

Adaptive Neuro-Fuzzy Control of a Spherical Rolling Robot Using Sliding Mode Control Theory-Based Online Learning Algorithm

Erkan Kayacan *Student Member* , Erdal Kayacan *Senior Member* , Herman Ramon and Wouter Saeys

Abstract—As a model is only an abstraction of the real system, unmodeled dynamics, parameter variations and disturbances can result in poor performance of a conventional controller based on this model. In such cases, a conventional controller cannot remain well-tuned. This paper presents the control of a spherical rolling robot by using an adaptive neuro-fuzzy controller in combination with a SMC theory-based learning algorithm. The proposed control structure consists of a neuro-fuzzy network and a conventional controller which is used to guarantee the asymptotic stability of the system in a compact space. The parameter updating rules of the fuzzy-neuro system using SMC theory are derived, and the stability of the learning is proven using a Lyapunov function. The simulation results show that the control scheme with the proposed SMC theory-based learning algorithm is able to not only eliminate the steady state error but also to improve the transient response performance of the spherical rolling robot without knowing its dynamic equations.

I. INTRODUCTION

MOBIL ROBOTS are increasingly used in a variety of non-industrial applications such as security surveillance, search and rescue, children education, entertainment, etc. Spherical rolling mechanisms exhibit a number of advantages with respect to wheeled and legged mechanisms. All mechanical and electrical components including the actuation mechanism are securely located inside a spherical shell rolling itself over the ground surface. The motion of a sphere rolling without slipping over a surface is governed by non-holonomic constraints, therefore spherical rolling robots are classified as non-holonomic mechanical systems.

The fundamental difference between spherical and wheeled rolling motions is the instantaneous number of degrees of freedom (DOF) between the mobile body and ground surface. Since the sphere can simultaneously rotate around the transverse and longitudinal axes, its instantaneous mobility is greater than that of a wheel. Besides, spherical rolling mechanisms can change their direction of motion easier than wheeled mechanisms. Unlike wheeled and legged mechanisms, spheres can not fall over. The general problem of stability of equilibrium frequently encountered in mobile robotics is naturally avoided with the use of spherical rolling. On the other hand, highly complex nonlinear equations are needed to describe the dynamics of spherical rolling systems.

E. Kayacan, E. Kayacan, H. Ramon and W. Saeys are with the Division of Mechatronics, Biostatistics and Sensors, Department of Biosystems, KU Leuven, Kasteelpark Arenberg 30, B-3001 Leuven, Belgium. e-mail: {erkan.kayacan, erdal.kayacan, herman.ramon, wouter.saeys}@biw.kuleuven.be

A number of spherical robot prototypes have been developed in recent years. One of the present mechanical structures consists of a wheeled vehicle located at the bottom of the sphere. In such a scheme, the motion is provided by the interaction of the vehicle's wheels and the sphere. The wheeled vehicle's motion inside the sphere being also non-holonomic, the overall system is the combination of two non-holonomic mechanical systems [1]. A linear model for the longitudinal dynamics of the vehicle has been proposed to simplify the equations of motion of the system [2]. Since spherical rolling robots have highly nonlinear and very complex dynamics, the linearization approach is not proper for the study of these systems. A similar design utilizes a single wheel resting on the bottom of the sphere [3].

In another mechanical design, the driving motion of the rolling robot is obtained by changing the position of the mass center inside the sphere, which provides a gravitational torque [4], [5]. Such a design has been proposed in [6], [7] where two DC motors associated with two flywheels are mounted inside the sphere. When one of the flywheels turns around an arbitrarily chosen reference direction, the spherical rolling robot rolls in the opposite direction due to the conservation of angular momentum. However, such a mechanism cannot roll over curvilinear trajectories. In order to change its direction of motion, the sphere must stop rolling and then turn left or right around the vertical axis. A similar design utilizes two perpendicular rotors mounted inside the sphere [8].

A spherical robot with one DOF pendulum as actuation mechanism was designed in [9]. A simplified dynamic model of a spherical robot with one DOF pendulum is derived under some assumptions in [10], [11]. One of these assumptions is that the longitudinal and lateral motions are decoupled from each other. Motion control has been established by feedback linearization considering the decoupled dynamics [12] and only driving motion [13].

Having different mechanical designs, spherical robots are used for very different purposes. For example, Michaud [14], [15] invented a spherical mobile robot called Roball, which is a tool in child-development studies. Roball, an autonomous robot, contributes to the development of children's language, affective, intellectual and social skills. Longitudinal and lateral models of Roball are discussed in [16].

Due to the highly nonlinear equations constituting the dynamic, a kinematic model of the spherical rolling robots is usually used for path planning and in the design of controllers. A trajectory was simulated using a kinematic model in [17].

Under some assumptions, a feedback controller for a kinematic model and a dynamic model based on the backstepping method are also designed [18].

In most of the papers in literature, some assumptions are made to obtain simplified dynamic models for spherical rolling robots. For example, feedback control and feedback linearization have been applied to the system by using these simplified dynamic models [19]. Furthermore, the linearized dynamic model is also obtained and some linear control methods are applied to the system. However, these approaches are not feasible in practice due to the highly complex nonlinear equations. Moreover, although the kinematic model has been used to control the system because of its simplicity, it does not consist of the system's properties such as, masses, moment of inertia, etc... Thus, kinematic models are not sufficient to control the spherical rolling robots.

In practice, the system behavior is affected by unmodeled dynamics, parameter variations, uncertainties and disturbances. For instance, when the spherical rolling robot rolls on a surface, the static and the dynamic friction cannot be neglected. Since conventional controllers are time-invariant controllers, these terms cause discontinuities and nonlinearities which render conventional control invalid. Therefore, advanced intelligent control techniques are needed to overcome the stated shortcomings. The control scheme used in this paper is called feedback-error-learning and was firstly proposed in [20] for the control of robot manipulators. It is based on the parallel work of a fuzzy neural network (FNN)-based controller and a conventional feedback controller.

All systems are nonlinear and continuous in nature but, the data flow between receiver and transmitter occurs in discrete-time domain in practice resulting in some modeling errors. In [21], the modeling errors are estimated by using an artificial neural network (ANN) estimator, and tracking errors are reduced in discrete nonlinear systems. Even if the dynamic equations for the system at hand would be available, uncertainties can exist because of the noise on the sensors, environmental changes and/or nonlinear characteristics of the actuators. Radial basis function neural networks with an online learning algorithm have been proposed to handle the uncertainty problem [22]. The dynamics of mobile robots with uncertainties and external disturbances have been studied in [23]. Self-recurrent wavelet neural networks are also proposed to estimate model uncertainties and external disturbances in the dynamics of nonholonomic wheeled mobile robots [24]. Dual neuro-adaptive control for the discrete-time dynamic control of non-holonomic mobile robots is introduced in [25] which results in a major improvement in tracking performance, despite the plant uncertainty and unmodeled dynamics.

