2018 European Control Conference (ECC)
June 12-15, 2018. Limassol, Cyprus

A Sparse ANN-based Fatigue Estimation for Wind Turbine Control based on NMPC

Julio Luna[1,*], *Member, IEEE*, Sébastien Gros[1], Ole Falkenberg[2] and Axel Schild[2]

***Abstract*— In this paper, an Artificial Neural Network (ANN) methodology for estimating the tower fatigue of a wind turbine (WT) is developed. Specifically, the fatigue is estimated using an ANN that receives frequency-domain measurements of the tower base position and velocity. The frequencies are selected using an algorithm that detects the most relevant values and generates a sampling grid. The complexity of the ANN-based fatigue estimation is analysed to study the viability of its deployment in a real-time Nonlinear Model Predictive Control (NMPC) formulation.**

I. Introduction

Renewable electricity generation from wind turbines (WTs) is one of the fastest expanding renewable energy sources over the last years, with over 486 GW installed all over the world at the end of 2016, as reported by the Global Wind Energy Council (GWEC) [1]. Wind energy represents the 51% of Europe electricity generating capacity [1].

WTs are exposed to variable winds, rotor loads and other structural and mechanical effects. These loads and their interaction affect WTs during their normal operation. WTs can last for decades by actively reducing the structural and actuator fatigue. In recent years, fatigue modelling for control purposes has considerably grown in importance [2], [3] with the purpose of further extending the lifetime of WTs,

In the literature, Nonlinear Model Predictive Control (NMPC) and Economic NMPC (ENMPC) are rising as the most appropriate candidates for closed-loop control of very large WTs. NMPC is capable of handling the complex and highly nonlinear dynamics of WTs and the constraints present in the system. ENMPC tends to have superior performance than classic NMPC, especially in the presence of transients [4] because it handles the operating cost in the objective function directly [5]. There is already an important work focusing on NMPC and ENMPC for WT control (WTC). The main control objectives that can be found in the literature are: power maximisation [6], cost reduction [7] and fatigue load reduction [8] among others.

In MPC schemes, fatigue is usually handled with L_2 norms [9]. While it has been proven to be a viable control objective, it is not feasible to deploy it in real-time strategies due to the dense nature of the Hessian matrices fed into the optimisation problem. In this paper, we propose a feature projection method that preserves the sparse structure of matrices for implementation in real-time optimisation problems.

To successfully deploy a real-time NMPC/ENMPC algorithm for tower fatigue reduction, reliable short-term fatigue estimation is needed in the control scheme. A common method to estimate the damage accumulation produced by cyclic loading is the combination of the Rainflow Counting (RFC) algorithm [10] and a damage-equivalent load approach. However, these methods are non-smooth, highly non-convex, and computationally expensive, making them inadequate for deployment in the context of real-time NMPC. To address this issue, we propose here to use Artificial Neural Networks (ANNs) to produce estimations of the RFC algorithm, and use the ANN output as a penalty in the cost function of the NMPC control scheme.

Because they are generic function approximators, ANNs can arguably be used as a surrogate to the RFC [11]. In the context of NMPC, the ANN is meant to provide a good approximation of the damage progression resulting from the loads appearing in the state predictions delivered by the model underlying the NMPC scheme. Hence the ANN ought to take as an input the load time series predicted within the NMPC prediction horizon and delivers an estimation of the damage progression. Unfortunately, when using the load time series predicted within the NMPC prediction horizon as an input to the ANN, the resulting fatigue estimator becomes a function of that entire time series. Hence the Hessian of the ANN output with respect to the state prediction formed inside the NMPC scheme becomes dense, which is highly detrimental for a real-time deployment of the NMPC scheme.

This paper proposes an alternative based on linear feature extraction operating the load time series, which allows one to recover a favourable structure in the NMPC scheme, at the expense of introducing more states in the underlying NMPC model. We argue that this alternative is better suited for a real-time deployment of the NMPC scheme than an ANN fatigue estimator based on time series. In this paper, we test a feature extraction based on different basis of frequency functions. The proposed methodology is tested on data stemming from simulations of the National Renewable Energy Laboratory (NREL) [12] 5MW wind turbine.

