Nuclear norm-based recursive subspace identification for wind turbine flutter detections

Navalkar, Sachin; van Wingerden, Jan-Willem

DOI
10.1109/TCST.2017.2692751

Publication date
2017

Document Version
Accepted author manuscript

Published in
IEEE Transactions on Control Systems Technology

Link to published article (IEEE): https://doi.org/10.1109/TCST.2017.2692751

Nuclear Norm-Based Recursive Subspace Identification for Wind Turbine Flutter Detection

Sachin T. Navalkar and Jan-Willem van Wingerden

***Abstract*— Commercial wind turbine blades are progressively becoming longer and more flexible; in order to achieve load reduction, the use of shape modifying devices is currently under research. While such modifications facilitate cost reduction, they also render the blade susceptible to the unstable aeroelastic phenomenon of flutter. To be able to detect the onset of flutter, and to modify the load control algorithm accordingly, it is desirable to perform online identification of system dynamics. In this paper, a recursive subspace identification algorithm is augmented with a nuclear norm-based cost function for the rapid identification of changes in the dominant system behavior. The time-consuming singular value thresholding step involved in the identification is replaced by a fast randomized algorithm. The method developed is used to identify the changes in the dynamics of an experimental wind turbine equipped with shape-modifying actuators, and operated under controlled conditions in a wind tunnel. The proposed identification method shows high sensitivity to changes in system dynamics, and is shown capable of stably and rapidly identifying the onset of aeroelastic flutter.**

***Index Terms*— Nuclear norm, randomized singular value decomposition (SVD), recursive identification, wind energy.**

I. Introduction

WIND TURBINES have over the past decades successively grown in size, in an effort to minimize the production and operation cost of wind energy, especially in the challenging offshore environment. Current wind energy research is attempting to synthesize and validate approaches to minimize wind turbine blade loading, which has a direct impact on the costs of the entire structure, as well as on the longevity of the turbine. One of the approaches toward load control is the use of individual pitch control [1], whereby the blades are pitched cyclically along their longitudinal axes, with a view toward mitigating the cyclic blade loads. While this approach is effective, the large blade inertia ensures that only low-frequency loads can be mitigated. Furthermore, this approach induces a large increase in pitch activity and a concomitant degradation of pitch subsystem reliability.

An alternative approach to blade load control is afforded by trailing edge flaps, located radially outboard along wind turbine blades [2]. Such devices typically have higher bandwidth and can mitigate a larger part of the blade spectrum, while possessing the ability to respond to localized wind disturbances. Although trailing edge flaps have not yet been deployed on commercial machines, their load reduction potential has been demonstrated experimentally in the wind tunnel [3] and in the field [4]. However, it is observed that the control authority of trailing edge flaps is typically lower than that of full-span pitch control. In an attempt to remedy this shortcoming, the concept of the free-floating flap, as described in [5] was introduced, which uses aerodynamic levering to enhance actuator stroke in the frequency range of interest. This concept was recently demonstrated experimentally on a scaled wind turbine in a wind tunnel [7]. In both of these references, it is seen that the system behavior is linear-parameter-varying in that it varies strongly with the ambient wind speed. Furthermore, the dynamic behavior of wind turbines is typically complicated by the presence of periodic exogenous loads arising out of azimuthally varying effects such as wind shear, tower shadow, and rotationally sampled turbulence.

It has been postulated in [8] and demonstrated experimentally by [9] that modern pitch-controlled wind turbines exhibit classical flutter through the aeroelastic coupling of bending and torsion modes. Field experience has shown that flutter does not occur during ordinary operation; however, off-design operating conditions such as overspeed or misaligned flow can induce flutter. Furthermore, with the increasing flexibility of wind turbine blades, the flutter speed is expected to reduce further. The use of trailing-edge flaps has the effect of further increasing the flexibility of the blades. Specifically, free-floating flaps, which show increased control authority with high bandwidth, have been shown experimentally to induce a low wind-speed form of flutter [5] on a fixed wing suspended in a wind tunnel. It should be noted that these flaps, described in further detail in the following section, introduce a new rigid-body blade mode, which couples aeroelastically with the flexible blade flapwise mode to cause instability. As such [6], this instability is distinct from the phenomenon of classical flutter, which typically involves the aeroelastic coupling of two flexible blade modes (usually bending and torsion). However, since the unstable behavior of the free-floating flaps has also been referred to in the literature [5], [6] as flutter, the same word is used in this paper. It should be noted that the methods described for identifying actuator-blade flutter in this paper can also be used for identifying classical flutter in wind turbine blades [10].

At this point, it is important to emphasize the objective of the trailing edge flaps [6]. They are, in the literature, primarily aimed at reducing blade flapwise loading; with other forms of

Manuscript received August 17, 2016; revised January 22, 2017; accepted February 11, 2017. Manuscript received in final form March 7, 2017. Recommended by Associate Editor L. Fagiano. This work was supported by the INNWIND.EU Program, a consortium of academic and industrial partners dedicated toward the development of innovations in advanced wind energy technology. *(Corresponding author: Sachin T. Navalkar.)*

The authors are with the Delft Centre for Systems and Control, Faculty of Mechanical, Maritime and Materials Engineering, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: j.w.vanwingerden@tudelft.nl).

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

Digital Object Identifier 10.1109/TCST.2017.2692751

cyclical loads forming a secondary load reduction objective. Classical overspeed-induced bending-torsion flutter may be another possible application, but this has not been exhaustively studied so far. However, the increase in blade flexibility caused by such devices, specifically by free-floating flaps, may induce another low wind speed form of actuator-blade flutter, which is also required to be identified and stabilized. This paper devotes its attention exclusively to this form of flutter.

In order to ensure structural integrity and load control performance in the event that the wind turbine enters the flutter regime, it would be necessary to track the evolution of system dynamics over time. System identification can be performed online for this purpose. Although parametric models and neural network-based models can also be identified [11], [12], this paper focuses on subspace identification. This is because subspace identification requires minimal system knowledge and typically involves a convex minimization, which is amenable to an online implementation. For the application of flexible wind turbines, due to the high modeling complexity arising out of the intercoupling of aerodynamics, structural dynamics, and control, subspace identification has been considered an attractive option for the synthesis of controller-relevant plant models. Linear system identification at multiple wind speeds has been performed by [13]. To account for the wind speed-varying nature of the plant, [14] describes the identification of turbine vibrational dynamics as an LPV system, while [15] uses subspace identification to model a wind turbine as a Hammerstein system. Such approaches to identification yield more accurate and globally valid models that have greater flexibility to describe the nonlinear behavior of the turbine. However, from the point of view of computational complexity and a limited availability of data, it is considered in this application more appropriate to investigate recursive linear identification.

Subspace methods have been used for online recursive fault identification [16] and for the online synthesis of adaptive control laws [17]. In these references, recursive identification is shown capable of detecting the variation in system dynamics over time. Furthermore, a certain class of predictor-based subspace identification techniques, such as $\text{PBSID}_{\text{opt}}$ [18], can also be used to perform identification in the closed loop, as is clearly required in order to identify an unstable plant, such as a wind turbine in flutter.

The method $\text{PBSID}_{\text{opt}}$ has been implemented recursively, as in [19] using projector tracking. However, in this case, the estimates do not necessarily converge to the same state basis. In order to circumvent this problem, [10] uses a propagator method to arrive at the instantaneous Markov parameters of the underlying plant. However, recursive subspace identification methods typically involve a variance-adaptivity tradeoff. In effect, in order to achieve stable system parameter estimates, it is necessary to utilize forgetting factors close to unity; unfortunately high forgetting factors cause a reduction in the responsiveness of the identification algorithm to changes in system dynamics. As shown in [10], where recursive $\text{PBSID}_{\text{opt}}$ is used to detect flutter, a high forgetting factor can produce a delay in the detection of the point at which the system becomes unstable.

Recent research in identification techniques has focussed on exploiting the low-rank nature of the data matrices by augmenting the traditional optimisation criterion with the nuclear norm [20]. For small-size data batches, nuclear norm-based methods [21] reduce the variance of system parameter estimates at the cost of introducing an estimate bias. While originally developed for open-loop batch identification, the method has been extended to recursive closed-loop identification [22]. In order to optimize a nuclear norm-based cost function recursively and online, the alternating direction method of multipliers (ADMM) [23] can be considered suitable. An ADMM approach can also be expected to be able to handle constraints in the optimization problem. With the addition of the nuclear norm, the estimate variance reduces, allowing the user to reduce the forgetting factor and enable rapid detection of system parameter changes, such as the onset of flutter.

One of the drawbacks of incorporating a nuclear-norm in the cost function is the need for singular value thresholding (SVT) at every ADMM iteration. If a full singular value decomposition (SVD) of the data matrix is computed at every step in the ADMM [23], then the computational cost for a recursive algorithm can easily become prohibitively expensive for a real-time implementation. Cheaper alternatives for SVT have been explored in the literature. For instance, Cai [24] explores SVT without computing an SVD. Instead, a polar decomposition of the argument is followed by projection for norm shrinkage; with reported reductions in computation time of up to 50%. In [25], a nuclear-norm based cost criterion is minimized, replacing the full SVD with an SVD of a reduced randomly sampled full-rank matrix. If, as in our application, the range of the full-rank matrix remains approximately the same across iterations, the computations can be further simplified via a method that the authors term as "range propagation." In [25], the algorithm has been applied to image processing, and significant advantages are observed in comparison with the nonregularized variant. However, this approach has not yet been applied for online system identification.

