

Towards Agrobots: Trajectory Control of an Autonomous Tractor Using Type-2 Fuzzy Logic Controllers

Erdal Kayacan, *Senior Member, IEEE*, Erkan Kayacan, *Student Member, IEEE*, Herman Ramon, Okayay Kaynak, *Fellow, IEEE*, and Wouter Saeys

Abstract—Provision of some autonomous functions to an agricultural vehicle would lighten the job of the operator but in doing so, the accuracy should not be lost to still obtain an optimal yield. Autonomous navigation of an agricultural vehicle involves the control of different dynamic subsystems, such as the yaw angle dynamics and the longitudinal speed dynamics. In this study, a proportional-integral-derivative controller is used to control the longitudinal velocity of the tractor. For the control of the yaw angle dynamics, a proportional-derivative controller works in parallel with a type-2 fuzzy neural network. In such an arrangement, the former ensures the stability of the related subsystem, while the latter learns the system dynamics and becomes the leading controller. In this way, instead of modeling the interactions between the subsystems prior to the design of a model-based control, we develop a control algorithm which learns the interactions online from the measured feedback error. In addition to the control of the stated subsystems, a kinematic controller is needed to correct the errors in both the x - and the y - axis for the trajectory tracking problem of the tractor. To demonstrate the real-time abilities of the proposed control scheme, an autonomous tractor is equipped with the use of reasonably priced sensors and actuators. Experimental results show the efficacy and the efficiency of the proposed learning algorithm.

Index Terms—Agricultural vehicles, autonomous tractor, fuzzy-neuro control, sliding-mode learning algorithm, type-2 fuzzy logic systems.

I. INTRODUCTION

IN recent years, considerable efforts have been made to automate functions in agricultural machinery and production machines that are still carried out by humans, through the use of self-learning controllers and continuously adapt the machine

settings to the crop/animal variability and the environmental conditions.

Agricultural production machines with higher efficiencies will be very important in the future because of the limited agricultural areas in the world and constantly rising energy and labor costs. One way to offset the rising labor costs may be to increase the capacity by increasing the size of the machines. However, the limits in this direction will soon be reached as there is a maximum size that will still permit road transportation. On the other hand, the requirement for the minimization of the energy costs may make the multiobjective optimization problem in hand a difficult one to solve. A different approach to increase effectiveness would be to increase the operational efficiency by the use of advanced learning algorithms, which can learn the operational dynamics online and adjust the operational parameters accordingly. The motivation behind the use of self-learning controllers instead of conventional controllers for the control of agricultural production machines is that there are different subsystems interacting with each other in these machines, and well tuning of the controller coefficients simultaneously is a difficult task. Even if the operator becomes proficient in proper adjustment of the different controller coefficients, crop/animal variability, and the environmental conditions force the operator to change the machine settings continuously resulting in the fact that adaptability is a must. In this study, the trajectory control of an autonomous tractor is considered and, for the provision of such adaptability, a Takagi–Sugeno–Kang (TSK) type-2 fuzzy neural network (T2FNN) with a sliding mode control (SMC) theory-based learning algorithm is proposed. Various uncertainties, disturbances, and nonlinearities that would inherently exist in such a system are thus handled.

As for the structure, the combination of a conventional controller and an intelligent controller is chosen. The former is a proportional-derivative (PD) controller and the latter is a T2FNN in this study. Such a structure is called *feedback error learning* (FEL) in the literature; it was originally proposed in [1] for robot control in which a neural network based controller works with a PD controller. In this approach, the output of the conventional controller is used as the learning error signal to train the intelligent controller. A recent extension of this approach to fuzzy-neural networks (FNNs) can be seen in [2]. This study, presents a further extension by using SMC theory to train T2FNNs. The novelty of the approach is that instead of trying to minimize an error function, the learning parameters are tuned by the proposed algorithm in a way to enforce the

Manuscript received February 25, 2013; revised June 21, 2013 and November 10, 2013; accepted November 16, 2013. Date of publication January 1, 2014; date of current version October 3, 2014. Recommended by Technical Editor D. Sun. This work was supported by the framework of projects IWT-SBO 80032 (LeCoPro) of the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen), Belgium.

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

O. Kaynak is with the Department of Electrical and Electronics Engineering, Bogazici University, 34342, Istanbul, Turkey (e-mail: okayay.kaynak@boun.edu.tr).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TMECH.2013.2291874

error to satisfy a stable equation. The parameter update rules of T2FNNs are derived and the stability of the learning algorithm in the Lyapunov sense is proved.

For the training of a FNN or a T2FNN, two methods are widely used in the literature. One of them is a gradient-descent-based [3] method and therefore requires the computation of partial derivatives or sensitivity functions. It can be considered as an extension of the commonly used learning algorithm with back propagation of the error. There are some drawbacks of this method such as slow speed of learning, long computational time, and difficulties are met in determining the convergence and the stability of the learning scheme in an analytical way. What is more, as is the case with all repetitive algorithms, a number of numerical robustness issues may emerge when the algorithm is run over a long period of time [4]. In addition to these drawbacks, the tuning process can easily be trapped into a local minimum [5]. Another well known learning algorithm is based on evolutionary computations with genetic algorithms [6], [7]. Since genetic algorithms-based methods basically do a random search, they are slower than gradient-descent algorithms and computationally more intensive. Moreover, the stability of such approaches is questionable and the optimal values for the stochastic operators are difficult to derive. In order to overcome these difficulties, SMC theory-based algorithms have been proposed in [8], [9] for the parameter update rules of artificial neural networks (ANNs) and FNNs as robust learning algorithms. Even though the main strength of the SMC is its robustness, there are some disadvantages of it as well. For instance, when the system dynamics are close to the sliding surface, high-frequency oscillations in the control input, also known as chattering, occur. One of the most common methods to eliminate the chattering is inserting a boundary layer to replace the corrective control by an equivalent one when the system is inside this layer [10].

SMC is an approach that guarantees the robustness of a system in the case of external disturbances, parameter variations, and uncertainties and as such has attracted the attention of many researchers to guarantee robustness in computationally intelligent architectures [11], [12]. The main idea behind this control scheme is to restrict the motion of the system in a plane referred to as the *sliding surface*, where the predefined function of the error is zero [13]. SMC-based learning algorithms can, not only make the overall system more robust, but also ensure faster convergence than the traditional learning techniques in online tuning of ANNs and type-1 fuzzy neural networks (T1FNNs). There are various studies in the literature that aim to use the robustness property of SMC in the learning process of ANNs and T1FNNs [14]. Conversely, the robustness and the stability properties of soft computing-based control strategies can also be analyzed through the use of SMC theory [15].

The most common tools used in the literature to implement model free designs are ANNs and fuzzy logic systems (FLSs). On the fuzzy logic theory, Zadeh argues that "*fuzzy logic is a precise conceptual system of reasoning, deduction, and computation in which the objects of discourse and analysis are, or are allowed to be, associated with imperfect information. Imperfect information is information which in one or more re-*

spects is imprecise, uncertain, incomplete, unreliable, vague, or partially true" [16]. ANNs are well known for their representation capability, even in the case of highly nonlinear systems. FNNs combine the advantages of both techniques, i.e., the fuzzy reasoning ability of FLSs and the learning ability of ANNs.

The use of type-2 fuzzy sets has an advantage when it is difficult to determine the place of the membership functions (MFs) precisely [17]. Since there are infinite number of type-1 MFs within an interval in a type-2 MF, type-2 fuzzy logic systems (T2FLSs) appear to be a more promising method than their type-1 counterparts for handling uncertainties such as noisy data and changing environments [18], [19]. In [20], [21], the effects of the measurement noise in type-1 and type-2 fuzzy logic controllers and identifiers were simulated to perform a comparative analysis. It was concluded that the use of a type-2 fuzzy logic controller can be a better option than the use of its type-1 counterpart. These claims have been verified in a real-time application in [22]. Even if there exists a number of papers in the literature that claim that the performance of T2FLSs is better than T1FLSs under noisy conditions, this claim is tried to be justified by simulation or real-time studies only for some specific systems. However, in [23], a simpler T2FLS is considered with a novel MF in which the effect of input noise in the rule base is shown numerically in a general way, and it is concluded that T2FLSs should be used when needed, i.e., in the presence of noise and uncertainties in the system.

Whereas, the secondary MFs can take values in the interval of [0,1] in generalized T2FLSs, they are uniform functions that only take on values of 1 in interval T2FLSs. Since the general T2FLSs are computationally very demanding (this is because the type-reduction is computationally expensive), the use of interval T2FLSs is more commonly seen in the literature. In an interval TSK T2FLS, there exists a design parameter that weighs the sharing of lower and upper firing levels of each fired rule. That parameter can either be fixed or optimized online. While the least mean square method is generally used to find the optimal value of this parameter, an SMC theory-based learning algorithm is proposed for tuning it in [24].

The contributions of this paper are as follows.

- 1) An SMC theory-based learning algorithm is proposed for the parameter tuning of T2FNNs, which includes the design parameter that weighs the sharing of the lower and the upper firing levels of each fired rule, and its stability in the Lyapunov sense is proved.
- 2) The proposed learning algorithm is tested on the trajectory tracking problem of an autonomous agricultural tractor in the presence of various nonlinearities and uncertainties in real time.
- 3) A practical mechatronic system, illustrating how control, sensing, and actuation can be integrated to achieve an intelligent system, is designed and presented.

