IFAC Conference on Advances in PID Control
PID'12
Brescia (Italy), March 28-30, 2012

Data–Driven Design of a PI Fuzzy Controller for a Wind Turbine Simulated Model

Silvio Simani *

* *Department of Engineering, University of Ferrara, Via Saragat 1E, 44122 Ferrara (FE) ITALY (e–mail: silvio.simani@unife.it).*

Abstract: This paper proposes a fuzzy modelling and identification approach oriented to the design of a PI fuzzy controller for regulating both the pitch angle and the reference torque of a wind turbine model. This strategy has been suggested for enhancing the regulator design that could represent an alternative to the standard switching controller, already implemented in the wind turbine test system. The controller project requires the knowledge of the dynamic model of the wind turbine, which is achieved by means of a fuzzy modelling and identification scheme. On the other hand, the proposed PI fuzzy controller structure is straightforward and easy to implement with respect to different strategies proposed in literature. Moreover, by means of these design procedures, the proposed strategy is also able to provide a robust and reliable controller. The results obtained with the designed PI fuzzy controller are compared to those of a switching controller already implemented for the wind turbine benchmark.

Keywords: Fuzzy modelling and identification, control system design, PI regulator, nonlinear systems, wind turbine benchmark.

1. INTRODUCTION

Modern technological systems rely on sophisticated control systems to meet increased performance and safety requirements. A conventional feedback control design for a complex system may result in an unsatisfactory performance, or even instability, in the event of malfunctions in actuators, sensors or other system components. To overcome such weaknesses, new approaches to control system design have been developed in order to tolerate uncertainty and modelling errors, while maintaining desirable stability and performance properties. This is particularly important for complex processes, where the consequences of control performance degradation can be catastrophic. Therefore, the demand on reliability and robustness is generally high. It is necessary to design control systems, which are capable of tolerating potential process errors and uncertainty, in order to improve the reliability, while providing a desirable performance.

The strategy proposed in this work, focusing on the application of fuzzy modelling and identification to model–based control design, has gained increasing attention in both theory and application (Babuška (1998)). Since a mathematical model is a description of system behaviour, accurate modelling for a complex nonlinear system can be very difficult to achieve in practice. Sometimes for nonlinear systems it can be impossible to describe them via analytical equations. Moreover, very often, the system structure, parameters, and measurements are not precisely known. Thus, parametric model identification represents an alternative for developing experimental models of complex systems, such as wind turbine systems (Odgaard and Stoustrup (2009)). In contrast to pure nonlinear identification methods, where detailed knowledge about the model's structure is required, fuzzy systems are capable of deriving nonlinear models directly from measured input–output data without detailed system assumptions (Babuška (1998)). Because of these considerations, this paper suggests to use the fuzzy system theory, since it seems to be a natural tool to handle uncertain conditions and measurements (Babuška (1998)). Thus, instead of exploiting complicated nonlinear models obtained by modelling techniques, it is suggested to describe the plant under investigation by a collection of local affine systems of the type of Takagi–Sugeno (TS) fuzzy prototypes (Takagi and Sugeno (1985)), whose parameters are obtained by identification procedures.

Regarding the controller design, classical linear control methods, such as Proportional Integral Derivative (PID) control, sometimes cannot guarantee a satisfactory behaviour at each operating point of a process, due to system nonlinearity, ageing of mechanical parts, environmental conditions, and uncertain measurements. Moreover, due to the behaviour of wind turbines along their nominal operating trajectory, wind turbine controllers typically consist of multiple gain–scheduled controllers, which are designed to operate in the proximity of a certain operating point (Odgaard et al. (2009)). These examples exploit almost classical controllers, or gain–scheduling approaches, whose design sometimes cannot be direct and straightforward.

In this paper, a fuzzy control approach for the adjustment of both the wind turbine blade pitch angle and the generator torque is proposed, and applied to the particular benchmark. In more detail, the design of the controller is performed according to the following steps. Firstly, a

PI regulator is devised using the classic Ziegler–Nichols method. Then, the corresponding PI fuzzy controller is built, by means of a suitable choice of the gains. The Membership Functions (MF)s and rules are derived directly from the identified TS fuzzy models. The fuzzy identification procedure is able also to provide good robustness properties for the designed regulator. Finally, the effectiveness of the proposed fuzzy modelling and control strategies are assessed on data sequences acquired from the considered benchmark, and compared with the results achieved in (Odgaard et al. (2009)).