The contributions of this paper are thus threefold: first, it can be considered as an extension of the conference paper [22], where a closed-loop subspace identification method is augmented with a nuclear-norm cost criterion so as to enhance its robustness to noise, and to enable the responsiveness to changes in system dynamics. In order to make it tractable in real time, use is made of a fast randomized version of SVT, and structure is exploited to reduce computational complexity through the use of range propagation. Since the application is the identification of wind turbine flutter, the designed identification method is extended to compensate for exogenous periodic disturbances typically observed in wind turbine loading data. Finally, the use of the method will be demonstrated for the detection of flutter on an experimental setup of a wind turbine operating in time-varying wind tunnel conditions.

This paper is organized as follows: in Section II, the experimental setup and operating conditions are described. Section III describes the theoretical framework for the online nuclear norm-based identification of system dynamics. Section IV presents the results obtained from the

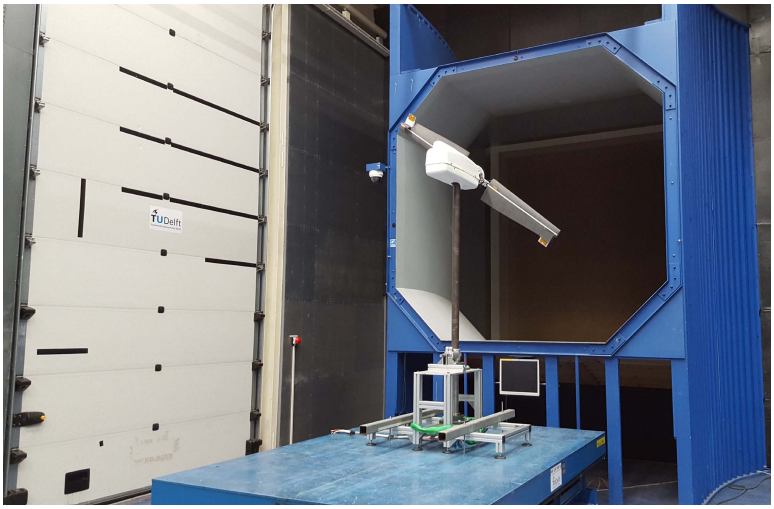

Fig. 1. Photograph of the wind turbine located close to the open jet of the wind tunnel. The rotor diameter is 2 m.

experimental data. Finally, in Section V, conclusions are drawn from the presented results.

II. Experimental Setup

The experimental setup consists of a scaled two-bladed wind turbine operated under controlled wind conditions in a wind tunnel. The wind turbine blades are each equipped with one free-floating flap close to the blade tip. The objective of this flap is to modify air flow in order to achieve blade load reductions. However, the additional degree of freedom causes aeroelastic flutter just outside the turbine operational envelope. A detailed discussion on the design and control of the wind turbine with free-floating flaps can be found in [7]. In Sections II-A–II-C, a brief overview of the experimental setup and operating conditions is given.

A. Open Jet Facility

The open jet facility of the Delft University of Technology is an open-jet wind tunnel with a hexagonal jet outlet, with an effective cross-sectional diameter of 3 m. The fan driving the wind flow in the wind tunnel is rated at 500 kW, with the maximum wind speed achievable 35 m/s. The experiments are conducted at wind speeds between 4 and 7 m/s. The turbulence intensity in the wind tunnel is less than 0.5% at these wind speeds, although it can, in principle, be raised through the use of turbulence screens or gust generators. However, for the present of experiments, since it was more desirable to understand the behavior of the setup and the algorithm under deterministic loading conditions, such flow modifiers were not used. Since the wind tunnel is of the closed-circuit type, it is equipped with a series of heat exchangers to regulate the temperature of the air flow. The wind tunnel is used for the testing of wind turbines under controlled wind conditions, and it has been used for the testing of wind turbine identification and load control validation in the recent past [3], [17]. On account of the dimensions of the open jet flow, wind turbine prototypes tested in this wind tunnel cannot exceed a size of 2-m rotor diameter.

TABLE I
Parameter Comparison of the Scaled Turbine

	Reference turbine [26]	Scaled rotor
Rotor diameter (m)	178.3	2
Rated wind speed (m/s)	11.4	4.5
Tip speed ratio (-)	7.86	5.35
Rated rotational speed (rpm)	9.6	230
Fore-aft tower mode (Hz)	0.25	20.73
First flapwise mode (Hz)	0.56	14.4
Ratio 1st blade freq. to 1P (-)	3.5	3.75

B. Wind Turbine and Blade Design

The scaled turbine used for experiments in the wind tunnel is shown in Fig. 1. The main aeroservoelastic characteristics of the turbine, relevant for blade load control design, have been compared against the 10-MW INNWIND reference turbine [26] in Table I. Since the objective of the experiment is to achieve blade load control, the tip speed ratio was limited to 5.35, which is lower than its optimal value of 7.86, but does not alter the aerodynamic performance significantly. The ratio of the first flapwise frequency of the blade to the rotor speed (1P) was kept as close as possible to the reference value of 3.5. This ratio describes the relation between the primary load forcing frequency, 1P, to the control authority peak, which occurs at the blade structural eigenfrequency.

The detailed aerodynamic design of the blades is given in [7]. Principally, the blades were designed as per [3]. Each blade has a length of 750 mm, with a root chord length of 200 mm that tapers to a tip chord length of 120 mm. The total twist over the blade is 12°. The cross-sectional aerofoil is of type DU96-W180, with 18% thickness. The basic aerodynamic shell was 3-D printed using the material PC-ABS, with a vertebral carbon-fibre layup of external spars to tune the structural strength and stiffness of the blade. The surface was then polished using sandpaper. The blades can be visualized in Figs. 2 and 3. An MFC strain sensor was

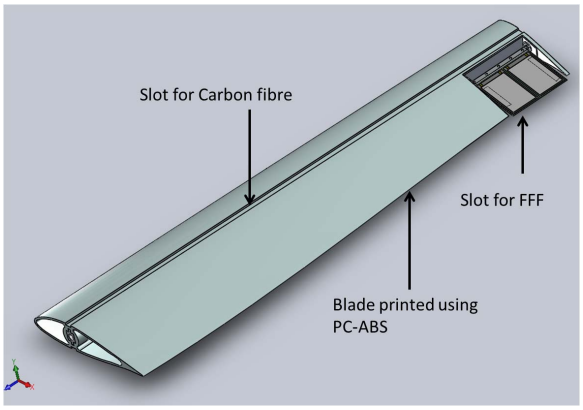

Fig. 2. Blade CAD model.

Fig. 3. Photograph of Blade.

affixed to the top and bottom at the root of each blade, in order to measure the blade loads. Actuation was facilitated by the free-floating flaps located at the tip of each blade. The design rotational speed was 230 rpm, yielding a Reynold's number of roughly 200 000.

For typical wind tunnel experiments for scaled turbine identification and load control, Reynold's scaling is practically infeasible, and was not attempted. The scaling objective was to achieve a reasonable tip speed ratio to replicate aerodynamic behavior, and eigenfrequency scaling to achieve similar structural behavior.

The turbine is fixed rigidly on the wind tunnel table. The tower is able to yaw freely about its base, this degree of freedom is, however, constrained for the duration of the experiments. The turbine is oriented such that the rotor is upwind and its plane is perpendicular to the direction of wind flow. The nacelle is fixed rigidly to the top of the tower and contains a generator and a rotating hub, connected directly without a gearbox. The connecting shaft is instrumented with a torque transducer and speed encoder. Also mounted on the shaft are slip rings that transfer power and control signals from the stationary part of the nacelle to the rotating hub and the blades. The blades are connected to the hub through pitch motors, thus they are able to pitch longitudinally along their axes. However, for the duration of the current experiments, the blade pitch is kept at fine, i.e., set to the maximal power production angle. Further details can be found in [17], which utilizes the same nacelle and tower.

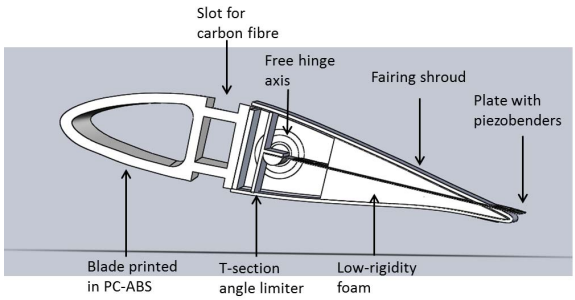

Fig. 4. Cross section of free-floating flap.

C. Actuator Design and the Control System

Blade load control is achieved by using the free-floating flaps located at the blade tips to modify the air flow around the blade. As can be seen in the cross-sectional drawing Fig. 4, the flap is free to move without any resistance about a hinge axis. The flap consists of a thin sheet of metal (spring steel), on the top and bottom of which one MFC piezobender each is bonded. In response to an imposed voltage, the MFC actuator causes the flap to bend either upward or downward. This camberline deformation causes rigid-body rotation of the flap around its hinge axis, producing an aerodynamic force that can be used to counteract blade loading. In order to ensure good aerodynamic behavior while maintaining deformability, the flap is faired to the rest of the blade using low-stiffness foam, and encapsulated in a low-rigidity casing.

