac{D}{H}k_{p2}\Delta t) \\ a_1 &= (B - \frac{DF}{H} + k_{d1})(A - \frac{DE}{H}) \\ a_2 &= (B - \frac{DF}{H})(B - \frac{DF}{H} + k_{d1}) + A - \frac{DE}{H} + k_{p1} \\ a_3 &= \frac{D(G+k_{d2})}{H}k_{p2} + (k_{d1} - \frac{DF}{H} + B)(-\frac{Dk_{p2}}{H} + C - \frac{DC}{H} - \frac{Dk_{d2}}{H}) \\ a_4 &= \frac{Dk_{d2}(G+k_{d2})}{H} - \frac{D}{H}k_{p2} \end{cases} \tag{33}$$

Integrating back to Eq. (31) yields the final expression of \dot{e}_3 :

$$\dot{e}_3 = \Delta t [a_1e_1 + a_2e_2 + a_3e_3 + a_4e_4] / (C + \frac{D}{H}k_{p2}\Delta t) \tag{34}$$

Denote the vector field of Eq. (31) as $F_{\text{linear}}(e_1, e_2, e_3, e_4)$, that is,

$$F_{\text{linear}}(e_1, e_2, e_3, e_4) = \begin{bmatrix} e_2 \\ (A - \frac{DE}{H})e_1 + (B - \frac{DF}{H})e_2 - \frac{Dk_{p2}}{H}e_3 + (C - \frac{DG}{H} - \frac{Dk_{d2}}{H})e_4 \\ \Delta t [a_1e_1 + a_2e_2 + a_3e_3 + a_4e_4] / (C + \frac{D}{H}k_{p2}\Delta t) + e_4 \\ -k_{p2}e_3 - k_{d2}e_4 \end{bmatrix} \tag{35}$$

Then, the Jacobian matrix can be derived easily, denoting $p = C + \frac{D}{H}k_{p2}\Delta t$:

$$DF_{\text{linear}}(e_1, e_2, e_3, e_4) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ A - \frac{DE}{H} & B - \frac{DF}{H} & -\frac{Dk_{p2}}{H} & C - \frac{DG}{H} - \frac{Dk_{d2}}{H} \\ \Delta ta_1/p & \Delta ta_2/p & \Delta ta_3/p & \Delta ta_4/p + 1 \\ 0 & 0 & -k_{p2} & -k_{d2} \end{bmatrix} \tag{36}$$

It is noticeable that $[e_1, e_2, e_3, e_4] = [0, 0, 0, 0]$ is a fixed point of the system (46). Under most conditions, the system is deemed locally asymptotic stable around the fixed point if all the eigenvalues of Eq. (36) have negative real parts, which represents an exponentially decaying term in time domain, and thus means stability. However, the explicit solution to the eigenvalues of Eq. (36) is too complex, and therefore, it better serves as an examiner of the stability through numerical calculation.

3.5. Semi-explicit cascaded PD controller design for nonlinear model

Following the same design procedure in Section 3.3, the proposed method is implemented in nonlinear model (1) in this section. The derivation for nonlinear model of cartpole system is far more complicated than its linear counterpart. Therefore, only necessary derivation is completed by hand, for example, the derivatives of error vectors, and then, the authors use SymPy [29] to complete the final symbolic as well as numerical calculation.

Splitting Eq. (1) into two subplants as follows, denoting $p_2 = (M_1 + M_2)(J + M_2l^2) - M_2^2l^2 \cos^2(x_1)$:

$$\text{subplant1: } \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{\left(-F_1(M_1 + M_2)x_2 - M_2^2l^2x_2^2 \sin(x_1) \cos(x_1) + F_0M_2lx_4 \cos(x_1) \right. \\ \left. + (M_1 + M_2)M_2gl \sin(x_1) - M_2l \cos(x_1)u \right)}{p_2} \end{cases} \tag{37}$$

$$\text{subplant2: } \begin{cases} \dot{x}_3 = x_4 \\ \dot{x}_4 = \frac{\left(F_1M_2lx_2 \cos(x_1) + (J + M_2l^2)M_2lx_2^2 \sin x_1 - F_0(J + M_2l^2)x_4 \right. \\ \left. - M_2^2gl^2 \sin(x_1) \cos(x_1) + (J + M_2l^2)u \right)}{p_2} \end{cases} \tag{38}$$

By transforming the dynamics of subplant1 into a virtual PD controller, and assigning the coupling term x_4 as the reference signal for subplant2, the following can be derived

$$k_{p1}e_1 + k_{d1}e_2 = \frac{\left(-F_1(M_1 + M_2)x_2 - M_2^2l^2x_2^2 \sin(x_1) \cos(x_1) + F_0M_2lx_{4d} \cos(x_1) \right) + (M_1 + M_2)M_2gl \sin(x_1) - M_2l \cos(x_1)u}{p_2} \tag{39}$$

where k_{p1}, k_{d1} are the proportional and derivative gains of PD1 controller, and $e_1 = -x_1, e_2 = -x_2$ are the angular and angular velocity errors of the pole dynamics. If this equation holds, subplant1 is equivalent to Eq. (21). Thus, the expression of x_{4d} can be initially given:

$$x_{4d} = \frac{\left(p_2k_{p1}e_1 + p_2k_{d1}e_2 + F_1(M_1 + M_2)x_2 + M_2^2l^2x_2^2 \sin(x_1) \cos(x_1) \right) - (M_1 + M_2)M_2gl \sin(x_1) + M_2l \cos(x_1)u}{F_0M_2l \cos(x_1)} \tag{40}$$

Turning focus to subplant2 and design a PD2 controller with feedback linearization

$$u = \frac{\left(-F_1M_2lx_2 \cos(x_1) - (J + M_2l^2)M_2lx_2^2 \sin(x_1) + F_0(J + M_2l^2)x_4 \right) + M_2^2gl^2 \sin(x_1) \cos(x_1) + p_2(k_{p2}e_3 + k_{d2}e_4)}{J + M_2l^2}, \tag{41}$$

where e_3, e_4 are the position and velocity errors of the cart, defined as

$$e_3 = x_{3d} - x_3 = (x_3 + \Delta t \cdot x_{4d}) - x_3 + (x_r - x_3) = \Delta t \cdot x_{4d} + (x_r - x_3) \tag{42}$$

$$e_4 = -x_4 \tag{43}$$

so that the original subplant2 becomes a second-order system manipulated by a PD controller as in Eq. (27). Substituting Eq. (41) into Eq. (40) and solving the semi-implicit equation of x_{4d} , the closed-form expression becomes:

$$x_{4d} = \frac{\left(p_2k_{p1}e_1 + p_2k_{d1}e_2 + F_1(M_1 + M_2)x_2 - (M_1 + M_2)M_2gl \sin(x_1) + \frac{M_2l \cos(x_1)}{J + M_2l^2} \left[-F_1M_2lx_2 \cos(x_1) \right] \right) + F_0(J + M_2l^2)x_4 + M_2^2gl^2 \sin(x_1) \cos(x_1) + p_2k_{d2}e_4 + p_2k_{p2}(x_r - x_3)}{F_0M_2l \cos(x_1) - \frac{p_2M_2l \cos(x_1)}{J + M_2l^2} k_{p2} \Delta t} \tag{44}$$

The explicit expression for u can be herein obtained by substituting Eq. (44) with Eq. (42), Eq. (43) back into Eq. (41).

3.6. Stability analysis using Jacobian matrix for nonlinear model

This section provides the stability analysis for system and controller designed in Section 3.5. Compared with the stability analysis of linear model in Section 3.3, the derivation in this section is intimidating and even beyond the human ability within reasonable time period. For example, the calculation of \dot{x}_{4d} would require compound function derivative calculation with trigonometric function existing in both the numerators and denominators. It is even more challenging to derive partial differentiation of \dot{x}_{4d} w.r.t the error vectors. Thus, we first derive all the necessary components required to calculate the final result and input everything into the PC to complete the final symbolic and numerical evaluations.

The error vectors for nonlinear model (1) is

$$\begin{cases} e_1 = -x_1 \\ e_2 = -x_2 \\ e_3 = \Delta t \cdot x_{4d} + (x_r - x_3) \\ e_4 = -x_4 \end{cases} \tag{45}$$

Differentiating Eq. (45) and replacing all state variables using $e_i, i = 1, 2, 3, 4$:

$$\begin{cases} \dot{e}_1 = e_2 \\ \dot{e}_2 = -\dot{x}_2 \\ \phantom{\dot{e}_2} = \frac{\left(F_1(M_1 + M_2)e_2 - M_2^2 l^2 e_2^2 \sin(e_1) \cos(e_1) + F_0 M_2 l e_4 \cos(e_1) \right) + (M_1 + M_2) M_2 g l \sin(e_1) + M_2 l \cos(e_1) u}{p_2} \\ \dot{e}_3 = \Delta t \cdot \dot{x}_{4d} + x_4 \\ \dot{e}_4 = -\dot{x}_4 \\ \phantom{\dot{e}_4} = -k_{p2} e_3 - k_{d2} e_4 \end{cases} \quad (46)$$

In the following, the term \dot{e}_3 is managed separately due to its complexity in calculation. Rewriting Eq. (44) using $e_i, i = 1, 2, 3, 4$. Hence,

$$x_{4d} = \frac{A_2}{B_2}, \quad (47)$$

where A_2 and B_2 are the numerator and denominator, respectively, expressed as below:

$$\begin{aligned} A_2 = & p_2 k_{p1} e_1 + p_2 k_{d1} e_2 - F_1(M_1 + M_2)e_2 + (M_1 + M_2) M_2 g l \sin(e_1) + \frac{M_2 l \cos(e_1)}{J + M_2 l^2} [F_1 M_2 l e_2 \cos(e_1) \\ & - F_0(J + M_2 l^2)e_4 - M_2^2 g l^2 \sin(e_1) \cos(e_1) + p_2(k_{p2} e_3 + k_{d2} e_4)] \end{aligned} \quad (48)$$

$$B_2 = F_0 M_2 l \cos(x_1) \quad (49)$$

Firstly, the term $p_2 = (M_1 + M_2)(J + M_2 l^2) - M_2^2 l^2 \cos^2(x_1)$ should be processed. Rewriting it using error variables and taking the derivative of it based on Eq. (45).

