# Efficient tuning of Individual Pitch Control

## A Bayesian Optimization Machine Learning approach

Mulders, S. P.; Pamososuryo, A. K.; Van Wingerden, J. W.

**DOI**
10.1088/1742-6596/1618/2/022039

**Publication date**
2020

**Document Version**
Final published version

**Published in**
Journal of Physics: Conference Series

# Efficient tuning of Individual Pitch Control: A Bayesian Optimization Machine Learning approach

# Efficient tuning of Individual Pitch Control: A Bayesian Optimization Machine Learning approach

**SP Mulders, AK Pamososuryo and JW van Wingerden**

Delft Center for Systems and Control, Faculty of Mechanical Engineering, Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands

E-mail: J.W.vanWingerden@tudelft.nl

**Abstract.** With the trend of increasing wind turbine rotor diameters, the mitigation of blade fatigue loadings is of special interest to extend the turbine lifetime. Fatigue load reductions can be partly accomplished using individual pitch control (IPC), and is commonly facilitated by the so-called multiblade coordinate (MBC) transformation. This operation transforms and decouples the blade load signals in a non-rotating yaw-axis and tilt-axis. However, in practical scenarios, the resulting transformed system still shows coupling between the axes. To cope with this phenomenon, earlier research has shown that the introduction of an additional MBC tuning variable – the azimuth offset – decouples the multivariable system. However, the introduction of this extra variable complicates the controller design process, and requires expert knowledge and specialized analysis software. To provide an efficient method for the optimization of fixed-structure IPC controllers, based on black box and computationally costly objective functions, this paper considers a Bayesian optimization controller tuning framework. Results show the efficiency of the framework to tune a combined $1P + 2P$ IPC implementation, without prior knowledge, and based on high-fidelity simulation results using a computationally expensive objective function.

## 1. Introduction

Wind turbine rotors are getting ever larger, which supports the sustained demand of increased wind turbine power ratings. For a turbine with such a large rotor, the wind varies spatially and temporally over the rotor surface because of the combined effects of turbulence, wind shear, yaw-misalignment and tower shadow [1]. These effects give rise to periodic blade loads, and the individual blades mainly experience a once-per-revolution (1P) cyclic load.

As a result of the increasing sizes, the blades are getting more flexible, which poses a need for sophisticated fatigue load reducing control methods [2]. One such control method is individual pitch control (IPC). This technique is applicable to more recent wind turbines, with their ability to pitch the blades to distinct angles, and to measure the blade root bending moments. The control method is commonly used to reduce harmonic out-of-plane blade loads, which are predominantly present at multiples ($n$P) of the turbine rotational speed. The individual pitch contributions are generally formed using the azimuth-dependent multiblade coordinate (MBC) transformation, acting on out-of-plane blade root bending moment measurements [3]. The MBC transformation, transforms the blade loads from a rotating into a non-rotating frame, and decouples the load signals into fixed yaw and tilt axes. The transformation allows for convenient load analysis and IPC controller implementations.

IPC for wind turbine fatigue load reductions using the MBC transformation is widely discussed in the literature, and an overview is given in [4]. In research, the transformation is often combined with more advanced controller techniques, such as linear-quadratic-Gaussian (LQG) control [5], $H_\infty$ techniques [6], repetitive control (RC) [7], and model predictive control (MPC) using short-term wind field predictions [8]. Recent work has shown that the MBC transformed system shows multivariable coupling [9]. By decoupling the system with the so-called azimuth offset [10], the application of multiple single-input single-output (SISO) loops with traditional controller elements is justified. The offset is shown to be especially important for higher load harmonics, $n > 1$. The performance benefits of optimal azimuth offset incorporation in a combined $1\mathrm{P} + 2\mathrm{P}$ implementation are outlined in [11]. By configuring the azimuth offset correctly, perfect load attenuation and minimal actuator effort is attained, using a simple controller structure.

An IPC implementation based on the MBC transformation and traditional proportional-integral-derivative (PID) control techniques often has a fixed parameterized controller structure. With the introduction of the azimuth offset, an extra tuning variable is added to the parameter set. The work in [10] shows that determination of the optimal offset value requires expert control knowledge and analysis software. For this reason, an automated controller tuning framework, by minimization of an optimization objective, forms an interesting opportunity for efficient parameter tuning. To this end, often grid search approaches are employed. However, for higher dimensional parameter spaces, the amount of required iterations becomes computationally intractable. Therefore, this work considers Bayesian optimization (BO) for fixed-structure controller tuning. Bayesian optimization is an efficient algorithm for the optimization of black box functions or systems, of which the evaluation is (computationally) expensive [12].
The main contributions of this paper are:

- providing the results of a time-domain MBC framework including the azimuth offset, applied to a combined $1\mathrm{P} + 2\mathrm{P}$ IPC implementation;
- giving an overview and theoretical summary of the Gaussian process and Bayesian optimization;
- identifying the crucial controller parameters in the presented IPC implementation, for efficient tuning with Bayesian optimization;
- proposing a convenient and easily implementable framework, for the optimization of nonconvex, multimodal and/or computationally costly problems;
- concluding on the benefits of implementing a Bayesian optimization framework for fixed-structure controller tuning, based on the obtained results.