The body of the paper contains five sections: In Section II, the dynamic equations of the autonomous tractor are presented. In Section III, the overall control scheme is given. The proposed sliding mode FEL approach is presented and the parameter update rules for T2FNNs are proposed for the case of triangular

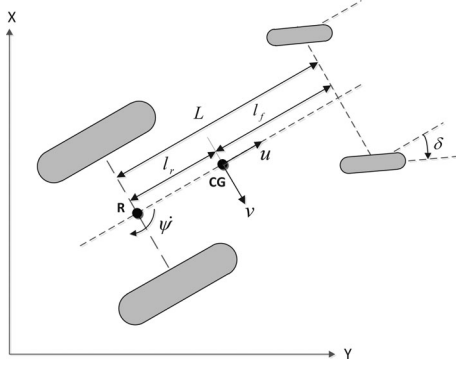


Fig. 1. Autonomous tractor.

MFs in Section IV. In Section V, real-time results are given. Finally, conclusions are presented in Section VI.

II. MATHEMATICAL DESCRIPTION OF THE TRACTOR

A. Kinematic Model

The schematic diagram of the autonomous tractor is presented in Fig. 1.

The linear velocities \dot{x} , \dot{y} , and the yaw rate $\dot{\psi}$ at the R point are written as follows:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} u \cos \psi \\ u \sin \psi \\ \frac{u \tan \delta}{L} \end{bmatrix} \quad (1)$$

where u , ψ , δ , and L represent the longitudinal velocity, the yaw angle defined on the point R on the tractor, the steering angle of the front wheel, the distance between the front and the rear axles of the tractor, respectively.

Considering the center of gravity (CG) shown in Fig. 1, the linear velocities \dot{x} , \dot{y} , and the yaw rate $\dot{\psi}$ of CG can be written as follows:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} u \cos \psi - v \sin \psi \\ u \sin \psi + v \cos \psi \\ \frac{u \tan \delta}{L} \end{bmatrix} \quad (2)$$

where v equals to the multiplication of $\dot{\psi}$ and l_r .

B. The Yaw Dynamics Model

The velocities and the side-slip angles on the rigid body of the tractor are presented in Fig. 2(a). Similarly, the forces on the rigid body of the tractor are shown in Fig. 2(b).

The lateral dynamics of the tractor can be written as follows:

$$m(\dot{v} + u\gamma) = F_{t,f} \sin \delta + F_{l,f} \cos \delta + F_{l,r} \quad (3)$$

where m , v , u , γ , $F_{t,f}$, $F_{l,f}$, $F_{l,r}$, and δ represent the mass of the tractor, the lateral velocity of the CG, the longitudinal velocity of the CG, the yaw rate, the traction and the lateral forces on the front wheel, the lateral force on the rear wheel, and the steering angle of the front wheel, respectively.

The yaw dynamics of the tractor are written as follows:

$$I_z \dot{\gamma} = l_f (F_{t,f} \sin \delta + F_{l,f} \cos \delta) - l_r F_{l,r} \quad (4)$$

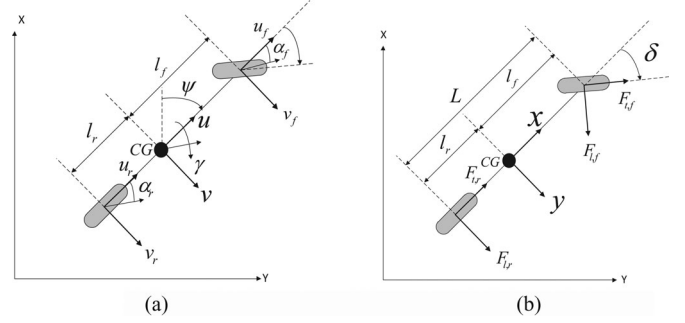


Fig. 2. Dynamic bicycle model for a tractor: (a) velocities and side-slip angles (b) forces on the rigid body of the system.

where l_f , l_r and I_z , respectively, represent the distance between the front axle and the CG of the tractor, the distance between the rear axle and the CG of the tractor, and the inertial moment of the tractor.

The tire side-slip angles must be calculated in order to determine the forces caused by the slip. It is assumed that the steering angle of the front wheel is small, and this allows to make the following approximations: $\sin \delta \approx \delta$ and $\cos \delta \approx 1$. The side-slip angles of the front (α_f) and the rear tires (α_r) are written as follows:

$$\alpha_f = \frac{v + l_f \gamma}{u} - \delta \quad \text{and} \quad \alpha_r = \frac{v - l_r \gamma}{u}. \quad (5)$$

To determine the lateral force on the tire, there are many different approaches in the literature. In this study, the lateral tire forces are calculated using a linear model which assumes these to be proportional to the side-slip angles in [25]–[27]

$$F_{l,i} = -C_{\alpha,i} \alpha_i \quad i = \{f, r\} \quad (6)$$

where $C_{\alpha,i}$, $i = \{f, r\}$, represents the cornering stiffness of the tires of the tractor. The tire cornering stiffness parameters are the averaged slopes of the lateral force characteristics in this method.

The equations of yaw motion of the autonomous tractor are written in state space form by combining (3), (4), (5), and (6) as follows:

$$\begin{bmatrix} \dot{v} \\ \dot{\gamma} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \begin{bmatrix} v \\ \gamma \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \delta(t) \quad (7)$$

where

$$\begin{aligned} A_{11} &= -\frac{C_{\alpha,f} + C_{\alpha,r}}{mu} \\ A_{12} &= \frac{-l_f C_{\alpha,f} + l_r C_{\alpha,r}}{mu} - u \\ A_{21} &= \frac{-l_f C_{\alpha,f} + l_r C_{\alpha,r}}{I_z u} \\ A_{22} &= -\frac{l_f^2 C_{\alpha,f} + l_r^2 C_{\alpha,r}}{I_z u} \\ B_1 &= \frac{C_{\alpha,f}}{m}, \quad B_2 = \frac{l_f C_{\alpha,f}}{I_z}. \end{aligned} \quad (8)$$

III. OVERALL CONTROL SCHEME

A. Kinematic Controller

The kinematic model is rewritten in a state-space as follows:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \cos \psi & -l_r \sin \psi \\ \sin \psi & l_r \cos \psi \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ \gamma \end{bmatrix} \quad (9)$$

where lateral velocity v equals to γl_r .

An inverse kinematic model is needed to calculate the reference speed and the yaw rate for the tractor. It is written as follows:

$$\begin{bmatrix} u \\ \gamma \end{bmatrix} = \begin{bmatrix} \cos \psi & \sin \psi \\ -\frac{1}{l_r} \sin \psi & \frac{1}{l_r} \cos \psi \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} \quad (10)$$

and the kinematic control law proposed in [28] to be applied to the tractor for trajectory tracking control is written as:

$$\begin{bmatrix} u_{\text{ref}} \\ \gamma_{\text{ref}} \end{bmatrix} = \begin{bmatrix} \cos \psi & \sin \psi \\ -\frac{1}{l_r} \sin \psi & \frac{1}{l_r} \cos \psi \end{bmatrix} \begin{bmatrix} \dot{x}_d + k_s \tanh(k_e e_x) \\ \dot{y}_d + k_s \tanh(k_e e_y) \end{bmatrix} \quad (11)$$

where $e_x = x_d - x$ and $e_y = y_d - y$ are the current position errors in the axes X and Y , respectively. The parameter k_e is the gain of the controller and k_s is the saturation constant. The coordinates (x, y) and (x_d, y_d) are the current and the desired coordinates, respectively. The parameters u_{ref} and γ_{ref} are the generated references for the speed and the yaw rate controllers, respectively.

B. Dynamic Controllers

The proposed control scheme used in this study is illustrated in Fig. 4. The arrow in Fig. 4 indicates that the output of the PD controller is used to tune the parameters of the T2FNN. The output of the PD+T2FNN controller is the steering angle of the front wheel. A low level controller is used to control the steering mechanism. A proportional-integral-derivative (PID) controller is used for the control of longitudinal velocity. For the control of yaw dynamics, a conventional controller (such as a PD controller) works in parallel with an intelligent controller. One of the main concerns of this study is to implement a novel learning algorithm for T2FNNs by using some novel SMC theory-based learning rules to a real-time system. As a testing environment, the yaw dynamics of the tractor in our laboratory has been chosen. On the other hand, the longitudinal dynamics could, of course, be selected. Moreover, there could be two T2FNNs running on the control of the two subsystems (yaw dynamics and longitudinal dynamics) simultaneously. The reason for such a selection in this study is that the yaw dynamics control of the tractor is more common in agricultural machines. Even if a time-based trajectory (both the yaw angle and the longitudinal speed of the tractor are controlled simultaneously) is given to the system in this study, a space-based trajectory (the longitudinal speed is fixed and only the yaw dynamics is controlled) is also very common in agricultural applications. Based on these concerns, the yaw dynamics of the tractor has been chosen for

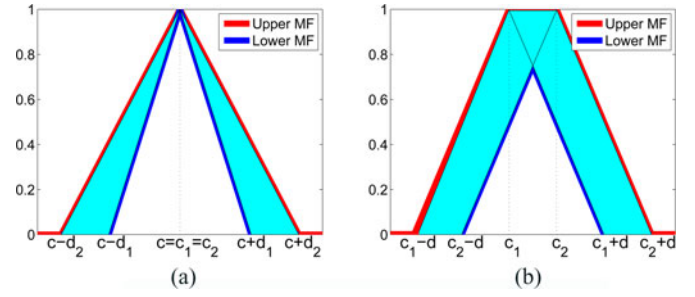


Fig. 3. Type-2 fuzzy triangular MF with (a) uncertain width and (b) uncertain center.

the implementation for the novel learning algorithm proposed in this study.