The paper has the following structure. Section 2 provides an overview of the wind turbine system. Section 3 recalls the fuzzy modelling and identification strategy exploited in this work. The proposed fuzzy controller design and the tuning strategy are presented in Section 3.2. The achieved results are summarised in Section 4, whilst Section 5 ends the paper by highlighting the main achievements of the work.

2. WIND TURBINE MODELLING

The three blade horizontal axis turbine considered in this paper works by the principle that the wind is acting on the blades, and thereby moving the rotor shaft. In order to up-scale the rotational speed to the needed one at the generator, a gear box is introduced. A more accurate description of the benchmark model and of the measurements available can be found in (Odgaard et al. (2009); Odgaard and Stoustrup (2009)).

2.1 Model Description

The wind turbine model is briefly recalled in this section in the continuous–time domain, and subsequently approximated as identified fuzzy discrete–time prototype.

The aerodynamic model is defined as in (1):

$$\tau_{aero}(t) = \frac{\rho\, A\, C_p\,(\beta(t),\, \lambda(t))\; v^3(t)}{2\,\omega_r(t)} \tag{1}$$

where ρ is the density of the air, A is the area covered by the turbine blades in its rotation, $\beta(t)$ is the pitch angle of the blades, $v(t)$ the wind speed, whilst $\lambda(t)$ is the tip–speed ratio of the blade, defined as:

$$\lambda(t) = \frac{\omega_r(t)\, R}{v(t)} \tag{2}$$

with R the rotor radius. C_p represents the power coefficient, here described by means of a two–dimensional map (look–up table) (Odgaard et al. (2009)). Equation (1) is used to estimate $\tau_{aero}(t)$ based on an assumed estimated $v(t)$, and measured $\beta(t)$ and $\omega_r(t)$. Due to the uncertainty of the wind speed, the estimate of $\tau_{aero}(t)$ is considered affected by an unknown measurement error, which can be estimated by means of the approach described in Section 3.1. Moreover, the nonlinearity represented by the relations (1) and (2), and the exploited wind turbine control strategy, have motivated the modelling approach suggested in Section 3.

A simple one–body model is used to represent the drive train, in the following form (Odgaard and Stoustrup (2009)):

$$\dot{\omega}_r(t) = \frac{1}{J}\,(\tau_{aero}(t) - \tau_g(t)) \tag{3}$$

where:

$$\dot{\tau}_g(t) = p_{gen}\,(\tau_{ref}(t) - \tau_g(t)) \tag{4}$$

The generator torque $\tau_g(t)$, and the reference $\tau_{ref}(t)$ are in this context transformed to the low speed side of the drive train (rotor side), whilst p_{gen} is the generator power coefficient. With these assumptions, the complete continuous–time description of the system under diagnosis has the following form:

$$\begin{cases} \dot{x}_c(t) \;=\; f_c\,(x_c(t), u(t)) \\ y(t) \quad = x_c(t) \end{cases} \tag{5}$$

where $u(t) = [\beta_r(t),\, \tau_g(t)]^T$ and $y(t) = x_c(t) = [P_g(t),\, \omega_g(t)]^T$ are the input and the monitored output measurements, respectively. $f_c(\cdot)$ represents the continuous–time nonlinear function that will be approximated via the discrete–time fuzzy prototype from N sampled data $u(t)$ and $y(t)$, with $t = 1, 2, \ldots N$ presented in Section 3.

2.2 Control System for Wind Turbines

The controller for a wind turbine operates in principle in four zones. Zone 1 is start–up of the turbine, zone 2 is power optimisation, zone 3 corresponds to constant power production, and zone 4 is high wind speed. Since the focus of the benchmark model is on the normal operation, *only zone 2 and zone 3 are considered* (Odgaard et al. (2009)). In zone 2 the turbine is controlled to obtain optimal power production. The optimal power is obtained if the blade pitch angle β_r is equal 0 degrees, and if the tip speed ratio is constant at its optimal value λ_{opt}.

