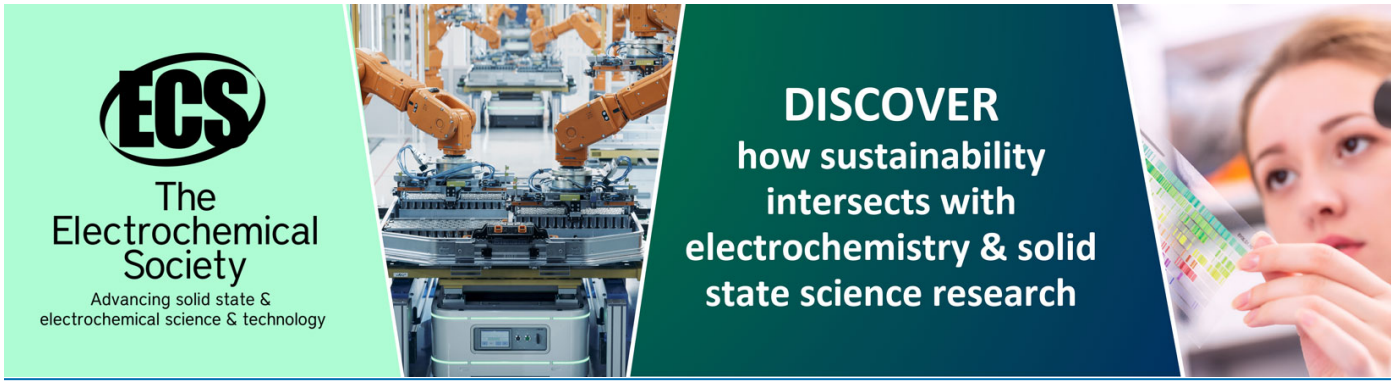
Model predictive control of wind turbine fatigue via online rainflow-counting on stress history and prediction



Model predictive control of wind turbine fatigue via online rainflow-counting on stress history and prediction

S Loew[1,2], D Obradovic[1] and C L Bottasso[2]

[1]Siemens AG, Corporate Technology, Otto-Hahn-Ring 6, 81739 Munich, Germany.
[2]TU Munich, Wind Energy Institute, Boltzmannstrasse 15, 85748 Garching, Germany.

E-mail: stefan.h.loew@tum.de

Abstract. The standard fatigue estimation procedure is implemented in Model Predictive Control via externalization of the Rainflow algorithm from the optimization problem. Additionally, stress history is considered in a consistent manner by employing a so-called stress residue. The formulation is implemented in the state-of-the-art MPC framework acados and tested in closed-loop with the 5MW onshore turbine in OpenFAST. Simulation results indicate that the new formulation outperforms conventional PID- and MPC-controllers over the entire wind regime, and that the consideration of stress history is highly beneficial.

1. Introduction

Fatigue is damage of a material caused by cyclic application of mechanical stress. For wind turbines, fatigue has large impact on lifetime e.g. of tower, blades and drivetrain, and is a main design driver. Model Predictive Controllers (MPC) enable optimal control of turbines by utilizing predictions of incoming wind by a *Light detection and ranging* (Lidar) device [1, 2]. Based on these input predictions, stress time series at crucial spots in the turbine structure can be predicted. Rainflow-counting (RFC) is the standard method for the decomposition of stress time series for fatigue estimation. Until recently, RFC could not be implemented in MPC [3] and could only be used for post-processing of measured and simulated data. In [4], a MPC formulation was presented that allows for the externalization of the RFC evaluation from the MPC algorithm, and the inclusion of its results into the MPC via time-varying parameters. Therefore, this formulation is referred to as *Parametric Online Rainflow-counting (PORFC)*. PORFC allows for the direct incorporation of monetary fatigue in the cost function of MPC, and thus for a true economic balancing with revenue from generated electricity.

In PORFC, fatigue is calculated based on stress information from the prediction horizon of the MPC, which is in the order of a few seconds. However, fatigue is a long-term effect where stress cycles are usually defined on much longer time spans. Therefore in [5] a systematic incorporation of historic stress samples ("residue") into the fatigue cost calculation of MPC was presented. In [4] and [5], these novel formulations were introduced in detail, but only preliminary closed-loop simulations were presented.

The main goal of the present work is to more thoroughly assess PORFC including stress history. Therefore, this paper is organized as follows. In Sec. 2, the phenomenon of fatigue and

cycle identification are assessed. This analysis is the basis for an application-focused description of PORFC in Sec. 3. In Sec. 4, aeroelastic simulations over the full range of turbulent wind conditions are carried out. Furthermore, PORFC is compared to a common MPC formulation from the literature and to a conventional controller.

2. Assessment of fatigue estimation

2.1. Definition of fatigue

In the following, the phenomenon of fatigue is defined for conditions and assumptions that apply to the wind energy domain: mechanical fatigue, normal ambient temperatures, neglection of irreversible strain effects and invariance w.r.t. time. In this setting, fatigue is damage of a material caused by cyclic application of mechanical stress. Without loss of information, the fatigue impact of a given stress-trajectory can be analyzed solely based on its extrema or "reversals". This implies that the shape and contained frequencies of the original continuous stress trajectory are considered to be irrelevant for fatigue estimation [6]. Therefore, the fatigue impact of such a reversal sequence is fully determined by its contained individual stress cycles. Each stress cycle can be represented by a cosine function. A stress trajectory typically contains full cycles, which are cosines of a full period, and half cycles, which are cosines of only a half period. Half cycles therefore represent either a rising or falling transient. Instead of storing three (full cycle) or two (half cycle) stress samples, it is common to store two stress samples and a weight, which is valued $w_c = 1$ (full cycle) or $w_c = 0.5$ (half cycle). The two stress samples can be the cycle stress maximum and minimum, or the stress amplitude $\sigma_{a,c}$ and mean $\sigma_{m,c}$. Instead of stress amplitude, stress range $\sigma_{r,c} = 2\sigma_{a,c}$ is frequently used as well.

