Available online at www.sciencedirect.com

SciVerse ScienceDirect

Fuzzy Sets and Systems 207 (2012) 77–93

FUZZY sets and systems

www.elsevier.com/locate/fss

Synthesized design of a fuzzy logic controller for an underactuated unicycle

Jian-Xin Xu[a,b,*], Zhao-Qin Guo[a,b], Tong Heng Lee[a,b]

[a] *Department of Electrical and Computer Engineering, National University of Singapore, 4 Engineering Drive 3, Singapore 117576, Singapore*
[b] *Graduate School for Integrative Sciences and Engineering, National University of Singapore, Singapore 117456, Singapore*

Received 18 June 2011; received in revised form 2 April 2012; accepted 4 April 2012
Available online 11 April 2012

Abstract

In this paper, we propose synthesized design of a fuzzy logic controller (FLC) for control of an underactuated unicycle system. The FLC objective is velocity control of the wheel while keeping the pendulum upright, which is an unstable equilibrium. The synthesized design consists of three phases. First, the FLC structures including the number of rules, membership functions, inference and parametric relations are chosen based on heuristic knowledge about the unicycle. Second, on the basis of a linearized model and linear feedback, the FLC output parameters are determined quantitatively for stabilization of the unicycle. Third, the FLC output parameters are tuned using an iterative learning tuning (ILT) algorithm, which minimizes an objective function that specifies the desired unicycle performance. The rationale for the synthesized FLC design is full utilization of the available information, which is achieved by combining model-based and model-free designs, and hence improves the FLC performance. We minimize the number of FLC rules and fuzzy labels. Six rules are used for regulation or setpoint tasks, whereas 10 rules are used with extra integral control to eliminate steady-state errors induced by system uncertainties and disturbances. Only two fuzzy labels are adopted for each fuzzy variable. The ILT process consists of two phases, exploration for stabilization and exploitation for better performance. The effectiveness of the proposed FLC is validated using intensive simulations and comparisons.

Keywords: Fuzzy logic controller; Underactuated unicycle; Model-free design; Model-based design; Iterative learning tuning

1. Introduction

Unicycles have many advantages compared to traditional vehicles with two or more wheels. Unicycles require less material for manufacture, reduce energy consumption and save space in traveling and parking. Riding on a unicycle is also attractive for exercise or entertainment. However, the unicycle system, which can be regarded as a wheeled inverted pendulum, is inherently unstable. In this work, the control objective is to use only one actuator to perform tracking tasks for the wheel while balancing the pendulum. This type of system is defined as an underactuated system, with fewer actuators than the number of independent variables to be controlled [1–14]. For the control of underactuated systems, conventional control designs for fully actuated systems are not applicable.

* Corresponding author at: Department of Electrical and Computer Engineering, National University of Singapore, 4 Engineering Drive 3, Singapore 117576, Singapore. Fax: +65 7791103.
E-mail addresses: elexujx@nus.edu.sg (J.-X. Xu), guozhaoqin@gmail.com (Z.-Q. Guo), eleleeth@nus.edu.sg (T.H. Lee).

0165-0114/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.fss.2012.04.004

Various control strategies for a wheeled inverted pendulum and similar underactuated mechanical systems have been developed [1–14]. Seo et al. applied a linear LQR controller to a two-wheeled inverted pendulum and obtained controller gains based on a linearized model determined using heuristically selected weighting matrices [1]. Pathak et al. designed velocity and position controllers based on a reformulated dynamic model using a partial feedback linearization method [2]. Sliding mode control has been studied for the control of underactuated systems with parameter uncertainties and external disturbance [3–5]. In most of the above studies, the controller design requires a relatively accurate system model, which is usually quite difficult to obtain in practice.

Fuzzy logic control (FLC) has been widely used in robotics control and applications as it provides a user-friendly interface for controller design and the designers' knowledge can be incorporated directly as a set of fuzzy rules. FLC design is generally model-free, which is complementary to model-based control design. FLC offers a nonlinear controller with robustness for systems with parametric and functional uncertainties, as well as disturbances [6,7,15,16]. With its flexible structure design and parameter selection, FLC design can easily be incorporated with other control methods, such as adaptive control [6,17], genetic algorithms [15], learning control [18], linear LQR control [19], and H_2 and H_∞ control [20].

Li and Zu investigated adaptive FLC of dynamic balance and motion for a wheeled inverted pendulum with parametric and functional uncertainties [6]. The use of fuzzy approximations avoids the need to develop a highly accurate mathematical model. However, the fuzzy approximator uses 324 fuzzy rules and computation for the control signal could be time-consuming. Tao et al. proposed a fuzzy hierarchical swing-up and sliding position controller for an inverted pendulum-cart system [7]. The FLC uses 18 fuzzy rules associated with 36 controller parameters. The FLC design is based on the authors' instinctive knowledge, whereas the sliding mode controller design is model-based. A limit of this method is that it may be sensitive to exogenous disturbances. In general, application of model-free FLC design for real platforms could be problematic considering the large number of fuzzy rules and controller parameters to be determined and the limited heuristic knowledge for complex dynamics for systems such as unicycles. A better alternative is to synergize a model-free design with heuristic knowledge and a model-based design with an available plant model, so that all information relevant to the control system can be fully utilized in FLC design.

In the present study our aim was to develop a pure FLC without incorporating any model-based controller, and hence an accurate mathematical model was not required. The FLC design is simple and easy to apply to a unicycle. A Takagi–Sugeno (TS)-type FLC using full-state feedback was developed for velocity control and pendulum balancing. On the basis of human experience, an FLC with three input variables and six fuzzy rules was first explored for regulation tasks. Six controller parameters need to be tuned. Three range parameters specify the universe of discourse of the input variables and were chosen using heuristic knowledge such as the physical boundaries for the input variables. The other three are output parameters that determine control output values for individual rules. In general, it is difficult to determine the output parameters based only on human intuition or heuristics.

The limitations of heuristic knowledge and model-free design motivated us to explore design based partially on a model. To capture the correct feedback control action for each state, a linear controller is designed based on a linearized model of the unicycle. The linearization is performed around the desired balance position, when the inverted pendulum is upright. Necessary conditions for the feedback gains are established to ensure local stability around the desired equilibrium point. Next, conditions for the selection of the FLC output parameters are identified, which makes parameter determination much easier. Considering the presence of uncertainties and disturbances in practice, an FLC with integral action is further proposed.

To improve FLC performance and avoid tedious manual tuning, a partially model-free iterative learning tuning (ILT) method is used to tune the FLC output parameters. The ILT process consists of three steps. First, a number of cost functions chosen to characterize unicycle behavior are calculated according to unicycle responses in the time domain. Next, an iterative learning algorithm is derived to minimize the cost functions and then used to update the FLC output parameters. After parametric updating, the same motion control task is executed again and the unicycle response is recorded for the next ILT run. The ILT only requires the process gradient information in selecting the learning gain, and hence guarantees learning convergence. Furthermore, the gradient can be numerically approximated when gradient information is not available [21]. In this sense, the ILT is partially model-free.

The main contributions and originality of the paper are summarized as follows.

(i) Synthesized FLC design is proposed for regulation and setpoint control of an underactuated unicycle system in the presence of disturbances and model uncertainties. The synthesized design consists of three phases: determination

of the FLC structure through heuristic knowledge about the unicycle; quantitative determination of the output parameters for stabilization of the unicycle; and tuning of the FLC output parameters using ILT. The main idea behind the proposed methodology is to maximize utilization of all the information available, which is achieved by combining partially model-based and partially model-free designs, and hence improve the FLC performance.

(ii) Compared with model-based control design, the synthesized FLC design has the advantage that it does not require an accurate model of unicycle system. The fuzzy rules, membership functions, direction of control action for the fuzzy output parameters, and cost function selection for the ILT are all determined by human experience. The relative amplitudes of the output parameters for different fuzzy rules are determined based on a linearized model. However, the unicycle parameter values are not required in the linearized model. Compared with conventional FLC and FLCs designed in [6,7], the synthesized FLC developed in this paper has fewer fuzzy rules and fewer parameters to be determined, which implies a simpler design.

(iii) The synthesized FLC is intelligent in the sense that learning is incorporated in FLC parameter tuning. Selection of the objective functions is based on human experience and the choice of different key features from the unicycle responses is flexible for meeting different control requirements. The learning process is similar to human learning, which utilizes knowledge about not only successful but also unsuccessful trials.

The remainder of the paper is organized as follows. In Section 2, the unicycle system is formulated. In Section 3, FLC designs are elaborated, including the basic FLC structure, FLC speed control, and FLC speed control with integral action. The principle of ILT is presented in Section 4 and the FLC output parameters are updated using ILT. Conclusions are drawn in Section 5.