IV. TYPE-2 FUZZY-NEURO CONTROL APPROACH

A. Type-2 Fuzzy Triangular Membership Functions

Among the existing MFs in the literature (that include a novel one proposed by the authors [23]), triangular MFs are preferred in this paper. Their advantages over the others are mentioned in [29].

The mathematical expression for a type-1 fuzzy triangular MF can be written as

$$\mu(x) = \begin{cases} 1 - \left| \frac{x-c}{d} \right|, & |x-c| < d \\ 0, & \text{otherwise} \end{cases} \quad (12)$$

where c and d are the *center* and the *width* of the MF, respectively, and x is the input vector. On the other hand, type-2 fuzzy triangular MFs with uncertain width and uncertain center are shown in Figs. 3(a) and (b), respectively. In this paper, MFs with uncertain width are preferred in the antecedent parts of the fuzzy *if-then* rules.

Each MF in the antecedent part is represented by an *upper* (red line in Fig. 3) and a *lower* (blue line in Fig. 3) MF. The membership values for the upper and the lower MFs are denoted as $\overline{\mu}(x)$ and $\underline{\mu}(x)$, respectively. The strength of the rule R_{ij} is calculated as a *T*-norm of the MFs in the premise part by using a multiplication operator

$$\underline{W}_{ij} = \underline{\mu}_{1i}(x_1) \underline{\mu}_{2j}(x_2) \quad \text{and} \quad \overline{W}_{ij} = \overline{\mu}_{1i}(x_1) \overline{\mu}_{2j}(x_2) \quad (13)$$

The type-2 fuzzy triangular membership values $\underline{\mu}_{1i}(x_1)$, $\overline{\mu}_{1i}(x_1)$, $\underline{\mu}_{2j}(x_2)$, and $\overline{\mu}_{2j}(x_2)$ of the inputs x_1 and x_2 in the above expression have the following appearance:

$$\underline{\mu}_{1i}(x_1) = \begin{cases} 1 - \left| \frac{x_1 - c_{1i}}{\underline{d}_{1i}} \right|, & |x_1 - c_{1i}| < \underline{d}_{1i} \\ 0, & \text{otherwise} \end{cases}$$

$$\overline{\mu}_{1i}(x_1) = \begin{cases} 1 - \left| \frac{x_1 - c_{1i}}{\overline{d}_{1i}} \right|, & |x_1 - c_{1i}| < \overline{d}_{1i} \\ 0, & \text{otherwise} \end{cases}$$

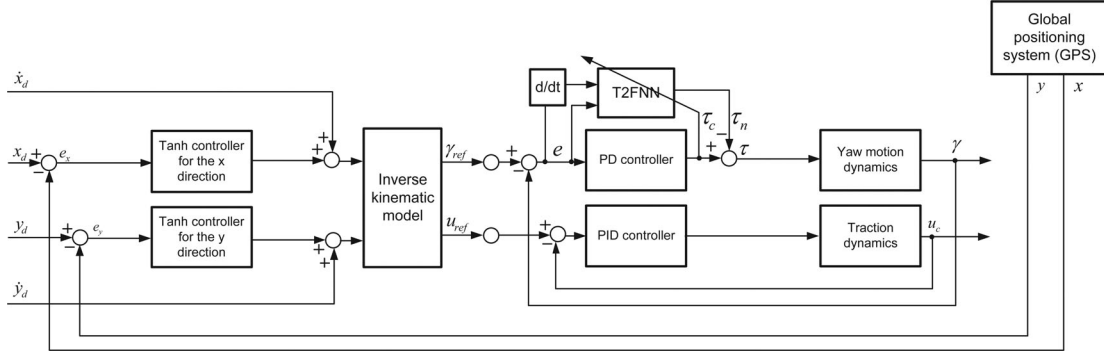


Fig. 4. Block diagram of the proposed adaptive fuzzy neuro scheme.

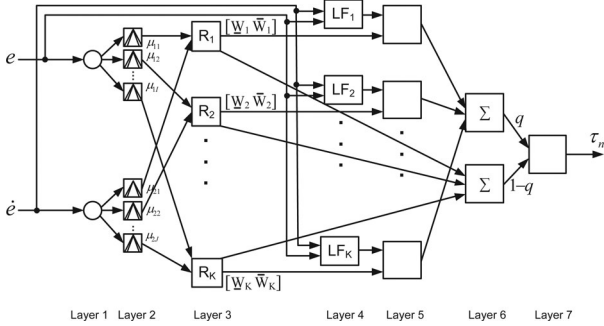


Fig. 5. Structure of the proposed T2FNN for two inputs.

$$\begin{aligned} \underline{\mu}_{2j}(x_2) &= \begin{cases} 1 - \frac{|x_2 - c_{2j}|}{d_{2j}}, & |x_2 - c_{2j}| < d_{2j} \\ 0, & \text{otherwise} \end{cases} \\ \overline{\mu}_{2j}(x_2) &= \begin{cases} 1 - \frac{|x_2 - c_{2j}|}{\overline{d}_{2j}}, & |x_2 - c_{2j}| < \overline{d}_{2j} \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (14)$$

B. Interval Type-2 A2-CO TSK Model

The interval T2FLS considered in this paper uses type-2 triangular MFs in the premise part and crisp numbers in the consequent part. This structure is called A2-C0 fuzzy system [30], and it is shown in Fig. 5. The fuzzy *If-Then* rule R_{ij} of a $zero^{th}$ -order type-2 TSK model with two input variables in which the consequent part is a crisp number can be defined as follows:

$$R_{ij} : \text{ If } x_1 \text{ is } \tilde{A}_{1i} \text{ and } x_2 \text{ is } \tilde{A}_{2j}, \text{ then } f_{ij} = d_{ij}. \quad (15)$$

The followings are the operations in each Layer in Fig. 5: In Layer 1, the input signals feed the system. The related figure shows the system for two inputs which are the error and the time derivative of the error. In Layer 2, the membership degrees $\underline{\mu}$ and $\overline{\mu}$ are determined for each input signal entering the system. Layer 3 calculates the firing strengths of the rules which are realized using the *prod* t-norm operator using (13). Layer 4 determines the outputs of the linear functions f_{ij} ($i = 1, \dots, I$ and $j = 1, \dots, J$), in the consequent parts for the two inputs

case

$$f_{ij} = d_{ij}. \quad (16)$$

Layer 5 computes the product of the membership degrees \underline{W}_{ij} and \overline{W}_{ij} and linear functions f_{ij} . Two summation blocks are in Layer 6. One of these blocks computes the sum of the output signals from Layer 5 [the numerator part of (17)] and the other block computes the sum of the output signal of Layer 3 [the denominator part of (17)]. Finally, Layer 7 calculates the output of the network using (18).

The output of the network is calculated as follows:

$$\tau_n = \int_{W_{11} \in [\underline{W}_{11}, \overline{W}_{11}]} \dots \int_{W_{IJ} \in [\underline{W}_{IJ}, \overline{W}_{IJ}]} \frac{1}{\sum_{i=1}^I \sum_{j=1}^J \underline{W}_{ij}(x) f_{ij}} \times \frac{\sum_{i=1}^I \sum_{j=1}^J \overline{W}_{ij}(x) f_{ij}}{\sum_{i=1}^I \sum_{j=1}^J \overline{W}_{ij}(x)} \quad (17)$$

where f_{ij} is given by the *If-Then* rule. The inference engine used in this paper replaces the type-reduction which is given as:

$$\begin{aligned} \tau_n &= \frac{q(t) \sum_{i=1}^I \sum_{j=1}^J \underline{W}_{ij} f_{ij}}{\sum_{i=1}^I \sum_{j=1}^J \underline{W}_{ij}} \\ &\quad + \frac{(1 - q(t)) \sum_{i=1}^I \sum_{j=1}^J \overline{W}_{ij} f_{ij}}{\sum_{i=1}^I \sum_{j=1}^J \overline{W}_{ij}} \end{aligned} \quad (18)$$

The design parameter q , weights the sharing of the lower and the upper firing levels of each fired rule [30]. This parameter can be a constant (equal to 0.5 in most cases) or a time-varying parameter. In this paper, the latter is preferred. In other words, the parameter update rules and the proof of the stability of the learning process are given for the case of a time-varying q .

