American Control Conference (ACC 2019)

A Practical Bayesian Optimization Approach for the Optimal Estimation of the Rotor Effective Wind Speed

Nikolaos Moustakis, Sebastiaan Paul Mulders, Jens Kober, and Jan-Willem van Wingerden

***Abstract*— Modern wind turbines require careful tuning of controller and estimator parameters. However, tuning requires expert control experience, and is therefore in practice often performed by a trial-and-error brute-force approach. The contribution of this work is twofold. Firstly, a framework for tuning the parameters for conventional control and estimator architectures with Bayesian optimization is proposed. Secondly, the proposed scheme is applied to the problem of tuning Kalman filter parameters for the estimation of the rotor effective wind speed. For accomplishing the beforementioned task, the Bayesian optimization machine learning algorithm uses entropy search as utility function. The NREL 5-MW reference wind turbine is used in high-fidelity simulation software to show the efficacy of the proposed methodology. The Bayesian optimized Kalman filter configuration, is shown to estimate the rotor effective wind speed with a root mean square error smaller than 5 %, with respect to the actual effective wind speed over all load cases.**

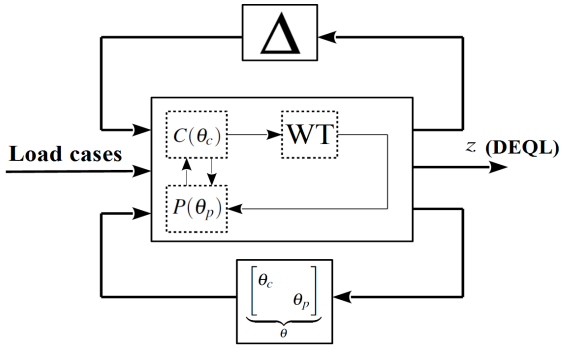

Fig. 1. Framework for estimating the controller $C(\theta)$ and estimator $P(\theta)$ parameters θ for a set of load cases.

I. Introduction

Wind turbines have received a lot of interest by industrial and research communities over the last decades. The attention is due to their small negative impact on environment and the huge potential to anticipate on the increasing global demand for renewable energy. A considerable effort has been made by the research community towards developing control algorithms for maximum wind power extraction, while simultaneously reducing fatigue loading [1], [2].

Wind turbine controller performance relies for a large extent on the quality of sensor measurements. The driving force that gives rise to the feedback control systems is the collection of precise information on the state of the controlled dynamic system, obtained or computed using sensor measurements. This information allows the control system to make appropriate decisions or compensate with computed signals [3].

A. Wind turbine control framework for finding optimal controller and estimator parameters

The above discussion outlines the importance of controller and estimator tuning. Present wind turbine controllers are mainly based on a measurement of the rotor or generator speed, complemented with additional control loops targeting specific loads. Depending on the operation region, torque and pitch control respectively maximize and limit the power extracted from the wind [4], [5]. However, there is an indirect relationship between the design equivalent loads (DEQL), which denote the compromise between the power production and the (fatigue) loads the wind turbine is subjected to. Furthermore, the DEQL cost function is highly non-linear with respect to the load variations. This results in convoluted tuning of the wind turbine controller as the trade-off between power yield and wind turbine loads is opaque.

The motivation of the current work stems from finding new ways to improve the wind turbine controller during its design process. This will be realized by incorporating machine learning (ML) methods to tune the parameters of conventional fixed-structure estimators and controllers. By using ML approaches, the wind turbine controller can be tuned to minimize the DEQL directly. Direct optimization of the controller and estimator parameters, combined with an accurate effective wind speed estimation, should allow the realization of optimal maximum power point tracking control algorithms [2, p. 153]. Moreover, a *rotor effective wind speed* (REWS) estimate allows common control objectives (e.g. power maximization, derating, fatigue load reduction) to be scheduled upon a single parameter for the whole operating region [6, p. 11].

The schematic representation proposed for tuning the controller and estimator parameters is shown in Fig. 1. The framework is general in the sense that it allows for optimization of the controller and estimator paremeters, respectively represented by $\{\theta_c, \theta_p\} \in \theta$, using the ML algorithm. Furthermore, the scheme is set up such that system uncertainties and disturbances are incorporated, represented by the Δ-block in Fig. 1.

All authors are from Delft University of Technology (TU Delft), 2628 CD, The Netherlands. Nikolaos Moustakis, Sebastiaan Paul Mulders and Jan-Willem van Wingerden are with the Delft Center for Systems and Control (DCSC), Jens Kober is with the Department of Cognitive Robotics (COR). S.P.Mulders@tudelft.nl, J.Kober@tudelft.nl, J.W.vanWingerden@tudelft.nl.

B. REWS optimal tuning approach

An overview of established methods towards REWS estimation can be found in [7]. A drawback of all REWS estimation approaches is the absence of a systematic procedure for tuning the parameters of the estimators, and one has to rely on intuition or rules of thumb. Most recently, an adaptive unscented Kalman filter was proposed [8], using a master-slave approach described earlier in [9]. The method allows for online adaptation of the noise covariance matrices for estimator tuning. The drawbacks of this method are the computational burden when the wind properties vary rapidly, and the poor noise covariance adaptation in case the observability condition number is large.

The current work examines the existence of an optimal configuration of static (Kalman filter) KF noise covariance matrices, resulting in satisfactory REWS estimation performance for different wind speed conditions. It is desirable to automate the optimization process, since numerous simulation iterations for different wind speed conditions need to be performed for performance evaluation purposes.