The tip speed ratio, λ, as already described by (2), can be written as in (6), where R is the radius of the blades, v_ω is the wind speed, and ω_r is the angular rotor speed:

$$\lambda = \frac{\omega_r\, R}{v_\omega} \tag{6}$$

The optimal value of λ, which is denoted with λ_{opt}, is determined as the optimum point in the power coefficient mapping of the wind turbine. This optimal value is achieve by setting the reference torque to the converter, $\tau_{g\,r}$.

The torque in this power optimisation zone can be written as:

$$\tau_{g\,r} = K_{opt}\,\omega_r^2 \tag{7}$$

where:

$$K_{opt} = \frac{1}{2}\,\rho\, A\, R^3\, \frac{C_{p_{\max}}}{\lambda_{opt}^3} \tag{8}$$

with ρ the air density, A the area swept by the turbine blades, $C_{p_{\max}}$ the maximal value of C_p (*i.e.* the power coefficient map), related the to λ_{opt}, *i.e.* the optimal tip–speed ratio.

Then the power reference is achieved and controller is switched to control zone 3. In this zone the control objective consists of following the power reference, P_r, which is obtained by controlling β_r, such that the C_p is decreased. In an industrial control scheme, a PI controller is used to keep ω_r at the prescribed value by changing β_r. The second control input is $\tau_{g\,r}$, whose value is computed by using (7) with the optimal gain derived from (8).

2.3 Benchmark Original Control Strategy

The wind turbine controller in this simulation model works in two regions as recalled in Section 2.2. *Region 1* is denoted power optimisation and *region 2* is the power reference following. The controller is implemented with a sample frequency at 100 Hz. The controller starts in mode 1 (region 1). Therefore, the control mode should switch from mode 1 to mode 2 if:

$$P_g(k) \geq P_r(k) \text{ and } \omega_g(k) \geq \omega_{nom} \tag{9}$$

where k indicates the acquired discrete–time measurements from the corresponding continuous–time signals, whilst ω_{nom} is the nominal turbine speed. On the other hand, the control switches from mode 2 to mode 1 if:

$$\omega_g(k) < \omega_{nom} - \omega_\Delta \tag{10}$$

where ω_Δ is a number that introduces hysteresis to ensure a minimum time between transitions. In particular, regarding the control *mode 1*, the converter reference signal is defined as in (7) and (8), and $\beta_{ref} = 0$, *i.e.*:

$$\begin{cases} \tau_{g\,r} &= K_{opt}\,\omega_r^2 \\ K_{opt} &= \dfrac{1}{2}\,\rho\,A\,R^3\,\dfrac{C_{p_{\max}}}{\lambda_{opt}^3} \end{cases}$$

On the other hand, for the control *mode 2*:

$$\begin{cases} \beta_r(k) &= \beta_r(k-1) + k_p\,e(k) + (k_i\,T_s - k_p)\;e(k-1) \\ e(k) &= \omega_r(k) - \omega_{nom} \end{cases} \tag{11}$$

where $k_i = 1$ and $k_p = 4$ according to the benchmark parameter tuning settled as described in (Odgaard and Stoustrup (2009)).

3. FUZZY MODELLING FOR CONTROL

This section describes the fuzzy modelling and identification scheme, briefly recalled in Section 3.1, which enhances the design procedure of the proposed fuzzy controller, as shown in Section 3.2.

3.1 Fuzzy Identification from Data Clustering

The modelling approach exploited in this work relies on the identification of transparent rule–based fuzzy models, which can accurately predict the quantities of interest, and at the same time provide insight into the system that generated the data. Attention is paid to the selection of appropriate model structures in terms of the dynamic properties, as well as the internal structure of the fuzzy rules (in particular, Takagi–Sugeno type) (Takagi and Sugeno (1985)). From the system identification point of view, a fuzzy model is regarded as a composition of local affine sub models. Fuzzy sets naturally provide smooth transitions between the submodels, and enable the integration of various types of knowledge within a common framework.