After the normalization of (18), the output signal of the T2FNN will obtain the following form:

$$\tau_n = q(t) \sum_{i=1}^I \sum_{j=1}^J f_{ij} \widetilde{\underline{W}_{ij}} + (1 - q(t)) \sum_{i=1}^I \sum_{j=1}^J f_{ij} \widetilde{\overline{W}_{ij}} \quad (19)$$

where $\widetilde{\underline{W}_{ij}}$ and $\widetilde{\overline{W}_{ij}}$ are the normalized values of the lower and the upper output signals of the neuron ij from the second hidden layer of the network

$$\widetilde{\underline{W}_{ij}} = \frac{\underline{W}_{ij}}{\sum_{i=1}^I \sum_{j=1}^J \underline{W}_{ij}} \text{ and } \widetilde{\overline{W}_{ij}} = \frac{\overline{W}_{ij}}{\sum_{i=1}^I \sum_{j=1}^J \overline{W}_{ij}}.$$

The following vectors can be specified:

$$\begin{aligned}\widetilde{W}(t) &= [\widetilde{W}_{11}(t) \ \widetilde{W}_{12}(t) \ \dots \ \widetilde{W}_{21}(t) \ \dots \ \widetilde{W}_{ij}(t) \\ &\quad \dots \ \widetilde{W}_{IJ}(t)]^T \\ \widetilde{W}(t) &= [\widetilde{W}_{11}(t) \ \widetilde{W}_{12}(t) \ \dots \ \widetilde{W}_{21}(t) \ \dots \ \widetilde{W}_{ij}(t) \\ &\quad \dots \ \widetilde{W}_{IJ}(t)]^T \\ F &= [f_{11} \ f_{12} \ \dots \ f_{21} \ \dots \ f_{ij} \ \dots \ f_{IJ}].\end{aligned}$$

The following assumptions have been used in this investigation: Both the input signals $x_1(t)$ and $x_2(t)$, and their time derivatives can be considered bounded:

$$|x_1(t)| \leq \widetilde{B}_x, \ |x_2(t)| \leq \widetilde{B}_x \forall t \quad (20)$$

$$|\dot{x}_1(t)| \leq \widetilde{B}_{\dot{x}}, \ |\dot{x}_2(t)| \leq \widetilde{B}_{\dot{x}} \forall t \quad (21)$$

where \widetilde{B}_x and $\widetilde{B}_{\dot{x}}$ are assumed to be some known positive constants. It is obvious that $0 < \widetilde{W}_{ij} \leq 1$ and $0 < \widetilde{W}_{ij} \leq 1$. In addition, it can be easily seen that $\sum_{i=1}^I \sum_{j=1}^J \widetilde{W}_{ij} = 1$ and $\sum_{i=1}^I \sum_{j=1}^J \widetilde{W}_{ij} = 1$. It is also considered that, τ and $\dot{\tau}$ will be bounded signals too, i.e.,

$$|\tau(t)| < B_\tau, \ |\dot{\tau}(t)| < B_{\dot{\tau}} \quad \forall t \quad (22)$$

where B_τ and $B_{\dot{\tau}}$ are some known positive constants.

C. The SMC Theory-based Learning Algorithm

Using the principles of SMC theory [31] the zero value of the learning error coordinate $\tau_c(t)$ can be defined as a time-varying sliding surface, i.e.,

$$S_c(\tau_n, \tau) = \tau_c(t) = \tau_n(t) + \tau(t) = 0 \quad (23)$$

which is the condition that the T2FNN is trained to become a nonlinear regulator to obtain the desired response during the tracking-error convergence movement by compensating the nonlinearity of the controlled plant.

The sliding surface for the nonlinear system under control $S_p(e, \dot{e})$ is defined as

$$S_p(e, \dot{e}) = \dot{e} + \chi e \quad (24)$$

with χ being a positive constant determining the slope of the sliding surface.

Definition: A sliding motion will appear on the sliding manifold $S_c(\tau_n, \tau) = \tau_c(t) = 0$ after a 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 $[t, t_h) \subset (0, t_h)$.

It is desired to devise a dynamical feedback adaptation mechanism, or an online learning algorithm for the parameters of the T2FNN considered, such that the sliding mode condition of the above definition is enforced.

D. The Proposed Parameter Update Rules for the T2FNN

The parameter update rules for the T2FNN proposed in this paper which has two inputs are given by the following theorem.

Theorem 1: If the adaptation laws for the parameters of the considered T2FNN are chosen as

$$\dot{\bar{c}}_{1i} = \bar{c}_{1i} = \dot{c}_{1i} = \dot{x}_1 \quad (25)$$

$$\dot{\bar{c}}_{2j} = \bar{c}_{2j} = \dot{c}_{2j} = \dot{x}_2 \quad (26)$$

$$\dot{\bar{d}}_{1i} = \frac{\mu_{1i}}{\mu_{1i}} \frac{-\alpha \bar{d}_{1i}^2}{x_1 - c_{1i}} \text{sgn}(\tau_c) \text{sgn}\left(\frac{x_1 - c_{1i}}{\bar{d}_{1i}}\right) \quad (27)$$

$$\dot{\bar{d}}_{1i} = \frac{\mu_{1i}}{\mu_{1i}} \frac{-\alpha \bar{d}_{1i}^2}{x_1 - c_{1i}} \text{sgn}(\tau_c) \text{sgn}\left(\frac{x_1 - c_{1i}}{\bar{d}_{1i}}\right) \quad (28)$$

$$\dot{\bar{d}}_{2j} = \frac{\mu_{2j}}{\mu_{2j}} \frac{-\alpha \bar{d}_{2j}^2}{x_2 - c_{2j}} \text{sgn}(\tau_c) \text{sgn}\left(\frac{x_2 - c_{2j}}{\bar{d}_{2j}}\right) \quad (29)$$

$$\dot{\bar{d}}_{2j} = \frac{\mu_{2j}}{\mu_{2j}} \frac{-\alpha \bar{d}_{2j}^2}{x_2 - c_{2j}} \text{sgn}(\tau_c) \text{sgn}\left(\frac{x_2 - c_{2j}}{\bar{d}_{2j}}\right) \quad (30)$$

$$\dot{f}_{ij} = - \frac{\left(q(t) \widetilde{W}_{ij} + (1 - q(t)) \widetilde{W}_{ij} \right) \alpha \text{sgn}(\tau_c)}{(q(t) \widetilde{W} + (1 - q(t)) \widetilde{W})^T (q(t) \widetilde{W} + (1 - q(t)) \widetilde{W})} \quad (31)$$

$$\dot{q}(t) = - \frac{\alpha \text{sgn}(\tau_c)}{F(\widetilde{W} - \widetilde{W})^T} \quad (32)$$

where α is a sufficiently large positive design constant satisfying the following inequality

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

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

Proof: The reader is referred to Appendix A.

In order to avoid division by zero in the adaptation laws of (27)–(32), an instruction is included in the algorithm to make the denominator equal to 0.001 when its calculated value is smaller than this threshold.

It is well known that sliding mode control suffers from high-frequency oscillations in the control input, which are called *chattering*. The following are the two common methods used to eliminate chattering [32]:

- 1) Using a saturation function to replace the signum function.
- 2) Inserting a boundary layer so that an equivalent control replaces the corrective one when the system is inside this layer.

In order to reduce the chattering effect, the following function is used in this paper with $\delta_s = 0.05$ instead of the signum function in the dynamic strategy described in (27)–(32):

$$\text{sgn}(\tau_c) := \frac{\tau_c}{|\tau_c| + \delta_s} \quad (34)$$

where $\delta_s = 0.05$.



Fig. 6. Autonomous tractor.

The relation between the sliding line S_p and the zero adaptive learning error level S_c is determined by the following equation:

$$S_c = \tau_c = k_D \dot{e} + k_P e = k_D \left(\dot{e} + \frac{k_p}{k_D} e \right) = k_D S_p. \quad (35)$$

The tracking performance of the feedback control system can be analyzed by introducing the following Lyapunov function candidate:

$$V_p = \frac{1}{2} S_p^2. \quad (36)$$

Theorem 2: If the adaptation strategy for the adjustable parameters of the T2FNN is chosen as in (25)–(32), then the negative definiteness of the time derivative of the Lyapunov function in (36) is ensured.

Proof: The reader is referred to Appendix B.

Remark: The obtained result means that, assuming that the SMC task is achievable, using τ_c as a learning error for the T2FNN together with the adaptation laws (25)–(32) enforces the desired reaching mode followed by a sliding regime for the system under control.

V. EXPERIMENTAL SETUP

The global objective in the following real-time experiments is to track a time-based trajectory with the Case New Holland TZ25DA tractor shown in Fig. 6.

