Identification–Oriented Control Designs with Application to a Wind Turbine Benchmark

Silvio Simani
Department of Engineering
University of Ferrara
44122 Ferrara (FE), Italy

Paolo Castaldi
Aerospace Engineering Faculty
University of Bologna
47100 Forlì (FC), Italy

***Abstract*—Wind turbines are complex dynamic systems forced by stochastic wind disturbances, gravitational, centrifugal, and gyroscopic loads. Since their aerodynamics are nonlinear, wind turbine modelling is thus challenging. Therefore, the design of control algorithms for wind turbines must account for these complexities, but without being too complex and unwieldy. Therefore, the main contribution of this study consists of providing two examples of robust and viable control designs with application to a wind turbine simulator. Due to the description of the considered process, extensive simulations of this test case and Monte–Carlo analysis are the tools for assessing experimentally the achieved features of the suggested control schemes, in terms of reliability, robustness, and stability, in the presence of modelling and measurement errors. These developed control methods are finally compared with different approaches designed for the same benchmark, in order to evaluate the properties of the considered control techniques.**

***Keywords*—control algorithms; fuzzy modelling and control; recursive estimation; adaptive PI controllers; wind turbine model.**

I. Introduction

Wind turbines are complex nonlinear dynamic systems forced by gravity, and stochastic wind disturbance, which are affected by gravitational, centrifugal, and gyroscopic loads. Their aerodynamics are nonlinear, and unsteady, whilst their rotors are subject to complicated turbulent wind inflow fields driving fatigue loading. Therefore, wind turbine modelling and control are challenging tasks [1], [2]. Accurate models should contain many degrees of freedom in order to capture the most important dynamic effects. Moreover, the rotation of the turbine adds further complexity to the dynamics modelling. In general, off–the–shelf commercial software usually is not adequate for wind turbine dynamics modelling, but special dynamic simulation codes are required. It is clear that the design of control algorithms for wind turbines has to take into account these complexities. On the other hand, control algorithms must capture the most important turbine dynamics, without being too complex [1], [2].

Today's wind turbines employ different control actuation and strategies to achieve the required goals and performances. Some turbines perform the regulation action through passive control methods, such as in fixed–pitch, stall control machines. In these machines, the blades are designed so that the power is limited above rated wind speed through the blade stall. Thus, no pitch mechanism is needed. Rotors with adjustable pitch are often used in constant–speed machines, in order to provide turbine power control better than the one achievable with blade stall. Therefore, blade pitching can be regulated to provide constant power above rated wind speed, in order to provide good power regulation in the presence of gusts and turbulence. Large commercial wind turbines can employ also yaw control to orient the machine into the wind. A yaw error signal from a nacelle–mounted wind direction sensor is used to calculate a control error. The control signal is usually just a command to yaw the turbine at a slow constant rate in one direction or the other. The yaw motor is switched on when the yaw error exceeds a certain amount and is switched off when the yaw error is less than some prescribed amount. Some recent control studies were addressed *e.g.* in [3], [4], [5], [6]. It is worth noting that the main disadvantage of these approaches consists of the need of an accurate model of the process under investigation, followed by the control design strategies, which usually require advanced mathematical methodologies.

On the other hand, this work describes the application of two control methods, which are quite direct and straightforward, as well as their testing through extensive simulations for a wind turbine prototype, which is freely available for the Matlab® and Simulink® environments [2].

In particular, the first proposed strategy consists of a scheme relying on a fuzzy identification approach to model–based control design. In contrast to pure nonlinear identification methods, fuzzy systems are capable of deriving nonlinear models directly from measured input–output data without detailed system assumptions, with arbitrary degree of accuracy. In particular, Takagi–Sugeno (TS) fuzzy prototypes are exploited, whose parameters are obtained by identification procedures from the data of the monitored process. The suggested fuzzy approach is motivated also by previous works by one of the same authors [7]. It is worth noting that the works by one of the same author [8], [9] presented a totally different solution to the design of the fuzzy regulators. In fact, even if the papers [8], [9] and the present study share the common fuzzy clustering methodology, this contribution focuses on the direct fuzzy *regulator identification*, whilst [8], [9] were based on fuzzy PI controllers, whose parameters were computed from the identified fuzzy prototypes.

With reference to the second control method proposed in this work, the application of an on–line identification mechanism in connection with a model–based adaptive control design is considered. This control scheme belongs to the field of adaptive control. On–line parametric model identification schemes represent an alternative for developing experimental models for complex systems, such as wind turbine systems. Therefore, this paper suggests the implementation of controllers based on adaptive identification schemes, used for the on–line estimation of the controlled process, which can be affected by uncertainty and errors. The recursive Frisch extended to the adaptive case making use of exponential forgetting is considered here [10]. It also overcomes potential numerical difficulties with the existing recursive scheme. The ability of the adaptive scheme to track changes in the system parameters is exploited here in connection with the on–line computation of time–varying controller parameters, in order to maintain the required control performances. The use of this identification procedure is motivated by its easy integration into the Simulink® toolbox for the design of on–line controllers [11].