The organization of this paper is as follows. In Section 2, the combined $1\mathrm{P} + 2\mathrm{P}$ time-domain MBC-IPC implementation, with the option for azimuth offsets is discussed. The IPC implementation has a fixed structure and is tuned by a predefined set of parameters. In Section 3, relevant theory is given on the Gaussian process and Bayesian optimization. The former mentioned concept is used to model the unknown objective function in a data-driven manner, whereas the latter leverages the surrogate model to efficiently select the consecutive and promising sampling points. Section 4 explains the methodology and practical implementation of the Bayesian framework, minimizing the computationally expensive damage equivalent load (DEL) objective function. Section 5 presents the optimization results for a range of operating conditions (wind speeds). Finally, Section 6 concludes on the effectiveness of the proposed framework for fixed-structure controller tuning, using realistic, computationally expensive, and high-fidelity simulation results.

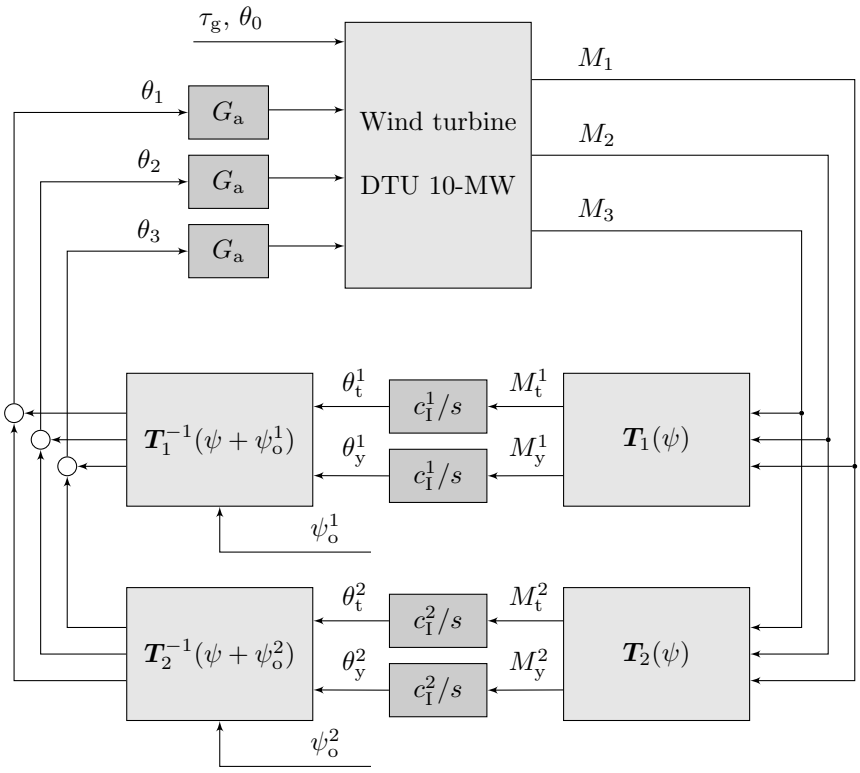



Figure 1: Implementation of IPC using the azimuth-dependent forward and reverse transformations, $\boldsymbol{T}_n(\psi)$ and $\boldsymbol{T}_n^{-1}(\psi+\psi_o^n)$, respectively. The transformations are used to form two feedback loops at the 1P and 2P rotational frequencies, transforming the blade load harmonics to a fixed reference frame. For both 1P and 2P loops in the fixed frame, the tilt- and yaw-axis loads signals are subject to integral action using equal gains for both axes. After the reverse transformation – including separate azimuth offsets $\psi_o^n$ – the 1P and 2P contributions are summed to form the implementable IPC signals $\theta_i$ for attenuation of the respective blade out-of-plane harmonic loads. A pitch actuator model $G_a$ is included. The collective pitch and generator torque control signals, $\theta_0$ and $\tau_g$, are generated by turbine controllers, which are omitted in this figure.

## 2. Theory on IPC, MBC transformation, and the azimuth offset

The MBC transformation is generally employed for the implementation of IPC. This section briefly introduces the considered IPC setup that consists out of distinct MBC transformations for the reduction of 1P and 2P blade load harmonics. A schematic visualization of this scheme is given in Fig. 1. In addition to the general formulation, the azimuth offset is incorporated in the reverse transformation. This additional tuning parameter is used for decoupling of the multivariable system. An elaborate frequency-domain analysis on the azimuth offset is performed in [10], and the reader is referred to that work for further details.

The IPC configuration is implemented on the three-bladed DTU 10-MW reference wind turbine model [13]. The system is commanded with the respective torque and pitch signals, $\tau_g$ and $\theta_0$, originating from eponymous controllers. As the wind turbine model does not include pitch actuator dynamics, the system is augmented with unity-gain first-order actuator models

$$G_a(s) = \frac{1}{\tau_a s + 1}, \tag{1}$$

in which $\tau_a \in \mathbb{R}^+$ is the actuator time constant.

