IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 25, 2021, accepted January 31, 2021, date of publication February 8, 2021, date of current version February 16, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3057658

Design of a Decoupling Fuzzy Control Scheme for Omnidirectional Inverted Pendulum Real-World Control

CHIH-HUI CHIU[1], YAO-TING HUNG[1], AND YA-FU PENG[2]
[1]Department of Communications, Navigation and Control Engineering, National Taiwan Ocean University, Keelung 20224, Taiwan
[2]Department of Electrical Engineering, Chien-Hsin University, Taoyuan 320, Taiwan

Corresponding author: Chih-Hui Chiu (chchiu@ntou.edu.tw)

ABSTRACT In this study, an omnidirectional inverted pendulum (ODIP) is controlled based on a dual Takagi–Sugeno (TS) fuzzy control scheme. The ODIP is a handmade system. It contains two subsystems. The lower mechanism system includes a brushless rim motor, a system platform, batteries, and an encoder. The upper mechanism system is mainly composed of a circuit system, a motor fixed platform, a motor, and a flywheel. The proposed controller combines two fuzzy control approaches for ODIP system control with disturbances and uncertainties. The core of the ODIP operating system is an embedded controller, which executes real-world control processes. Moreover, to address the coupling problem, the shafts of the two motors are oriented in orthogonal directions. Then, the two fuzzy controllers can be designed independently without coupling. In the proposed controller, the Takagi–Sugeno fuzzy model is adopted for fuzzy modeling of the ODIP. The conception of parallel distributed compensation (PDC) is utilized to develop fuzzy control from TS fuzzy models. The format of linear matrix inequalities (LMIs) can formulate sufficient conditions. The main contributions of this study are (1) the implementation of an ODIP and (2) the application of the proposed dual Takagi–Sugeno (TS) fuzzy control scheme for real-time control of the ODIP. Finally, the efficiency of the proposed control scheme is illustrated by the experimental results presented in this study.

INDEX TERMS Omnidirectional inverted pendulum, TS fuzzy control scheme, parallel distributed compensation, linear matrix inequality, Lyapunov function, embedded controller.

I. INTRODUCTION

Modern control systems are characterized by actions that require high speed and precision. This performance is typically realized by the use of machine-mounted devices with motors. These transmission parts significantly slow the system speed and system response. They also produce chatter and friction. In some practical applications, actuation system simplification is a popular research topic. It can reduce system costs and energy consumption and simplify the mechanical structure design. Systems where the number of actuators is less than their degrees of freedom are referred to as underactuated systems.

The system control of an inverted pendulum has been studied by many researchers. The inverted pendulum is an underactuated system. The purpose of such a study is to maintain pendulum balance when the pendulum is upright. Recently, many papers have discussed the extension application of one-dimensional inverted pendulum control systems. The most challenging task is to control this inverted pendulum system when a cart is no longer on a setting rail.

Currently, intelligent control methodologies (fuzzy control, neural network, or expert systems) [1]–[9] provide a natural framework for the design of online controllers of nonlinear systems. Fuzzy logic has been widely adopted in the adaptive control of nonlinear systems. This theory is typically used to translate and formulate the human experience to suitably control strategies. Recently, a fuzzy control strategy was used in complicated control problems because of its effortless implementation and uncomplicated computation [10], [11]. This makes fuzzy logic suitable for nonlinear system real-time control.

Fuzzy controllers (FCs) are designed based on human experience. In other words, compared with a conventional feedback control method, FCs can be considered control algorithms without system information. In recent years, fuzzy

The associate editor coordinating the review of this manuscript and approving it for publication was Ton Duc Do.

control designs based on the Takagi–Sugeno (TS) fuzzy model have been applied to diverse applications. In [12], the authors proposed a real-time application to a specific two-rigid-link underactuated robot. A ball robot based on the TS fuzzy model was proposed in [13]. In [14], the authors proposed analytical results to control a two-link robot. An adaptive Takagi-Sugeno fuzzy model-based generalized predictive controller was proposed for a pump storage unit in [15]. In [16], the authors presented event-triggered control for Takagi-Sugeno (T-S) fuzzy networked systems with transmission delay. Moreover, an integral state feedback control method based on the TS fuzzy model was proposed for nonlinear and unstable magnetic levitation ball systems in [17]. Apparently, fuzzy control has been widely and successfully applied to many nonlinear systems. Generally, TS fuzzy control techniques have represented an alternative design direction for control problems. The most useful property of TS fuzzy controllers is their ability to approximate arbitrary linear or nonlinear systems. In other words, a nonlinear system can be transformed into a TS fuzzy model, and then, a parallel distributed compensation (PDC) fuzzy controller design is accomplished using linear matrix inequality (LMI) approaches [18].

Generally, TS fuzzy control techniques have represented an alternative design direction for the control problem. The most useful property of the TS fuzzy control method is its ability to approximate arbitrary linear or nonlinear systems. In general, the TS fuzzy modeling scheme divides the nonlinear system dynamics into several linearized subsystems. Parallel distributed compensation (PDC) is a model-based design procedure to partition nonlinear system dynamics into linearized subsystems. The format of linear matrix inequalities (LMIs) is used to formulate a stable feedback gain. The LMI technique offers a way to stably design PDC controllers. In other words, a nonlinear system can be transformed into a TS fuzzy model, and then, the PDC fuzzy controller design is accomplished using linear matrix inequality approaches. They can be solved efficiently by convex programming techniques for LMIs. Recently, the MATLAB LMI Control Toolbox has been used to refine the asymptotically stable feedback gain value. The TS model with PDC has been successfully used in many control applications.

In our previous research, many real-world dynamic control systems were proposed. In [19], a robust fuzzy controller was proposed for a ball inverted pendulum system control problem. A Mamdani-type fuzzy controller and a compensated control technique were combined in the proposed control system. Moreover, the ball inverted pendulum mechanic structure was designed by our research team. In [20], a single-wheel transportation vehicle was designed and implemented. An adaptive Mamdani-like fuzzy controller was adopted for real-time single-wheel transportation vehicle control. In [21], a two-wheel robot system was implemented. To control the two-wheel robot well, a position- and angle-decoupled intelligent backstepping control system was proposed. Other related works proposed by our team include biomimetic flapping-wing robots, wheeled human transportation vehicles, and bicycle robots [22]–[24].

In this study, an ODIP is designed and implemented based on our past experience. The dynamics of an ODIP are similar to those of an inverted pendulum. In fact, an ODIP can be considered as two inverted pendulums in two independent dimensions combined. This system undoubtedly has many nonlinear components. Therefore, the TS fuzzy control method divides the ODIP nonlinear system into several linearized subsystems. Moreover, the PDC fuzzy controller design with the LMI technique is used to ensure that the ODIP system runs stably. In general, the ODIP is constructed from a chassis, a flywheel, and one wheel. Two motors with gearboxes comprise its drive mechanism. To avoid the coupling problem, in this study, the two motors are mounted in orthogonal directions; therefore, there is no coupling between the actuators. An inclinometer and a gyroscope are installed on the chassis. Its weight and height are approximately 5 kg and 80 cm, respectively. The core of its operating system is the 86duino One development system and comprises some handmade hardware components.