As a solution to the modeling errors and uncertainties problems mentioned above, an adaptive FNN-based controller with sliding mode control (SMC) theory-based learning algorithm is proposed for the control of spherical rolling robots. FNNs are a fusion of the capability of fuzzy reasoning to handle uncertain information and the capability of ANNs to learn from input-output data sets. Thus, FNNs are preferable approaches in engineering fields [26]–[28]. Various learning algorithms have been proposed for FNNs. The gradient-based

algorithm works well when the system at hand has very slow variations in its dynamics. However, since the gradient-based algorithms (e.g. dynamic back propagation) include partial derivatives, the convergence speed may be slow. Especially when the search space is complex. Moreover, the tuning process can easily be trapped into a local minimum [29]. To alleviate the problems mentioned, the use of evolutionary approaches has been suggested [30]. However, the stability of such approaches is questionable and the optimal values for the stochastic operators are difficult to derive. Furthermore, the computational burden is very high. To overcome these issues, a novel SMC theory-based algorithm is proposed in this study. The system can provide adaptation to parameter variations, uncertainties and disturbances through the SMC theory-based online learning algorithm and also shows robust behaviour through the nature of SMC theory. SMC theory-based learning algorithms cannot only make the overall system more robust, but also ensure faster convergence than the traditional learning techniques in online tuning of ANNs and FNNs [31]. The main contributions of this study beyond the state of the art are a novel SMC theory-based online learning algorithm for FNNs and the use of the proposed control algorithm in the velocity control of spherical rolling robots. To the best knowledge of the authors, this is the first time such an approach is ever used for the velocity control of a spherical rolling robot.

This paper is organized as follows: The mathematical model of the spherical rolling robot is presented in Section II. In Section III, the adaptive neuro-fuzzy control approach is used to design an intelligent controller, and the basics of the SMC theory-based online learning algorithm are given. Simulation results and comparisons are presented in Section IV. Finally, a brief conclusion of the study is given in Section V.

II. THE MATHEMATICAL MODEL OF A SPHERICAL ROLLING ROBOT

The mathematical model of the spherical rolling robot is similar to the one presented by Kayacan et. al. [19]. The main difference is that viscous friction has been added into the equations of motion. A schematic illustration of the spherical rolling robot is given in Fig. 1.

A. Kinematic Model

Formulation of the kinematic and dynamic equations governing the motion of the spherical rolling mechanism with a pendulum is based on the following assumptions:

- 1) The sphere rolls over a perfectly horizontal surface without slipping.
- 2) The center of mass of the entire system is also the geometric center of the spherical shell.
- 3) The pendulum is in vertical downward position when the sphere is in static equilibrium.

A schematic illustration of the spherical rolling robot is given in Fig. 1. R_{f_0} , R_{f_1} and R_{f_2} denote respectively, the reference frames $O_0 - X_0 Y_0 Z_0$, $O_1 - X_1 Y_1 Z_1$ and $O_2 - X_2 Y_2 Z_2$. R_{f_0} represents the inertial reference frame fixed to the motion ground. R_{f_1} is a moving frame attached to the center of the sphere and allowed to translate only with respect to R_{f_0} . R_{f_2}

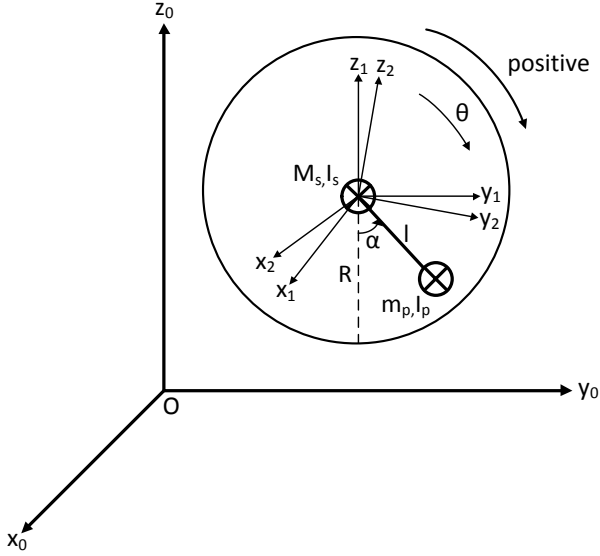


Fig. 1. Modeling of rolling motion about transversal axis for overall translation along $O - y$

is another moving frame attached also to the center of the sphere but allowed to rotate only with respect to R_{f1} . Relative angular positions between these frames can be described by several methods such as Euler angles, Tait-Bryan angles, Roll-Pitch-Yaw angles, etc.

The variables of the spherical rolling robot are represented in Table I.

TABLE I
NOMENCLATURE

θ	Rolling angle of the sphere around the x axis
α	Degree of freedom of pendulum
R	Radius of the sphere
l	Distance between the center of the sphere and the center of the pendulum
g	Gravitational acceleration
ζ	Damping coefficient

The angular velocity vector ω_s and linear velocity vector \mathbf{v}_s of the center of the sphere, the position vector \mathbf{r}_{p0} , the angular velocity vector ω_p and the linear velocity vector \mathbf{v}_p of the mass center of the pendulum are given by:

$$\omega_s = -\dot{\theta}\mathbf{i} \quad (1)$$

$$\mathbf{v}_s = -R\dot{\theta}\mathbf{j} \quad (2)$$

$$\mathbf{r}_p = l \sin(\alpha - \theta)\mathbf{j} - l \cos(\alpha - \theta)\mathbf{k} \quad (3)$$

$$\omega_p = (\dot{\alpha} - \dot{\theta})\mathbf{i} \quad (4)$$

$$\mathbf{v}_p = (-R\dot{\theta} + (\dot{\alpha} - \dot{\theta})l \cos(\alpha - \theta))\mathbf{j} + ((\dot{\alpha} - \dot{\theta})l \sin(\alpha - \theta))\mathbf{k} \quad (5)$$

where θ , α , R and l represent the rolling angle of the sphere around the x axis, the rotation of the pendulum around the x axis, the radius of the sphere and the distance between the center of the sphere and the center of the pendulum, respectively. Similarly, \mathbf{i} , \mathbf{j} and \mathbf{k} represent the unit vector on

the x, y and z axes, respectively. It is to be noted that x, y and z axes denote the inertial reference frame fixed to the motion ground.

B. Dynamic Model

Let E_k and E_p denote respectively the total kinetic and potential energy of the system. With M_s and m_p representing the masses of the sphere and pendulum, I_s and I_p representing the mass moment of inertia, and v_s , ω_s , v_p and ω_p representing the linear and angular velocities of the sphere and pendulum, r_{p-z} representing the vertical position of the mass center of the pendulum, the Lagrangian function L including only the terms due to rotations around the transversal axis is then written as follows:

$$\begin{aligned} L &= E_k - E_p \\ &= \frac{1}{2}M_s\|\mathbf{v}_s\|^2 + \frac{1}{2}I_s\|\omega_s\|^2 + \frac{1}{2}m_p\|\mathbf{v}_p\|^2 + \frac{1}{2}I_p\|\omega_p\|^2 \\ &\quad - m_p g r_{p-z} \\ &= \frac{1}{2}M_s(-R\dot{\theta})^2 + \frac{1}{2}I_s(-\dot{\theta})^2 + \frac{1}{2}I_p(\dot{\alpha} - \dot{\theta})^2 \\ &\quad + \frac{1}{2}m_p\left((-R\dot{\theta} + (\dot{\alpha} - \dot{\theta})l \cos(\alpha - \theta))^2\right. \\ &\quad \left.+ ((\dot{\alpha} - \dot{\theta})l \sin(\alpha - \theta))^2\right) - m_p g l \cos(\alpha - \theta) \quad (6) \end{aligned}$$