As seen in Figs. 2 and 4, the designed chord length is unchanged at the outboard section where the free-floating flap is located. Such a design is typical for this application, and has been shown in the literature to demonstrate adequate control authority. Ensuring a smoothly varying chord length over the span of the blade is also desirable from an aerodynamic point of view. However, out of practical considerations, and to provide a safety margin for the demonstration of free-floating flap load reduction control authority, the chord length at the location of the flap was increased by 20% for the experimental blade [see Fig. 3]. In future work, as for scaled-up versions of this concept, it is expected that such a chord extension will not be deemed necessary nor desirable.

The angle through which the free-floating flap can freely rotate is limited by using angle limiters, and the rotational angle is measured with the help of a contactless angle encoder that is aligned with the free hinge shaft. Since the open-loop system can become unstable due to the phenomenon of flutter, this collocated sensor is used to provide feedback to a simple loop-shaped PID controller that keeps the closed-loop system stable.

The control input and output signals are transferred through the slip rings to the stationary subsystem, and fed into an NI-DAQ data acquisition system, which is connected to a real-time PC running MATLAB-Simulink xPCTarget. The generator power is transferred to an electronic dump load. The dump load resistance can be controlled to regulate the turbine torque and speed. However, for the duration of the experiments, the load resistance is kept constant.

In constant-resistance operation, the rotational speed of the turbine varies linearly with the ambient wind speed, thus this corresponds to the below-rated operational regime of a commercial wind turbine.

As described in Section I and in [7], the experimental blade manifests a low wind-speed form of flutter, which is caused by the coupling of flap rigid-body motion and blade flapwise motion. This paper seeks to identify the onset of this form of flutter. As such, instrumentation is included to obtain a measure of both forms of motion: the blade root strain gauges measures its flexible deflection, while the angle encoder measures flap rigid-body motion. Load control is the ultimate objective of employing the free-floating flaps, but this paper focuses purely on identification. Feedback from the angle encoder is used to stabilize the system in the unstable operating regime.

This experimental setup serves as a test bench for investigating the online nuclear-norm-based identification strategy developed in Section III.

III. Recursive PBSID With the Nuclear Norm

In this section, the algorithm for online system identification and flutter detection is developed. First, the closed-loop PBSID algorithm is augmented with a nuclear norm-based cost function for the identification of low-order systems, while retaining convexity. Then, a recursive implementation of this algorithm is described. Finally, the time-consuming SVD step of the identification algorithm is replaced by a lower-dimensional version to facilitate online implementation.

A. $PBSID_{opt}$ With the Nuclear Norm

In conventional $\text{PBSID}_{\text{opt}}$, the first step involves the synthesis of an extended vector autoregressive with exogenous inputs (VARX) model, using the input and output data. The parameters of this model, also called the Markov parameters of the system, are identified optimally in a least-squares sense. On the basis of this model, the second step involves state estimation and the construction of the state-space matrices. In the first step, the identified VARX model can be forced to be low rank by adding a nuclear norm-based term to the optimization cost function, with a view toward reducing the effect of noise on the estimate variance.

Let us consider that the turbine blade system, from flap input to load output, be represented by linear time-invariant (LTI) dynamics for a given wind speed. In the predictor form, the system description is as follows:

$$x_{k+1} = Ax_k + Bu_k + Ky_k \tag{1}$$

$$y_k = Cx_k + Fd_k + e_k. \tag{2}$$

As typically considered in the subspace identification literature [18], $x_k \in \mathbb{R}^n$ is a nonphysical state vector, the dimension n of which is unknown. The state may or may not correspond directly to the flapwise deflection of the blade. The input $u_k \in \mathbb{R}$ is the control input to the system, which is the flap voltage, while the output of the system is $y_k \in \mathbb{R}$, the flapwise blade load measured by the strain sensor located at the root (in volts). Although coupled in principle, the two blades are considered to have no influence on each other, and the system from u_k to y_k is treated as an SISO system. The state-space matrices A, B, C, and K have the appropriate dimensions. The disturbance consists of two parts: the periodic signal $d_k \in \mathbb{R}^4$, overlaid with white noise e_k. The periodic signal arises out of rotational effects on the turbine, such as rotor imbalance, wind shear, tower shadow, and the sampling of turbulent structures. The dominant frequency content of this signal occurs at frequencies 1P (rotor speed) and 2P (two times rotor speed). As such, the signal d_k can be replaced by

$$d_k = \begin{bmatrix} \sin\psi_k \\ \cos\psi_k \\ \sin 2\psi_k \\ \cos 2\psi_k \end{bmatrix} \tag{3}$$

where ψ_k is the rotor azimuth. The signal d_k is shaped by the unknown matrix F to obtain the correct phase and magnitude of the periodic disturbance.

The effect of the state on the output can be expressed in terms of the past input–output data using an ARX approximation, as is commonly done in PBSID algorithms

$$y_k \approx C\mathcal{K}^{(s)} z_k + e_k \tag{4}$$

where the data vector $z_k \in \mathbb{R}^{2s+4}$ is given, for a past window size of s, as

$$z_k = \begin{bmatrix} u_{k-s} \\ y_{k-s} \\ u_{k-s+1} \\ y_{k-s+1} \\ \vdots \\ u_{k-1} \\ y_{k-1} \\ d_k \end{bmatrix}. \tag{5}$$

In the predictor equation (4), the unknown quantity $C\mathcal{K}^{(s)}$ is defined as Ξ, the Markov parameters of the system, which in terms of the state-space matrices, can be described as

$$\Xi = [\Xi_1 \quad \Xi_2 \quad \cdots \quad \Xi_{2s} \quad \Xi_F] \tag{6}$$

$$= [CA^{s-1}B \quad CA^{s-1}K \quad \cdots \quad CB \quad CK \quad F]. \tag{7}$$

Thus, (4) becomes

$$y_k = \Xi z_k + e_k. \tag{8}$$

The unknown terms Ξ can be estimated in a least-squares sense. However, in order to ensure that high-frequency dynamics in the noise are not estimated, it is desirable that the order of the identified system is minimized. To exploit this low-order nature, the data is stacked in the following manner:

$$Y_{k,s,N} = \mathcal{O}_s X_{k,1,N} + T_{u,s} U_{k,s,N} + T_{y,s} Y_{k,s,N} + \bar{F} D_{k,s,N} + E_{k,s,N}. \tag{9}$$

In this expression, the data is arranged in Hankel matrices in the form

$$Y_{k,s,N} = \begin{bmatrix} y_k & y_{k+1} & \cdots & y_{k+N-s-1} \\ y_{k+1} & y_{k+2} & \cdots & y_{k+N-s} \\ \vdots & \vdots & \ddots & \vdots \\ y_{k+s-1} & y_{k+s-2} & \cdots & y_{k+N-2} \end{bmatrix} \tag{10}$$

with similar expressions for x_k, u_k, d_k, and e_k. The term $\mathcal{O}_s$ is the extended observability matrix, defined as

$$\mathcal{O}_s^T = [C^T \quad (CA)^T \quad \cdots \quad (CA^{s-1})^T]. \tag{11}$$

Since the window size s is typically chosen to be much larger than the order of the system n, the term $\mathcal{O}_s X_{k,1,N}$ is low rank. This property will be made use of in regularizing the least squares solution. In terms of the past input–output data, the state matrix can be estimated as

$$X_{k,1,N} = \mathcal{K}^{(s)}[z_k \quad z_{k+1} \quad \cdots \quad z_{k+N-1}] \tag{12}$$

$$X_{k,1,N} = \mathcal{K}^{(s)} Z_{k,1,N}. \tag{13}$$

The unknown Toeplitz matrix $T_{u,s}$ in (9) is defined as

$$T_{u,s} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 \\ CB & 0 & 0 & \cdots & 0 \\ CAB & CB & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{s-2}B & CA^{s-3}B & CA^{s-4}B & \cdots & 0 \end{bmatrix} \tag{14}$$

with a similar expression for $T_{y,s}$, replacing B by K. Furthermore, the matrix $\bar{F}$ is a block diagonal matrix with the unknown matrix F along its diagonal. The data equation (9) can herewith be restated purely in terms of the collected input-output data as

$$\begin{aligned} Y_{k,s,N} = {} & \mathcal{O}_s \mathcal{K}^{(s)} Z_{k,1,N} + T_{u,s} U_{k,s,N} \\ & + T_{y,s} Y_{k,s,N} + \bar{F} D_{k,s,N} + E_{k,s,N}. \end{aligned} \tag{15}$$

As mentioned previously, the term $\mathcal{O}_s \mathcal{K}^{(s)} Z_{k,1,N}$ has low rank. This low-rank structure can be exploited by adding a rank minimization term to the identification cost function. However, the rank function is nonconvex. So, in order to convexify the optimization function, the rank penalty is approximated by penalizing the nuclear norm of $\mathcal{O}_s \mathcal{K}^{(s)} Z_{k,1,N}$. The nuclear norm, defined as the sum of singular values of a matrix, has been used as a proxy for rank minimization [20] in system identification [21].