$$p_2 = (M_1 + M_2)(J + M_2 l^2) - M_2^2 l^2 \cos^2(e_1) \quad (50)$$

$$\dot{p}_2 = 2M_2^2 l^2 \cos(e_1) \sin(e_1) e_2 \quad (51)$$

So the derivative of A_2 is

$$\begin{aligned} \dot{A}_2 = & \dot{p}_2 k_{p1} e_1 + p_2 k_{p1} \dot{e}_1 + \dot{p}_2 k_{d1} e_2 + p_2 k_{d1} \dot{e}_2 - F_1(M_1 + M_2)\dot{e}_2 + (M_1 + M_2) M_2 g l \cos(e_1) e_2 \\ & + \frac{M_2 l}{J + M_2 l^2} \left[F_1 M_2 l (\dot{e}_2 \cos^2(e_1) - 2e_2 \cos(e_1) \sin(e_1) e_2) - F_0(J + M_2 l^2) (\dot{e}_4 \cos(e_1) - e_4 \sin(e_1) e_2) \right. \\ & - M_2^2 g l^2 \left(-\sin(e_1) e_2 \frac{\sin(2e_1)}{2} + \cos(e_1) \cos(2e_1) e_2 \right) + k_{d2} (\dot{p}_2 e_4 + p_2 \dot{e}_4) \cos(e_1) - p_2 e_4 \sin(e_1) e_2 \\ & \left. + k_{p2} (\dot{p}_2 e_3 + p_2 (\Delta t \dot{x}_{4d} + e_4)) \cos(e_1) - p_2 e_3 \sin(e_1) e_2 \right]. \end{aligned} \quad (52)$$

Noticeably, \dot{x}_{4d} appears again in \dot{A}_2 , which means one more implicit equation to solve in order to calculate \dot{x}_{4d} . Similarly, the derivative of B_2 is

$$\dot{B}_2 = -F_0 M_2 l \sin(e_1) e_2 \quad (53)$$

At last, the derivative of x_{4d} can be formulated based on fractional derivation rule

$$\dot{x}_{4d} = \frac{\dot{A}_2 B_2 - \dot{B}_2 A_2}{B_2^2}. \quad (54)$$

Solving the above implicit equation arrives at the final equation:

$$\dot{x}_{4d} = \frac{(\dot{A}_2 - p_3 \dot{x}_{4d}) B_2 - \dot{B}_2 A_2}{B_2^2 - p_3 B_2} \quad (55)$$

$$p_3 = \frac{M_2 l}{J + M_2 l^2} k_{p2} p_2 \Delta t \cos(e_1) \quad (56)$$

Table I. Influence of parameter selection.

Params	x_1 overshoot	x_1 oscillate	x_1 converge	x_3 overshoot	x_3 oscillate	x_3 converge
k_{p1}	—	—	+	—	—	—
k_{d1}	—	—	+	+	—	—
k_{p2}	+	+	+	+	+	—
k_{d2}	+	+	+	—	+	+

Denote the vector field of Eq. (46) as $F_{\text{nonlinear}}(e_1, e_2, e_3, e_4)$

$$F_{\text{nonlinear}}(e_1, e_2, e_3, e_4) = \begin{bmatrix} \frac{e_2}{\left(F_1(M_1 + M_2)e_2 - M_2^2 l^2 e_2^2 \sin(e_1) \cos(e_1) + F_0 M_2 l e_4 \cos(e_1) \right) + (M_1 + M_2) M_2 g l \sin(e_1) + M_2 l \cos(e_1) u} \\ p_2 \\ \Delta t \dot{x}_{4d} + e_4 \\ -k_{p2} e_3 - k_{d2} e_4 \end{bmatrix} \quad (57)$$

Then, the Jacobian matrix can be derived easily:

$$DF_{\text{nonlinear}}(e_1, e_2, e_3, e_4) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{\partial \dot{e}_2}{\partial e_1} & \frac{\partial \dot{e}_2}{\partial e_2} & \frac{\partial \dot{e}_2}{\partial e_3} & \frac{\partial \dot{e}_2}{\partial e_4} \\ \frac{\partial \dot{e}_3}{\partial e_1} & \frac{\partial \dot{e}_3}{\partial e_2} & \frac{\partial \dot{e}_3}{\partial e_3} & \frac{\partial \dot{e}_3}{\partial e_4} \\ 0 & 0 & -k_{p2} & -k_{d2} \end{bmatrix} \quad (58)$$

In Eq. (58), $\frac{\partial \dot{e}_i}{\partial e_j}, i = 2, 3; j = 1, 2, 3, 4$ are the partial derivatives of corresponding terms w.r.t the error variables. Those are too complicated to be manually derived and are solved instead by the PC symbolically. It is noticeable that $[e_1, e_2, e_3, e_4] = [0, 0, 0, 0]$ is a fixed point of the system (46). If all eigenvalues of Eq. (58) have negative real parts, then the system is sure to be locally asymptotic stable around the fixed point.

Remark 5 (Parameters selection): In this paper, an analytical solution to the specific range of those parameters that stabilizes the system is not provided, which is a potential research direction. However, a critical advantage of the proposed controller lies in similarly intuitive tuning process as conventional PID controller. With that being said, based on the derived Jacobian matrix, one can use whatever optimization method to find an approximate range of parameters that stabilizes the system by examining the eigenvalues of the resulting Jacobian matrix. In addition, to further understand the influence of each parameter on the system performance, an ablation study is implemented (around the chosen parameters) on both the baseline and proposed controller. The observation has been concluded in Table I. In the table, “+” means increased overshoot, increased oscillation or increased convergence rate under the increase of corresponding parameters accordingly, and vice versa for “—.” The results (see Appendix) also show that the chosen parameters are Pareto optimal. That is, by perturbing the current parameters, no further simultaneous improvement on the performance of x_1 and x_3 . This is also important to ensure that the baseline method has achieved its optimal performance.

Remark 6 (Stability analysis using eigenvalues): In control theory, the Jacobian matrix of the closed system can be used to ensure stability, by examining that all eigenvalues have strictly negative real parts. Under this condition, the Jacobian matrix is called a stable matrix (or sometimes Hurwitz matrix) and that the system is asymptotically stable around the equilibrium points [30].

Table II. Parameter of dynamic model.

Parameters	Description	Values
g	Gravity acceleration	9.8 m/s ²
M	Mass of the cart	0.5 kg
m	Mass of the pole	0.2 kg
l	Length of the pole	0.3 m
J	Inertia of link	0.006 kg m ²
Δt	Simulation timestep	0.01 s
F_0	Friction factor of the pole	0.1 N/m/s
F_1	Friction factor of the cart	0.1 N/m/s

Table III. Parameter of simulation environment.

Description	Values
Simulation timestep	0.01 s
ODE solver	Semi-implicit Euler
Control timestep	0.01 s
Episode	8 s
Env	OpenAI Gym

4. Simulation

This section implements the proposed method to cartpole system and retrieve numerical simulation results as well as stability analysis. Firstly, necessary parameters for the dynamic model, simulation environment and cascaded PD controller are specified. Secondly, the simulation results are presented, closely followed by stability analysis. The results on linear approximated model are foremost presented and then comes the nonlinear model.

4.1. Parameters specification

Table II shows the parameters of the dynamic models (1) and (5). Table III lists the parameters of the simulation environment setup. We are using OpenAI Gym [31] environment to carry out the simulation. Gym is a popular simulation platform with both continuous and discrete environment setup written in Python. Table IV are the parameters of the proposed method and the baseline. Similarly, The initial states are set as $[x_1, x_2, x_3, x_4] = [0.5, 0, 0, 0]$. The reference signal for cartpole system is $[x_{1d}, x_{2d}, x_{3d}, x_{4d}] = [0, 0, 0.5, 0]$, which represent the desired angle and angular velocity are all 0. Naturally, it is hoped that the ending position of the cart is not too far away from its initial location, which means also that the linear velocity of the cart should converge to 0 with passage of time. Under these considerations, the cost of an episode is defined as

$$L = - \sum_{\text{timestep}=0}^{800} e_1^2 + 0.1e_2^2 + e_3^2, \tag{59}$$

then many optimization algorithms can be used to find the optimal controller parameters for cascaded PD controller. In this paper, Bayesian optimization is implemented as a baseline, plus manual fine-tune to determine the final parameters. Bayesian optimization is chosen because of its data efficiency in optimization process [32]. By constructing a surrogate probabilistic model, it can retrieve the next location with high probability of getting better result. However, one disadvantage is that it may only find a sub-optimal solution [33]. Thus, using Bayesian optimization as a baseline helps us get closer

Table IV. Parameter of controllers.

Method	k_{p1}	k_{d1}	k_{p2}	k_{d2}
Ours	240.00	34.10	60.00	79.18
Baseline	200.00	39.48	20.82	60.76

to the optimal parameters quickly, and then the parameters are fine tuned manually aiming for best performance.