Bayesian optimization (BO) provides an elegant approach for this particular set-up, as a global optimization machine learning algorithm. It offers black-box and sample-efficient optimization, without relying on gradient or Hessian approximations [10]. More precisely, BO employs a Gaussian process (GP) as a non-parametric model for the unknown objective function that captures all prior knowledge. The underlying utility function suggests the next evaluation point to learn most about the location of the optimum.

The combination of controller tuning and BO has been successfully applied in the past. A framework for linear–quadratic regulator (LQR) tuning resulted in improved balancing performance of an inverted pendulum setup [11]. This idea is further expanded in [12] for LQR tuning of a humanoid robot balancing poles. Therein, the authors consider the entropy search (ES) optimization algorithm [13]. The same algorithm is adopted in this work, as it is demonstrated in numerical and real-world set-ups to deliver the optimal configuration in a minimum number of function evaluations.

The remainder of this paper is organized as follows. Section II formulates the REWS problem. Then, Section III presents the main result of this paper of modeling the objective function as a GP. Furthermore, the same section illustrates how the BO-ES combination is used as a utility function to estimate the optimal KF estimator parameters. The efficacy of the proposed methodology is illustrated through simulations in Section IV. Finally, Section V concludes this paper, stating the main contributions and proposing recommendations for future work.

II. REWS ESTIMATION

For estimating the REWS, this section reformulates the *power balance estimator* approach [6] to include an additional degree of freedom (DOF): the drivetrain rotational-flexibility. This DOF is usually ignored in literature with the aim to simplify the proposed design.

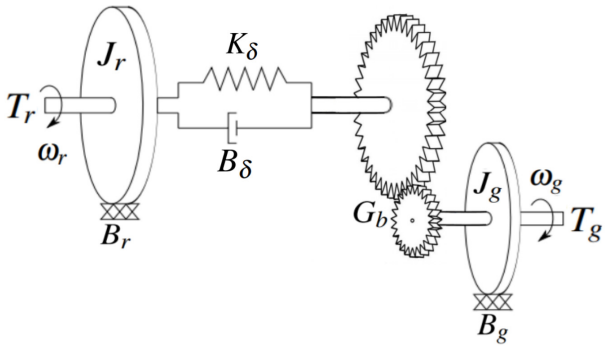

Fig. 2. Schematic representation of the wind turbine drivetrain [7, p. 1156].

A. Wind turbine drivetrain model

The considered wind turbine mechanical drivetrain model is illustrated in Fig. 2. The rotor and shaft inertia are combined in J_r, while B_r represents the friction of the rotor bearings. The drivetrain is modeled as a spring-damper system, with the respective stiffness and damping coefficients K_δ and B_δ. The angular velocity of the low-speed shaft is transmitted to the high-speed shaft through the gearbox with gearing ratio $G_\mathrm{b} \geq 1$. The combined inertia of the high-speed shaft, gearbox and generator is represented by J_g, and B_g is the friction induced by the generator. The dynamical system of the drivetrain is derived as:

$$\begin{aligned} J_\mathrm{r}\dot{\omega}_\mathrm{r} &= T_\mathrm{r} - B_\mathrm{r}\omega_\mathrm{r} - B_\delta\left(\omega_\mathrm{r} - \frac{\omega_\mathrm{g}}{G_\mathrm{b}}\right) - K_\delta\theta_\delta, \\ J_\mathrm{g}\dot{\omega}_\mathrm{g} &= \eta\frac{B_\delta\left(\omega_\mathrm{r} - \omega_\mathrm{g}/G_\mathrm{b}\right) + K_\delta\theta_\delta}{G_\mathrm{b}} - B_\mathrm{g}\omega_\mathrm{g} - T_\mathrm{g}, \\ \dot{\theta}_\delta &= \omega_\mathrm{r} - \frac{\omega_\mathrm{g}}{G_\mathrm{b}}, \end{aligned} \tag{1}$$

in which ω_r, ω_g, and T_g respectively represent the rotor speed, generator speed, and generator torque. The drivetrain torsion angle is denoted by θ_δ. The drivetrain properties are in accordance with [14]. The aerodynamic rotor torque is defined by

$$T_\mathrm{r} = \frac{1}{2}\rho\pi\frac{R^5\omega_\mathrm{r}^2}{\lambda^3}C_\mathrm{p}(\beta,\lambda), \tag{2}$$

which after separation in terms of the tip-speed ratio λ and the pitch angle β gives

$$\frac{2T_\mathrm{r}}{\rho\pi R^5\omega_\mathrm{r}^2} = \frac{C_\mathrm{p}(\beta,\lambda)}{\lambda^3} = F(\beta,\lambda). \tag{3}$$

The $F(\beta,\lambda)$ curve for the NREL 5-MW wind turbine is depicted in Fig. 3, and is obtained using the high-fidelity wind turbine simulation software FAST v8.16 [15]. The importance of the $F(\beta,\lambda)$-curve will be highlighted in Section II-B.