A. Global Positioning System

A global positioning system (GPS) antenna is located straight up the center of the tractor rear axle to provide highly accurate positional information for the autonomous tractor. The height of the antenna is 2 m above ground level. It is connected to a Septentrio AsteRx2eH RTK-DGPS receiver (Septentrio Satellite Navigation NV, Belgium) with a specified position accuracy of 2 cm at a 20-Hz sampling frequency (sampling period

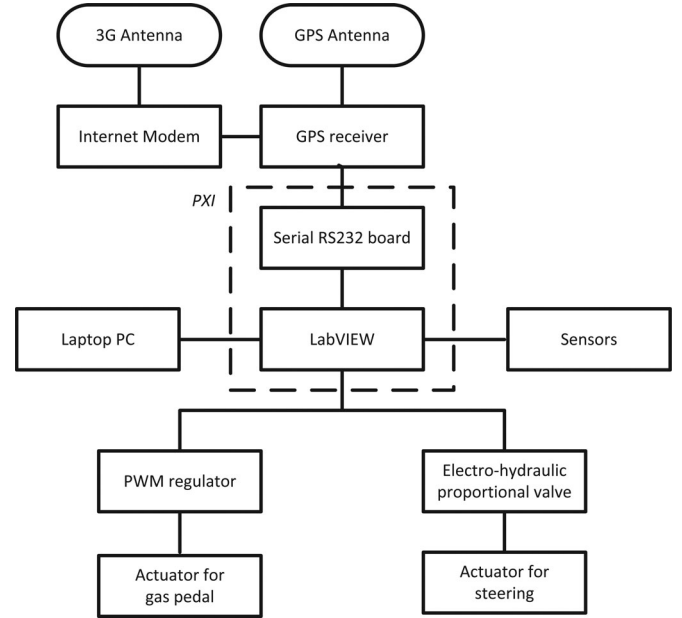


Fig. 7. Block diagram of the hardware.

$T_s = 0.05$ s). The RTK correction signals are obtained from the Flepos network through a wireless internet connection established with a Digi Connect WAN 3G modem.

B. Hardware, Software, and Sensors

The block diagram of hardware is shown in Fig. 7. The GPS receiver and the internet modem are connected to a real-time operating system (PXI platform, National Instrument Corporation, USA) via a RS232 serial communication. The PXI system gathers the steering angles and the GPS data, and it controls the tractor by sending signals to the actuators. A laptop connected via a wireless network to the PXI system functions as the user interface of the autonomous tractor. The control algorithms are implemented in *LabVIEW*TM version 2011, National Instrument, USA. They are executed in real time on the PXI and updated at a rate of 20 Hz.

The PD+T2FNN control algorithm calculates the desired steering angle for the front wheels. In the inner closed loop, the steering mechanism is controlled by using an electro-hydraulic valve from Sauer Danfoss with a flow of 12 liter/min. The electro-hydraulic valve characteristics are highly nonlinear and include a saturation and a dead-band region. The voltage, limited between 0 – 12 volt and the steering angle, limited between $\pm 45^\circ$ constitute the input and the output for the steering system, respectively. The position of the front wheels is measured using a potentiometer mounted on the front axle, yielding a position measurement with a resolution of 1° . The position of the front wheel is measured 100 times at every 50 ms. The data are filtered by using a Savitzky–Golay filter which tends to preserve features of the distribution such as relative maxima, minima, and width. After filtering process, the steering angle is found by taking the mean value of the filtered data. This procedure is repeated every 50 ms.

The speed of the tractor is controlled by using an electro-mechanic valve. Two PID type controllers are used in cascade fashion. The one in the outer loop generates the desired pedal position with respect to the speed of the tractor and the one in the inner loop generates the voltage value for the electro-mechanic valve with respect to the pedal position. Like in the measuring of the steering angle, the pedal position is measured 100 times at every 50 ms and the data are filtered by using a Savitzky–Golay filter and taking the mean value of the filtered data. Since the measured speed signal coming from the GPS is noisy, a discrete Kalman filter (KF) is used to reduce noise. A position-velocity model described in [33] is used where vehicle velocity is assumed as a random-walk process. The KF assumes that the vehicle moves with a constant velocity between discrete-time steps. The state vector of the model used in the KF and the state transition matrix are as follows:

$$\begin{aligned}\hat{x}_{k+1} &= \Phi(T_s)\hat{x}_k \\ &= \begin{bmatrix} 1 & T_s & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T_s \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_k \\ v_{x,k} \\ y_k \\ v_{y,k} \end{bmatrix} \quad (37)\end{aligned}$$

where $\Phi(T_s)$, $v_{x,k}$, and $v_{y,k}$ are the state transition matrix, easting, and northing velocities coming from the GPS, respectively.

C. State Estimation

Some states of the autonomous tractor cannot be measured. Even when the states can be measured directly, the measurements will contain delay and noise. Moreover, at some time steps, no useful position data are obtained from the GPS receiver.

An extended Kalman filter (EKF) is used for state estimation. Since the GPS antenna is located at the R point on the tractor, the kinematic model in (1) is used. The discrete-time kinematic model used by the EKF is written with a sampling interval T_s as follows:

$$\begin{aligned}x_{k+1} &= x_k + T_s u_k \cos \psi_k \\ y_{k+1} &= y_k + T_s u_k \sin \psi_k \\ \psi_{k+1} &= \psi_k + T_s u_k \frac{\tan \delta_k}{L}.\end{aligned} \quad (38)$$

The general form of the estimated system model is

$$\begin{aligned}\hat{x}_{k+1} &= f(\hat{x}_k, u_k) + w_k \\ \hat{y}_{k+1} &= h(\hat{x}_k) + v_k\end{aligned} \quad (39)$$

where f is the estimation model for the system and h is the measurement function. The differences between the kinematic model and the estimation model are the process noise w_k and the observation noise v_k both in the state and the measurement equations. They are both assumed to be independent with zero mean multivariate Gaussian noises with covariance matrices Q_k and R_k , respectively,

$$\begin{aligned}w_k &\sim N(0, Q_k) \\ v_k &\sim N(0, R_k).\end{aligned} \quad (40)$$

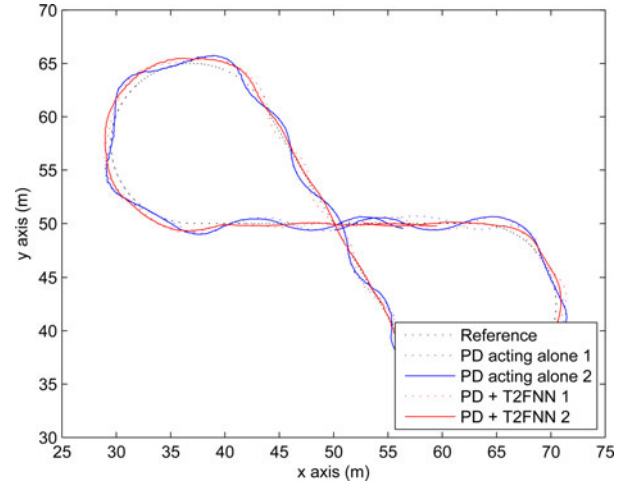


Fig. 8. Trajectory of the autonomous tractor.

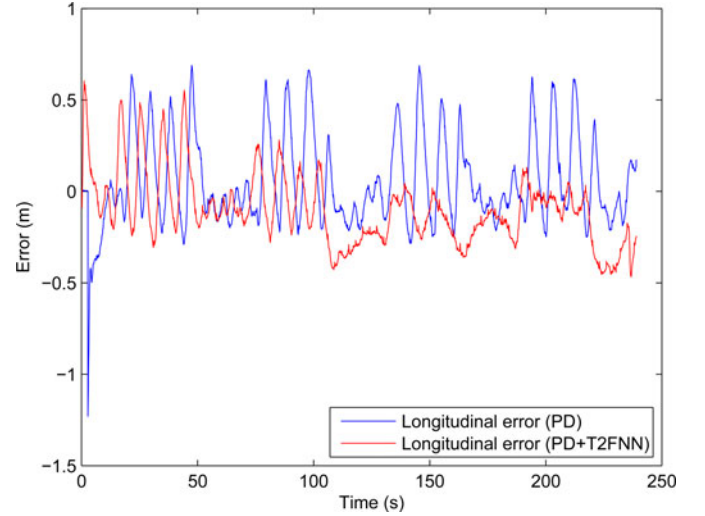


Fig. 9. Longitudinal error.

Since only one GPS antenna was mounted on the tractor, the yaw angle of the tractor was not measured directly. It is to be noted that the knowledge of the yaw angle of the tractor plays a very important role in the accuracy of trajectory tracking control. The estimated value of the yaw angle is used in the inverse kinematic model to generate the desired speed and the desired yaw rate for the system. The inputs of the EKF are the position and the velocity, the velocity values from GPS and the steering angle value from the potentiometer in the front wheels. The outputs of the EKF are the position of the tractor in the x - and the y -coordinates system and the yaw angle. These estimated values are then used in the trajectory control.

D. Experimental Results

An eight-shaped reference trajectory is applied to the system. The reference and the actual trajectories of the system, both the longitudinal and the lateral error values on the related trajectory are shown in Fig. 8, Fig. 9, and Fig. 10 for two different controllers, respectively. The results show that the control scheme

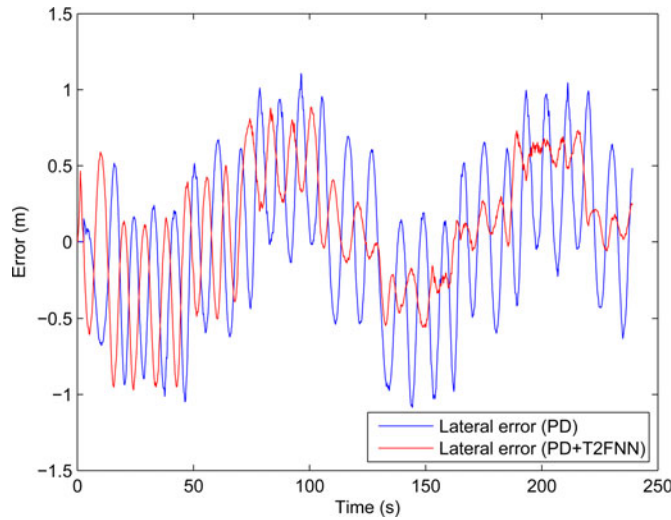


Fig. 10. Lateral error.