4.2. Double-loop PD controller as baseline

Double-loop PD controller is a well-established method for the control of the cartpole systems and is also one of the most fundamental controller that inspires the invention of many other methods [34]. The intuitive idea of double-loop PD controller lies in using PD controller each for the stabilization of the pole and the cart, respectively. The final control input is a direction summation of those two PD controllers. The formula is represented as

$$u_{\text{double}} = \bar{k}_{p1}e_1 + \bar{k}_{d1}e_2 + \bar{k}_{p2}e_3 + \bar{k}_{d2}e_4, \quad (60)$$

where e_1, e_2, e_3, e_4 are errors for x_1, x_2, x_3, x_4 , respectively, and $\bar{k}_{p1}, \bar{k}_{d1}, \bar{k}_{p2}, \bar{k}_{d2}$ are the parameters. Conceivably, although this controller has been proved effective in practice, its implementation is too intuitive to exploit any information that the dynamical system has to provide.

4.3. Results and evaluation

4.3.1. Results of linear approximated model

The results of the linear approximated model are depicted in Figs. 4–8. Figure 4 is the output of the angle of the pole. Compared with the baseline, the proposed method outputs slightly higher magnitude and frequency oscillation, with shorter settling time. Figure 5 also coincides with Fig. 4 by showing corresponding oscillation in the velocity level. Figure 6 depicts the position of the cart, where the baseline method presents a very slow asymptotic convergence. Figure 7 shows the outputs of x_4 . Lastly, Fig. 8 presents how the torque input changes alongside the episode. It also oscillates at first and gradually converges to 0. A severe overshoot is observed for the baseline controller at the very beginning, but the oscillation is alleviated afterwards.

Next, the Jacobian matrix of this linear system is to be investigated to understand some phenomena that happened in the simulation results. Substituting the controller parameters into Eq. (36), the Jacobian matrix can be calculated out:

$$DF_{\text{linear}}(e_1, e_2, e_3, e_4) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 24.50 & -4.17 & 200.00 & 212.95 \\ -4.75 & -0.62 & 71.26 & 75.58 \\ 0 & 0 & -80.00 & -85.18 \end{bmatrix} \quad (61)$$

and the eigenvalues with corresponding eigenvectors are

$$\begin{cases} \lambda_1 = -3.60 \\ \lambda_2 = -4.05 + 11.24j \\ \lambda_3 = -4.05 - 11.24j \\ \lambda_4 = -1.24 \end{cases} \quad (62)$$

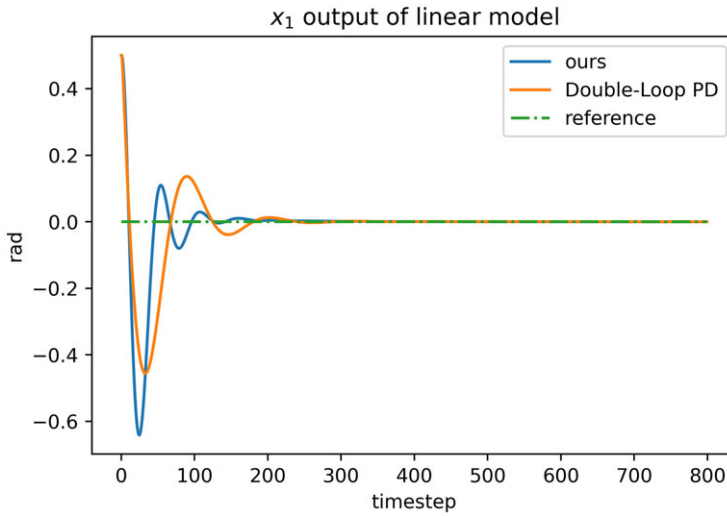


Figure 4. x_1 output of linear model.

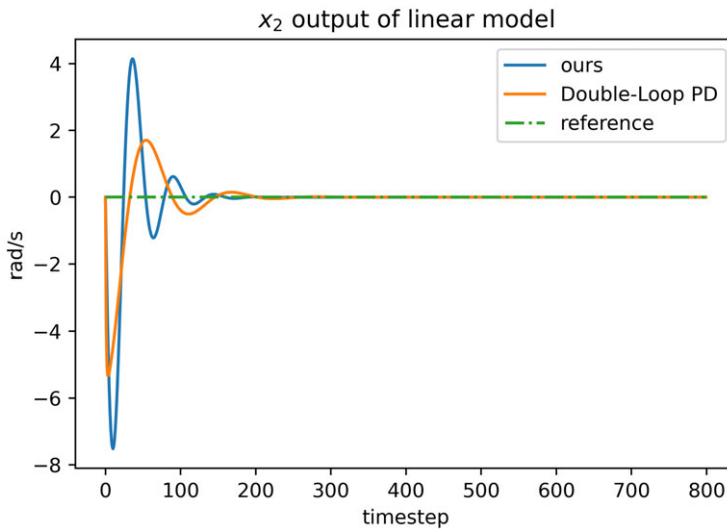


Figure 5. x_2 output of linear model.

$$\begin{cases} \vec{v}_1 = [-0.10 - 0.14j, 0.36 + 0.49j, -0.37 - 0.50j, 0.30 + 0.40j]^T \\ \vec{v}_2 = [0.063 - 0.036j, 0.15 + 0.85j, 0.20 + 0.44j, -0.21 - 0.32j]^T \\ \vec{v}_3 = [-0.047 - 0.049j, -0.36 + 0.73j, -0.30 + 0.34j, 0.27 - 0.23j]^T \\ \vec{v}_4 = [0.080 + 0.029j, -0.099 - 0.036j, 0.94 + 0.34j, -0.73 - 0.26j]^T \end{cases} \tag{63}$$

In Eqs. (62) and (63), j is the imaginary unit and T represents transpose of vectors. All the eigenvalues have negative real part, which ensure local stability near the equilibrium point.

Remark 7 (Global stability): The system is at least locally asymptotic stability but not guaranteed to be globally stable. The attempt to elevate the stability conclusion to global is intuitive and has its background from Markus-Yamabe’s theorem [35]. However, Markus-Yamabe’s theorem only holds for second-order system, and many counterexamples have been discovered for higher-order systems [36].

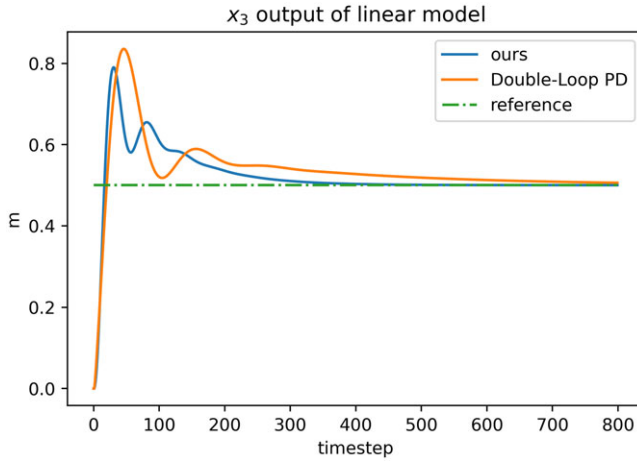


Figure 6. x_3 output of linear model.

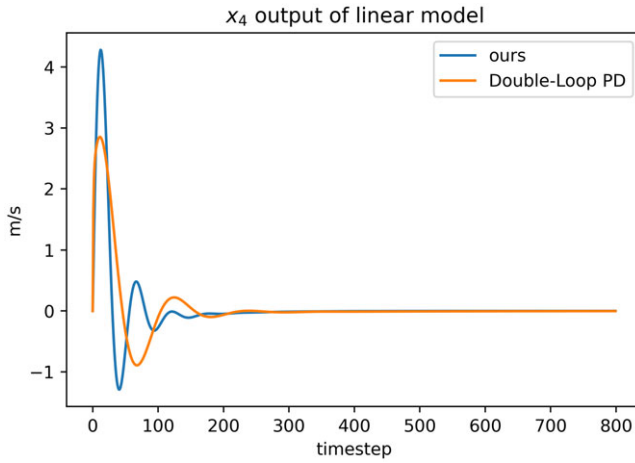


Figure 7. x_4 output of linear model.

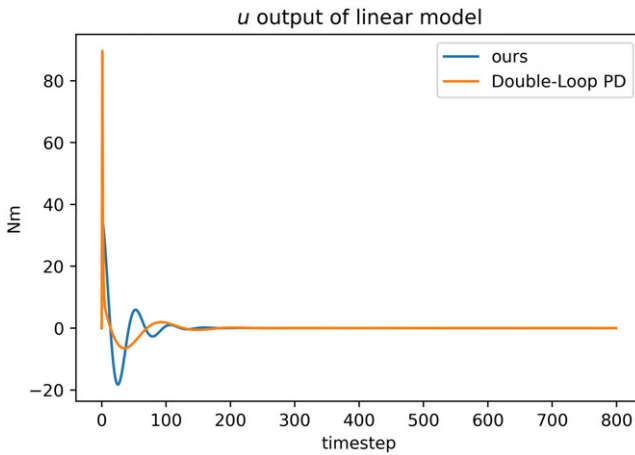


Figure 8. u output of linear model.

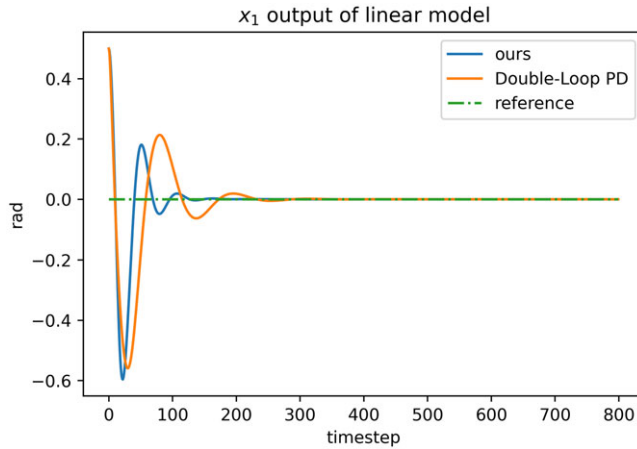


Figure 9. x_1 output of nonlinear model.