The effectiveness of the proposed control strategies has been assessed on data sequences acquired from the considered benchmark. Several simulation results show the achieved performances with respect also to different control methods specifically developed very recently for the same wind turbine benchmark [12]. In particular, three alternative control schemes are considered in this work, which are based on Unknown Input Observers (UIOs) [6], virtual sensors/actuators (VAS) [5], and LMI–based LPV controllers [13]. Since it is necessary to evaluate the impact on the designed control systems of modelling uncertainties, disturbance, and measurement errors, the overall scheme verification uses extensive Monte–Carlo simulations for the analysis and the assessment of the robustness, the stability, and their final performance evaluation. In fact, as shown in the following, the wind turbine system may contain elements that cannot be described by any analytical model obtained via first principles.

Finally, the paper has the following structure. Section II provides an overview of the wind turbine system considered in this work. Section III recalls the strategy exploited for the identification of the fuzzy controller. On the other hand, the second parameter–varying controller design is described in Section IV-A. The achieved results are summarised in Section V. Section VI ends the paper by highlighting the main achievements of the work.

II. Wind Turbine Benchmark Description

The three–blade horizontal axis turbine considered in this paper works according to 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 gearbox is introduced.

Note that a controller for a wind turbine operates in principle in four zones. Zone 1 is start–up of the turbine, zone 2 is the so-called power optimisation, zone 3 corresponds to constant power production, and zone 4 with high wind speed. Since the benchmark model works in normal operating conditions, only zone 2 and zone 3 are considered here, as described e.g. in [14]. In zone 2, which will be denoted as region 1, the turbine is controlled to obtain optimal power production. This working condition is known as partial load condition. On the other hand, zone 3 corresponds to region 2, i.e. the so-called full load working situation.

The wind turbine model in the continuous–time domain is briefly recalled in the following. The aerodynamic model is defined as in (1):

$$\tau_{aero}(t) = \rho\, A\, C_p\big(\beta_r(t), \lambda(t)\big)\, v^3(t)/2\, \omega_r(t) \qquad (1)$$

Where ρ is the density of the air, A is the area covered by the turbine blades in its rotation, $\beta_r(t)$ is the pitch angle of the blades, v(t) the wind speed, whilst $\lambda(t)$ is the tip–speed ratio of the blade [12]. $C_p(.)$ represents the power coefficient, here described by means of a two–dimensional map (look–up table) [12]. Equation (1) is used to compute $\tau_{aero}(t)$ based on an assumed estimated wind speed v(t), the measured $\beta_r(t)$ and the rotor speed $\omega_r(t)$. Due to the uncertainty of the wind speed v(t), the estimate of $\tau_{aero}(t)$ is considered affected by an unknown measurement error, which motivates the approaches proposed in this study. A simple one–body model is used to represent the drive train, whilst the hydraulic pitch model is described as a closed–loop transfer function of the hydraulic pitch system modelled as a second order transfer function [14]. The converter dynamics are modelled by a first–order transfer function, and the measurement sensors are modelled by adding the actual variable values with stochastic noise processes. These noise signals are described as Gaussian processes with fixed mean and standard deviations values, depending on the considered measurement sensors. A more accurate description of the benchmark model can be found in [12].

With these assumptions, the complete continuous–time description of the system under diagnosis has the form of (2):

$$\begin{aligned} \dot{x}_c(t) &= f_c\big(x_c(t), u(t)\big) \\ y(t) &= x_c(t) \end{aligned} \qquad (2)$$

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 control inputs and the monitored output measurements, respectively. $f_c(.)$ represents the continuous–time nonlinear function that describes the complete behaviour of the wind turbine process. Regarding the input and output signals, $\omega_g(t)$ is the generator speed measurement, $P_g(t)$ the generator power measurement, and $\tau_g(t)$ generator torque measurement. Finally, the model parameters, and the map $C_p(\beta_r, \lambda)$ are chosen in order to represent a realistic turbine, which is used as benchmark system in this study [12].

Finally, the next Section III will recall the scheme for obtaining the fuzzy description of the wind turbine controller.

III. Fuzzy Identification for Controller Design

This section recalls the approach exploited for obtaining the fuzzy description of the wind turbine controller, whilst the proposed controller model estimation is shown in Section III-A, which represents one of the main contributions of the paper.

The approach suggested in this section employs fuzzy clustering techniques to partition the available data into subsets characterised by linear behaviours. Relationships between clusters and linear regression are exploited, thus allowing for the combination of fuzzy logic techniques with system identification tools. In addition, an implementation in the Matlab® Toolbox of the Fuzzy Modelling and IDentification (FMID) technique presented in the following is available [15].

In this study, TS fuzzy models are exploited [16], as they are able to provide the mathematical description of the nonlinear system. The switching and the scheduling between the submodels is achieved through a smooth function of the system state, the behaviour of which is defined using fuzzy set theory.

In more detail, the fuzzy modelling and identification algorithm is based on a two–step procedure, in which at first, the operating regions are determined using the data clustering technique, and in particular, the Gustafson–Kessel (GK) fuzzy clustering, since already available in [15]. Then, in the second stage, the estimation of the controller parameters is achieved using the identification algorithm already proposed by one of the same authors in [7], which can be seen as a generalisation of classical least–squares.