In order to generate fuzzy models automatically from measurements, a comprehensive methodology is used. This employs fuzzy clustering techniques to partition the available data into subsets characterised by a linear behaviour. The relationships between the presented identification method and linear regression are exploited, allowing for the combination of fuzzy logic techniques with system identification tools. In addition, the implementation in the Matlab ® toolbox of the Fuzzy Modelling and IDentification (FMID) techniques presented in the following is available (Babuška (2000)). Fuzzy identification usually refers to methods and algorithms for constructing fuzzy models from data.

In this study, fuzzy models are viewed as a class of local modelling approaches, which attempt to solve a complex modelling problem by decomposing it into a number of simpler sub–problems. The theory of fuzzy sets offers an excellent tool for representing the uncertainty associated with the decomposition task, for providing smooth transitions between the individual local sub models, and for integrating various types of knowledge within one common framework. In particular, fuzzy logic is exploited to define a TS fuzzy model (Takagi and Sugeno (1985)).

A large part of fuzzy modelling and identification algorithms (see *e.g.* (Babuška (1998)) and references therein) share a common two–step procedure, in which at first, the operating regions are determined using heuristics or data clustering techniques. Then, in the second stage, the identification of the parameters of each submodel is achieved using the identification algorithm in particular proposed by the author (Simani et al. (1999)), which can be seen as a generalisation of classical least–squares. From this perspective, fuzzy identification can be regarded as a search for a decomposition of a nonlinear system, which gives a desired balance between the complexity and the accuracy of the model, effectively exploring the fact that the complexity of systems is usually not uniform. Since it cannot be expected that sufficient prior knowledge is available concerning this decomposition, methods for automated generation of the decomposition, primarily from system data, are developed. A suitable class of fuzzy clustering algorithms can be used for this purpose, and in particular, the well–known Gustafson–Kessel (GK) fuzzy clustering is exploited in this work (Babuška (1998)), since already implemented in (Babuška (2000)).

The fuzzy rule–based model suitable for the approximation of a large class of nonlinear systems was introduced by Takagi and Sugeno (TS) (Takagi and Sugeno (1985)). In the TS fuzzy model, the rule consequents are crisp functions of the model inputs:

$$R_i: \;\; \textbf{IF } \mathbf{x}(k) \text{ is } A_i \text{ THEN } y_i = f_i\big(\mathbf{x}(k)\big) \tag{12}$$

where $i = 1, 2, \cdots, M$, $\mathbf{x}(k) \in \Re^p$ is the input (antecedent) variable and $y_i \in \Re$ is the output (consequent) variable. R_i denotes the i–th rule, and M is the number of rules in the rule base. A_i is the antecedent fuzzy set of the i–th rule, defined by a (multivariate) membership function. The consequent functions f_i are typically chosen as instances of a suitable parameterised function, whose structure remains equal in all the rules and only the parameters vary. A simple and practically useful parameterisation is the affine form:

$$y_i = \mathbf{a}_i\, \mathbf{x} + b_i, \tag{13}$$

where a_i is a parameter vector, and b_i is a scalar offset. This model is referred to as *affine TS model*, and it can be written as (Takagi and Sugeno (1985)):

$$y = \frac{\sum_{i=1}^{M} \mu_i(\mathbf{x})\, y_i}{\sum_{i=1}^{M} \mu_i(\mathbf{x})} \tag{14}$$

Therefore, the effective approach for the identification of complex systems consists of partitioning the available data into subsets and approximate each subset by a simpler affine model. Therefore, fuzzy clustering represents the tool to obtain a partition of data where the transitions between the subsets are gradual rather than abrupt. The *antecedent* fuzzy sets A_i can be computed analytically in the antecedent product space, or can be extracted from the fuzzy partition matrix (Babuška (2000, 1998)). The *consequent parameters* $\mathbf{a}_i$ and b_i are estimated from the data using the method developed by the author (Simani et al. (1999)), and recalled below. This identification scheme exploited for the estimation of the TS model parameters has been integrated into the FMID toolbox for Matlab® by the author.