The out-of-plane blade bending moments $\boldsymbol{M}(t) \in \mathbb{R}^{\mathrm{B}}$ are supplied to the forward MBC transformation

$$\begin{bmatrix} M_{\mathrm{t},n}(t) \\ M_{\mathrm{y},n}(t) \end{bmatrix} = \mathbf{T}_n(\psi) \underbrace{\begin{bmatrix} M_1(t) \\ M_2(t) \\ M_3(t) \end{bmatrix}}_{\boldsymbol{M}(t)}, \tag{2}$$

with

$$\mathbf{T}_n = \frac{2}{\mathrm{B}} \begin{bmatrix} \cos(n\psi_1) & \cos(n\psi_2) & \cos(n\psi_3) \\ \sin(n\psi_1) & \sin(n\psi_2) & \sin(n\psi_3) \end{bmatrix},$$

in which $n \subset \mathbb{Z}^+$ is the harmonic number, B the total amount of blades, and $\psi_b \subset \mathbb{R}$ the azimuth angle for blade $b$, where $\psi = 0$ indicates the vertical upright position. The forward transformation transforms the rotating blade moments into a non-rotating reference frame. The fixed-frame and azimuth-independent tilt- and yaw-moments are represented by $M_{\mathrm{t}}$ and $M_{\mathrm{y}}$, respectively.

Subsequently, the fixed-frame tilt and yaw pitch angles are formed by the IPC controller, implemented in this paper as two decoupled SISO control loops. The SISO controllers take the form of two pure integrators $c_{\mathrm{I}}^n/s$ with equal gains on both axes, but different gains for the harmonics $n = \{1, 2\}$. The variable $s$ represents the Laplace operator. The non-rotating signals are converted to implementable IPC pitch contributions in the rotating frame by the reverse MBC transformation

$$\begin{bmatrix} \theta_{1,n}(t) \\ \theta_{2,n}(t) \\ \theta_{3,n}(t) \end{bmatrix} = \mathbf{T}_n^{-1}(\psi + \psi_{\mathrm{o}}^n) \begin{bmatrix} \theta_{\mathrm{t},n}(t) \\ \theta_{\mathrm{y},n}(t) \end{bmatrix}, \tag{3}$$

with

$$\mathbf{T}_n^{-1} = \begin{bmatrix} \cos\left[n\left(\psi_1 + \psi_{\mathrm{o}}^n\right)\right] & \sin\left[n\left(\psi_1 + \psi_{\mathrm{o}}^n\right)\right] \\ \cos\left[n\left(\psi_2 + \psi_{\mathrm{o}}^n\right)\right] & \sin\left[n\left(\psi_2 + \psi_{\mathrm{o}}^n\right)\right] \\ \cos\left[n\left(\psi_3 + \psi_{\mathrm{o}}^n\right)\right] & \sin\left[n\left(\psi_3 + \psi_{\mathrm{o}}^n\right)\right] \end{bmatrix},$$

where $\theta_{\mathrm{t}}$ and $\theta_{\mathrm{y}}$ are respectively the fixed-frame tilt and yaw pitch signals, and $\psi_{\mathrm{o}}^n$ is the azimuth offset for each harmonic. The offset is used in this paper for further decoupling of the tilt and yaw axes enabling the implementation of SISO IPC control loops. The 1P and 2P IPC pitch signals are summed into $\theta_i$, with $i = \{1, 2, 3\}$.

## 3. Theory on Gaussian processes and Bayesian optimization

The optimal tuning of controller parameters can lead to significant gains in terms of system performance. System performance is often quantified by an (unknown) objective function $f$. To find the minimizer or maximizer parameter set $\mathbf{x}^*$ for $f(\mathbf{x})$, generally, inefficient grid search approaches are employed. Explorations are commonly limited to a parameter space $\mathbf{x} \in \mathcal{X}$, where $\mathcal{X}$ is a compact $d$-dimensional subset of $\mathbb{R}$ [12].

Bayesian optimization (BO) has gained a lot of traction in the last decade, by its ability to data-efficiently optimize systems, of which evaluations are (computationally) costly. BO generally relies on a nonparametric probabilistic surrogate model, which is commonly a Gaussian process (GP). This model is consecutively updated and refined as new measurements or function evaluations become available. The data-driven model represents the believes on the objective

function at hand, and the described model-updating mechanism is often referred to as Bayesian posterior updating.

The following two sections provide the reader with the fundamental theory: the Gaussian process is discussed in Section 3.1, and the Bayesian optimization procedure is outlined in Section 3.2. The theory is summarized based on the work or Shahriari et al. [12], and the reader is referred to this work for further information.

### *3.1. The Gaussian process*

The Gaussian process is a nonparametric model, and commonly acts as a surrogate model for the unknown objective function $f$. A GP is fully defined by two ingredients: the mean function $\mu_0(\mathbf{x}) : \mathcal{X} \to \mathbb{R}$, and covariance function $k(\mathbf{x}, \mathbf{x}) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, the latter of which is also referred to as the positive-definite kernel. The prior mean is in practice often assumed to be a constant offset for all $\mathbf{x} \in \mathcal{X}$, such that $\mu_0(\mathbf{x}) \equiv \mu_0$.