The TS fuzzy models have the form of:

$$y(k+1) = \Sigma^{M}_{i=1}\mu_i(\mathbf{x}(k))\, y_i \,/\, \Sigma^{M}_{i=1}\mu_i(\mathbf{x}(k)) \tag{3}$$

where $y_i = \mathbf{a}^T_i + b_i$, with $\mathbf{a}_i$ the parameter vector (regressand), and b_i is the scalar offset. $\mathbf{x} = \mathbf{x}(k)$ represents the regressor vector, which can contain delayed samples of u(k) and y(k).

The *antecedent* fuzzy sets $\mu_i(.)$ are extracted from the fuzzy partition matrix [16]. The *consequent parameters* $\mathbf{a}_i$ and b_i are estimated from the data using the procedure presented in [7]. This identification scheme exploited for the estimation of the TS model parameters has been integrated into the FMID toolbox for Matlab® by one of the same authors. This approach developed by one of the same authors is usually preferred when the TS model should serve as predictor, as it computes the consequent parameters via the Frisch scheme, developed for the Errors–In–Variables (EIV) descriptions [10].

A. Fuzzy Controller Estimation

This section addresses the design of the nonlinear fuzzy controller of the wind turbine process, which relies on the *model inverse control principle*, and solved within the fuzzy identification framework.

Therefore, the fuzzy identification scheme will be used for both predicting the wind turbine behaviour and *estimating the inverse model controller structure*. An optimal control strategy is thus obtained by minimising a cost function, which includes the difference between the desired and controller outputs, and a penalty on the system stability. Constraints on the complete system stability are thus included as a part of the optimisation problem. Generally, a non–convex optimisation problem must be solved at each control sample, which hampers the direct and practical application of the approach. However, to solve this problem, the optimisation scheme described in [7], which is based on a parameterised search technique, is applied at a higher level to formulate the control objectives and constraints.

The proposed method is implemented for model with a unitary delay $n_k = 1$. For systems with larger delays, the same method can be applied after performing $n_k - 1$ steps of prediction with the fuzzy model. When $n_k = 1$, the general rule–based model (3) corresponds to the following regression system:

$$y(k+1)=f(\mathbf{x}(k), u(k)) \tag{4}$$

The inputs of the model are the current state $\mathbf{x}(t) = [y(k), \ldots, y(k-n+1), \ldots, u(k-1), \ldots, u(k-n)]^T$ and the current input u(k). The output is a prediction of the system's output at the next sample y(k+1). The objective of the control algorithm is to compute the control input u(k), such that the system output at the next sampling instant is equal to the desired (reference) output r(k+1). In principle, this can be achieved by inverting the model of the process. Given the current state $\mathbf{x}(k)$ and the reference r(t+1), the control input is given by:

$$u(k) = f^{-1}(\mathbf{x}(k), r(k+1)) \tag{5}$$

where the reference r(k+1) is replaced by y(k+1). Generally, it is difficult to find the analytical inverse function $f^{-1}(.)$. Therefore, the method exploited here makes use of the identified fuzzy TS of the process under investigation for providing the particular state $\mathbf{x}(k)$ at each time step k. From this mapping, the inverse mapping (5) is easily *identified* as a model in the form of (3), provided the controlled system is stable.

Therefore, the series connection of the controller and the identified inverse model, should give an identity mapping (perfect control):

$$y(k+1) = f(\mathbf{x}(k), u(k)) = f\big(\mathbf{x}(k), f^{-1}(\mathbf{x}(k), r(k+1))\big) = r(k+1) \tag{6}$$

when u(k) exists such that $r(k+1) = f(\mathbf{x}(k), u(k))$. However, due to modelling errors, noise, and disturbance, by means of the fuzzy identification procedure, the difference $|r(k+1)-f(\mathbf{x}(k),u(k))|$ is made arbitrarily small by an appropriate choice of the identification parameters, *i.e.* the fuzzy membership functions, the number of clusters, and the regressand. The *process fuzzy model* is used for the recursive prediction of the state vector $\mathbf{x}(k)$. Apart from the computation of the membership degrees, both the process model and the controller are estimated using standard matrix operations and linear interpolations, which makes the algorithm suitable for *real–time implementation*.

Note however that, with the fuzzy control strategy proposed here, disturbances acting on the process, measurement noise and model–plant mismatch can cause differences in the behaviour of the process and of the model. A mechanism to compensate this error can be exploited *e.g.* via *on–line adaptation* of the process model. On–line adaptation can be applied to cope with the mismatch between the plant and the fuzzy model. In many cases, a mismatch occurs as a consequence of (temporary) changes of process parameters. Therefore, Section IV motivates the adaptive strategy based on linear models, whose parameters are adapted on–line and exploited for the controller parameter estimation.

