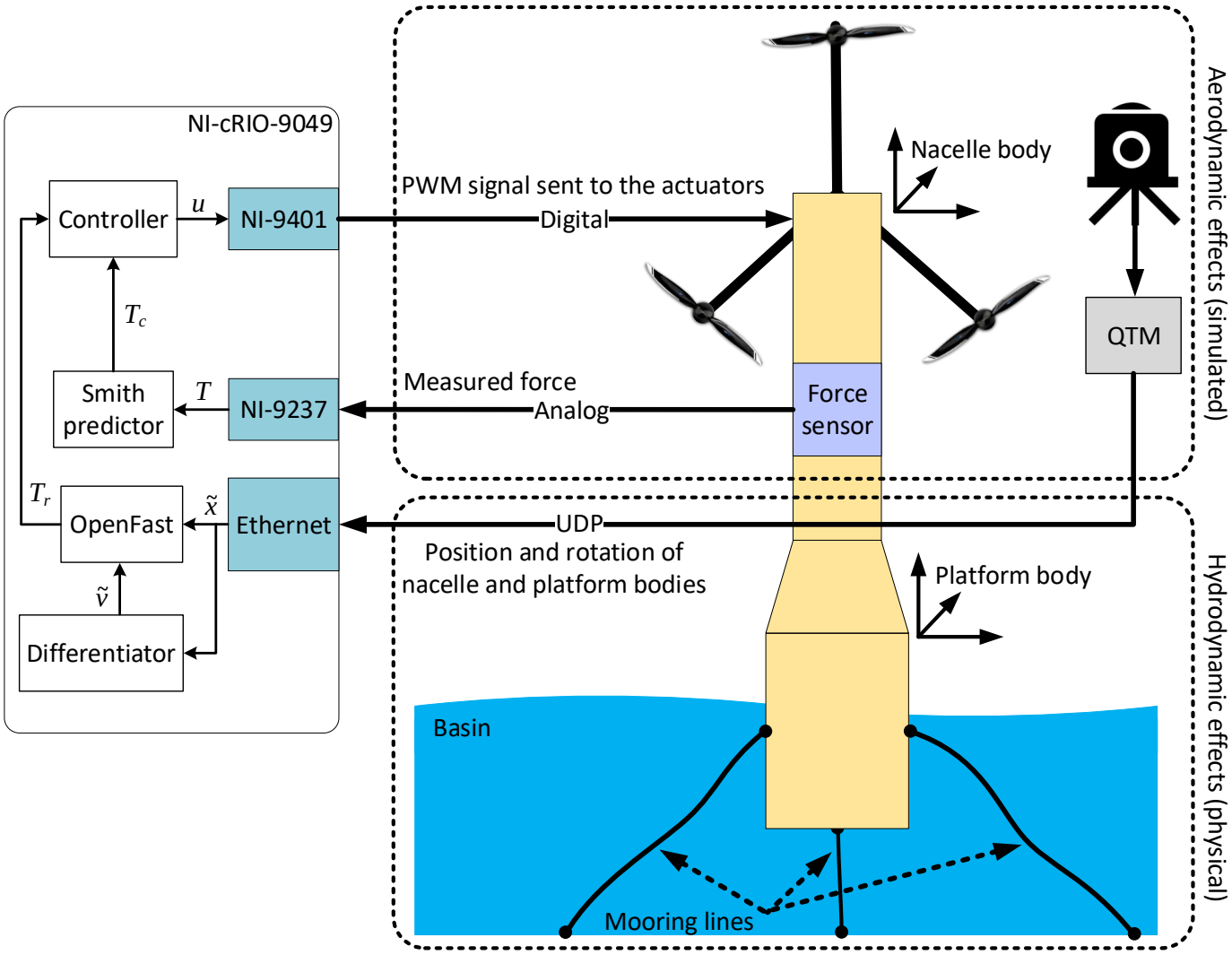
Control design for thrust generators with application to wind turbine wave-tank testing: a sliding-mode control approach with Euler backward time-discretization

Mohammad Rasool Mojallizadeh, Félicien Bonnefoy, Vincent Leroy, Franck Plestan, Sylvain Delacroix, Jérémy Ohana, Benjamin Bouscasse

Graphical Abstract

Control design for thrust generators with application to wind turbine wave-tank testing: a sliding-mode control approach with Euler backward time-discretization

Mohammad Rasool Mojallizadeh, Félicien Bonnefoy, Vincent Leroy, Franck Plestan, Sylvain Delacroix, Jérémy Ohana, Benjamin Bouscasse



Highlights

Control design for thrust generators with application to wind turbine wave-tank testing: a sliding-mode control approach with Euler backward time-discretization

Mohammad Rasool Mojallizadeh, Félicien Bonnefoy, Vincent Leroy, Franck Plestan, Sylvain Delacroix, Jérémy Ohana, Benjamin Bouscasse

- A sliding-mode controller has been developed for a thrust generator.
- Backward time-discretization of the controller as well as its properties are addressed.
- The method has been compared with feedforward, gain scheduling PID, and Fuzzy controllers.
- The effect of the time-differentiation on the closed-loop control system is studied.
- The comparative experimental results are provided for 5 controllers and 4 differentiators.

Control design for thrust generators with application to wind turbine wave-tank testing: a sliding-mode control approach with Euler backward time-discretization

Mohammad Rasool Mojallizadeh[a,b,*], Félicien Bonnefoy[b], Vincent Leroy[b], Franck Plestan[a], Sylvain Delacroix[b], Jérémy Ohana[b], Benjamin Bouscasse[b]

[a] *Nantes Université, École Centrale Nantes, CNRS, LS2N, UMR 6004, F-44000 Nantes, France*
[b] *Nantes Université, École Centrale Nantes, CNRS, LHEEA, UMR 6598, F-44000 Nantes, France*

Abstract

The control design for a propeller-based thrust generator used in a wind turbine testing platform is studied in this work. A mathematical model has been developed for the system including the motors and propeller. Subsequently, a continuous-time sliding-mode controller is designed based on the developed model and its stability and robustness have been addressed. An Euler backward time discretization method has been developed for the continuous-time sliding-mode controller to achieve a chattering-free implementation. The properties of the sliding-mode controller under the developed time discretization method *e.g.*, finite-time convergence, and gain insensitivity have been studied analytically. In order to evaluate the developed sliding-mode control law under the discretization method, three known control strategies, *i.e.*, gain-scheduling proportional-integral control, fuzzy control, and feedforward compensator strategies have been designed for the system. Some remarks are also given for the differentiator selection used to estimate the velocity. The experiments under different scenarios are then conducted and the results corresponding to all four controllers are provided along with a comparative analysis to identify the properties of each control configuration.

Keywords: propeller-based actuator, sliding-mode control, thrust control, Euler backward discretization, floating wind turbine, fuzzy control, feedforward compensator, differentiator

1. Introduction

Floating wind turbines (FWT) have recently attracted attention as an alternative renewable source of energy. FWTs are installed offshore, far away from the coast, and compared to the on-shore wind turbines, they can potentially generate more electricity because the wind energy is more abundant on the sea surface (see [1] for instance). A real-scale FWT requires a significant capital budget and is usually difficult to access for experimental tests. As a result, pure numerical simulators may be used for the early design stages (a review of the software packages used for the pure numerical solutions is available in [2]). However, such software packages provide limited accuracy and cannot model all the systems' dynamics, *e.g.*, strongly nonlinear hydrodynamic loads. Hence, a small-scale implementation of the FWT seems to be mandatory before building and implementing a real-scale one to avoid further modifications and extra costs. As has been reported [2–10], down-scaling all the elements of a FWT does not lead to realistic aerodynamic effects because of scaling of the

*Corresponding author
Email addresses: rasool.mojallizadeh@ec-nantes.fr (Mohammad Rasool Mojallizadeh), felicien.bonnefoy@ec-nantes.fr (Félicien Bonnefoy), vincent.leroy@ec-nantes.fr (Vincent Leroy), franck.plestan@ec-nantes.fr (Franck Plestan), sylvain.delacroix@ec-nantes.fr (Sylvain Delacroix), jeremy.ohana@ec-nantes.fr (Jérémy Ohana), benjamin.bouscasse@ec-nantes.fr (Benjamin Bouscasse)

Reynolds number [11, 12], which dominates the aerodynamic loads. To solve this issue, the turbine's rotor of the laboratory-scale FWT is usually replaced by a drag disk to achieve a realistic aerodynamic effect [12]. This approach has its own limitations and cannot be used to model variable wind velocity profiles and wind turbulence. Moreover, the turbine's controller, used to control the pitch of the blades and mechanical torque of the turbine, cannot be studied in this approach [2]. Alternatively, to solve the problems caused by the down-scaling, the rotor can be totally replaced with an actuator to emulate the aerodynamics numerically using the so-called *software-in-the-loop* (SIL) strategy while keeping the real physical hydrodynamics. In this case, the force acting on the turbine is calculated online using a simulation tool, *e.g.*, `OpenFAST`, and sent to the actuator as a control reference signal to be generated. Following the literature, to generate the required force, two different types of actuators, *i.e.*, propellers [2–10], and actuated cables [13–23] are used. Throughout this paper, the propellers are the only actuators used to generate the aerodynamic force.

The accuracy of the aerodynamic thrust generated by the actuator when tracking the reference force plays a key role in the accuracy of the SIL approach. This topic was addressed in several references [2–10] using the feedforward compensator (FFC). The common strategy in all these references is to measure the static gains of the actuators for a range of inputs in the open-loop configuration to build a lookup table. Afterward, the inverses of these gains are used as a FF controller. The drawbacks of such an open-loop method are known to the control community, *e.g.*, sensitivity to the perturbations, and undesired transient and steady-state responses. Since the closed-loop controller design has not been yet addressed for this specific application, the results corresponding to controller design for other applications with propeller-based thrust generators, including aircraft and multi-rotors have been reviewed in the sequel.

Apart from the open-loop FFC known as the calibration used in [2–10], the majority of the references, considering the propeller-based thrust generators, are based on a linear control strategy, *e.g.*, proportional–integral–derivative (PID) control [24–27] and its gain-scheduling (GSPID) version [28, 29], linear quadratic regulator [24, 30], and H-infinity control [31, 32] to control the thrust generated by the propeller. The PID controller is usually employed without addressing the stability, and its parameters are selected based on trial and error [24, 25]. Because of the nonlinear behavior of propeller-based systems, the classic PID control may not be able to provide the required performance in all operating conditions. Hence, gain-scheduling PI (GSPI) [28, 29] has been used for such systems. The linear quadratic regulators are another linear controllers that can optimize the behavior of the system in terms of time or energy consumption [24, 30]. Moreover, since propeller-based applications, like quadrotors, are usually exposed to the disturbances, *e.g.*, external wind, robust linear controllers including the H_∞ control has been developed for them [31].

While the design of the linear controllers is straightforward and can be made based on the linearized model, they share the same drawbacks, *i.e.*, their performance or even stability may not be guaranteed for all the operating conditions. As a result, nonlinear control strategies are developed including Lyapunov-based designs [33], sliding-mode control (SMC) [26, 34], backstepping control [35, 36], feedback linearization [37], and model predictive control (MPC) [38, 39]. The Lyapunov theory allows studying the stability of the closed-loop system with nonlinear equations. In addition, adaptation laws can be derived based on the Lyapunov method to improve performance. The SMC is another nonlinear control strategy that achieves robustness by employing discontinuous or (set-valued) terms in the control law. SMC can also be used along with other control methods, *e.g.*, feedback linearization, to improve

their robustness [40]. In aerospace applications, the MPC is widely used because it can achieve other objectives including collision avoidance and path planning [41], as well.

While the aforementioned nonlinear control laws can potentially achieve global stability as well as other properties like finite-time convergence, they require nonlinear mathematical modeling which might be cumbersome to calculate. Alternatively, the so-called model-free approaches, *e.g.*, fuzzy logic, or neural network-based controllers are developed for such systems [42]. The model-free control strategies may control the system without requiring a mathematical model, with the price of lacking stability proof.

Apart from the controller design, the software, *e.g.*, `OpenFAST`, used to calculate the aerodynamic reference force, requires the velocity of the FWT body which is not available. To this end, some known differentiators are employed, in this work, to estimate the velocity by differentiating the measured position. In addition, the performance of the control system integrated with the differentiators has been investigated experimentally. A review of the differentiators is available in [43]. The contributions and developments made in this work are as follows:

- A $1/40^{\text{th}}$ scale of the 10MW FWT, originally designed by the Technical University of Denmark [44] has been built in this work based on the SIL architecture.
- The mathematical modeling of the aerodynamic part is studied and a model mapping the duty cycle of the pulse width modulated (PWM) signal applied to the electric motor (as the control input) and the generated thrust (as the control output) is achieved. Such modeling is based on the equations of the electric motor presented in [45], and the propeller's aerodynamics characteristics [46].
- A classic continuous-time SMC has been designed for the derived mathematical model and its asymptotic stability has been addressed in Theorem 1. Note that the Lyapunov function and the procedure used to show the stability are derived from [47, 48].
- An Euler backward discretization has been developed for the continuous-time SMC in Sec. 4.1.3. While the backward discretization of the continuous-time SMC has been addressed before (see [49]), the graphical interpretation, Fig. 4, used to solve the generalized equation is different.
- The characteristic of the developed discrete-time SMC is studied, and the required time for the convergence (Theorem 2), chattering suppression (Corollary 1), calculation burden (Remarks 1 and 6), and gain-insensitivity (Corollary 2) for the unperturbed system are addressed. Moreover, Some remarks (Remarks 2, 3 and 5) are presented to explain the robustness of the discrete-time SMC under a perturbed condition. In addition, gain tuning of the discrete-time SMC, GSPI, and fuzzy controllers are addressed in Remarks 4, 8 and 9, respectively.
- From the class of the reviewed controllers, three control strategies, *i.e.*, linear (GSPI [50]), model-free (fuzzy [51]), and open-loop (FFC [52]) have been adopted from the mentioned references and specifically designed for the system to be compared with the developed discrete-time SMC. Moreover, the effect of the time delay in the control loop is addressed and compensated through the Smith predictor design [53, 54].
- Since the velocity of the FWT body is required for the control synthesis, four differentiators, *i.e.*, Euler [43], linear filter [43], forward super-twisting differentiator [55], and backward super-twisting differentiator [43], have been borrowed, and their parameters are uniformly tuned based on the differentiation

toolbox [43, 56][1] for this application.