It is assumed that the viscous friction operates between the sphere and the surface. The loss due to the viscous friction is written in an energy dissipation function that depends on the velocities of the system and the damping constant:

$$S = \frac{1}{2}\zeta\dot{q}_i^2 = \frac{1}{2}\zeta(\dot{\theta}^2 + \dot{\alpha}^2) \quad (7)$$

For translation along $O - y$, the Euler-Lagrange equations of the system are written as follows:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} + \frac{\partial S}{\partial \dot{q}_i} = Q_i \quad (8)$$

where $q_1 = \theta$ and $q_2 = \alpha$ are the generalized coordinates. In fact, when the pendulum is rotated through an input torque, a reaction torque about the shaft occurs in the opposite direction [9]. $Q_1 = Q_2 = \tau$ represents the input torque to rotate the pendulum.

$$\begin{aligned} Q_\theta &= \tau \\ Q_\alpha &= \tau \end{aligned} \quad (9)$$

As the equations of motion of a mechanical system can be written as follows:

$$M(q(t))\ddot{q}(t) + C(q(t), \dot{q}(t)) + G(q(t)) = u(t) \quad (10)$$

The equations of motion can be finally written in the following matrix form:

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\alpha} \end{bmatrix} + \begin{bmatrix} C_{11} \\ C_{21} \end{bmatrix} + \begin{bmatrix} G_{11} \\ G_{21} \end{bmatrix} = \begin{bmatrix} \tau \\ \tau \end{bmatrix} \quad (11)$$

where

$$\begin{aligned}
M_{11} &= M_s R^2 + m_p R^2 + m_p l^2 + I_s + I_p \\
&\quad + 2m_p R l \cos(\alpha - \theta) \\
M_{12} &= M_{21} = -m_p l^2 - I_p - m_p R l \cos(\alpha - \theta) \\
M_{22} &= m_p l^2 + I_p \\
C_{11} &= m_p R l \sin(\alpha - \theta)(\dot{\alpha} - \dot{\theta})^2 + \zeta \dot{\theta} \\
C_{21} &= \zeta \dot{\alpha} \\
G_{11} &= -G_{21} = -m_p g l \sin(\alpha - \theta)
\end{aligned}$$

In this study, velocity control of the spherical rolling robot is studied by using the above equations of motion. If rolling motion over curvilinear trajectories is desired, the equations of motion presented in [19] can be used. Different from the approach in [19], a viscous friction term is added in this study.

III. THE ADAPTIVE NEURO-FUZZY CONTROL APPROACH

A. The Control Scheme and the Adaptive Neuro-Fuzzy Inference System

In real-time applications, structural and parametric uncertainties such as unmodeled dynamics and physical disturbances cause unwanted effects on the system behaviour. Like tire dynamics, the spherical rolling robot dynamics are affected by the slip ratio and variations of the surface conditions. Thus, spherical rolling robots should have an adaptive intelligent controller with respect to the high uncertainty and the variation in the surface properties. While conventional controllers struggle with these problems, neuro-fuzzy control can overcome these limitations and provide higher robustness.

In the presented approach, a conventional PD or PID controller and a neuro-fuzzy controller are working in parallel. The conventional controller is used to guarantee the global asymptotic stability of the system in compact space. The PD control law is written as follows:

$$\tau_c = k_p e + k_d \dot{e} \quad (12)$$

where $e = \theta_d - \theta$ is the feedback error, θ_d is the desired position value, k_p and k_d are the controller gains.

The proposed control scheme for the PD controller case, the PID controller case and the structure of the proposed FNN are illustrated in Fig. 2, Fig. 3 and Fig. 4, respectively. The terms of α and β are the coefficients of the PI controller. As can be seen from the proposed control scheme, a FNN which has two inputs $x_1(t) = e(t)$, $x_2(t) = \dot{e}(t)$ and one output f_{ij} , is applied as a feedback controller.

The fuzzy *If-Then* rule of a zeroth-order Takagi-Sugeno-Kang (TSK) model with two input variables where the consequent part is a linear function of the input variables can be defined as follows:

$$R_{ij} : \text{ If } x_1 \text{ is } A_i \text{ and } x_2 \text{ is } B_j, \text{ Then } f_{ij} = d_{ij} \quad (13)$$

where x_1 and x_2 are the inputs of the FNN, A_i and B_j are fuzzy sets corresponding to the input 1 and input 2, respectively. The zeroth-order function, f_{ij} is the consequent part of the rules where I ($i = 1, \dots, I$) and J ($j = 1, \dots, J$)

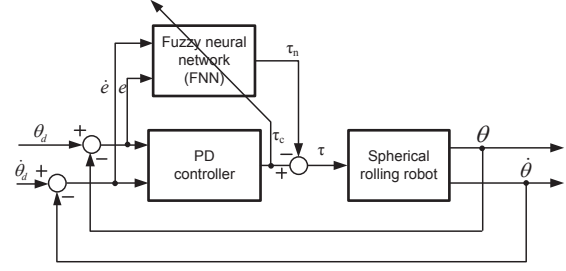


Fig. 2. Block diagram of the proposed control scheme for the PD controller case

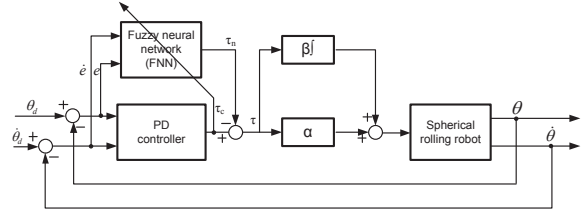


Fig. 3. Block diagram of the proposed control scheme for the PID controller case

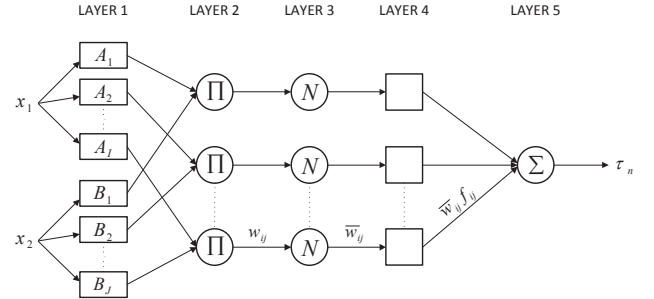


Fig. 4. The FNN

being the number of membership functions used for the input 1 and input 2, respectively.

The following are the basic explanations for Fig. 4:

•Layer 1: The first layer is called input later. This layer maps the crisp inputs x_1 and x_2 into fuzzified values using membership functions. Every node has a node function described as follows:

$$\begin{aligned}
O_i^1 &= \mu_{A_i}(x_1) \\
O_j^1 &= \mu_{B_j}(x_2)
\end{aligned} \quad (14)$$

where $\mu_{A_i}(x_1)$ and $\mu_{B_j}(x_2)$ are the membership values for the inputs 1 and 2, respectively. In the case of Gaussian membership functions, $\mu_{A_i}(x_1)$ and $\mu_{B_j}(x_2)$ are written as follows:

$$\mu_{A_i}(x_1) = \exp\left(-\left(\frac{x_1 - c_{A_i}}{\sigma_{A_i}}\right)^2\right)$$

$$\mu_{B_j}(x_2) = \exp\left(-\left(\frac{x_2 - c_{B_j}}{\sigma_{B_j}}\right)^2\right) \quad (15)$$

where c and σ are respectively the mean and the standard deviation of the membership functions. These parameters $c, \sigma > 0$ are the tunable parameters of the neuro-fuzzy structure.