2. Unicycle system description

Fig. 1 shows a model of the unicycle, which consists of a pendulum and a wheel with a wide surface. Therefore, the lateral stability of the unicycle is guaranteed and the unicycle is limited to longitudinal motion. The wheel motion is defined along the surface. The wheel displacement and velocity are denoted by x and $\dot{x}$, respectively, and the positive direction is rightward. ϕ is the angular rotation of the wheel and the positive direction is clockwise. We have $\phi = x/r$, where r is the radius of the wheel. θ is the tilt angle of the pendulum (saddle) with the upright position as the zero point and clockwise rotation as the positive direction. $\dot{\theta}$ is the angular velocity of the pendulum. φ is the angle of tilt for the road. For a unicycle traveling on a flat surface, $\varphi = 0$. f_r is the friction between the wheel and the ground. τ is the torque generated by the motor and acting on the wheel, with clockwise rotation as the positive direction, which is also the control input u to the system. Note that the motor driving the wheel is directly mounted on the pendulum. There is a reaction torque $\tau_{reaction}$ applied to the pendulum and $\tau_{reaction} = -\tau$. Other parameters of the unicycle system are defined as:

m_w mass of the wheel;
m_p mass of the pendulum;
I_w rotation inertia of the wheel;
I_p rotation inertia of the pendulum; and
l distance between the center of gravity of the pendulum and the center of the wheel.

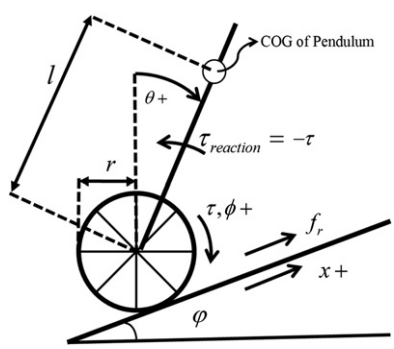

Fig. 1. Unicycle model.

A Lagrangian mechanics method is used to derive the mathematical model of the unicycle system (Appendix A), which leads to a second-order nonlinear model given by

$$a\ddot{x} + b\ddot{\theta} - m_p l \sin(\theta+\varphi)\dot{\theta}^2 + \sin\varphi(m_p+m_w)g = \frac{1}{r}(\tau - r f_r), \tag{1}$$

$$b\ddot{x} + c\ddot{\theta} - m_p l g \sin\theta = -\tau, \tag{2}$$

where $a = m_w + m_p + I_w/r^2$, $b = m_p l\cos(x_3+\varphi)$ and $c = I_p + m_p l^2$. In this work, f_r is modeled as a combination of viscous friction and Coulomb friction, that is, $f_r = f_v\dot{x} + f_c\,\mathrm{sgn}(\dot{x})$, where f_v is a viscous-friction constant, f_c is a Coulomb-friction constant and $\mathrm{sgn}(\cdot)$ is a signum function.

The control objective for the wheel is to follow a given velocity reference v_r. The control objective for the pendulum is to balance at equilibrium ($\theta = \theta_e$, $\dot{\theta} = 0$). Note that θ_e varies according to the friction f_r and slope φ. We define the state space vector as $\mathbf{x} = [x_1, x_2, x_3]^T = [\dot{x}, \theta, \dot{\theta}]^T$ and the state error vector as $\mathbf{e} = [e_1, e_2, e_3]^T = [x_1 - v_r, \theta - 0, \dot{\theta} - 0]^T$.

3. FLC

We adopt TS-type FLC for simplicity of the controller structure and ease of FLC parameter tuning. FLC with three input variables is first explored for regulation tasks. We focus on determining the output parameter for each fuzzy rule, which is critical in TS FLC design. First, to simplify the FLC design, a minimum number of fuzzy rules are used, which yields a minimum number of output parameters to be determined. Second, the direction of the control action (sign of the output parameters, positive or negative) is chosen by following the direction of the pendulum tilt angle. Next, the relative amplitudes of the output parameters for different rules are analyzed using the feedback knowledge of a linear controller, which is designed based on a linearized model of the unicycle. Linearization is performed around the desired balance position, namely when the inverted pendulum is upright. Necessary conditions for the feedback gains are established to ensure local stability around the desired equilibrium point. Considering the presence of uncertainties and disturbances for a practical unicycle, FLC with integral action is further proposed.

3.1. FLC structure

States ($\dot{x}$, θ, $\dot{\theta}$) are the FLC input variables. Each of the input variables is associated with two fuzzy sets, positive (P) and negative (N), and the degree to which set they belong is determined by the membership functions illustrated in Fig. 2.

Let μ_P and μ_N denote the degree of matching to the fuzzy sets P and N, respectively. μ_P is given by

$$\mu_P(e_i) = \begin{cases} 0 & (e_i < -m_i), \\ 2\left(\dfrac{e_i + m_i}{2m_i}\right)^2 & (-m_i \le e_i \le 0), \\ 1 - 2\left(\dfrac{m_i - e_i}{2m_i}\right)^2 & (0 \le e_i \le m_i), \\ 1 & (e_i > m_i), \end{cases} \tag{3}$$

and $\mu_N = 1 - \mu_P$.

The parameter ranges $[m_1, m_2, m_3]$ are determined under consideration of the physical constraints of the unicycle system. m_1 specifies the wheel speed range, and one choice is to let m_1 equal the maximum speed that is physically achievable. The maximum wheel speed is determined by the motor capacity. In this work, we assume that the maximum speed is limited to approximately 1.8 km/h, and hence we choose $m_1 = 0.5$ m/s. m_2 specifies the range for angular displacement of the pendulum. To keep the pendulum moving within a safe range around the balance position, we choose $m_2 = \pi/3$ rad, which is 60°. m_3 specifies the range for the angular velocity of the pendulum. We found that m_3 is less influential on FLC performance, and we chose a value of 2 rad/s for this study.

FLC consists of rules in the following form:

$$R^i: \text{If } (e_1 \text{ is } A_i) \text{ AND } (e_2 \text{ is } B_i) \text{ AND } (e_3 \text{ is } C_i), \text{ THEN } (u_i = \tau_i),$$

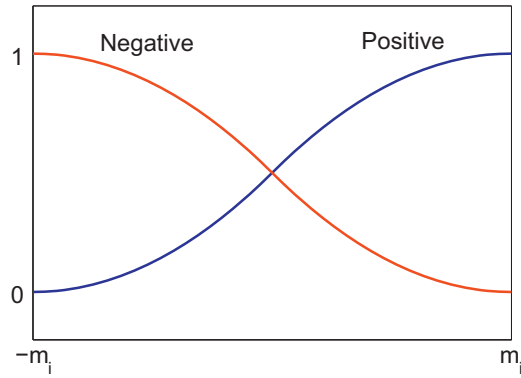

Fig. 2. FLC membership functions. The range for the inputs is specified by the interval $[-m_i, m_i]$, $i = 1, 2, 3, 4$.

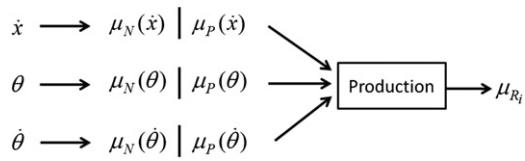

Fig. 3. TS-type fuzzy inference for the ith rule of the FLC. Each input yields two membership values, μ_N and μ_P. The AND logic operator is chosen for production of the fuzzy membership values.

where $A_i, B_i, C_i \in \{\mathrm{P}, \mathrm{N}\}$ are fuzzy sets, u_i is the rule output and τ_i is a constant representing the inferred control torque. Each fuzzy rule describes a specific relationship between the FLC inputs and output.

Each rule contributes to the final FLC output according to the matching for the IF part of the fuzzy rule. The output τ_i for each rule is weighted by the firing strength μ_{R_i} for that rule. The calculation for the firing strength for each rule is shown in Fig. 3. The TS-type fuzzy inference takes a weighted average of the individual outputs for each rule, and the final output of the fuzzy controller is computed as

$$u = \frac{\sum_{i=1}^{n} \mu_{R_i} u_i}{\sum_{i=1}^{n} \mu_{R_i}}, \tag{4}$$

where n is the number of rules.

3.2. Fuzzy logic speed controller

To derive fuzzy rules, we start with a regulation control task, namely, $v_r = 0$. Since there are three input variables and each variable is associated with two fuzzy sets P and N, we could have $2^3 = 8$ fuzzy rules in total. However, to simplify the FLC design, the number of fuzzy rules is minimized by combining several cases into one. Table 1 summarizes the six FLC rules, where $\times$ denotes either N or P. Fig. 4 shows a graphical representation of the six rules corresponding to six scenarios. Let $(\cdot, \cdot, \cdot)$ denote a fuzzy state with respect to three variables $(\dot{x}, \theta, \dot{\theta})$ and $\cdot$ is either P or N. R^1 is a combination of the two cases (P,P,P) and (N,P,P), and R^6 is a combination of the two cases (P,N,N) and (N,N,N).