B. REWS estimation framework

The assumption is made that and pitch angle β and generator speed ω_g are measured quantities. Furthermore, the rotor speed ω_r and aerodynamic torque T_r are assumed to be perfectly reconstructed from measurements: this assumption is further clarified in the next section. Then, using (3), the

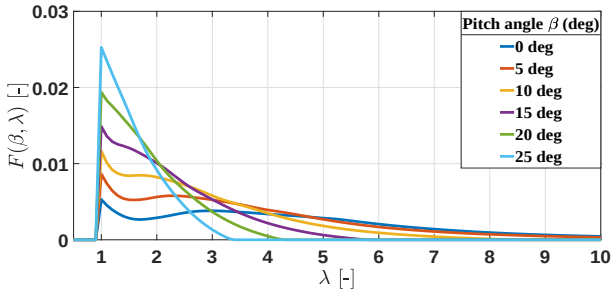

Fig. 3. The NREL 5-MW $F(\beta,\lambda)$-curve, generated using FAST.

effective wind speed V is calculated by first solving (3) for λ and then calculating the REWS as

$$V = \frac{\omega_r R}{\lambda}. \tag{4}$$

Remark 1: In principle, there are two methods to compute λ from (3): the polynomial or tabular implementation. In the former case, one solves the polynomial relation with respect to λ, whereas for the latter, λ is tabulated against F in a precalculated look-up table. Because both methods lead to similar results, the tabular approach is employed in this work for reasons of simplicity.

C. Estimation of aerodynamic torque and rotor speed

The aerodynamic torque T_r and the rotor speed ω_r in (4) are unknown, and thus must be estimated. Referring to the drivetrain model in (1), the following discrete state-space model is proposed:

$$x(k+1) = Ax(k) + B_1 T_\mathrm{r}(k) + B_2 T_\mathrm{g}(k), \tag{5}$$

in which the term k denotes the discrete-time operator and the state $x(k)$ is defined as $x(k) = [\omega_\mathrm{r}(k),\, \omega_\mathrm{g}(k),\, \theta_\delta(k)]$. For the purpose of estimating the unknown input, T_r is recast as a *random-walk* process [16]:

$$T_\mathrm{r}(k+1) = T_\mathrm{r}(k) + w_\mathrm{t}(k), \tag{6}$$

where w_t is a zero-mean white-noise sequence, uncorrelated with process and measurement noise. The augmented steady-state KF is now derived as

$$\begin{aligned}\begin{bmatrix}\hat{x}(k+1)\\ \hat{T}_\mathrm{r}(k+1)\end{bmatrix} &= \underbrace{\begin{bmatrix}A & B_1\\ 0 & 1\end{bmatrix}}_{\tilde{A}}\begin{bmatrix}\hat{x}(k)\\ \hat{T}_\mathrm{r}(k)\end{bmatrix} + \underbrace{\begin{bmatrix}B_2\\ 0\end{bmatrix}}_{\tilde{B}} T_\mathrm{g}(k)\\ &\quad + L(y(k)-\hat{y}(k)),\\ \hat{y}(k) &= \underbrace{\begin{bmatrix}C & 0\end{bmatrix}}_{\tilde{C}}\begin{bmatrix}\hat{x}(k)\\ \hat{T}_\mathrm{r}(k)\end{bmatrix},\end{aligned} \tag{7}$$

where the $\hat{(\cdot)}$-notation denotes the estimated values. The measurement y identifies with the generator speed ω_g, corrupted by zero-mean white noise w_g such that $y = \tilde{\omega}_\mathrm{g} = \omega_g + w_\mathrm{g}$. The $\tilde{(\cdot)}$-notation denotes noisy variables.

The Kalman gain L is computed by

$$\begin{aligned}P &= \tilde{A}\Big(\tilde{A} - P\tilde{C}^T(\tilde{C}P\tilde{C}^T + R)^{-1}\tilde{C}P\Big)\tilde{A}^T + \tilde{B}Q\tilde{B}^T\\ L &= P\tilde{C}^T(\tilde{C}P\tilde{C}^T + R)^{-1}\end{aligned} \tag{8}$$

in which P represents the steady-state error covariance matrix, and Q and R are to the tunable KF parameters being the process and measurement noise covariance matrices, respectively. The necessary background theory for derivation of (8) is comprehensively treated in [17]. The estimates $\hat{T}_\mathrm{r}$, $\hat{\omega}_\mathrm{r}$ along with the measured pitch angle β are used to compute an estimate of the tip-speed ratio

$$\hat{\lambda} = F^{-1}(\hat{\omega}_r, \hat{T}_r, \beta), \tag{9}$$

which is used to approximate the REWS by $\hat{V} = \hat{\omega}_\mathrm{r} R/\hat{\lambda}$.

Remark 2: Since for a single operating point (1) corresponds to a linear time-invariant (LTI) system, evaluation of (8) results in a constant Kalman gain L. Therefore, for each wind speed condition, the Kalman gain needs to be re-evaluated. Because it is more intuitive to tune the filter in terms of process and measurement noise variances, as opposed to the eigenvalues of the error-model, the former mentioned approach is employed.

D. Kalman filter optimization problem

As explained earlier in Section II-B and II-C, $\hat{V}$ depends on $\hat{\omega}_\mathrm{r}$ and $\hat{T}_\mathrm{r}$. The estimation quality of these two variables depends on the tuning of the noise covariance matrices Q and R. To account for the dependence of variables, the estimated REWS is parametrized as follows:

$$\hat{V}(\theta) := \hat{V}(W_Q(\theta), W_R(\theta)) \tag{10}$$

where $W_Q(\theta)$, $W_R(\theta)$ are defined as design matrices corresponding to noise covariance matrices. The elements in θ are the parameters to be tuned by the ES algorithm. To evaluate the performance of the learning algorithm, the minimization objective function is described as

$$J(\theta) = \frac{1}{\mathrm{N}}\sum_{t=1}^{\mathrm{N}} \Big\| V_t - \hat{V}_t(\theta) \Big\|_2, \tag{11}$$

where $\hat{V} \in \mathbb{R}^\mathrm{K}$ is the estimated REWS signal parametrized by the unknown θ, and $V \in \mathbb{R}^\mathrm{K}$ is the actual REWS signal provided by the simulation. The amount of samples for each simulation load case is indicated by K, and N represents the number of simulated scenarios. The solution to the REWS problem is given by

$$\theta_\mathrm{min} = \arg\min_{\theta} J(\theta) \quad \text{and} \quad \theta \in \mathbb{R}^\mathrm{D}. \tag{12}$$

III. The BO-ES learning algorithm approach

It is important to note that the shape of the non-parametric objective function (11) is unknown. In effect, one can not rely on gradient and Hessian approximations, which excludes the use of first or second order methods. In the considered setting, function evaluations are expensive by numerous simulations to be performed. Therefore, it is important to

spend the computational effort efficiently, by making clever choices on where to seek the parameters that are most likely to approach the optimum.