•Layer 2: The strength of the rule is obtained in this layer. The output of each node, which calculates the firing strength w_{ij} of a rule, is multiplied with all incoming signals. The node function of this layer is written as follows:

$$O_{ij}^2 = w_{ij} = \mu_{A_i}(x_1)\mu_{B_j}(x_2) \quad (16)$$

The previous equation (16) is rewritten considering (15) as follows:

$$O_{ij}^2 = w_{ij} = \exp\left(-\left(\frac{x_1 - c_{A_i}}{\sigma_{A_i}}\right)^2 - \left(\frac{x_2 - c_{B_j}}{\sigma_{B_j}}\right)^2\right) \quad (17)$$

•Layer 3: The normalized firing strength of each node is written as follows:

$$O_{ij}^3 = \bar{w}_{ij} = \frac{w_{ij}}{\sum_{i=1}^I \sum_{j=1}^J w_{ij}} \quad (18)$$

•Layer 4: Each node in this layer is an adaptive node with node function written by the following equation:

$$O_{ij}^4 = \bar{w}_{ij} f_{ij} \quad (19)$$

•Layer 5: There is one node where all incoming signals are summed in this layer. The output signal of the neuro-fuzzy network τ_n is calculated by the function described as follows:

$$O^5 = \tau_n(t) = \sum_{i=1}^I \sum_{j=1}^J \bar{w}_{ij} f_{ij} = \sum_{i=1}^I \sum_{j=1}^J \bar{w}_{ij} d_{ij} \quad (20)$$

The control input fed to the system is the overall torque τ determined as follows:

$$\tau = \tau_c - \tau_n \quad (21)$$

where τ_c and τ_n are the input torques generated by the PD controller and the neuro-fuzzy feedback controller, respectively.

B. The Sliding Mode Learning Algorithm

The input signals and their time derivatives $x_1(t), x_2(t), \dot{x}_1(t), \dot{x}_2(t)$ cannot have infinite values. Thus, they are bounded as follows:

$$\begin{aligned} |x_1(t)| &\leq B_x, & |x_2(t)| &\leq B_x & \forall t \\ |\dot{x}_1(t)| &\leq B_{\dot{x}}, & |\dot{x}_2(t)| &\leq B_{\dot{x}} & \forall t \end{aligned} \quad (22)$$

Similarly, the input torque to the system and its time derivative are bounded:

$$|\tau(t)| \leq B_\tau, \quad |\dot{\tau}(t)| \leq B_{\dot{\tau}} \quad \forall t \quad (23)$$

The output of the PD controller $\tau_c(t)$ is defined as a time-varying sliding surface by using the principles of SMC theory [32]:

$$S_c(\tau_n, \tau) = \tau_c(t) = \tau_n(t) + \tau(t) = 0 \quad (24)$$

The sliding surface $S_p(e, \dot{e})$ is defined as follows:

$$S_p(e, \dot{e}) = \dot{e} + \lambda e \quad (25)$$

where λ is a constant determining the slope of the sliding surface.

Definition: A sliding motion exists on the sliding surface $S_c(\tau_n, \tau) = \tau_c(t) = 0$ after time t_h , if the condition $S_c(t)\dot{S}_c(t) = \tau_c(t)\dot{\tau}_c(t) < 0$ is satisfied for all t in some nontrivial semi-open subinterval of time of the form $[t, t_h) \subset (-\infty, t_h)$.

Since it is desired to design a dynamical feedback adaptation mechanism, or online learning algorithm for the FNN parameters, the sliding mode condition of the above definition is applied.

Theorem 1. *The adaptation laws for the parameters of the proposed FNN are given by the following equations:*

$$\dot{c}_{A_i} = \dot{x}_1 + s_{A_i} \alpha \text{sgn}(\tau_c) \quad (26)$$

$$\dot{c}_{B_j} = \dot{x}_2 + s_{B_j} \alpha \text{sgn}(\tau_c) \quad (27)$$

$$\dot{\sigma}_{A_i} = -\left(\sigma_{A_i} + \frac{(\sigma_{A_i})^3}{s_{A_i}^T s_{A_i}}\right) \alpha \text{sgn}(\tau_c) \quad (28)$$

$$\dot{\sigma}_{B_j} = -\left(\sigma_{B_j} + \frac{(\sigma_{B_j})^3}{s_{B_j}^T s_{B_j}}\right) \alpha \text{sgn}(\tau_c) \quad (29)$$

$$\dot{f}_{ij} = -\frac{\bar{W}_{ij}}{W^T W} \alpha \text{sgn}(\tau_c) \quad (30)$$

The vectors in (26)-(29) are defined as follows:

$$s_{A_i} = x_1 - c_{A_i} \text{ and } s_A = [s_{A_1} \ s_{A_2} \ \dots \ s_{A_I}]^T \quad (31)$$

$$s_{B_j} = x_2 - c_{B_j} \text{ and } s_B = [s_{B_1} \ s_{B_2} \ \dots \ s_{B_J}]^T \quad (32)$$

where α is the learning rate which is a sufficiently large positive design constant which satisfies the following inequality:

$$B_{\dot{\tau}} < \alpha \quad (33)$$

This ensures that for a given arbitrary initial condition $\tau_c(0)$, the learning error $\tau_c(t)$ will converge to zero within finite time t_h .

Proof: The reader is referred to Appendix A.

If λ is considered $\lambda = \frac{k_p}{k_d}$, the relation between the sliding line S_p and the zero adaptive learning error level S_c can be written as follows:

$$S_c = \tau_c = k_d \dot{e} + k_p e = k_d \left(\dot{e} + \frac{k_p}{k_d} e\right) = k_d S_p \quad (34)$$

The tracking performance of the velocity control system in the spherical rolling robot is analyzed by the following Lyapunov function candidate:

$$V_p = \frac{1}{2} S_p^2 \quad (35)$$

Theorem 2. *If the adaptation strategy for the adjustable parameters of the FNN is chosen as in (26)-(30), then the*

negative definiteness of the time derivative of the Lyapunov function in (35) is ensured.

Proof: The reader is referred to Appendix B.

IV. SIMULATION RESULTS AND DISCUSSION

The numerical values used in [4], [6], [9] are considered for the numerical values in this study which are $M_s = 3 \text{ kg}$, $m_p = 2 \text{ kg}$, $R = 0.2 \text{ m}$, $l = 0.075 \text{ m}$ and $g = 9.81 \text{ m/s}^2$. The damping coefficient, ζ , in the equations of motion is set to 0.2, and the sampling period of the simulations is set to 0.001 s. The number of membership functions for the input 1 and input 2 is chosen as $I = J = 3$ for all the simulations.

SMC theory suffers from high-frequency oscillations called *chattering* because of the input control law. Several approaches have been suggested to get rid of this problem. In this paper, the *sgn* function in (26)-(30) is replaced by the following equation to decrease the chattering effect:

$$\text{sgn}(\tau_c) := \frac{\tau_c}{|\tau_c| + \delta} \quad (36)$$

where $\delta = 0.05$.

A. Case 1: PD controller

The coefficients of the PD controller are set to $k_p = 1$ and $k_d = 0.05$ by trial-and-error method. The following reference signal has been applied to the system (37):

$$\text{Reference}(t) = \begin{cases} 1 \text{ rad/s} & \text{if } 0 < t \leq 5 \\ 2 \text{ rad/s} & \text{if } 5 < t \leq 10 \\ 1.5 \text{ rad/s} & \text{if } 10 < t \leq 15 \end{cases} \quad (37)$$

Figures 5-6 show the velocity response and the error of the system for the PD controller working alone and the PD controller working in parallel with the FNN, respectively. As can be seen from Fig. 5, the PD controller cannot eliminate the steady state error for a time-varying step input. The system does not have any steady state error for the case of the FNN working in parallel with a PD controller.