Thus, with the addition of the nuclear norm to regularize the estimation of the system Markov parameters, the identification cost criterion can be stated as

$$\min_{\Xi} \|\mathcal{O}_s \mathcal{K}^{(s)} Z_{k,1,N}\|_* + \frac{\lambda}{N} \sum_{i=1}^{N} \|y_i - \Xi z_i\|_2^2. \tag{16}$$

It can be observed that the first term $\mathcal{O}_s \mathcal{K}^{(s)} Z_{k,1,N}$ depends on Ξ, and it is not an independent variable to be minimized. It can be restated as a function of Ξ in the following manner:

$$\begin{aligned} & \mathcal{O}_s \mathcal{K}^{(s)} Z_{k,1,N} \\ & = \begin{bmatrix} CA^{s-1}B & CA^{s-1}K & \cdots & CB & CK \\ 0 & 0 & \cdots & CAB & CAK \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & CA^{s-1}B & CA^{s-1}K \end{bmatrix} \\ & \quad \times Z_{k,1,N} \\ & = \begin{bmatrix} \Xi_1 & \Xi_2 & \cdots & \Xi_{2s-1} & \Xi_{2s} \\ 0 & 0 & \cdots & \Xi_{2s-3} & \Xi_{2s-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & \Xi_1 & \Xi_2 \end{bmatrix} Z_{k,1,N} = \mathcal{A}(\Xi). \end{aligned}$$

The cost criterion to be minimized for system identification can hence be written as

$$\min_{\Xi} \|\mathcal{A}(\Xi)\|_* + \frac{\lambda}{N} \sum_{i=1}^{N} \|y_i - \Xi z_i\|_2^2. \tag{17}$$

Once the optimal system Markov parameters Ξ^* have been obtained by minimizing this cost function, the standard $\text{PBSID}_{\text{opt}}$ procedure can be followed to arrive at a state-space realization of the underlying system. In order to facilitate rapid online system identification, the above-mentioned methodology has to be reformulated such that identification can be performed recursively and online.

B. Recursive PBSID With Nuclear Norm

The procedure for performing closed-loop identification with $\text{PBSID}_{\text{opt}}$ in a recursive and online manner has been delineated in [10]. This reference uses a 2-norm minimization criterion for identifying the system Markov parameters, which can be done efficiently using a recursive least-squares (RLSs) approach. Since the minimization criterion defined in Section II-A includes a penalty on the nuclear norm, direct RLS cannot be used. However, the nuclear norm is convex and differentiable in variables, so as long as the condition on the persistency of excitation holds, it is still possible to obtain a unique solution in an efficient manner. Due to the mixture of norms used, an ADMM [23] is used here to solve the minimization problem.

Initially, to separate the norms in the cost criterion, (17) is rewritten as

$$\min_{\mathcal{X},\Xi} \ \|\mathcal{X}\|_* + \text{trace}((\Xi - \Xi^{(0)})\mathcal{C}(\Xi - \Xi^{(0)})^T) \tag{18}$$

$$\text{s.t. } \mathcal{X} = \mathcal{A}(\Xi). \tag{19}$$

Here, the term $\Xi^{(0)}$ is the RLS solution to the 2-norm minimization of $\sum_{i=1}^{N} \|y_i - \Xi z_i\|_2^2$. In a batchwise sense, its value is given by

$$\Xi^{(0)} = Y_{k,1,N} Z_{k,1,N}^{\dagger} \tag{20}$$

where the $\dagger$ represents the Moore–Penrose pseudoinverse. Furthermore, the weighting matrix in the cost criterion (18) can be worked out to be $\mathcal{C} = \lambda Z_{k,1,N} Z_{k,1,N}^T$. To solve (18) using the ADMM, the saddle point of its Langrangean is sought, i.e., the new optimization criterion is given by

$$\max_{\mathcal{Z}} \min_{\mathcal{X},\Xi} \|\mathcal{X}\|_* + \text{trace}((\Xi - \Xi^{(0)})\mathcal{C}(\Xi - \Xi^{(0)})^T) \tag{21}$$

$$+ \mathcal{Z}^T(\mathcal{X} - \mathcal{A}(\Xi)) + \frac{t}{2}\|\mathcal{X} - \mathcal{A}(\Xi)\|_F^2. \tag{22}$$

Each of the decision variables can be iteratively optimized to converge to the required optimal value of the Markov parameters, Ξ^*. Here, t is the constant penalty parameter that determines the tradeoff between the primal and dual residuals; and can be tuned for achieving the desired rate of convergence. The ADMM procedure is given in the following.

1) Initialize the variables Ξ, $\mathcal{X}$, and $\mathcal{Z}$ to zero. Alternatively, assuming that the system dynamics do not change drastically, these variables can be initialized to the values from the previous time instant to facilitate a warm start.

2) Update the Markov parameters by setting the gradient with respect to Ξ to zero

$$\Xi = (\mathcal{C} + tM)^{-1}\mathcal{A}_{\text{adj}}(t\mathcal{X} - \mathcal{Z} + \mathcal{C}\Xi^{(0)}). \quad (23)$$

Here, $\mathcal{A}_{\text{adj}}(.)$ is the adjoint of $\mathcal{A}(.)$ and $\mathcal{A}_{\text{adj}}(\mathcal{A}(\Xi)) = M\Xi$.

3) Next, the terms involving $\mathcal{X}$ are minimized by SVT

$$U\text{diag}(\sigma)V^T = \mathcal{A}(\Xi) + \mathcal{Z}/t \quad (24)$$

$$\mathcal{X} = U\text{diag}(\max(0, \sigma - 1/t)V^T. \quad (25)$$

This step is defined using the SVT operator

$$\mathcal{X} = \mathbb{S}(\mathcal{A}(\Xi) + \mathcal{Z}/t). \quad (26)$$

4) The Lagrange variable can be updated linearly

$$\mathcal{Z} = \mathcal{Z} + t(\mathcal{A}(\Xi) - \mathcal{X}). \quad (27)$$

5) The procedure is repeated until the primal and dual residuals are acceptable.

For the given definition of $\mathcal{A}(.)$, its adjoint $\mathcal{A}_{\text{adj}}(.)$ and the matrix M depend on the input–output data and can be synthesized as follows:

$$\mathcal{A}_{\text{adj}}(\mathcal{X}) = \begin{bmatrix} \sum_{j=1}^{s} \bar{x}_j \bar{z}_{k,j}^T \\ \sum_{j=1}^{s-1} \bar{x}_j \bar{z}_{k,j+1}^T \\ \vdots \\ \bar{x}_1 \bar{z}_{k,s}^T \end{bmatrix} \quad (28)$$

$$M = [m_{ij}] \quad (29)$$

$$m_{ij} = \sum_{p=i}^{2s-i+j} \xi_{p(p+i-j}, \quad i \geq j \quad (30)$$

$$m_{ji} = m_{ij}, \quad i < j. \quad (31)$$

Here, the matrices $\mathcal{X}$ and $Z_{k,1,N}$ are partitioned into block row vectors as

$$\mathcal{X}^T = \begin{bmatrix} \bar{x}_1^T & \cdots & \bar{x}_s^T \end{bmatrix} \quad (32)$$

$$Z_{k,1,N}^T = \begin{bmatrix} \bar{z}_{k,1}^T & \cdots & \bar{z}_{k,N}^T \end{bmatrix}. \quad (33)$$

Furthermore, the matrix $Z_{k,1,N}Z_{k,1,N}^T$ is partitioned into 2×2 blocks such that each block is $\xi_{ij} \in \mathbb{R}^{2\times 2}$

$$[\xi_{ij}] = Z_{k,1,N}Z_{k,1,N}^T. \quad (34)$$

Hereby, with the ADMM, it is possible to recursively identify the nuclear norm-regularized Markov parameters of the system. The subsequent steps of the recursive $\text{PBSID}_{\text{opt}}$ procedure as in [10] can be followed in order to obtain online the required state-space realization of the system.

It has been assumed here that the available data from the measurement system is ideal in the sense that there is no missing input–output data. Furthermore, the sensor failures have not been considered. The robustness of the identification algorithm can be further improved by optimizing an $\mathcal{H}_\infty$ criterion as in [27] in order to ensure that the model approximation is robust to these forms of uncertainties.

It should be noted that this ADMM procedure for recursive identification requires a SVT step, which is computationally expensive and may not be tractable for flutter detection in real time. This issue is addressed by replacing the SVD step by a fast randomized variant.

C. Fast Singular Value Thresholding

1) Randomized Singular Value Thresholding : Since the bottleneck in the online implementation of the ADMM is the SVT step, an attempt is made to reduce the computational complexity through a randomized variant [25]. The procedure described in the previous step requires the SVT of a large matrix $(\mathcal{A}(\Xi) + \mathcal{Z}/t) \in \mathbb{R}^{s\times(N-s)}$. To ease the computational requirements, use can be made of the fact that this matrix is low rank. The matrix can be sampled by a random Gaussian matrix $\Omega \in (\mathbb{N} - \sim) \times \ell$ to obtain a smaller dimensional matrix

$$\mathcal{Y} = (\mathcal{A}(\Xi) + \mathcal{Z}/t)\Omega, \quad Y \in \mathbb{R}^{s\times\ell}. \quad (35)$$

As per [28], as long as the rank of the matrix n is smaller than the chosen column size ℓ of the Gaussian matrix, the range of the matrix Y will coincide with that of $(\mathcal{A}(\Xi) + \mathcal{Z}/t)$ and the following holds:

$$\mathbb{S}(\mathcal{Y}) \approx \mathbb{S}(\mathcal{A}(\Xi) + \mathcal{Z}/t). \quad (36)$$

The approximation error decays exponentially as ℓ grows, and is practically insignificant for an $\ell - n$ value of more than 5. Furthermore, if the underlying system is LTI, the SVT solution ergodically approaches the true value. Herewith, the SVT of a large matrix is replaced by the SVT of a much smaller matrix, and the computational time reduces proportionately [25].