A representation of the objective function $f$ is formed based upon the set of $l$ observations $\mathcal{O}_l = \{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$, where $\mathbf{x}$ represents an arbitrary test point, and $y$ the (possibly noisy) output of $f$. As a result, $f(\mathbf{x})$ is – like the observations – normally distributed, and described by the mean and variance functions:

$$\mu_n(\mathbf{x}) = \mu_0(\mathbf{x}) + \mathbf{k}(\mathbf{x})^{\mathrm{T}}(\mathbf{K} + \sigma^2\mathbf{I})^{-1}(\mathbf{y} - \mathbf{m}), \tag{4}$$

$$\sigma_n^2(\mathbf{x}) = k(\mathbf{x}, \mathbf{x}) - \mathbf{k}(\mathbf{x})^{\mathrm{T}}(\mathbf{K} + \sigma^2\mathbf{I})^{-1}\mathbf{k}(\mathbf{x}), \tag{5}$$

in which the covariance matrix $\mathbf{K}$ if formed by evaluating all covariance function pairs $K_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j)$, the mean vector $\mathbf{m}$ consists out of elements $m_i = \mu_0(\mathbf{x}_i)$, and $\mathbf{k}(\mathbf{x})$ is a one-dimensional covariance vector between $\mathbf{x}$ and the elements in $\mathbf{x}_{1:l} = \{\mathbf{x}_1, \ldots, \mathbf{x}_l\}$. The mean and variance evaluations of $f$ at $\mathbf{x}$ respectively represent the model prediction and corresponding uncertainty.

The covariance function $k$ has a fixed structure, and should be chosen to match the response function characteristics. The most common choice for kernel functions are Matérn and squared exponential kernels. These kernels are respectively characterized by the smoothness parameters $\nu = 1/2$ and $\nu \to \infty$. The squared exponential kernel is used in this work, and has the form:

$$k_{\mathrm{sq-exp}}(\mathbf{x}, \mathbf{x}') = \theta_0^2 \exp(-\frac{1}{2}r^2), \tag{6}$$

in which $r^2 = (\mathbf{x} - \mathbf{x}')^T\mathbf{\Lambda}(\mathbf{x} - \mathbf{x}')$ and the $(\cdot)'$-notation indicates the current test point, $\mathbf{\Lambda}$ is a diagonal matrix with squared hyperparameters $\theta = \{\theta_0^2, \ldots, \theta_d^2\} \in \mathbb{R}^{d+1}$, where the first and remainder elements respectively represent the amplitude and length scale parameters. The statistical information from the GP framework, is leveraged by the Bayesian optimization routine to select the next evaluation point $\mathbf{x}_{n+1}$, and is outlined in the next section.