A dual TS control scheme is used for the ODIP real-time control in this study. The system has a planform and one wheel. This is an expansion of the traditional inverted pendulum system. Obviously, the ODIP can be arbitrarily moved in all directions. The ODIP can be considered two independent inverted pendulum systems that are combined. Because of the physical structure, controller design is even more difficult. To overcome this hard control problem, according to the two DC motors mounting in orthogonal directions [13], [25], [26], a dual TS model-based approach for an ODIP control system is suggested. The shafts of the two DC motors are oriented in orthogonal directions. Therefore, the coupling problem in the side plane and front plane can be neglected. Obviously, an ODIP nonlinear system dynamic function can be represented by a set of rule-type local linearizations. Then, the PDC controller can be constructed, and the control gains are obtained by solving LMI methods using the MATLAB LMI Control Toolbox. Finally, from the experimental results, the effectiveness of the proposed intelligent controller is verified by ODIP real-world implementation.

II. BALANCING STRATEGY

The two DC motors are mounted in orthogonal directions; therefore, there is no coupling between the actuators. First, the coordinate system, system parameters, and balance action strategy for an ODIP will be defined in this section. Figure 1 shows the definition of the three-dimensional coordinate system of an ODIP. In this study, the system is decoupled into two subsystems for system analysis.

Newton's third law of motion, which identifies action and reaction force in pairs, is the main balancing concept of the ODIP. In this paper, the ODIP is decoupled into two subsystems, and each subsystem represents the dynamic motions of the ODIP on the side plane and front plane. The first subsystem (shown in Fig. 2) consists of the vertical

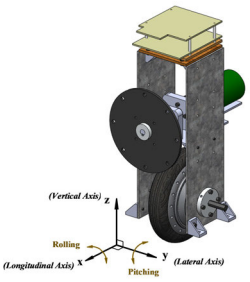


FIGURE 1. Balancing progress of pitching posture on side plane.

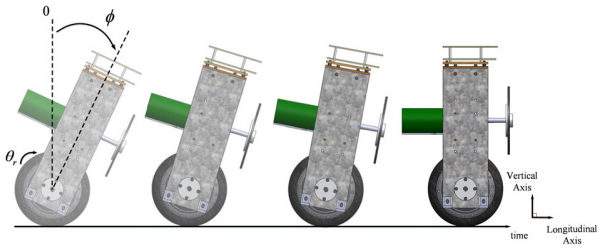


FIGURE 2. Balancing progress of pitching posture on side plane.

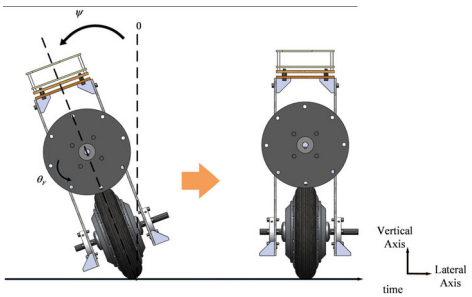


FIGURE 3. Balancing progress of rolling posture on front plane.

axis and longitudinal axis, and the changing velocity of the ground-contacting wheel makes the pitching angle ϕ approach zero. Similarly, the second subsystem (shown in Fig. 3) is made up of the vertical axis and lateral axis, and the changing velocity of the flywheel makes the rolling angle ψ approach zero. As long as the two subsystems are controlled properly, the ODIP can stand straight.

III. DYNAMIC ANALYSIS

A. EQUATION OF MOTION: GENERAL PLANE MOTION

Considering that a rigid body with a mass m kg is externally applied forces and moments, it can be expressed by a free-body diagram and kinetic diagram, which are shown in Fig. 4. Therefore, the three equations of motion can be written as follows [21]:

$$\sum F_x = m\,(\mathbf{a}_G)_x \tag{1}$$

$$\sum F_y = m\,(\mathbf{a}_G)_y \tag{2}$$

$$\sum M_G = I_G\alpha \tag{3}$$

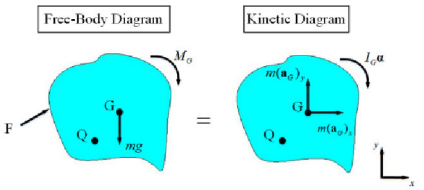


FIGURE 4. Analysis of general plane motion.

where point G is the center of mass and F and M_G are the external force and torque, respectively. $\mathbf{a}_G$ is the body's acceleration caused by external force and moment, I_G is the moment of inertia of the body, and α is the body's angular acceleration. Moreover, in the suffix parameters, x denotes the x-axis and y denotes the y-axis.

In some cases, it is more convenient to sum torques about point Q than G to eliminate unknown forces from the torque summation in Eq. (3). As a result, Eq. (3) can be modified as [27]

$$\sum M_Q = \sum (\Phi_k)_Q \tag{4}$$

where $\sum (\Phi_k)_Q$ represents the moment sum of $I_G\alpha$ and $m\mathbf{a}_G$ about point Q.

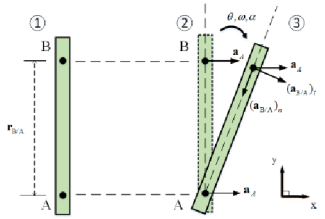


FIGURE 5. Relative-acceleration analysis.

B. RELATIVE MOTION: ACCELERATION

There are two basic motions of transformation for ODIP during the balancing process: translation and rotation. For example, a rectangular plate moving from ① to ③, as shown in Fig. 5, can decompose into two processes. First, the rectangular plate translated from ① to ② gives the same acceleration to point A and point B. Second, the rectangular plate rotates from ② to ③ based on point A, which causes tangential acceleration and normal acceleration in point B. Hence, the relative acceleration equation can be written in the following form [27]:

$$\begin{aligned}\mathbf{a}_B &= \mathbf{a}_A + \mathbf{a}_{B/A} \\ &= \mathbf{a}_A + \left(\mathbf{a}_{B/A}\right)_t + \left(\mathbf{a}_{B/A}\right)_n \\ &= \mathbf{a}_A + \alpha \times \mathbf{r}_{B/A} - \omega^2 \times \mathbf{r}_{B/A}\end{aligned} \tag{5}$$

where

$\mathbf{a}_B$	acceleration of point B;
$\mathbf{a}_A$	acceleration of point A;
$\mathbf{a}_{B/A}$	relative acceleration of B with respect to A;
$\left(\mathbf{a}_{B/A}\right)_t$	relative tangential acceleration component of B with A;

$(\mathbf{a}_{B/A})_n$ relative normal acceleration of B with respect to A;
$\mathbf{r}_{B/A}$ relative position drawn from A to B;
α angular acceleration of the rectangular plate;
ω angular velocity of the rectangular plate;
θ Angle of inclination of the rectangular plate;

C. MATHEMATICAL MODEL OF SIDE PLANE

Figure 8 shows a motion analysis of the ODIP at the side plane, including the free-body diagram and kinetic diagram in the balancing process.

Point R is the center of mass of the rim motor and the position of the rim motor shaft, which is connected with the body of the ODIP. Point P is the center of mass of the ODIP body, which is considered an inverted pendulum system (IPS). If line $\overline{PR}$ is not perpendicular to the longitudinal axis, the IPS will tilt due to the force of gravity. To avoid letting the IPS down, the rim motor has to accelerate for balancing.