This section describes the BO-ES optimizer, which can globally explore the parameter space, and infer the optimum set of parameters that minimizes $J(\theta)$. It has to be noted that in the current work, the BO-ES combination is presented only from a high-level and practical viewpoint; for the mathematical details of GP, BO and ES as BO optimizer, the reader is referred to [13], [18] and [19].

A. Objective function viewed as a Gaussian process

To form a probabilistic belief over the objective function, J is modeled as the GP

$$J(\theta) \sim \mathcal{GP}\left(\mu(\theta), k_{\mathrm{SE}}(\theta,\theta')\right), \tag{13}$$

where $\mu(\theta)$ is the mean of $J(\theta)$, usually chosen equal to 0, and $k_{\mathrm{SE}}(\theta,\theta')$ the shift invariant squared exponential kernel

$$k_{\mathrm{SE}}(\theta,\theta') = \sigma_\theta^2 \exp\left[-\frac{1}{2}(\theta-\theta')^T \Lambda(\theta-\theta')\right], \tag{14}$$

where σ_θ represents the variance of the input θ and $\Lambda = \mathrm{diag}(\lambda_1, \lambda_2 \ \ldots \ \lambda_{\mathrm{D}})$ are the parameters *(length scales)* that determine how quickly $J(\theta)$ varies with the input.

The objective function in (11) assumes that the simulated REWS V is perfectly measured. In a more realistic situation however, one has to accommodate for noisy function evaluations of $J(\theta)$ to model uncertainty in the simulated REWS V or uncertainty of the tabulated solution of F in (3). Hence, noisy function evaluations of $J(\theta)$ are assumed

$$\tilde{J}(\theta) = J(\theta) + w_{\mathrm{J}} \tag{15}$$

where w_{J} denotes independent identically distributed Gaussian noise with variance σ_{J}^2. The total set of hyperparameters is defined as $\mathcal{H} = \{\lambda_1, \lambda_2 \ldots \lambda_D, \sigma_\theta, \sigma_{\mathrm{J}}\}$.

Assume the unknown function $\tilde{J}(\theta)$ is evaluated M times in respective locations of θ. Then the training data for the GP is defined as $\{\Theta, Y\} = \{\theta_i, \tilde{J}_i \mid i = 1, 2 \ldots M\}$. The joint distribution of the training outputs Y and the test outputs Y^* according to the prior is given by

$$\begin{bmatrix} Y \\ Y^* \end{bmatrix} \sim \mathcal{N}\left(0, \begin{bmatrix} \mathcal{K}(\Theta,\Theta) + \sigma_{\mathrm{J}}^2 I & \mathcal{K}(\Theta,\Theta^*) \\ \mathcal{K}(\Theta^*,\Theta) & \mathcal{K}(\Theta^*,\Theta^*) \end{bmatrix}\right), \tag{16}$$

in which $\mathcal{N}$ represents the normal distribution and the covariance matrix $\mathcal{K}$ is evaluated element-wise using (14). The predictive distribution of the test data Y^* conditioned on the test inputs and prior training data is represented as

$$Y^*|\Theta, Y, \Theta^* \sim \mathcal{N}\left(\overline{J}^*, \mathrm{cov}(J^*)\right). \tag{17}$$

The posterior mean and the posterior variance are given respectively as

$$\begin{aligned} \overline{J}^* &= \mathcal{K}(\Theta^*,\Theta)[\mathcal{K}(\Theta,\Theta) + \sigma_{\mathrm{J}}^2 I]^{-1} Y, \\ \mathrm{cov}(J^*) &= \mathcal{K}(\Theta^*,\Theta^*) \\ &\quad -\mathcal{K}(\Theta^*,\Theta)[\mathcal{K}(\Theta,\Theta) + \sigma_{\mathrm{J}}^2 I]^{-1} \mathcal{K}(\Theta,\Theta^*). \end{aligned} \tag{18}$$

The probability of the minimum of $J(\theta)$ in (13) is represented as

$$p_{\min}(\theta) := p(\theta = \arg\min_\theta J(\theta)), \ \theta \in \mathbb{R}^{\mathrm{D}}. \tag{19}$$