- The controllers as well as the differentiators are implemented on the built FWT, and the results corresponding to the integration of the controllers with the differentiators are given and an exhaustive comparative analysis is made to identify the properties of integration of each set of controllers and differentiators.
- The experiments study the effect of the discretization on the excitation of unmodeled dynamics, *i.e.*, bending modes, that exist in the structure of the system.

The remainder of this manuscript is as follows. The structure of the developed FWT is explained in Sec. 2, and a control-oriented mathematical model has been presented in Sec. 3. The control design based on the developed model is the topic of Sec. 4. The necessity of signal differentiation in the SIL strategy and a review of the differentiation methods are presented in Sec. 5. Subsequently, the experimental results corresponding to each controller are presented in Sec. 6 accompanied by the conclusions and discussions in Sec. 7.

2. Floating wind turbine model implemented in the test tank

FWTs are constructed based on different architectures [58]. The one that is considered in this study is based on the spar architecture shown in Fig. 1. This architecture is composed of a hull (a cylinder) immersed in the water and attached to the anchor points via catenary mooring. The blades are installed on the nacelle located on the top of the support tower. The built FWT, in this work, is a 1/40 scale of a 10MW FWT designed by the Technical University of Denmark [44] Fig. 2 with the block diagram in Fig. 3 (see [52] for more details). FWTs are

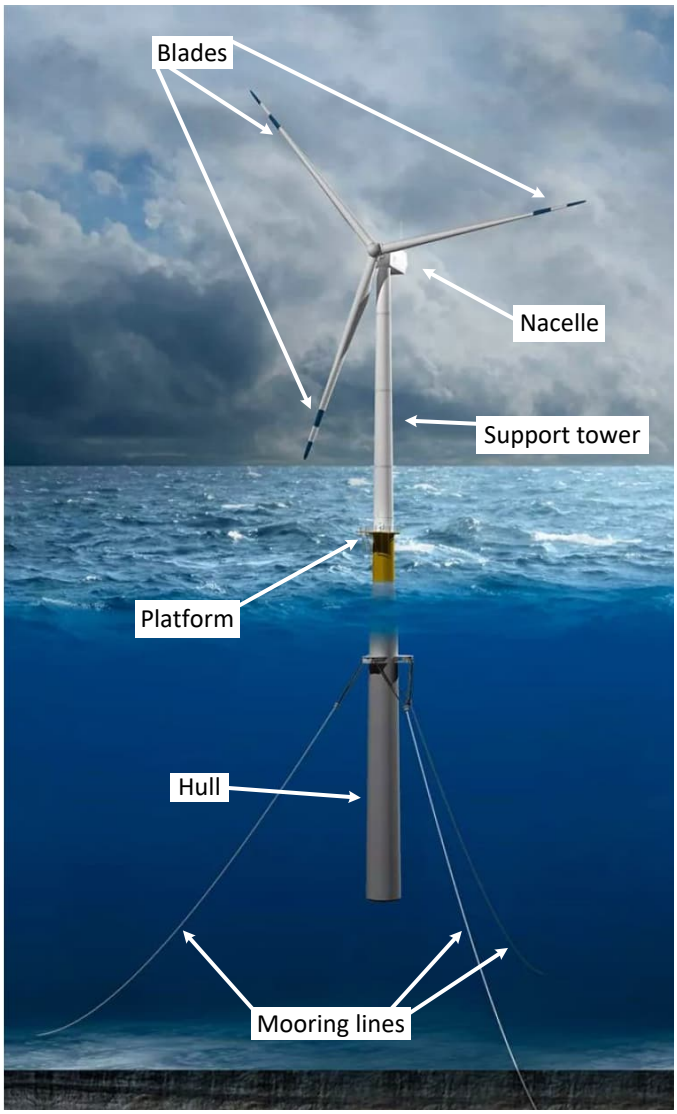


Figure 1: Real-scale FWT based on the spar architecture [57]

subjected to the hydrodynamic and aerodynamic forces, simultaneously, caused by the wave and the wind. As it has been reported by [2–10], while the hydrodynamic parts of a reduced-order FWT can be realized physically for a down-scaled FWT, it is not possible to achieve accurate physical aerodynamic loads. Alternatively, the SIL approach is used in this work to calculate the aerodynamic forces using computer software that has to be generated by the actuators. In other words, the turbine has been

[1] `http://dt.mojalli.com`.

replaced by three propellers[2]. As can be seen in Fig. 3, a motion-tracking `Qualisys track manager (QTM)` system is installed to measure the movements of the nacelle and platform bodies. The measured position vector ($\tilde{x}$) as well as the estimation of its time differentiation ($\tilde{v}$) are sent to the `OpenFAST` software package[3] through an Ethernet cable to calculate the aerodynamic reference thrust $T_r(t)$ that has to be generated by three propellers installed on the nacelle body [52]. To this end, a controller is required to generate the PWM signal $u(t)$ sent to the electric motors attached to the propellers' shafts such that the generated thrust $T(t)$ by the propellers tracks its reference value $T_r(t)$. The control design is the subject of Sec. 4. Before that, a mathematical model will be derived in Sec. 3 to be used for the controller design.

3. Dynamic modeling of the system

The developed model includes the dynamics of the electric motors as well as the propeller. Following [45], the dynamic equation corresponding to the electric motor connected to the propeller can be formulated as follows:

$$J\dot{\omega}(t) + \left(k_f + \frac{k_e k_t}{R_m}\right)\omega(t) + k_l\omega^2(t) = \frac{k_t}{R_m}u(t)V_b, \quad (1)$$

where $J(\mathrm{k/m^2})$ is the total inertia of the motor's shaft and propeller, ω (rad/s) is the angular velocity of the motor's shaft, $k_f(\mathrm{Nm/(rad/s)})$ is the friction constant, $k_e(\mathrm{V/rad/s})$ is the back emf constant, $k_t(\mathrm{Nm/A})$ is the motor's speed constant, and R_m (Ω) is the winding's resistance. The term $k_l\omega^2(t)$ corresponds to the torque induced by the propeller on the shaft, where $k_l(\mathrm{NM})$ depends on the propeller's structure. Moreover, $V_b(\mathrm{V})$ is the voltage of the battery installed in the system, and $u(t)(\%)$ is the duty cycle of the PWM signal applied to the motor drive. The aerodynamic thrust generated by the propeller is $T(t) = K_a\omega^2(t)$, where K_a is the thrust coefficient corresponding to the propeller [46]. Substituting it in (1) gives (the time input argument is removed in some equations for the sake of space):

$$\begin{aligned} &\tfrac{J}{2}\sqrt{\tfrac{1}{k_a T}}\dot{T} + \left(k_f + \tfrac{k_t k_e}{R_m}\right)\sqrt{\tfrac{T}{k_a}} + k_l\tfrac{T}{k_a} = \tfrac{k_t}{R_m}uV_b \Rightarrow \\ &\dot{T}(t) = f(t) + g(t)u(t), \end{aligned} \quad (2)$$

where

$$\begin{cases} f(t) = -\left(k_f + \frac{k_t k_e}{R_m}\right)\frac{2}{J}T - \frac{2}{J}\frac{k_l}{\sqrt{k_a}}T^{3/2}(t) \\ g(t) = \frac{2}{J}\sqrt{k_a T(t)}\frac{k_t}{R_m}V_b. \end{cases} \quad (3)$$

Note that $f(t) \in \mathbb{R} < 0$ and $g(t) \in \mathbb{R} > 0$ hold for all $T > 0$. Moreover, these parameters depend on the system's uncertain parameters which are only partially

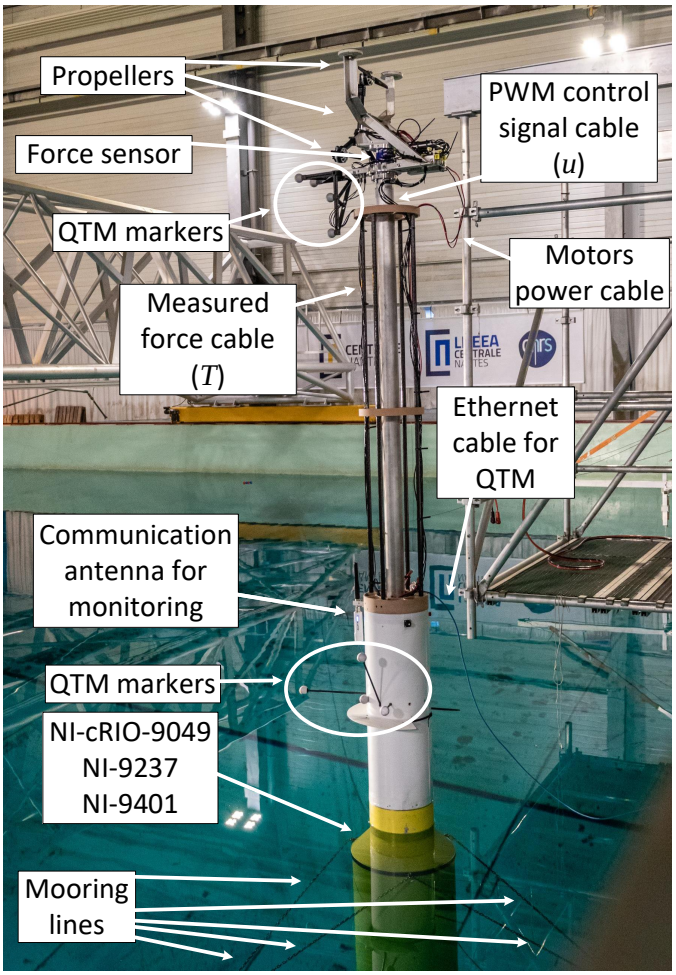


Figure 2: Photograph of the developed reduced-order FWT located in the test tank

[2]It should be emphasized that in the reduced-scale FWT, the turbine (used to convert the wind energy into electricity) does not exist physically. Instead, its effect has been simulated using the SIL strategy. Note that the propellers are used to generate the corresponding aerodynamic effects, and clearly not to generate electricity.

[3]See `https://openfast.readthedocs.io/`

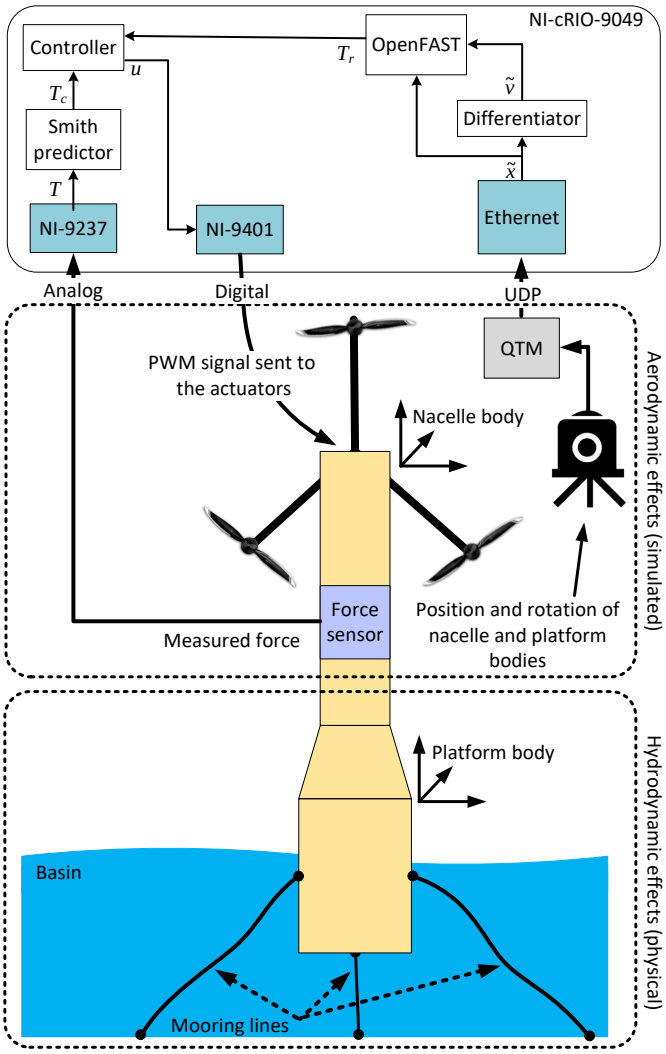


Figure 3: Block diagram of the hardware used for the implementations.

known. In other words, in a real application, only the nominal values of f and g are available. Such uncertainties may be modeled as bounded matched disturbances as made in Theorem 1 (see (15)). Equation (3) provides a map between the duty cycle $u(t)$ and the control output $T(t)$, which are the input control signal and output control variable, respectively. In other words, the control objective is to design the control signal $u(t)$ such that the generated aerodynamic thrust $T(t)$ tracks its desired reference value $T_r(t)$, which is calculated by the software, *i.e.*, `OpenFAST`. The tangent linearization of (3) is calculated as follows:

$$\dot{T}(t) = \Big(\frac{\partial f}{\partial T} + \frac{\partial g}{\partial T}u^*\Big)\Big|_{T=T^*}\Delta T(t) + g(T^*)\Delta u(t), \quad (4)$$

where T^* and u^* are the values of thrust and control signal corresponding to the operating condition where the system is linearized around, $\Delta T = T - T^*$, and $\Delta u = u - u^*$. The nominal values of the parameters adopted from the data sheet of the motor's manufacturer are as follows:

$$\begin{aligned} &J = 1.1\times10^{-4}, \quad k_f = 2.5\times10^{-5}, \quad k_e = 0.043,\\ &k_t = 0.043, \quad k_l = 1.5\times10^{-7}, \quad v_b = 50,\\ &R_m = 0.078, \quad k_a = 1.1\times10^{-5}. \end{aligned} \quad (5)$$

Three operating points (OP) have been selected and the corresponding transfer functions have been calculated below:

$$\begin{aligned} &\text{OP 1} \rightarrow T = 0.99 \rightarrow \frac{T(\hat{s})}{u(\hat{s})} = \frac{1654}{\hat{s}+225.4}\\ &\text{OP 2} \rightarrow T = 3.97 \rightarrow \frac{T(\hat{s})}{u(\hat{s})} = \frac{3308}{\hat{s}+226.2}\\ &\text{OP 3} \rightarrow T = 8.91 \rightarrow \frac{T(\hat{s})}{u(\hat{s})} = \frac{49.62}{\hat{s}+227}, \end{aligned} \quad (6)$$

where $\hat{s}$ is the Laplace operator.