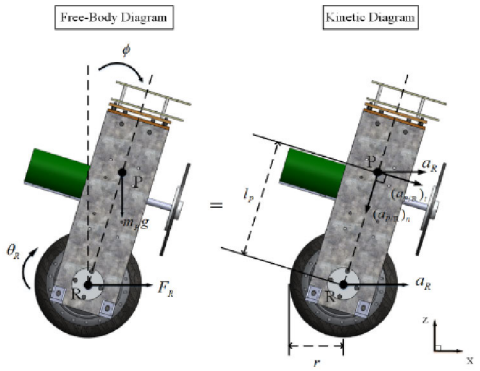


FIGURE 6. Motion analysis of side plane.

The motion of the IPS in the side plane can be considered the same as in Fig. 6, which consists of translation and rotation. According to Eq. (1), Eq. (3) and Fig. 6, the equation of motion for the side plane can be expressed as follows:

$$\begin{aligned} F_R &= m_P[a_R + (a_{P/R})_t \cos\varphi - (a_{P/R})_n \sin\varphi] \\ &\quad + m_R a_R \\ &= (m_P + m_R) r\ddot{\theta}_R + m_P(l_{PR}\ddot{\varphi}\cos\varphi \\ &\quad - l_{PR}\dot{\varphi}^2 \sin\varphi) \qquad (6) \\ m_P g_P \sin\varphi &= m_P[a_R l_{PR}\cos\varphi + l_{PR}^2 (a_{P/R})_n] \\ &= m_P(r\ddot{\theta}_R l_{PR}\cos\varphi + l_{PR}^2\ddot{\varphi}) \qquad (7) \end{aligned}$$

where

m_P mass of the pendulum;
m_R mass of the rim motor;
a_R acceleration of the rim motor;
$(a_{P/R})_t$ relative tangential acceleration component of P with R;
$(a_{P/R})_n$ relative normal acceleration of P with respect to R;
l_{PR} distance between point P and point R;
r radius of the rim motor;
ϕ inclination angle for side plane;
θ_R rotation angle of the rim motor;
F_R the force exerted on point R by rim motor;

Let the side plane's control input $U_{(S)}$ be the torque of the rim motor, i.e.,

$$\begin{aligned} U_{(S)} &= \tau_R \\ &= F_R \times r \qquad (8) \end{aligned}$$

Note that ϕ is also called the pitching angle in the side plane. By combining Eq. (6)Eq. (8), the mathematical model of the side plane is

$$\begin{aligned} \ddot{\varphi} = &\frac{(m_P + m_R) g \sin\varphi}{[m_P(1 - \cos^2\varphi) + m_R] l_{PR}} \\ &- \frac{\cos\varphi}{[m_P(1 - \cos^2\varphi) + m_R] l_{PR} r} U_{(S)} \qquad (9) \end{aligned}$$

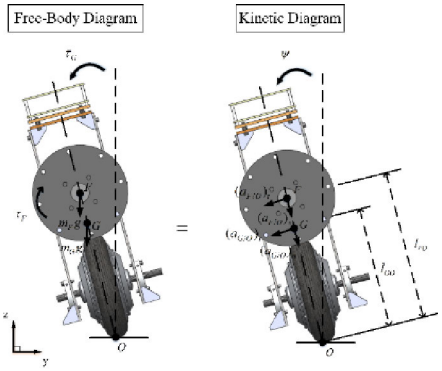


FIGURE 7. Motion analysis of front plane.

D. MATHEMATICAL MODEL OF FRONT PLANE

Figure 7 shows a motion analysis of the ODIP at the front plane, including the free-body diagram and kinetic diagram in the balancing process.

Point F is the center of mass of the flywheel. Point G is the center of mass of the ODIP except for the flywheel.

Because the front plane only rotates during the balancing process, the equation of motion for the front plane can be expressed in the following form by Eqs. (3) and (4):

$$\begin{aligned} &(m_G g l_{GO} + m_F g l_{FO}) \sin\psi + \tau_G \\ &\qquad = (m_G l_{GO}^2 + m_F l_{FO}^2 + I_F)\ddot{\psi} \qquad (10) \end{aligned}$$

where ψ is the inclination angle of the ODIP for the front plane, which is called the rolling angle, and l_{GO} and l_{FO} represent the distances from point O to point F and point G, respectively. τ_G denotes the torque reaction caused by the flywheel, which balances the ODIP in the front plane, and I_F denotes the moment of inertia of the flywheel.

Let the front plane's control input $U_{(F)}$ be the torque of the flywheel. The relationship between the torque generated

by the flywheel and the torque acting on the ODIP can be expressed as follows:

$$\tau_G = k_{(torque)} U_{(F)} \tag{11}$$

where $k_{(torque)}$ is a negative constant that is known by experience. By combining Eq. (10) and Eq. (11), the mathematical model of front plane is

$$\ddot{\psi} = \frac{(m_G g l_{GO} + m_F g l_{FO}) g \sin\psi}{2m_G l_{GO}^2 + m_F l_{FO}^2} + \frac{k_{(torque)}}{2m_G l_{GO}^2 + m_F l_{FO}^2} U_{(F)} \tag{12}$$

IV. FUZZY MODEL

Let the state vectors of the side plane and front plane be defined as $\mathbf{x}_{(S)}(t) = [\,\phi(t)\ \dot{\phi}(t)\,]^T$ and $\mathbf{x}_{(F)}(t) = [\,\psi(t)\ \dot{\psi}(t)\,]^T$, respectively. For the realization of fuzzy modeling, the pitching angle $\phi(t)$ is limited within the range of $\varphi(t) \in [-\pi/6, \pi/6]$, and the rolling angle $\psi(t)$ is limited within the range of $\psi(t) \in [-\pi/36, \pi/36]$.

A. FUZZY MODEL OF SIDE PLANE

The fuzzy model of the side plane is presented as follows:

Model rule i:

$$\text{If } \phi(t) \text{ is } \mu_{i(S)}, \text{ then } \dot{\mathbf{x}}_{(S)}(t) = \mathbf{A}_i \mathbf{x}_{(S)}(t) + \mathbf{B}_i U_{(S)}(t), \quad (i = 1, 2) \tag{13}$$

where the system matrices are

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 1 \\ \frac{(m_P + m_R) g}{m_R l_{PR}} & 0 \end{bmatrix}$$

$$\mathbf{B}_1 = \begin{bmatrix} 0 \\ -\frac{1}{m_R l_{PR} r} \end{bmatrix}$$

$$\mathbf{A}_2 = \begin{bmatrix} 0 & 1 \\ \frac{(m_P + m_R) g \alpha_{(S)}}{[m_P(1-\beta_{(S)}^2) + m_R] l_{PR}} & 0 \end{bmatrix}$$

$$\mathbf{B}_2 = \begin{bmatrix} 0 \\ -\frac{\beta_{(S)}}{[m_P(1-\beta_{(S)}^2) + m_R] l_{PR} r} \end{bmatrix}$$

where $\alpha_{(S)} = \sin(\pi/6)/(\pi/6)$ and $\beta_{(S)} = \cos(\pi/6)$. The membership functions of the side plane of ϕ are presented by the following representation.

$$\mu_{1(S)}(\phi(t)) = \begin{cases} 1, & \phi(t) = 0 \\ 1 - \frac{1 - e^{-\phi(t)/\sigma_{(S)}^2}}{1 - e^{-(\pi/6)^2/\sigma_{(S)}^2}}, & \text{otherwise} \end{cases} \tag{14}$$

$$\mu_{2(S)}(\phi(t)) = \begin{cases} 0, & \phi(t) = 0 \\ \frac{1 - e^{-\phi(t)/\sigma_{(S)}^2}}{1 - e^{-(\pi/6)^2/\sigma_{(S)}^2}}, & \text{otherwise} \end{cases} \tag{15}$$

where $\mu_{1(S)}$ and $\mu_{2(S)}$ are composed of normalized Gaussian functions (shown in Fig. 8) and $\sigma_{(S)}$ is the variance of Gaussian functions.