### *3.2. Bayesian optimization*

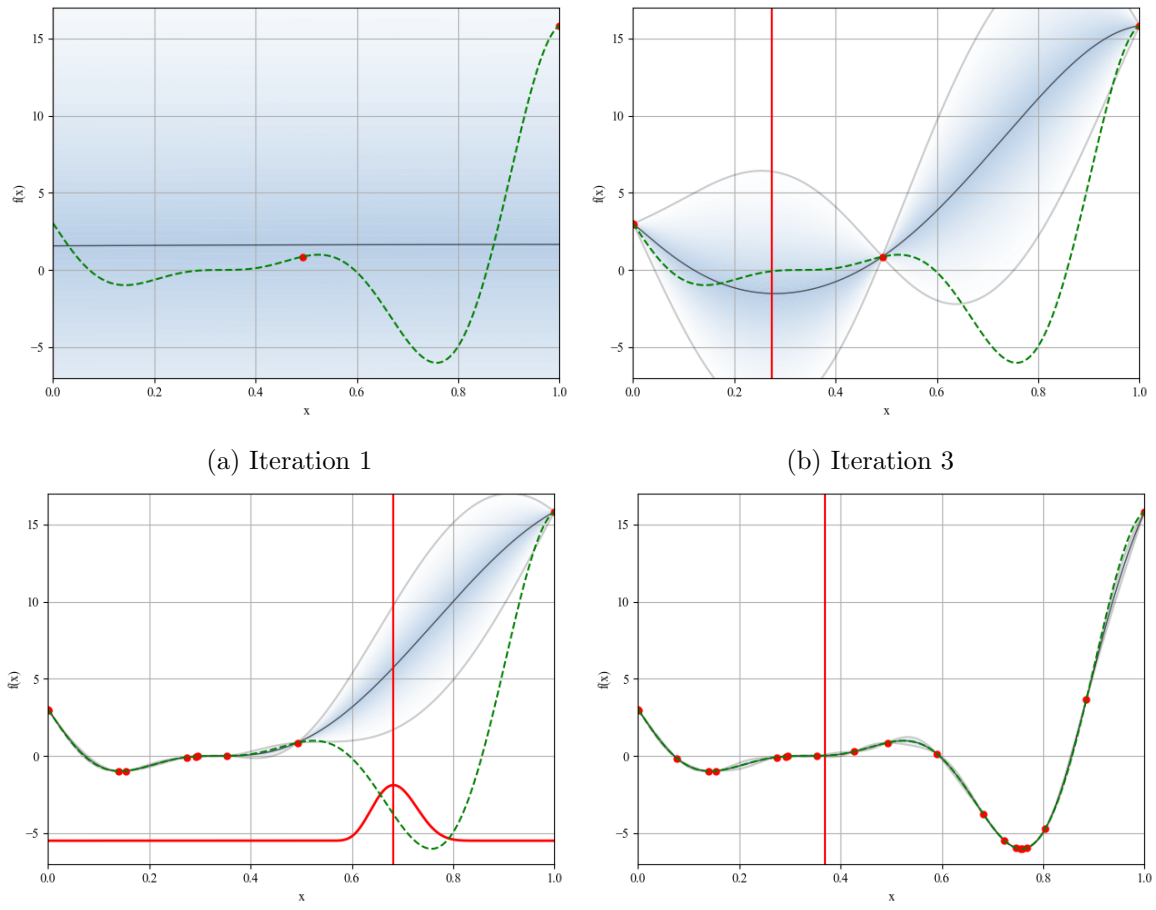
The previous section described the GP working principles as a way to statistically model the unknown objective function $f$. This section describes the mechanism to intelligently choose subsequent query points $\mathbf{x}_{n+1}$, based on information contained in the posterior model. The enabling ingredient in guiding this search is the acquisition function, commonly indicated by $\alpha$. Acquisition functions are subject to optimization for trading off search-space exploration and fruitful region exploitation. Commonly applied acquisition functions are Thompson sampling (TS), probability of improvement (PI), expected improvement (EI), upper confidence bounds (UCB), and entropy search (ES). As this work uses EI, only improvement-based policies are briefly described; for an explanation of the other acquisition function types, the reader is referred to [12].

The PI and EI strategies are both part of improvement-based acquisition functions, and optimize for new point evaluations that have a likely probability of improvement. Because PI considers all improvements to be equal, the strategy might behave rather aggressively [14]. EI is seen as an evolution of PI, as it also takes into account the amount of improvement, which results in less greedy searches.

*3.3. Illustrating the Bayesian working principles on a simple non-convex problem*

This section briefly illustrates the working principles of Bayesian optimization by an illustrative example. For this purpose, the Forrester function is employed [15], which is commonly used for the evaluation of optimization algorithms. The Forrester function is given by:

$$f(x) = (6x - 2)^2 \sin(12x - 4), \tag{7}$$




(a) Iteration 1

(b) Iteration 3

(c) Iteration 9

(d) Iteration 20

Figure 2: Bayesian optimization progress of the Forrester function (green dashed line). Function evaluations are used to learn the unknown objective by a Gaussian process in a data-driven way (black solid line). The statistical uncertainty is illustrated by the blue-shaded bounds. A total of 20 function evaluations (red dots) are performed to learn the function. Exploration and exploitation is driven by the acquisition function (red solid line): The maximum acquisition value, and the next evaluation point, is indicated by the red solid vertical line.

and is shown in Figure 2 (green dashed line). The function is multimodal in the sense that is has a global minimum, a local minimum, and a zero-gradient inflection point. The same figure shows the Bayesian optimization progress: in each evaluation the Gaussian process is refined by means of posterior updating. The acquisition function (red) explores regions with high statistical uncertainty (blue-shaded bounds), and exploits the areas to find the global minimizer. It is shown that the algorithm is effective in representing the true function by a nonparametric probabilistic surrogate model, and to find the function minimizer in a limited amount of iterations.

## 4. Methodology and implementation of the framework

This section combines the knowledge of the two previous sections, and describes the IPC optimization framework using Bayesian optimization. First, the methodology is formally described in Section 4.1, after which the practical implementation is outlined in Section 4.2.

### *4.1. Methodology for IPC optimization*

The proposed framework in this paper is to minimize the damage equivalent load (DEL) of the out-of-plane (OoP) blade root bending moments $\boldsymbol{M}_{\mathbf{y}}$. The DEL is taken as a quantitative measure for blade fatigue loading, which leverages the rainflow counting method for fatigue analysis. The DEL measure quantifies the amplitude of a certain harmonic load variation that would cause the same damage level when repeated for a given amount of cycles [16, 17]. Since the analysis is based on single load cases, the short-term DELs are calculated using the 1 Hz equivalent load [18]. The computation is executed offline using time series of high-fidelity simulation data. Because of the offline character, the DEL is considered to be a computationally intensive measure for optimization. Because Bayesian optimization is mostly employed for problems where evaluation of the objective function is expensive, the algorithm is seen as a good fit for the considered problem.