IV. Recursive Identification for Adaptive Control

This section describes the recursive approach exploited for obtaining the mathematical description of the wind turbine system, which is used for the design of the second control strategy. A modification of the Frisch scheme algorithm is proposed here to identify dynamical Errors–In–Variables (EIV) models [10, 17]. For the update of the estimated model parameters, a recursive bias–compensating strategy is also implemented. Thus, a recursive Frisch scheme identification approach is extended to enhance its on–line applicability. It is shown that by incorporating adaptation via the introduction of exponential forgetting, the algorithm is able to compensate for the systematic errors, which arise in the original scheme [10]. Therefore, this adaptive recursive Frisch scheme is able to deal with linear time–varying systems, and it is used in connection with the design of an adaptive control scheme, shown in Section IV-A.

Thus, the recalled scheme is used for the on–line identification of the process modelled by the following transfer function G(z):

$$G(z) = B(z^{-1})/A(z^{-1}) = \\ = \left(b_1 z^{-1} + \ldots + b_{nb} z^{-nb}\right)/\left(1 + a_1 z^{-1} + \ldots + a_{na} z^{-na}\right) \quad (7)$$

where a_i, b_i, n_a and n_b represent the unknown parameters and the structure of the model, defining the polynomials $A(z^{-1})$ and $B(z^{-1})$ whilst z is the discrete–time complex variable.

The parameter vector describing the linear relationship is given by:

$$\theta = \left[a_1 \ldots a_{na}\, b_1 \ldots b_{nb}\right]^T \quad (8)$$

and its extended version is denoted with:

$$\vartheta = \left[1\ \theta^T\right]^T \quad (9)$$

Hence, an alternative expression for the considered difference equation is given by:

$$\Psi^T(k)\, \vartheta = 0 \quad (10)$$

Where:

$$\Psi(k) = \left[-y(k), \ldots, -y(k-na+1), \ldots, u(k-1), \ldots, u(k-nb)\right]^T \quad (11)$$

Is the extended regressor vector.

The Frisch scheme provides estimates for the measurement errors affecting the input and output signals u(k) and y(k), *i.e.* σ_u, σ_y, and θ for a linear time–invariant dynamical system. Moreover, it can also be exploited to determine the polynomial orders n_a and n_b, as shown *e.g.* in [7].

However, since this work is oriented to the design of an adaptive controller, the polynomial orders are assumed to be fixed in advance. In this adaptive control application, it is essential to obtain on–line estimates of the model parameters θ(k) in (7), while the process generating the data is running. In fact, this application study is focusing on adaptive control, where the control action at time step k relies on a current estimate of the plant model, which is estimated using data up to the sample k. Therefore, the Frisch scheme relying on batch expressions has to be modified in a recursive algorithm, as described in [10]. The on–line identification method described here was implemented by the author in the Matlab® and Simulink® environments, and integrated in the Simulink® toolbox [11].

Finally, once the time–varying parameters θ(k) of the discrete–time linear model approximating the nonlinear process (2) have been computed at each time step k, the adaptive controller is designed as described in Section IV-A.

A. *Adaptive Controller Design*

With reference to the particular benchmark under diagnosis, adaptive controllers for processes of second order ($n_a = n_b = n = 2$) are exploited. Moreover, the considered adaptive controllers are based on the trapezoidal method of discretization.

Therefore, with reference to (7), with $n_a = n_b = n = 2$, the transfer function parameters estimated on–line are:

$$\theta = [a_1\ a_2\ b_1\ b_2]^T \quad (12)$$

Note that the subscript k for model and controller parameters is dropped in order to simplify equations and formulas.

The control law corresponding to discrete–time PI adaptive controller has the form:

$$u(k) = K_p \left[e(k) + \tfrac{1}{2} T_s/T_I \left(e(k)-e(k-1)\right)\right] + u(k-1) \quad (13)$$

where e(k) is the tracking error, *i.e.* e(k) = r(k) – y(k), with r(k) the set–point or reference signal, T_s the sampling time. The (time–varying) controller variables K_p and T_I are now computed from the time–varying model parameters θ(k) [11].

In particular, the controller parameters K_p and T_I are computed using the Ziegler-Nichols relations depending on the (time–varying) critical gain and the critical period of oscillations [11]. Also these variables are functions of the time–varying model parameters θ(k).

Finally, the next Section V will show the results achieved by using the control design schemes described above.

V. Simulation Results

Regarding the fuzzy modelling method, the GK clustering algorithm with M = 3 clusters and a number of shifts n = 2 was applied to the estimation and validation sampled data sets $\{P_g(k), \omega_g(k), \beta_r(k)\}$, with k = 1, 2, …, N and $N = 440 \times 10^3$. 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 $\{P_g(k), \omega_g(k), \tau_g(k)\}$. After clustering, the TS model parameters for each output $P_g(k)$ and $\omega_g(k)$ were estimated.

Therefore, the two outputs y(t) of the wind turbine continuous–time model (2) are approximated by two TS fuzzy prototypes (3). The relative mean square errors of the output estimations are 0.0254 for the first output, and 0.0125 for the second one. The fuzzy model estimation procedure was implemented in order to guarantee the identification of stable TS prototypes via the parameters n, μ_i, a_i, and b_i. The fitting capabilities of the estimated fuzzy models can be expressed also in terms of the so–called Variance Accounted For (VAF) index [16]. In particular, the VAF value for first output was bigger than 90%, whilst bigger than 99% for the second one. Hence, the fuzzy multiple models seem to approximate the process under investigation quite accurately.