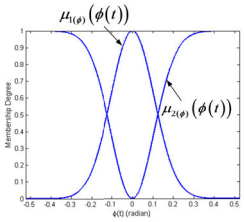


FIGURE 8. Membership functions for fuzzy modeling of side plane.

Then, the defuzzification of the above fuzzy model of the side plane can be deduced as follows:

$$\dot{\mathbf{x}}_{(S)}(t) = \sum_{i=1}^{2} \mu_{i(S)}((t))[\mathbf{A}_i \mathbf{x}_{(S)}(t) + \mathbf{B}_i U_{(S)}(t)] \tag{16}$$

B. FUZZY MODEL OF FRONT PLANE

The fuzzy model of the front plane is designed as follows:

Model rule i:

$$\text{If } \psi(t) \text{ is } \mu_{i(F)}, \text{ then } \dot{\mathbf{x}}_{(F)}(t) = \mathbf{C}_i \mathbf{x}_{(F)}(t) + \mathbf{D}_i U_{(F)}(t), i = (1, 2) \tag{17}$$

The system parameters are defined as follows:

$$\mathbf{C}_1 = \begin{bmatrix} 0 & 1 \\ \frac{(m_G l_{GO} + m_F l_{FO}) g}{2m_G l_{GO}^2 + m_F l_{FO}^2} & 0 \end{bmatrix}$$

$$\mathbf{D}_1 = \begin{bmatrix} 0 \\ \frac{k_{(torque)}}{2m_G l_{GO}^2 + m_F l_{FO}^2} \end{bmatrix}$$

$$\mathbf{C}_2 = \begin{bmatrix} 0 & 1 \\ \frac{(m_G l_{GO} + m_F l_{FO}) g \alpha_{(F)}}{2m_G l_{GO}^2 + m_F l_{FO}^2} & 0 \end{bmatrix}$$

$$\mathbf{D}_2 = \begin{bmatrix} 0 \\ \frac{k_{(torque)}}{2m_G l_{GO}^2 + m_F l_{FO}^2} \end{bmatrix}$$

where $k_{(torque)} = -0.4$, $\alpha_{(F)} = \sin(\pi/36)/(\pi/36)$ and $\beta_{(F)} = \cos(\pi/36)$. The membership functions of the front plane of ψ are presented by

$$\mu_{1(F)}(\psi(t)) = \begin{cases} 1, & \psi(t) = 0 \\ 1 - \frac{1 - e^{-\psi^2(t)/\sigma_{(F)}^2}}{1 - e^{-(\pi/36)^2/\sigma_{(F)}^2}}, & \text{otherwise} \end{cases} \tag{18}$$

$$\mu_{2(F)}(\psi(t)) = \begin{cases} 0, & \psi(t) = 0 \\ \frac{1 - e^{-\psi^2(t)/\sigma_{(F)}^2}}{1 - e^{-(\pi/36)^2/\sigma_{(F)}^2}}, & \text{otherwise} \end{cases} \tag{19}$$

where $\mu_{1(F)}$ and $\mu_{2(F)}$ are composed of a normalized Gaussian function (shown in Fig. 9) and $\sigma_{(F)}$ is the variance of Gaussian functions.

Consequently, the defuzzification of the abovementioned fuzzy model of the front plane can be derived as

$$\dot{\mathbf{x}}_{(F)}(t) = \sum_{i=1}^{2} \mu_{i(F)}(\varphi(t))[\mathbf{C}_i \mathbf{x}_{(F)}(t) + \mathbf{D}_i U_{(F)}(t)] \tag{20}$$

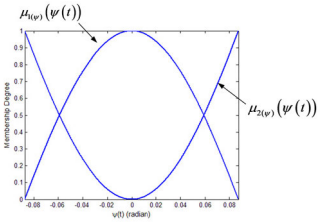


FIGURE 9. Membership functions for fuzzy modeling of front plane.

Algorithm 1 denotes the 86duino One code to obtain the system plant model.

Algorithm1: Fuzzy model algorithm

Input: System angle $\boldsymbol{x}$, mean value of function *mm*, variance value of Gaussian function *msd*

Output: Desire fuzzy model

1. Limit the input variable into the interval between the largest mean value and the smallest mean value
2. Set the input variables between which two mean values of the Gaussian function, for example:
   if(((x >= mm1)&&(x <= mm2))
   where *mm1*, and *mm2* are the mean value for *x*
3. Calculate the value of the fuzzy membership function, for example:
   mem_model_1 = 1-(1-(*exp*(-*pow*(*x*,2)/*pow*(*sd1*,2)))/(1-*exp*(-*pow*(*pi*/36,2)/*pow*(*sd1*,2));
   mem_model_2 = (1-(*exp*(-*pow*(*x*,2)/*pow*(*sd1*,2)))/(1-*exp*(-*pow*(*pi*/36,2)/*pow*(*sd1*,2));

End Algorithm

V. FUZZY CONTROL DESIGN

In this paper, the model-based fuzzy controller design employs the concept of parallel distributed compensation (PDC).

A. FUZZY CONTROLLER OF SIDE PLANE

The fuzzy controller of the side plane is presented as follows:

Controller rule i:

$$\text{If } \phi(t) \text{ is } \mu_{i(S)}, \text{ then } U_{(S)}(t) = \mathbf{F}_{i(S)}\mathbf{x}_{(S)}(\text{t}), \quad (\text{i}=1, 2) \tag{21}$$

where $\mathbf{F}_{1(S)}$ and $\mathbf{F}_{2(S)}$ are the feedback gains. The whole fuzzy controller can be expressed as

$$U_{(S)}(t) = \sum_{i=1}^{2} \mu_{i(S)}(\phi(t))\mathbf{F}_{i(S)}\mathbf{x}_{(S)}(\text{t}) \tag{22}$$

Hence, each feedback gain $\mathbf{F}_{i(S)}$ in a consequent part needs to be resolved. By substituting (22) into (16), the closed-loop fuzzy model can be obtained as follows:

$$\begin{aligned}\dot{\mathbf{x}}_{(S)}(t) = &\sum_{i=1}^{2} \mu_{i(S)}(\phi(t))\mathbf{G}_{ii(S)}\mathbf{x}_{(S)}(t) \\ &+2\mu_{1(S)}(\phi(t))\mu_{2(S)}(\phi(t)) \\ &\times \left\{\frac{\mathbf{G}_{12(S)} + \mathbf{G}_{21(S)}}{2}\right\}\mathbf{x}_{(S)}(t)\end{aligned} \tag{23}$$

where $\mathbf{G}_{ij(S)} = \mathbf{A}_i + \mathbf{B}_i\mathbf{F}_{j(S)}$.