The paper is organised as follows. In Section II, the general problem description and preliminaries are introduced. Section III presents the elements to consider when deploying real-time NMPC schemes for WTs. The linear feature extractor is detailed in Section IV. Section V presents the proposed ANN for fatigue estimation. Simulation results are shown in Section VI. Finally, Section VII summarises the

[1]J. Luna and S. Gros are with the Electrical Engineering Department, Chalmers University of Technology, Hörsalsvägen 11, SE-412 96, Göteborg, Sweden.
[2] O. Falkenberg and A. Schild are with IAV, GmbH, Rockwellstrasse, Gifhorn, Germany.
∗ Corresponding author: julio.luna@chalmers.se

conclusions of this paper and research lines for future work are proposed.

II. Preliminaries

In this section, the fundamental concepts for fatigue computation using counting methods are briefly introduced. Moreover, the WT dataset that will be used for the ANN training is presented.

A. Fatigue damage estimation

Fatigue is the weakening of a material subject to stress, specially the cyclic ones. Even with nominal maximum stress values much lower than the ultimate stress limit of the material, cycling loading and fatigue can lead to the propagation of cracks and eventually, to the fracture of the structure.

While the amount of fatigue that a material has suffered is complex to calculate, there are widely used methods for fatigue estimation in the industry. In the case of WTs, the International Standard IEC 61400-1 [13], suggests the use of the RFC counting method, combined with the Palmgren-Miner linear damage accumulation rule.

Given time series signal u, the material ultimate design load σ_u and the mean stress $|\sigma|$, the RFC algorithm is computed as follows:

$$[\Delta\sigma_i, n_i] = \mathrm{RFC}(u(t), \sigma_u, |\sigma|), \tag{1}$$

where $\Delta\sigma_i$ is the stress cycle ranges amplitude, n_i is the number of cycles with amplitude $\Delta\sigma_i$ and RFC is the counting method function, which in this paper is the MLife algorithm from the NREL [12]. After applying the RFC algorithm, the maximum number of cycles at the i-th stress (N_i) that the material can tolerate before failure is obtained

$$N_i = \left(\frac{\sigma_u - |\sigma|}{0.5\gamma\Delta\sigma_i}\right)^m, \tag{2}$$

where γ is a safety factor and m the S-N curve slope [14].

With Equation (2), the estimated damage fraction D is obtained using the Palmgren-Miner rule

$$D = \sum_i \frac{n_i}{N_i} \in [0, 1]. \tag{3}$$

Including Equation (3) in a NMPC scheme would add the fatigue reduction as a control objective. However, D is a function of the time series signal u. Directly using Equation (3) would generate a dense Hessian matrix, thus state projection and feature extraction methods are needed prior to include fatigue reduction as a control objective in real-time NMPC.

B. Tower base data

The fatigue estimation methodology proposed in this work is going to be analysed for the tower base, one of the most critical points for the system fatigue. However, the proposed methodology will be extended in the future to the rest of the WT (blades, gearbox, etc.).

Figure 1 represents the wind speed profile W and the tower base position (x) and velocity ($\dot{x}$) along the perpendicular axis to the wind. The data has been extracted from the WT simulator FAST [12]. The WT is a NREL 5 MW reference turbine, the average wind speed for the dataset is 11 m/s and the total simulation time is 3500 seconds.