As already remarked, the set of optimal parameters $\mathbf{a}_i$ and b_i with respect to the model outputs are estimated by using the procedure developed in (Simani et al. (1999)). This approach can be formulated as minimisation of the total prediction error of the TS model. This approach developed by the author is usually preferred when the TS model should serve as predictor (Simani et al. (1999)), and it computes the consequent parameters by the so–called *Frisch scheme*. After the clustering of the data has been obtained via the GK algorithm, data subsets are processed according the Frisch scheme identification procedure (Simani et al. (1999)), in order to estimate the TS parameters for each affine submodels.

Note that the estimation procedure applied here is able to provide the required robustness and reliability properties. To this aim, the GK fuzzy clustering algorithm has been applied to two different data sets. The first data set is the so–called estimation data, whilst the second one is the so–called validation data. The optimal number of clusters M, the estimated MFs, and the set of the optimal parameters $\mathbf{a}_i$ and b_i have been determined on the basis of the minimisation of the reconstruction error, that is the difference between the actual output and the one from the identified TS fuzzy model. In this way, the optimal TS prototype, which is able to model both the data sets, is determined.

3.2 Fuzzy Controller Design

The proposed fuzzy logic controller is fed by the error signal $e(k)$, *i.e.* the tracking error defined as the difference between the considered set–point $r(k)$ and the plant controlled output $y(k)$ at the sample k:

$$e(k) = r(k) - y(k) \tag{15}$$

The fuzzy PI controller uses a second input signal, defined as the sum of the system errors, which is computed using the following expression:

$$\delta e(k) = \sum_{i=1}^{k} e(i) \tag{16}$$

It is known from digital control theory that the most frequently used digital PI control algorithm can be described as follows:

$$u(k) = k_p\, e(k) + k_i\, \delta e(k) \tag{17}$$

where $k_i = k_p \frac{T_s}{T_i}$, T_s is the sampling time, T_i is the integral time constant of the conventional controller, k_p is the proportional gain, and $u(k)$ is the output control action.

The Sugeno's fuzzy rules into the fuzzy PI controller can be composed in the generalised form of "IF–THEN" composition with a premise and an antecedent part to describe the control policy. The rule base comprises a collection of M rules, where the index (j) represents the rule number:

$$\begin{aligned} &R_j: \ \ \textbf{IF}\ \mathbf{x}(k) \text{ is } A_j \text{ THEN } f_u^{(j)}(k) = \\ &= K_P^{(j)}\, e(k) + K_I^{(j)}\, \delta e(k) \quad j = 1, 2, \cdots, M, \end{aligned} \tag{18}$$

where $e(k)$ and $\delta e(k)$ are the input variables. In this expression, a similarity between the equation of the conventional digital PI controller (17) and the Sugeno's output function (18) can be found. In this case, the fuzzy PI controller is considered as a collection of several local PI controllers, which are represented by the Sugeno's functions into the different fuzzy rules. In this way, it is possible to approximate the nonlinear characteristic of the controlled plant.

For a discrete universe with M quantisation levels in the fuzzy output, the control action $u = u_F$ is expressed as a weighted average of the Sugeno's output functions f_u and their membership degrees μ_i of the quantisation levels, with $i = 1, \ldots, M$. Also in this case, before the output can be inferred, the degree of fulfilment of the antecedent denoted by $\mu_i(\mathbf{x})$ must be computed. Thus, the degree of fulfilment is simply equal to the membership degree of the given input $\mathbf{x}$, *i.e.*, $\mu_i = \mu_{A_i}(\mathbf{x})$. By recalling the identified Takagi–Sugeno model, the inference is reduced to a simple expression, similar to the fuzzy–mean defuzzification formula (Babuška (1998)):

$$u_F = \frac{\sum_{j=1}^{M} \mu_j(\mathbf{x})\, f_u^{(j)}}{\sum_{i=1}^{M} \mu_j(\mathbf{x})} \tag{19}$$

or, by substituting the expression of the fuzzy PI terms:

$$u_F(k) = \frac{\sum_{j=1}^{M} \mu_j\Big(\mathbf{x}(k)\Big)\Big(K_P^{(j)}\, e(k) + K_I^{(j)}\, \delta e(k)\Big)}{\sum_{i=1}^{M} \mu_j\Big(\mathbf{x}(k)\Big)} \tag{20}$$

where the time dependence at the instant k has been highlighted. It is worth noting that the PI controller parameters $K_P^{(j)}$ and $K_I^{(j)}$ (with $j = 1, \ldots, M$) are settled according to the Ziegler–Nichols rules applied to the identified local linear TS submodels. Then, in order to obtain a quick reaction to set–point variations, gain scheduling of the fuzzy regulator parameters is performed depending on the error, as shown by (20).