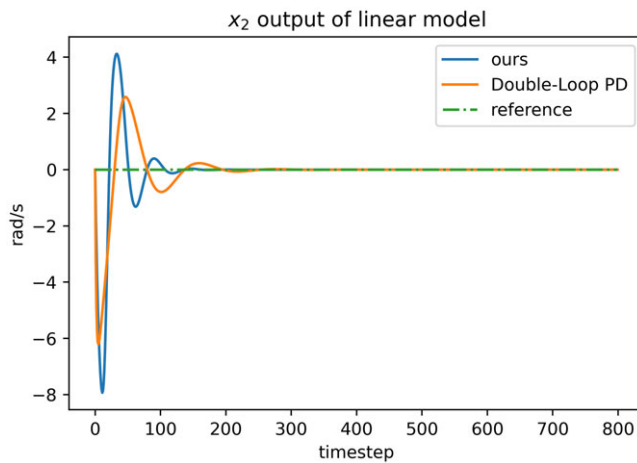


Figure 10. x_2 output of nonlinear model.

With that being said, in the simulation, the system can be stabilized whatever the initial states, as long as the pole is placed within the upper half plane.

4.3.2. Results of nonlinear model

The results of the original nonlinear model are depicted in Figs. 9–20, which are similar to the linear case. Figure 9 is the output of the angle of the pole. It converges to 0 soon after some oscillation of decaying magnitude. Figure 10 shows the profile of the angular velocity of the pole, which shares similar pattern with Fig. 9. In comparison, the oscillation magnitude of baseline controller is similar to the proposed method, but with a smaller frequency and therefore slower convergence rate. Figure 11 depicts the position of the cart. The proposed method converges much faster than baseline controller without comprising the convergence performance of x_1 , while the baseline controller shows a very slow asymptotic convergence of x_3 . Figure 12 shows the outputs of x_4 . Lastly, Fig. 13 presents how the torque input changes alongside the episode. It also oscillates at first and gradually converge to 0. A big overshoot is rendered by the baseline controller.

Next, the Jacobian matrix of this nonlinear system is to be investigated for local stability analysis. Substituting the controller parameters into Eq. (58), the Jacobian matrix near the fixed point

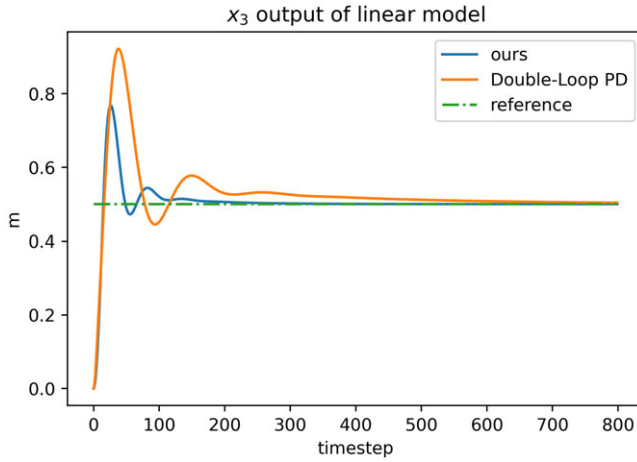


Figure 11. x_3 output of nonlinear model.

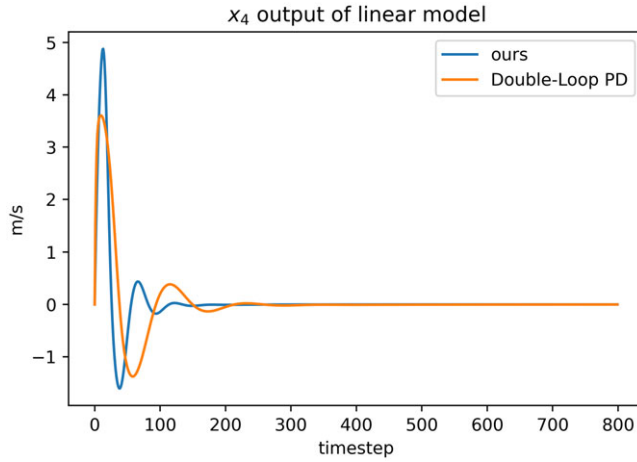


Figure 12. x_4 output of nonlinear model.

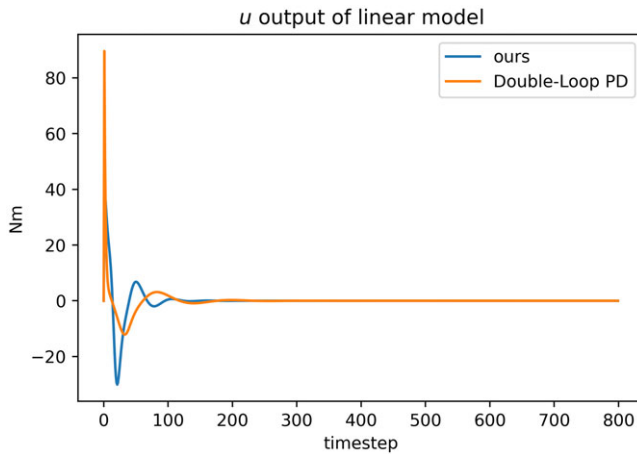


Figure 13. u output of nonlinear model.

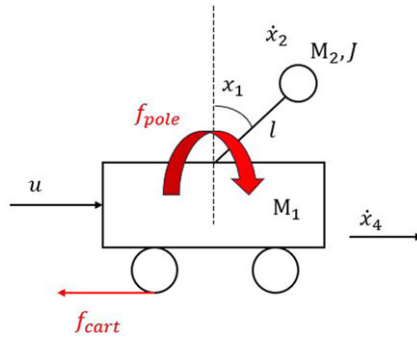


Figure 14. Friction force analysis.

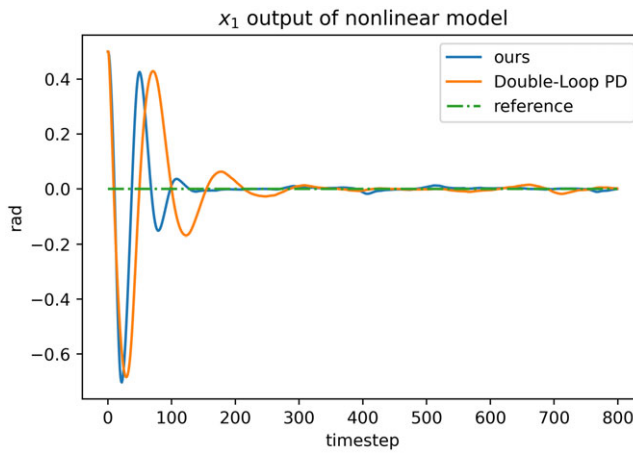


Figure 15. x_1 output of nonlinear model.

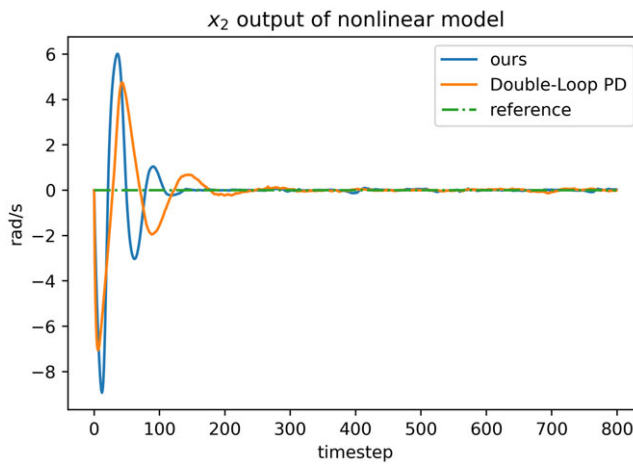


Figure 16. x_2 output of nonlinear model.

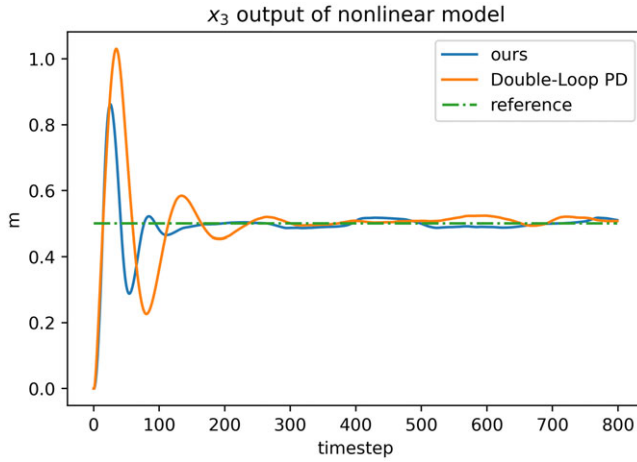


Figure 17. x_3 output of nonlinear model.

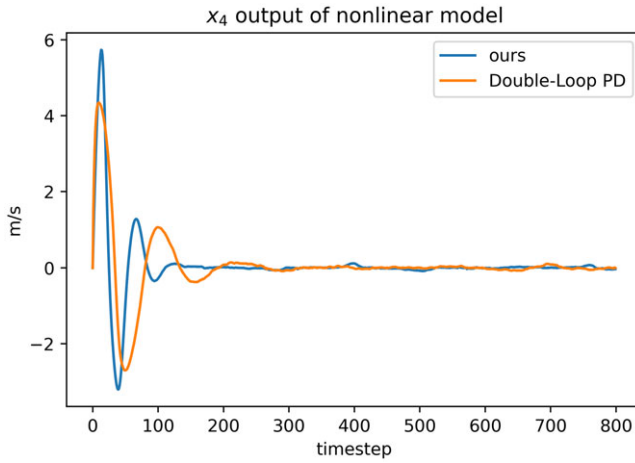


Figure 18. x_4 output of nonlinear model.