Before any simulation or optimization has been performed, the shape of the DEL objective function is unknown, and the optimal set of controller tuning parameters is yet to be found. To this end, a Gaussian process is employed for modeling the DEL objective function. This data-driven surrogate model is subsequently leveraged by the Bayesian optimization routine to find the optimal set of controller parameters in a minimum amount of iterations. The Bayesian approach utilizes the GP mean and variance information to effectively select and explore the promising locations, to efficiently find the global/local optimizer.

The parameter set is defined as $\mathbf{\Gamma} = \left\{c_{\mathrm{I}}^1,\, \psi_{\mathrm{o}}^1,\, c_{\mathrm{I}}^2,\, \psi_{\mathrm{o}}^2\right\}$ and are considered as the decision variables of the DEL minimization cost function:

$$\begin{aligned} \mathbf{\Gamma}^* &= \arg\min_{\mathbf{\Gamma}} \ \mathrm{DEL}(\boldsymbol{M}_{\mathbf{y}}) \\ \text{subject to} \quad & \text{Operational conditions: } U(\bar{\mathrm{U}},\, \mathrm{TI}), \bar{\tau}_{\mathrm{g}}, \bar{\theta}_0 \end{aligned} \tag{8}$$

in which $U$ the spatially and temporally distributed wind field with turbulence intensity TI and mean wind speed $\bar{U}$, $\bar{\tau}_{\mathrm{g}}$ is the applied mean rated generator torque, and $\bar{\theta}_0$ is the collective mean pitch angle. The considered DEL is calculated from high-fidelity time-series simulation data with a realistic turbulent input disturbance [19].

### *4.2. Implementation of the optimization framework*

A collection of tools is employed to implement the outlined framework for fixed-structure controller parameter optimization. The optimization framework has been implemented on the high-performance cluster of the Delft University of Technology, the Netherlands. The Bayesian optimization framework is implemented using the GPyOpt Python library [20], which is based on the GPy [21] library from the Sheffield machine learning group. The DTU 10-MW reference wind

turbine [13] is used in NREL's high-fidelity wind turbine simulation software OpenFAST [22]. The Delft Research Controller (DRC) [23] – an open-source baseline wind turbine controller – is employed in the automatic sequential simulation and evaluation framework to evaluate the IPC controller gain suggestions by the Bayesian routine.

For each Bayesian iteration, 6 high-fidelity OpenFAST simulations are run simultaneously. These simulations are subject to the same controller parameters $\Gamma_i$, but with different turbulent wind seeds. Each simulation has a length of 660 seconds, of which the first 60 seconds are discarded to exclude transient effects from the results. The pitch actuator model time constants are set to $\tau_\mathrm{a} = 0.4$ s. The simulation data is subsequently post-processed with `MLife` [24] for DEL calculation. The above described automated tuning framework has been made publicly available in an online repository [25].