Now we derive the amplitudes of the control outputs. Note the skew symmetry between scenarios (1) and (6), and hence rules 1 and 6, whereby the input variables have opposite fuzzy labels P and N. We have output skew symmetry whereby the two rules give the same control amplitude n_1 but in opposite directions. Similar skew symmetry can be observed between rules 2 and 5, and between rules 3 and 4. Thus, only three output parameters $n_i > 0$ $(i = 1, 2, 3)$ need to be determined. In scenario (1), the pendulum tilt angle and angular velocity are in the same direction ($\theta > 0, \dot{\theta} > 0$), and thus the pendulum tends to fall down clockwise. In scenarios (2) and (3), the pendulum tilt angle and angular velocity are in opposite directions ($\theta > 0, \dot{\theta} < 0$), and thus the pendulum returned to the balance position.

Table 1
Fuzzy rules for the speed controller in the regulation task.

Rule	$\dot{x}$	θ	$\dot{\theta}$	u
1	×	P	P	$\tau_1 = +n_1$
2	P	P	N	$\tau_2 = +n_2$
3	N	P	N	$\tau_3 = +n_3$
4	P	N	P	$\tau_4 = -n_3$
5	N	N	P	$\tau_5 = -n_2$
6	×	N	N	$\tau_6 = -n_1$

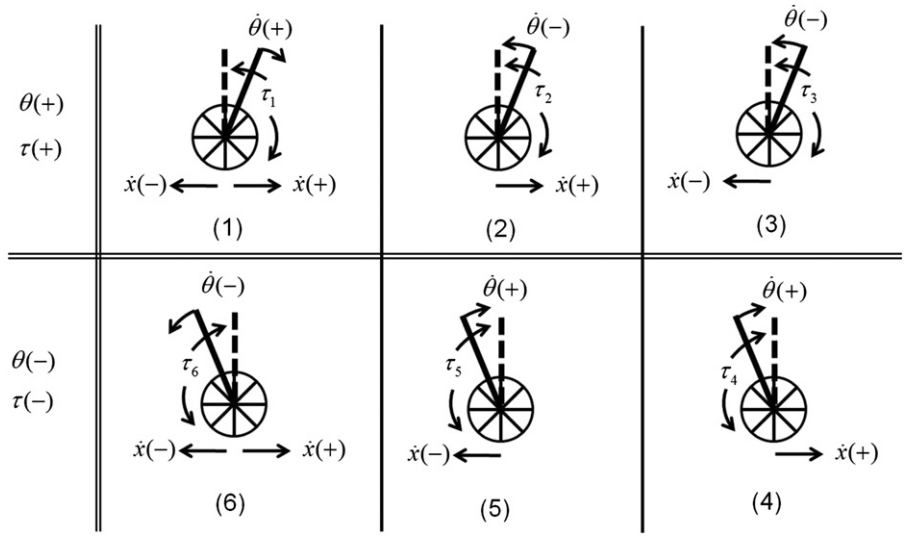

Fig. 4. Graphical representation of the fuzzy rules corresponding to six scenarios. Each scenario is associated with a rule, namely, scenario (i) is associated with rule i for $i = 1, \ldots, 6$. The control priority is to balance the pendulum. When θ is P, as shown in scenarios (1)–(3), a positive torque is provided so that the wheel moves rightwards and the pendulum moves anticlockwise, regardless of the values of $\dot{x}$ and $\dot{\theta}$. Likewise, when θ is N, as shown in scenarios (4)–(6), a negative torque is provided so that the wheel moves leftwards and the pendulum moves clockwise. In this way, rules 1–3 have positive outputs and rules 4–6 have negative outputs.

Intuitively, a larger torque should be applied in scenario (1) to bring the pendulum back to the balance position, i.e., we should choose $n_1 > n_i$ $(i = 2, 3)$.

Next we need to decide the relative amplitudes of n_2 and n_3. The only difference between scenarios (2) and (3) is the direction of the wheel velocity, and thus deciding the relative amplitudes of n_2 and n_3 involves control of the wheel. However, the difference in velocity is not adequate for deciding the relative amplitudes of n_2 and n_3. From Newton's mechanical law, pendulum motion is related to wheel acceleration instead of velocity, as we can observe from the unicycle dynamic equations (1) and (2). In many practical control tasks, acceleration is not available. For mechanical systems such as a unicycle, the full state feedback uses only position and velocity information.

Since n_2 and n_3 cannot be decided using intuitive derivation or heuristic knowledge, trial and error is an alternative. Through numerical tests, we find that $n_2 < n_3$ yields unstable responses, and $n_2 > n_3$ leads to stable behavior. Nevertheless, in this study we seek a systematic way to determine the FLC parameters, in particular the relative amplitudes of n_2 and n_3.

FLC can be regarded as a state feedback controller with varying feedback gains. In fact, from Figs. 3 and 4, the control output can be expressed as $u = -\mathbf{k}^T(\mathbf{x})\mathbf{x}$. To stabilize the unicycle system, feedback control should be taken appropriately for all states. A linear controller helps to reveal how a feedback controller works. Based on the linearized model in (B.1), we design a linear controller as

$$u = -\mathbf{k}^T\mathbf{x} = -k_1x_1 - k_2x_2 - k_3x_3, \tag{5}$$

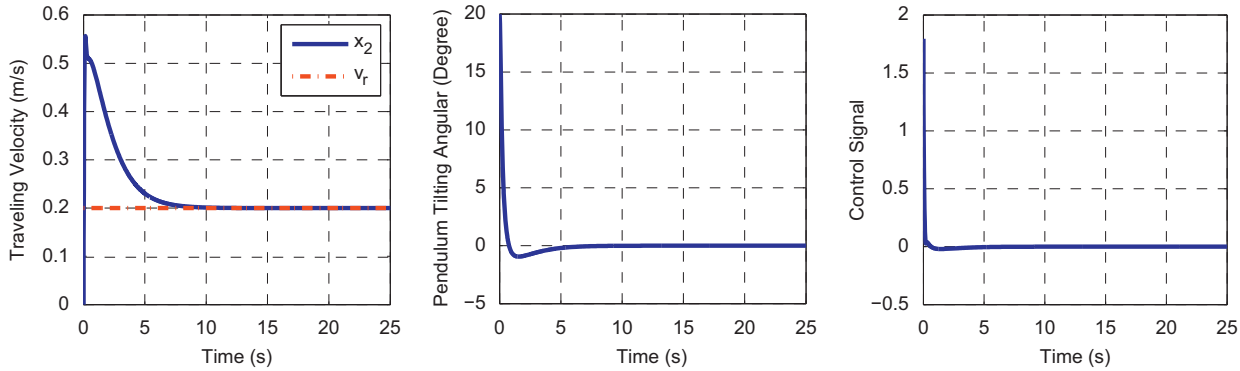

Fig. 5. Time responses of the wheel velocity, the pendulum tilt angle and the control signal profile under setpoint control with $v_r = 0.2\,\mathrm{m/s}$. The FLC consists of six rules.

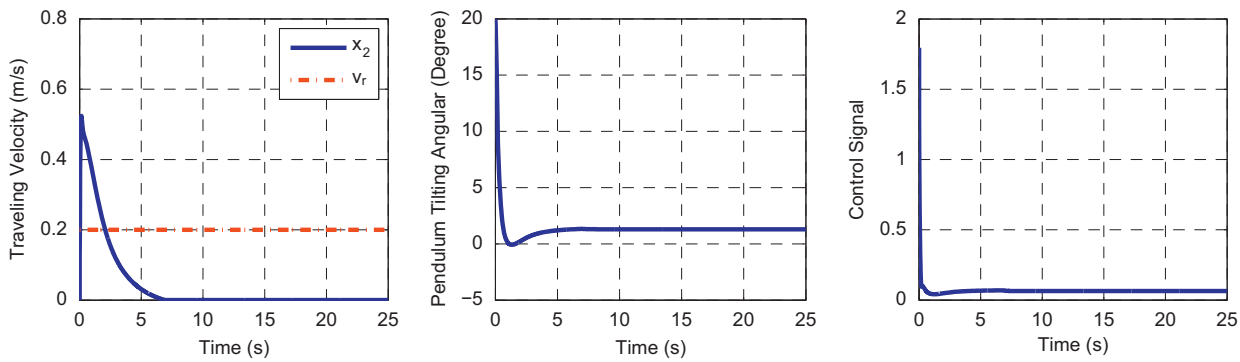

Fig. 6. Time responses of the wheel velocity, the pendulum tilt angle and the control signal profile under setpoint control $v_r = 0.2\,\mathrm{m/s}$ in the presence of unknown friction and slope. The FLC consists of six rules.

where $\mathbf{k} = [k_1, k_2, k_3]^T$ is the feedback gain vector. To ensure local stability of the desired equilibrium point, all feedback gains need to be negative, i.e., $k_1 < 0$, $k_2 < 0$, $k_3 < 0$ (Appendix B).