Typically, fatigue impact of a stress cycle mainly correlates with its stress amplitude: a positive stress mean increases and a negative stress mean decreases fatigue impact. Quantitatively, this *mean stress effect* is expressed by the Goodman equation [7] (p. 184) which leads to the equivalent stress $\sigma_{eq,c}$. Consequently, equivalent stress is used to calculate the number of cycles to failure

$$N_c = f_{SN}^{-1}(\sigma_{eq,c}) \tag{1}$$

via the inverse S-N or *"Woehler"* curve, which typically has a piecewise definition over the stress-axis. Fatigue damage of a given stress cycle $D_{\text{fatigue,c}} = 1/N_c$ is obtained by the reciprocal of the number of cycles to failure. Total damage of the given stress-trajectory is obtained by linear accumulation $D_{\text{fatigue}} = \sum_c D_{\text{fatigue,c}}$ of damages of individual stress cycles according to the *Miner-Palmgren-Rule* [8].

2.2. Cycle identification via the Rainflow algorithm

Cycle identification is straightforward if, e.g., a simple sinusoid is analysed. There, amplitudes, mean values and number of cycles are obvious. However, realistic stress trajectories usually are highly complex and contain stress cycles that can be nested *("nested cycles")*. Additionally, half and full cycles can be present, as stated above. The most widely accepted algorithm for cycle identification from complex trajectories is the Rainflow(-counting) algorithm (RFC) [9]. A flowchart of the Rainflow algorithm is displayed in Fig. 1.

At the beginning of the algorithm, RFC receives as input a stress trajectory and extracts its reversals (extrema). Throughout the algorithm, reversals are read consecutively from left to right. Each new reversal is stored in an operational memory. From this memory, cycles are identified based on a triplet of reversals. The Rainflow algorithm contains four main loops. *Loop 1* initiates the reading of a new reversal sample, if less than three reversals are in the operational memory. *Loop 2* initiates the reading of a new reversal if, based on the current operational memory, no cycle could be identified. *Loop 3* and *Loop 4* initiate the subsequent check for a cycle in the current operational memory and are triggered after identification of a

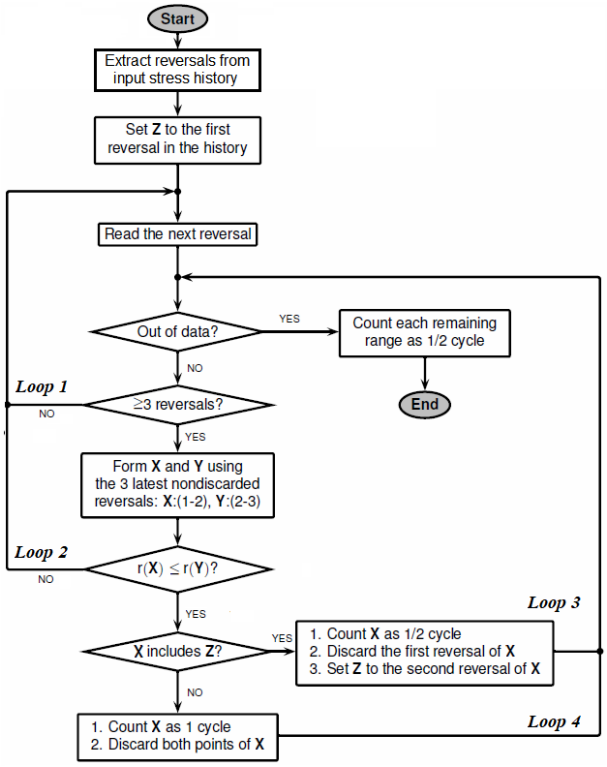


Figure 1. Flowchart of the MATLAB-implementation `rainflow()` of the *Three-point algorithm* (simplified from [10]). Stress extrema are called "reversals". The range $r(\boldsymbol{X}) = |X(2) - X(1)|$ of a stress value pair $\boldsymbol{X}$ is the absolute value of the difference between both stresses.

half or full cycle, respectively. A more comprehensive explanation of the algorithm can be found in [10].

As shown above, the Rainflow algorithm contains algorithmic branches and loops. Thus, a crucial property of the Rainflow algorithm is its discontinuous output behavior. Furthermore, the number N_c of identified cycles is not known before execution, but bounded by the number of extrema.

The characteristics of the identified cycles that are output by RFC for each cycle c are stress range $\sigma_{r,c}$ [Pa], stress mean $\sigma_{m,c}$ [Pa], sample index of cycle start $k_{start,c}$ [-], sample index of cycle end $k_{end,c}$ [-] and cycle weight $w_c[-]$. In the present work, these characteristics will be used in a converted form of stress amplitude $\sigma_{a,c}$ [Pa], stress mean $\sigma_{m,c}$ [Pa], sample index of cycle maximum $k_{max,c}$ [-], sample index of cycle minimum $k_{min,c}$ [-] and cycle weight $w_c[-]$.

2.3. Temporal range of cycle amplitudes and damage

It is important to note that stress cycles are not only caused by instantaneous oscillations, but also by long-term changes of deflection. This phenomenon is expressed by so-called *transition cycles* which grow over a long period of time and can reach high stress amplitudes with a dominating fatigue impact [11]. Since transition cycles, by definition, have not been closed yet, they appear as half cycles in the Rainflow analysis. In the wind turbine context, transition cycles are caused, e.g., by long-term evolution of mean wind velocity and can span across the turbulent and intra-day range (O(10s) to O(1day)). For the example of the turbine tower, a cycle can reach from turbine start-up (beginning of positive deflection) until shutdown (return to vertical orientation). To gain a quantitative impression of this long-term impact, in the following, an exemplary 620s tower base stress trajectory from a DLC 1.2 simulation is analyzed w.r.t. period time of contained stress cycles.

For the following analysis, the Rainflow cycles are binned w.r.t. their period time, and their

amplitudes $\sigma_{a,i}$ within the respective bins are summed. This is done according to

$$f_{\mathrm{RF}}(T_i) = \sum_i \sigma_{a,i} \quad \forall i |\ T_i \leq t_{\mathrm{end,i}} - t_{\mathrm{start,i}} < T_i + \Delta T, \tag{2}$$

where T_i are discrete period times, $\Delta T = 0.005$s is the chosen bin size, and $(t_{\mathrm{start,i}}, t_{\mathrm{end,i}})$ denote start and end time of a cycle, respectively. The resulting trajectory $f_{\mathrm{RF}}(T)$ is called *"Rainflow spectrum"* in the following.