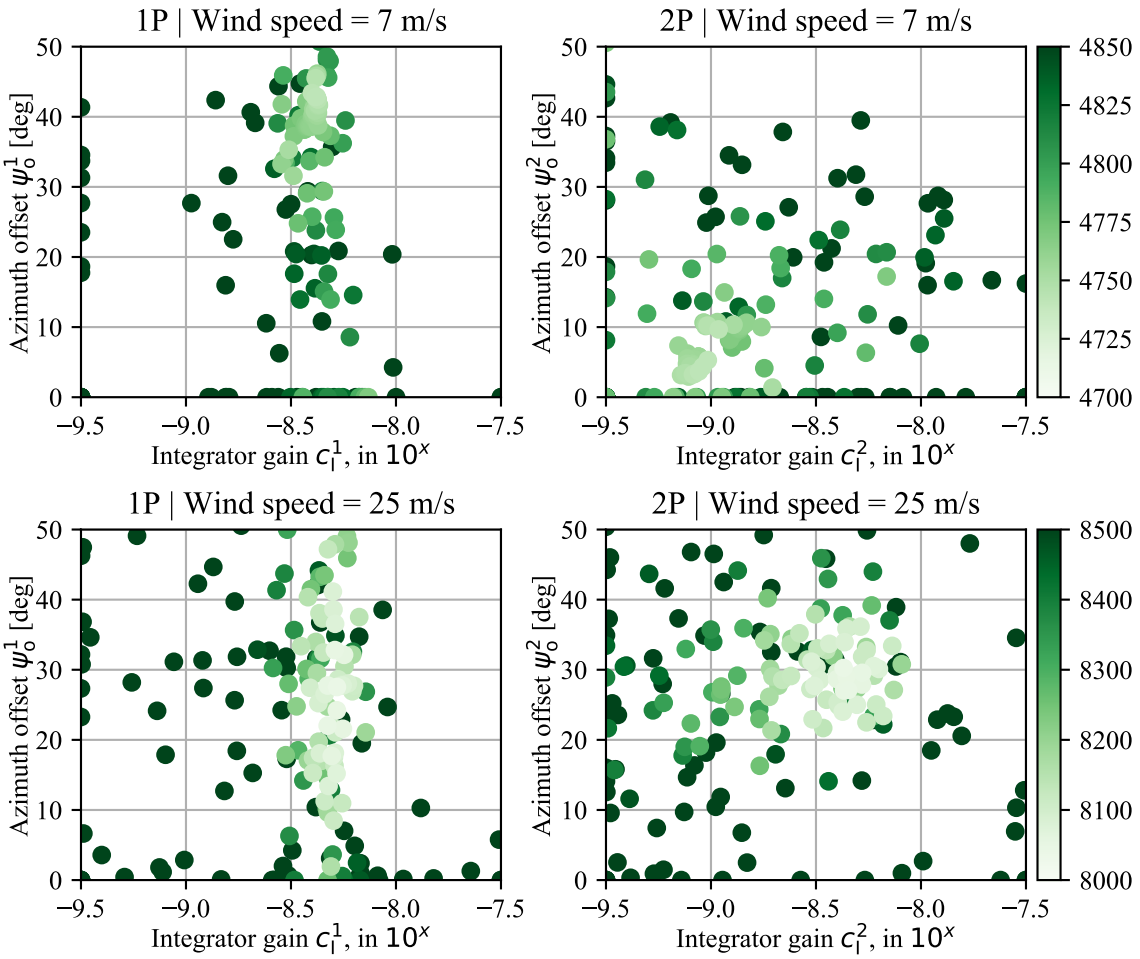



Figure 3: Bayesian optimization results for the combined optimization of the 4-DOF fixed-structure 1P + 2P IPC controller. The Bayesian routine clearly shows exploration and exploitation behavior: it explores the entire search domain, but exploits the posterior knowledge to search in promising areas (lighter colored dots). Each row shows the optimization at different wind speeds, and the left and right columns show the optimization progress for 1P and 2P IPC, respectively. The colorbar represents the DEL magnitude: values higher than the maximum colorbar level are saturated at that number.

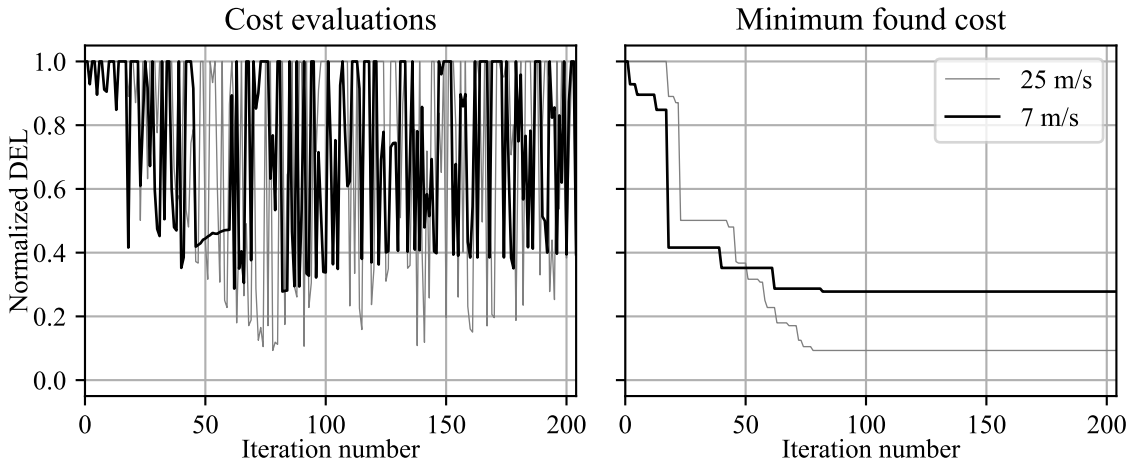



Figure 4: Bayesian optimization progress of the below- and above-rated wind speed cases. The left plot shows the raw objective function evaluations, whereas the right figure shows the progression of the minimum found cost so far, also in a chronological fashion. For both cases, the Bayesian routine finds a satisfactory minimum in approximately 80 iterations.

## 5. IPC optimization results using the Bayesian framework

The Bayesian optimization framework described in Section 4 has been employed for high-fidelity simulation cases at a wide range of wind speeds. The results of two wind speed cases of 7 and 25 m/s are presented in Figure 3. The results are split into 1P and 2P optimization for each case. All subfigures clearly show the ability of the Bayesian optimization framework to search the domain intelligently: it explores to reduce uncertainty in certain regions, while it exploits the posterior knowledge in promising areas.

The minimization progress of the objective function is presented in Figure 4. The objective function values are normalized according to the colorbar extrema in Figure 3, to allow for clear presentation of the different cases. The left plot shows the chronological search sequence, while the right plot gives a sense of the optimization convergence rate. The left plot keeps track of the found minimum DEL value so far, and shows that the objective function minimum is found very rapidly. For both feedback loops, a clear optimum is present and found in an efficient number of 75-100 iterations, without the need for any prior knowledge or parameter initialization. After finding this minimum, the algorithm performs additional exploration to search in uncertain regions, but also keeps exploiting the fruitful minimal area.