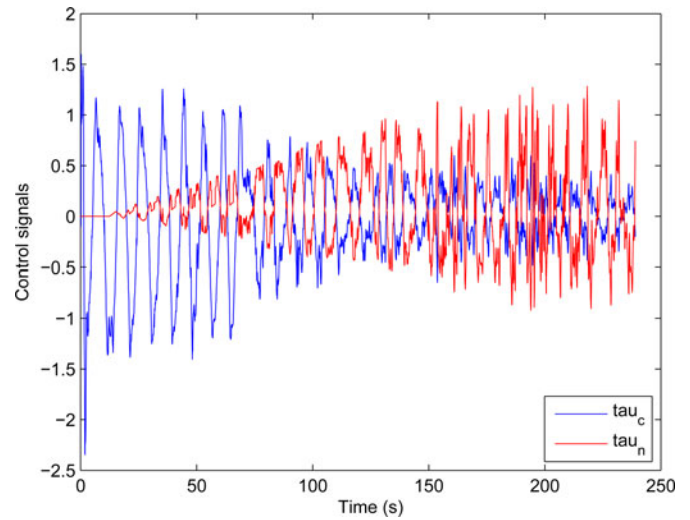


Fig. 11. Control signals.

consisting of a T2FNN working in parallel with a PD controller gives a better trajectory following accuracy than the one where only a PD controller acts alone. It can here be argued that the performance of the conventional controller acting alone can be improved by better tuning, but as it has already been stated that in real life, this is a challenging task; because in addition to the interactions of the subsystems, there exist unmodeled dynamics and uncertainties in real world applications. Thus, the proposed control structure, consisting of an intelligent controller and a conventional controller would be preferable in real life.

In Fig. 8, while the dotted lines show the first turns, the solid lines represent the second turns. The control accuracy for the PD controller is the same for the first and second turn which is the expected case. However, the control accuracy for the T2FNN working in parallel with a PD controller is better for the second turn. We can observe similar behavior in Figs. 9 and 10. The mean square of the lateral error in Fig. 10 is equal to 0.2575 and 0.1803 for the case of the PD controller working alone and the case of the PD controller working in parallel with the T2FNN, respectively. The results show a performance improvement of 30% in the latter case.

Fig. 11 shows the control signals coming from the conventional PD controller and T2FNN when the PD controller works in parallel with the T2FNN. As can be seen from Fig. 11, at the beginning (in the first turn), the dominating control signal is the one coming from the PD controller. After the first turn (starting from 120th second), the T2FNN is able to take over the control, thus becoming the leading controller. Moreover, when the reference signal changes, the output of the PD controller increases. However, after a finite time, the output of the PD controller again comes back to approximately zero.

Although there exist two independent subsystem controllers in the autonomous tractor control system, the T2FNN works in parallel with a PD controller only for the control of the yaw rate of the system. Thus, the error signals for both PD controller acting alone and in parallel with the T2FNN are shown in Fig. 12.

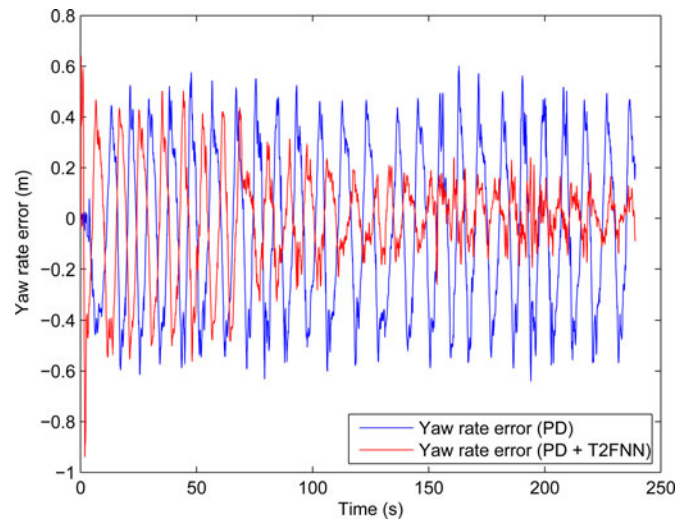
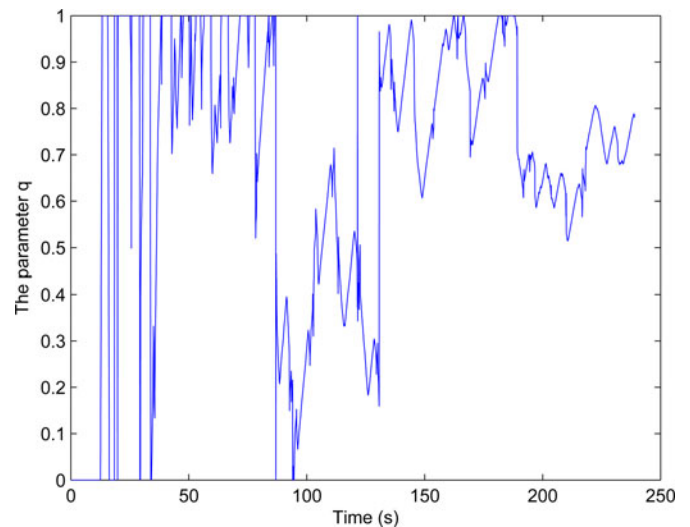


Fig. 12. Yaw rate error.

Fig. 13. Parameter q .

As can be seen from Fig. 12, the T2FNN significantly increases the control accuracy of the yaw dynamics of the system.

Fig. 13 represents the online tuning of the parameter q in (19) which weights the sharing of the lower and the upper firing levels of each fired rule. The learning algorithms derived in this paper include a parameter adaptation rule for q too and we can see the effect of this on this figure when the reference trajectory changes abruptly. It can also be observed that the change in this parameter is not very big in the second turn when compared to the first turn. During the training of the parameter q , a constraint is put to the parameter in which it is bounded between 0 and 1. When it reaches its limits, the tuning is turned off for that parameter.

VI. CONCLUSION

In an interval TSK T2FLS, there exists a design parameter that balances the sharing of the lower and the upper firing levels of each fired rule, and this parameter can be tuned during the real-time control of the system. One novelty of this paper is that an sliding mode control theory-based learning algorithm is proposed for the tuning of that parameter, the effect of which is evaluated in real time for the control of the yaw dynamics of an autonomous tractor. Instead of forcing the designer to choose a fixed value for the aforementioned parameter, this study allows the system to tune it online providing an additional degree of freedom for the overall controller. The real-time results show that when the T2FNN is used in parallel with a conventional PD controller, the overall system learns the system dynamics to perform a better performance in terms of a smaller settling time and near zero steady state error. The use of the combination of fuzzy logic control, artificial neural networks, and sliding mode control theory harmoniously allows us to better handle the interactions in the subsystems, uncertainties and lack of modeling information. In addition to its well known robustness property, another prominent feature of the proposed approach is its computational simplicity.

APPENDIX A PROOF OF THEOREM 1

The time derivatives of (14) are as follows:

$$\dot{\mu}_{1i}(x_1) = \left(-\frac{\dot{x}_1 - \dot{c}_{1i}}{d_{1i}} + \frac{x_1 - c_{1i}}{d_{1i}^2} \dot{d}_{1i} \right) \text{sgn} \left(\frac{\overbrace{x_1 - c_{1i}}^{E_1}}{d_{1i}} \right) \quad (41)$$

$$\dot{\mu}_{1i}(x_1)Z = \left(-\frac{\dot{x}_1 - \dot{c}_{1i}}{d_{1i}} + \frac{x_1 - c_{1i}}{d_{1i}^2} \dot{d}_{1i} \right) \text{sgn} \left(\frac{\overbrace{x_1 - c_{1i}}^{E_1}}{d_{1i}} \right) \quad (42)$$

$$\dot{\mu}_{2j}(x_2) = \left(-\frac{\dot{x}_2 - \dot{c}_{2j}}{d_{2j}} + \frac{x_2 - c_{2j}}{d_{2j}^2} \dot{d}_{2j} \right) \text{sgn} \left(\frac{\overbrace{x_2 - c_{2j}}^{E_2}}{d_{2j}} \right) \quad (43)$$

$$\dot{\mu}_{2j}(x_2) = \left(-\frac{\dot{x}_2 - \dot{c}_{2j}}{d_{2j}} + \frac{x_2 - c_{2j}}{d_{2j}^2} \dot{d}_{2j} \right) \text{sgn} \left(\frac{\overbrace{x_2 - c_{2j}}^{E_2}}{d_{2j}} \right). \quad (44)$$