Remark 1. Conditions $k_i < 0$ $(i = 1, 2, 3)$ are established without knowing the unicycle parameter values.

Note that the feedback term associated with x_1 is $-k_1x_1$. In scenario (2), where x_1 is P, $-k_1x_1 > 0$. On the contrary, in scenario (3), where x_1 is N, $-k_1x_1 < 0$. Thus, when states (x_2, x_3) in scenario (2) are of the same value as in case (3), the control output in scenario (2) should be greater than that in case (3). Therefore, we have $n_2 > n_3$, which is consistent with the numerical tests.

For simulation, the unicycle parameter values are $m_w = 0.575\,\mathrm{kg}$, $m_p = 1.45\,\mathrm{kg}$, $I_w = 2.3 \times 10^{-3}\,\mathrm{kg\,m^2}$, $I_p = 5.8 \times 10^{-2}\,\mathrm{kg\,m^2}$, r=0.09 m and l=0.2 m. The FLC with six rules is directly applicable to setpoint control by replacing x_1 with e_1 as the first input variable. To verify the effectiveness of the proposed FLC, choose $[n_1, n_2, n_3] = [5, 2, 1.5]$. The desired velocity for the wheel is $v_r = 0.2\,\mathrm{m/s}$. The results are shown in Fig. 5. Setpoint control of the wheel velocity is achieved while the pendulum is balanced.

In practice, disturbances and model uncertainties exist in the unicycle system, such as friction f_r and slope φ. Assume that the unicycle travels on a tilted surface with $\varphi = 2^\circ$ and the friction is $f_r = 0.2\dot{x} + 0.1\,\mathrm{sgn}(\dot{x})$. The results are shown in Fig. 6. The unicycle fails to follow the desired velocity of 0.2 m/s; in other words, the FL speed controller is not robust for exogenous disturbances and model uncertainties.

Table 2
Fuzzy rules for a speed controller with integral action for the setpoint task.

Rule	E_I	$e_1(\dot{x}-\dot{x}_r)$	$e_2(\theta-0)$	$e_3(\dot{\theta}-0)$	u
1	×	×	P	P	$\tau_1 = +n_1$
2	P	P	P	N	$\tau_2 = +n_2$
3	N	P	P	N	$\tau_3 = +n_3$
4	P	N	P	N	$\tau_4 = +n_4$
5	N	N	P	N	$\tau_5 = +n_5$
6	P	P	N	P	$\tau_6 = -n_5$
7	N	P	N	P	$\tau_7 = -n_4$
8	P	N	N	P	$\tau_8 = -n_3$
9	N	N	N	P	$\tau_9 = -n_2$
10	×	×	N	N	$\tau_{10} = -n_1$

3.3. Fuzzy logic speed controller with integral action

To enhance the FLC robustness, we introduce another input, $E_I = \int_0^t (\dot{x} - v_r)\, d\tau$, which is the integration of the wheel velocity error. Input E_I is also associated with two fuzzy sets P and N, and the degree to which set the values belong to is determined by the membership functions illustrated in Fig. 2. We assume that the average value of the velocity tracking error e_1 for the whole test time, i.e., $1/t_s \cdot \int_0^{t_s} (\dot{x} - v_r)\, d\tau$, should be approximately 0.1 m/s. Thus, the range for E_I is chosen to be approximately $m_1 = 0.1 t_s$ m. The ranges of other input variables $(e_1, \theta, \dot{\theta})$ are $[m_2, m_3, m_4] = [0.5, \pi/3, 2]$, which are selected the same as in Section 3.2.

For a conventional FLC design, we could have $2^4 = 16$ fuzzy rules. Similarly, the number of fuzzy rules is minimized to simplify the FLC design. Fuzzy rules are shown in Table 2. R^1 is a combination of four cases, (P,P,P,P), (P,N,P,P), (N,P,P,P) and (N,N,P,P), and R^{10} is a combination of four cases, (P,P,N,N), (P,N,N,N), (N,P,N,N) and (N,N,N,N). Here R^1 and R^{10} describe the same cases as R^1 and R^6 do in Table 1. R^2 and R^3 in Table 2 are obtained from R^2 in Table 1 with additional P and N for E_I. Similarly, we establish rules R^4–R^9.

There are five output parameters, n_1–n_5, to be determined. First, considering that θ and $\dot{\theta}$ are in the same direction in R^1 and R^{10}, based on the analysis in Section 3.2, we have $n_1 > n_i > 0$ $(i = 2, 3, 4, 5)$. Next, we use feedback knowledge from a linear controller to determine the relative amplitude of n_2–n_5 because intuitive derivation is not straightforward.

With the additional state E_I, we have an augmented linearized state space model as given in (C.1). Now the linear feedback control law is $u = -\mathbf{k}^T \mathbf{e}$ with $\mathbf{k} = [k_1, k_2, k_3, k_4]^T$. To ensure local stability of the desired equilibrium point, all the feedback gains need to be negative (Appendix C).

The relative amplitudes of the output parameters are determined by adopting the method used in Section 3.2. Between R^2 and R^3 or R^4 and R^5, the only difference in fuzzy inputs is P and N for E_I. Thus, the control output for rules with positive E_I should be greater than that for rules with negative E_I, that is, $n_2 > n_3, n_4 > n_5$. Between R^2 and R^4 or R^3 and R^5, the only difference in fuzzy inputs is P and N for e_1. Thus the control output for rules with positive e_1 should be greater than that for rules with negative e_1, that is, $n_2 > n_4$ and $n_3 > n_5$. To simplify the design, we could choose $n_3 = n_4$. To summarize, we have $n_2 > n_3 = n_4 > n_5$.

Calculation of the firing strength for each rule is shown in Fig. 7. The final output of the fuzzy controller is computed as (4) with $n = 10$.

FLC with 10 rules is applied with $[n_1, n_2, n_3, n_4, n_5] = [8, 4.5, 1.2, 1.2, 0.6]$, and parameter ranges $[m_1, m_2, m_3, m_4] = [5, 0.5, \pi/3, 2]$. The friction and slope are presented as in the preceding example. The simulation results are shown in Fig. 8. The wheel velocity reaches the desired value of 0.2 m/s while the pendulum is balanced at a new equilibrium point, $\theta = 1.5011^\circ$. The results can be explained as follows.

When system is in a steady state, we have $\ddot{x}, \ddot{\theta}, \dot{\theta} = 0$. From the unicycle dynamic equation (1), $\tau = \sin\varphi(m_w + m_p)gr + rf_r$ is obtained. From (2), we obtain $\theta = \arcsin(\tau/m_p lg)$. It is clear that to maintain a constant velocity, a constant torque is needed to overcome the effects of friction and slope, which thus results in a new balance position for the pendulum.

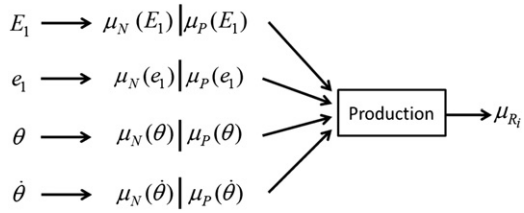

Fig. 7. TS-type fuzzy inference for the ith rule of the FLC with integral action.

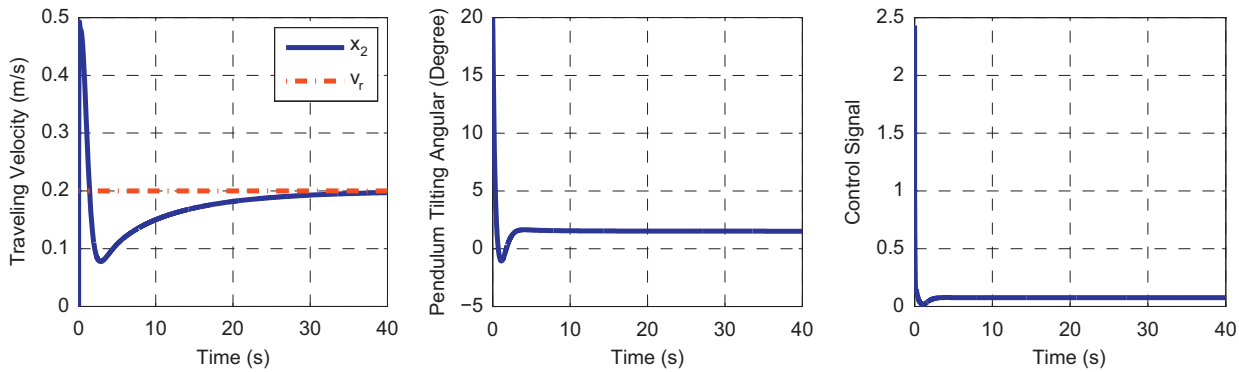

Fig. 8. Time responses of the wheel velocity, the pendulum tilt angle and the control signal profile under setpoint control with $v_r = 0.2\,\mathrm{m/s}$ in the presence of unknown friction and slope. The FLC consists of 10 rules.