Regarding the fuzzy controllers, the experimental set–up employs 2 (Multiple–Input Single–Output) MISO fuzzy regulators used for the control of the blade pitch angles $\beta_r(t)$ and the generator control torque $\tau_g(t)$, respectively, that were identified according to the fuzzy inverse model scheme suggested in Section III-A. Also in this case, the GK clustering algorithm was applied again for the estimation of the two fuzzy inverse model regulators. A number of M = 3 clusters and a number of shifts n = 3 were applied to the estimation and validation sampled data sets $\{\beta_r(k), P_g(k), \omega_g(k)\}$. On the other hand, a number of clusters M = 3 and n = 3 were considered again for achieving a description of the second fuzzy inverse model regulator via the clustering of the data $\{\tau_g(k), P_g(k), \omega_g(k)\}$.

The controller capabilities were assessed in simulation by considering different data sequences. In Table II the per–cent Normalised Sum of Squared tracking Error (NSSE) values defined as:

$$NSSE^2\% = 100\, \Sigma^N_{k=1}(r(k) - y(k))^2 / \Sigma^N_{k=1} r^2(k) \qquad (14)$$

are computed for the designed fuzzy controllers. Table II refers to the full–load operation, where the performance depends on the generator speed, ω_g, with respect to the nominal one, ω_{nom}.

With reference to the second adaptive control design, the two outputs $P_g(t)$ and $w_g(t)$ of the wind turbine continuous–time nonlinear model (2) were approximated by 2 time–varying MISO discrete–time second order prototypes of the type (7) with 2 inputs. The approach described in Section IV for SISO models can be easily extended to the MISO case. Using these two on–line identified prototypes, the model–based approach for determining the adaptive controllers shown in Section IV-A was exploited. Thus, the parameters of the adaptive controllers were computed on–line. In particular, the adaptive regulator parameters in (13) were computed analytically at each time step k. Simulations were performed in the same conditions of the fuzzy controllers, and 2 adaptive regulators were used. As an example, the initial values for the parameters of the on–line estimation algorithm are listed in Table I.

In order to analyse the performance of the proposed adaptive strategy, Table II reports the NSSE values computed also for these controllers. In full load operation the performance depends on the generator speed ω_g with respect to the nominal one, ω_{nom} = 162 rad/s.

According to these simulation results, good tracking capabilities of the suggested controllers seem to be reached, and the adaptive solution seems better than the fuzzy one.

TABLE I. INITIALISATION PARAMETERS OF THE ADAPTIVE ALGORITHM.

Parameter	Value
$\theta(0)$	$[0.1, 0.20, 0.30, 0.4]^T$

TABLE II. FUZZY AND ADAPTIVE CONTROLLER NSSE% VALUES.

Data Set	Fuzzy	Adaptive
Data set #1	16.57%	12.95%
Data set #2	17.85%	13.67%

A. Robustness Evaluation

In this section, further experimental results were reported. They regard the performance evaluation of the developed control schemes with respect to modelling errors and measurement uncertainty. In particular, the simulation of different data sequences was performed by exploiting the wind turbine benchmark simulator, and a Matlab® Monte–Carlo analysis. In fact, the Monte–Carlo tool is useful at this stage as the control strategy performances depend on the error magnitude due to the model approximation and uncertainty, as well as on input–output measurement errors.

In particular, the nonlinear wind turbine simulator originally developed in the Simulink® environment [14] was modified by one of the authors in order to vary the statistical properties of the signals used for modelling possible process parameter uncertainty, and measurement errors. Under this assumption, Table III reports the nominal values of the considered wind turbine model parameters with respect to their simulated but realistic uncertainty.

The Monte–Carlo analysis describes these variables as Gaussian stochastic processes, with zero–mean and standard deviations corresponding to the maximal error values in Table III.

Therefore, for performance evaluation of the control schemes, the best, average, and worst values of the NSSE% index were computed, and experimentally evaluated with 500 Monte–Carlo runs, as shown in Table IV.

TABLE III. REALISTIC WIND TURBINE UNCERTAINTY.

System	Error	Physical Effect
$\beta_r(t)$	11%	Pitch position measurement accuracy
$\omega_g(t)$	18%	Generator speed measurement accuracy
$\tau_g(t)$	21%	Generator torque measurement accuracy
$P_g(t)$	20%	Electrical power measurement accuracy
Pitch system	49%	Hydraulic system pressure change
Drive–train	5%	Drive train dynamics change
Converter dynamics	50%	Offset in Converter torque control

TABLE IV. NSSE% VALUES FOR THE MONTE-CARLO ANALYSIS.

Test case	Fuzzy	Adaptive
Best case	15.57%	11.05%
Average case	17.94%	13.72%
Worst case	19.94%	15.04%

In particular, Table IV summarises the values of the considered performance index NSSE% according to the best, worst and average cases, with reference to the possible combinations of the parameters described in Table III. Table IV shows that the proposed control schemes, and in particular the adaptive solution, allow for good control performances even in the presence of considerable error and uncertainty effects.