For comparison, the amplitude spectrum for the same stress trajectory is calculated by a Fast Fourier Transform (*"Fourier spectrum"*). Both spectra are cumulated over time and normalized w.r.t. their end value. As shown in Fig. 2, the cumulative Rainflow spectrum is lower than the cumulative Fourier spectrum for low period times because, for the former, open stress cycles are not closed when they are interrupted by nested cycles. Instead, an open cycle is continued after closing of a nested cycle and can grow into a larger cycle with longer period time. This translates into a higher amplitude spectrum at higher period times for the cumulated Rainflow spectrum.

Consequently, for each bin, damage is calculated based on the output of the Rainflow analysis. Damage is as well cumulated over period time and normalized w.r.t. its end value, as shown in Fig. 3. The following properties of the cumulative damage trajectory are worth mentioning:

- Damage remains close to zero until period times of $T = 3$s, which is close to the first fore-aft eigenfrequency of the turbine tower. In this set, only low-amplitude cycles are present. Due to the superlinear nature of the damage function, these cycles only cause minor damage.
- For analysis windows of about $T = 4$s, already cycles with a 40% damage-equivalent effect can be seen entirely within the window.
- Analysis windows of more than $T = 20$s would be necessary to see cycles with a 80% damage-equivalent effect within the window.

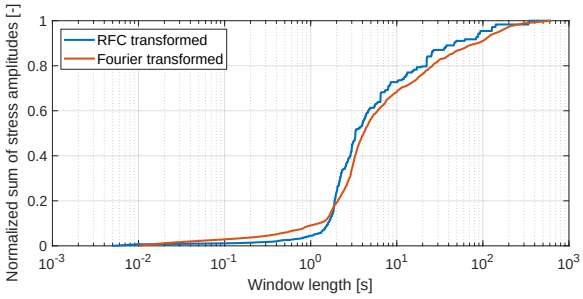


Figure 2. Normalized cumulative amplitude spectrum obtained via the Rainflow algorithm and a Fourier transformation, respectively.

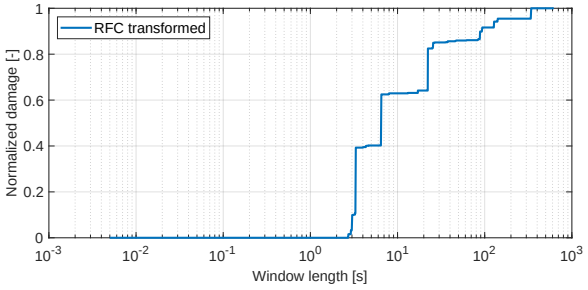


Figure 3. Normalized cumulative fatigue damage obtained via the Rainflow algorithm.

2.4. Batchwise cycle identification & residue

As demonstrated in Sec. 2.3, wind turbine stress trajectories can contain long-term cycles. Thus, the Rainflow analysis has to be carried out over the entire length of an available stress trajectory. For offline purposes, this mode is perfectly adequate. However, for online monitoring and control, a complete Rainflow analysis for each newly measured stress sample is computationally infeasible. As a solution, in [12] it is shown that Rainflow analysis also can be performed batchwise if a so-called *residue* is used for carrying along the half-cycle stress samples. Residue, therefore, denotes a set of stress samples that occured in the past and have not formed full cycles as yet.

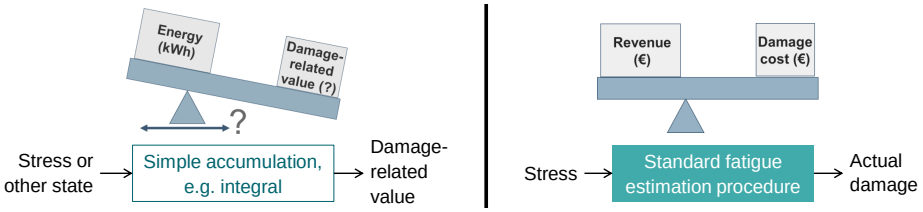


Figure 4. Left: Indirect fatigue metric. Right: Direct fatigue metric.

Depending on the stress signal, a high number of samples can be accumulated in the residue. The highest possible dimensions of the residue vector result from diverging and converging stress time series, because they result in a very high number of half cycles [13]. However, long-term diverging series are unrealistic, because unstable machine behavior typically is counteracted by the controller or an emergency shutdown. Long-term converging series are irrelevant, since very low-amplitude cycles can be discarded without significant errors in fatigue estimation. To conclude, the dimension N_{res} of the residue vector is finite and remained well below 100 in practical tests [5].

3. Fatigue in Model Predictive Control of wind turbines

Wind turbine tower base fatigue usually is implemented in MPC within the cost function. Common cost types in MPC are *Stage cost* and *Terminal cost*. Stage costs comprise a summation of state samples or a time integral of state trajectories over the prediction horizon, and are preferred for the present application. Terminal costs are defined as a function of the sole state samples at the end of the prediction horizon [14].

3.1. Indirect fatigue metrics in MPC

Several approaches reported in the literature involve indirect fatigue metrics [3], [15], [16]. However, indirect fatigue metrics have two main disadvantages:

- Instead of actual damage, only a damage-related value is obtained and optimized, as illustrated in Fig. 4.
- Indirect fatigue terms have different units from harvested energy. Thus, weighting both terms in the cost function is not straightforward.

The most common approach involves quadratic penalization of tower tip deflection rate $\dot{d}_T$. This also can be interpreted as a penalization of kinetic energy of the lumped tower mass m_T, averaged over the prediction horizon $T_{horizon}$. In the present work, therefore, the stage cost

$$J_{fatigue,TTVP} = \int_{t_0}^{t_{end}} \frac{1}{2T_{horizon}P_{\mathrm{g,max}}} m_T \dot{d}_T^2 \mathrm{d}t \tag{3}$$

is used for comparison and referred to as *Tower tip velocity penalization (TTVP)*. An additional division by rated power $P_{\mathrm{g,max}}$ is used for scaling the cost, which is beneficial for optimization.