4. FLC with iterative tuning

The relative amplitudes of n_i are given in the previous section. Parameter tuning now needs to be addressed, but manual tuning is a tedious and time-consuming process. Tuning becomes even more challenging when model mismatch exists, for instance when the pendulum mass m_p and slope φ are unknown and if the actuator dynamics are not modeled. We use ILT to tune the output parameters. This problem is formulated as a minimization process with respect to a selected cost function. Parameter tuning is carried out via an updating law that is derived using gradient information to minimize the cost function iteratively. The result after tuning is an optimal solution with respect to the formulated cost function. With the ILT, the FLC design allows unstable system responses in the first few trials, so the initial values for the output parameters can be freely assigned. Given that the system response is unstable with the initial parameters, after the first IL tuning stage, a stable response can be reached. After further fine tuning, stable and fairly good performance can be achieved.

4.1. Selection of the cost function

Let t_s denote the time duration for an evaluation period. We consider the overall performance in the interval $[t_0, t_s]$. Different indices and features of unicycle behavior can be used to evaluate the control performance, which can be either a stable or an unstable response. From a practical point of view, if $|\theta| > \pi/2$, we consider that the controller failed. We define t_w as the total time for which the pendulum stays above the horizontal plane, i.e., $|\theta| < \pi/2$ for $t \in [0, t_w]$.

For an unstable response ($t_w < t_s$), the following index can be used to evaluate the performance:

$$G_1 = \frac{10}{t_w + 0.01}.$$

A smaller G_1 implies a larger t_w and hence a more stable response.

For a stable response ($t_w = t_s$), the following indexes can be chosen to evaluate the unicycle performance. To keep the pendulum around the balance position, we can use

$$\int_0^{t_s} |\theta|^q \, dt \quad \text{or} \quad \max(|\theta|),$$

where q is a positive number. To avoid sharp changes in the pendulum angle and reduce pendulum oscillation, we can use

$$\int_0^{t_s} |\dot{\theta}|^q \, dt \quad \text{or} \quad \max(|\dot{\theta}|).$$

To achieve a velocity tracking task, we may consider

$$\int_0^{t_s} |e_1|^q \, dt \quad \text{or} \quad \max(|e_1|).$$

Since saturation problems commonly exist in real situations, to avoid a large control signal we can use

$$\int_0^{t_s} |u|^q \, dt \quad \text{or} \quad \sum n_i \ (i = 1, \ldots, 5).$$

When multiple performance indexes are taken into consideration, controller design and tuning become a multi-objective optimization issue, which can be reduced to a scalar case using weighted sums. For instance, a cost function as the weighted sum of all the indices is

$$G_2 = \mathbf{w}^T \mathbf{f}, \tag{6}$$

where $\mathbf{w}$ is a vector of weighting coefficients and $\mathbf{f}$ is a vector of selected indices.

Remark 2. We can take various costs, not limited to those mentioned above, into consideration, such as the settling time, model nonlinearities and the linear approximation error. To meet different control requirements, we need to choose different performance indices and different weighting values.

Remark 3. The cost function is selected based on human experience, which captures the key features of the unicycle response.

For fuzzy logic speed controller with integral action, there are five output parameters, n_i $(i = 1, 2, 3, 4, 5)$, to be determined. Note that $n_1 > n_2 > n_3 = n_4 > n_5$. We define $p_1 = n_1 - n_2$, $p_2 = n_2 - n_3$ and $p_3 = n_4 - n_5$, and denote $\mathbf{p} = [p_1, p_2, p_3]^T$, where $p_1, p_2, p_3 > 0$. n_5 is fixed at a given value, although it can also be learned. $\mathbf{p}$ is tuned to achieve satisfactory performance.

The tuning procedure for $\mathbf{p}$ can be divided into two parts. For a given initial value of $\mathbf{p}$, denoted as $\mathbf{p}_0$, if the first trial fails (i.e., unstable position), the cost function J is chosen to be G_1. The objective is to increase the time t_w for which a stable response lasts, and finally reach $J_{min} = 10/(t_s + 0.01)$, whereby the updated $\mathbf{p}$ makes the pendulum stay above the horizontal plane during the whole evaluation period. Then we switch to cost function $J = G_2$ for fine tuning. Note that now the optimal design is to minimize the cost function J by updating $\mathbf{p}$, which is in fact a search task:

$$\min_{\mathbf{p} \in \Omega_{\mathbf{p}}} J(\mathbf{e}, \mathbf{p}),$$

where $\Omega_{\mathbf{p}} = \{\mathbf{p} \in \mathbb{R}^3 | p_1 > 0, p_2 > 0, p_3 > 0\}$.

We propose the following typical first-order ILT law:

$$\mathbf{p}_{i+1} = \mathbf{p}_i - \gamma_i J_i,$$

where the subscript i denotes the ith updating, $\mathbf{p} = [p_{1,i}, p_{2,i}, p_{3,i}]^T$. $\gamma_i = [\gamma_{1,i}, \gamma_{2,i}, \gamma_{3,i}]^T$ is a learning gain vector that should be chosen to ensure the convergence of J_i. To speed up the learning process, the learning gain is chosen to be the inverse of the gradient $\partial J / \partial p$. When the gradient is not available analytically, a numerically computed gradient can be used [21]. Note that

$$J(\mathbf{p}_{i+1}) = J(\mathbf{p}_i) + [J(\mathbf{p}_{i+1}) - J(\mathbf{p}_i)] = J(\mathbf{p}_i) + \frac{dJ(\mathbf{p}^*)}{d\mathbf{p}}(\mathbf{p}_{i+1} - \mathbf{p}_i),$$

where $\mathbf{p}^*$ is a value between $\mathbf{p}_i$ and $\mathbf{p}_{i+1}$.

Substituting the ILT law, we have

$$J(\mathbf{p}_{i+1}) = \left[1 - \frac{dJ(\mathbf{p}_i^*)}{d\mathbf{p}}\gamma_i\right] J(\mathbf{p}_i). \tag{7}$$

To guarantee a contractive mapping in (7), the following condition should be satisfied:

$$\left|1 - \frac{dJ(\mathbf{p}_i^*)}{d\mathbf{p}}\gamma_i\right| < 1.$$

We define $D_{j,i} = dJ(\mathbf{p}_i^*)/dp_{j,i}$ $(j = 1, 2, 3)$, which is estimated numerically as

$$\hat{D}_{j,i} = \frac{J(\mathbf{p}_i) - J(\mathbf{p}_{i-1})}{p_{j,i} - p_{j,i-1}}. \tag{8}$$

Then the learning gain is $\gamma_{j,i} = \lambda_j/\hat{D}_{j,i}$, where λ_j is a constant gain and $0 < \lambda_j \leq 1$.

Remark 4. For the first iteration, the gradient information is unavailable. We can choose a sufficiently small learning gain. An alternative is to update each element of $\mathbf{p}$ in opposite directions (increasing and decreasing directions), which yields eight ($2^3 = 8$) directions in 3D space, and select the best result for the next updating.

Remark 5. For the learning gain selection, theoretically, the objective function J converges faster as λ_i increases. However, in this study we use a numerical method to estimate the gradient information as in (8); the information might not be accurate if there is a large difference in response between consecutive iterations due to parameter updating. A small λ_i allows the parameters to be updated slowly and provides a better learning result. Conversely, the number of learning iterations increases as λ_i decreases.

4.2. Learning results

A setpoint control task is considered with $v_r = 0.2$ m/s. The initial states are $(\dot{x}, \theta, \dot{\theta}) = (0, 20^\circ, 0)$. The pendulum load is $m_p = 8$ kg, the slope is $\varphi = 30^\circ$ and friction is $f_r = 0.2\dot{x} + 0.1\,\mathrm{sgn}(\dot{x})$. The total time for evaluation is 40 s.

FLC with integral action is applied with the same parameter ranges and output parameters as in the preceding example, which yields $\mathbf{p} = [3.5, 3.3, 0.6]^T$. However, the pendulum falls down at $t = 0.77$ s. We conclude that the FLC with parameters that stabilize the unicycle system for $m_p = 1.45$ kg and $\varphi = 2^\circ$ in the preceding example does not work when $m_p = 8$ kg and $\varphi = 30^\circ$.

The ILT method is applied with the cost function $J = G_1$. The learning gains are $[\lambda_1, \lambda_2, \lambda_3] = [0.5, 0.3, 0.3]$. The learning process is iterated until $t_w = t_s$. We obtain $\mathbf{p} = [4.7379, 4.4608, 0.8111]^T$ after IL tuning. The results are shown in Figs. 9 and 10. The pendulum stays above the horizontal plane during the whole evaluation period. However, the overshoot of the wheel velocity is large and a steady-state error for the wheel velocity exists. The maximum pendulum angle is 27.1°. Thus, the system performance is still poor and further fine tuning is required.