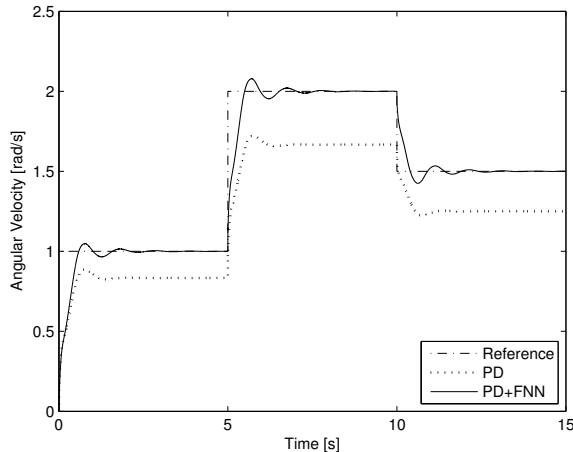


Fig. 5. The velocity response of the system for PD and PD+FNN controllers

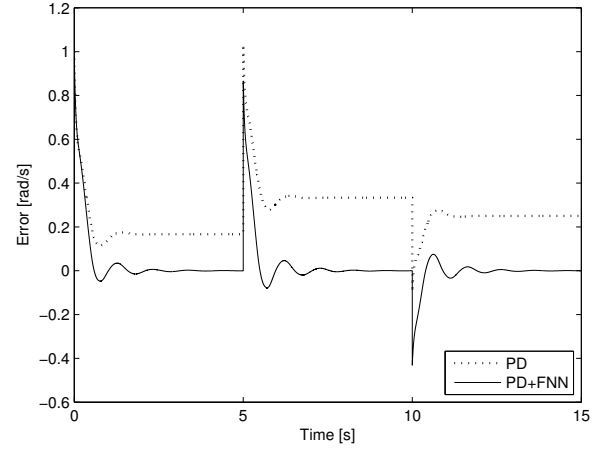


Fig. 6. The error of the system for PD and PD+FNN controllers

B. Case 2: PID controller

The coefficients of the PID controller are set to $k_p = 1$, $k_d = 0.05$, $\alpha = 1$ and $\beta = 2$ by trial-and-error method. The reference signal for this simulation is the same the one (See 37) in Case 1.

In Figure 7 the responses of the PID controller and the combination of the FNN in parallel with this PID controller are illustrated. The FNN learns the system dynamics after a finite time duration, and results in a smaller rise time, overshoot and settling time. From Fig. 7 and 8 it can be concluded that the control scheme with the proposed SMC theory-based learning algorithm is able to not only eliminate the steady state error (shown in Case 1) but also to improve the transient response performance of the system (shown in Case 2). Figure 8 shows the zoomed view of Fig. 7 between $0^{\text{th}} - 3^{\text{rd}}$, $5^{\text{th}} - 8^{\text{th}}$ and $10^{\text{th}} - 13^{\text{th}}$.

To test the robustness of this approach, the damping coefficient is set to 0.5 with a noise level $SNR = 20\text{dB}$. In Figure 9 the responses of the PID controller and the combination of the

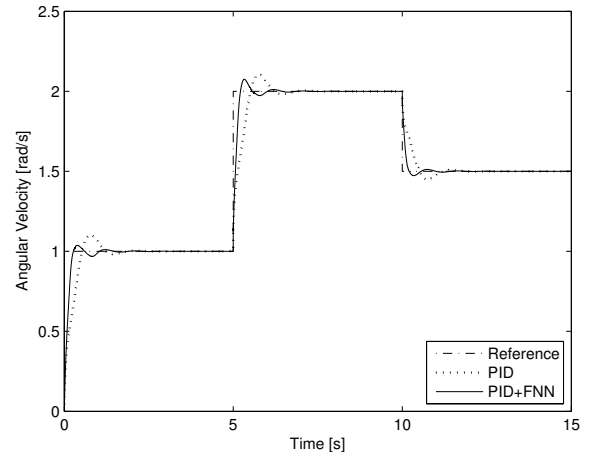


Fig. 7. The velocity response of the system for PID and PID+FNN controllers

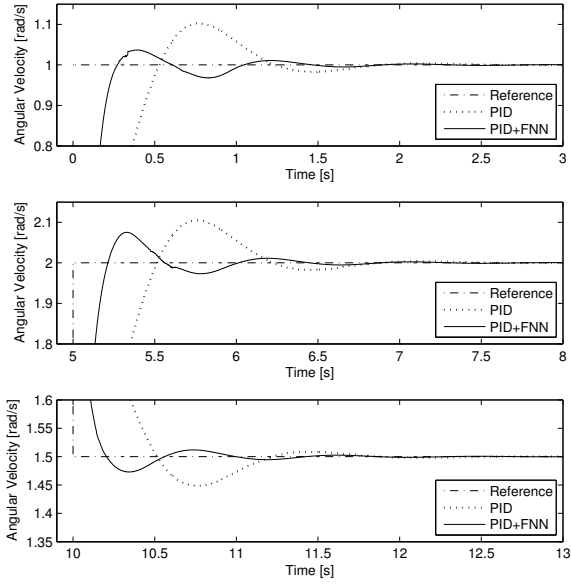


Fig. 8. Zoomed view of Figure 7

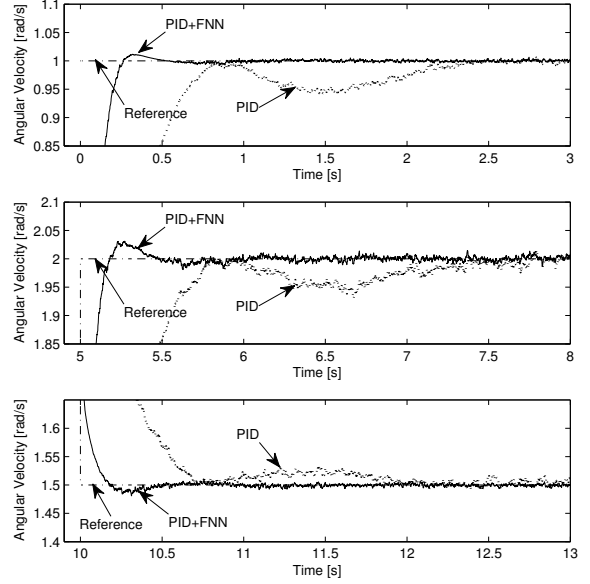


Fig. 10. Zoomed view of Figure 9

FNN working in parallel with the PID controller are illustrated. In Fig. 10 close ups of Fig. 9 are shown. It can be observed that the adaptive neuro-fuzzy control approach is much more robust to the uncertainties and gives more satisfactory results regarding smaller rise time and settling time than the PID controller working alone.