3.2. Direct fatigue metrics in MPC

In contrast to indirect fatigue metrics, direct fatigue metrics return actual damage, which can be readily converted to monetary fatigue cost, as visualized in Fig. 4. This conversion is achieved for instance by multiplication with Initial Capital Cost of the respective component or the entire turbine. Since harvested energy also can be converted to revenue by electricity price, the optimization algorithm can directly maximize profit.

As shown in Sec. 2, direct fatigue estimation involves the Rainflow algorithm. Implementation of RFC within a gradient-based optimization seemed impossible until now due to following obstacles:

- RFC is a function of all stress samples. Therefore, neither the concept of stage nor of terminal cost applies.
- RFC contains branches. Therefore, it exhibits discontinuous outputs and is not continuously differentiable.
- RFC contains "while" loops, which lead to a changing function execution structure depending on the stress input.

Thus, in all known references, the Rainflow algorithm is approximated to some extent. In [17], a version of *Simple Range Counting* is applied, which is standardized in [9]. In [6], hysteresis operators are used to adapt parameters of a cost function in MPC. This cost function penalizes deflection rates, comparable to TTVP. In [18], damage estimation including standard RFC is performed on a large number of stress time series which are used to train a surrogate Artificial Neural Network. The latter seems to be very promising in terms of correct damage estimation. However, the approach involves a high a priori engineering effort, as well as a significantly increased computational load in the MPC [18].

Stress history is not included in any of these approaches. In [6], the hysteresis operators only have memory of damage evolution. Similarly, in [18], only the previous fatigue rate output of the ANN is memorized until the next evaluation.

3.3. Parametric Online Rainflow-counting - Concept

The above mentioned obstacles for a direct implementation of RFC in MPC are overcome by the method of *Parametric Online Rainflow-counting (PORFC)*. In PORFC, all discontinuous parts of the fatigue estimation procedure are carried out before each execution of the MPC algorithm, as shown in Fig. 5. Additionally, the stress history is incorporated via a residue which is inspired by the batchwise cycle identification in Sec. 2.4. The algorithmic workflow is as follows:

- **Simulation:** The reduced wind turbine model is simulated over the prediction horizon using the current measured states as initial values. Relevant result is a stress prediction, as visualized in Fig. 6 (right).
- **Merge:** The residue is merged with the stress prediction.
- **Rainflow:** The Rainflow algorithm is used to identify stress cycles over this merged trajectory. Consequently, it is assumed that the structure of identified cycles does not change within the upcoming optimization run. The term "structure" denotes here positions ($k_{\mathrm{min,c}}$, $k_{\mathrm{max,c}}$) and weights (w_c) of cycles. As shown in Fig. 6, this assumption implies that the controllable extrema in the prediction horizon only can be shifted vertically by the optimization.
- **Residue update:** Stress cycles can be composed by stress samples only from residue or prediction, or by a combination of both. However, only controllable samples within the prediction horizon can be altered by the optimization. Especially the measured initial value at prediction step 0 cannot be altered and, therefore, is added to the residue. If a full cycle is detected entirely within the *residue*, both contributing values are discarded from the residue. The reason for this is that also in the future they will never anymore form a cycle with a sample from the prediction and, therefore, are irrelevant for the MPC.
- **Time-varying parameters:** Information from cycle identification is used to fill vectors of time-varying parameters, which are forwarded to the cost function of the MPC. Details on this step are provided in Sec. 3.4.

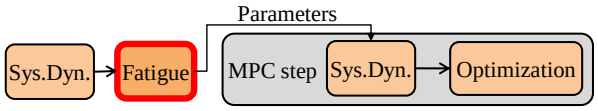


Figure 5. Externalization of fatigue estimation (Rainflow algorithm) from MPC.

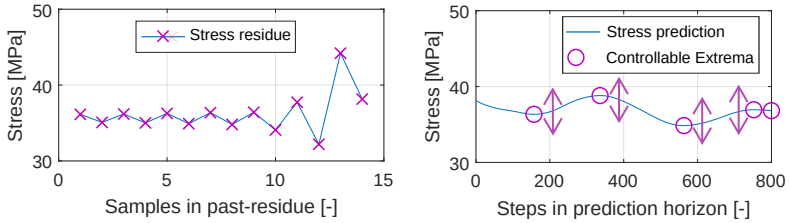


Figure 6. Left: Stress residue from the past. Right: Stress prediction in the future.

- **Optimization/MPC:** In the cost function of the MPC, the parameters are used to time-continuously calculate fatigue cost over the horizon and accumulate it via integration. Finally, the optimization problem is solved and the resulting control variables are applied to the wind turbine plant.

3.4. Parametric Online Rainflow-counting - Time-varying parameters & cost function

Distribution of damage over time: Since information from cycle identification is forwarded to the MPC via parameters, which are varying over the prediction horizon, the total fatigue damage has to be distributed over the prediction horizon, as visualized in Fig. 7 (right). Therefore, the damage of each stress cycle is split into two halves, which are allocated to the two contributing stress samples. For example, cycle 4 is formed by samples $k = 5$ and $k = 8$. Their fatigue cost terms therefore are allocated to these samples, as shown by the blocks in Fig. 7 (right). This example also shows an important property of the Rainflow algorithm, which identifies cycle 4 even though it is interrupted by the nested cycle 2. If, for a given stress sample, the complementary stress sample is not controllable (residue or initial value), all damage is allocated to the given sample. Here, this is the case for cycle one, where all damage is allocated to sample $k = 2$ since the complementary stress sample at $k = 0$ is not controllable.

Setup of the time-varying parameters: Figure 7 (left) visualizes the generation of the time-varying parameters. Since each stress extremum belongs to one or two stress cycles [19], the Rainflow algorithm provides one or two mean stresses per extremum. These mean stresses (M1 - M4) are considered as optimization- or tracking-goals for the current MPC-step. A more detailed derivation and explanation can be found in [4].