Next, ILT is applied with cost function $J = G_2$ and the following cost function is used:

$$G_2 = \int_0^{t_s} [w_1|\theta(t)| + w_2|\dot{x}(t) - v_r(t)| + w_3|u(t)|]\,dt. \tag{9}$$

A smaller G_2 implies a smaller deviation from the equilibrium or setpoint. The weighting factors in the cost function chosen are $w_1 = 750$, $w_2 = 750$ and $w_3 = 600$. The initial value of $\mathbf{p}$ is $[4.7379, 4.4608, 0.8111]^T$, which is the result after four learning iterations with G_1. Learning gains are chosen as $[\lambda_1, \lambda_2, \lambda_3] = [0.3, 0.3, 0.3]$. The learning process is iterated until J no longer decreases. We obtain $\mathbf{p} = [7.6416, 7.1947, 1.3081]^T$ after ILT. The results are shown in Figs. 11 and 12. It is clear that the unicycle responses improve after fine tuning. Compared with the results in Fig. 9, the wheel can perfectly track the desired velocity with no tracking error and the pendulum remains in a smaller range around the balance position.

For comparison, ILT is applied with an alternative cost function for fine tuning:

$$G_2 = w_1 \max(|\theta|) + w_2 \max(|e_1|) + w_3(p_1 + p_2 + p_3). \tag{10}$$

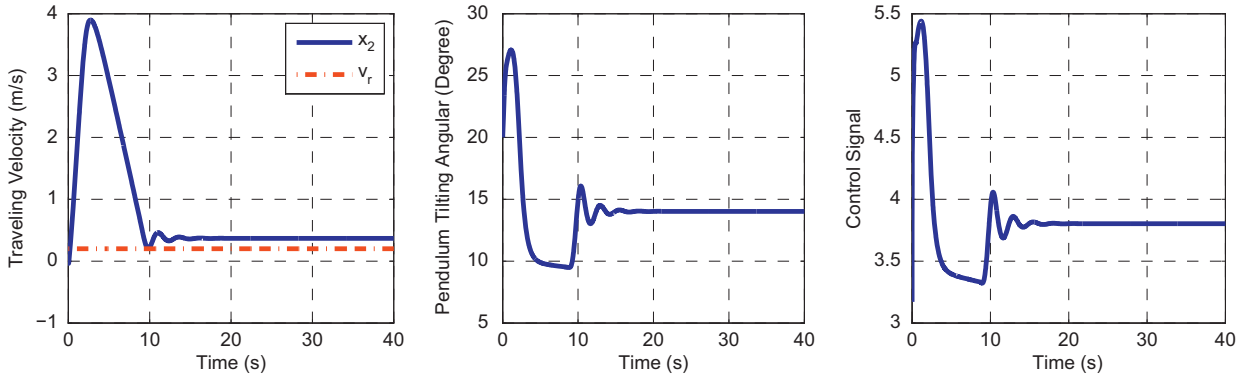

Fig. 9. Time responses of the wheel velocity, the pendulum tilt angle and the control signal profile after four learning iterations with the cost function G_1. Before learning, $[n_1, n_2, n_3, n_4, n_5] = [8, 4.5, 1.2, 1.2, 0.6]$; after learning, $[n_1, n_2, n_3, n_4, n_5] = [10.6098, 5.8719, 1.4111, 1.4111, 0.6]$.

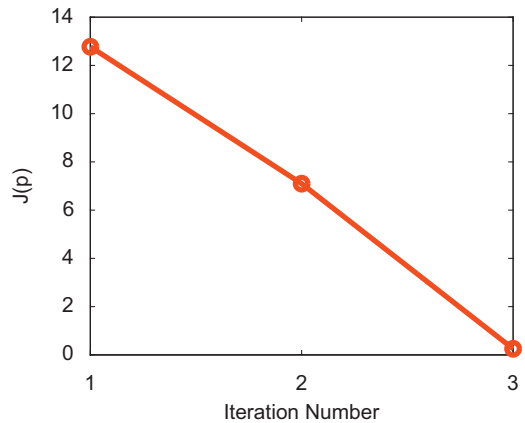

Fig. 10. Evolution of the cost function G_1.

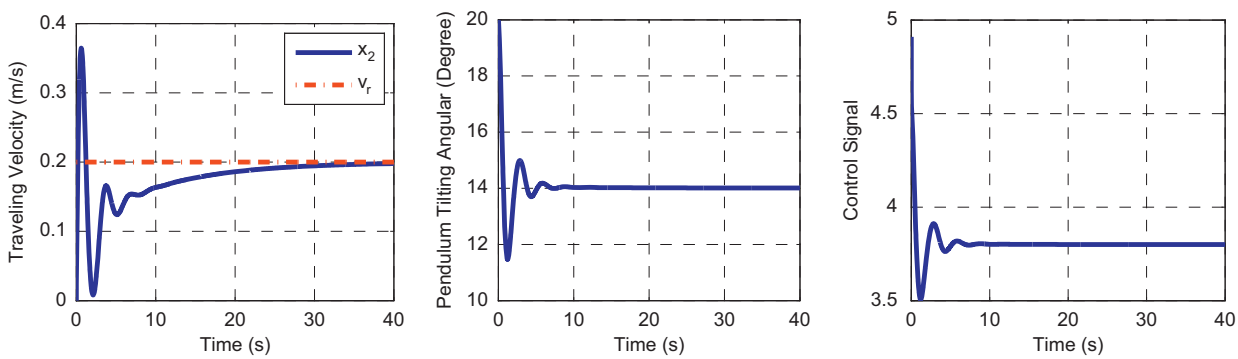

Fig. 11. Time responses of the wheel velocity, the pendulum tilt angle and the control signal profile after fine-tuning of three iterations with the cost function (9). Before learning, $[n_1, n_2, n_3, n_4, n_5] = [10.6098, 5.8719, 1.4111, 1.4111, 0.6]$; after learning, $[n_1, n_2, n_3, n_4, n_5] = [16.7444, 9.1028, 1.9081, 1.9081, 0.6]$.

The weighting factors in the cost function were $w_1 = 100$, $w_2 = 800$ and $w_3 = 20$. The other parameters and initial conditions were the same as in the preceding example. The learning process was iterated until J no longer decreased. We obtained $\mathbf{p} = [6.9954, 6.5863, 1.1975]^T$ after learning tuning over iterations. The results are shown in Figs. 13 and 14. Compared with the preceding simulation results, it is evident that ILT with different cost functions leads to quite similar results, which indicates the flexibility in choosing cost functions.

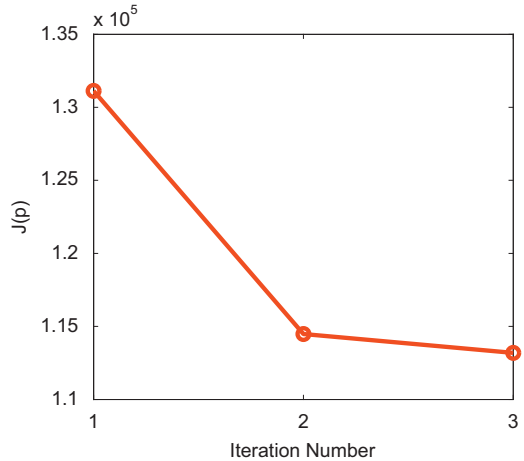

Fig. 12. Evolution of the cost function G_2 given in (9).

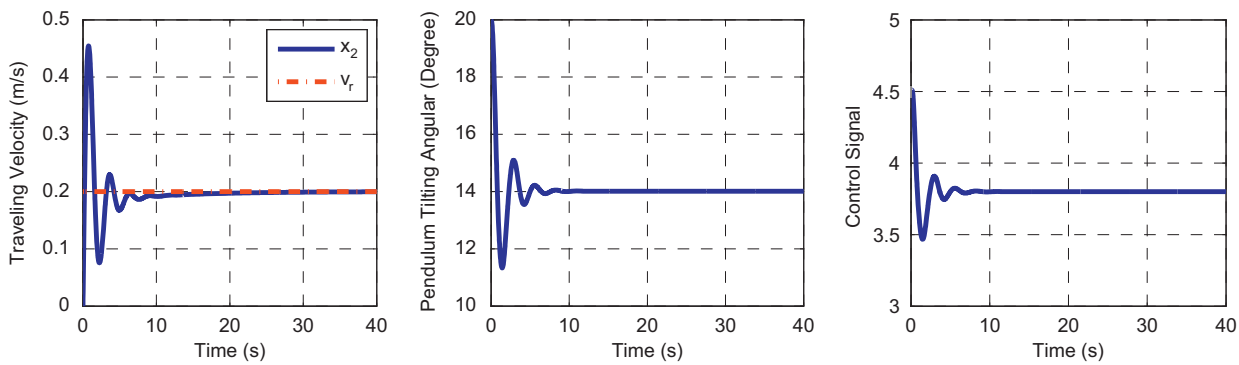

Fig. 13. Time responses of the wheel velocity, the pendulum tilt angle and the control signal profile after fine-tuning of five iterations with the cost function (10). Before learning, $[n_1, n_2, n_3, n_4, n_5] = [10.6098, 5.8719, 1.4111, 1.4111, 0.6]$; after learning, $[n_1, n_2, n_3, n_4, n_5] = [15.3792, 8.3838, 1.7975, 1.7975, 0.6]$.