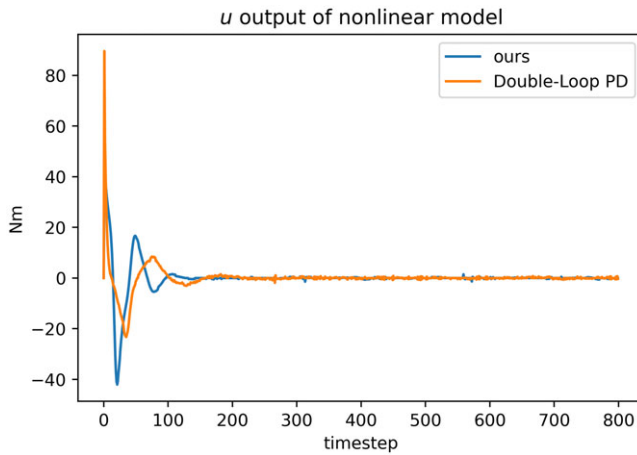


Figure 19. u output of nonlinear model.

Table V. Performance indices overview.

Method	t_1/s	t_2/s	MAE ₁ /rad	MAE ₂ /m	Energy
Ours/Linear	2.00	4.00	0.02654	0.5212	11,940.29
Baseline/Linear	2.50	8.00	0.03234	0.5366	10,847.11
Ours/Nonlinear	1.50	3.00	0.02326	0.5034	17,521.81
Baseline/Nonlinear	3.00	7.00	0.03762	0.5277	15,224.77

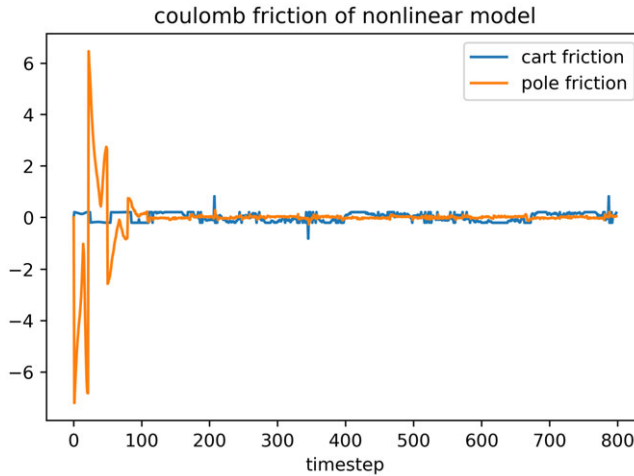


Figure 20. Coulomb friction of nonlinear model.

$[e_1, e_2, e_3, e_4] = [0, 0, 0, 0]$ can be calculated. The results are exactly the same with Eqs. (36)–(63). This also verifies the derivation process, since in the equilibrium point, the linear model should be equivalent to the nonlinear model.

4.3.3. Performance indices overview

To conclude the results showcase section, a performance overview of both linear and nonlinear models with two controllers respectively is shown in Table V. t_1, t_2 are the convergence time for the pole and the cart, respectively. MAE₁, MAE₂ are the mean absolute error of the pole and the cart respectively. “energy” means the mean square sum of control input, which represents the energy consumption of the controller. We can safely conclude that the proposed method is outstanding compared with double-loop PD controller in terms of convergence rates and tracking errors of x_1, x_3 . However, the ensued cost of superiority lies in increased control efforts and slightly severer oscillation. The advantages of proposed controller originate from the exploitation of the internal dynamics of the model through a semi-implicit process, thus a system-level consistent intermediate target is derived. However, for double-loop PD controller, the control efforts required by the cart and the pole are competing, resulting in a compromise between performance of those two and limiting the overall performance.

4.3.4. Robust performance

To illustrate the robustness of the proposed controller, this subsection presents the results of simulation under both Coulomb friction and random noise. Coulomb friction is an approximation of dry friction in practice, including both the static friction and kinetic friction, with different coefficients. According to the Coulomb’s law of friction, the magnitude of the friction between two dry sliding surface is independent of the magnitude of the relative velocity. However, the direction of the friction is opposed to the

relative velocity. Therefore, Coulomb friction is a highly nonlinear type of disturbance [37]. Accordingly, the cartpole system dynamics with disturbance is

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{\left(-F_1(M_1 + M_2)x_2 - M_2^2 l^2 x_2^2 \sin(x_1) \cos(x_1) + F_0 M_2 l x_4 \cos(x_1) \right. \\ \qquad \qquad \qquad \left. + (M_1 + M_2) M_2 g l \sin(x_1) - M_2 l \cos(x_1)(u - f_{\text{cart}}) \right)}{p_2} + \frac{f_{\text{pole}} R_{\text{joint}}}{J} + d_1 \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = \frac{\left(F_1 M_2 l x_2 \cos(x_1) + (J + M_2 l^2) M_2 l x_2^2 \sin x_1 - F_0 (J + M_2 l^2) x_4 \right) \\ \qquad \qquad \qquad \left. - M_2^2 g l^2 \sin(x_1) \cos(x_1) + (J + M_2 l^2)(u - f_{\text{cart}}) \right)}{p_2} + d_2 \end{cases} \quad (64)$$

where f_{cart} is the friction acting on the cart because of rolling. This force will counteract the control input u directly, and therefore u is directly deducted by f_{cart} . f_{pole} is the friction acting on the revolte joint that connects the cart and the pole. To convert it into angular acceleration, it is multiplied by the radius of the joint $R_{\text{joint}} = 0.01$ m and then divided by the inertia J . $-1 \leq d_1, d_2 \leq 1$ are bounded random total disturbance added to the acceleration. The force analysis figure is plotted in Fig. 14.

According to the Coulomb friction theory, the friction is proportional to the normal force, and cannot revert the relative motions between two surfaces. Firstly, f_{cart} is considered. The sliding surfaces are the wheels and the ground. This is a rolling motion, and the friction coefficient is chosen slightly smaller than sliding friction. The static friction coefficient $c_{\text{cart_static}} = 0.2$, and the kinetic friction coefficient $c_{\text{cart_kine}} = 0.05$. The normal force is affected by both the mass gravity and the lifting force generated by the centrifugal force of the pole, but cannot be negative. Accordingly, the normal force of the cart is:

$$f_{\text{cart_nor}} = \max \left[0, (M_1 + M_2)g - \int_{r=0}^l \frac{M_2}{l} \dot{x}_2^2 r dr \right] \quad (65)$$

On the other hand, f_{cart} cannot revert the influence of u , which means if $u - f_{\text{cart}}$ has different sign with u , then $f_{\text{cart}} = u$. Therefore, the total expression of f_{cart} is:

$$f_{\text{cart}} = \begin{cases} c_{\text{cart_static}} \times f_{\text{cart_nor}}, & \text{if static and } [(u \times (u - f_{\text{cart}})) \geq 0] \\ c_{\text{cart_kine}} \times f_{\text{cart_nor}}, & \text{if kinetic and } [(u \times (u - f_{\text{cart}})) \geq 0] \\ 0, & \text{if } [(u \times (u - f_{\text{cart}})) < 0] \end{cases} \quad (66)$$

The f_{pole} is modelled as follows. The normal force of the pole is a vector summation of the centrifugal force and the force generated by the cart acceleration. Therefore, the normal force should be expressed as:

$$f_{\text{pole_nor}} = \left\| \left[\int_{r=0}^l \frac{M_2}{l} \dot{x}_2^2 r dr \cos(x_1), M_2 \dot{x}_4 - \int_{r=0}^l \frac{M_2}{l} \dot{x}_2^2 r dr \sin(x_1) \right] \right\|_2 \quad (67)$$

Similarly, the full expression of f_{pole} is:

$$f_{\text{pole}} = \begin{cases} c_{\text{pole_static}} \times f_{\text{pole_nor}}, & \text{if static} \\ c_{\text{pole_kine}} \times f_{\text{pole_nor}}, & \text{if kinetic} \end{cases} \quad (68)$$

where the static friction coefficient $c_{\text{pole_static}} = 0.5$, and the kinetic one $c_{\text{pole_kine}} = 0.3$.

Figures 15–20 are the comparative results of the proposed controller and double-loop PD controller under added disturbance. In comparison with previous sections without disturbance, the results here are similar, only with some oscillation and chattering near the equilibrium point. This is due to the existence of friction and random noise, which slightly impairs the control performance. Nonetheless, the system is still stable under both controllers. Besides, Fig. 20 illustrates the Coulomb friction profile, which features abrupt change, nonlinearity and clipping as the theory suggests.

5. Conclusion

A control method for underactuated cartpole systems based on cascaded PD controller is proposed in this article. The gist is to transform the pole dynamics into a virtual PD controller, with the coupling term exploited as the design variable. The desired value of the coupling term x_{Ad} is then fed into the cart dynamics for the realization of a second PD controller. The expressions of the control input as well as x_{Ad} are derived by solving a semi-implicit equation. This method absorbs all the blessings that conventional PID controller has to offer (i.e., very simple in design and relatively intuitive to understand) and can be carried out on the original state-space equations without coordinate transformation, along with all the assumptions ensue. Besides, contrary to many other PID controller research, a stability analysis method for the fourth-order cascaded PD controller is proposed using Jacobian matrix of the residual system, although it only concludes locally asymptotic stability in this system and bears with it some drawbacks. The simulation results illustrate the advantages of proposed method in terms of stabilizing the cart and the pole simultaneously compared with widely used double-loop PD controller. In addition, the robustness against Coulomb friction and random noise is verified through simulation. The superiority is derived from the exploitation of internal dynamical structure of the system through solving a semi-implicit equation.