B. FUZZY CONTROLLER OF FRONT PLANE

The fuzzy controller of the front plane is shown as follows:

Controller rule i:

$$\text{If } \psi(t) \text{ is } \mu_{i(F)}, \text{ then } U_{(F)}(t) = \mathbf{F}_{i(F)}\mathbf{x}_{(F)}(t), \quad (\text{i} = 1, 2) \tag{24}$$

where $\mathbf{F}_{1(F)}$ and $\mathbf{F}_{2(F)}$ are the local feedback gains. The overall fuzzy controller is obtained as

$$U_{(F)}(t) = \sum_{i=1}^{2} \mu_{i(F)}(\phi(t))\mathbf{F}_{i(F)}\mathbf{x}_{(F)}(\text{t}) \tag{25}$$

Here, each $\mathbf{F}_{i(F)}$ needs to be determined to push the system error to approach zero. By substituting (25) into (20), the closed-loop fuzzy model can be obtained as follows:

$$\begin{aligned}\dot{\mathbf{x}}_{(F)}(t) = &\sum_{i=1}^{2} \mu_{i(F)}(\psi(t))\mathbf{G}_{ii(F)}\mathbf{x}_{(F)}(t) \\ &+2\mu_{1(F)}(\psi(t))\mu_{2(F)}(\psi(t)) \\ &\left\{\frac{\mathbf{G}_{12(F)} + \mathbf{G}_{21(F)}}{2}\right\}\mathbf{x}_{(F)}(t)\end{aligned} \tag{26}$$

where $\mathbf{G}_{ij(F)} = \mathbf{C}_i + \mathbf{D}_i\mathbf{F}_{j(F)}$.

Algorithm 2 is the 86duino One code to send PWM to drive motors.

Algorithm3: Fuzzy controller algorithm

Input: Angle error $\boldsymbol{E}$, mean value of Gaussian function*cm*,
variance value of Gaussian function *csd*,
$F1$ and $F2$
are local feedback gains.

Output: Desire fuzzy output for balance control $\boldsymbol{U}$

1. Limit the input variable into the interval between the largest mean value and the smallest mean value
2. Set the input variables between which two mean values of the Gaussian function, for example:
   if (((x >= cm1)&&(x <= cm2))
   where *cm1*, and *cm2* are the mean value for *E*
3. Calculate the value of the fuzzy membership function, for example:
   mem_controller_1 = 1-(1-(*exp*(-*pow*(*E*,2)/*pow*(*csd1*,2)))/(1-*exp*(-*pow*(*pi*/6,2)/*pow*(*csd1*,2));
   mem_controller_2 = (1-(*exp*(-*pow*(*E*,2)/*pow*(*csd1*,2)))/(1-*exp*(-*pow*(*pi*/6,2)/*pow*(*csd1*,2));
4. Calculate the output of the fuzzy control algorithm by the center of gravity defuzzification as:
   $\boldsymbol{U}$= (**mem_controller_1**∗*(F1)*∗*E*
   + **mem_controller_2**∗*(F2)*∗*E)*

End Algorithm

C. STABILITY ANALYSIS OF CLOSED-LOOP T-S FUZZY MODEL

According to Eqs. (23) and (26), the closed-loop T-S fuzzy model of the side plane and front plane have the same mathematical form. Therefore, the stability analysis of the closed-loop T-S fuzziness for the overall ODIP can be derived as follows.

Lemma 1 [7]: The fuzzy control system of (23) and (26) is quadratically stable at the equilibrium points if symmetric matrices $\mathbf{P}_{(z)}$ and $\mathbf{W}_{ij(z)}$ exist for $z = S$and F such that (27)–(30) are satisfied:

$$\mathbf{P}_{(z)} > 0 \tag{27}$$

$$\mathbf{G}_{ii(z)}^{T}\mathbf{P}_{(z)} + \mathbf{P}_{(z)}\mathbf{G}_{ii(z)} + W_{ii(z)} < 0 \tag{28}$$

$$\left(\frac{\mathbf{G}_{12(z)} + \mathbf{G}_{21(z)}}{2}\right)^{T}\mathbf{P}_{(z)} + \mathbf{P}_{(z)}\left(\frac{\mathbf{G}_{12(z)} + \mathbf{G}_{21(z)}}{2}\right) + \mathbf{W}_{12(z)} < 0 \tag{29}$$

$$\tilde{\mathbf{W}}_{(z)} \equiv \begin{bmatrix} \mathbf{W}_{11(z)} & \mathbf{W}_{21(z)} \\ \mathbf{W}_{12(z)} & \mathbf{W}_{22(z)} \end{bmatrix} > 0 \tag{30}$$

By defining the matrices $\mathbf{F}_{k_{(z)}} = \mathbf{N}_{k(z)}\mathbf{R}_{(z)}^{-1}$, $\mathbf{R}_{(z)} = \mathbf{P}_{(z)}^{-1}$, and $\mathbf{Y}_{ij_{(z)}} = \mathbf{R}_{(z)}\mathbf{W}_{ij_{(z)}}\mathbf{R}_{(z)}$ for $k = i$and$j$ to meet the format of the LMT toolbox, we can apply them into the LMI method to obtain $\mathbf{P}_{(z)}$, $\mathbf{F}_{k_{(z)}}$, and $\mathbf{W}_{ij(z)}$.

In practical applications, the maximum limitation of the control force is used in this study to reduce the probability of burning out the hardware circuit. Moreover, a control force with a maximum upper bound is easier to implement in reality.

Lemma 2 [7]: For the fuzzy control system of (23) and (26), suppose that the initial error vector is unknown but its positive upper bound $\varepsilon_{(z)}$ is known for $z = S$and$F$. Then, the control inputs (22) and (25) can be enforced to derive the constraint positive upper bound $\rho_{(z)}$ if the following conditions are added into the LMIs used in Lemma 1:

$$\varepsilon_{(z)}^{2}\mathbf{I} \le \mathbf{R}_{(z)} \tag{31}$$

$$\begin{bmatrix} \mathbf{R}_{(z)} & \mathbf{N}_{k(z)}^{T} \\ \mathbf{N}_{k(z)} & \rho_{(z)}^{2}I \end{bmatrix} \ge 0 \tag{32}$$

In Fig. 10, ϕ^* and ψ^* are command inputs and set to zero in this study. Moreover, e_ϕ and Δe_ϕ are the pitch angle tracking error and the change rate of the pitch angle tracking error, respectively. e_ψ and Δe_ψ are the roll angle tracking error and the change rate of the roll angle tracking error. Fig. 11 shows the program flow chart.

VI. HARDWARE ARCHITECTURE

This section introduces the ODIP hardware design based on an open-source embedded controller. Figure 12 shows a photograph of the ODIP, and Fig. 13 shows the hardware configuration of the ODIP.

A. HARDWARE STRUCTURE

The size of the ODIP is approximately 33 cm long, 34 cm wide, and 60 cm high. The total weight of ODIP approaches

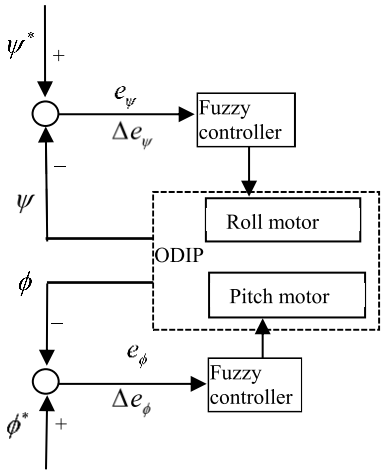


FIGURE 10. The configuration of proposed control system.