Cost function: Consequently, the fatigue cost term of PORFC is defined. One comment on notation: the variable notation with hat $\hat{a}$ means fixed for one MPC-step and with bar $\bar{a}$ means sampled on the control intervals of the prediction horizon.
The fatigue cost function is defined by an integral over two cost terms, each one representing one potential cycle contribution of a stress sample, i.e.:

$$J_{fatigue,PORFC}\left(\sigma, \bar{\boldsymbol{p}}\right) = \frac{1}{T_{cntrl}} \int_{t_0}^{t_{end}} \left(J_{fatigue,c}(\sigma(t), \hat{\bar{\sigma}}_{m,c1}(t), \hat{\bar{w}}_{c1}(t)) + J_{fatigue,c}(\sigma(t), \hat{\bar{\sigma}}_{m,c2}(t), \hat{\bar{w}}_{c2}(t)) \right) \mathrm{d}t \quad [€]. \quad (4)$$

The cost terms are "switched on" by nonzero cycle weights $\hat{\bar{w}}_{c1/2}(t)$. Mean stresses $\hat{\bar{\sigma}}_{m,c1/2}(t)$

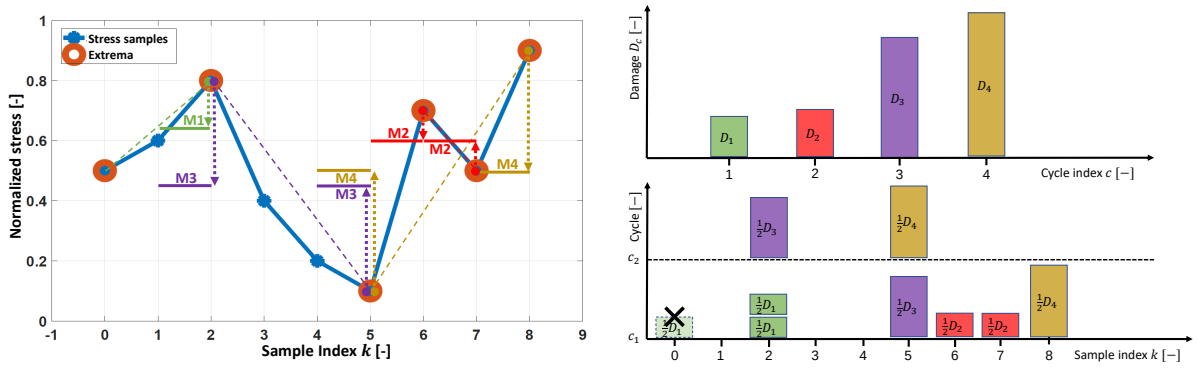


Figure 7. Left: Initial stress trajectory (blue), its extrema (orange circles), generated time-varying reference stresses (solid green, red, purple, yellow) and optimization goals (dotted arrows) for PORFC. Right: Corresponding distribution of damage over the prediction horizon. Both figures are modified from [20].

and cycle weights $\hat{\bar{w}}_{c1/2}(t)$ are collected in the parameter vector

$$\bar{\boldsymbol{p}} = \left(\hat{\bar{\sigma}}_{m,c1}, \hat{\bar{\sigma}}_{m,c2}, \hat{\bar{w}}_{c1}, \hat{\bar{w}}_{c2}\right), \tag{5}$$

which is defined as piecewise constant over the control intervals of the prediction horizon. The cost of individual cycles is defined by

$$J_{fatigue,c} = \hat{\bar{w}}_{c1/2}(t)\ a_m\ |\sigma(t) - \hat{\bar{\sigma}}_{m,c1/2}(t)|^m. \tag{6}$$

3.5. Optimization problem for TTVP and PORFC

The Economic MPC of a wind turbine is defined by the following optimization problem

$$\min_{\bar{\mathbf{u}},\bar{\mathbf{s}}} \left(-\alpha_{\text{revenue}} J_{\text{revenue}} + \alpha_{\text{fatigue}} J_{\text{fatigue}} + \int_{t_0}^{t_{end}} \left(10^{-2}|\dot{\beta}_b| + 10^7\ \bar{s}_\omega^2 + 10^7\ \bar{s}_P^2 \right) \mathrm{d}t \right), \tag{7}$$

which maximizes the revenue J_{revenue} and minimizes the fatigue $J_{\text{fatigue,PORFC}}$, where α_{revenue} and α_{fatigue} are weighting factors. Instead of generator power, aerodynamic power is maximized by $J_{\text{revenue}} = \omega_r\, T_w(\omega_r, V_{rel}, \beta_b)$ to avoid a greedy extraction of rotor kinetic energy by the MPC (*turnpike effect*), as suggested by [15]. Furthermore, pitch travel $|\dot{\beta}_b|$, and slack variables for rotational speed $\bar{s}_\omega$ and generator power $\bar{s}_P$ are penalized (see their use in the constraints below). The optimization variables are the demanded pitch angle and torque rate $\bar{\mathbf{u}} = (\bar{\boldsymbol{\beta}}_{b,d}, \bar{\dot{\mathbf{T}}}_{g,d})$, and the slack variables $\bar{\mathbf{s}} = (\bar{\mathbf{s}}_\omega, \bar{\mathbf{s}}_P)$.

For both TTVP and PORFC, revenue is weighted by the current electricity price $\alpha_{\text{revenue,PORFC}} = p_{\text{elec}}$ [€/Ws] to match the monetary nature of (4). The fatigue weight remains free and will be determined later in this work.

The optimization problem is subject to:

- The system dynamics of a reduced turbine model $\dot{\boldsymbol{x}} = \boldsymbol{F}(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{d}(t))$ with six states: rotational speed of the rotor, tower tip deflection, tower tip velocity, pitch angle, pitch rate and generator torque. More details about the model are given in [21].
- Inequality constraints over the horizon, to keep rotational speed, tower deflection (yield strength), pitch angle, pitch rate, generator torque and generator power within their limits. In order to maintain feasibility of the optimization despite model uncertainties and spontaneous constraint violations, the constraints on rotational speed and generator power are augmented by slack variables, as suggested by [22].
- Box constraints on control and slack variables.