Considering that this is a preliminary research of a control method for underactuated cartpole systems using cascaded PD controller, there are many efforts in urgent need to solve the following problems. Firstly, a stability analysis approach is required that can reach the conclusion of global stability. For example, Lyapunov-based stability theorem may be a good alternative to Jacobian matrix-based method in this article. With that being said, in the numerical simulation, the cartpole system can be stabilized with a wide range of values of the initial states of the system. Noticeably, for some systems, the Jacobian matrix-based analysis can actually conclude global stability using relevant theorem proposed by Markus and Yamabe [38] for high-dimension systems. Moving one step forward, how to ensure that all eigenvalues of a high dimensional (>2) Jacobian matrix are negative everywhere is an open question. A closed-form calculation is obviously infeasible for complicated matrix like in Eq. (58). Secondly, although this paper is targeted on cartpole system only, the authors envision that the proposed method should be able to be implemented to other kinds of underactuated systems and be expanded to a class of underactuated systems. Last but not least, a systematic and theoretic way of parameter selection should be investigated. The method of tuning in this article is still a combination of Bayesian optimization and trials. To achieve this, a more capable method for stability proof should be employed, for example, Lyapunov stability theorem.

References

- [1] Z. Chen, Y. Liu, W. He, H.-Y. Qiao and H. Ji, "Adaptive-neural-network-based trajectory tracking control for a nonholonomic wheeled mobile robot with velocity constraints," *IEEE Trans. Ind. Electron.* **68**(10), 5057–5067 (2021).
- [2] S. Heshmati-alamdari, A. Nikou and D. V. Dimarogonas, "Robust trajectory tracking control for underactuated autonomous underwater vehicles in uncertain environments," *IEEE Trans. Autom. Sci. Eng.* **18**(3), 1288–1301 (2021).
- [3] Y. Liu, W. Zhan, M. Xing, Y. Wu, R. Xu and X. Wu, "Boundary control of a rotating and length-varying flexible robotic manipulator system," *IEEE Trans. Syst. Man Cybern. Syst.* **52**(1), 377–386 (2022).
- [4] L. Messikh, E.-H. Guechi and S. Blai, "Stabilization of the cart-inverted-pendulum system using state-feedback pole-independent MPC controllers," *Sensors (Basel, Switzerland)* **22**(1), 243 (2021).
- [5] R. Banerjee and A. Pal, "Stabilization of Inverted Pendulum on Cart Based on Pole Placement and LQR," *In: 2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET)* (2018) pp. 1–5.
- [6] H. Balaga and M. Deepthi, "Stabilization of Cart Inverted Pendulum System Using LQR, Two-Loop Pid, and Regional Pole Placement Techniques," *In: 2021 Asian Conference on Innovation in Technology (ASIANCON)* (2021) pp. 1–6.
- [7] M. A. Eizadiyan and M. Naseriyan, "Control of inverted pendulum cart system by use of PID controller," *Sci. Int.* **27**(2), 1063–1068 (2015).
- [8] J.-J. E. Slotine and W. Li, *Applied nonlinear control* (Prentice Hall, Englewood Cliffs, NJ, 1991).
- [9] Y. Shao and J. Li, "Modeling and switching tracking control for a class of cart-pendulum systems driven by DC motor," *IEEE Access* **8**, 44858–44866 (2020).
- [10] J. Jiang and A. Astolfi, "Stabilization of a class of underactuated nonlinear systems via underactuated back-stepping," *IEEE Trans. Autom. Control* **66**(11), 5429–5435 (2020).

- [11] S. H. Lakmesari, M. J. Mahmoodabadi and M. Y. Ibrahim, “Fuzzy logic and gradient descent-based optimal adaptive robust controller with inverted pendulum verification,” *Chaos Solitons Fractals* **151**, 111257 (2021).
- [12] P. N. Dao and Y.-C. Liu, “Adaptive reinforcement learning strategy with sliding mode control for unknown and disturbed wheeled inverted pendulum,” *Int. J. Control Autom. Syst.* **19**(2), 1139–1150 (2020).
- [13] P. Ordaz and A. Poznyak, “‘KL’-gain adaptation for attractive ellipsoid method,” *IMA J. Math. Control Inf.* **32**(3), 447–469 (2015).
- [14] E. A. Kennedy, E. King and H. Tran, “Real-time implementation and analysis of a modified energy based controller for the swing-up of an inverted pendulum on a cart,” *Eur. J. Control* **50**, 176–187 (2019).
- [15] R. Ranasinghe, P. Manoharan, A. Pallegedara and D. Kodithuwakku, “Develop a Cascaded Observer Based Controller to an Inverted Pendulum Platform for Experimental Research and Teaching,” *In: 2022 IEEE IAS Global Conference on Emerging Technologies (GlobConET)* (2022) pp. 215–220.
- [16] M. D. Ratolikar and R. P. Kumar, “Neural Network Control of an Inverted Pendulum on a Two DOF Cart Moving in the Vertical Plane,” *In: 2021 6th International Conference on Robotics and Automation Engineering (ICRAE)* (2021) pp. 84–88.
- [17] Z. Peng, Y. Jiang and J. Wang, “Event-triggered dynamic surface control of an underactuated autonomous surface vehicle for target enclosing,” *IEEE Tran. Ind. Electron.* **68**(4), 3402–3412 (2021).
- [18] T. Rojsiraphisal, S. Mobayen, J. H. Asad, M. T. Vu, A. Chang and J. Puangmalai, “Fast terminal sliding control of underactuated robotic systems based on disturbance observer with experimental validation,” *Mathematics* **9**(16), 1935 (2021).
- [19] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. M. O. Heess, T. Erez, Y. Tassa, D. Silver and D. Wierstra, “Continuous control with deep reinforcement learning,” *CoRR*, abs/1509.02971 (2015).
- [20] Q. Shi, H.-K. Lam, C. Xuan and M. Chen, “Adaptive neuro-fuzzy PID controller based on twin delayed deep deterministic policy gradient algorithm,” *Neurocomputing* **402**, 183–194 (2020).
- [21] S. A. Hiremath and N. Bajcinca, “DNN Based Learning Algorithm for State Constrained Stochastic Control of a 2D Cartpole System,” *In: 2022 European Control Conference (ECC)* (2022) pp. 1132–1139.
- [22] H. K. Lam and F. H. F. Leung, “Fuzzy controller with stability and performance rules for nonlinear systems,” *Fuzzy Set. Syst.* **158**(2), 147–163 (2007).
- [23] C. T. Chen, *Linear System Theory and Design*, The Oxford Series in Electrical and Computer Engineering (Oxford University Press, Oxford, 2014).
- [24] P. Tomei, “A simple PD controller for robots with elastic joints,” *IEEE Trans. Autom. Control* **36**(10), 1208–1213 (1991).
- [25] N. J. Higham, “Gaussian elimination,” *Wiley Interdiscip. Rev. Comput. Stat.* **3**(3), 230–238 (2011).
- [26] C. Deng and W. Liu, “Semi-implicit Euler-Maruyama method for non-linear time-changed stochastic differential equations,” *BIT Numer. Math.* **60**, 1133–1151 (2019).
- [27] C. Zhao and L. Guo, “On the capability of PID control for nonlinear uncertain systems,” *IFAC-PapersOnLine* **50**(1), 1521–1526 (2017).
- [28] F. A. A. Andrade, I. P. Guedes, G. F. Carvalho, A. R. L. Zachi, D. B. Haddad, L. F. Almeida, A. G. de Melo and M. F. Pinto, “Unmanned aerial vehicles motion control with fuzzy tuning of cascaded-PID gains,” *Machines* **10**(1), 12 (2021).
- [29] A. Meurer, C. P. Smith, M. Paprocki, Ořej Čertík, S. B. Kirpichev, M. Rocklin, A. M. I. Kumar, S. Ivanov, J. K. Moore, S. Singh, T. Rathnayake, S. Vig, B. E. Granger, R. P. Muller, F. Bonazzi, H. Gupta, S. Vats, F. Johansson, F. Pedregosa, M. J. Curry, A. R. Terrel, Štěpán Roučka, A. Saboo, I. Fernando, S. Kulal, R. Cimrman and A. Scopatz, “Sympy: Symbolic computing in python,” *PeerJ Comput. Sci.* **3**, e103 (2017).
- [30] H. K. Khalil, *Nonlinear Systems*, 3rd edition (Prentice Hall, Upper Saddle River, NJ, 2008).
- [31] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang and W. Zaremba, “Openai gym,” *arXiv preprint*, arXiv:1606.01540 (2016).
- [32] M. Neumann-Brosig, A. Marco, D. Schwarzmann and S. Trimpe, “Data-efficient autotuning with bayesian optimization: An industrial control study,” *IEEE Trans. Control Syst. Technol.* **28**(3), 730–740 (2020).
- [33] M. A. Solis and S. S. Thomas, “Generalized state-feedback controller synthesis for underactuated systems through bayesian optimization,” *ArXiv*, abs/2103.17158 (2021).
- [34] X. Wang, Z. Sun and S. Zai, “Application of Double-Loop PID Controller in the Inversed Pendulum Real-Time Control System,” *In: Green Communications and Networks: Proceedings of the International Conference on Green Communications and Networks (GCN 2011)* (Springer, Dordrecht, 2012) pp. 619–626.
- [35] R. Feßler, “A proof of the two-dimensional markus-yamabe stability conjecture and a generalization,” *Ann. Pol. Math.* **62**(1), 45–74 (1995).
- [36] N. V. Kuznetsov, O. A. Kuznetsova, D. Koznov, R. N. Mokaev and B. Andrievsky, “Counterexamples to the Kalman conjectures,” *IFAC-PapersOnLine* **51**(33), 138–143 (2018). 5th IFAC Conference on Analysis and Control of Chaotic Systems CHAOS 2018.
- [37] P. Lötstedt, “Coulomb friction in two-dimensional rigid body systems,” *ZAMM* **61**, 605–615 (1981).
- [38] L. Markus and H. Yamabe, “Global stability criteria for differential systems,” *Osaka Math. J.* **12**(2), 305–317 (1960).