Let $S(\theta^*, \theta_{\min})$ describe the loss function when θ^* is chosen as the minimum of $J(\theta)$, instead of the real minimum $\theta_{\min}$. Then, this loss function induces utility for the knowledge about $\theta_{\min}$ through the loss functional

$$\mathcal{L}(p_{\min}) = \int_D \min_{\theta^*} S(\theta^*, \theta_{\min}) p_{\min}(\theta_{\min}) d\theta_{\min}, \tag{20}$$

where D a bounded domain. ES proposes to maximize the *Relative Entropy* $\mathcal{L}_{\mathrm{KL}}$ (Kullback-Leibler (KL) divergence). Hence we use its negative value as the loss functional in (20)

$$\mathcal{L}_{\mathrm{KL}} = -\int_D p_{\min}(\theta) \log \frac{p_{\min}(\theta)}{b(\theta)} d\theta, \tag{21}$$

where $b(\theta)$ is a uniform distribution. In mathematical statistics, the Kullback-Leibler (KL) divergence [20] measures how one probability distribution diverges from a second expected probability distribution, and corresponds to the information gain when comparing statistical models of inference. With this particular choice of loss functional, the loss is maximized for a uniform belief over the minimum and approaches negative infinity if p approaches the Dirac distribution. As a consequence, ES chooses those evaluation points for which the first Taylor expansion of $\mathcal{L}_{\mathrm{KL}}$, $\Delta\mathcal{L}_{\mathrm{KL}}$ is minimal. Thus, evaluating the points for which the information gain about the location of the minimum is maximized.

Remark 3: Continuous armed bandit algorithms like Probability of Improvement (PI) [21], the Expected Improvement (EI) [22] and the Upper Confidence Bound (UCB) [23], aim to minimize *regret*: the sum over function values at evaluation points. In contrast, probabilistic optimizers like ES, aim to obtain the minimum over the function horizon, regardless the function values at evaluation points. Although the two setups seem similar, they employ different strategies: the former methods explore as much as possible of the input space, whereas the latter performs a trade-off (*exploration*) of the search input space versus *exploitation* of current promising areas [19].

A last step is to define the hyperparameter set $\mathcal{H}$ for the GP in (13). At the beginning of the ES algorithm, the hyperparameter set is initialized at randomly chosen values. Then it is updated at every iteration of the ES algorithm so that the posterior distribution of the observations Y given the hyperparameters is maximized, i.e.,

$$\mathcal{H} = \arg\max_{\mathcal{H}} p(Y|0, \mathcal{H}). \tag{22}$$

The algorithm for tuning the KF parameters towards estimating the REWS is given in Algorithm 1.

IV. Simulation results

In this section, the BO-ES is employed as optimization algorithm to learn the optimal weighting matrices for the REWS KF defined in (7).

Algorithm 1 REWS KF tuning using BO-ES

function ES(Θ,Y) ▷ Initialize ES at random point
  $n \leftarrow 1$ ▷ Initialize number of function evaluations
  while $n \leq N$ **do** ▷ For N evaluation function points do:
    $[\overline{J}^*, \mathrm{cov}(J^*)] \leftarrow \mathrm{GP}(\Theta, Y)$ ▷ Condition GP on data
    $p_{\min} \leftarrow \mathrm{approx}(\overline{J}^*, \mathrm{cov}(J^*))$ ▷ Approximate $p_{\min}$
    $\theta \leftarrow \arg\min \Delta\mathcal{L}_{\mathrm{KL}}(p_{\min})$ ▷ ES proposes next evaluation point
    $\tilde{J} \leftarrow \mathrm{COST}(\theta)$ ▷ Evaluate cost at point
    $\{\Theta, Y\} \leftarrow \{\Theta, Y\} \cup \{\theta, \tilde{J}\}$ ▷ Update training data
    $\mathcal{H} \leftarrow \arg\max_{\mathcal{H}} p(Y|0, \mathcal{H})$ ▷ Optimize hyperparameters
    $\theta_{\mathrm{op}} \leftarrow \arg\max_{\theta} p_{\min}$ ▷ Update currently best optimum
    $n \leftarrow n+1$ ▷ Increment number of function evaluations
  end while
  return θ_{op} ▷ Return ES optimum
end function
function COST(θ)
  $L \leftarrow \mathrm{KALMAN}(W_{\mathrm{R}}(\theta), W_{\mathrm{Q}}(\theta))$ ▷ Compute Kalman gain
  $J_{\mathrm{Out}} \leftarrow 0$ ▷ Initialize cost
  for i=1 to K **do** ▷ For K in number simulations:
    $(V_i, \hat{V}_i(\theta)) \leftarrow \mathrm{SIM}(L)$ ▷ Run simulation and record $\hat{V}$, V
    $J = \mathrm{norm}(V_i - \hat{V}_i, 2)$ ▷ Compute cost of each simulation
    $J_{\mathrm{Out}} \leftarrow J_{\mathrm{Out}} + J$ ▷ Compute the cost over all simulations
  end for
  $J_{\mathrm{Out}} = J_{\mathrm{Out}}/K$ ▷ Average the total cost with number of simulations
  return J_{Out}
end function

A. Simulation set-up

For this work, the NREL 5-MW reference wind turbine is used and simulated in FAST v8.16. The NREL 5-MW reference model represents a conventional three-bladed upwind variable-speed, variable-pitch controlled turbine. The community-driven and open-source wind turbine baseline Delft Research Controller (DRC) [24] is employed, which among other features, includes below-rated torque control and above-rated collective pitch control.

For simulation purposes, the blades are assumed stiff so that flap- and edgewise blade motions are negligible and thus these DOFs are disabled. Another assumption is that the wind turbine operates in normal operating regimes, thus extreme operating conditions are not considered.

The turbulent wind profiles are generated according to the IEC 61400-3 standard [25], [26]. The load cases are of A, B, and C turbulence intensity (TI) classes for various mean wind speeds within the operating region ($3-25$ m s^{-1}). A schematic representation of the simulation set-up is depicted in Fig. 4.