In a next step, the following variation of the damping coefficient over time was applied to the system to test the robustness of the controllers:

$$\zeta(t) = \begin{cases} 0.2 \text{ rad/s} & \text{if } 0 < t \leq 5 \\ 0.5 \text{ rad/s} & \text{if } 5 < t \leq 10 \\ 0.8 \text{ rad/s} & \text{if } 10 < t \leq 15 \end{cases} \quad (38)$$

In Figure 11 the velocity responses of a conventional PID controller and the combination of the FNN working in parallel with this PID controller are shown, while close ups are shown in Fig. 12. As can be seen from Fig. 11 - 12, the FNN can adapt its parameters when the coefficient of viscous friction changes suddenly. For the case when the coefficient of the viscous friction changes, while the settling time for the PID controller working alone is approximately 2.5 seconds, it is less than 1 second for the FNN working in parallel with a PID controller. From these simulations it can be concluded that the adaptive neuro-fuzzy control approach is much more robust to the parameter variations compared to the case of the PID controller working alone.

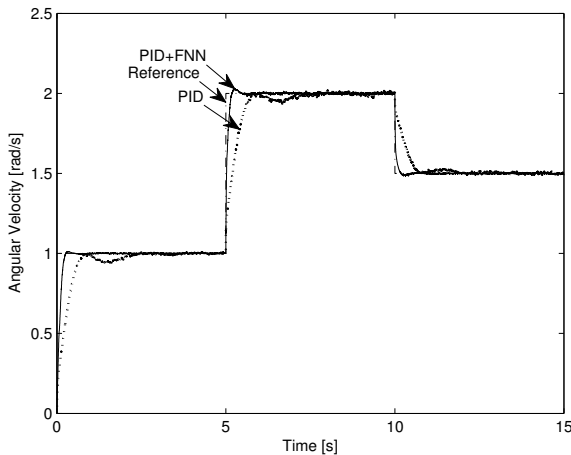


Fig. 9. The velocity response of the system for PID and PID+FNN controllers when the damping coefficient is set to 0.5 with a noise level $SNR = 20dB$

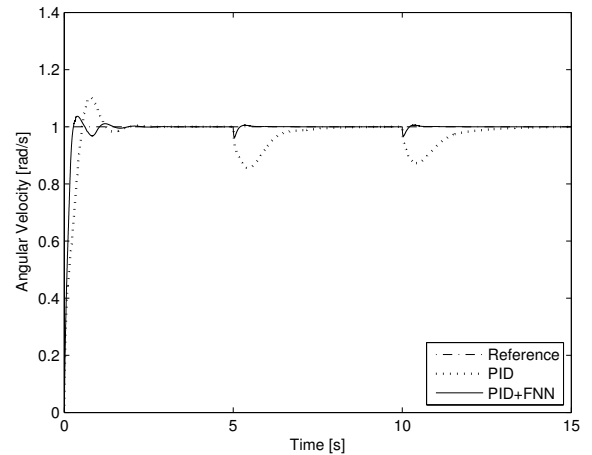


Fig. 11. The velocity response of the system for PID and PID+FNN controllers when the damping coefficient is set to 0.2, 0.5 and 0.8 at 0^{th} second, 5^{th} second and 10^{th} second, respectively

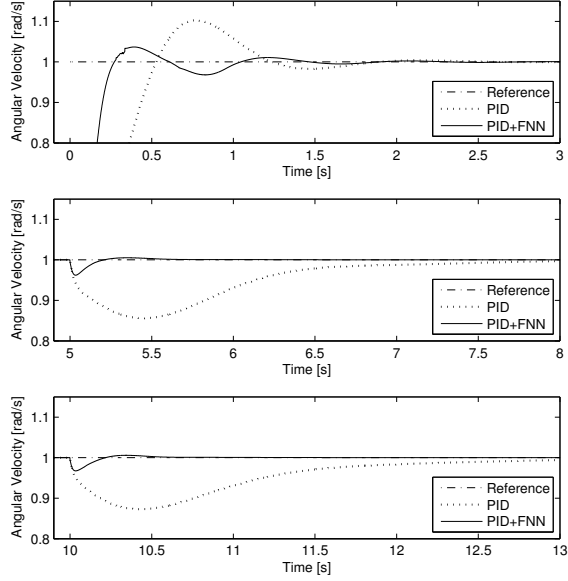


Fig. 12. Zoomed view of 11

In Figure 13 the control signals coming from the PID controller alone and the FNN working in parallel with the PID controller are illustrated for the case when the damping coefficient is set to 0.2, 0.5 and 0.8 at 0^{th} second, 5^{th} second and 10^{th} second, respectively. In order not to confuse the reader, since the control input to the system is equal to $\tau_c - \tau_n$ when the PID controller works in parallel with the FNN, τ_n is multiplied by -1 in this figure. As can be seen from Fig. 13, the output of the PID controller is approximately zero while the output of the FNN changes as the damping coefficient changes as well. In other words, the FNN takes the responsibility for controlling the system after a finite time duration.

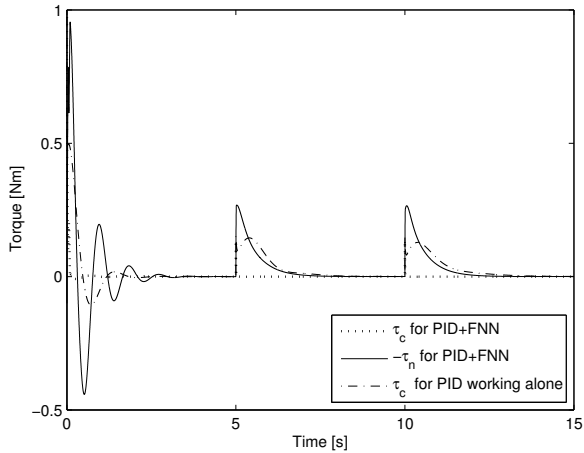


Fig. 13. The control signals coming from the PID controller and the FNN when the damping coefficient is set to 0.2, 0.5 and 0.8 at 0^{th} second, 5^{th} second and 10^{th} second, respectively

V. CONCLUSION

In this study, an adaptive neuro-fuzzy controller with SMC theory-based online learning has been elaborated for velocity control of a spherical rolling robot subject to parameter variations and uncertainties in its dynamics. The simulation studies show that the proposed adaptive neuro-fuzzy control scheme (a conventional controller working in parallel with the FNN) results in a better performance and higher robustness than when compared to the conventional stand-alone controller. The proposed control algorithm is able to not only eliminate the steady state error in the case of conventional PD stand-alone controller but also improve the transient response performance of the system in the case of conventional PID stand-alone controller. Thanks to the SMC theory-based online learning algorithm the parameters of the controller are automatically adapted to cope with the parameter variations and uncertainties. In addition to its robustness, the control approach with the proposed learning algorithm is computationally simple, especially when compared to the gradient descent and the evolutionary algorithms.