Appendix A: Ablation study of semi-implicit cascaded PD controller

This appendix illustrates the ablation study of the proposed method, where the parameters of the controllers are perturbed one by one and illustrate the outputs of x_1, x_3 in order to see the influence of each parameter. The presented results not only feature the process of manual tuning but also prove that the chosen parameters in the paper are the OPTIMAL ones, by showing that the perturbation of parameters can only render Pareto optimum w.r.t x_1, x_3 convergence. Of all the pictures, the green line most approximates the actual performance, which lies in the middle of the perturbation bounds. By showing that the increase of performance on x_1/x_3 means the decrease of the other, the authors make sure that the chosen parameters are nearly Pareto Optimal.

Figures 21 and 22 show the x_1, x_3 outputs respectively under the perturbation of k_{p1} . When k_{p1} increases, the convergence of x_1 is accelerated, and its oscillation is suppressed. However, the convergence rate of x_3 is decreased. Figures 23 and 24 show the x_1, x_3 outputs respectively under the perturbation of k_{p2} , which shares the same discussion with k_{p1} .

Figures 25 and 26 show the x_1, x_3 outputs respectively under the perturbation of k_{d1} . When k_{d1} increases, the convergence of x_1 is decelerated, with smaller oscillation. In the meantime, the convergence of x_3 is also deteriorated. Note that too high-frequency oscillation is unfavourable, and the chosen parameter actually strikes a balance by leaning to the convergence performance. Figures 27 and 28 show the x_1, x_3 outputs respectively under the perturbation of k_{d2} , whose discussion is similar to k_{d1} .

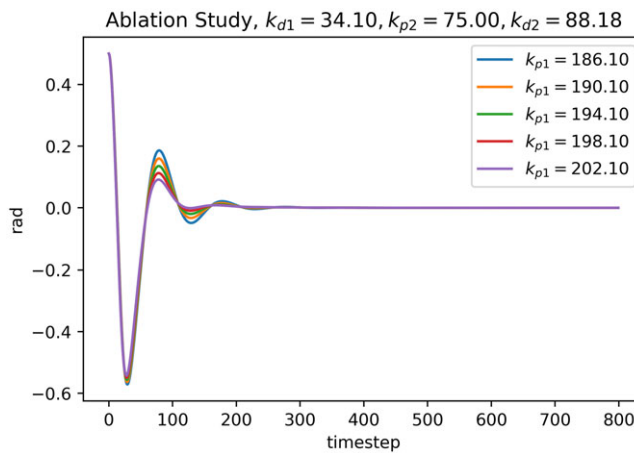


Figure 21. x_1 outputs perturbing k_{p1} of proposed method.

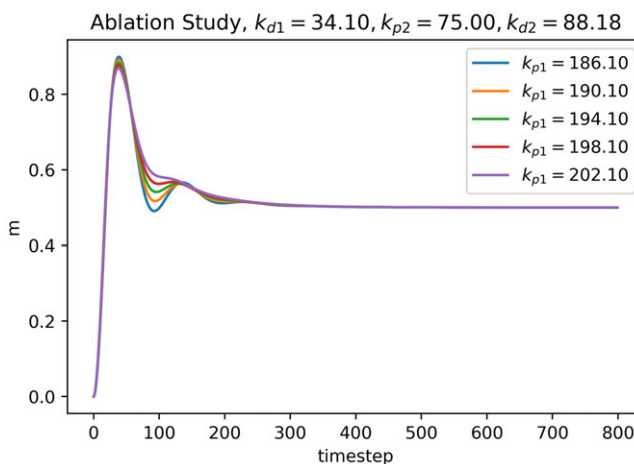


Figure 22. x_3 outputs perturbing k_{p1} of proposed method.

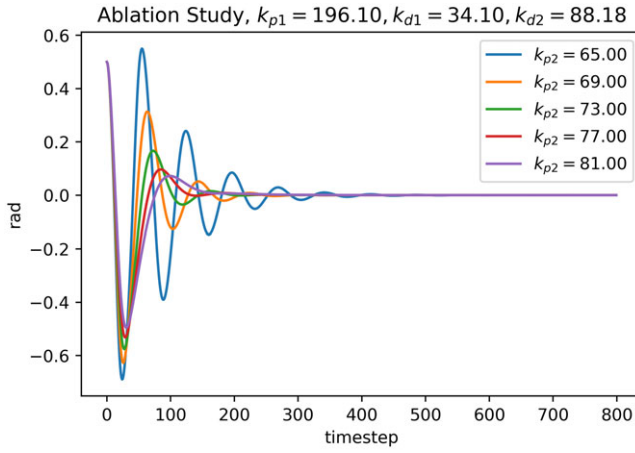


Figure 23. x_1 outputs perturbing k_{p2} of proposed method.

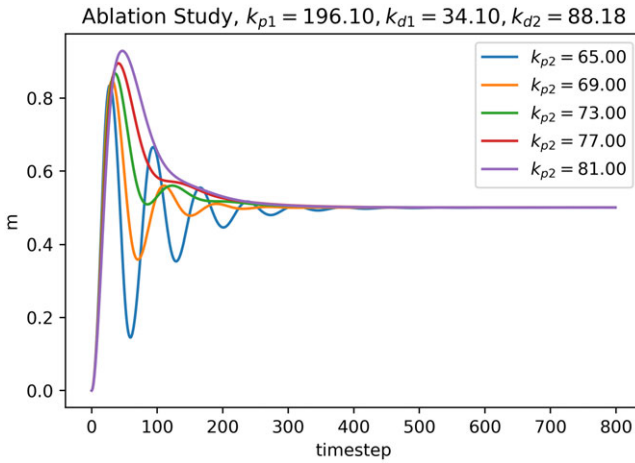


Figure 24. x_3 outputs perturbing k_{p2} of proposed method.

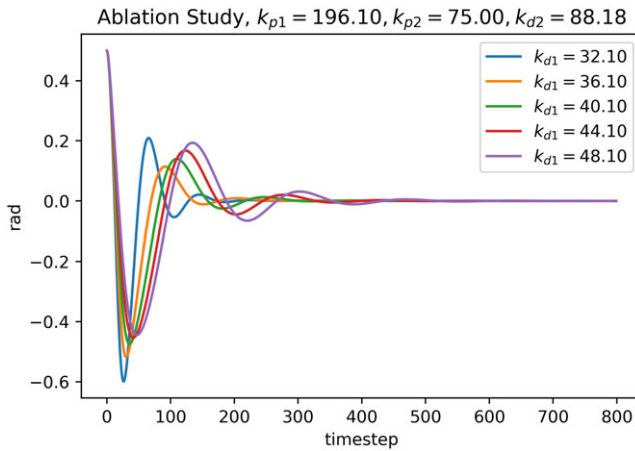


Figure 25. x_1 outputs perturbing k_{d1} of proposed method.

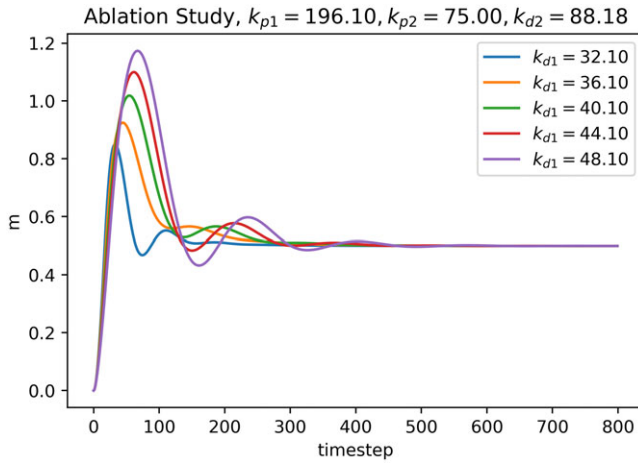


Figure 26. x_3 outputs perturbing k_{d1} of proposed method.

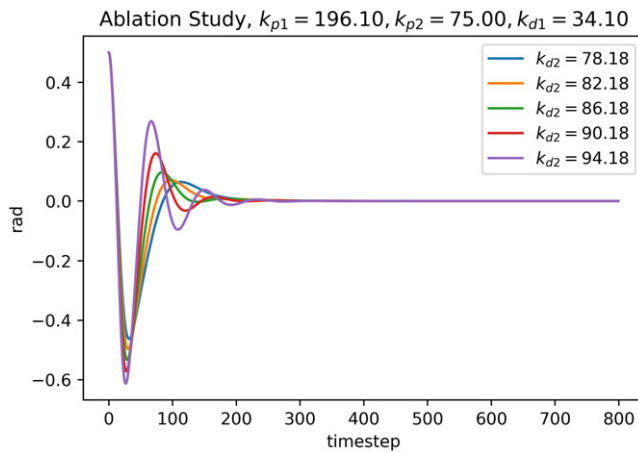


Figure 27. x_1 outputs perturbing k_{d2} of proposed method.

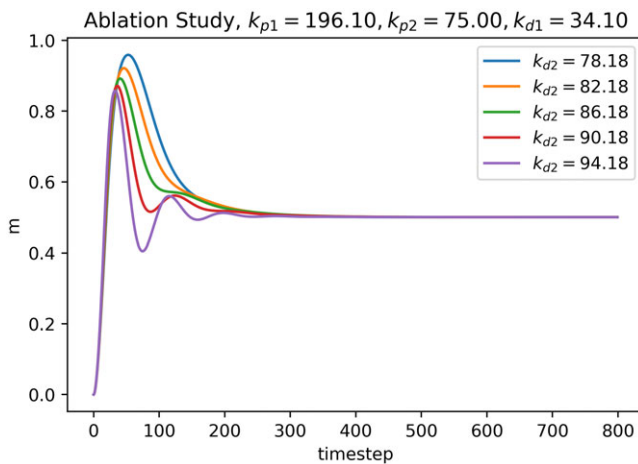


Figure 28. x_3 outputs perturbing k_{d2} of proposed method.