4. Design of the thrust controller

The aim of this section is to design the control signal $u(t)$ for the system (2) such that T tracks T_r. To this end, from the reviewed class of controllers in Sec. 1, four controllers, *i.e.*, SMC, FFC, GSPI, and fuzzy control methods are selected and designed for the system in Secs. 4.1 to 4.4, respectively. Moreover, to handle the time-delay exists in the control loop, the Smith prediction is addressed in Sec. 4.5.

4.1. Sliding-mode control

As was seen in the introduction, several different types of controllers have been proposed for applications with propeller-based thrust generators. Since the system (2) exhibits nonlinear behavior, an SMC has been developed in this section for (2). The motivation for such a selection is that the SMC can potentially handle the perturbations in nonlinear systems [47]. Moreover, its structure is relatively simple, and compared to the adaptive controllers it

needs fewer resources to implement. The design and analysis of the continuous-time SMC are presented in Secs. 4.1.1 and 4.1.2, respectively. Subsequently, a time-discretization method is proposed in Sec. 4.1.3 to implement the SMC on digital computers.

4.1.1. Continuous-time sliding-mode control

The theory of the SMC presented in this section can be found in [47, 48]. The following sliding surface is selected in this study:

$$s(t) = e(t) = T(t) - T_r(t). \tag{7}$$

As can be seen from (7), the sliding surface does not contain the derivative terms of the measurements to avoid the differentiation of noisy measurements. The equivalent control (u_{eq}), *i.e.*, the required effort to keep the nominal system on the sliding surface is calculated based on the following condition:

$$\dot{s}(t) = 0 \Rightarrow \big(\dot{T}(t) - \dot{T}_r(t)\big) = 0. \tag{8}$$

Substituting $\dot{T}(t)$ from (2) into (8) gives

$$\begin{aligned} &\dot{s}(t)|_{u=u_{eq}} = 0 \Rightarrow \bar{f} + \bar{g}u_{eq} - \dot{T}_r = 0 \quad \Rightarrow \\ &u_{eq}(t) = \frac{1}{\bar{g}(t)}\left[-\bar{f}(t) + \dot{T}_r(t)\right] \Rightarrow \\ &u_{eq}(t)|_{\dot{T}_r(t)=0} = -\frac{\bar{f}(t)}{\bar{g}(t)}, \end{aligned} \tag{9}$$

where $\bar{f}(t)$ and $\bar{g}(t)$ are the nominal values of $f(t)$ and $g(t)$, respectively. The SMC control signal $u(t)$ contains $u_{eq}(t)$ as well as a discontinuous signal $u_c(t)$ as follows:

$$u(t) = u_{eq}(t) + u_c(t). \tag{10}$$

In this study, the continuous-time set-valued controller is considered as follows:

$$u_c(t) \triangleq -\alpha \operatorname{sgn}\big(s(t)\big), \tag{11}$$

where sgn is the set-valued signum function defined below:

$$\operatorname{sgn}(s) \triangleq \begin{cases} -1 & \text{for} \quad s \in \mathbb{R}^- \\ [-1,+1] & \text{for} \quad s = 0 \\ +1 & \text{for} \quad s \in \mathbb{R}^+, \end{cases} \tag{12}$$

where the value of this function for $s = 0$ belongs to the set $[-1, 1]$. It will be shown in Corollary 1 that this property can lead to a chattering-free implementation.

4.1.2. Stability analysis of the continuous-time SMC

The stability of the system is studied in Theorem 1.

Theorem 1 ([48]). *The system* (2) *under the continuous-time control law* (9) *to* (11) *is globally finite-time stable.*

Poof. Considering the Lyapunov function V and (7):

$$\begin{aligned} &V(t) \triangleq \tfrac{1}{2}s^2(t) > 0,\ \forall s(t) \neq 0 \ \Rightarrow\ \dot{V}(t) = s(t)\dot{s}(t) \\ &\dot{V}(t) = s(t)\left[(\dot{T}(t) - \dot{T}_r(t))\right]. \end{aligned} \tag{13}$$

The regulation case is considered in this work, *i.e.* $\dot{T}_r = 0$. From (2) and (10), and substituting u_{eq} from (9) into (11), one has:

$$\begin{aligned} &\dot{s}(t) = \ d(t) - g(t)\alpha \operatorname{sgn}\big(s(t)\big) \quad \Rightarrow \\ &\dot{V}(t) = s(t)\left[d(t) - g(t)\alpha \operatorname{sgn}\big(s(t)\big)\right] \quad \Rightarrow \\ &\dot{V}(t) = -g(t)\alpha|s(t)| + |s(t)|d(t). \end{aligned} \tag{14}$$

Assuming that the uncertainties can be expressed as the bounded matched disturbance expressed by

$$d = f - \frac{g}{\bar{g}}\bar{f}, \tag{15}$$

For

$$\alpha > \frac{|d(t)|}{g(t)}, \tag{16}$$

one has $\dot{v} = s\dot{s} \leq -\eta|s|$ and a $\eta > 0$, for all $s(t) \neq 0$, which indicates the robust global finite-time stability of the continuous-time sliding-mode control system [48]. ■

4.1.3. Discretization of the SMC

A time-discretization method has to be used in order to obtain a discrete form of the continuous-time SMC appropriate for implementation on digital computers. In this context, the Euler forward discretization (known also as the Euler explicit discretization) has been used in almost all references as follows:

$$\begin{cases} u(k) = u_{eq}(k) + u_c(k) \\ u_c(k) = -\alpha \operatorname{sgn}\big(s(k)\big), \end{cases} \tag{17}$$

where the input argument k indicates the time step at $k = t/h$, and h is the sampling time. It is known in the control community that such a discretization leads to the digital chattering [59], *i.e.*, the high-frequency oscillations, where the amplitude of the chattering depends on the control gain α. According to (16), a large α might be necessary to ensure robustness, which, in turn, increases the chattering. Several methods, *e.g.*, using a boundary layer [47], adaptive control gain [60], and state-dependent control gain [61] have already been proposed in the literature presenting their own drawbacks, *e.g.*, performance degradation and increasing the complexity of the control algorithm. This study addresses the problem of digital chattering by applying a suitable discretization method instead of altering the continuous-time structure of the SMC (9) to (11). This method is known as the "emulation", which does not necessarily lead to the same behavior when the controller is directly designed in the discrete-time configuration. More clearly, the Euler backward discretization is used to achieve a discrete-time SMC without numerical chattering. Such a time-discretization method (also known as the implicit discretization) has been studied for different systems, *e.g.*, linear time-invariant systems [49], classic SMCs [62–65], twisting SMC [66, 67], super-twisting SMC [68], linear complementarity systems [69], as well as differentiators [43], and it will be specifically adapted for (2) in this work.

The Euler backward discretization of the set-valued term u_c (see (10) and (11)) reads as:

$$\begin{cases} u(k) = u_{eq}(k) + u_c(k) \\ u_c(k) \in -\alpha \operatorname{sgn}\big(s(k+1)\big). \end{cases} \tag{18}$$

It can be seen, in (18), that the control signal, in the Euler backward discretization, at each time step $u(k)$ depends on the set-valued control at the next time step $k+1$ which is not the case for the forward discretization (17). Note that the inclusion ($\in$) is used in (18) to emphasize that the sgn function (12) is a set-valued function. The Euler backward discretization of $\dot{s}(t)$ (first row of (14)) for the unperturbed system, *i.e.*, $d(t) = 0$, gives:

$$\begin{cases} q_1\big(s(k+1)\big) = q_2\big(s(k+1)\big) \\ q_1\big(s(k+1)\big) = s(k+1) - s(k) \\ q_2\big(s(k+1)\big) \in -\beta \operatorname{sgn}\big(s(k+1)\big) \\ \beta = h\alpha. \end{cases} \tag{19}$$

Equation (19) is a generalized equation since q_1 is a singular value while q_2 is a set. The graphical interpretation shown in Fig. 4 is used in this study to solve this equation. In this figure, $q_1\big(s(k+1)\big)$ and $q_2\big(s(k+1)\big)$ are illustrated, and the intersection of these two graphs indicates the solution of the generalized equation (19) as explained below:

- Case 1 $\big(s(k) < -\beta\big)$: In this case, one has $s(k+1) < 0$, leading to $\operatorname{sgn}\big(s(k+1)\big) = -1$. Hence (19) gives:

$$s(k+1) = \beta + s(k). \tag{20}$$

- Case 2 $\big(|s(k)| \leq \beta\big)$: From Fig. 4, for this case, $s(k+1) = 0$. According to (19), it indicates that $s(k) \in \beta \operatorname{sgn}\big(s(k+1)\big)$, and considering (12):

$$\begin{aligned} &s(k) = \beta\gamma \ \text{ for some } \ \gamma \in [-1, 1] \Rightarrow \\ &\operatorname{sgn}\big(s(k+1)\big) = \gamma = \frac{s(k)}{\beta}. \end{aligned} \tag{21}$$

- Case 3 $\big(s(k) > \beta\big)$: From Fig. 4, one can see that the intersection of two graphs always leads to $s(k+1) > 0$, and therefore, $\operatorname{sgn}\big(s(k+1)\big) = 1$. The value of $s(k+1)$ can be calculated as follows:

$$s(k+1) = -\beta + s(k). \tag{22}$$

The flowchart of the developed backward SMC is presented in Fig. 5. The convergence of the developed discrete-time SMC algorithm is studied in Theorem 2.

Theorem 2. *Considering the unperturbed system ($d(k) = 0$), the tracking error $e(k) = T(k) - T_r(k)$ under the developed discrete-time control, shown in Fig. 5, converges to zero in finite-time (see [59] for similar results).*

Proof. The convergence will be studied for two phases as follows:

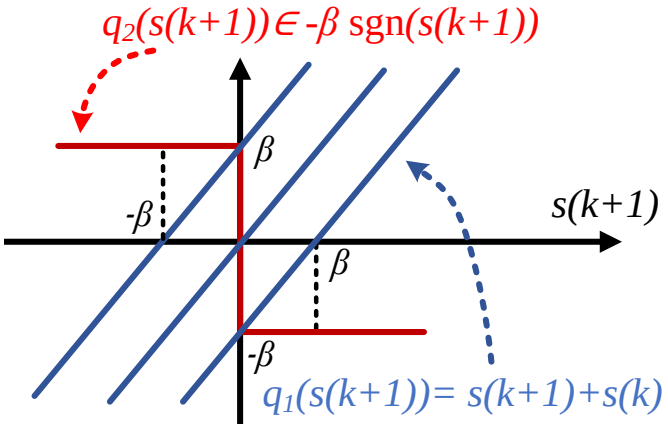


Figure 4: Graphical interpretation of the generalized equation (19), where intersection of these two plots shows the solutions of (19).

- Reaching phase: The reaching phase indicates that $s(k) \neq 0$. Three cases are studied:
  - Case 1: In this case, $s(k) < -\beta$. From (20), one can see that the value of $s(k)$ will increase by time, and for a sufficiently large k, one has $|s(k)| < \beta$. It means that Case 1 always leads to Case 2.
  - Case 3: For this case, one has $s(k) > \beta$. Considering (22), the value of $s(k)$ will decrease. Similar to Case 1, Case 3 always leads to Case 2 ($|s(k)| < \beta$) for a sufficiently large k.
  - Case 2: It was shown that Cases 1 and 3 always lead to Case 2. From (19) and (21), it can be seen that as soon as the system reaches Case 2, *i.e.* $|s(k)| < \beta$ at time step k_s, for all $k > k_s$ one always has $s(k+1) = 0$, indicating that the system starts the sliding phase one step after.
- Sliding phase ($s(k) = 0$): According to (7), in this case, one has $s(k) = 0 \rightarrow T(k) = T_r$ meaning that the sliding phase for such a controller is equivalent to the reference tracking.