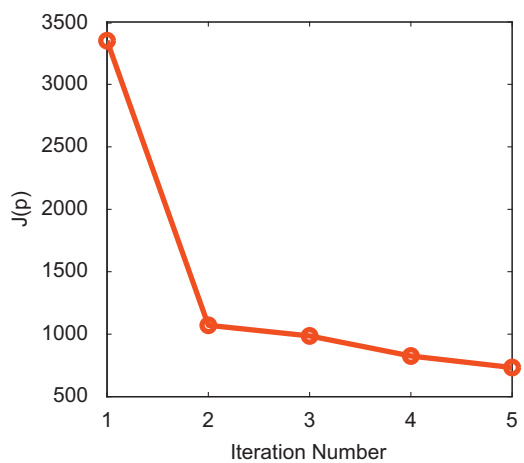

Fig. 14. Evolution of the cost function G_2.

5. Conclusion

In this work, synthesis of a design for TS FLC for an underactuated unicycle was described. The FLC design is based on both human experience and information from the dynamic model of the system. The effectiveness of the FLC and ILT was verified using simulations. The proposed FLC is simple and easy to apply. The ultimate objective of the design is to maximize the utilization of system information from either human experience or an analytical model. As a result, the design is easily understood and offers great flexibility. Our next aim is to address implementation on a real-time platform.

Appendix A. Derivation of the dynamic equations

The mathematical model of the unicycle system shown in Fig. 1 is derived using a Euler–Lagrange formulation. We first present the kinetic and potential energies used to compute the Lagrangian function [22]. The potential energy of the wheel is

$$V_{wheel} = m_w g x \sin\varphi$$

and the kinetic energy of the wheel is

$$T_{wheel} = \tfrac{1}{2} m_w \dot{x}^2 + \tfrac{1}{2} I_w \dot{\phi}^2 = (\tfrac{1}{2} m_w r^2 + \tfrac{1}{2} I_w)\dot{\phi}^2.$$

We define x_p as the horizontal position and y_p as the vertical position of the centroid of the pendulum. We then have

$$x_p = x\cos\varphi + l\sin\theta, \quad y_p = x\sin\varphi + l\cos\theta.$$

The potential energy for the pendulum is

$$V_{pendulum} = m_p g y_p = m_p g(x\sin\varphi + l\cos\theta)$$

and the kinetic energy is

$$T_{pendulum} = \tfrac{1}{2} m_p(\dot{x}_p^2 + \dot{y}_p^2) + \tfrac{1}{2} I_p \dot{\theta}^2 = \tfrac{1}{2} m_p[\dot{x}^2 + l^2\dot{\theta}^2 + 2\dot{x}\dot{\theta} l\cos(\theta+\varphi)] + \tfrac{1}{2} I_p \dot{\theta}^2.$$

Therefore, the Lagrangian function of the unicycle is given by

$$\mathcal{L} = T_{wheel} + T_{pendulum} - V_{wheel} - V_{pendulum} = (\tfrac{1}{2} m_w r^2 + \tfrac{1}{2} I_w)\dot{\phi}^2 + \tfrac{1}{2} m_p[\dot{x}^2 + l^2\dot{\theta}^2 + 2\dot{x}\dot{\theta} l\cos(\theta+\varphi)] + \tfrac{1}{2} I_p \dot{\theta}^2 - m_w g x\sin\varphi - m_p g(x\sin\varphi + l\cos\theta).$$

The equations of motion for the unicycle are then given by the Euler–Lagrange equations

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\phi}}\right) - \frac{\partial \mathcal{L}}{\partial \phi} = \tau - r f_r, \tag{A.1}$$

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}}\right) - \frac{\partial \mathcal{L}}{\partial \theta} = -\tau. \tag{A.2}$$

The terms on the right-hand side of the equations represent torques applied externally to the system. For torques acting on the wheel on the right-hand side of (A.1), τ is the torque generated by the driving motor and rf_r is the torque due to friction f_r. The torque acting on the pendulum shown in (A.2) is due to the reaction torque of the driving motor.

In Eq. (A.1),

$$\frac{\partial \mathcal{L}}{\partial \phi} = 0$$

and

$$\frac{\partial \mathcal{L}}{\partial \dot{\phi}} = \frac{\partial}{\partial \dot{\phi}}\left[\left(\frac{1}{2} m_w r^2 + \frac{1}{2} I_w\right)\dot{\phi}^2 + m_p \dot{x}\dot{\theta} l\cos(\theta+\varphi) + \frac{1}{2} m_p \dot{x}^2\right] = (m_w r^2 + I_w + m_p r^2)\dot{\phi} + m_p r\dot{\theta}\cos(\theta+\varphi)$$

and thus

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\phi}}\right) - \frac{\partial \mathcal{L}}{\partial \phi} = (m_w r^2 + m_p r^2 + I_w)\ddot{\phi} - m_p r l \sin(\theta+\varphi)\dot{\theta}^2 + m_p r l \cos(\theta+\varphi)\ddot{\theta}.$$

Recall that $x = r\phi$, so we have

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\phi}}\right) - \frac{\partial \mathcal{L}}{\partial \phi} = \left(m_w r + m_p r + \frac{I_w}{r}\right)\ddot{x} - m_p r l \sin(\theta+\varphi)\dot{\theta}^2 + m_p r l \cos(\theta+\varphi)\ddot{\theta}.$$

Substituting the above result in (A.1) and dividing both sides of the equation by r, we have

$$a\ddot{x} + b\ddot{\theta} - m_p l \sin(\theta+\varphi)\dot{\theta}^2 + \sin\varphi(m_p + m_w)g = \frac{1}{r}(\tau - r f_r), \tag{A.3}$$

where $a = m_w + m_p + I_w/r^2$ and $b = m_p l\cos(\theta+\varphi)$.

In Eq. (A.2),

$$\frac{\partial \mathcal{L}}{\partial \theta} = m_p l g \sin\theta - m_p l \dot{x}\dot{\theta}\sin(\theta+\varphi)$$

and

$$\frac{\partial \mathcal{L}}{\partial \dot{\theta}} = \frac{\partial}{\partial \dot{\theta}}\left\{\frac{1}{2}m_p[l^2\dot{\theta}^2 + 2\dot{x}\dot{\theta}l\cos(\theta+\varphi)] + \frac{1}{2}I_p\dot{\theta}^2\right\} = (I_p + m_p l^2)\dot{\theta} + m_p l\dot{x}\cos(\theta+\varphi),$$

and thus

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}}\right) - \frac{\partial \mathcal{L}}{\partial \theta} = (I_p + m_p l^2)\ddot{\theta} + m_p l\cos(\theta+\varphi)\ddot{x} - m_p l g\sin\theta.$$

Substituting the above result in (A.2), we have

$$b\ddot{x} + c\ddot{\theta} - m_p l g \sin\theta = -\tau, \tag{A.4}$$

where $c = I_p + m_p l^2$.

The dynamic behavior of the unicycle system described by (A.3) and (A.4) is given by (1) and (2).

Appendix B. Analysis of feedback control for stabilization of the linearized unicycle model

Linearizing the nonlinear unicycle model (1)–(2) around the equilibrium $(\theta, \dot{\theta}) = (0, 0)$ by assuming $\sin\theta \approx \theta$, $\cos\theta \approx 1$ and $\dot{\theta}^2 \approx 0$, the resulting linear error dynamic is

$$\dot{\mathbf{e}} = A\mathbf{e} + \mathbf{b}u, \tag{B.1}$$

where

$$A = \begin{bmatrix} 0 & a_{23} & 0 \\ 0 & 0 & 1 \\ 0 & a_{43} & 0 \end{bmatrix}, \quad \mathbf{b} = \begin{bmatrix} b_1 \\ 0 \\ b_2 \end{bmatrix}, \tag{B.2}$$

with

$$a_{23} = -\frac{b_0 m_p l g}{ac - {b_0}^2}, \quad a_{43} = \frac{a m_p l g}{ac - {b_0}^2},$$

$$b_1 = \frac{1}{r}\frac{c}{ac - {b_0}^2} + \frac{b_0}{ac - {b_0}^2}, \quad b_2 = \frac{1}{r}\frac{-b_0}{ac - {b_0}^2} + \frac{-a}{ac - {b_0}^2},$$

and $b_0 = m_p l\cos\varphi$ is obtained from b with $x_3 = 0$.