The second step consists in building the fuzzy controller of (20). The input MFs $\mu_j(\mathbf{x})$ coincide with the ones of the identified TS model, as described in (Babuška (1998)). The number of the input MFs determines the number of rules and output MFs. In this work, the optimal number of rules M is equal to the minimal number of clusters used to identify the nonlinear system, as described at the end of 3.1. Finally, the adopted fuzzy operators are the product as AND operator, the bounded sum as OR operator, MIN as implication method, the Center of Gravity (COG) as defuzzification method.

Also in this case, it is worth highlighting the strategy applied for achieving the required robustness characteristics. With reference to (20), the PI controller parameters $K_P^{(j)}$ and $K_I^{(j)}$ are tuned via the Ziegler–Nichols rules, applied to the identified local linear TS submodels, and by considering the *validation* data set. Therefore, the optimal controller performances with respect to set–point variations are validated and enhanced for different working conditions. In this way, if both the TS model identification and fuzzy regulator tuning procedures are properly preformed, the gain scheduling mechanism of the fuzzy regulator parameters leads to good robustness properties.

Note finally that, even if the simple Ziegler–Nichols rules have been exploited, the implementation of the classical PI controllers as a unique fuzzy regulator provides the required properties of the proposed strategy. Thus, Section 4 will show the achieved results regarding the fuzzy PI controller parameter tuning using the data sequences from the wind turbine benchmark.

4. EXPERIMENTAL RESULTS

This section describes the experimentations with the methods proposed for the fuzzy modelling technique oriented to the design of the fuzzy controller relying on the multiple–model TS approach.

The GK clustering algorithm with $M = 3$ clusters and a number of shifts $n = 2$ was applied to the estimation and validation data sets $\{P_g(t),\, \omega_g(t),\, \beta_r(t)\}$. On the other hand, a number of clusters $M = 3$ and $n = 2$ was considered for achieving a suitable clustering of the sampled data sets $\{P_g(t),\, \omega_g(t),\, \tau_g(t)\}$. After clustering, the TS model parameters for each output were estimated. Therefore, the i–th output $y(t)$ of the wind turbine ($i = 1, \ldots, m$ and $m = 2$) continuous–time model (5) is approximated by a Takagi–Sugeno fuzzy Multiple–Input Single–Output (MISO) discrete–time prototype (14) with $r = 2$ inputs. The fuzzy multiple models are able to approximate the process under diagnosis quite accurately.

Using this identified TS fuzzy prototype, the model–based approach for determining the fuzzy controller is exploited and applied to the actual wind turbine benchmark. According to Section 3.2, the parameters of the fuzzy PI controllers have been computed. In particular, as the identified TS models for each output consists of a fuzzy collection of 3 MISO second order ($n = 2$) models, the regulator parameters in (18) are computed analytically.

In more detail, by considering a second order local model described by its identified parameters $\mathbf{a}_i = \left[\alpha_2^{(i)}\,,\, \alpha_1^{(i)}\right]^T$ and $b_i \left[\delta_2^{(i)}\,,\, \delta_1^{(i)}\right]^T$, the so–called critical gain $K_0^{(i)}$ and critical period of oscillations $T_0^{(i)}$ required by the Ziegler–Nichols method are computed as follows (Bobál et al. (2005)):

$$\begin{cases} K_0^{(i)} = \dfrac{\alpha_1^{(i)} - \alpha_2^{(i)} - 1}{\delta_2^{(i)} - \delta_1^{(i)}} \\[2ex] T_0^{(i)} = \dfrac{2\pi T_s}{\arccos \gamma^{(i)}} \qquad \text{with } \gamma^{(i)} = \dfrac{\alpha_2^{(i)}\,\delta_1^{(i)} - \alpha_1^{(i)}\,\delta_2^{(i)}}{2\,\delta_2^{(i)}} \end{cases} \tag{21}$$