Appendix B: Ablation study of double-loop PD controller

This appendix illustrates the ablation study of baseline method, where the parameters of the controllers are perturbed one by one and illustrate the outputs of x_1, x_3 in order to see the influence of each parameter. The presented results not only feature the process of manual tuning but also prove that the chosen parameters in the paper are nearly the OPTIMAL ones, by showing that the perturbation of parameters can only render Pareto optimum w.r.t x_1, x_3 convergence.

Figures 29 and 30 show the x_1, x_3 outputs respectively under the perturbation of k_{p1} . When k_{p1} increases, the convergence of x_1 is accelerated, and its oscillation is suppressed. However, the convergence rate of x_3 is slightly decreased. Figures 31 and 32 show the x_1, x_3 outputs respectively under the perturbation of k_{p2} , which shares the opposite discussion with k_{p1} . When k_{p2} increases, the performance of x_1 is worse at the cost of better x_3 convergence.

Figures 33 and 34 show the x_1, x_3 outputs respectively under the perturbation of k_{d1} . The decrease of k_{d1} leads to more precise tracking of x_1 . However, it takes longer for x_3 to reach the reference location. Figures 35 and 36 show the x_1, x_3 outputs respectively under the perturbation of k_{d2} , which illustrates a trade-off between overshoot and settling time for x_1 and x_3 . The selected parameters achieve a middle performance.

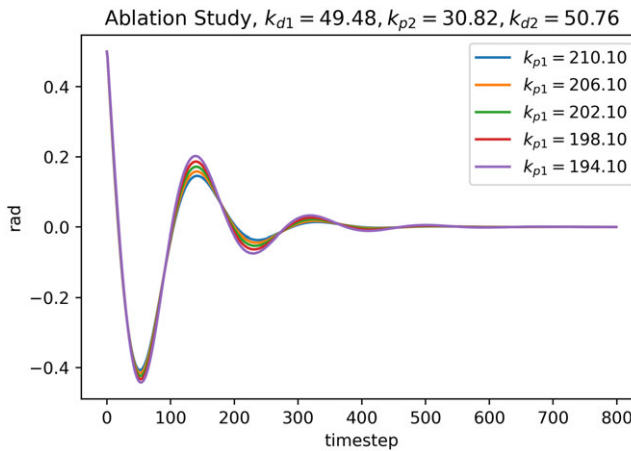


Figure 29. x_1 outputs perturbing k_{p1} of baseline method.

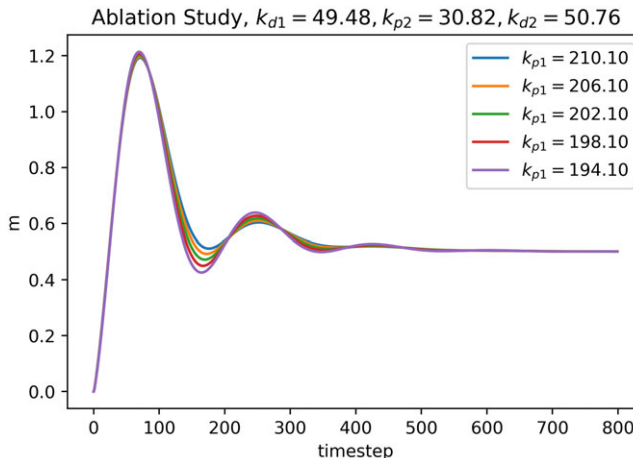


Figure 30. x_3 outputs perturbing k_{p1} of baseline method.

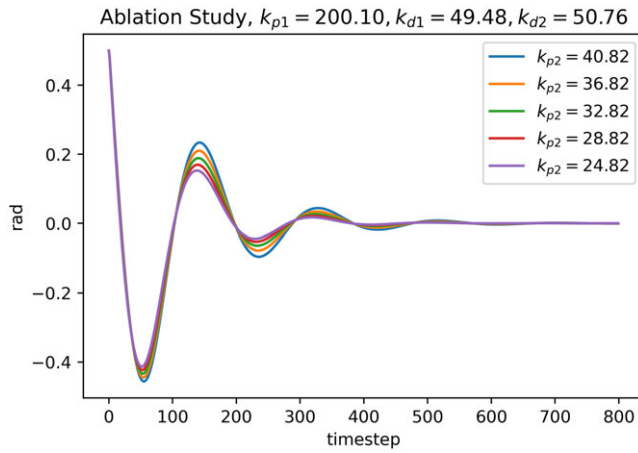


Figure 31. x_1 outputs perturbing k_{p2} of baseline method.

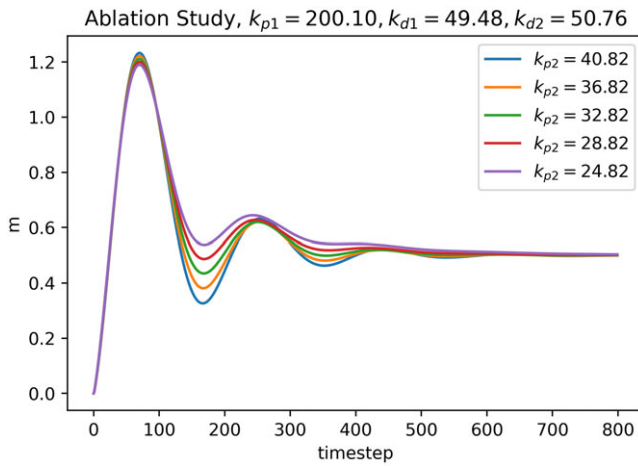


Figure 32. x_3 outputs perturbing k_{p2} of baseline method.

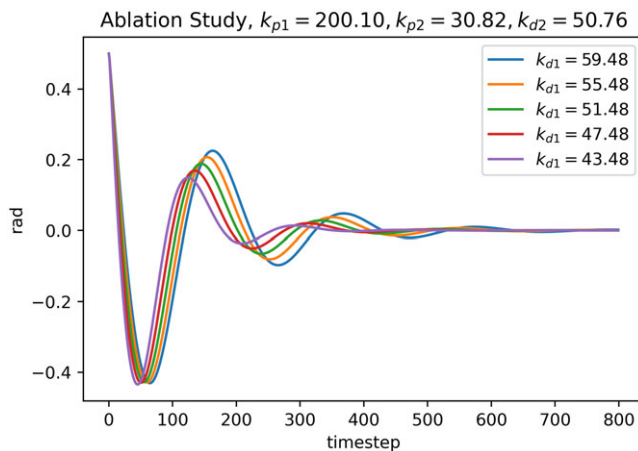


Figure 33. x_1 outputs perturbing k_{d1} of baseline method.

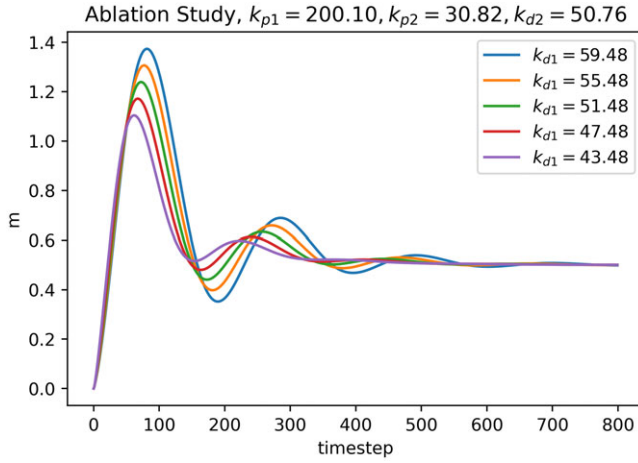


Figure 34. x_3 outputs perturbing k_{d1} of baseline method.

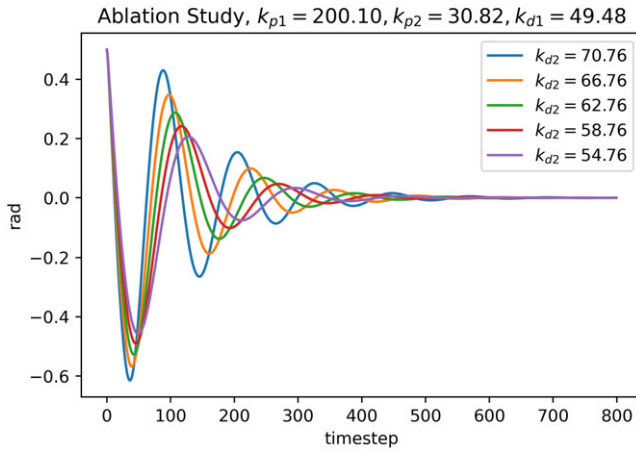


Figure 35. x_1 outputs perturbing k_{d2} of baseline method.

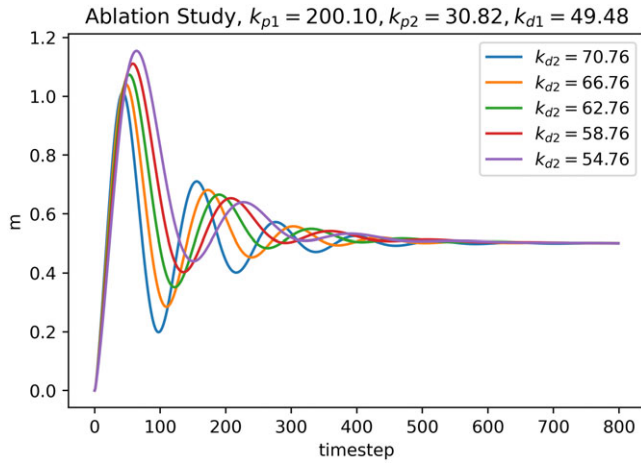


Figure 36. x_3 outputs perturbing k_{d2} of baseline method.