Remark 4: A simulation environment is convenient for development and testing of the proposed REWS estimator optimization framework. One can likewise use real-world wind speed data obtained from meteorological mast or a LIght Detection And Ranging (LIDAR) device [27], and perform BO-ES for tuning of the estimator parameters. In this case, the REWS V is approximated by averaging the measurements from anemometers installed on the meteorological mast and shift the weighted value in time. The latter facilitates the direct extension of field-proven controllers to new advanced control designs without major modifications.

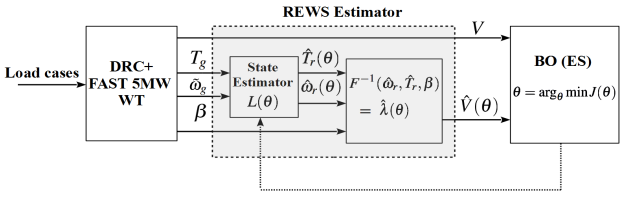

Fig. 4. ML framework for tuning KF parameters towards REWS estimation.

B. 2D tuning of the REWS KF estimator

To illustrate the performance of the BO-ES framework, the design matrices W_{Q} and W_{R} are chosen to satisfy the following configuration

$$W_{\mathrm{Q}}(\theta) = \mathrm{diag}(1e^{-6}, 10^{\theta_1}, 1e^{-6}, 10^{\theta_2}),\ W_{\mathrm{R}}(\theta) = 1, \quad (23)$$

where $Q = W_{\mathrm{Q}}$, $R = W_{\mathrm{R}}$ and $\theta \in \mathbb{R}^2$. The first and the third entry in Q, corresponding to the rotor speed and torsion angle, are chosen much smaller compared to R. This selection is based on the intuition that KF should put more weight on the correction based on the measurement rather than the prediction from the system model. This is a rather crude conservative design assumption but is adopted in this preliminary analysis to illustrate how the BO-ES algorithm performs. The manipulable entries in W_{Q} correspond to the generator speed and rotor torque.

The performance of BO-ES depends heavily on the dimensions of θ. More precisely, it is desirable to design W_{Q} and W_{R} to be sensitive with respect to the changes in θ to improve convergence performance of the optimizer, i.e., to cover a large space with small changes in the parameters.

Each choice of θ corresponds to 66 simulations with a mean wind speeds varying from $3-25$ m/sec, for normal and extreme turbulence classes. The generator speed signal ω_{g} to the KF is assumed to be corrupted by zero-mean white noise satisfying $\omega_{\mathrm{g}} \sim \mathcal{N}(0, \sigma_{\mathrm{g}}^2)$ with $\sigma_{\mathrm{g}} = 0.3$. Each simulated load case lasts 50 seconds and the first 5 seconds are omitted to remove transient effects. Besides, choices of θ leading to a non-convergent KF, or choices that lead to unacceptable prediction performance, are highly penalized. The results of BO-ES as optimizer for the REWS KF optimal parameter estimation problem are graphically depicted in Fig. 5.

To obtain an insight on the learned GP cost function surface, Fig. 6 shows the posterior mean versus the actual

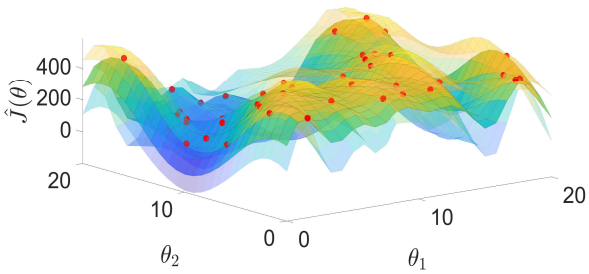

Fig. 5. The apparent colorful grid represents the GP posterior mean and the transparent areas the standard deviations. The red dots represent the evaluation points.

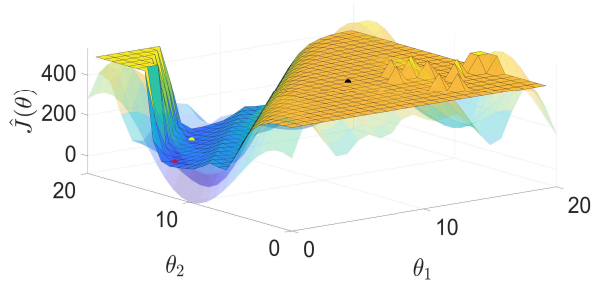

Fig. 6. Response surface versus the actual objective function evaluated on a grid. The black dot represents the initial point, the red dot corresponds to the minimum of the objective function and the yellow dot corresponds to the minimum found through OB-ES.

objective function evaluated over a grid of 400 points. The predicted values of $\tilde{J}(\theta)$ are very close to the actual ones at evaluation points. In contrast, the same figure reveals that far from the evaluation points, uncertainty represented by the posterior standard deviation increases. The minimum of $\tilde{J}(\theta)$ evaluated on the grid is computed 64.18, whereas the minimum found with BO-ES is 64.72. It is evident that the initial point resides in an area of high cost of $\tilde{J}(\theta)$.

Fig. 7 shows the cost function per iteration at evaluation points. The minimum is obtained at the 34th iteration. The key observation is the remarkable exploration strategy employed by ES: instead of following a greedy approach attempting to minimize the cost close to the obtained minimum, it evaluates regions where expected knowledge for the minimum is maximum. This results in an efficient sampling of the input space and learning of the unknown function, resulting in a solution very close to the global minimum.