Note that while the Lyapunov theorem Theorem 1 does not say anything about the convergence time, the discretization allows studying the required time for the convergence. ■

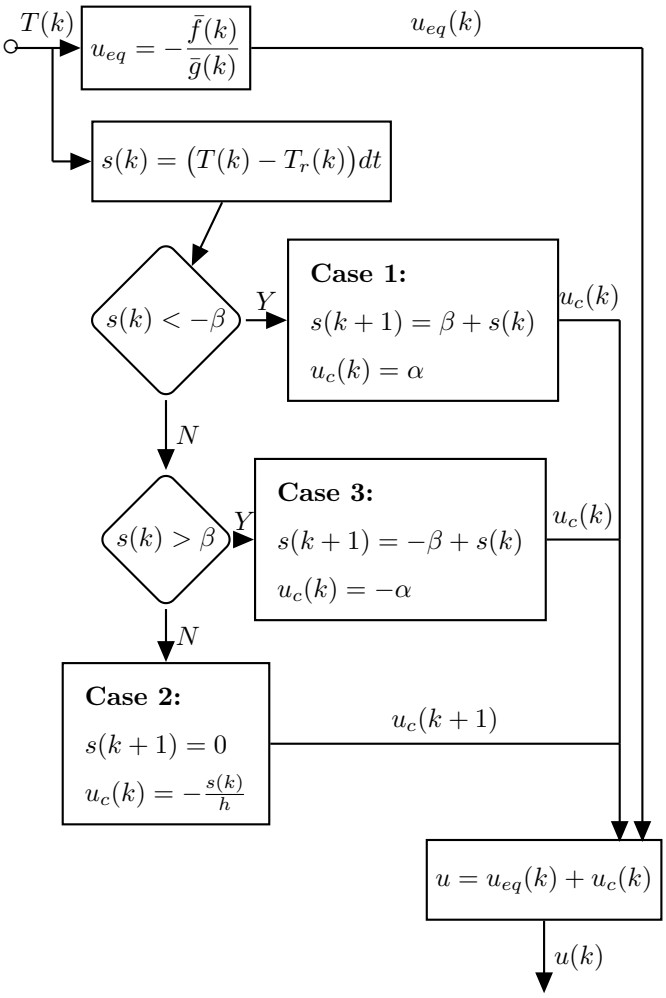


Figure 5: Flowchart of the SMC based on the developed Euler backward discretization. Note that the value of $s(k+1)$ is not used for the next time step in the blocks corresponding to Cases 1 and 3.

Corollary 1. *The backward time discretization, presents an inherent numerical chattering suppression without any modification on its continuous-time counterpart* (9) *to* (11).

Proof. From (21), once $s(k_c) = 0$ is achieved, for all $k > k_c$, $s(k) = 0$ will hold. It indicates that the system under the backward discretization stays in the origin thereafter and presents an inherent chattering suppression without modifying its continuous-time form (9) to (11). ■

Remark 1. *Note that the dynamics of systems may be slow enough allowing to increase the sampling time in order to reduce the calculation burden required to synthesize the control signal. However, increasing the sampling time may not be possible when using the forward discretization because of the numerical chattering augmentation. Such a drawback does not exist in the backward discretization*

as shown in Corollary 1, and therefore, the backward discretization may allow increasing the sampling time and therefore, it may need fewer calculation resources compared to the forward counterpart.

Corollary 2. *The discrete-time SMC under the backward discretization is insensitive to the gains during the sliding phase.*

Proof. As can be seen in Fig. 5, during the sliding phase, *i.e.*, Case 2, the parameters α and β do not appear in the control law $u(k)$ indicating the gain-insensitivity of the developed discretization method under the sliding phase. ■

Remark 2. *While the convergence of the developed time-discretization has been shown for the unperturbed case,* i.e., $d = 0$*, it seems that the feedback can compensate for the perturbations. Considering the flowchart shown in Fig. 5, the value of the $s(k+1)$ is not used for the next step (see the blocks corresponding to Cases 1 and 3), and $s(k)$ is calculated based on the measurements at time step $s(k)$. Hence, the perturbation is compensated at each time step without accumulating its effect on the next time steps.*

Remark 3. *According to* (9)*, u_{eq} is calculated based on the nominal parameters $\bar{f}(t)$ and $\bar{g}(t)$ but not the real $f(t)$ and $g(t)$ because of the uncertainties exists in the model* (2) *and* (3)*. Hence, it may not guarantee the sliding phase. To solve this issue, the static values of u_{eq} have been calculated based on a series of experiments in the open-loop setting for a range of different inputs. To this end, the gain of the system has been calculated for 100 points uniformly selected in the interval* $[0\ 1]$ *to form a lookup table. Subsequently, the inverse of the lookup table along with a linear interpolation method has been used to calculate u_{eq} without using* (9)*. Such an approach has been used for the FFC approach in the literature (see Sec. 4.2). However, the authors couldn't find a trace of such a solution in the literature for the SMC.*

Remark 4. *According to* (16)*, increasing α leads to better robustness to the matched uncertainty d* (15)*. According to Corollary 2, the backward discretization leads to a gain-insensitive implementation for the unperturbed system. However, the presence of the measurement noise on T (and therefore s according to* (7)*) can disturb the system in Case 2 and push it toward Cases 1 and 3 where the gain α appears in the control law (see Fig. 5). Hence, a trade-off should be made between the robustness to the perturbation d* (15) *and the measurement noise on T. This is also the case for the forward discretization of the SMC* (17)*. Unlike the backward discretization, the gain α always appears in the control law leading to the numerical chattering, meaning that $s = 0$ never holds. Hence, in addition to the measurement noise, the sampling time h should be taken into account when tuning α for the forward discretization to avoid large numerical chattering.*

Remark 5. *As detailed in [70], the sliding mode controller under backward implementation precisely compensates for the matched perturbation upon activation of the sliding phase. In practice, the precise compensation turns into compensation with a one-sampling-time delay when a backward discretization is applied, as shown in equation (119) in [70].*

Remark 6. *According to Fig. 5, the backward discretization of the SMC* (9) *to* (11) *is as simple as the projection of the measured sliding variable at time step t_k. This implies that the backward discretization does not incur extra computational resources compared the the forward one, as previously reported in [65] (see also equation (118b) in [70]).*

Remark 7. *Numerical simulation of the whole components is not straightforward since the interaction between the fluid,* i.e.*, water, and solid bodies (here FWT) must be studied by the so-called computational fluid dynamics (CFD) software. Such software has not yet been developed for this specific system. Nevertheless, it is possible to perform the numerical simulations by neglecting the hydrodynamic effect leading to loss of degrees-of-freedoms corresponding to the FWT body position and velocity. In this case, the mathematical model presented in* (2) *and* (3) *can be used for the numerical simulations to study the inner control loop with an arbitrary thrust reference T_r. The readers are encouraged to see [59] for similar pure numerical results.*

Note that the acronyms F-SMC and B-SMC stand for the SMCs which are implemented based on the forward (17) and backward (Fig. 5) discretizations, respectively.

4.2. Open-loop feedforward compensator

According to the literature, [2–10], the FFC is the only method used to control the generated thrust in the FWTs with SIL architecture. In this method, the static gain of the system with duty-cycle u as the input and generated thrust T as the output (see (2)) is calculated for one hundred different values of u, selected uniformly in the interval $[0\ 1]$. Afterward, the inverses of the calculated static gains are used to form a lookup table. Subsequently, this lookup table along with a linear interpolation method is employed as the FFC to achieve a unit static gain in the

feedforward path, *i.e.*, $\lim_{t\to\infty} T(t)/T_r(t) = 1$ as shown in Fig. 6. In other words, the whole control signal u in the FFC is equivalent to u_{eq} in the SMC, *i.e.*, $u = u_{eq}$ (see Remark 3). Considering (10), one can see that the term u_c containing the feedback terms does not appear in the feedforward compensation method. As a result, compared to the SMC, it is not able to compensate for the perturbations. This fact will be validated in Sec. 6.

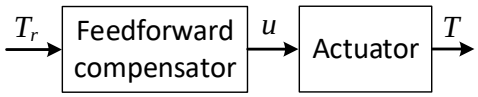


Figure 6: Diagram of the feedforward compensator

4.3. Gain-scheduling PI control

Following the literature review presented in Sec. 1, the PI control has been used for most propeller-based thrust generators, *e.g.*, quadrotors. However, it has not been yet implemented for the SIL FWTs to control the generated force. In this study, three PI controllers are designed for the linearized systems around three operating points listed in (6) (see [71] for the theory of the PI control and [50] for its gain-scheduling version). The PI control reads as:

$$u(t) = k_p e(t) + k_i \int e(t)dt, \tag{23}$$

where $e(t)$ is defined in (7) and k_p and k_i are the proportional and integral gains, respectively, to be designed.

Remark 8. *In this study, the PID tuner that exists in the* `MATLAB` *software package is used to design the parameters listed in Table 1. These parameters are obtained for the settling time 0.012s with* 6% *of overshoot and are scheduled according to the value of the T_r indicated in Table 1.*

4.4. Fuzzy control

The concept of the fuzzy controller used in this study has been derived from Sec. 13 of [51] where the aim is to implement the PI control strategy through the fuzzy logic, leading to a nonlinear control strategy. Compared to the gain-scheduling PI control, the fuzzy control leads to a continuous control law without discontinuous switches. All the design has been made based on the *fuzzy system designer* that exists in the `LabVIEW` software package.

To design a fuzzy controller, a series of input and output membership functions are defined to convert the crisp sets into fuzzy ones. Subsequently, a set of fuzzy rulers are used to implement the desired control law. These two stages are explained in Secs. 4.4.1 and 4.4.2, respectively.

4.4.1. Fuzzy membership functions

The inputs of the fuzzy controller are the error $e(k)$ and the change of error $de(k)$ and the only output is the change of the control signal $du(k)$. These so-called crisp variables need to be assigned to the linguistic fuzzy sets using the defined membership functions shown in Fig. 7. These membership functions are composed of seven linguistic variables, *i.e.*, negative maximum, negative high, negative, zero, positive, positive high, and positive maximum. The ranges of these membership functions are selected based on the behavior of the real system observed in the experiments [51]. Moreover, while there is no rule to determine the shape of the membership functions, the triangular ones are usually selected for ease of analysis [51].

4.4.2. Fuzzy rules

The fuzzy controller is a linguistic implementation of the PI control. According to [51], the fuzzy rules corresponding to the PI control are not unique and can be implemented using different combinations of linguistic variables. In this work, 23 fuzzy rules are used for this purpose which are not given in the paper for the sake of space. Instead, the input/output relationship corresponding to the fuzzy rules is shown in Fig. 8. The diagram of the implemented fuzzy controller is illustrated in Fig. 9, where c_p, c_d, and c_o are the proportional input, derivative input, and output constants, respectively which are presented in Table 1.

Remark 9. *The fuzzy controller is designed based on the system's behavior rather than the dynamic equations. As*

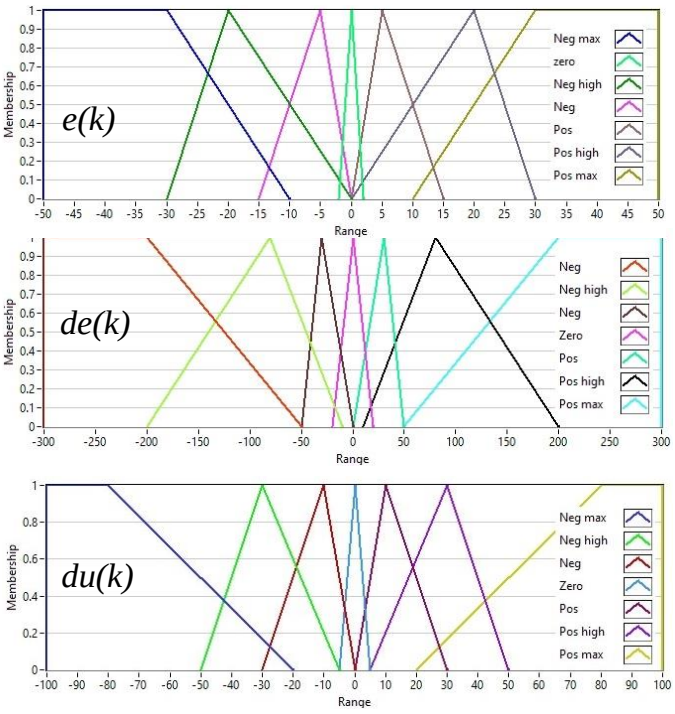


Figure 7: Fuzzy membership functions

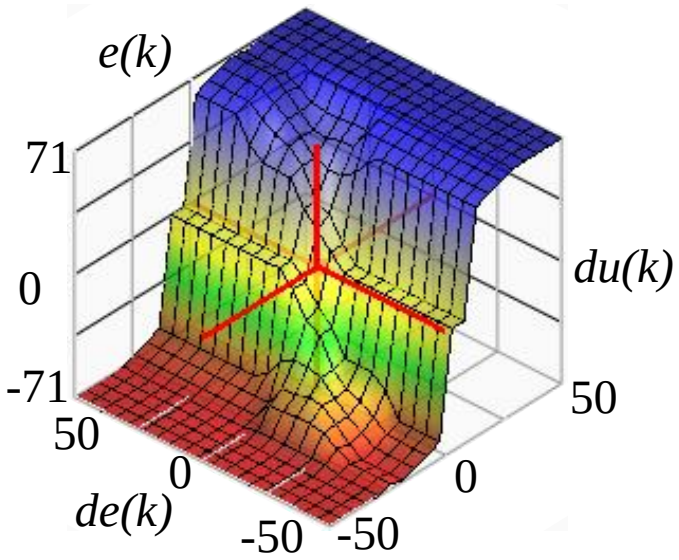


Figure 8: Fuzzy membership functions

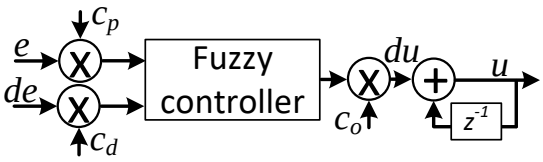


Figure 9: Synthesize the control signal using the fuzzy control block. Note that z^{-1} is the unit delay operator.

a result, there is no straightforward way to tune its parameters. In fact, the membership functions and the fuzzy rules are designed based on the system's behavior observed in the experiments. Three parameters of this controller, i.e., *c_p, c_d and c_o can be tuned based on the same arguments that exist in PI control tuning. Since the output of the fuzzy block is the change of control inputs, the values of c_p and c_d correspond to the integration and proportional gains. Moreover, the value of c_o should be scaled well in order to avoid the signal saturation while ensuring a short enough transient time.*

4.5. Smith predictor

The above-mentioned closed-loop controllers, *e.g.*, GSPI, fuzzy, and SMC, may not be able to provide a satisfactory performance because of the time delay that has not been yet taken into account in the modeling and design. Such a delay is caused by the data acquisition module (`NI-9401`) and PWM signal generator (`NI-9237`), as well as the required time to synthesize the control signal implemented on the computer (`NI-cRIO-9049`). A very basic solution to compensate for such a delay in linear systems is to use a Smith predictor [53, 54]. In this method, a controller is designed without considering the time delay. Subsequently, the effect of the time delay is compensated as follows:

$$T_c(k) = T(k) - P(z)(1 - z^{-\gamma})u(k-1), \qquad (24)$$

where $T_c(k)$ is the compensated measured torque sent to the controller instead of $T(k)$, $P(z)$ is the discrete-time transfer function of the plant without delay, and $z^{-\gamma}$ is the delay operator with γ steps of delay. Such a time delay compensation method has been only developed for linear systems and may not be efficient for the nonlinear system (2). Moreover, the exact amount of the time delay γ is unknown for the system. Note that the transfer function corresponding to OP 2 is selected to calculate $P(z)$.