The linear feedback control law is $u = -\mathbf{k}^T\mathbf{e}$ with $\mathbf{k} = [k_1, k_2, k_3]^T$. We denote the closed-loop system matrix by A_c. We have

$$A_c = A - \mathbf{b}\mathbf{k}^T = \begin{bmatrix} -b_1k_1 & a_{23} - b_1k_2 & -b_1k_3 \\ 0 & 0 & 1 \\ -b_2k_1 & a_{43} - b_2k_2 & -b_2k_3 \end{bmatrix}.$$

The characteristic equation of A_c is

$$|\lambda I - A_c| = \lambda^3 + (b_1k_1 + b_2k_3)\lambda^2 + (b_2k_2 - a_{43})\lambda + k_1(a_{23}b_2 - a_{43}b_1) = 0.$$

To ensure local stability around the desired equilibrium point, A_c should be a Hurwitz matrix. Applying the Routh criterion, we obtain the following necessary conditions for selection of feedback gains:

$$b_1k_1 + b_2k_3 > 0, \tag{B.3}$$

$$b_2k_2 - a_{43} > 0, \tag{B.4}$$

$$k_1(a_{23}b_2 - a_{43}b_1) > 0. \tag{B.5}$$

Note that $a, b_0, c > 0$ and $ac - b_0^2 = (m_w + I_w/r^2)(I_p + m_pl^2) + I_pm_p + (m_pl\sin\varphi)^2 > 0$, and thus we have $b_1, a_{43} > 0$ and $b_2, a_{23} < 0$. It follows from (B.4) that $k_2 < a_{43}/b_2 < 0$. Since $a_{23}b_2 - a_{43}b_1 = -m_plg/[r(ac - b_0^2)] < 0$, we have $k_1 < 0$ from (B.5) and $k_3 < -k_1b_1/b_2 < 0$ from (B.3). Finally, we conclude that all the feedback gains must be negative.

Appendix C. Analysis of feedback control for stabilization of the augmented linearized model

The augmented linearized state space model with integration is

$$\begin{bmatrix} \dot{E}_I \\ \dot{e}_1 \\ \dot{e}_2 \\ \dot{e}_3 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & a_{23} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & a_{43} & 0 \end{bmatrix} \begin{bmatrix} E_I \\ e_1 \\ e_2 \\ e_3 \end{bmatrix} + \begin{bmatrix} 0 \\ b_1 \\ 0 \\ b_2 \end{bmatrix} u. \tag{C.1}$$

The linear feedback control law is $u = -\mathbf{k}^T\mathbf{e}$ with $\mathbf{k} = [k_1, k_2, k_3, k_4]^T$.
The closed-loop system matrix

$$A_c = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -b_1k_1 & -b_1k_2 & a_{23} - b_1k_3 & -b_1k_4 \\ 0 & 0 & 0 & 1 \\ -b_2k_1 & -b_2k_2 & a_{43} - b_2k_3 & -b_2k_4 \end{bmatrix}$$

should be a Hurwitz matrix. The characteristic equation of A_c is

$$|\lambda I - A_c| = \lambda[\lambda^3 + (b_1k_2 + b_2k_4)\lambda^2 + (b_2k_3 - a_{43})\lambda + k_2(a_{23}b_2 - a_{43}b_1)] + b_1k_2\lambda^2 + k_1(a_{23}b_2 - a_{43}b_1) = 0.$$

To ensure local stability around the desired equilibrium point, we have the following necessary conditions for selection of feedback gains:

$$b_1k_2 + b_2k_4 > 0, \tag{C.2}$$

$$b_2k_3 - a_{43} + b_1k_1 > 0, \tag{C.3}$$

$$k_2(a_{23}b_2 - a_{43}b_1) > 0, \tag{C.4}$$

$$k_1(a_{23}b_2 - a_{43}b_1) > 0. \tag{C.5}$$

Since $a_{23}b_2 - a_{43}b_1 < 0$, we have $k_2 < 0$ from (C.4) and $k_1 < 0$ from (C.5). Thus, we have $k_3 < a_{43} - b_1k_1/b_2 < 0$ from (C.3) and $k_4 < -b_1k_2/b_2 < 0$ from (C.2). Finally, we conclude that all the feedback gains must be negative.

References

[1] S.Y. Seo, S.H. Kim, S.-H. Lee, S.H. Han, H.S. Kim, Simulation of attitude control of a wheeled inverted pendulum, in: International Conference on Control, Automation and Systems, 2007, pp. 2264–2269.

[2] K. Pathak, J. Franch, S.K. Agrawal, Velocity and position control of a wheeled inverted pendulum by partial feedback linearization, IEEE Trans. Robot. 21 (2005) 505–513.

[3] H. Ashrafiuon, R.S. Erwin, Sliding mode control of underactuated multibody systems and its application to shape change control, Int. J. Control 81 (2008) 1849–1858.

[4] M.-S. Park, D. Chwa, Swing-up and stabilization control of inverted-pendulum systems via coupled sliding-mode control method, IEEE Trans. Ind. Electron. 56 (2009) 3541–3555.

[5] J. Huang, Z.-H. Guan, T. Matsuno, T. Fukuda, K. Sekiyama, Sliding-mode velocity control of mobile-wheeled inverted-pendulum systems, IEEE Trans. Robot. 26 (2010) 750–758.

[6] Z. Li, C. Xu, Adaptive fuzzy logic control of dynamic balance and motion for wheeled inverted pendulums, Fuzzy Sets Syst. 160 (2009) 1787–1803.

[7] C.W. Tao, J.S. Taur, C.M. Wang, U.S. Chen, Fuzzy hierarchical swing-up and sliding position controller for the inverted pendulum-cart system, Fuzzy Sets Syst. 159 (2008) 2763–2784.

[8] D. Chwa, Nonlinear tracking control of 3-D overhead cranes against the initial swing angle and the variation of payload weight, IEEE Trans. Control Syst. Technol. 17 (2009) 876–883.

[9] L. Freidovich, A. Shiriaev, F. Gordillo, F. Gómez-Estern, J. Aracil, Partial-energy-shaping control for orbital stabilization of high-frequency oscillations of the Furuta pendulum, IEEE Trans. Control Syst. Technol. 17 (2009) 853–858.

[10] R. Olfati-Saber, Nonlinear Control of Underactuated Mechanical Systems with Application to Robotics and Aerospace Vehicles, Ph.D. Thesis, MIT, 2001.

[11] F. Grasser, A. Arrigo, S. Colombi, A.C. Rufer, JOE: a mobile inverted pendulum, IEEE Trans. Ind. Electron. 49 (2002) 107–114.

[12] X. Ruan, J. Chen, J. Cai, L. Dai, Balancing control of two-wheeled upstanding robot using adaptive fuzzy control method, in: IEEE Conference on Intelligent Computing and Intelligent Systems, vol. 2, 2009, pp. 95–98.

[13] Y. Xu, S.K.-W. Au, Stabilization and path following of a single wheel robot, IEEE/ASME Trans. Mechatronics 9 (2004) 407–419.

[14] K. Brkic, Z. Kovacic, Decoupled control and path tracking of a two-wheeled self-balancing mobile robot, in: IEEE International Symposium on Industrial Electronics, 2009, pp. 642–647.

[15] R. Martínez-Soto, O. Castillo, L.T. Aguilar, Optimization of interval type-2 fuzzy logic controllers for a perturbed autonomous wheeled mobile robot using genetic algorithms, Inf. Sci. 179 (2009) 2158–2174.

[16] L. Astudillo, O. Castillo, L.T. Aguilar, R. Martínez-Soto, Hybrid control for an autonomous wheeled mobile robot under perturbed torques, in: Proceedings of the 12th IFSA World Congress, 2007, pp. 594–603.

[17] M. Nie, W.W. Tan, Stable adaptive fuzzy PD plus PI controller for nonlinear uncertain systems, Fuzzy Sets Syst. 179 (2011) 1–18.

[18] V. Derhami, V.J. Majd, M.N. Ahmadabadi, Exploration and exploitation balance management in fuzzy reinforcement learning, Fuzzy Sets Syst. 161 (2010) 578–595.

[19] C.W. Tao, J.S. Taur, Y.C. Chen, Design of a parallel distributed fuzzy LQR controller for the twin rotor multi-input multi-output system, Fuzzy Sets Syst. 161 (2010) 2081–2103.

[20] J. Dong, Y. Wang, G.-H. Yang, H_∞ and mixed H_2/H_∞ control of discrete-time T–S fuzzy systems with local nonlinear models, Fuzzy Sets Syst. 164 (2011) 1–24.

[21] J.-X. Xu, D.Q. Huang, S. Pindi, Optimal tuning of PID parameters using iterative learning approach, J. Control Meas. Syst. Integration 1 (2008) 143–154.

[22] I. Fantoni, R. Lozano, Nonlinear Control for Underactuated Mechanical Systems, Springer, 2002.