4. Simulation setup and results

4.1. Simulation setup

Plant model: The designed MPC formulations are tested with the NREL 5MW onshore reference turbine [23] in the aeroelastic simulator `OpenFAST`. The present preliminary study is focused on assessment of the formulations in ideal conditions. Therefore, exact measurements of the required turbine states are provided to the MPC. Furthermore, in `OpenFAST`, only the tower fore-aft and the rotor Degree of Freedom (DOF) are activated. Future work will be directed to extending the MPC-internal turbine model and finally enabling all DOFs in `OpenFAST`. All results in this work are mean values of 12 simulations (each with a different seed) of 600s length in DLC 1.2 with turbulence category A.

Lidar model: The Lidar-simulator of `OpenFAST` is used and set to provide exact pointwise wind measurements. These measurements are obtained via circular scan patterns of four different radii $r = \{0.2, 0.4, 0.6, 0.8\}R_{\text{rotor}}$ which are applied in alternating fashion. For each radius, 20 azimuthal stops are distributed equally (every 18 degrees). Inspired by [2], each measurement is weighted by the cubic of the scanning radius, since the associated area increases quadratically, and the stationary spanwise variation $\frac{\partial C_p}{\partial r}$ of the power coefficient increases approximately linearly with the scanning radius. Every 5 ms a new wind sample is obtained. Due to vertical wind shear, this raw wind prediction signal oscillates, an effect which is counteracted by applying a moving mean filter, whose window length corresponds to two full scans over all radii and azimuths. The longitudinal scanning distance is set to $x_{scan} = V_{w,ref}\, T_{\text{horizon}}$, where $V_{w,ref}$ is the reference wind velocity of a given DLC, and $T_{\text{horizon}} = 8$s is the horizon length of the MPC. This value for the horizon length was chosen based on the findings in Sec. 2.3 which indicate, that cycles of at least 60% of damage-equivalent effect will be contained in the controllable prediction horizon.

MPC framework: The MPC is implemented in the state-of-the-art `acados` framework [24], using the interior-point solver `HPIPM` for the underlying Quadratic Programs (QP). The controller sample time is 100 ms. Maximum 5 QPs are solved per MPC step to ensure results close to convergence. The Hessian matrix is automatically convexified to account for possible numerical issues due to the highly non-standard cost formulation of PORFC.

Controller variants: In the following, performance of five MPC formulations and the baseline conventional controller (CC) from NREL [23] are compared. The MPCs involve the conventional formulation of Tower tip velocity penalization (TTVP, Sec. 3.1) and the novel formulation of Parametric Online Rainflow-counting (PORFC, Sec. 3.3). For PORFC, a fatigue exponent of $m = 5$ (see (6)) is used, which is present at low stress amplitudes in the actual S-N-curve of the tower material. This case is assessed in combination with (PORFC-5R) and without (PORFC-5) the use of residue. Since especially PORFC-5R will not lead to satisfactory results, additional formulations (PORFC-2R, PORFC-2) with fatigue exponent $m = 2$ are assessed which result in quadratic cost functions, and thus are more suitable for Quadratic Programming.

Performance indicators: Considered performance indicators are revenue (analog to energy), fatigue cost (based on a realistic piecewise defined S-N-curve, see Sec. 2.1), profit (revenue subtracted by fatigue cost), pitch travel and torque travel.

4.2. Simulation results for weight variation

As shown in Sec. 3.5, the fatigue weight α_{fatigue} is a free tuning variable. In contrast to the Online-Rainflow MPC setup in [25], in the present setup a choice of $\alpha_{\text{fatigue}} = 1$ does not seem to result in maximum profit. One reason is the additional presence of non-monetary cost terms next to revenue and fatigue in the MPC cost function (7). Another reason may be the approximating nature of PORFC, where fatigue is distributed over and decoupled in time (see Sec. 3.4).

For the purpose of weight-tuning, Fig. 8 shows results for variations of fatigue weight for all MPC setups and a reference wind velocity of 7 m/s. This wind velocity was chosen, since

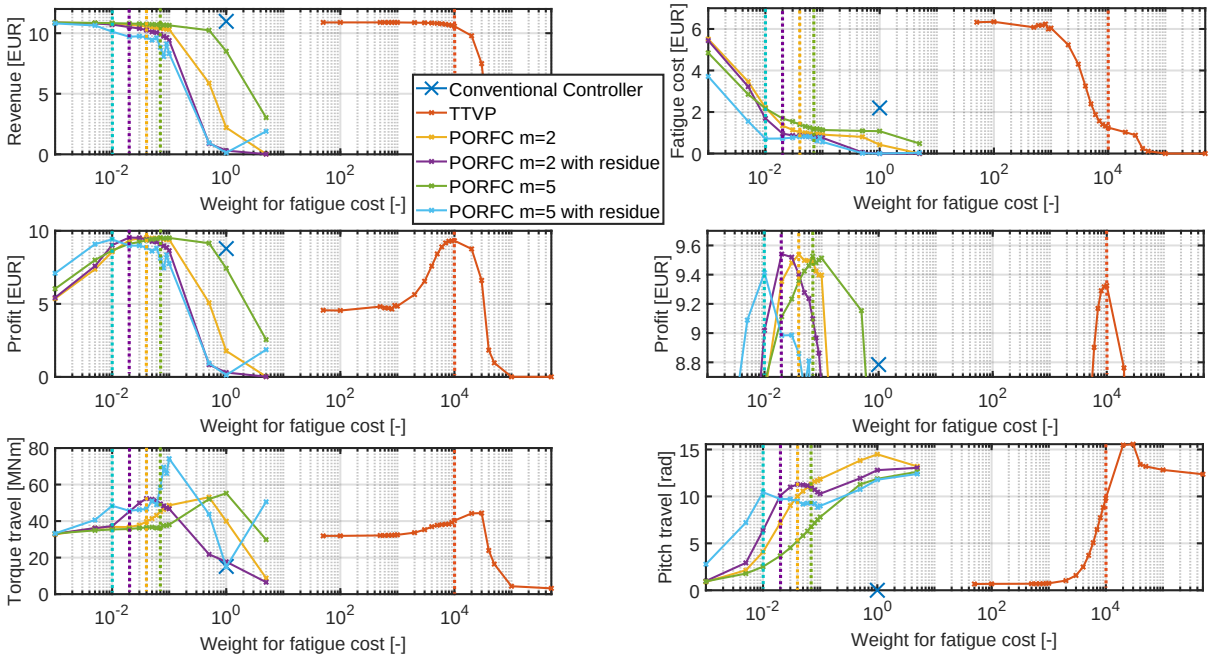


Figure 8. Variation of fatigue weights. Vertical dashed lines: location of profit-optimal weights. Middle row, right: Zoomed version of profit plot.