5. Signal differentiation

The measured motion vector $\tilde{x}(t)$ can be written as follows (see Fig. 3):

$$\tilde{x} = [\tilde{x}_n^T(t), \tilde{x}_p^T(t)]^T, \qquad (25)$$

where $\tilde{x}_n \in \mathbb{R}^6$ and $\tilde{x}_p \in \mathbb{R}^6$ are the motions of the nacelle and the platform, respectively. Assuming that the measurements are polluted by additive noise, one has:

$$\tilde{x}(t) = x(t) + n_x(t), \tag{26}$$

where $x(t)$ is the real position and $n_x(t)$ is the measurement noise. As can be seen in Fig. 3, the `OpenFAST` software package requires both the motion vector $x(t) \in \mathbb{R}^{12}$ and the velocity vector, *i.e.*, its time-differentiation $\tilde{v}(t) \in \mathbb{R}^{12}$. The `QTM` can directly measure the motion $\tilde{x}(t)$. However, the velocity vector cannot be measured and has to be estimated using a differentiator. Some reviews of the differentiators are available in [43, 72, 73]. In this study, four known differentiators, *i.e.*, Euler, Linear filter (LF), and super-twisting differentiators (STD) are used to estimate the time differentiation of the motion vector $\tilde{x}(t)$. The differentiators are briefly reviewed in Secs. 5.1 to 5.3.

5.1. Euler differentiator

The Euler differentiator calculates the time difference of the signal divided by the sampling time as follows:

$$\tilde{v}_k = \frac{\tilde{x}_k - \tilde{x}_{k-1}}{h}. \tag{27}$$

Considering (26), the Euler differentiator calculates the differentiation of the whole signal including the noise $n_x(t)$, leading to the noise amplification.

5.2. Linear filter

To avoid the noise amplification that exists in the Euler differentiator, LFs are usually preferred instead of the Euler differentiation. The LF is composed of an Euler differentiator and a low-pass linear filter. The LF used in this study contains a first-order linear filter as follows:

$$\tilde{v}_k = \frac{\tilde{v}_{k-1} + c(\tilde{x}_k - \tilde{x}_{k-1})}{1 + hc}, \tag{28}$$

where c determines the cutoff frequency of the filter to be tuned.

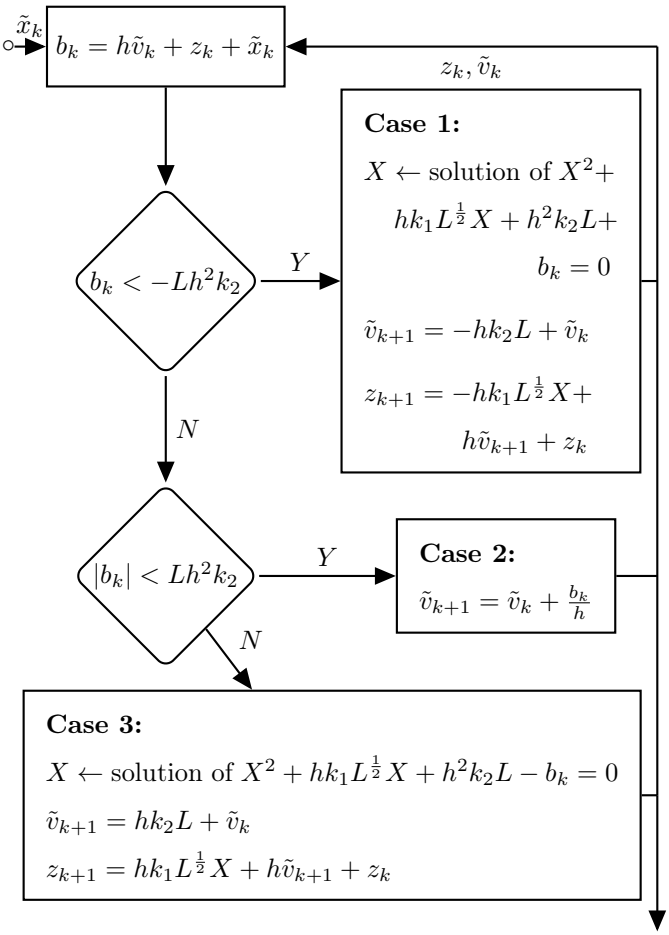


Figure 10: Flowchart of the STD derived from [43]

5.3. Super-twisting differentiator

The LF always shows a phase lag depending on the cutoff frequency c [43]. Such a phase lag usually impacts the performance of closed-loop control systems. Hence, the exact differentiators, *e.g.*, STD are proposed to achieve a phase lag-free implementation, in theory, [43]. The continuous-time form of the STD reads as:

$$\begin{cases} \dot{z}(t) = -k_1 L^{\frac{1}{2}} \lceil (z(t) - \tilde{x}(t)) \rfloor^{\frac{1}{2}} + \tilde{v}(t) \\ \dot{\tilde{v}}(t) \in -k_2 L \operatorname{sgn}(z(t) - \tilde{x}(t)), \end{cases} \tag{29}$$

where k_1, k_2 are constants presented in [43], L is a parameter to be tuned, and $z(t)$ is an estimation of $x(t)$. Moreover, $\lceil \cdot \rfloor^{1/2} = |\cdot|^{1/2} \operatorname{sgn}(\cdot)$, and sgn is the set-valued signum function defined before. In order to implement (29), two different time discretizations, *i.e.*, forward [43, Sec. 4] and backward discretizations are used as explained

in detail in [43, Sec. 5]. The acronyms F-STD and B-STD stand for the STD implemented under forward and backward discretizations of the STD. In contrast to the F-STD, which is widely spread in the literature, the B-STD is more recent with the flowchart presented in Fig. 10.

6. Experiments

The experiments made in this study are presented in this section, where the hardware development is explained in Sec. 6.1, the conditions considered in the experiments are mentioned in Sec. 6.2, and the experimental results are given in Sec. 6.3.

6.1. Hardware development

According to Fig. 11, the implemented system contains two loops that shall be executed in real time. The outer loop is dedicated to `OpenFAST` calculations. This loop receives a wind profile as well as the motion ($\tilde{x}$) and velocity ($\tilde{v}$) of the wind turbine's body and calculates the corresponding aerodynamic thrust reference T_r. This aerodynamic thrust is a reference for the internal loop, where the designed controllers are implemented to synthesize the control signal $u(k)$. This control signal is sent to the actuators, *i.e.*, three propellers installed on the wind turbine's nacelle (see Fig. 3) to generate the aerodynamic thrust such that $T(k) \to T_r$. The generated thrust causes the motions $x(k)$ on the body and this motion is sent back to the outer loop to generate the new reference T_r at each time instant.

The calculation loops introduced above have been implemented on the computer `NI-cRIO-9049` from `National Instruments`. This computer allows assigning different CPUs to execute different real-time tasks without interrupting each other. An analog multi-component load cell is installed on the nacelle to measure the thrust generated by the actuators[4]. A `NI-9237` analog full-bridge input module transfers the measured force to the computer. Moreover, the generated control signal $u(k)$ is sent to the actuator as a PWM signal generated through a `NI-9401` digital output module.

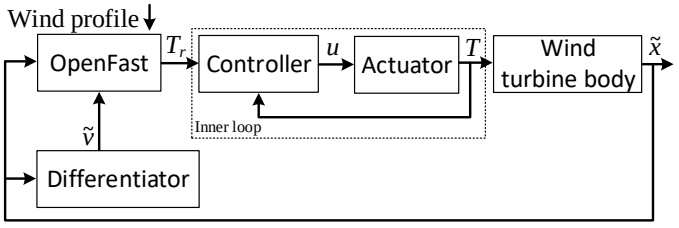


Figure 11: Calculation loops implemented on the computer.

6.2. Condition of the experiment

As it was mentioned before, a $1/40th$ scale FWT is considered in this study. To achieve realistic results, the parameters of the aerodynamic model (`OpenFAST` in this study) need to be scaled using the Froude scaling law [52]. To this end, the parameters corresponding to the length and time must be multiplied by 40 and $\sqrt{40}$, respectively (see [52] for more details the on Froude scaling law, commonly used in wave tank model testing).

All the experiments are carried out with the same wind and wave conditions to enable both deterministic and statistical comparisons. The simulated wind profile (given to the `OpenFAST` according to Fig. 11) is uniform with a constant velocity of 16 m/s given here at full-scale. It should be noted that a turbine controller [74] exists in the `OpenFAST` software to control the blade pitch angle corresponding to region III (the controller design for the turbine and analyzing different regions is out of the scope of this research. The readers are encouraged to see [75]). The wave profile generated physically in the test tank is an irregular wave (random sea) that is typically seen in a real installation site. The significant height and peak period of the generated wave are 7 meters and 12 seconds, respectively, at the full scale. Each experiment lasts for

[4]The force is composed of the thrust, gravity of the above part, and the inertial force that is corrected (removing the inertial force and the weight) through analytical calculations and an inertial measurement unit.

20 minutes at the model scale which corresponds to two hours for the full-scale FWT.

The parameters of the SMC (F-SMC, B-SMC), GSPI, and fuzzy controllers are designed based on Remark 4, Remark 8, and Remark 9, respectively. Moreover, the parameters of the differentiators have been tuned using the differentiation toolbox presented in [43, 56]. Following this reference, the integration of the first-order differentiation, *i.e.*, $\bar{x} = \int \tilde{v}(k)dt$ is used to recover the position $\bar{x}(k)$. Subsequently, $\bar{x}(k)$ is compared with the measured position $\tilde{x}(k)$ to tune the parameters based on the following performance function:

$$J = ||\bar{x}(k) - \tilde{x}(k)|| + \xi \,\mathrm{var}(\bar{x}(k)), \tag{30}$$

where $\xi = 10^{-4}$ is a constant to make a trade-off between the exactness and noise filtration and,

$$\begin{aligned} &\mathrm{var}(y(k)) = \textstyle\sum_k |y(k) - y(k-1)| \\ &||y(k)|| = \sqrt{\textstyle\sum_k y^2(k)} \end{aligned} \tag{31}$$

The term $||\bar{x}(k) - \tilde{x}(k)||$ guarantees the convergence of the differentiators while $\mathrm{var}(\bar{x}(k))$ ensures a proper level of measurement noise filtration. The parameters of the controllers and differentiators used in the experiments are listed in Table 1.

All the possible combinations of the controllers and differentiators are implemented and the summarized results under the condition explained in Sec. 6.2 are presented in Tables 3 to 5. Considering the control laws corresponding to the F-SMC (17), B-SMC Fig. 5, FFC Fig. 6, GSPI control (23), and fuzzy control Fig. 9, one can see that the GSPI and fuzzy controllers do not contain the FF terms and are totally dependent on the feedback. For such controllers, the delay in the control loop Fig. 11 causes large overshoots and even oscillations in the steady-state response. Hence, as indicated in Table 2, the Smith predictor (24) compensates for the delay for such controllers. For both F-STD and B-STD, the feedback is directly used without the Smith predictor since, according to (17), the equivalent control $u_{eq}(k)$ (9) is calculated based on Remark 3 which is not affected by the delay existing in the loop. In addition, the experiments show no improvements when using the Smith predictor for the SMC. Note that the FFC does not use any feedback terms and therefore is independent of the feedback delay.

Table 1: Parameters of the controllers and differentiators

Method	Parameters
Controllers	
FFC	No parameter
GSPI	$T_r < 4 : k_p = 0.2, k_i = 79.7$
	$4 \leq T_r \leq 17 : k_p = 0.1, k_i = 40.1$
	$17 < T_r : k_p = 0.07, k_i = 27$
Fuzzy	c_f=150 c_p=0.1 c_d=2 c_o=0.09
F-SMC	$\alpha = 3$
B-SMC	$\alpha = 3$
Differentiators	
Euler	No parameter
LF	c=20
F-STD	L=0.63
B-STD	L=0.63

6.3. Results of the experiments

To compare the results obtained from different configurations, two performance functions, *i.e.*, variation (var(.)),

Table 2: Status of the Smith predictor for different configurations. ✓ and ✗ indicate the presence and absence of the Smith predictor, respectively.