2) Recursive Randomized SVT: Further reduction in computational complexity can be achieved by using range propagation [25]: i.e., considering that the range of $\mathcal{Y}$ does not change significantly over iterations. The randomly sampled matrix $\mathcal{Y}$ can be decomposed in the following manner:

$$\mathcal{Y} = Q\mathcal{B} \quad (37)$$

where Q has orthonormal columns. Based on the knowledge of $\mathcal{Y}$ and Q, $\mathcal{B}$ can be determined as $\mathcal{B} = Q^T\mathcal{Y}$. In this case, the SVT operation becomes

$$\mathbb{S}(\mathcal{A}(\Xi) + \mathcal{Z}/t) = \mathbb{S}(\mathcal{Y}) = Q\mathbb{S}(\mathcal{B}). \quad (38)$$

Knowledge of the term $Q^{(i-1)}$ from the previous iteration can be used to synthesize $Q^{(i)}$ at the current iteration, assuming that the range of the matrix $\mathcal{Y}$ is almost the same. The new vectors Y_p are sampled such that

$$Y_p = (\mathcal{A}(\Xi) + \mathcal{Z}/t)\Omega_p, \quad \Omega \in \mathbb{R}^{s\times p} \quad (39)$$

where p is a small number that depends on the extent to which the range space is expected to evolve. The term $Q^{(i)}$ can be obtained by updating its previous value in the following manner [25]:

$$\tilde{Q}^{(i)} = \text{PartialOrthogonalisation}([Q^{(i)} \quad Y_p]) \quad (40)$$

$$Q^{(i)} = \text{Normalisation}(\tilde{Q}^{(i)}). \quad (41)$$

The term $\mathcal{B}$ can be updated based on the new value of Q, and the SVT operation can be carried out as before.

For the original ADMM approach to recursive nuclear norm-based PBSID, performing SVD on the data matrix is the most computationally expensive step, and it is required to be performed at each iteration. Since the computational complexity scales linearly with the column size $N - s$ of this matrix, postmultiplication by a randomized matrix of column size ℓ reduces computation time by a factor of $(\ell/N - s)$. The second computational complexity saving is achieved by range propagation, which reduces a full SVD operation to a reduced QR-update operation.

Hereby, using the ADMM with fast randomized SVT, it is possible to obtain the system Markov parameters online in a recursive manner. These identified parameters can then be manipulated to obtain an estimate of the state-space model of the system.

D. Recursive PBSID With Nuclear Norm: Algorithm

In order to obtain estimates of the state-space matrices based on the estimated Markov parameters, two further RLS problems are required to be solved. For each of the RLS problems, a forgetting factor $0 < f \leq 1$ and a covariance estimate P of the appropriate dimensions is required to be initialized. The full algorithm from input–output data to model estimation is given in Algorithm 1.

With this algorithm, it is possible to obtain an estimate of the A, B, C, and K matrices at every instant of time. For rapid variations in system dynamics, it is desirable to have a low value of forgetting factor $f^{(1)}$ so that the changes in the Markov parameters can be tracked rapidly. However, reducing the value of $f^{(1)}$ makes the RLS solution more sensitive to noise. As has been shown in [21] and [22], the augmented cost function including the nuclear norm reduces the sensitivity of the Markov parameter estimate to the effect of noise and allows for the lowering of the forgetting factor. Hereby, the identification process can be made more sensitive to changes in system dynamics without affecting the variance of the estimates.

Experimental data-driven identification, using the methodology developed in this section, is demonstrated in Section IV.

Algorithm 1 Recursive Identification With the Nuclear Norm

input: s, n, $\begin{bmatrix} A_{-1} & B_{-1} & K_{-1} \end{bmatrix}$, $f^{(1),(2),(3)}$, $P_{-1}^{(1),(2),(3)}$

for $k = 0, 1, 2 \ldots$

input: u_k, y_k

Step 1: Update $\Xi_k^{(0)}$

$P_k^{(1)} = \frac{1}{f^{(1)}} P_{k-1}^{(1)} - \frac{1}{f^{(1)}} P_{k-1}^{(1)} z_k (f^{(1)} + z_k^T P_{k-1}^{(1)} z_k)^{-1} z_k^T P_{k-1}^{(1)}$,
$\Xi_k^{(0)} = \Xi_{k-1}^{(0)} + (y_k - \Xi_{k-1}^{(0)} z_k) z_k^T P_k^{(1)}$.

Step 2: Update Ξ_k
From $\Xi_k^{(0)}$, obtain Ξ_k using ADMM as described in Section III-C. The ADMM variables Ξ, $\mathcal{X}$ and $\mathcal{Z}$ are initialised from the values at the previous iteration. The SVT step in ADMM is carried out using the randomised variant, with range propagation, as described in Section III-D.

Step 3: Estimate x_k
From Ξ_k, construct $\mathcal{O}_s \mathcal{K}^{(s)}$ and $T_{y,s}$.
With selection matrix S described in [10] Section IV-B,
$x_k = S(I_{\ell s} - T_{y,s})^{-1} \mathcal{O}_s \mathcal{K}^{(s)} z_k$.

Step 4: Estimate the C matrix at time k, Equation 4

$P_k^{(2)} = \frac{1}{f^{(2)}} P_{k-1}^{(2)} x_k (f^{(2)} + x_k^T P_{k-1}^{(2)} x_k)^{-1} x_k^T P_{k-1}^{(2)}$,
$C_k = C_{k-1} + (y_{k-1} - C_{k-1} x_k) x_k^T P_k^{(2)}$.

Step 5: Estimate e_k
$e_k = y_k - C_k x_k$.

Step 6: Estimate the A, B and K matrices at time k,

$\phi_k = \begin{bmatrix} x_{k-1} \\ u_k \\ e_k \end{bmatrix}$, $\Theta_k = \begin{bmatrix} A_k & B_k & K_k \end{bmatrix}$,
$P_k^{(3)} = \frac{1}{f^{(3)}} P_{k-1}^{(3)} \phi_k (f^{(3)} + \phi_k^T P_{k-1}^{(3)} \phi_k)^{-1} \phi_k^T P_{k-1}^{(3)}$,
$\Theta_k = \Theta_{k-1} + (x_k - \Theta_{k-1} \phi_k) \phi_k^T P_k^{(3)}$.

end for

IV. RESULTS

Experiments are conducted on the setup described in Section II. The two-bladed upwind turbine is operated at a number of different wind speeds in the wind tunnel, and trailing edge flap actuation is used to investigate the effect on the blade loads. The pitch motors are used to keep the blade pitch fixed at fine, while the rotor speed is controlled by constant load generator operation. Constant load implies that the rotor speed varies linearly with the ambient wind speed; rated operation at 230 rpm occurs at a wind speed of 4.5 m/s. The turbine blades display flutter at a wind speed of 6 m/s, which corresponds to a rotor speed of 315 rpm.

As mentioned in Section I, this form of flutter is distinct from ordinary wind turbine flutter, which typically involves an aeroelastic coupling of the torsional and bending modes of the blade, and typically manifests itself in fault cases such as high wind/rotor speeds in the presence of a grid fault. The form of flutter observed in the wind tunnel is caused by a coupling of the rigid-body flap motion with the first flexible bending mode of the blade; this also occurs under overspeed conditions. Flutter was physically observed in the wind tunnel by a limit-cycling of flap motion, constrained by the flap angle limiters, and an unbounded increase in the loads measured by the strain gauges, at a frequency equal to the calculated blade bending frequency. Open-loop experiments at such high speeds had to be terminated rapidly to avoid structural damage to the system, but it was possible herewith to gain approximate knowledge of the flutter speed, which occurred at a rotor speed of 315 rpm. The rotor and generator speed are measured at all time instants using a speed sensor located in the wind turbine nacelle.

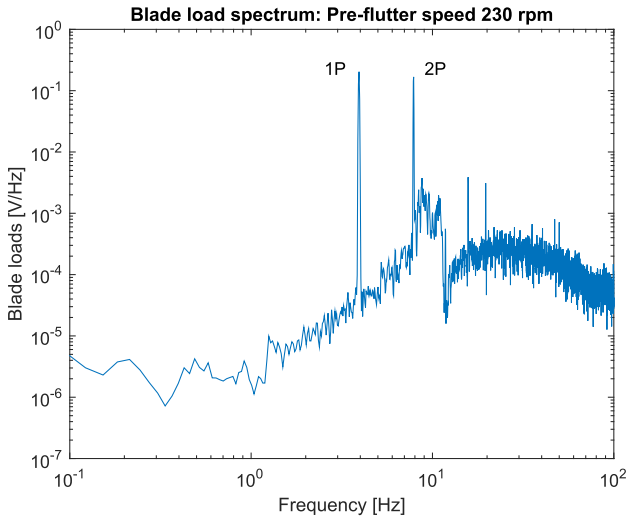

Fig. 5. Blade load spectrum at the preflutter speed of 230 rpm.

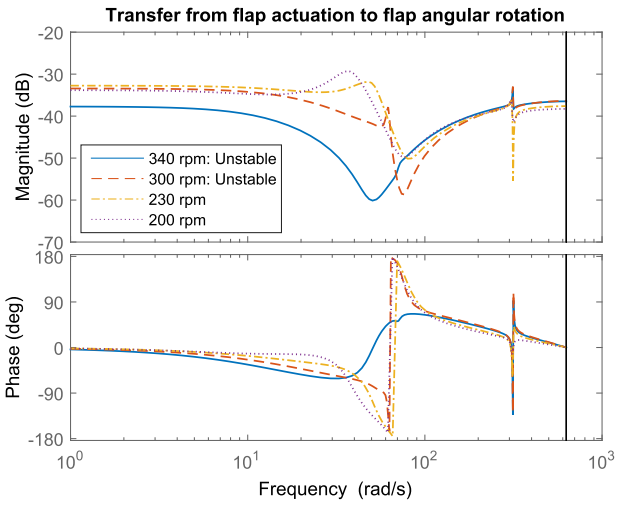

Fig. 6. Transfer from flap actuators to flap angle for different wind speeds.