APPENDIX A PROOF OF THEOREM 1

The time derivative of (15) is written as follows:

$$\begin{aligned}\dot{\mu}_{A_i}(x_1) &= -2N_{A_i}(N_{A_i})' \mu_{A_i}(x_1) \\ \dot{\mu}_{B_j}(x_2) &= -2N_{B_j}(N_{B_j})' \mu_{B_j}(x_2)\end{aligned}\quad (39)$$

where

$$\begin{aligned}N_{A_i} &= \frac{x_1 - c_{A_i}}{\sigma_{A_i}} \\ N_{B_j} &= \frac{x_2 - c_{B_j}}{\sigma_{B_j}}\end{aligned}\quad (40)$$

The time derivative of (40) is written as follows:

$$\begin{aligned}\dot{N}_{A_i} &= \frac{(\dot{x}_1 - \dot{c}_{A_i})\sigma_{A_i} - (x_1 - c_{A_i})\dot{\sigma}_{A_i}}{\sigma_{A_i}^2} \\ \dot{N}_{B_j} &= \frac{(\dot{x}_2 - \dot{c}_{B_j})\sigma_{B_j} - (x_2 - c_{B_j})\dot{\sigma}_{B_j}}{\sigma_{B_j}^2}\end{aligned}\quad (41)$$

Combination of (40) and (41) gives:

$$N_{A_i}\dot{N}_{A_i} = N_{B_j}\dot{N}_{B_j} = \alpha \text{sgn}(\tau_c)\quad (42)$$

The time derivative of (18) is written as follows:

$$\dot{\bar{w}}_{ij} = -\bar{w}_{ij}\dot{k}_{ij} + \bar{w}_{ij} \sum_{i=1}^I \sum_{j=1}^J \bar{w}_{ij}\dot{k}_{ij}\quad (43)$$

where

$$\dot{k}_{ij} = 2 \left(N_{A_i}(N_{A_i})' + N_{B_j}(N_{B_j})' \right)\quad (44)$$

The stability of the proposed control approach is investigated by using the following Lyapunov function:

$$V = \frac{1}{2} (\tau_c)^2\quad (45)$$

The time derivative of (45) is written as follows:

$$\dot{V} = \dot{\tau}_c \tau_c = \tau_c (\dot{\tau}_n + \dot{\tau})\quad (46)$$

where

$$\dot{\tau}_n = \sum_{i=1}^I \sum_{j=1}^J (\dot{f}_{ij} \bar{w}_{ij} + f_{ij} \dot{\bar{w}}_{ij}) \quad (47)$$

Substitution of (47), (43), (44), (42) and (30) into (46), gives:

$$\begin{aligned} \dot{V} &= \tau_c \left[\sum_{i=1}^I \sum_{j=1}^J \left(\dot{f}_{ij} \bar{w}_{ij} + f_{ij} (-\bar{w}_{ij} \dot{k}_{ij} \right. \right. \\ &\quad \left. \left. + \bar{w}_{ij} \sum_{i=1}^I \sum_{j=1}^J \bar{w}_{ij} \dot{k}_{ij} \right) \right) + \dot{\tau} \Big] \\ &= \tau_c \left[\sum_{i=1}^I \sum_{j=1}^J \left(\dot{f}_{ij} \bar{w}_{ij} \right. \right. \\ &\quad \left. \left. + f_{ij} \left(-2\bar{w}_{ij} (N_{A_i} (N_{A_i})' + N_{B_j} (N_{B_j})') \right. \right. \right. \\ &\quad \left. \left. \left. + 2\bar{w}_{ij} \sum_{i=1}^I \sum_{j=1}^J \bar{w}_{ij} (N_{A_i} (N_{A_i})' + N_{B_j} (N_{B_j})') \right) \right) \right) + \dot{\tau} \Big] \\ &= \tau_c \left[\sum_{i=1}^I \sum_{j=1}^J \left(\dot{f}_{ij} \bar{w}_{ij} + f_{ij} \left(-4\bar{w}_{ij} \alpha \text{sgn}(\tau_c) \right. \right. \right. \\ &\quad \left. \left. \left. + 4\bar{w}_{ij} \sum_{i=1}^I \sum_{j=1}^J \bar{w}_{ij} \alpha \text{sgn}(\tau_c) \right) \right) \right) + \dot{\tau} \Big] \\ &= \tau_c \left[\sum_{i=1}^I \sum_{j=1}^J \dot{f}_{ij} \bar{w}_{ij} + \dot{\tau} \right] \\ &= \tau_c \left[-\alpha \text{sgn}(\tau_c) + \dot{\tau} \right] \\ &= \left[-\alpha \mid \tau_c \mid + \dot{\tau} \mid \tau_c \mid \right] \quad (48) \end{aligned}$$

The time derivative \dot{V} of the Lyapunov function V must be smaller than zero to satisfy the stability of the learning.

$$\dot{V} = \left[-\alpha \mid \tau_c \mid + \dot{\tau} \mid \tau_c \mid \right] < 0 \quad (49)$$

When $\dot{\tau}$ reaches its maximal value $B_{\dot{\tau}}$, (49) can be rewritten as follows:

$$\dot{V} = \left[-\alpha \mid \tau_c \mid + B_{\dot{\tau}} \mid \tau_c \mid \right] < 0 \quad \text{if} \quad B_{\dot{\tau}} < \alpha \quad (50)$$

APPENDIX B PROOF OF THEOREM 2

The time derivative of the Lyapunov function in (35) is written as follows:

$$\begin{aligned} \dot{V}_p &= \dot{S}_p S_p = \frac{1}{k_d^2} \dot{S}_c S_c = \frac{1}{k_d^2} \dot{\tau}_c \tau_c \\ &= \frac{1}{k_d^2} \dot{V} < 0, \quad \forall S_c, S_p \neq 0 \quad (51) \end{aligned}$$