B. Comparative Simulations

This section compares the control methods suggested in this paper with respect to alternative control approaches.

The first control scheme proposed for comparison purposes relies on UIOs [6], which were used for estimating measurements used by the control system. The UIO–based scheme is chosen since it enables the possibility to include robustness towards the uncertainty of the wind speed, which is difficult to measure.

On the other hand, the second control approach uses the idea of virtual sensors/actuators (VAS) [5]. An estimation of the uncertainty acting on the process is provided on the basis of a batch least squares approach. The use of on–line disturbance estimation is essential for all compensation approaches.

The third approach relies on an LMI–based method for designing and synthesising the LPV controller, which is based on the LPV controller design method presented in [13]. In fact, the considered wind turbine model has varying parameters caused by nonlinearities in the aerodynamic model along the nominal operating trajectory and due to the model uncertainty.

In order to provide a brief but clear insight into the above-mentioned techniques, the comparison was performed in the same previous working conditions, and based on the NSSE% index suggested at the beginning of Section V. Table V summarises the results obtained by comparing the three control techniques recalled above with the ones proposed in this study.

Table V shows that the scheme using the VAS and the LPV strategies allow to achieve better performances in terms of tracking error. However, the LPV controller can increase the computational time considerably with respect to the other solution, without any gain scheduling, whilst the LPV and UIO control methods can require larger computational effort at the design stage.

Few further comments can be drawn here. When the modelling of the dynamic system can be perfectly obtained, in general model–based control strategies are preferred. On the other hand, when modelling errors and uncertainty are present, alternative control schemes relying on adaptation mechanisms, or passive robust control methods, showed interesting robustness properties. The fuzzy logic–based scheme relies on the learning accumulated from off–line simulations, but the on–line estimation stage could be computationally heavy. Finally, regarding the proposed methods using LPV or fuzzy tools, they seem rather simple and straightforward, even if optimisation stages can be required.

TABLE V. MONTE-CARLO ANALYSIS AND NSSE VALUES WITH THE COMPARED CONTROLLERS.

Test case	UIO	VAS	LPV
Best case	26.55%	19.85%	10.95%
Average case	27.72%	17.94%	13.02%
Worst case	28.44%	16.33%	14.94%

C. Experimental Results

Also the stability properties of the overall control strategies were checked by means of a Monte–Carlo campaign based on the wind turbine benchmark. In fact, as pointed out above, the Monte–Carlo analysis represents the only method for estimating the efficacy of the developed control schemes when applied to the monitored process.

It is worth noting that the work [18] provided an analytical demonstration of the stability of an adaptive control scheme for wind turbines. However, model parameter variations, recursive Frisch scheme adaptive methods, or complete wind models were not taken into account there.

All simulations were performed by considering noise signals modelled as Gaussian processes, according to the standard deviations reported in Table III. The wind turbine benchmark simulator generated different wind sequences. Moreover, the initial conditions of the dynamic models and recalled in Section II (*i.e.* the drive–train, the generator/converter, and the pitch system) were changed randomly. Therefore, the random wind signal v(t), the parameters of Table III, and the dynamic model initial conditions allowed to obtain different sequences of the wind turbine signals $\beta_r(t)$, $\tau_g(t)$, $\lambda(t)$, $\omega_g(t)$, and $P_g(t)$ for each Monte–Carlo simulation.

As an example of a single Monte–Carlo run, Fig. 1 highlights that the main wind turbine model variables, such as the generator torque $\tau_g(t)$, the tip-speed–ratio $\lambda(t)$, and the generator power $P_g(t)$ remain bounded around the reference values, proving the overall system stability in simulation, even in the presence of disturbance and uncertainty. These results refer to the case of partial load operation with the fuzzy controllers.

Fig. 1 shows also that in the first part of the simulation the output power P_g becomes larger than the theoretical one, $P_{g,max}$, as the kinetic energy from the rotor shaft is converted into electrical energy produced by the generator. On the other hand, $P_{g,max}$ can be above the generated power, since the inertia of the rotor is accelerated before $P_{g,max}$ can be matched.

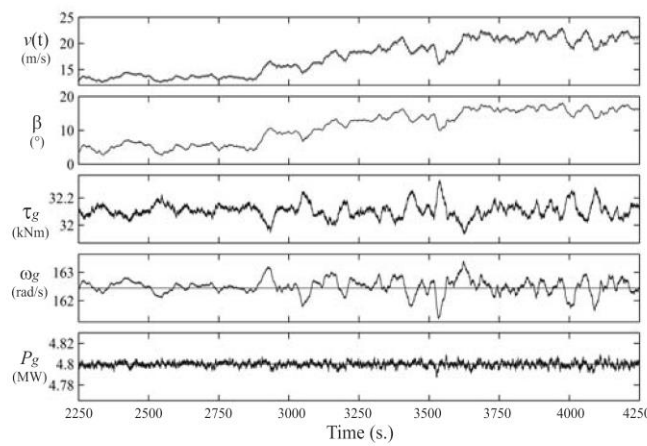

Fig. 1. Simulations of the wind turbine benchmark with the fuzzy controllers.