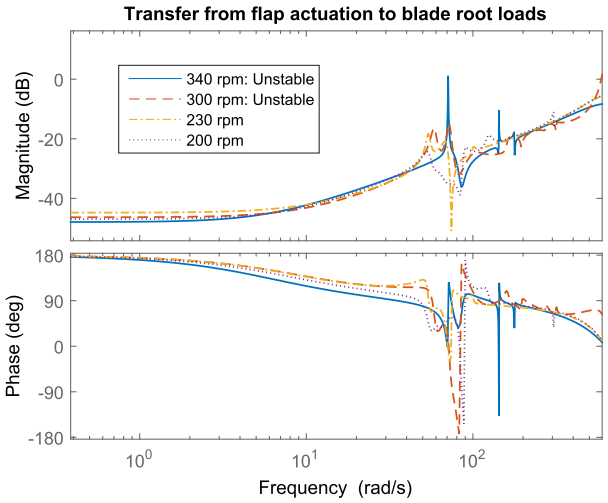

Fig. 7. Transfer from flap actuators to blade root loads for different wind speeds.

The trailing edge flap actuators are designed to reduce the blade loads (in the stable and unstable operating range), as measured by the blade root load sensors. The dominant frequencies in the blade load spectrum [see Fig. 5] are the 1P and 2P frequencies, which correspond to the rotor speed and its first harmonic. These loads are caused by exogenous forcing agents, such as rotor imbalance and tower shadow. These frequencies are accounted for by considering a sinusoidal periodic signal d_k in the recursive algorithms described in Section III, where

$$d_k = \begin{bmatrix} \sin \psi_k \\ \cos \psi_k \\ \sin\ 2\psi_k \\ \cos 2\psi_k \end{bmatrix} \tag{42}$$

with ψ_k the rotor azimuth measured by the rotor shaft angle encoder in the nacelle. Since the azimuth is directly measured, the signal d_k is always phase locked with the load peaks 1P and 2P, irrespective of the wind speed. So, the same signal can be used for operating conditions where the wind speed, and hence the rotor speed, vary with time.

The objective of the recursive identification is to estimate the aeroelastic plant between the actuators and the blade load sensor. Blade angle encoders form an auxiliary collocated sensing element. A stabilizing flap controller is used in a closed loop that acts on feedback received from the collocated sensors and issues flap deflection commands. The design of the controller is described in [7]. Thus, for identification of the underlying system from flap actuation to blade loads, it is necessary to operate in a closed loop and this motivates the use of a closed-loop identification scheme like RPBSID.

Reference [7] also describes the batch identification performed on the data collected from the wind turbine, operating at different fixed wind speeds (and hence different fixed rotor speeds). The Bode diagrams of the identified systems, from flap actuation to flap angle deflection and from flap actuation to blade load measurement, can be seen in Figs. 6 and 7, respectively. It can be seen that the system varies strongly with wind speed (rotor speed). There are two dominant low-frequency modes that change the system behavior with wind speed. The flap angle mode, seen in Fig. 6, has a frequency that rises linearly with wind speed until it reaches a value of 15 Hz at flutter. This mode becomes progressively more damped at high wind speeds. On the other hand, the blade flexible mode, as measured by the blade root sensors in Fig. 7, shows a constant frequency of 15 Hz with a damping that progressively becomes smaller until the mode becomes unstable beyond 290 rpm (5.8-m/s wind speed). These results corroborate well with the numerical results described in [7].

It should be noted that [7] describes the use of two controllers: a classically designed feedback controller intended to stabilize the plant in the closed loop and a feedforward controller that targets load alleviation. In the experiments reported in this paper, only the stabilizing feedback controller is connected in a closed loop, and load reduction is not targetted, since our present aim at this point is online system identification. A persistently exciting random signal is

superposed on the output of the feedback controller; however, this does not affect the stability of the closed loop. There is an increase in blade loads as a result of this random signal. However, due to low turbulence, a sufficiently high signal-to-noise ratio is achieved with a low covariance random signal, so that the load increase is slight.

The persistency of excitation has very little effect on actuator usage. On the other hand, the actuator usage needed for stabilization increases exponentially with an increase in ambient wind speed.

Recursive identification is done of this system, based on the theory of Section III, under constant and variable operating conditions.

A. Constant Operating Conditions

First, the turbine is operated at a preflutter wind speed of 4.5 m/s, which corresponds to a rotor speed of 230 rpm. The stabilizing load controller is switched OFF, and white noise low-pass filtered through 50 Hz is superposed on the flap actuator to provide persistency of excitation. Thus, the plant from flap actuation to blade load sensors is operating in open loop. The recursive algorithms are cold-started, in other words, the ADMM and RLS parameters are initialized to 0 at time 0. For rapid convergence to the steady-state values, a relatively low forgetting factor of 0.997 is used in the estimation of the system Markov parameters. Furthermore, the state vector is considered to have length 4 (on the basis of the modal analysis [7]), and the past window for state estimation is taken to have length 20 samples. Although the sampling frequency of the real-time system is 2000 Hz, identification is done at a lower frequency of 500 Hz, in order to focus on low-frequency system dynamics.

The recursive algorithms described in Section III are evaluated, and the results are shown in Fig. 8. The damping of the mode corresponding to the highest system singular values has been plotted. The damping of the modes corresponding to the other singular values is significantly noisy, and gives no additional information. The plots thereof have hence been omitted. It can be seen that the damping of the poles estimated by the recursive algorithm reaches the value obtained from standard batchwise PBSID, within a period of 40 s. The nuclear norm-enhanced algorithm shows faster convergence and lowered variance as compared with the standard recursive algorithm.

Next, the turbine is operated at the postflutter wind speed of 6.5 m/s, which corresponds to a rotor speed of 340 rpm. At this wind speed, the turbine cannot be operated in open loop. The stabilizing controller is switched ON; it commands the flap actuators based on collocated sensor feedback to ensure operational stability. The same filtered white noise that was used earlier for flap actuation is superposed on the flap control action, in order to provide sufficient persistency of excitation. As before, the algorithms are cold-started, and a low forgetting factor is used. The comparison between the recursive and the batchwise algorithms can be seen in Fig. 9. The recursive algorithms identify the system correctly to be unstable, with a negative value of damping. Once again,

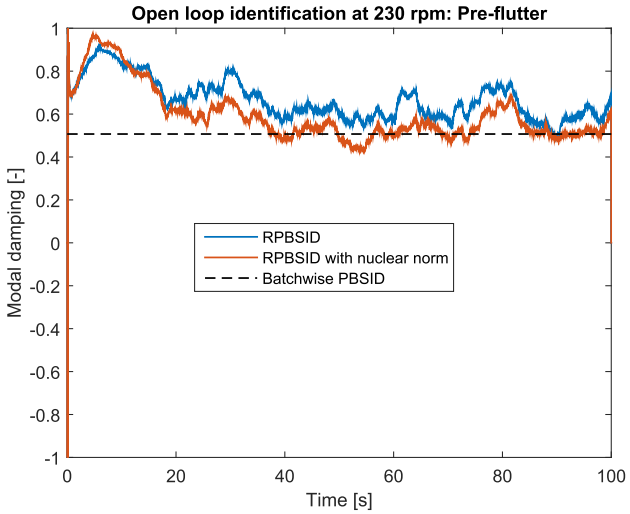

Fig. 8. Open-loop tracking of damping of the mode corresponding to the highest singular values. Preflutter speed and comparison of recursive and batch algorithm.

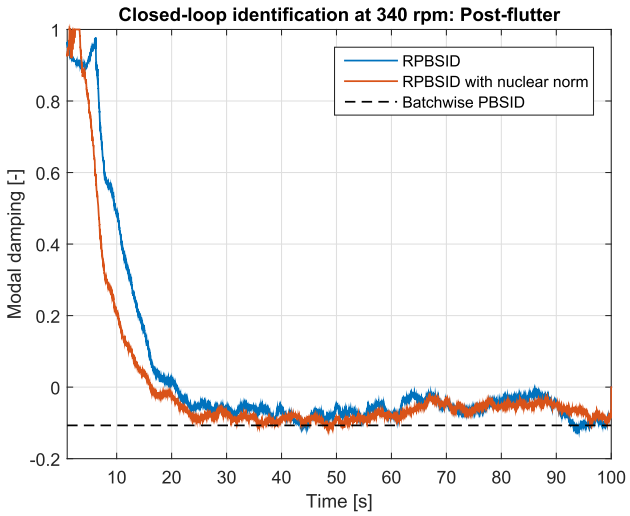

Fig. 9. Closed-loop tracking of damping of the mode corresponding to the highest singular values. Postflutter speed and comparison of recursive and batch algorithm.

the estimated modal damping converges to its steady-state value within 40 s. The presence of the nuclear norm speeds up the convergence to the steady-state value. Furthermore, as seen before, the variance of the damping estimate is lower due to the presence of the nuclear norm.