Relations (22) are thus used for calculating the parameters $K_P^{(i)}$ and $K_I^{(i)}$ of the (local) i–th PI controller of (18):

$$\left\{ K_P^{(i)} = 0.6\, K_0^{(i)} \left(1 - \frac{T_s}{T_0^{(i)}}\right) \;\; K_I^{(i)} = \frac{1.2\, K_0^{(i)}}{K_P^{(i)}\, T_0^{(i)}} \right. \tag{22}$$

where T_s is the sampling time.

In the following, the suggested fuzzy PI controllers and the original switching strategy described in Section 2.3 have been implemented and compared in the Matlab® and Simulink® environments.

The experimental set–up employs 2 MISO fuzzy PI regulators used for the control of the blade pitch angles and the generator control torque, respectively. As an example, by using the previous relations of (21) and (22), the following tuned parameter sets have been computed for the pitch angle control:

$$\begin{cases} \{K_P^{(1)},\, \cdots,\, K_P^{(3)}\} = \{4.3,\, 4.1,\, 4.2\} \\ \{K_I^{(1)},\, \cdots,\, K_I^{(3)}\} = \{1.2,\, 1.4,\, 1.5\} \end{cases} \tag{23}$$

In order to compare the advantages of the proposed fuzzy PI strategy, the obtained results are also compared with the ones achieved by using the original switching wind turbine benchmark regulator recalled in Section 2.3.

The controller capabilities have been assessed in simulation by considering different data sequences. In Tables 1 and 2, the per–cent Normalised Sum of Squared tracking Error ($NSSE$) values defined as:

$$NSSE\% = 100 \sqrt{\frac{\sum_{k=1}^{N} \left(r(k) - y(k)\right)^2}{\sum_{k=1}^{N} r^2(k)}} \tag{24}$$

are computed for both the controllers. It is worth noting that in *partial load operation*, the performance is represented by the comparison between the power produced by the generator P_g with respect to the theoretical maximum power output, P_r given the instant wind speed. On the other hand, in *full load operation* the performance depends on the generator speed ω_g with respect to the nominal one, ω_{nom}.

According to these simulation results, the robustness properties of the suggested fuzzy PI controllers seem to be reached, and they are slightly better than the original switching regulator.

Table 1. Controllers in partial load operation: $NSSE\%$ values.

Data Set	Benchmark Controller	Fuzzy PI
Estimation data	39.34%	36.36%
Validation data	42.19%	37.17%

Table 2. Controllers in full load operation: $NSSE\%$ values.

Data Set	Benchmark Controller	Fuzzy PI
Estimation data	19.53%	16.57%
Validation data	21.01%	17.85%

The robustness and reliability properties of the designed fuzzy controllers have been tested and assessed in simulation. Therefore, the parameters of the wind turbine benchmark model described in Section 2 have been modified of 20% with respect to their nominal values (Odgaard and Stoustrup (2009)). The obtained results summarised in Table 3 seem to show that the performances of the proposed fuzzy controller are almost unchanged with respect to the nominal situation.

Table 3. Fuzzy controller $NSSE\%$ values.

Data Set	Partial load	Full load
Estimation data	36.37%	16.57%
Validation data	37.19%	17.94%

As a final example, the wind turbine power $P_g(t)$, which represents one of the main controlled variables, is reported in the following. In particular, Figure 1 represents the reference variable (continuous lines) compared with the corresponding controlled signal (dashed lines) tracked by the model.