TABLE VI. HIL RESULTS WITH RESPECT TO THE NSSE% INDEX.

Controller	Partial load	Full load
Fuzzy	38.72%	18.94%
Adaptive	28.81%	14.83%

In order to evaluate the potential of utilising the proposed control algorithms also in real applications and investigate their capability to on–board implementation, the remainder of this section presents the results of the Hardware In the Loop (HIL) tests. These experimental results serve to validate definitely the designed control algorithms considering almost real conditions that the wind turbine may experiment with during its working situations. For this purpose, HIL test–bed already described in [9] was exploited, in order to provide the capabilities to validate the developed control algorithms in an almost real–time condition. The results achieved from one test are summarised in Table VI for the proposed identified fuzzy and adaptive controller solutions.

Table VI illustrates that there are some deviations between the achieved results, but consistent with the ones from the Monte–Carlo analysis. Although there are some deviations between the simulation and the experimental results, these deviations are not critical and the results obtained are accurate enough for future wind turbine real applications.

VI. FUTURE WORKS

The increasing dimensions of wind turbines lead to the increase in the loads on wind turbine structures. Because of increasing rotor size and spatially varying loads along the blade, individual blade pitch control can reduce the negative effects of sub-rotor-sized turbulent structures. Additional pitch control loops can be used to damp the tower motion or additional structural vibrations in the full load working condition [13].

Given the complexity of the wind turbine system, the stability of the complete plant plus control system cannot be proven. The multiple control loops interact, as do the multiple degrees of freedom of the turbine, especially as wind turbines become larger and have lower natural frequencies. A unified Multiple-Input Multiple-Output (MIMO) framework for individual blade pitch control can achieve significant load reduction for floating offshore wind turbines with strong coupling across degrees of freedom [13].

Because wind turbine control is often achieved using two distinct control loops for the working regions between partial load and full load conditions, the transition between these regions can be problematic. For some turbines, the maximum structural damage occurs due to extreme and fatigue loads during this transition. Often, the act of switching between these region controllers contributes to the problem.

Advanced control strategies, for example, uses an additional control region between partial and full load conditions to facilitate switching between these two conditions. The primary objective of this control strategy, described in [13], [1] is to connect these switching controllers linearly in the generator torque versus generator speed plane. Unfortunately, this linear connection does not result in smooth transitions, and the discontinuous slopes in the torque control curve can contribute to excessive loading on the turbine.

Wind turbines can also be damaged when they are stopped as a result of supervisory control action due to high winds or fault conditions [13], [1], [2], [3]. However, little or no active control is performed when the turbine is stopped, although the yaw angle can be changed to accommodate changes in wind direction, which can prevent some damage.

In addition to the possibility of improving control when the turbine is stopped, advanced fault detection and turbine protection schemes are of interest to the wind industry [2], [3]. Stopping the turbine in the case of emergency, which might entail pitching the blades to a predetermined stop position at maximum pitch rate and setting the mechanical brakes with which the rotor is equipped, can also cause damage to the machine and must be done only when a turbine failure is suspected.

Finally, controller performance depends on modelling accuracy. For instance, as shown in Section V, realistic modelling error in the optimal tip-speed ratio can cause an energy loss of around a few per cent in the partial load condition [13], which can be a significant loss in this industry. Even disregarding model errors, the dynamical behaviour of a wind turbine changes over time due to wear, debris build-up on the blades, and environmental conditions. As such, adaptive methods shown in this work can be used to tune controllers to improve performance compared to time-invariant methods [8], [9].

While wind turbine dynamics can be modelled using first principles, this study shown that efficient methods for obtaining models from measurements also exist, including the development of closed-loop identification methods for deter mining linear parameter-varying models [7]. These models can be used for robust control. Modelling of wind turbines and wind farms is further discussed in [1], [13].

VII. CONCLUSION

The paper is focused on two examples of control designs for a nonlinear wind turbine prototype. The proposed control designs represent viable and easy-to-use methods for the straightforward derivation of proper controller models, as data-driven and system identification from data approaches are exploited. Tests on the considered benchmark process and Monte–Carlo analysis were the tools for assessing experimentally the properties of the proposed control schemes, in the presence of modelling and measurement errors. The developed control methods were also compared with different approaches, in order to evaluate the considered techniques. These comparisons highlight that the proposed design methodologies can constitute reliable and robust approaches for application to real wind turbine processes.

REFERENCES

[1] P. F. Odgaard and K. Johnson, “Wind Turbine Fault Diagnosis and Fault Tolerant Control – an Enhanced Benchmark Challenge,” in Proc. of the 2013 American Control Conference – ACC, (Washington DC, USA), pp. 1–6, IEEE Control Systems Society & American Automatic Control Council, June 17 – 19 2013.

[2] P. F. Odgaard, J. Stoustrup, and M. Kinnaert, "Fault–Tolerant Control of Wind Turbines: A Benchmark Model," IEEE Transactions on Control Systems Technology, vol. PP, no. 2013, pp. 1–15, 2013. Available Online. DOI: 10.1109/TCST.2013.2259235.