REFERENCES

- [1] J. Alves and J. Dias, "Design and control of a spherical mobile robot," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 217, pp. 457–467, 2003.
- [2] C. Camicia, F. Conticelli, and A. Bicchi, "Nonholonomic kinematics and dynamics of the sphericle," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, Takamatsu, Japan, 2000, pp. 805 – 810.
- [3] A. Halme, T. Schonberg, and Y. Wang, "Motion control of a spherical mobile robot," in *Proceedings of fourth International Workshop on Advanced Motion Control*, vol. 1, Albuquerque, NM, USA, 1996, pp. 259–264.
- [4] A. A. H. Javadi and P. Mojabi, "Introducing glory: A novel strategy for an omnidirectional spherical rolling robot," *Journal of Dynamic Systems, Measurement, and Control*, vol. 126, pp. 678–683, 2004.
- [5] R. Mukherjee, M. A. Minor, and J. T. Pukrushpan, "Motion planning for a spherical mobile robot: Revisiting the classical ball-plate problem," *ASME Journal of Dynamic Systems Measurement and Control*, vol. 124, pp. 502–511, 2002.
- [6] V. A. Joshi and R. N. Banavar, "Motion analysis of a spherical mobile robot," *Robotica*, vol. 27, pp. 343 – 353, 2009.
- [7] V. A. Joshi, R. N. Banavar, and R. Hippalgaonkar, "Design and analysis of a spherical mobile robot," *Mechanism and Machine Theory*, 2009.
- [8] S. Bhattacharya and S. K. Agrawal, "Spherical rolling robot: a design and motion planning studies," *IEEE Transactions on Robotics and Automation*, vol. 16, pp. 835–839, 2000.
- [9] Y. Ming, D. Zongquan, Y. Xinyi, and Y. Weizhen, "Introducing HIT spherical robot: Dynamic modeling and analysis based on decoupled subsystem," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, Harbin, China, 2006, p. 181–186.
- [10] Q. Jia, H. Sun, and D. Liu, "Analysis of actuation for a spherical robot," in *Proceedings of the 2008 IEEE Conference on Robotics, Automation and Mechatronics*, Chengdu, 2008, pp. 266–271.
- [11] D. Liu, H. Sun, and Q. Jia, "Stabilization and path following of a spherical robot," in *Proceedings of the 2008 IEEE Conference on Robotics, Automation and Mechatronics*, Chengdu, 2008, pp. 676 – 682.
- [12] D. Liu, H. Sun, Q. Jia, and L. Wang, "Motion control of a spherical mobile robot by feedback linearization," in *Proceedings of the 7th World Congress on Intelligent Control and Automation*, Chongqing, China, 2008, pp. 965–970.
- [13] D. Liu, H. Sun, and Q. Jia, "A family of spherical mobile robot: Driving ahead motion control by feedback linearization," in *Proceedings of 2nd International Symposium on Systems and Control in Aerospace and Astronautics*, Shenzhen, China, 2008.
- [14] F. Michaud, J.-F. Laplante, H. Larouche, A. Duquette, S. Caron, D. Letourneau, and P. Masson, "Autonomous spherical mobile robot for child-development studies," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 35, pp. 471 – 480, 2005.
- [15] F. Michaud and S. Caron, "Roball, the rolling robot," *Autonomous Robots*, vol. 12, pp. 211 – 222, 2002.
- [16] J.-F. Laplante, P. Masson, and F. Michaud, "Analytical longitudinal and lateral models of a spherical rolling robot," Laborius, Department of Electrical and Computer Engineering, Universite de Shebroke, Canada, Tech. Rep., 2002.
- [17] H. Alizadeh and M. Mahjoob, "Effect of incremental driving motion on a vision-based path planning of a spherical robot," in *Proceedings of the 2009 Second International Conference on Computer and Electrical Engineering*, vol. 1, Dubai, United Arab Emirates, 2009, pp. 299 – 303.
- [18] T. Otani, T. Urakuboi, S. Maekawa, H. Tamaki, and Y. Tada, "Position and attitude control of a spherical rolling robot equipped with a gyro," in *Proceedings of the 9th IEEE International Workshop on Advanced Motion Control*, Istanbul, 2006, pp. 416 – 421.
- [19] E. Kayacan, Z. Y. Bayraktaroglu, and W. Saeys, "Modeling and control of a spherical rolling robot: A decoupled dynamics approach," *Robotica*.
- [20] H. Gomi and M. Kawato, "Neural network control for a closed-loop system using feedback-error-learning," *Neural Networks*, vol. 6, no. 7, pp. 933 – 946, 1993. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S089360800980004X>
- [21] D. Munoz and D. Sbarbaro, "An adaptive sliding-mode controller for discrete nonlinear systems," *IEEE Transactions on Industrial Electronics*, vol. 47, pp. 574 – 581, 2000.
- [22] S. J. Huang, K. S. Huang, and K. C. Chiou, "Development and application of a novel radial basis function sliding mode controller," *Mechatronic*, vol. 13, p. 313–329, 2003.

- [23] T. Das and I. N. Kar, "Design and implementation of an adaptive fuzzy logic-based controller for wheeled mobile robots," *IEEE Transactions on Control Systems Technology*, vol. 14, pp. 501 – 510, 2006.
- [24] B. S. Park, S. J. Yoo, J. B. Park, and Y. H. Choi, "Adaptive neural sliding mode control of nonholonomic wheeled mobile robots with model uncertainty," *IEEE Transactions on Control Systems Technology*, vol. 17, pp. 207 – 214, 2009.
- [25] M. Bugeja, S. Fabri, and L. Camilleri, "Dual adaptive dynamic control of mobile robots using neural networks," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 39, no. 1, pp. 129 –141, feb. 2009.
- [26] A. V. Topalov, Y. Oniz, E. Kayacan, and O. Kaynak, "Neuro-fuzzy control of antilock braking system using sliding mode incremental learning algorithm," *Neurocomputing*, vol. 74, no. 11, pp. 1883 – 1893, 2011.
- [27] E. Kayacan, Y. Oniz, A. C. Aras, O. Kaynak, and R. Abiyev, "A servo system control with time-varying and nonlinear load conditions using type-2 fuzzy neural system," *Applied Soft Computing*, vol. In Press, Corrected Proof, pp. –, 2011. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S1568494611001049>
- [28] M. Ahmadih Khanesar, E. Kayacan, M. Teshnehlab, and O. Kaynak, "Extended kalman filter based learning algorithm for type-2 fuzzy logic systems and its experimental evaluation," *Industrial Electronics, IEEE Transactions on*, vol. PP, no. 99, p. 1, 2011.
- [29] A. Topalov and O. Kaynak, "Online learning in adaptive neurocontrol schemes with a sliding mode algorithm," *IEEE Transactions on Systems, Man, and Cybernetics Part B: Cybernetics*, vol. 31, no. 3, pp. 445 –450, december 2001.
- [30] Y. Zhou and M. Er, "An evolutionary approach toward dynamic self-generated fuzzy inference systems," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 38, no. 4, pp. 963 –969, aug. 2008.
- [31] M. O. Efe, O. Kaynak, and X. Yu, "Sliding mode control of a three degrees of freedom anthropoid robot by driving the controller parameters to an equivalent regime," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 122, no. 4, pp. 632 –640, December 2000.
- [32] V. I. Utkin, *Sliding Modes in Control Optimization*. Springer-Verlag, 1992.



nonfood materials, from the field to the end user. He is author or co-author of more than 40 papers.

Herman Ramon graduated as an agricultural engineer from Gent University. In 1993 he obtained a Ph.D. in applied biological sciences at the Katholieke Universiteit Leuven. He is currently Professor at the Faculty of Agricultural and Applied Biological Sciences of the Katholieke Universiteit Leuven, lecturing on agricultural machinery and mechatronic systems for agricultural machinery. He has a strong research interest in precision technologies and advanced mechatronic systems for processes involved in the production chain of food and



Erkan Kayacan (S'12) was born in Istanbul, Turkey, on April 17, 1985. He received the B.Sc. and the M.Sc. degrees in mechanical engineering from Istanbul Technical University, Istanbul, in 2008 and 2010, respectively. He is currently working toward the Ph.D. degree in the Department of Biosystems, KU Leuven.

He is currently a Research Assistant in KU Leuven at the Department of Biosystems (BIOSYST) in the Division of Mechatronics, Biostatistics and Sensors (MeBioS). His research interests include

robotics, nonlinear control, intelligent control, system identification, fuzzy theory, and grey system theory.



Erdal Kayacan (S'06-SM'12) was born in Istanbul, Turkey on January 7, 1980. He received the B.Sc. degree in electrical engineering from Istanbul Technical University, Istanbul, Turkey, in 2003. He received the M.Sc. and Ph.D. degrees in systems and control engineering and electrical and electronics engineering from Bogazici University, Istanbul, Turkey, in 2006 and 2011, respectively. He is currently a post-doctoral researcher in KU Leuven at the department of biosystems (BIOSYST) in the division of mechatronics, biostatistics and sensors (MeBioS).

His research interests include large scale systems, soft computing, intelligent control, fuzzy logic theory.

Dr. Kayacan is active in IEEE CIS Student Activities Subcommittee, IEEE CIS Social Media Subcommittee and IEEE SMC Technical Committee on Grey Systems.



and member of the editorial board of Biosystems Engineering.

Wouter Saey is currently Assistant Professor in Biosystems Engineering at the Department of Biosystems of the University of Leuven in Belgium. He obtained his Ph.D. at the same institute and was a visiting postdoc at the School for Chemical Engineering and Advanced Materials of the University of Newcastle upon Tyne, UK and at the Norwegian Food Research Institute - Nofima Mat in Norway. His main research interests are optical sensing, process monitoring and control with applications in food and agriculture. He is author of 50 articles (ISI)