		Differentiator			
		Euler	LF	F-STD	B-STD
Controller	FFC	✗	✗	✗	✗
	GSPI	✓	✓	✓	✓
	Fuzzy	✓	✓	✓	✓
	F-SMC	✗	✗	✗	✗
	B-SMC	✗	✗	✗	✗

Table 3: $||e(k)||$ corresponding to all combinations of the controllers and differentiators

		Differentiator			
		Euler	LF	F-STD	B-STD
Controller	FFC	1366.1	1333.4	1246.4	1342.0
	GSPI	223.3	147	Osc.	195.2
	Fuzzy	175.8	126.45	Osc.	161.6
	F-SMC	295.6	251.6	282.4	284.9
	B-SMC	182.5	148.4	Osc.	167.5

Table 4: var($\tilde{v}(k)$) corresponding to all combinations of the controllers and differentiators

		Differentiator			
		Euler	LF	F-STD	B-STD
Controller	FFC	36046.9	3009.8	3863.2	14089.1
	GSPI	42146.6	2582.4	Osc.	10911
	Fuzzy	39905.9	2550.6	Osc.	10542
	F-SMC	41968.6	2574.5	3069.35	10697.3
	B-SMC	42416	2580.6	Osc.	10762.8

and the second norm $||\cdot||$ are used as defined in (31). The value of $||e(k)||$ indicates the L_2 norm of the tracking error which is listed in Table 3 for different configurations. A smaller amount of this value indicates better tracking performance. Moreover, var is used to quantify the chattering that exists in the system. In this case, var($\tilde{v}(k)$) in Table 4 and var($u(k)$) in Table 5 indicate the chattering exists on the estimated velocity and the control signal, respectively.

As can be seen, five controllers, *i.e.*, FFC, GSPI, fuzzy, F-SMC, and B-SMC can be integrated with four differentiators, *i.e.*, Euler, LF, F-STD, and B-STD, leading to 20 possible configurations. Since the separation principle does not exist for nonlinear systems, it is not clear how to study the effect of the controller and differentiators separately. In this study, it is first tried to see the effect of the differentiators for a specific controller, *i.e.*, B-STD, as shown in Fig. 12. This figure denotes that the Euler differentiator shows the maximum amount of chattering

Table 5: var($u(k)$) corresponding to all combinations of the controllers and differentiators

		Differentiator			
		Euler	LF	F-STD	B-STD
Controller	FFC	47623	12440	27634	40647
	GSPI	223922	80298	Osc.	176976
	Fuzzy	225282	220477	Osc.	224703
	F-SMC	121208	57914	73717	99684
	B-SMC	168884	43769	Osc.	123496

on the estimated velocity $\tilde{v}(k)$ since the Euler is basically a pure differentiator without any filtration (see (27)). To study the phase lag exerted by the differentiators, the estimated velocities are integrated to recover the position $\bar{x}(k)$ shown in Fig. 12. From this figure, while the LF leads to the smoothest velocity estimation, it shows the maximum amount of phase lag. Such a phase lag can deteriorate the performance of the closed-loop control system. On the other hand, while the B-STD shows better noise filtration than the Euler method, it presents a smaller amount of phase lag. As can be seen in Fig. 12, it seems that the estimations provided by the F-STD are not reliable and diverge from the real one. The reason is that this differentiator shows a large amount of chattering that hinders to increase in the gain L (see (29)) leading to poor convergence. Note that, during the experiments, the F-STD usually leads to oscillations with the resonant frequency of the FWT model which is considered as the excitation of the unmodeled dynamics. In such conditions, for safety, the experiments are stopped by the operator, and the word Osc. is written instead of the results to indicate the instability state (see Tables 3 to 5). This instability is caused by the excitation of the 1st bending modes of the FWT body at a frequency around 3.7 Hz. Before the experiments, the Hammer test is made on the structure to identify the vibration modes of the FWT. Such a test revealed the existence of two modes appearing around 3.7 Hz and

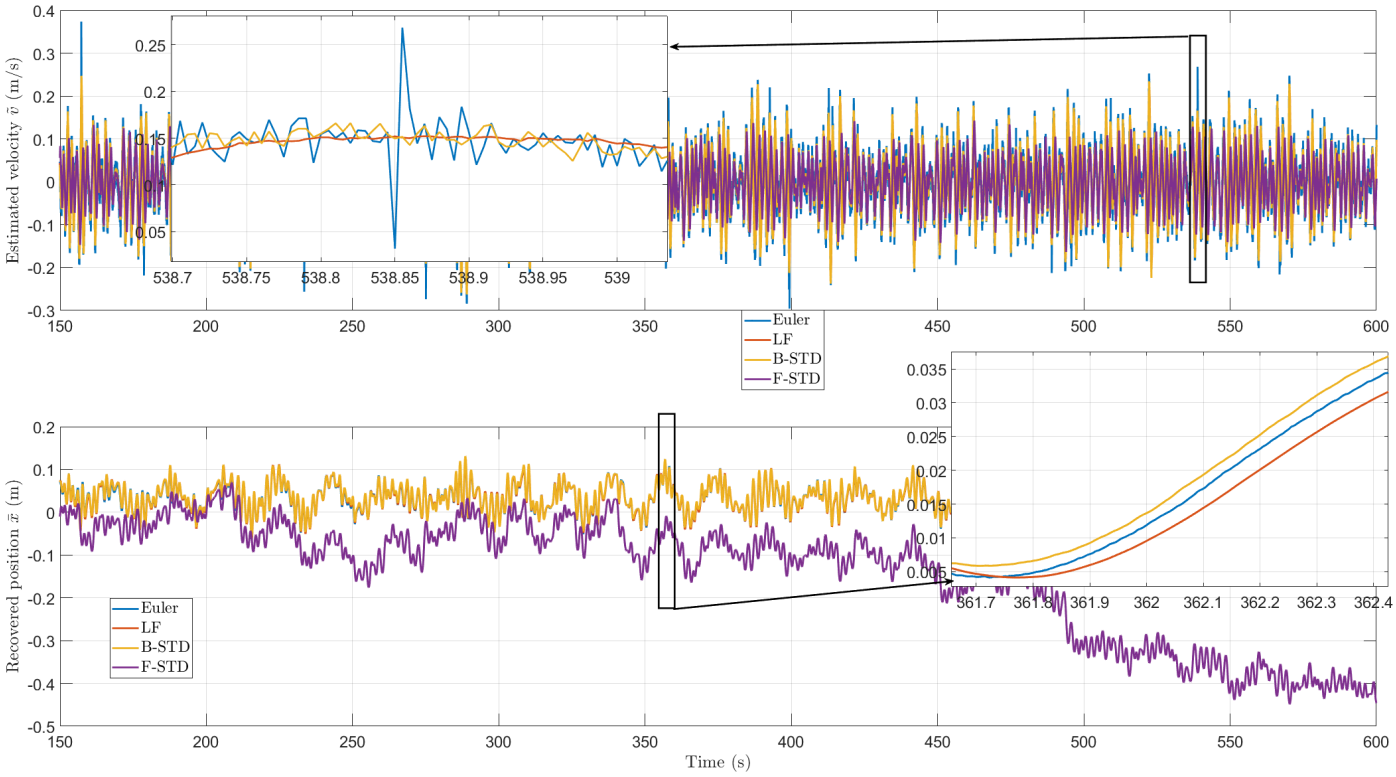


Figure 12: Estimated velocities and the recovered position (integration of the estimated velocity) when the B-SMC is in the loop.

13 Hz, corresponding to the tower and nacelle structures. While the tower is an aluminum cylinder, the nacelle is made from carbon fiber to obtain the required lightweight structure. However, it is known that carbon fiber leads to high-frequency vibrations because of its flexible structure.

The tracking error $e(k)$ corresponding to all possible configurations of the controllers and differentiators is given in Table 3. To evaluate the performances of the controllers, each column should be considered separately. The reason is that, according to Fig. 11, the differentiators directly affect the reference thrust T_r calculated by `OpenFAST`, and therefore each differentiator may lead to different T_r. Considering each column separately, one can see that the fuzzy control always provides the minimum tracking error. Moreover, the B-STD also shows good responses close to the fuzzy control. In global, the GSPI shows average tracking performances compared to others. According to Table 3, the FFC leads to the worst responses because it is mainly an open-loop method that does not use the feedback to compensate for the perturbations. For this specific case, the FFC presented in Fig. 6 is sensitive to the perturbations, *e.g.*, ambient temperature change and battery voltage variation. Comparing the F-SMC and the B-SMC, while these two controllers are derived from the same continuous-time settings and parameters (see (16) and Table 1), their performances are quite different in the discrete-time form as can be seen in Tables 3 to 5.

Considering the rows of Table 4 separately, the LF always leads to the smallest amount of variation on the estimated velocity, according to the values of $\text{var}(\tilde{v}(k))$, indicating that this differentiator shows the best noise filtration ability. However, such a good noise filtration leads to the largest time delay among other differentiators. On the other hand, the Euler differentiator shows the maximum amount of variation. It should be noted that the B-STD shows one of the maximum values for $\text{var}(\tilde{v}(k))$ after the Euler method. According to [43], the backward discretization provides a digital chattering-free implemen-

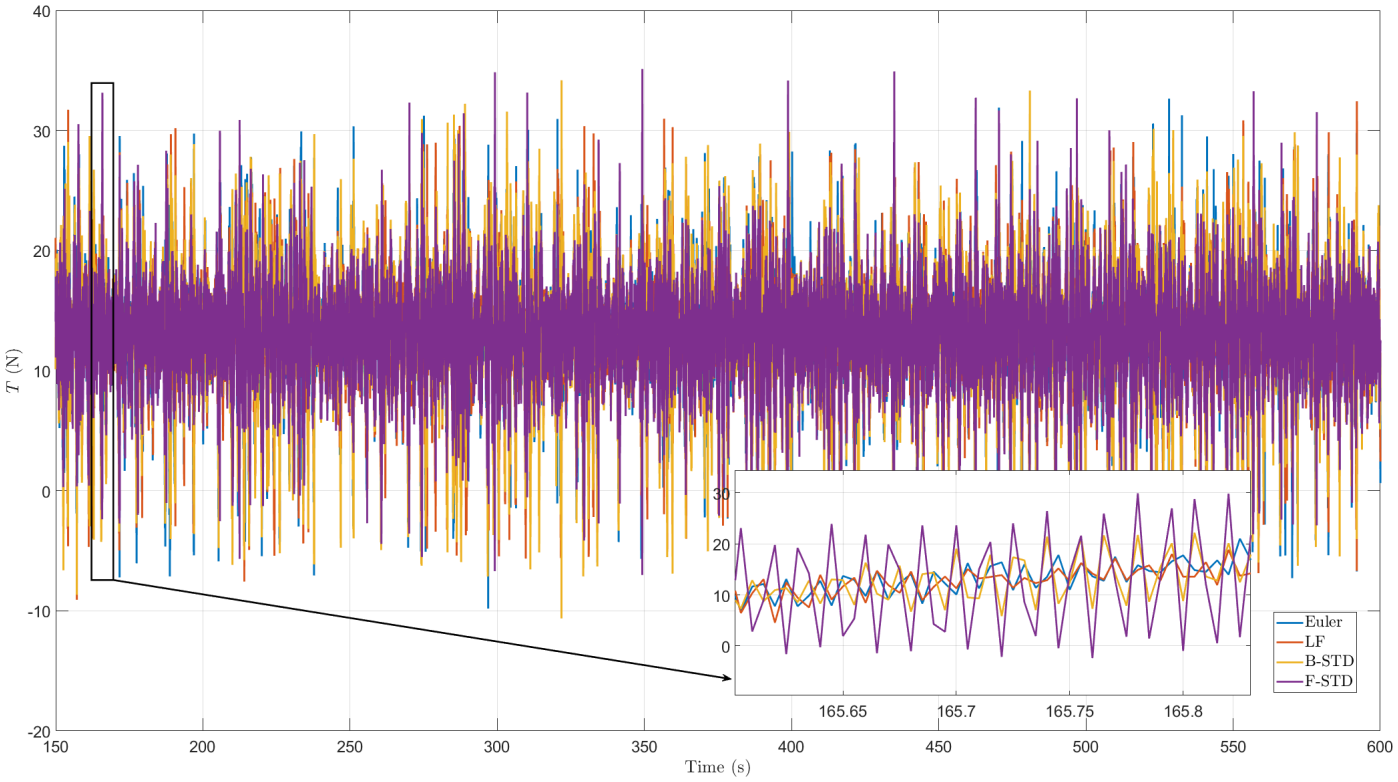


Figure 13: Generated thrust corresponding to the B-SMC integrated with different differentiators

tation under noise-free conditions. Hence, it seems that the variations that exist in the B-STD are caused by the measurement noise and the exactness of this differentiator on both the signal and the measurement noise. Unlike the other case studies that have already been considered [76, 76], the sampling time in this work is $h = 5$ ms. For such a small sampling time (compared with $h = 50$ ms in [43, 76]), the dominant source of the variation is the measurement noise but not the digital chattering. Hence, for the small sampling times, the backward discretization of the differentiator may not bring advantages over the forward one. In any case, even for such a small sampling time, the B-STD shows the minimum phase lag while the velocity estimation corresponding to the F-STD diverges.

As can be seen in Table 5, the FFC shows the minimum amount of variation on the control signal var$(u(k))$ since this method does not use the feedback and therefore is insensitive to the measurement noise. Another observation is that the fuzzy control always shows one of the maximum values for var$(e(k))$. In fact, the bandwidth of this controller is higher than the other controllers since according to Table 3 it shows the smallest $||e(k)||$. However, such a large bandwidth amplifies the noise effect leading to a large variation on the control signal.

According to Fig. 13, the F-STD shows the largest amount of variation on the generated thrust. Such variations, can lead to oscillatory movements corresponding to excitation of the vibration modes leading to instability as indicated in Tables 3 to 5. Hence, it seems that even for the small sampling time ($h = 5$ ms) the backward discretization of the controllers can provide fewer variations on the systems' output compared to the forward counterpart.