it is close to the mean annual wind velocity for the given hub height and an onshore site. Profit-optimality is the prime goal and achieved by the following weights: 10^4 (TTVP), $4 \cdot 10^{-2}$ (PORFC-2), $2 \cdot 10^{-2}$ (PORFC-2R), $7 \cdot 10^{-2}$ (PORFC-5) and 10^{-2} (PORFC-5R).

For very low fatigue weights, all MPCs (TTVP, PORFC) are able to achieve similar revenue that, however, is about 1% lower than the revenue of CC. The investigation of reasons for this behavior is part of ongoing research. With increasing weights, revenue is sacrificed to reduce fatigue cost. For all MPCs, the major drop in fatigue cost appears for lower weights than the drop of revenue. As a result, pronounced peaks in profit are visible. Optimum profit of TTVP is 6.2% higher than the profit of CC. The profit-peaks of PORFC-2, -2R and -5 are at almost equal levels and about 8.6% higher than the profit of CC. Unfortunately, the performance of PORFC-5R is slightly deteriorated, but still higher than the one of TTVP.

Width of the profit-peaks is suggested as an indicator for robustness of tuning. Width, e.g., can be measured by the weight range where profit is above a certain threshold, which here is defined as 95% of the individual maximum profit. In the present case, width is half an order of magnitude for TTVP, almost one order of magnitude for PORFC-2(R) and PORFC-5R, and substantially more than one order of magnitude for PORFC-5. To conclude, PORFC-5 is considered as the most robust against inaccurate tuning.

4.3. Simulation results for velocity variation

As a next step, the above-presented profit-optimal fatigue weights are fixed and the controllers are applied to a comprehensive range of reference wind velocities. The results are normalized w.r.t. TTVP and are presented in Fig. 9. Strikingly, all MPCs are extracting significantly less energy than CC at low wind velocities. Just like in [2], this is due to the MPCs' strategy of reducing fatigue via significantly decreasing rotor speed. Above rated wind velocity, the MPCs result in about 1% higher revenue than CC. An exception is PORFC-5R, which performs unreliably over the wind regime and is excluded from the remaining analysis.

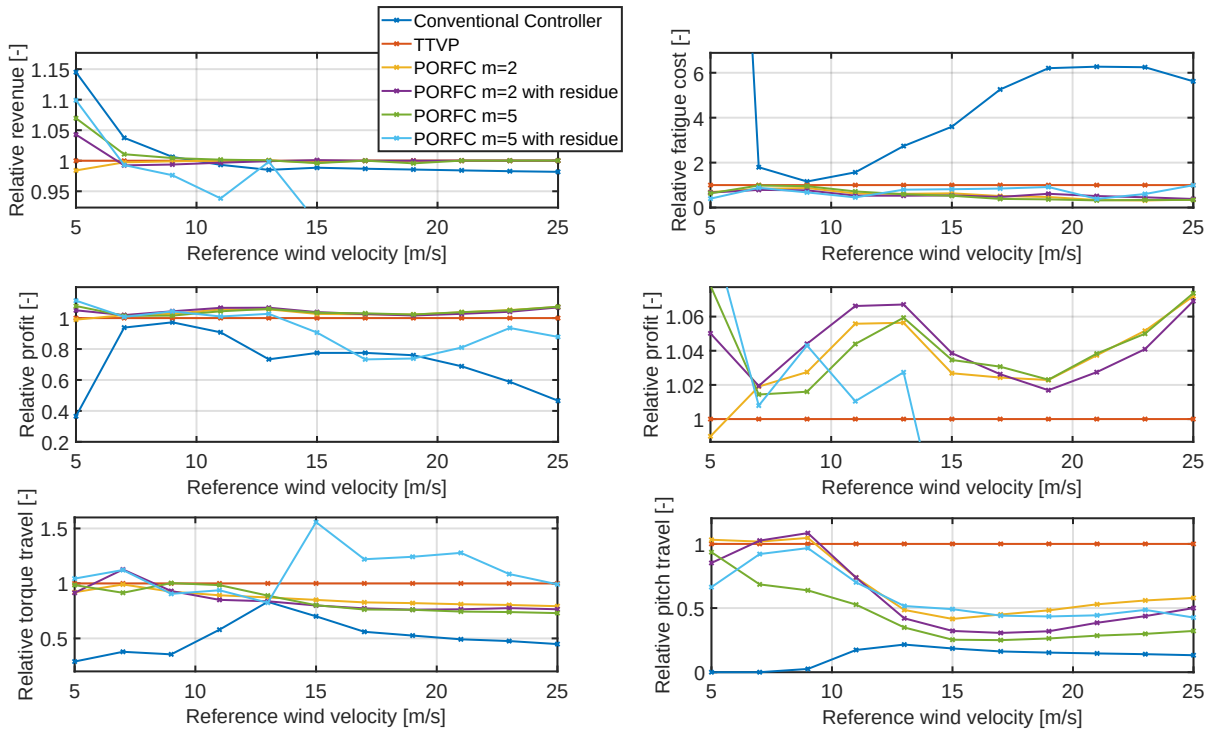


Figure 9. Variation of reference wind velocities. All resulting quantities are normalized w.r.t. the MPC using TTVP. Middle row, right: Zoomed version of profit plot.