C. 5D tuning of the REWS KF estimator

This section presents the main result of this paper, the optimal tuning of Q and R noise covariance matrices for the KF in (7). The weight configuration (10) is chosen

$$W_Q(\theta) = \text{diag}(\theta_1, 10^{\theta_2}, \theta_3, 10^{\theta_4}),\ W_R = \theta_5. \tag{24}$$

The optimal parameter set $\theta_{\text{op}} \in \mathbb{R}^5$ is computed as

$$\theta_{\text{op}} = [0.6835, 7.77, 0.99, 19.29, 0.0622] \tag{25}$$

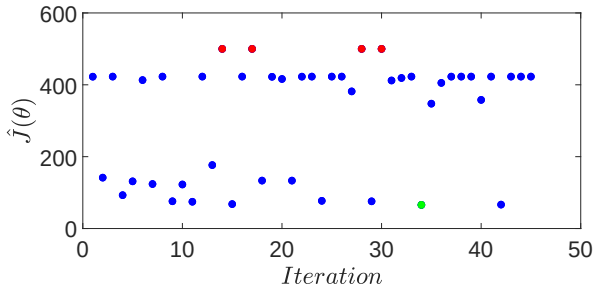

Fig. 7. Objective function cost versus number of iterations. The red dots correspond to non-convergent KF or very poor REWS performance. The green dot represents the smallest obtained value.

TABLE I: Cost $J(\theta)$ for training and test load cases.

Cost $\tilde{J}(\theta)$ for training and test data	
Training data	Testing data
mean 65.52	mean 65.72
standard deviation 1.89	standard deviation 3.58

TABLE II: Mean and standard deviation over the RRMSE for A, B and C TI wind load cases.

$\text{RRMSE}\left(V, \hat{V}(\theta)\right)$	A	B	C
mean	3.89%	4.31%	3.85%
std	1.43%	1.51%	1.15%

resulting in $\tilde{J}(\theta_{\text{op}}) = 65.34$. From (24), Q and R are computed

$$\begin{aligned} Q &= \text{diag}\left(0.684, 5.01 \cdot 10^7, 0.995, 19.49 \cdot 10^{18}\right) \\ R &= 0.0622. \end{aligned} \tag{26}$$

To validate the result, the obtained configuration is evaluated in load cases generated by different wind turbulence seeds. Table I shows the mean and standard deviation for training and testing data over 50 BO-ES realizations. It has to be noted that θ_{op} is found over all TI wind load cases, i.e., constant Q and R over A, B and C TI classes. To argue whether a scheduling approach for Q and R based on each TI class would yield significant improvement in the REWS estimation, the relative root mean square error (RRMSE) is computed for each load case using θ_{op} from (25):

$$\text{RRMSE}\left(V, \hat{V}(\theta)\right) = \sqrt{\frac{1}{\text{K}}\left(\frac{V - \hat{V}(\theta)}{V} \cdot 100\%\right)^2}. \tag{27}$$

Table II shows the mean and standard deviation of the RRMSE over the A, B and C TI classes for randomly generated load cases different from the ones used for training. The mean and standard deviation for all classes is small and hence, it can be concluded that the constant Q, R configuration yields satisfactory REWS estimation performance for all TI classes spanning the operation region of the WT. Figures 8-10 depict the estimated REWS versus the simulated REWS for three wind speed realizations.

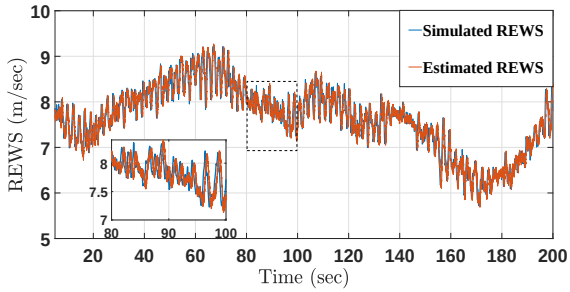

Fig. 8. IEC3NTM (normal turbulence) mean velocity 8 m s^{-1} TI class C. Magnified detail (80-100 s) represented by the dashed box.

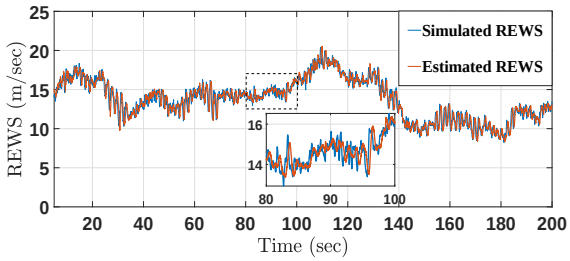

Fig. 9. IEC3ETM (extreme turbulence) mean velocity 16 $\mathrm{m\,s^{-1}}$ TI class A. Magnified detail (80-100 s) represented by the dashed box.

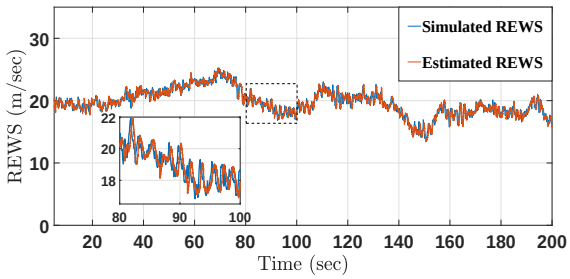

Fig. 10. IEC3ETM (extreme turbulence) mean velocity 20 $\mathrm{m\,s^{-1}}$ TI class B. Magnified detail (80-100 s) represented by the dashed box.

V. Conclusion

This work introduces framework towards tuning Kalman filter parameters for estimating the rotor effective wind speed using Bayesian optimization with entropy search. The obtained parameters were validated against randomly chosen data having realistic turbulence characteristics. The resulting effective wind speed estimator is proven to perform reasonably well, with a mean error less than 5 % over the nominal operation region of the WT. Future research directions include the extension of the introduced machine learning framework for tuning the parameters of WT controllers.