The effect of using the nuclear norm during the identification can be seen in Fig. 10, where the Markov parameters with and without nuclear norm regularization are shown. It can be seen that, with the nuclear norm, the low-order Markov parameters CB and CK, are virtually unchanged, while the higher order ones are attenuated. Since the higher-order Markov parameters usually correspond to noise or high-frequency dynamics, it can be appreciated that the use of the nuclear norm in recursive identification makes the parameter estimation less sensitive to noise and more responsive to fundamental changes to low-frequency system behavior.

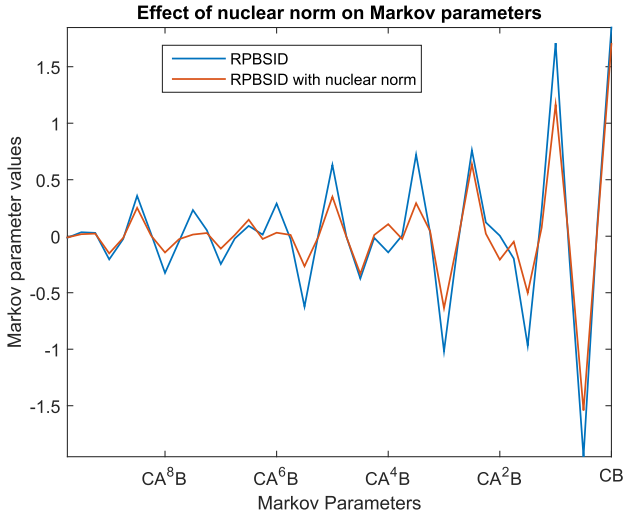

Fig. 10. Effect of using the nuclear norm on identified Markov parameters.

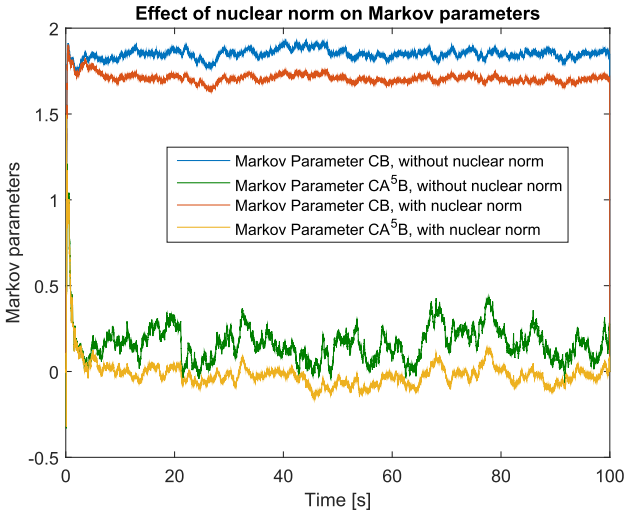

Fig. 11. Effect of using the nuclear norm on identified Markov parameters.

In Fig. 11, the variation of the Markov parameters over time can be observed. The Markov parameter CB relates to the highest singular values found in the data, while the Markov parameter CA^5B corresponds to lower singular values, which are affected to a larger extent by noise in the data. It can be seen that, while there is a small but clear offset between the Markov parameters CB when the nuclear norm is used, they show similar behavior and are relatively unaffected by process noise. On the other hand, when the nuclear norm is not used, the Markov parameter CA^5B is extremely sensitive to noise and varies substantially over time. The utility of the nuclear norm lies in driving down the value of the higher-order Markov parameters to an average of zero, and significantly reducing the variance in their estimation.

B. Varying Operating Conditions

The objective of the recursive identification algorithm is to be able to track the variation in dynamics as the wind turbine is subject to variations in operating conditions; specifically,

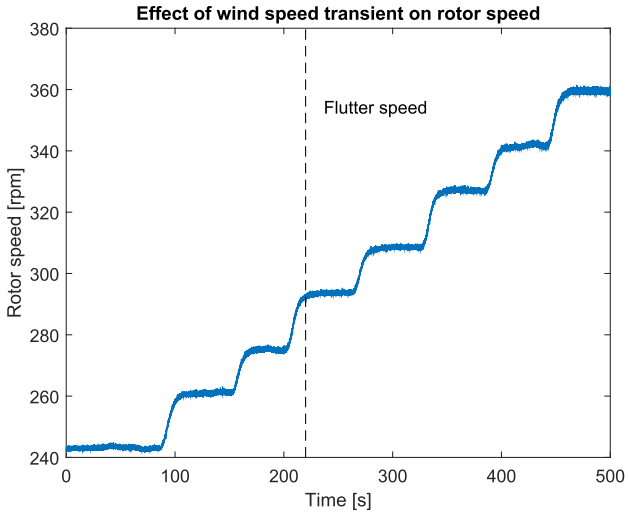

Fig. 12. Changes in the rotor speed as a consequence of varying the wind speed in a stepwise manner in the wind tunnel.

the changes in dynamics occurring due to wind speed variations. In order to validate the response of the recursive algorithm developed previously to such varying wind conditions, the wind speed in the tunnel is increased in a stepwise manner from the preflutter wind speed of 4.5 m/s to a postflutter wind speed of 7 m/s. Correspondingly, the rotor speed changes from 240 to 360 rpm, as shown in Fig. 12. The rotor enters into flutter at a rotor speed of 290 rpm, as indicated by the broken vertical line, at a time instant of 220 s.

Since the wind turbine is expected to operate in the preflutter as well as the postflutter regime, the stabilizing controller is operational throughout the experiment. As such, the recursive algorithm uses closed-loop data for system identification. As in the previous experiments, the control signal is perturbed by a noise signal of low amplitude to ensure adequate persistency of excitation. Furthermore, at time $t = 0$, the ADMM and the RLS parameters are initialized to zero. However, in the course of the experiment, as the wind evolves, these parameters are not reinitialized and in effect, the RPBSID algorithms are warm-started for different wind speeds over the course of the experiment.

The results of the recursive identification algorithm can be seen in Fig. 13. It can be seen that, for the same operating wind speeds, the algorithm converges temporarily to the same steady-state values of the modal damping at the fixed operating conditions found in Section IV-A. Thus, the system is in principle not just time-varying, but parameter-varying, and its dynamics can be uniquely determined by the current operating wind speed.

It should be noted here that the varying operating conditions, in this section, refer to a stepwise increase in wind speed generated in the wind tunnel. Such a wind field realization is unlikely to occur in practice; however, the tests give insight into the behavior of the different identification algorithms. Due to practical considerations, however, a more realistic wind field could not be generated consistently in the wind tunnel. For a true turbulent wind field realization, the wind speed will vary arbitrarily: it is expected that the algorithms will,

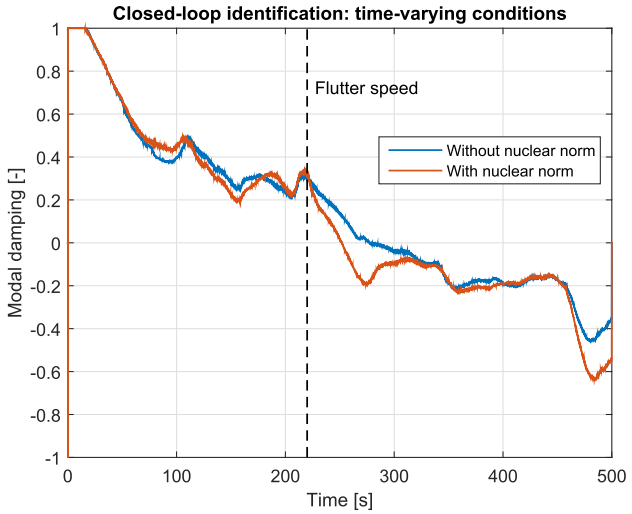

Fig. 13. Closed-loop tracking of damping of the mode corresponding to the highest singular values. Stepwise increase in ambient wind speed.

with some lag, be able to follow the trend in the changing system dynamics with a relatively good degree of fidelity; while ignoring high-frequency spikes in wind speed variation that contribute little or nothing to wind turbine dynamic behavior. Furthermore, it is expected that, as in the wind tunnel, the nuclear norm-enhanced version of the algorithm [recursive PBSID with nuclear norm (RPBSID-NN)] will be more responsive: it converges faster and with lower variance to the correct value of modal damping. In the experiments, the RPBSID-NN has been able to detect the onset of flutter within 32.4 s, much sooner than the standard algorithm, which takes 60.8 s to report that the system has become unstable.

V. Conclusion

The experimental setup that has been designed for the advanced load control of wind turbines using trailing-edge free-floating flaps shows dynamics that vary strongly with wind speed, with the system becoming open-loop unstable at wind speeds beyond the flutter point. In order to track the operating point of the wind turbine, and to adjust the controller accordingly, it may become necessary to perform online identification. Since instability may occur during operation, it is necessary, from the point of view of safety, to operate the turbine at all times in closed loop with a stabilizing controller. As such, a closed-loop online identification algorithm was devised in order to detect changes in system dynamics.

The recursive version of the closed-loop $\text{PBSID}_{\text{opt}}$ algorithm (RPBSID) was used to perform system identification of the experimental setup. Since the setup is subject to exogenous periodic disturbances at the 1P and 2P frequencies, the recursive algorithm was extended to include the estimation of the effect of the external forcing on the system output.

Since the plant dynamics can also change rapidly with a change in the ambient wind conditions, it was found desirable to keep the forgetting factor as low as possible. In the batchwise sense, this is the same as reducing the size of the dataset; it produces a bias in the estimation of the parameters. Furthermore, due to the increased sensitivity to noise, the variance of the estimates also increases.