The synthesized control signals corresponding to the SMC under forward and backward discretizations are shown in Fig. 14 when the STD is in the loop and discretized under both forward and backward discretizations. The first observation is that all control signals present a significant amount of chattering which may correspond to

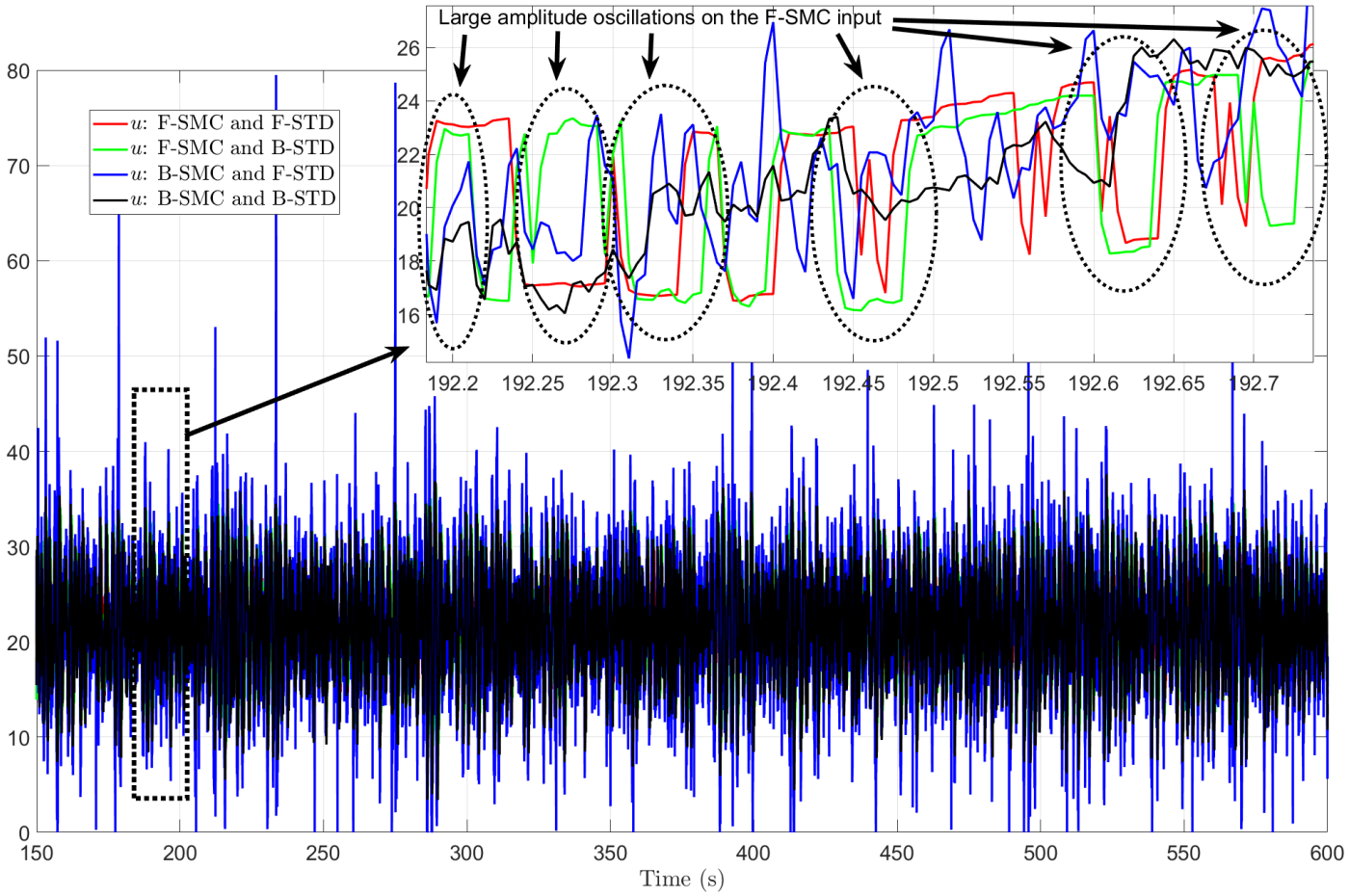


Figure 14: Control signal when the SMC and STD are in the loop. Four combinations of the forward and backward discretizations of the controller and differentiator are presented.

the measurement noise and the numerical (digital) chattering caused by the forward discretization. Compared to other experimental studies, *e.g.*, [73], this process is highly affected by the measurement noise since the nacelle is made up of a carbon fiber structure. Such a structure is not enough rigid and presents flexibilities in different directions leading to high-frequency measurement noise. While the sliding-mode controller is robust against disturbances, it is generally sensitive to measurement noise. Hence, the ideal sliding phase seen in the theory may not be always achieved. Therefore, increasing the gains affects both forward and backward discretizations. Moreover, increasing the gains can amplify the noise effect as well as numerical chattering (corresponding to the forward discretization) leading to the excitation of unmodeled dynamics like two bending modes of the FWT body at frequencies around 3.7 Hz and 13 Hz mentioned above. As a result, the gains must be selected carefully for all methods including the forward and backward implementations.

It is worth noting that according to Fig. 14, with the backward differentiator, the backward controller presents fewer high-amplitude variations compared to the case where the forward controller is used. The reason is that in the presence of a controller under forward discretization, the mentioned 13 Hz nacelle bending mode is excited leading to vibrations in the nacelle structure and therefore high-amplitude oscillations shown in Fig. 14 which push away the F-SMC from the sliding phase results in a degraded tracking (compare F-SMC and B-SMC in

Table 3). In such conditions, the F-SMC is rather in the reaching phase to compensate for the unmodeled excited dynamic leading to a small chattering (compare two last lines in Table 5). Such excitation is less observable when the control input is synthesized based on the backward method. This is an important observation that should be considered as a new advantage of the backward discretization which has not yet been addressed in the literature. More clearly, it seems that high-frequency chattering corresponding to F-SMC excites the nacelle bending mode at 13 Hz leading to high-frequency vibration on the acelle structure. Afterward, with appearing such an unmodeled dynamic, the F-SMC tries to achieve the tracking by treating this unmodeled oscillatory dynamic as a disturbance. As a result, according to Fig. 14, the F-SMC shows large amplitude efforts on the control signal to compensate such perturbation (six large amplitude efforts are visible during 0.5 ms indicating the compensation of the 13Hz bending mode corresponding to the nacelle). In this condition, the F-SMC, according to the fourth row of Table 5 shows less chattering compared to the B-SMC on the last row, which may be misleading. In fact, such a smaller variation of the F-SMC is because the sliding mode is not activated for the F-SMC caused by the excitation of the mentioned unmodeled dynamics. In summary, to achieve the minimum variation, both the controller and differentiator should be implemented using backward discretization.

7. Conclusions

This work studied the controller design for a reduced-scale FWT based on the SIL architecture where the software is used in the loop to calculate the aerodynamic force to be generated by the propellers. To this end, a backward time discretization method has been developed for a typical SMC in order to achieve a chattering-free implementation without modifying the continuous-time setting while keeping several useful properties, *e.g.*, finite-time convergence, and gain-insensitivity. Subsequently, four controllers, *i.e.*, FFC, GSPI control, fuzzy control, and forward SMC are derived from the literature and implemented for the comparisons. Moreover, since the software needs the velocity of the wind turbine motions, four differentiators, *i.e.*, Euler, LF, and STD (both forward and backward discrete forms) are used to estimate the velocity and the effect of the differentiation methods on the overall performance of the closed-loop control system is studied in detail.

The experimental results show that the fuzzy controller always presents the minimum tracking error in terms of the L_2 norm. However, this controller shows the maximum amount of variation in the control signal caused by the measurement noise and the high bandwidth of the fuzzy controller. On the other hand, the FFC shows the worst tracking performance because it relies on the static input-output gain of the plant calculated offline which varies depending on the environment, *e.g.* ambient temperature and battery voltage. Among other controllers, the GSPI shows an average performance. The F-SMC excites the nacelle banding mode around 13 Hz which can be considered as the excitation of unmodeled dynamics. The F-SMC treats such unmodeled dynamic as perturbation and, as can be seen in Fig. 14, it shows high amplitude oscillations in the control to compensate for it. In this case, the F-SMC is rather in the reaching phase (but not the sliding phase) because of the excited unmodeled dynamics leading to a small chattering compared to the B-SMC (compare two last rows in Table 5), which should not be misleading. In any case, it can be seen that even for such a small sampling time 5 ms compared to other works with larger sampling times (see [73]), the SMC should be implemented based on the backward discretization to avoid the excitation of such unmodeled dynamics.

Considering the differentiators, it seems that the effect of the discretization is quite visible compared to the discretization of the controller reported above, where the

chattering generated by the F-STD may even lead to instability (see the word Osc. in Table 5). On the other hand, considering the phase lag, the B-STD shows the minimum amount of phase lag leading to better closed-loop performances among all differentiators. The Euler differentiation presents the maximum variation since the Euler method is a pure differentiation without any filtration causing high-frequency noise amplification.

As it was mentioned above, the SMC is designed based on the propeller and motor dynamics without taking into account the structural flexibilities. As a result, under specific conditions such unmodeled dynamics are excited and appeared in the results. The authors believe that the effect of the discretization on the unmodeled dynamics corresponding to the structural bending modes should be studied in more detail for other applications.

Acknowledgements: This work was supported by the ANR project CREATIF, France (ANR-20-CE05-0039). The authors also thank the experimental team at LHEEA / Centrale Nantes - CNRS, in particular: Laurent Davoust, Sylvain Haquin, Stéphane Lambert, Anne Lesvesque, Sylvain Mazo, and Matthieu Weber who contributed to the experimental campaign and the corresponding database.

References

[1] A. S. Darwish, Floating Offshore Wind Technology Development: Current State and Future Vision, Springer International Publishing, 2022, Ch. 72, pp. 651–659.

[2] A. Otter, J. Murphy, V. Pakrashi, A. Robertson, C. Desmond, A review of modelling techniques for floating offshore wind turbines, Wind Energy 25 (2022) 831–857.

[3] J. Azcona, F. Bouchotrouch, M. González, J. Garciandía, X. Munduate, F. Kelberlau, T. A. Nygaard, Aerodynamic thrust modelling in wave tank tests of offshore floating wind turbines using a ducted fan, Journal of Physics: Conference Series 524 (1) (2014) 012089.

[4] F. Vittori, F. Bouchotrouch, F. Lemmer, J. Azcona, Hybrid Scaled Testing of a 5MW Floating Wind Turbine Using the SiL Method Compared With Numerical Models, in: International Conference on Offshore Mechanics and Arctic Engineering, Vol. Volume 10: Ocean Renewable Energy, 2018.

[5] J. Azcona, F. Bouchotrouch, F. Vittori, Low-frequency dynamics of a floating wind turbine in wave tank–scaled experiments with sil hybrid method, Wind Energy 22 (2019) 1402–1413.

[6] C. Matoug, B. Augier, B. Paillard, G. Maurice, C. Sicot, S. Barre, An hybrid approach for the comparison of vawt and hawt performances for floating offshore wind turbines, Journal of Physics: Conference Series 1618 (3) (2020) 032026.

[7] E. Oguz, D. Clelland, A. H. Day, A. Incecik, J. A. López, G. Sánchez, G. G. Almeria, Experimental and numerical analysis of a tlp floating offshore wind turbine, Ocean Engineering 147 (2018) 591–605.

[8] A. Meseguer, R. Guanche, Wind turbine aerodynamics scale-modeling for floating offshore wind platform testing, Journal of Wind Engineering and Industrial Aerodynamics 186 (2019) 49–57.

[9] O. Pires, J. Azcona, F. Vittori, I. Bayati, S. Gueydon, A. Fontanella, Y. Liu, E. de Ridder, M. Belloli, J. van Wingerden, Inclusion of rotor moments in scaled wave tank test of a floating wind turbine using sil hybrid method, Journal of Physics: Conference Series 1618 (3) (2020) 032048.

[10] A. Otter, J. Murphy, C. J. Desmond, Emulating aerodynamic forces and moments for hybrid testing of floating wind turbine models, Journal of Physics: Conference Series 1618 (3) (2020) 032022.

[11] Design and Testing of Scale Model Wind Turbines for Use in Wind/Wave Basin Model Tests of Floating Offshore Wind Turbines, Vol. Volume 8: Ocean Renewable Energy of International Conference on Offshore Mechanics and Arctic Engineering.

[12] H. R. Martin, R. W. Kimball, A. Viselli, A. J. Goupee, Methodology for wind/wave basin testing of floating offshore wind turbines, Journal of Offshore Mechanics and Arctic Engineering-transactions of The Asme 136 (2012) 020905.

[13] T. Sauder, V. Chabaud, M. Thys, E. E. Bachynski, L. O. Sæther, Real-Time Hybrid Model Testing of a Braceless Semi-Submersible Wind Turbine: Part I — The Hybrid Approach, in: International Conference on Offshore Mechanics and Arctic Engineering, Vol. Volume 6: Ocean Space Utilization; Ocean Renewable Energy, 2016.

[14] E. E. Bachynski, M. Thys, T. Sauder, V. Chabaud, L. O. Sæther, Real-Time Hybrid Model Testing of a Braceless Semi-Submersible Wind Turbine: Part II — Experimental Results, in: International Conference on Offshore Mechanics and Arctic Engineering, Vol. Volume 6: Ocean Space Utilization; Ocean Renewable Energy, 2016.

[15] P. A. Berthelsen, E. E. Bachynski, M. Karimirad, M. Thys, Real-Time Hybrid Model Tests of a Braceless Semi-Submersible

Wind Turbine: Part III — Calibration of a Numerical Model, in: International Conference on Offshore Mechanics and Arctic Engineering, Vol. Volume 6: Ocean Space Utilization; Ocean Renewable Energy, 2016.