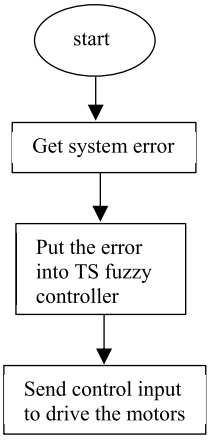


FIGURE 11. The program flow chart.

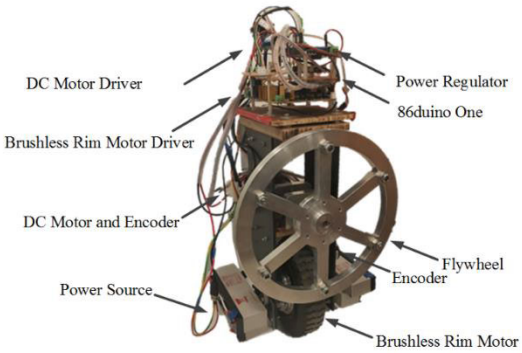


FIGURE 12. Photograph of the ODIP.

10 kg. The 2.17-kg flywheel is tightly connected to the front of a DC motor shaft made of aluminum, and its weight and shape directly influence the control performance. Carbon fiber is chosen as the material for the body of the ODIP; its tensile strength is approximately 7 times greater than aluminum, and its weight is nearly 4 times less than aluminum under a same volume. The abovementioned advantages make

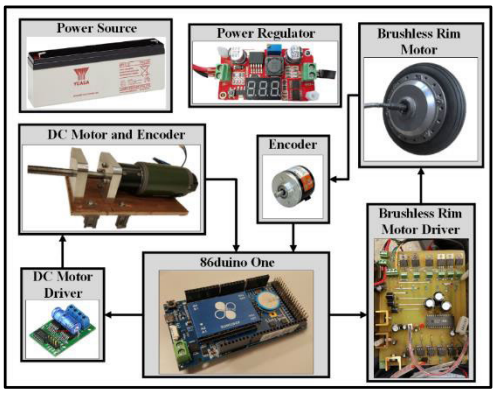


FIGURE 13. Hardware Configuration of ODIP.

the ODIP sturdy and durable, so the whole system can ideally be considered a rigid body.

All electrical circuits are placed on the top of the ODIP, which contains a power regulator, an embedded controller, a motor driver module, and a brushless rim motor driver board. All motor driver circuits are designed to prevent counter electromotive force, which may cause damage to the motor deriver's IC and embedded controller when the motor changes its rotating direction instantly.

The ODIP has two actuators: a 24-V DC motor manufactured by King Motor Company Ltd. and a 24-V DC brushless rim motor manufactured by Elebike Company, Ltd. A planetary gearbox with a reduction ratio of 1:6 is installed at the head of the DC motor, which makes the DC motor more powerful so that the ODIP has antidisturbance ability. To address the coupling problem owing to the two DC motors, the shafts of the two DC motors are oriented in orthogonal directions. Therefore, the platform can move directly in any direction. Moreover, the control signals for the two actuators, pulse-width modulation (PWM), are generated from the embedded controller. The interface for data transmission between the personal computer (PC) and embedded controller is a universal serial bus (USB) 2.0, and it can also program the embedded controller from the PC.

B. EMBEDDED CONTROLLER

The embedded controller employed in this paper is the 86duino One controller, which is a high-performance board based on a 32-bit x86 processor Vortex86EX system-on-a-chip (SoC). The 86duino One provides 45 digital I/O pins in which 11 pins can generate 32-bit PWM, 7 analog-to-digital converter (ADC) input pins, 128-MB DDR3 memory, 8-MB flash memory, and a clock speed of 300 MHz. Signals from the inclinometer and gyro are acquired by sensors and a filter circuit. The inclinometer is used to measure the system angle, and the gyro is used to determine the system angle derivative.

The inertia measurement unit (IMU), an LSM330DLC, is integrated into the 86duino One. The LSM330DLC is a system-in-package featuring a 3D digital accelerometer and a 3D digital gyroscope; therefore, the processor can obtain the ODIP's posture in real time by data fusion.

The proposed control algorithm can be realized by a coded program in the 86duino integrated development environment (IDE), and the syntax of the 86duino IDE is similar to the C/C++ and Arduino IDE syntax.

TABLE 1. Parameters and values of the ODIP.

Symbol	Value
m_F	2.17 (kg)
m_G	10.305 (kg)
m_R	2.4 (kg)
m_P	10.075 (kg)
l_{FO}	0.35 (m)
l_{GO}	0.2652 (m)
r	0.1 (m)
l_{PR}	0.3185 (m)

VII. EXPERIMENT RESULTS

In this section, the proposed TS fuzzy controller is used to control the ODIP. The system parameters and values of the ODIP are shown in Table 1. Generally, the Arduino 86duino One unit is suitable for control algorithm implementation and system control. Moreover, the proposed two motors are sufficient for system control. Unfortunately, the ODIP will be uncontrollable when a large extra disturbance is given. The reason is that the given disturbance exceeds the control ability to maintain system stability. When a large extra disturbance is added, the ODIP will fall down. In this work, the proposed ODIP can afford maximum angles of 10 degrees and 5 degrees in the pitch and roll, respectively.

After substituting the parameters into the fuzzy models (13) and (17), the system matrices are

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 1 \\ 127.4789 & 0 \end{bmatrix}, \quad \mathbf{B}_1 = \begin{bmatrix} 0 \\ -10.4167 \end{bmatrix}$$
$$\mathbf{A}_2 = \begin{bmatrix} 0 & 1 \\ 59.3972 & 0 \end{bmatrix}, \quad \mathbf{B}_2 = \begin{bmatrix} 0 \\ -4.4017 \end{bmatrix}$$

and

$$C_1 = \begin{bmatrix} 0 & 1 \\ 19.9728 & 0 \end{bmatrix}, \quad D_1 = \begin{bmatrix} 0 \\ -0.2332 \end{bmatrix}$$
$$C_2 = \begin{bmatrix} 0 & 1 \\ 19.9475 & 0 \end{bmatrix}, \quad D_2 = \begin{bmatrix} 0 \\ -0.2332 \end{bmatrix}$$

The ODIP is modeled in the range $\phi(t) \in [-\pi/6, \pi/6]$ and $\psi(t) \in [-\pi/36, \pi/36]$. According to Lemma 2, the initial error upper bounds are $\varepsilon_{(S)} = \pi/6$ and $\varepsilon_{(F)} = \pi/36$.