In order to overcome these drawbacks, the RPBSID cost function was augmented with a nuclear norm term (RPBSID-NN). While the optimization problem to be solved for parameter estimation remains convex, an ADMM technique for iterating to the solution was developed. The ADMM involves an SVT step; this step was made computationally tractable by replacing the SVD of a large matrix by a smaller matrix obtained by randomly sampling the columns of the large matrix. Furthermore, use was made of the fact that the range of the characteristic system matrices does not change drastically across time instants, in order to speed up the ADMM process.

It was seen that the addition of the nuclear norm regularizes the identified Markov parameters by reducing the value and variance of the higher-order parameters. As these higher-order parameters are typically associated with noise and high frequency, RPBSID-NN is able to identify the system parameters with a lower bias and variance than the conventional RPBSID version. The algorithm is shown to work on both preflutter and postflutter data, while operating in both open and closed loops, with damping estimates close to those predicted by the batchwise version of PBSID. As such, the proposed algorithm is suitable for the application actuator-induced instability detection, and can be used for both fault detection as well as adaptive control.

In comparison with a neural network-based identification method [11], [12], where a direct least-squares method has been used, the method described in this paper has higher computational complexity, and is not suitable for highly nonlinear systems. However, the experiments show that for a plant such as the experimental wind turbine, with slowly varying dynamics, the method is able to track the system behavior with a relatively high degree of fidelity. Furthermore, the method yields a state-space system descriptor that can be readily used for online adaptive control.

For the case with a varying wind speed, where the system crosses the stability boundary and becomes unstable, the RPBSID-NN algorithm is shown to correctly track the system model damping throughout the duration of operation. The onset of flutter is detected almost twice as fast as with the conventional RPBSID algorithm. In conclusion, for a scaled wind turbine in realistic operating conditions, the developed algorithm is able to stably and accurately track system parameters, and identify the manifestation of system instability, such as flutter, within a short period of its occurrence.

Acknowledgment

The authors would like to thank C. J. Slinkman and W. J. M. van Geest for the design and construction of the experimental setup. They would also like to thank B. Telsang for her investigation of fast SVD methods suitable for online real-time implementation.

References

[1] E. A. Bossanyi, “Individual blade pitch control for load reduction,” *Wind Energy*, vol. 6, no. 2, pp. 119–128, 2003.

[2] T. K. Barlas and G. A. M. van Kuik, “Review of state of the art in smart rotor control research for wind turbines,” *Prog. Aerosp. Sci.*, vol. 46, no. 1, pp. 1–27, Jan. 2010.

[3] J. W. van Wingerden *et al.*, "Two-degree-of-freedom active vibration control of a prototyped 'smart' rotor," *IEEE Trans. Control Syst. Technol.*, vol. 19, no. 2, pp. 284–296, Mar. 2011.
[4] D. Castaignet *et al.*, "Full-scale test of trailing edge flaps on a Vestas V27 wind turbine: Active load reduction and system identification," *Wind Energy*, vol. 17, no. 4, pp. 549–564, 2013.
[5] L. O. Bernhammer, R. de Breuker, M. Karpel, and G. J. van der Veen, "Aeroelastic control using distributed floating flaps activated by piezoelectric tabs," *J. Aircraft*, vol. 50, no. 3, pp. 732–740, 2013.
[6] S. T. Navalkar, L. O. Bernhammer, J. Sodja, C. J. Slinkman, J. W. van Wingerden, and G. A. M. van Kuik, "Aeroelastic design and LPV modelling of an experimental wind turbine blade equipped with free-floating flaps," *J. Phys., Conf. Ser.*, vol. 753, no. 4, p. 042010, 2016.
[7] S. T. Navalkar, L. O. Bernhammer, J. Sodja, E. van Solingen, G. A. M. van Kuik, and J. W. van Wingerden, "Wind tunnel tests with combined pitch and free-floating flap control: Data-driven iterative feedforward controller tuning," *Wind Energy Sci.*, vol. 1, no. 2, pp. 205–220, 2016.
[8] M. H. Hansen, "Aeroelastic instability problems for wind turbines," *Wind Energy*, vol. 10, no. 6, pp. 551–577, 2007.
[9] B. S. Kallesøe and K. A. Kragh, "Field validation of the stability limit of a multi MW turbine," *J. Phys. Conf. Ser.*, vol. 753, no. 4, p. 042005, 2016.
[10] I. Houtzager, J. W. van Wingerden, and M. Verhaegen, "Recursive predictor-based subspace identification with application to the real-time closed-loop tracking of flutter," *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 4, pp. 934–949, Jul. 2012.
[11] X. Dong, Y. Zhao, H. R. Karimi, and P. Shi, "Adaptive variable structure fuzzy neural identification and control for a class of MIMO nonlinear system," *J. Franklin Inst.*, vol. 350, no. 5, pp. 1221–1247, Jun. 2013.
[12] L. Huang, K. Wang, P. Shi, and H. R. Karimi, "A novel identification method for generalized T-S fuzzy systems," *Math. Problems Eng.*, vol. 2012, 2012, Art. no. 893807.
[13] M. Jelavic, N. Perc, and I. Petrovic, "Identification of wind turbine model for controller design," in *Proc. 12th Int. Power Electron. Motion Control Conf.*, Aug. 2006, pp. 1608–1613.
[14] P. M. O. Gebraad, J. van Wingerden, P. A. Fleming, and A. D. Wright, "LPV Identification of wind turbine rotor vibrational dynamics using periodic disturbance basis functions," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 4, pp. 1183–1190, Jul. 2013.
[15] G. J. van der Veen, J.-W. van Wingerden, and M. Verhaegen, "Global identification of wind turbines using a Hammerstein identification method," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 4, pp. 441–454, Jul. 2013.
[16] R. Hallouzi, V. Verdult, R. Babuška, and M. Verhaegen, "Fault detection and identification of actuator faults using linear parameter varying models," *IFAC Proc.*, vol. 16, no. 1, pp. 119–124, 2005.
[17] S. T. Navalkar, E. van Solingen, and J.-W. van Wingerden, "Wind tunnel testing of subspace predictive repetitive control for variable pitch wind turbines," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 6, pp. 2101–2116, Nov. 2015.
[18] A. Chiuso, "The role of vector autoregressive modeling in predictor-based subspace identification," *Automatica*, vol. 43, no. 6, pp. 1034–1048, Jun. 2005.
[19] M. Lovera, "Recursive subspace identification based on projector tracking," in *Proc. IFAC Symp. Syst. Identificat.*, Rotterdam, The Netherlands, 2003.
[20] M. Fazel, H. Hindi, and S. P. Boyd, "A rank minimization heuristic with application to minimum order system approximation," in *Proc. Amer. Control Conf.*, Jun. 2001, pp. 4734–4739.
[21] M. Verhaegen and A. Hansson, "Nuclear norm subspace identification (N2SID) for short data batches," in *Proc. IFAC World Congr.*, 2014, pp. 9528–9533.
[22] S. T. Navalkar and J. W. van Wingerden, "Nuclear norm-enhanced recursive subspace identification: Closed-loop estimation of rapid variations in system dynamics," in *Proc. Amer. Control Conf.*, Jul. 2016, pp. 936–941.
[23] M. Annergren, A. Hansson, and B. Wahlberg, "An ADMM algorithm for solving ℓ_1 regularised MPC," in *Proc. Amer. Control Conf.*, Dec. 2012, pp. 4486–4491.
[24] J.-F. Cai, "Fast singular value thresholding without singular value decomposition," Univ. California, Los Angeles, Los Angeles, CA, USA, CAM Rep. CAM10-24, 2010.
[25] T.-H. Oh, Y. Matsushita, Y.-W. Tai, and I. S. Kweon, "Fast randomized singular value thresholding for nuclear norm minimization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, Jun. 2015, pp. 4484–4493.
[26] C. Bak *et al.*, "Description of the DTU 10 MW reference turbine," DTU Wind Energy, Roskilde, Denmark, Tech. Rep. I-0092, 2013.
[27] Y. Wei, J. Qiu, H. R. Karimi, and M. Wang, "Model approximation for two-dimensional Markovian jump systems with state-delays and imperfect mode information," *Multidimensional Syst. Signal Process.*, vol. 26, no. 3, pp. 575–597, Jul. 2015.
[28] N. Halko, P.-G. Martinsson, and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," *SIAM Rev.*, vol. 53, no. 2, pp. 217–288, 2011.

Sachin T. Navalkar was born in Mumbai, India, in 1986. He received the B.E. degree in mechanical engineering from the University of Mumbai, Mumbai, India, and the M.Sc. (*cum laude*) degree in sustainable energy technology from the Delft University of Technology, The Netherlands, where he is currently pursuing the Ph.D. degree within the Delft Center for Systems and Control, with a focus on integrated control design for wind turbines with advanced actuation and sensing.

Jan-Willem van Wingerden was born in Ridderkerk, The Netherlands, in 1980. He received the B.S. and Ph.D. (*cum laude*) degrees from the Delft Center for Systems and Control, Delft University of Technology, The Netherlands, in 2004 and 2008, respectively. His Ph.D. thesis was entitled Smart Dynamic Rotor Control for Large Offshore Wind Turbines.

He was with Philips Applied Technologies, Eindhoven, The Netherlands. He is currently an Associate Professor with the Delft University of Technology, Delft, Netherlands. His current research interests include linear parameter varying identification, subspace identification, smart structures and control, and identification of wind turbines and wind farms.