## 6. Conclusions

The presented Bayesian optimization tuning framework provides a convenient implementation for fixed-structure controller optimization, based on computationally costly objective functions. In this work, the framework is applied to a combined $1P + 2P$ IPC implementation, to efficiently find the optimal set of integrator gains and azimuth offsets. The latter mentioned variable is especially important for larger rotors with more flexible blades, as omittance of the offset results in multivariable coupling and adverse performance consequences. The objective is the minimization of the computationally expensive DEL of the OoP blade root bending moments. The DEL is computed offline from high-fidelity simulation data, and thereby results in a realistic controller tuning. The optimal set of controller parameters are efficiently found without the need for parameter initialization, and after a minimal amount of iterations. The implementation circumvents the need for expert control knowledge and specialized analysis software.

## References

[1] T Fischer. *Integrated Wind Turbine Design - Final report Task 4.1*. Tech. rep. UpWind, 2006.

[2] P Caselitz, W Kleinkauf, T Krüger, et al. "Reduction of fatigue loads on wind energy converters by advanced control methods". In: *European Wind Energy Conference (EWEC)* (1997).

[3] G. Bir. "Multi-blade coordinate transformation and its application to wind turbine analysis". In: *46th AIAA Aerospace Sciences Meeting and Exhibit* (2008).

[4] E. J. N. Menezes, A. M. Araújo, and N. S. B. da Silva. "A review on wind turbine control and its associated methods". In: *Journal of Cleaner Production* 174 (2018).

[5] E. Bossanyi. "Individual blade pitch control for load reduction". In: *Wind Energy* 6.2 (2003).

[6] M Geyler and P Caselitz. "Individual blade pitch control design for load reduction on large wind turbines". In: *European Wind Energy Conference (EWEC)* (2007).

[7] S. T. Navalkar, J. W. van Wingerden, E. Van Solingen, et al. "Subspace predictive repetitive control to mitigate periodic loads on large scale wind turbines". In: *Mechatronics* 24.8 (2014).

[8] M. D. Spencer, K. A. Stol, C. P. Unsworth, et al. "Model predictive control of a wind turbine using short-term wind field predictions". In: *Wind Energy* 16.3 (2013).

[9] Q Lu, R Bowyer, and B. L. Jones. "Analysis and design of Coleman transform-based individual pitch controllers for wind-turbine load reduction". In: *Wind Energy* 18.8 (2015).

[10] S. P. Mulders, A. K. Pamososuryo, G. E. Disario, et al. "Analysis and optimal individual pitch control decoupling by inclusion of an azimuth offset in the multiblade coordinate transformation". In: *Wind Energy* 22.3 (2019).

[11] S. P. Mulders and J. W. van Wingerden. "On the Importance of the Azimuth Offset in a Combined 1P and 2P SISO IPC Implementation for Wind Turbine Fatigue Load Reductions". In: *American Control Conference (ACC)*. 2019.

[12] B Shahriari, K Swersky, Z Wang, et al. "Taking the Human Out of the Loop: A Review of Bayesian Optimization". In: *Proceedings of the IEEE* 104.1 (2016).

[13] C. Bak, F Zahle, R. Bitsche, et al. "Design and performance of a 10 MW wind turbine". In: *Wind Energy* (2013).

[14] D. R. Jones. "A Taxonomy of Global Optimization Methods Based on Response Surfaces". In: *Journal of Global Optimization* 21.4 (2001), pp. 345–383.

[15] A. Forrester, A. Sobester, and A. Keane. *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons, 2008.

[16] G. Freebury and W. D. Musial. *Determining equivalent damage loading for full-scale wind turbine blade fatigue tests*. Tech. rep. Golden, Colorado: National Renewable Energy Laboratory (NREL), 2000.

[17] E. A. Bossanyi, P. A. Fleming, and A. D. Wright. "Validation of individual pitch control by field tests on two-and three-bladed wind turbines". In: *IEEE Transactions on Control Systems Technology (CST)* 21.4 (2013).

[18] T. Burton, N. Jenkins, D. Sharpe, et al. *Wind energy handbook*. Chichester, United Kingdom: John Wiley & Sons, 2001.

[19] NWTC Information Portal. *TurbSim*. `https://nwtc.nrel.gov/TurbSim`. June 2016.

[20] The GPyOpt authors. *GPyOpt: A Bayesian optimization framework in Python*. 2016.

[21] GPy. *GPy: A Gaussian process framework in python*. `http://github.com/SheffieldML/GPy`.

[22] NWTC. *OpenFAST*. `http://openfast.readthedocs.io`. [Online; accessed 14-November-2017].

[23] S. P. Mulders and J. W. van Wingerden. "Delft Research Controller: an open-source and community-driven wind turbine baseline controller". In: *TORQUE 2018*. 2018.

[24] NWTC Information Portal. *MLife*. `https://nwtc.nrel.gov/MLife`. Oct. 2015.

[25] S. P. Mulders, A. K. Pamososuryo, and J. W. van Wingerden. *Software implementation: Efficient tuning of Individual Pitch Control: A Bayesian Optimization Machine Learning approach*. DOI: `http://doi.org/10.5281/zenodo.3694181`.