Moreover, the control input upper bounds are $\rho_{(S)} = 12.5$ and $\rho_{(Z)} = 60$. Then, the MATLAB LMI toolbox can be

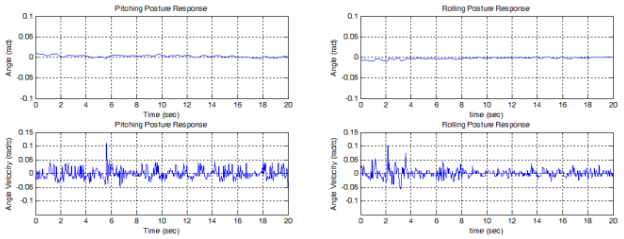
utilized to obtain $\mathbf{P}_{(z)}$, $\mathbf{F}_{k_{(z)}}$, and $\mathbf{W}_{ij(z)}$ for $z = S$and F as follows.

$$\mathbf{F}_{1(S)} = \begin{bmatrix} 13.8545 & 1.1335 \end{bmatrix}$$
$$\mathbf{F}_{2(S)} = \begin{bmatrix} 14.8238 & 1.7238 \end{bmatrix},$$
$$\mathbf{P}_{(S)} = \begin{bmatrix} 2 & 0.2379 \\ 0.2379 & 0.0507 \end{bmatrix},$$
$$\tilde{\mathbf{W}}_{(S)} = \begin{bmatrix} 3.8711 & 0.7685 & -0.1118 & 0.0038 \\ 0.7685 & 0.351 & 0.0038 & 0.0742 \\ -0.1118 & 0.0038 & 1.3813 & 0.0536 \\ 0.0038 & 0.0742 & 00536 & 0.1556 \end{bmatrix}$$

and

$$\mathbf{F}_{1(F)} = \begin{bmatrix} 114.0779 & 25.5282 \end{bmatrix}$$
$$\mathbf{F}_{2(F)} = \begin{bmatrix} 114.1485 & 25.5560 \end{bmatrix},$$
$$\mathbf{P}_{(F)} = \begin{bmatrix} 5.0884 & 1.1389 \\ 1.1389 & 0.2549 \end{bmatrix},$$
$$\tilde{\mathbf{W}}_{(F)} = \begin{bmatrix} 7.6061 & 1.7032 & -1.0214 & -0.2286 \\ 1.7032 & 0.3814 & -0.2286 & -0.0512 \\ -1.0214 & -0.2286 & 7.6835 & 1.719 \\ -0.2286 & -0.0512 & -1.719 & 0.3846 \end{bmatrix}.$$

Moreover, in this system, $\sigma_S = 0.15$ and $\sigma_F = 0.15$. Two cases, balance control and balance control with external disturbance, are illustrated here.



(a) (b)

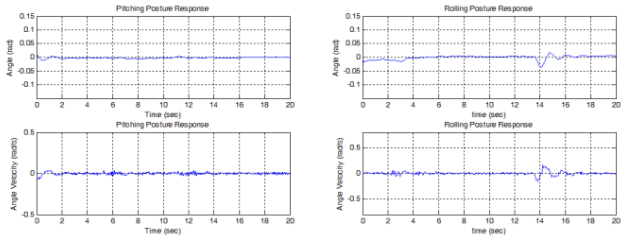
FIGURE 14. The system performance of the ODIP balance control at a fixed point. (a) The inclination angle and the angular velocity for the side plane. (b) The inclination angle and the angular velocity for the front plane.

A. BALANCE CONTROL

In this subsection, the ODIP balance control at a fixed position is tested. The ODIP is required to stay at the origin point. Fig. 14 shows the system responses with the initial states $\begin{bmatrix} \phi(0) & \psi(0) \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix}$. Fig. 14(a) shows the inclination angle and the angular velocity for the side plane. Fig. 14(b) shows the inclination angle and the angular velocity for the front plane. Obviously, when the ODIP system starts, the angle of the ODIP is at approximately zero degrees. This means the ODIP is standing upright. In this test duration, the ODIP always stands at approximately zero degrees. It can be seen that the ODIP can stably stand upright at the origin point.

B. BALANCE CONTROL WITH EXTERNAL DISTURBANCE

In this experimental case, the ODIP is initially at the origin. Fig. 15 shows the system responses with the initial states $\begin{bmatrix} \phi(0) & \psi(0) \end{bmatrix} = \begin{bmatrix} 0 & 0 \end{bmatrix}$. Then, at 0.5 seconds, an external disturbance is given by tapping to the side plane. Moreover, at 14 seconds, an external disturbance is also added to the front plane by tapping. Fig. 15 shows the system performance of the ODIP with an external disturbance. Fig. 15(a) shows the inclination angle and the angular velocity for the side plane. The system chattering converges rapidly. Fig. 15(b) shows the inclination angle and the angular velocity for the front plane. In Fig. 15(b), a large overshoot occurs at approximately 14 seconds because of an external disturbance. However, the system error convergence is rapid. Moreover, because the two DC motors are oriented in orthogonal directions, the external disturbance given to the front plane has no effect on the side plane. Therefore, the platform can move directly in any direction. Obviously, the proposed controller can deal with external disturbances.



(a) (b)

FIGURE 15. The system performance of the ODIP balance control at a fixed point with external disturbance. (a) The inclination angle and the angular velocity for the side plane. (b) The inclination angle and the angular velocity for the front plane.

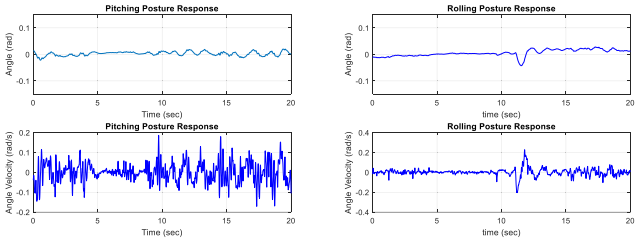


FIGURE 16. The system performance of the ODIP balance control for PID at a fixed point with external disturbance. (a) The inclination angle and the angular velocity for the side plane. (b) The inclination angle and the angular velocity for the front plane.

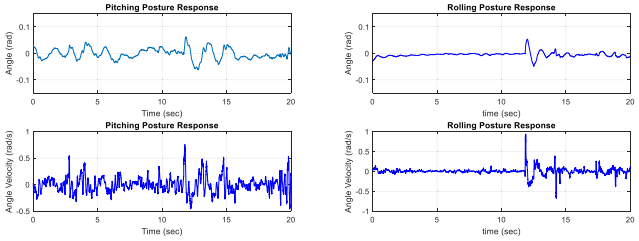


FIGURE 17. The system performance of the ODIP balance control for FLC at a fixed point with external disturbance. (a) The inclination angle and the angular velocity for the side plane. (b) The inclination angle and the angular velocity for the front plane.

For comparison, the classic PID and fuzzy logic control (FLC) are used to control the ODIP.

Figures 16 and 17 show the comparison results for the classic PID and the fuzzy logic control. Clearly, the system responses in Fig. 15 are better than those in the other figures.

Based on the above experimental results, the controller shows good control performance. This indicates that the proposed TS fuzzy control scheme is suitable for the real-time control of an ODIP.

VIII. CONCLUSION

In conclusion, an ODIP has been successfully implemented by using a novel fuzzy control scheme in this study. An ODIP is a new type of highly nonlinear controlled plant. Moreover, the mathematic model of an ODIP is derived. The proposed TS fuzzy controller is designed to maintain the ODIP balance. When the ODIP is hit by an external disturbance, the controller can still prevent the system from falling down. The experimental results verify that the proposed control schemes are effective for ODIP real-world control.