TTVP exhibits between 14% (9m/s) and 98% (5m/s) less fatigue cost than CC. This is further improved by PORFC-5, which exhibits between 2% (7m/s) and 68% (21m/s) less fatigue cost than TTVP. As a result, the MPCs generally exhibit higher profit than CC, especially for very low wind velocity and above rated. Specifically TTVP exhibits between 2.8% (9m/s) and 174% (5m/s) more profit than CC. The advanced formulations of PORFC add further improvement, where, e.g., PORFC-5 exhibits up to 7.8% more profit than TTVP. The beneficial effect of residue can be observed from PORFC-2 to -2R, where significant profit is added for low to medium wind velocities, while pitch travel is equal or lower.

Pitch activity usually is a main concern for the control of fatigue. Indeed, for low wind velocities, the MPCs exhibit substantial pitch activity in comparison to almost none for CC. Above rated, PORFC reliefs the pitch system by about 50-75% in comparison to TTVP, and almost reaches the low levels of CC.

The deteriorated behavior of PORFC-5R cannot be fully explained yet. It may be caused by numerical difficulties, which arise from the approximation of the cost function of exponent $m = 5$ by a *Quadratic* Program in the optimization algorithm. This phenomenon may manifest itself with the use of residue, since here higher (long-term) stress amplitudes are detected which lie in a steep region of the damage function (6).

5. Conclusion & Outlook

In the present work, the MPC formulation of *Parametric Online Rainflow-counting* (PORFC) has been presented in an application-focused way. It has been highlighted how PORFC directly incorporates mechanical fatigue in predictive wind turbine control. A study on period times of tower base stress cycles has demonstrated that long observation windows are required to see cycles with sufficient damage-equivalent effect. As a solution for MPC, which typically is based

on short prediction horizons, stress history can be considered in a consistent manner by carrying along a *residue*. Aeroelastic simulations show that PORFC outperforms conventional PID- and MPC-controllers in terms of profit over the entire wind regime. Particularly above rated wind velocity, this is achieved by a moderate amount of extra pitch activity compared to the PID-controller. Furthermore, a simple test has shown that all variants of PORFC are more robust against inaccurate tuning compared to the conventional MPC formulation of TTVP.

Future research concerning PORFC will be directed to monetary cost functions for actuator usage, application with a sophisticated Lidar-simulator, and further assessment of influencing factors like MPC settings and state uncertainty.

References

[1] Bottasso C L, Pizzinelli P, Riboldi C and Tasca L 2014 *Renewable Energy* **71** 442–452 ISSN 09601481
[2] Schlipf D, Schlipf D J and Kühn M 2013 *Wind Energy* **16** 1107–1129
[3] Barradas-Berglind J J and Wisniewski R 2016 *Wind Energy* **19** 2189–2203
[4] Loew S, Obradovic D, Anand A and Szabo A Stage cost formulations of online rainflow-counting for model predictive control of fatigue *Accepted to European Control Conference 2020*
[5] Loew S and Obradovic D 2020 Formulation of fatigue dynamics as hybrid automaton for model predictive control *Accepted for IFAC World Congress 2020 2020*
[6] Barradas-Berglind J d J, Wisniewski R and Soltani M 2015 *IET Control Theory & Applications* **9** 1042–1050 ISSN 1751-8644
[7] Haibach E 2006 *Betriebsfestigkeit: Verfahren und Daten zur Bauteilberechnung* 3rd ed VDI-Buch (Berlin: Springer) ISBN 9781280618024
[8] Miner M A 1945 *Journal of Applied Mechanics* 159–164
[9] ASTM 1985 Standard practices for cycle counting in fatigue analysis
[10] The MathWorks Inc 2018 Rainflow counts for fatigue analysis
[11] Marsh G, Wignall C, Thies P R, Barltrop N, Incecik A, Venugopal V and Johanning L 2016 *International Journal of Fatigue* **82** 757–765 ISSN 0142-1123
[12] Heinrich C, Khalil M, Martynov K and WEVER U 2019 *Mechanical Systems and Signal Processing* **119** 312–327 ISSN 0888-3270
[13] Köhler M, Jenne S, Pötter K and Zenner H 2012 *Zählverfahren und Lastannahme in der Betriebsfestigkeit* (Dordrecht: Springer) ISBN 978-3-642-13163-9
[14] Grüne L and Pannek J 2017 *Nonlinear Model Predictive Control: Theory and Algorithms* 2nd ed Communications and Control Engineering (Springer International Publishing) ISBN 978-3-319-46023-9
[15] Gros S and Schild A 2017 *International Journal of Control* 1–14 ISSN 0020-7179
[16] Evans M A, Cannon M and Kouvaritakis B 2015 *IEEE Transactions on Control Systems Technology* **23** 290–296 ISSN 1063-6536
[17] Sanchez H, Escobet T, Puig V and Odgaard P F 2015 *IFAC-PapersOnLine* **48** 1363–1368 ISSN 2405-8963
[18] Luna J, Falkenberg O, Gros S and Schild A 2020 *Renewable Energy* **147** 1632–1641 ISSN 09601481
[19] Shi Y, Xu B, Tan Y and Zhang B A convex cycle-based degradation model for battery energy storage planning and operation
[20] Anand A 2020 *Optimal Control of Battery Energy Storage System for Grid Integration of Wind Turbines* Master thesis TU Munich Munich
[21] Löw S and Obradovic D 2018 *atp edition* **60** 46–53 ISSN 2364-3137
[22] Gros S 2013 An economic nmpc formulation for wind turbine control *52nd IEEE Conference on Decision and Control* pp 1001–1006
[23] Jonkman J, Butterfield S, Musial W and Scott G 2009 *Definition of a 5-MW Reference Wind Turbine for Offshore System Development: Technical Report NREL/TP-500-38060* (National Renewable Energy Laboratory)
[24] Robin Verschueren, Gianluca Frison, Dimitris Kouzoupis, Niels van Duijkeren, Andrea Zanelli, Branimir Novoselnik, Jonathan Frey, Thivaharan Albin, Rien Quirynen and Moritz Diehl 2019 acados: a modular open-source framework for fast embedded optimal control arXiv
[25] Loew S, Obradovic D and Bottasso C L 2019 Direct online rainflow-counting and indirect fatigue penalization methods for model predictive control *2019 18th European Control Conference (ECC)* (IEEE) pp 3371–3376 ISBN 978-3-907144-00-8