[16] C. Luan, Z. Gao, T. Moan, Design and Analysis of a Braceless Steel 5-MW Semi-Submersible Wind Turbine, in: International Conference on Offshore Mechanics and Arctic Engineering, Vol. Volume 6: Ocean Space Utilization; Ocean Renewable Energy, 2016.

[17] M. Karimirad, E. E. Bachynski, P. A. Berthelsen, H. Ormberg, Comparison of Real-Time Hybrid Model Testing of a Braceless Semi-Submersible Wind Turbine and Numerical Simulations, in: International Conference on Offshore Mechanics and Arctic Engineering, Vol. Volume 10: Ocean Renewable Energy, 2017.

[18] E. E. Bachynski, V. Chabaud, T. Sauder, Real-time hybrid model testing of floating wind turbines: Sensitivity to limited actuation, Energy Procedia 80 (2015) 2–12, 12th Deep Sea Offshore Wind R&D Conference, EERA DeepWind'2015.

[19] M. Thys, V. Chabaud, T. Sauder, L. Eliassen, L. O. Saether, O. B. Magnussen, Real-Time Hybrid Model Testing of a Semi-Submersible 10MW Floating Wind Turbine and Advances in the Test Method, in: International Conference on Offshore Mechanics and Arctic Engineering, Vol. ASME 2018 1st International Offshore Wind Technical Conference, 2018.

[20] F. Madsen, F. Lemmer, K. Müller, A. Pegalajar-Jurado, R. Faerron-Guzman, H. Bredmose, Model validation against experiments andmap of model accuracy across load cases., project's report LIFES50+D4.6 (2018).

[21] V. Chabaud, L. Eliassen, M. Thys, T. Sauder, Multiple-degree-of-freedom actuation of rotor loads in model testing of floating wind turbines using cable-driven parallel robots, Journal of Physics: Conference Series 1104 (1) (2018) 012–021.

[22] M. Hall, A. J. Goupee, Validation of a hybrid modeling approach to floating wind turbine basin testing, Wind Energy 21 (2018) 391–408.

[23] R. Antonutti, J.-C. Poirier, S. Gueydon, Coupled Testing of Floating Wind Turbines in Waves and Wind Using Winches and Software-in-the-Loop, in: OTC Offshore Technology Conference, Vol. Day 2 Tue, May 05, 2020, 2020.

[24] S. Bouabdallah, A. Noth, R. Siegwart, Pid vs lq control techniques applied to an indoor micro quadrotor, in: 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566), Vol. 3, 2004, pp. 2451–2456 vol.3.

[25] G. Hoffmann, H. Huang, S. Waslander, C. Tomlin, Quadrotor helicopter flight dynamics and control: Theory and experiment, in: AIAA guidance, navigation and control conference and exhibit, 2007, p. 6461.

[26] O. Mechali, L. Xu, X. Xie, J. Iqbal, Theory and practice for autonomous formation flight of quadrotors via distributed robust sliding mode control protocol with fixed-time stability guarantee, Control Engineering Practice 123 (2022) 105150.

[27] K. Guo, J. Jia, X. Yu, L. Guo, L. Xie, Multiple observers based anti-disturbance control for a quadrotor uav against payload and wind disturbances, Control Engineering Practice 102 (2020) 104560.

[28] A. Moutinho, J. R. Azinheira, E. C. de Paiva, S. S. Bueno, Airship robust path-tracking: A tutorial on airship modelling and gain-scheduling control design, Control Engineering Practice 50 (2016) 22–36.

[29] D. Rotondo, F. Nejjari, V. Puig, Quasi-lpv modeling, identification and control of a twin rotor mimo system, Control Engineering Practice 21 (6) (2013) 829–846.

[30] A. Sir Elkhatem, S. Naci Engin, Robust lqr and lqr-pi control strategies based on adaptive weighting matrix selection for a uav position and attitude tracking control, Alexandria Engineering Journal 61 (8) (2022) 6275–6292.

[31] I. Postlethwaite, E. Prempain, E. Turkoglu, M. C. Turner, K. Ellis, A. Gubbels, Design and flight testing of various h_∞ controllers for the bell 205 helicopter, Control Engineering Practice 13 (3) (2005) 383–398, aerospace IFAC 2002.

[32] D. Schafroth, C. Bermes, S. Bouabdallah, R. Siegwart, Modeling, system identification and robust control of a coaxial micro helicopter, Control Engineering Practice 18 (7) (2010) 700–711, special Issue on Aerial Robotics.

[33] L. Sun, K. Wang, A. H. A. Mishamandani, G. Zhao, H. Huang, X. Zhao, B. Zhang, A novel tension-based controller design for the quadrotor–load system, Control Engineering Practice 112 (2021) 104818.

[34] O. Mechali, L. Xu, Y. Huang, M. Shi, X. Xie, Observer-based fixed-time continuous nonsingular terminal sliding mode control of quadrotor aircraft under uncertainties and disturbances for robust trajectory tracking: Theory and experiment, Control Engineering Practice 111 (2021) 104806.

[35] M. Bhargavapuri, S. R. Sahoo, M. Kothari, Abhishek, Robust nonlinear control of a variable-pitch quadrotor with the flip maneuver, Control Engineering Practice 87 (2019) 26–42.

[36] D. Cabecinhas, R. Cunha, C. Silvestre, A nonlinear quadrotor trajectory tracking controller with disturbance rejection, in: 2014 American Control Conference, 2014, pp. 560–565.

[37] L. Martins, C. Cardeira, P. Oliveira, Inner-outer feedback linearization for quadrotor control: two-step design and validation, Nonlinear Dynamics 110 (1) (2022) 479–495.

[38] K. Alexis, G. Nikolakopoulos, A. Tzes, Switching model predictive attitude control for a quadrotor helicopter subject to atmospheric disturbances, Control Engineering Practice 19 (10)

(2011) 1195–1207.

[39] C. Liu, W.-H. Chen, J. Andrews, Tracking control of small-scale helicopters using explicit nonlinear mpc augmented with disturbance observers, Control Engineering Practice 20 (3) (2012) 258–268.

[40] M. Yazid, H. Bouadi, N. Hamdadou, N. Hebablia, Hexarotor attitude stabilization based feedback linearization and sliding mode controllers, in: 2021 29th Mediterranean Conference on Control and Automation (MED), 2021, pp. 1143–1148.

[41] S. Vargas, H. M. Becerra, J.-B. Hayet, Mpc-based distributed formation control of multiple quadcopters with obstacle avoidance and connectivity maintenance, Control Engineering Practice 121 (2022) 105054.

[42] H. Ushida, T. Yamaguchi, K. Goto, T. Takagi, Fuzzy-neuro control using associative memories, and its applications, Control Engineering Practice 2 (1) (1994) 129–145.

[43] M. R. Mojallizadeh, B. Brogliato, V. Acary, Time-discretizations of differentiators: Design of implicit algorithms and comparative analysis, International Journal of Robust and Nonlinear Control 31 (16) (2021) 7679–7723.

[44] C. Bak, F. Zahle, R. Bitsche, T. Kim, A. Yde, L. C. Henriksen, M. H. Hansen, J. P. A. A. Blasques, M. Gaunaa, A. Natarajan, The DTU 10-mw reference wind turbine, Tech. rep., Danish wind power research (2013).

[45] G. M. Hoffmann, H. Huang, S. L. Waslander, C. J. Tomlin, Precision flight control for a multi-vehicle quadrotor helicopter testbed, Control Engineering Practice 19 (9) (2011) 1023–1036.

[46] P. E. Pounds, R. E. Mahony, P. I. Corke, Design of a static thruster for microair vehicle rotorcraft, Journal of Aerospace Engineering 22 (1) (2009) 85–94.

[47] J.-J. E. Slotine, W. Li, et al., Applied nonlinear control, Prentice hall Englewood Cliffs, NJ, 1991.

[48] A. Fossard, T. Floquet, Introduction: An overview of classical sliding mode control, in: Sliding mode control in engineering, CRC Press, 2002, Ch. 1, pp. 22–49.

[49] O. Huber, V. Acary, B. Brogliato, Comparison between explicit and implicit discrete-time implementations of sliding-mode controllers, in: 52nd IEEE Conference on Decision and Control, 2013, pp. 2870–2875.

[50] D. J. Leith, W. E. Leithead, Survey of gain-scheduling analysis and design, International Journal of Control 73 (11) (2000) 1001–1025.

[51] T. J. Ross, Fuzzy logic with engineering applications 3rd ed., John Wiley & Sons, Inc., 2010.

[52] V. Arnal, Modélisation expérimentale d'une éolienne flottante par une approche sofware-in-the-loop., Ph.D. thesis, LHEEA, thèse de doctorat dirigée par Aubrun-Sanches, Sandrine Mécanique des Milieux Fluides Ecole centrale de Nantes 2020 (2020).

[53] O. J. Smith, Closer control of loops with dead time, Chemical engineering progress 53 (1957) 217–219.

[54] V. L. Korupu, M. Muthukumarasamy, A comparative study of various smith predictor configurations for industrial delay processes, Chemical Product and Process Modeling 17 (6) (2022) 701–732.

[55] A. Levant, Robust exact differentiation via sliding mode technique, Automatica 34 (3) (1998) 379–384.

[56] M. R. Mojallizadeh, B. Brogliato, V. Acary, Discrete-time differentiators: design and comparative analysis, Report on INRIA-HAL: hal-02960923 (2021) 7679–7723.

[57] M. Kreider, F. Oteri, A. Robertson, C. Constant, E. Gill, Offshore wind energy: Technology below the water, Tech. rep., National Renewable Energy Lab.(NREL), Golden, CO (United States) (2022).

[58] J. López-Queija, E. Robles, J. Jugo, S. Alonso-Quesada, Review of control technologies for floating offshore wind turbines, Renewable and Sustainable Energy Reviews 167 (2022) 112787.

[59] B. Brogliato, A. Polyakov, Digital implementation of sliding-mode control via the implicit method: A tutorial, International Journal of Robust and Nonlinear Control 31 (9) (2021) 3528–3586.

[60] M. R. Mojallizadeh, M. A. Badamchizadeh, Second-order fuzzy sliding-mode control of photovoltaic power generation systems, Solar Energy 149 (2017) 332–340.

[61] M. R. Mojallizadeh, M. Badamchizadeh, S. Khanmohammadi, M. Sabahi, Designing a new robust sliding mode controller for maximum power point tracking of photovoltaic cells, Solar Energy 132 (2016) 538–546.

[62] R. Kikuuwe, S. Yasukouchi, H. Fujimoto, M. Yamamoto, Proxy-based sliding mode control: A safer extension of pid position control, IEEE Transactions on Robotics 26 (4) (2010) 670–683.

[63] Y. Yamamoto, J. Qiu, Y. Munemasa, T. Doi, T. Nanjo, K. Yamashita, R. Kikuuwe, A sliding-mode set-point position controller for hydraulic excavators, IEEE Access 9 (2021) 153735–153749.

[64] O. Huber, B. Brogliato, V. Acary, A. Boubakir, F. Plestan, B. Wang, Experimental results on implicit and explicit time-discretization of equivalent control-based sliding mode control, in: Recent Trends in Sliding Mode Control, Control, Robotics and Sensors, Institution of Engineering and Technology, 2016, pp. 207–235.

[65] V. Acary, B. Brogliato, Y. V. Orlov, Chattering-free digital sliding-mode control with state observer and disturbance rejection, IEEE Transactions on Automatic Control 57 (5) (2012) 1087–1101.

[66] O. Huber, V. Acary, B. Brogliato, F. Plestan, Discrete-time

twisting controller without numerical chattering: analysis and experimental results with an implicit method, in: 53rd IEEE Conference on Decision and Control, 2014, pp. 4373–4378.

[67] O. Huber, V. Acary, B. Brogliato, Lyapunov stability analysis of the implicit discrete-time twisting control algorithm, IEEE Transactions on Automatic Control 65 (6) (2020) 2619–2626.

[68] O. Huber, V. Acary, B. Brogliato, F. Plestan, Implicit discrete-time twisting controller without numerical chattering: Analysis and experimental results, Control Engineering Practice 46 (2016) 129–141.

[69] B. Brogliato, Analysis of the implicit euler time-discretization of semiexplicit differential-algebraic linear complementarity systems, SIAM Journal on Control and Optimization 60 (4) (2022) 2159–2183.

[70] F. Miranda-Villatoro, F. Castaños, B. Brogliato, When proximal-point algorithms meet set-valued systems. An Optimization point of view of discrete-time sliding modesWorking paper or preprint (hal-04362282) (Dec. 2023).
URL https://inria.hal.science/hal-04362282

[71] F. Golnaraghi, B. C. Kuo, Automatic control systems, McGraw-Hill Education, 2017.

[72] M. R. Mojallizadeh, B. Brogliato, Effect of Euler explicit and implicit time discretizations on variable-structure differentiators, in: Sliding-Mode Control and Variable-Structure Systems: The State of the Art, Vol. 490, Springer, 2023, pp. 165–180.

[73] M. R. Mojallizadeh, B. Brogliato, A. Polyakov, S. Selvarajan, L. Michel, F. Plestan, M. Ghanes, J.-P. Barbot, Y. Aoustin, A survey on the discrete-time differentiators in closed-loop control systems: Experiments on an electro-pneumatic system, Control Engineering Practice 136 (2023) 105546.

[74] A. Hegazy, P. Naaijen, J.-W. van Wingerden, Wave feedforward control for large floating wind turbines, in: 2023 IEEE Conference on Control Technology and Applications (CCTA), 2023.

[75] L. Y. Pao, K. E. Johnson, Control of wind turbines, IEEE Control Systems Magazine 31 (2) (2011) 44–62.

[76] M. R. Mojallizadeh, B. Brogliato, C. Prieur, Modeling and control of overhead cranes: A tutorial overview and perspectives, Annual Reviews in Control (2023) 100877.