The time derivative of the strength of the rule R_{ij} for the lower MF in (13) is as follows:

$$\begin{aligned} \dot{\widetilde{W}}_{ij} &= \frac{\overbrace{(\dot{\mu}_{1i}(x_1)\mu_{2j}(x_2) + \mu_{1i}(x_1)\dot{\mu}_{2j}(x_2))}^A}{\sum_{i=1}^I \sum_{j=1}^J \widetilde{W}_{ij}} \\ &\quad - \frac{\widetilde{W}_{ij}(\sum_{i=1}^I \sum_{j=1}^J \overbrace{(\dot{\mu}_{1i}(x_1)\mu_{2j}(x_2) + \mu_{1i}(x_1)\dot{\mu}_{2j}(x_2))}^A)}{\sum_{i=1}^I \sum_{j=1}^J \widetilde{W}_{ij}}. \end{aligned} \quad (45)$$

If the parameter update rules in Theorem 1 are used, (46) is achieved

$$\begin{aligned} A &= \left(\left(\frac{x_1 - c_{1i}}{d_{1i}^2} \dot{d}_{1i} - \left(\frac{\dot{x}_1 - \dot{c}_{1i}}{d_{1i}} \right) \right) \text{sgn}(\underline{E}_1) \overbrace{(1 - \underline{E}_2 \text{sgn}(\underline{E}_2))}^{\mu_{2j}} \right) \\ &\quad + \left(\left(\frac{x_2 - c_{2j}}{d_{2j}^2} \dot{d}_{2j} - \left(\frac{\dot{x}_2 - \dot{c}_{2j}}{d_{2j}} \right) \right) \text{sgn}(\underline{E}_2) \overbrace{(1 - \underline{E}_1 \text{sgn}(\underline{E}_1))}^{\mu_{1i}} \right) \\ &= -2\alpha \text{sgn}(\tau_c) \mu_{1i} \mu_{2j} \\ &= -2\alpha \text{sgn}(\tau_c) \widetilde{W}_{ij}. \end{aligned} \quad (46)$$

If (46) is inserted into (45), (47) can be achieved

$$\begin{aligned} \dot{\widetilde{W}}_{ij} &= -2\alpha \widetilde{W}_{ij} \text{sgn}(\tau_c) - \frac{\widetilde{W}_{ij} \sum_{i=1}^I \sum_{j=1}^J (-2\alpha \widetilde{W}_{ij} \text{sgn}(\tau_c))}{\sum_{i=1}^I \sum_{j=1}^J \widetilde{W}_{ij}} \\ &= -2\alpha \widetilde{W}_{ij} \text{sgn}(\tau_c) + \widetilde{W}_{ij} \sum_{i=1}^I \sum_{j=1}^J (2\alpha \widetilde{W}_{ij} \text{sgn}(\tau_c)) \\ &= -2\alpha \widetilde{W}_{ij} \text{sgn}(\tau_c) + 2\alpha \text{sgn}(\tau_c) \widetilde{W}_{ij} \sum_{i=1}^I \sum_{j=1}^J \widetilde{W}_{ij} \\ &= -2\alpha \widetilde{W}_{ij} \text{sgn}(\tau_c) + 2\alpha \widetilde{W}_{ij} \text{sgn}(\tau_c) \\ &= 0. \end{aligned} \quad (47)$$

Similarly, it can easily be shown that:

$$\dot{\widetilde{W}}_{ij} = 0. \quad (48)$$

By using the following Lyapunov function, the stability condition is checked as follows:

$$V_c = \frac{1}{2} \tau_c^2(t) \quad (49)$$

$$\dot{V}_c = \tau_c \dot{\tau}_c = \tau_c (\dot{\tau}_n + \dot{\tau}) \quad (50)$$

$$\begin{aligned}\tau_n &= \frac{q(t) \sum_{i=1}^I \sum_{j=1}^J f_{ij} \underline{W}_{ij}}{\sum_{i=1}^I \sum_{j=1}^J \underline{W}_{ij}} + \frac{(1-q(t)) \sum_{i=1}^I \sum_{j=1}^J f_{ij} \overline{W}_{ij}}{\sum_{i=1}^I \sum_{j=1}^J \overline{W}_{ij}} \\ &= q(t) \sum_{i=1}^I \sum_{j=1}^J f_{ij} \widetilde{W}_{ij} + (1-q(t)) \sum_{i=1}^I \sum_{j=1}^J f_{ij} \widetilde{W}_{ij}. \quad (51)\end{aligned}$$

The time derivative of (51) is as follows:

$$\begin{aligned}\dot{\tau}_n &= q(t) \sum_{i=1}^I \sum_{j=1}^J (\dot{f}_{ij} \widetilde{W}_{ij} + f_{ij} \dot{\widetilde{W}}_{ij}) \\ &+ (1-q(t)) \sum_{i=1}^I \sum_{j=1}^J (\dot{f}_{ij} \widetilde{W}_{ij} + f_{ij} \dot{\widetilde{W}}_{ij}) \\ &+ \dot{q}(t) \sum_{i=1}^I \sum_{j=1}^J f_{ij} \widetilde{W}_{ij} - \dot{q}(t) \sum_{i=1}^I \sum_{j=1}^J f_{ij} \widetilde{W}_{ij}. \quad (52)\end{aligned}$$

If (52) is inserted into (50), the following is obtained:

$$\begin{aligned}\dot{V}_c &= \tau_c \left(q(t) \sum_{i=1}^I \sum_{j=1}^J (\dot{f}_{ij} \widetilde{W}_{ij} + f_{ij} \dot{\widetilde{W}}_{ij}) \right. \\ &\quad \left. + (1-q(t)) \sum_{i=1}^I \sum_{j=1}^J (\dot{f}_{ij} \widetilde{W}_{ij} + f_{ij} \dot{\widetilde{W}}_{ij}) \right. \\ &\quad \left. + \dot{q}(t) \sum_{i=1}^I \sum_{j=1}^J f_{ij} (\widetilde{W}_{ij} - \widetilde{W}_{ij}) + \dot{\tau} \right) \\ &= \tau_c \left(\sum_{i=1}^I \sum_{j=1}^J \left(q(t) \widetilde{W}_{ij} \dot{f}_{ij} + (1-q(t)) \widetilde{W}_{ij} \dot{f}_{ij} \right) \right. \\ &\quad \left. + \dot{q}(t) \sum_{i=1}^I \sum_{j=1}^J f_{ij} (\widetilde{W}_{ij} - \widetilde{W}_{ij}) + \dot{\tau} \right) \\ &= \tau_c \left(\sum_{i=1}^I \sum_{j=1}^J \dot{f}_{ij} \left(q(t) \widetilde{W}_{ij} + (1-q(t)) \widetilde{W}_{ij} \right) \right. \\ &\quad \left. + \dot{q}(t) \sum_{i=1}^I \sum_{j=1}^J f_{ij} (\widetilde{W}_{ij} - \widetilde{W}_{ij}) + \dot{\tau} \right). \quad (53)\end{aligned}$$

If (31) and (32) are inserted to aforementioned equation, the following can be obtained:

$$\dot{V}_c = \tau_c \left(-2\alpha \text{sgn}(\tau_c) + \dot{\tau} \right) < \left(-2\alpha |\tau_c| + |\tau_c| B_{\dot{\tau}} \right) < 0. \quad (54)$$

APPENDIX B PROOF OF THEOREM 2

Evaluating the time derivative of the Lyapunov function in (36) yields

$$\begin{aligned}\dot{V}_p &= \dot{S}_p S_p = \frac{1}{k_D} \dot{S}_c S_c \\ &\leq \frac{|\tau_c|}{k_D} (-\alpha + B_{\dot{\tau}}) < 0, \forall S_c, S_p \neq 0. \quad (55)\end{aligned}$$

ACKNOWLEDGMENT

The authors would like to thank S. Akpinar for his technical support for the preparation of the experimental setup.