[3] M. Sami and R. J. Patton, "An FTC Approach to Wind Turbine Power Maximisation Via T–S Fuzzy Modelling and Control," in Proceedings of the 8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes – SAFEPROCESS 2012 (C. Verde, C. M. Astorga Zaragoza, and A. Molina, eds.), vol. 8, (National Autonomous University of Mexico, Mexico City, Mexico), pp. 349–354, August 2012. DOI: 10.3182/20120829–3–MX–2028.00131.

[4] P. F. Odgaard and J. Stoustrup, "Fault Tolerant Wind Speed Estimator used in Wind Turbine Controllers," in Proceedings of the 8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes – SAFEPROCESS 2012 (C. Verde, C. M. Astorga Zaragoza, and A. Molina, eds.), vol. 8, (National Autonomous University of Mexico, Mexico City, Mexico), pp. 331–337, August 2012. DOI: 10.3182/20120829-3-MX-2028.00009.

[5] D. Rotondo, F. Nejjari, V. Puig, and J. Blesa, "Fault Tolerant Control of the Wind Turbine Benchmark using Virtual Sensors/Actuators," in Proceedings of the 8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes – SAFEPROCESS 2012 (C. Verde, C. M. Astorga Zaragoza, and A. Molina, eds.), vol. 8, (National Autonomous University of Mexico, Mexico City, Mexico), pp. 114–119, August 2012. DOI: 10.3182/20120829-3-MX-2028.00185.

[6] P. F. Odgaard and J. Stoustrup, "Fault Tolerant Control of Wind Turbines using Unknown Input Observers," in Proceedings of the 8th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes – SAFEPROCESS 2012 (C. Verde, C. M. Astorga Zaragoza, and A. Molina, eds.), vol. 8, (National Autonomous University of Mexico, Mexico City, Mexico), pp. 313–319, August 2012. DOI: 10.3182/20120829-3-MX-2028.00010.

[7] S. Simani, C. Fantuzzi, R. Rovatti, and S. Beghelli, "Parameter identification for piecewise linear fuzzy models in noisy environment," International Journal of Approximate Reasoning, vol. 1, pp. 149–167, September 1999. Publisher: Elsevier.

[8] S. Simani, "Data–Driven Design of a PI Fuzzy Controller for a Wind Turbine Simulated Model," in Proceedings of the IFAC Conference on Advances in PID Control – PID'12 (R. Vilanova and A. Visioli, eds.), vol. 2, (University of Brescia, Brescia, Italy), pp. 667–672, DII, Faculty of Engineering, University of Brescia, Italy, IFAC, March 28–30 2012. Invited paper. ISBN: 978-3-902823-18-2. DOI: 10–3182/20120328–3–IT–3014.00113.

[9] S. Simani, "Application of a Data–Driven Fuzzy Control Design to a Wind Turbine Benchmark Model," Advances in Fuzzy Systems, vol. 2012, pp. 1–12, November 2nd 2012. Invited paper for the special issue: Fuzzy Logic Applications in Control Theory and Systems Biology (FLACE) . ISSN: 1687–7101, e–ISSN: 1687-711X. DOI: 10.1155/2012/504368.

[10] J. G. Jens G. Linden, T. Larkowski, and K. J. Burnham, "Algorithms for recursive/semi–recursive bias-compensating least squares system identification within the errors–in–variables framework," International Journal of Control, vol. 85, pp. 1625–1643, June 2012. DOI:10.1080/00207179.2012.696145.

[11] V. Bobál, J. Böhm, J. Fessl, and J. Machácek, Digital Self–Tuning Controllers: Algorithms, Implementation and Applications. Advanced Textbooks in Control and Signal Processing, Springer, 1st ed., 2005.

[12] P. F. Odgaard, J. Stoustrup, and M. Kinnaert, "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, vol. 1, (Barcelona, Spain), pp. 155–160, June 30 – July 3 2009. DOI: 10.3182/20090630–4–ES–2003.0090.

[13] F. D. Bianchi, H. D. Battista, and R. J. Mantz, Wind Turbine Control Systems: Principles, Modelling and Gain Scheduling Design. Advances in Industrial Control, Springer, 1st ed., July 2007. ISBN: 1–84628–492–9.

[14] P. F. Odgaard and J. Stoustrup, "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, vol. 1, (Barcelona, Spain), pp. 161–166, IFAC – Elsevier, June 30 – July 3 2009. DOI: 10.3182/20090630–4–ES–2003.0048.

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

[16] R. Babuška, Fuzzy Modeling for Control. Kluwer Academic Publishers, 1998.

[17] S. Van Huffel and P. Lemmerling, eds., Total Least Squares and Errors–in–Variables Modeling: Analysis, Algorithms and Applications. Springer–Verlag, 1st ed., February 2002. ISBN: 1402004761.

[18] K. E. Johnson, L. Y. Pao, M. J. Balas, V. Kulkarni, and L. J. Fingersh, "Stability analysis of an adaptive torque controller for variable speed wind turbines," in Proceedings of the 43rd IEEE Conference on Decision and Control – CDC'04, vol. 4, (Paradise Island, Bahamas), pp. 4087–4094, December 14–17 2004.