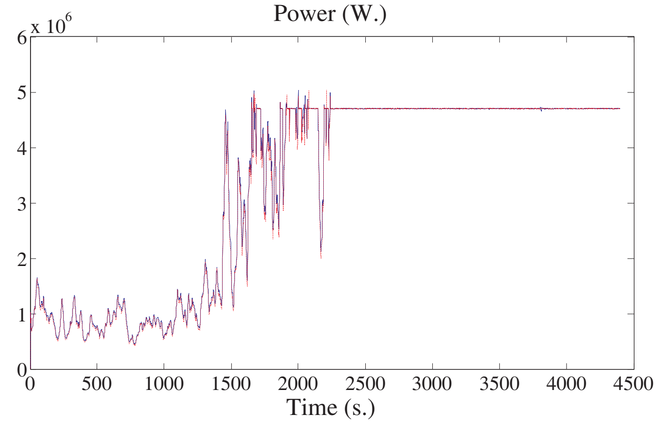

Fig. 1. Set–point (continuous line) and controlled (dashed line) power signal $P_g(t)$.

On the other hand, Figure 2 depicts the controlled generator speed $\omega_g(t)$, with respect to the corresponding set–point.

5. CONCLUSION

This paper proposes a fuzzy modelling and identification approach oriented to the design of a PI fuzzy controller for regulating both the pitch angle and the generator torque of a wind turbine benchmark. This strategy was suggested for enhancing the regulator design that could represent an alternative to standard switching controllers, already implemented for wind turbine benchmark. The control design requires the knowledge of a wind turbine model, which is achieved by means of the suggested fuzzy estimation scheme. It is shown also that the proposed controller design procedure is easy to implement. Further studies will concern the application of the suggested methods to real process data.

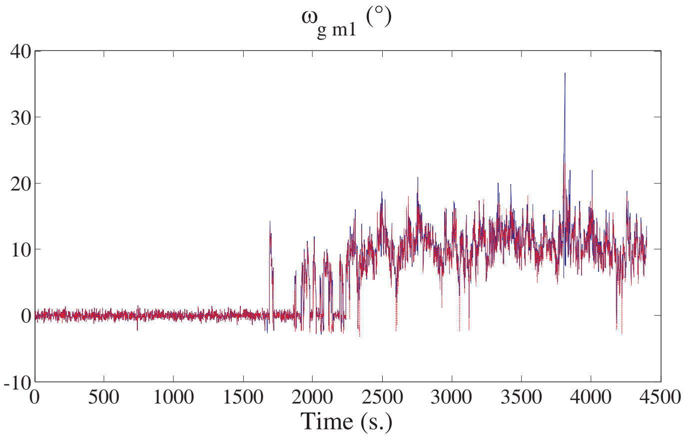

Fig. 2. Reference (continuous line) and controlled (dashed line) generator speed $\omega_g(t)$.

REFERENCES

Babuška, R. (2000). *Fuzzy Modelling and Identification Toolbox*. Control Engineering Laboratory, Faculty of Information Technology and Systems, Delft University of Technology, Delft, The Netherlands, version 3.1 edition. (Available at http://lcewww.et.tudelft.nl/ ~babuska).

Babuška, R. (1998). *Fuzzy Modeling for Control*. Kluwer Academic Publishers.

Bobál, V., Böhm, J., Fessl, J., and Machácek, J. (2005). *Digital Self–Tuning Controllers: Algorithms, Implementation and Applications*. Advanced Textbooks in Control and Signal Processing. Springer, 1st edition.

Odgaard, P.F. and Stoustrup, J. (2009). Unknown Input Observer Based Scheme for Detecting Faults in a Wind Turbine Converter. In *Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, volume 1, 161–166. IFAC – Elsevier, Barcelona, Spain. DOI: 10.3182/20090630–4–ES–2003.0048.

Odgaard, P.F., Stoustrup, J., and Kinnaert, M. (2009). Fault Tolerant Control of Wind Turbines – a Benchmark Model. In *Proceedings of the 7th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, volume 1, 155–160. Barcelona, Spain. DOI: 10.3182/20090630–4–ES–2003.0090.

Simani, S., Fantuzzi, C., Rovatti, R., and Beghelli, S. (1999). Parameter identification for piecewise linear fuzzy models in noisy environment. *International Journal of Approximate Reasoning*, 1(22), 149–167. Publisher: Elsevier.

Takagi, T. and Sugeno, M. (1985). Fuzzy identification of systems and its application to modeling and control. *IEEE Transaction on System, Man and Cybernetics*, SMC-15(1), 116–132.