REFERENCES

- [1] M. Kawato, Y. Uno, M. Isobe, and R. Suzuki, "Hierarchical neural network model for voluntary movement with application to robotics," *IEEE Contr. Syst. Mag.*, vol. 8, no. 2, pp. 8–15, Apr. 1988.
- [2] A. Topalov, E. Kayacan, Y. Oniz, and O. Kaynak, "Adaptive neuro-fuzzy control with sliding mode learning algorithm: Application to antilock braking system," in *Proc. 7th Asian Contr. Conf.*, 2009, pp. 784–789.
- [3] S. Beyhan and M. Alci, "Extended fuzzy function model with stable learning methods for online system identification," *Int. J. Adapt. Control Signal Process.*, vol. 25, no. 2, pp. 168–182, 2011.
- [4] K. Astrom and B. Wittenmark, *Adaptive Control*. New York, NY, USA: Addison-Wesley, 1995.
- [5] A. Topalov and O. Kaynak, "Online learning in adaptive neurocontrol schemes with a sliding mode algorithm," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 31, no. 3, pp. 445–450, Dec. 2001.
- [6] A. Celikyilmaz and I. B. Turksen, "Uncertainty modeling of improved fuzzy functions with evolutionary systems," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 38, no. 4, pp. 1098–1110, Aug. 2008.
- [7] S. P. Panigrahi, S. K. Nayak, and S. K. Padhy, "A genetic-based neuro-fuzzy controller for blind equalization of time-varying channels," *Int. J. Adapt. Contr. Signal Process.*, vol. 22, no. 7, pp. 705–716, 2008.
- [8] G. Parma, B. Menezes, and A. Braga, "Sliding mode algorithm for training multilayer artificial neural networks," *Electron. Lett.*, vol. 34, no. 1, pp. 97–98, 1998.
- [9] Y. Shuanghe, X. Y., and Z. M., "Fuzzy sets and systems," *Fuzzy Neural Netw. Approx. Fast Terminal Sliding Mode Appl.*, vol. 148, no. 3, pp. 469–486, 2004.
- [10] J. J. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1991.
- [11] X. Yu and O. Kaynak, "Sliding-mode control with soft computing: A survey," *IEEE Trans. Ind. Electron.*, vol. 56, no. 9, pp. 3275–3285, Sep. 2009.
- [12] O. Kaynak, K. Erbatur, and M. Ertugrul, "The fusion of computationally intelligent methodologies and sliding-mode control-a survey," *IEEE Trans. Ind. Electron.*, vol. 48, no. 1, pp. 4–17, Feb. 2001.
- [13] Y. Oniz, E. Kayacan, and O. Kaynak, "A dynamic method to forecast the wheel slip for antilock braking system and its experimental evaluation," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 2, pp. 551–560, Apr. 2009.
- [14] 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 J. Dyn. Syst., Meas., Contr.*, vol. 122, no. 4, pp. 632–640, Dec. 2000.
- [15] B. Yoo and W. Ham, "Adaptive fuzzy sliding mode control of nonlinear system," *IEEE Trans. Fuzzy Syst.*, vol. 6, no. 2, pp. 315–321, May 1998.
- [16] L. Zadeh, "Toward extended fuzzy logic-A first step," *Fuzzy Sets Syst.*, vol. 160, pp. 3175–3181, 2009.
- [17] Q. Liang, N. Karnik, and J. Mendel, "Connection admission control in ATM networks using survey-based type-2 fuzzy logic systems," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 38, no. 3, pp. 329–339, Aug. 2000.
- [18] C.-F. Juang and C.-H. Hsu, "Reinforcement interval type-2 fuzzy controller design by online rule generation and q-value-aided ant colony optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 39, no. 6, pp. 1528–1542, Dec. 2009.
- [19] C.-F. Juang and C.-H. Hsu, "Reinforcement ant optimized fuzzy controller for mobile-robot wall-following control," *IEEE Trans. Ind. Electron.*, vol. 56, no. 10, pp. 3931–3940, Oct. 2009.
- [20] R. Sepulveda, P. Melin, A. Rodriguez, A. Mancilla, and O. Montiel, "Analyzing the effects of the footprint of uncertainty in type-2 fuzzy logic controllers," *Eng. Lett.*, vol. 13, pp. 138–147, 2006.
- [21] M. Roopaei, B. Sahraei, T.-C. Lin, and M.-C. Chen, "Synchronization of two different chaotic systems using chattering-free adaptive interval type-2 fuzzy sliding mode control," in *Proc. 5th IEEE Conf. Ind. Electron. Appl.*, Jun. 2010, pp. 121–126.
- [22] M. Biglarbegian, W. Melek, and J. Mendel, "Design of novel interval type-2 fuzzy controllers for modular and reconfigurable robots: Theory

and experiments," *IEEE Trans. Ind. Electron.*, vol. 58, no. 4, pp. 1371–1384, Apr. 2011.

- [23] M. Khanesar, E. Kayacan, M. Teshnehlab, and O. Kaynak, "Analysis of the noise reduction property of type-2 fuzzy logic systems using a novel type-2 membership function," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 5, pp. 1395–1406, Oct. 2011.
- [24] E. Kayacan, W. Saeys, E. Kayacan, H. Ramon, and O. Kaynak, "Intelligent control of a tractor-implement system using type-2 fuzzy neural networks," in *Proc. IEEE WCCI World Congr. Computat. Intell.*, Brisbane, Australia, Jun. 10–15, 2012, pp. 171–178.
- [25] M. Karkee and B. L. Steward, "Study of the open and closed loop characteristics of a tractor and a single axle towed implement system," *J. Terramech.*, vol. 47, no. 6, pp. 379–393, 2010.
- [26] D. Piyabongkarn, R. Rajamani, J. A. Grogg, and J. Y. Lew, "Development and experimental evaluation of a slip angle estimator for vehicle stability control," *IEEE Trans. Contr. Syst. Technol.*, vol. 17, no. 1, pp. 78–88, Jan. 2009.
- [27] C. Geng, L. Mostefai, M. Dena, and Y. Hori, "Direct yaw-moment control of an in-wheel-motored electric vehicle based on body slip angle fuzzy observer," *IEEE Trans. Ind. Electron.*, vol. 56, no. 5, pp. 1411–1419, May 2009.
- [28] F. N. Martins, W. C. Celeste, R. Carelli, M. Sarcinelli-Filho, and T. F. Bastos-Filho, "An adaptive dynamic controller for autonomous mobile robot trajectory tracking," *Contr. Eng. Pract.*, vol. 16, no. 11, pp. 1354–1363, 2008.
- [29] E. Kayacan and O. Kaynak, "Sliding mode control theory-based algorithm for online learning in type-2 fuzzy neural networks: application to velocity control of an electro hydraulic servo system," *Int. J. Adapt. Contr. Signal Process.*, vol. 26, no. 7, pp. 645–659, 2012.
- [30] M. B. Begian, W. W. Melek, and J. M. Mendel, "Parametric design of stable type-2 TSK fuzzy systems," in *Proc. North Amer. Fuzzy Inf. Process. Syst.*, 2008, pp. 1–6.
- [31] V. I. Utkin, *Sliding Modes in Control Optimization*. New York, NY, USA: Springer-Verlag, 1992.
- [32] J.-J. Slotine and W. Li, *Applied Nonlinear Control*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1991.
- [33] R. G. Brown and P. Y. C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering*, 4th ed. New York, NY, USA: Wiley, 2012.



Erdal Kayacan (S'06–M'11–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, and the M.Sc. degree in systems and control engineering and the Ph.D. degree in electrical and electronic engineering from Bogazici University, Istanbul, Turkey, in 2006 and 2011, respectively. He is currently working toward the Postdoctoral degree at the University of Leuven (KU Leuven), Leuven, Belgium, in the Division of Mechatronics, Biostatistics and Sensors

(MeBioS).

His research areas include robotics, mechatronics, soft computing methods, sliding-mode control, and model predictive control.

Dr. Kayacan is active in the IEEE CIS Student Activities Subcommittee, the IEEE CIS Social Media Subcommittee, the IEEE CIS GOLD Subcommittee, and the IEEE SMC Technical Committee on Grey Systems. He has been serving as an Editor for the *Journal on Automation and Control Engineering* and the Editorial Advisory Board of *Grey Systems Theory and Application*.



Erkan Kayacan (S'12) was born in Istanbul, Turkey, on April 17, 1985. He received the B.Sc. and 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 and is a Research Assistant at the University of Leuven (KU Leuven), Leuven, Belgium, in the Division of Mechatronics, Biostatistics and Sensors (MeBioS).

His research interests include model predictive control, moving horizon estimation, distributed and decentralized control, intelligent control, and vehicle dynamics.

Mr. Kayacan is active in the IEEE CIS GOLD Subcommittee and the IEEE Webinars Subcommittee.



Herman Ramon received the Graduate degree in agricultural engineering from Gent University, Gent, Belgium. In 1993, he received the Ph.D. degree in applied biological sciences from the KU Leuven, Leuven, Belgium.

He is currently a Professor with the Faculty of Agricultural and Applied Biological Sciences at KU Leuven, lecturing on agricultural machinery and mechatronics 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 nonfood materials, from the field to the end user. He is the author or coauthor of more than 40 papers.



Okyay Kaynak (M'80–SM'90–F'03) received the B.Sc. degree with first class honors and the Ph.D. degree in electronic and electrical engineering from the University of Birmingham, Birmingham, U.K., in 1969 and 1972, respectively.

From 1972 to 1979, he held various positions in industry. In 1979, he joined the Department of Electrical and Electronics Engineering, Bogazici University, Istanbul, Turkey, where he is currently a Full Professor, holding the UNESCO Chair on Mechatronics. He has held long-term (near to or more than a year)

Visiting Professor/Scholar positions at various institutions in Japan, Germany, the U.S., and Singapore. His current research interests are in the fields of intelligent control and mechatronics. He has authored three books and edited five and authored or coauthored almost 400 papers that have appeared in various journals and conference proceedings. Currently, he is the Editor-in-Chief of the IEEE/ASME TRANSACTIONS ON MECHATRONICS.

Dr. Kaynak has served on many committees of the IEEE and was the President of the IEEE Industrial Electronics Society from 2002 to 2003.



Wouter Saeys received the Ph.D. degree from the University of Leuven, Leuven, Belgium.

He is currently an Assistant Professor of Biosystems Engineering with the Department of Biosystems at the University of Leuven. He was a Visiting Postdoctoral Researcher at the School for Chemical Engineering and Advanced Materials of the University of Newcastle upon Tyne, Newcastle upon Tyne, U.K., and at the Norwegian Food Research Institute—Nofima Mat, Ås, Norway. His main research interests include optical sensing, process monitoring, and control with applications in food and agriculture. He is author of 50 articles and a Member of the Editorial Board of *Biosystems Engineering*.

control with applications in food and agriculture. He is author of 50 articles and a Member of the Editorial Board of *Biosystems Engineering*.