The use of effective REWS estimation using the presented approach, in combination with optimal wind turbine controllers tuning is expected to improve benchmarked values for power extraction and load mitigation. Moreover, the choice of kernel encodes assumed objective function properties as its shape and smoothness. A possible future direction is the development of new kernels specially designed for REWS estimation and wind turbine controllers, leading to improved machine learning performance.

Acknowledgments

This research has been (partially) funded by Wind Turbine Brain (TEWZ117008) as a TSE-17-09-01-Wind op zee R&D project, supported by TKI Wind op zee.

References

[1] J. G. Njiri and D. Söffker, "State-of-the-art in wind turbine control: Trends and challenges," *Renewable and Sustainable Energy Reviews*, vol. 60, pp. 377 – 393, 2016.

[2] E. A. Bossanyi, "The design of closed loop controllers for wind turbines," *Wind Energy*, vol. 3, no. 3, pp. 149–163, 2000.

[3] B. Ritter, A. Schild, and U. Konigorski, "Making nonlinear estimation techniques ready for use in industrial wind turbine control systems," in *Proceedings of Wind Europe Summit*, 2016.

[4] M. H. Hansen, A. Hansen, T. J. Larsen, S. Øye, P. Sørensen, and P. Fuglsang, "Control design for a pitch-regulated, variable speed wind turbine," Risø National Laboratory, Tech. Rep., 2005.

[5] T. G. van Engelen, E. L. van der Hooft, and P. Schaak, "Development of wind turbine control algorithms for industrial use," Energy research Centre of the Netherlands (ECN), Tech. Rep., 2001.

[6] E. L. van der Hooft and T. G. van Engelen, "Estimated wind speed feed forward control for wind turbine operation optimisation," in *Proc. Eur. Wind Energy Conf.*, 2004.

[7] M. Soltani, T. Knudsen, M. Svenstrup, R. Wisniewski, P. Brath, R. Ortega, and K. Johnson, "Estimation of rotor effective wind speed: A comparison," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1155–1167, 2013.

[8] B. Ritter, E. Mora, T. Schlicht, A. Schild, and U. Konigorski, "Adaptive sigma-point Kalman filtering for wind turbine state and process noise estimation," *Journal of Phys.: Conf. Ser.*, vol. 1037, no. 3, 2018.

[9] Q. Song and Y. He, "Adaptive unscented Kalman filter for estimation of modelling errors for helicopter," in *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2009.

[10] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," *Advances in Neural Information Processing Systems*, 2012.

[11] S. Trimpe, A. Millane, S. Doessegger, and R. D'Andrea, "A self-tuning LQR approach demonstrated on an inverted pendulum," *Proc. Vol.*, vol. 47, no. 3, 2014.

[12] A. Marco, P. Hennig, J. Bohg, S. Schaal, and S. Trimpe, "Automatic LQR tuning based on Gaussian process global optimization," in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2016.

[13] P. Hennig and C. J. Schuler, "Entropy search for information-efficient global optimization," *J. Mach. Learn. Res.*, vol. 13, no. 1, 2012.

[14] J. Jonkman, S. Butterfield, W. Musial, and G. Scott, "Definition of a 5-MW reference wind turbine for offshore system development," *NREL/TP*, 2009.

[15] J. Jonkman and B. Jonkman, "NWTC information portal (FAST v8)," *Journal of Physics: Conference Series*, vol. 1037, no. 3, 2015. [Online]. Available: https://nwtc.nrel.gov/FAST8

[16] K. Selvam, S. Kanev, J. W. van Wingerden, T. van Engelen, and M. Verhaegen, "Feedback–feedforward individual pitch control for wind turbine load reduction," *Int. Journal of Robust and Nonlinear Control*, vol. 19, no. 1, 2009.

[17] L. Xie, D. Popa, and F. L. Lewis, *Optimal and robust estimation: with an introduction to stochastic control theory*. CRC Press, 2007.

[18] C. E. Rasmussen and C. K. I. Williams, "Gaussian processes for machine learning (adaptive computation and machine learning)," *The MIT Press*, 2005.

[19] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas, "Taking the human out of the loop: A review of Bayesian optimization," *Proceedings of the IEEE*, vol. 104, pp. 148–175, 2016.

[20] S. Kullback and R. A. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, pp. 79–86, 1951.

[21] H. J. Kushner, "A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise," *Journal of Basic Engineering*, vol. 86, no. 1, p. 97, 1964.

[22] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *Journal of Global Optimization*, vol. 13, no. 4, pp. 455–492, 1998.

[23] N. Srinivas, A. Krause, S. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," in *Int. Conf. on Machine Learning (ICML)*, 2010.

[24] S. P. Mulders and J. W. van Wingerden, "Delft Research Controller: an open-source and community-driven wind turbine baseline controller," *Journal of Physics: Conference Series*, vol. 1037, no. 3, 2018.

[25] B. J. Jonkman and M. L. Buhl, "TurbSim User Guide," NREL, Tech. Rep. 1, 2006.

[26] NSAI, "Wind turbines – part 3: Design requirements for offshore wind turbines," Int. Electrotechnical Comm., Tech. Rep., 2006.

[27] J. Bao, H. Yue, W. E. Leithead, and J. Q. Wang, "Feedforward control for wind turbine load reduction with pseudo-lidar measurement," *International Journal of Automation and Computing*, vol. 15, no. 2, pp. 142–155, 2018.