REFERENCES

[1] H. Zhou, H. Deng, and J. Duan, "Hybrid fuzzy decoupling control for a precision maglev motion system," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 1, pp. 389–401, Feb. 2018.
[2] S. Huang, W. Liang, and K. K. Tan, "Intelligent friction compensation: A review," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 4, pp. 1763–1774, Aug. 2019.
[3] C. S. Chin and W. P. Lin, "Robust genetic algorithm and fuzzy inference mechanism embedded in a sliding-mode controller for an uncertain underwater robot," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 2, pp. 655–666, Apr. 2018.
[4] K. Zhu and Y. Zhang, "A cyber-physical production system framework of smart CNC machining monitoring system," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 6, pp. 2579–2586, Dec. 2018.
[5] S.-K. Oh, H.-J. Jang, and W. Pedrycz, "A comparative experimental study of type-1/type-2 fuzzy cascade controller based on genetic algorithms and particle swarm optimization," *Expert Syst. Appl.*, vol. 38, no. 9, pp. 11217–11229, Sep. 2011.
[6] T. Haidegger, L. Kovács, S. Preitl, R. E. Precup, B. Benyo, and Z. Benyo, "Controller design solutions for long distance telesurgical applications," *Int. J. Artif. Intell.*, vol. 6, no. S11, pp. 48–71, Mar. 2011.
[7] S.-K. Oh, W.-D. Kim, and W. Pedrycz, "Design of optimized cascade fuzzy controller based on differential evolution: Simulation studies and practical insights," *Eng. Appl. Artif. Intell.*, vol. 25, no. 3, pp. 520–530, Apr. 2012.
[8] R.-E. Precup and M. L. Tomescu, "Stable fuzzy logic control of a general class of chaotic systems," *Neural Comput. Appl.*, vol. 26, no. 3, pp. 541–550, Apr. 2015.
[9] A. Turnip and H. J. Panggabean, "Hybrid controller design based magneto-rheological damper lookup table for quarter car suspension," *Int. J. Artif. Intell.*, vol. 18, no. 1, pp. 193–206, Mar. 2020.
[10] C.-H. Chiu and C.-C. Chang, "Design and development of Mamdani-like fuzzy control algorithm for a wheeled human-conveyance vehicle control," *IEEE Trans. Ind. Electron.*, vol. 59, no. 12, pp. 4774–4783, Dec. 2012.
[11] C. M. Druitt and G. Alici, "Intelligent control of electroactive polymer actuators based on fuzzy and neurofuzzy methodologies," *IEEE/ASME Trans. Mechatronics*, vol. 19, no. 6, pp. 1951–1962, Dec. 2014.
[12] O. Begovich, E. N. Sanchez, and M. Maldonado, "Takagi-Sugeno fuzzy scheme for real-time trajectory tracking of an underactuated robot," *IEEE Trans. Control Syst. Technol.*, vol. 10, no. 1, pp. 14–20, Jan. 2002.
[13] C.-H. Chiu and Y.-F. Peng, "Design of Takagi-Sugeno fuzzy control scheme for real world system control," *Sustainability*, vol. 11, no. 14, pp. 1–10, Jul. 2019.
[14] Y.-W. Liang, S.-D. Xu, D.-C. Liaw, and C.-C. Chen, "A study of T–S model-based SMC scheme with application to robot control," *IEEE Trans. Ind. Electron.*, vol. 55, no. 11, pp. 3964–3971, Nov. 2008.
[15] J. Zhou, N. Zhang, C. Li, Y. Zhang, and X. Lai, "An adaptive Takagi–Sugeno fuzzy model-based generalized predictive controller for pumped-storage unit," *IEEE Access*, vol. 7, pp. 103538–103555, 2019.
[16] S. Yan, M. Shen, S. K. Nguang, G. Zhang, and L. Zhang, "A distributed delay method for event-triggered control of T–S fuzzy networked systems with transmission delay," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 10, pp. 1963–1973, Oct. 2019.
[17] J. Zhang, X. Wang, and X. Shao, "Design and real-time implementation of Takagi–Sugeno fuzzy controller for magnetic levitation ball system," *IEEE Access*, vol. 8, pp. 38221–38228, Feb. 2020.
[18] E. Kim and H. Lee, "New approaches to relaxed quadratic stability condition of fuzzy control systems," *IEEE Trans. Fuzzy Syst.*, vol. 8, no. 5, pp. 523–534, Oct. 2000.
[19] C.-H. Chiu and W.-J. Wang, "Implementation of a ball inverted pendulum with omnidirectional moving ability using a robust fuzzy control strategy," *ISA Trans.*, vol. 86, pp. 287–298, Mar. 2019.
[20] C.-H. Chiu, "Adaptive fuzzy control strategy for a single-wheel transportation vehicle," *IEEE Access*, vol. 7, pp. 113272–113283, Aug. 2019.
[21] C.-H. Chiu and Y.-F. Peng, "Position and angle control for a two-wheel robot," *Int. J. Control, Autom. Syst.*, vol. 15, no. 5, pp. 2343–2354, Oct. 2017.
[22] C.-H. Chiu and C.-M. Lin, "Implementation of a biomimetic flapping-wing robot based on Internet of Things technology," in *Advanced Mechanical Science and Technology for the Industrial Revolution 4.0*. Singapore: Springer, 2018.
[23] C. Chiu and C. Chang, "Wheeled human transportation vehicle implementation using output recurrent fuzzy control strategy," *IET Control Theory Appl.*, vol. 8, no. 17, pp. 1886–1895, Nov. 2014.
[24] C.-H. Chiu and C.-Y. Wu, "Adaptive fuzzy control strategy for a single-wheel transportation vehicle," *IEEE Access*, vol. 7, pp. 84837–84849, May 2019.
[25] J. Lee, S. Han, and J. Lee, "Decoupled dynamic control for pitch and roll axes of the unicycle robot," *IEEE Trans. Ind. Electron.*, vol. 60, no. 9, pp. 3814–3822, Sep. 2013.
[26] S. I. Han and J. M. Lee, "Balancing and velocity control of a unicycle robot based on the dynamic model," *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, pp. 405–413, Jan. 2015.
[27] R. C. Hibbeler, *Engineering Mechanics: Dynamics*. Upper Saddle River, NJ, USA: Prentice-Hall, 2012.

CHIH-HUI CHIU was born in Taiwan. He received the B.S. degree in electrical engineering from the Tatung Institute of Technology, Taipei, Taiwan, in 1994, and the M.S. and Ph.D. degrees in electrical engineering from National Central University, Taiwan, in 1996 and 2000, respectively.

He is currently a Professor with the Department of Communications, Navigation and Control Engineering, National Taiwan Ocean University, Keelung, Taiwan. His research interests include fuzzy logic theory, intelligent control, robots, mechatronics, and control theory applications.

YAO-TING HUNG was born in Taiwan. He received the B.S. degree from National Taiwan Ocean University, Keelung, Taiwan, and the M.S. degree in electrical engineering from Yuan Ze University, Taoyuan, Taiwan. His research interests include fuzzy logic theory, intelligent control, and control theory applications.

YA-FU PENG was born in Taoyuan, Taiwan, in 1969. He received the B.S. degree from Yuan Ze University, Taiwan, the M.S. degree from National Taiwan Ocean University, Keelung, Taiwan, and the Ph.D. degree from Yuan Ze University, in 1994, 1996, and 2003, respectively, all in electrical engineering.

Since 1996, he has been with the Department of Electrical Engineering, Chien-Hsin University of Science and Technology, Taiwan, where he is currently a Professor. His research interests include motor control, adaptive control, intelligent control, and flight control systems.