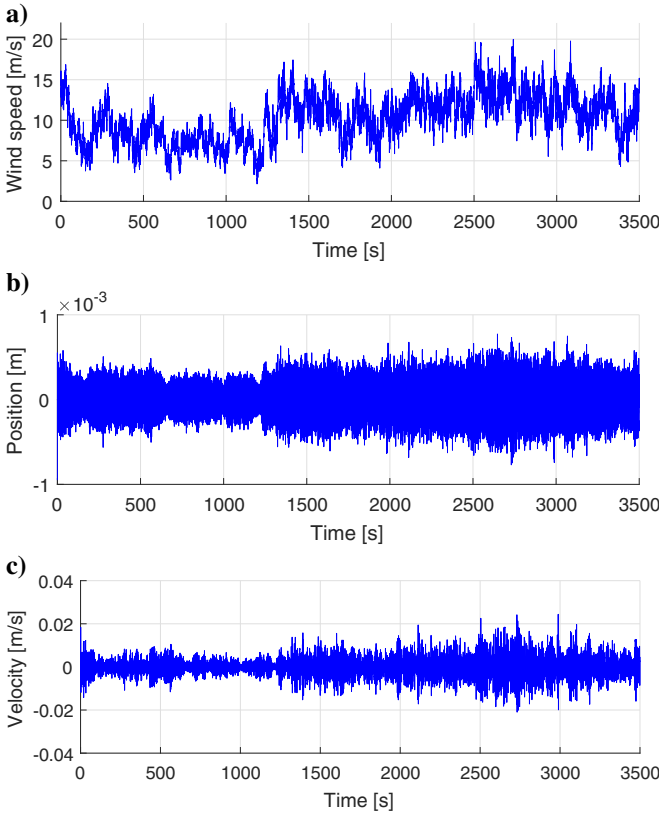

Fig. 1. Wind speed profile with an average value of 11 m/s (a), time domain response for x (b) and $\dot{x}$ (c)

III. Real-time NMPC scheme

Real-time NMPC for WTs is based on solving at every discrete time t_i a Nonlinear Program of the form [4]:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & \varphi\left(\mathbf{x}_N\right) + \sum_{k=0}^{N-1} L\left(\mathbf{x}_k, \mathbf{u}_k\right), \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{F}(\mathbf{x}_k, \mathbf{u}_k, W_k), \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \\ & \mathbf{x}_0 = \hat{\mathbf{x}}(t_i), \end{aligned} \tag{4}$$

where $\hat{\mathbf{x}}(t_i)$ is an estimation of the state of the WT at the physical time t_i, N is the prediction horizon, $\mathbf{x}_k, \mathbf{u}_k$ are an ad-hoc discretisation of the state and input trajectories of the WT, and W_k is a sequence of estimated wind speed on the prediction horizon of the NMPC scheme. Function $\mathbf{F}$ is a numerical integration method deployed on the WT model, and function $\mathbf{h}$ is used to enforce the physical limitations of the WT in the NMPC scheme. The stage cost functions L and φ are the penalties to be minimised, typically holding

penalty terms regarding power capture maximisation, fatigue penalties, and penalties on the use of the actuators [15].

Solution approaches to tackle Problem (4) in real-time are typically based on SQP-type methods or on the Real-Time Iteration method [16]. Both rely on solving at every time instant t_i at least one instance of the Quadratic Program (QP)

$$\begin{aligned}\min_{\Delta\mathbf{x},\Delta\mathbf{u}} \quad & \frac{1}{2}\Delta\mathbf{x}_N^\top \mathcal{T}\Delta\mathbf{x}_N + \mathbf{f}^\top\mathbf{x}_N \\ & + \frac{1}{2}\sum_{k=0}^{N-1}\begin{bmatrix}\Delta\mathbf{x}_k\\ \Delta\mathbf{u}_k\end{bmatrix}^\top \mathcal{H}_k \begin{bmatrix}\Delta\mathbf{x}_k\\ \Delta\mathbf{u}_k\end{bmatrix} + \mathbf{f}_k^\top\begin{bmatrix}\Delta\mathbf{x}_k\\ \Delta\mathbf{u}_k\end{bmatrix}, \\ \text{s.t.} \quad & \mathbf{x}_{k+1}+\Delta\mathbf{x}_{k+1} = \mathbf{F} + \frac{\partial\mathbf{F}}{\partial\mathbf{x}_k}\Delta\mathbf{x}_k + \frac{\partial\mathbf{F}}{\partial\mathbf{u}_k}\Delta\mathbf{u}_k\Big|_{\mathbf{x}_k,\mathbf{u}_k}, \\ & \mathbf{h} + \frac{\partial\mathbf{h}}{\partial\mathbf{x}_k}\Delta\mathbf{x}_k + \frac{\partial\mathbf{h}}{\partial\mathbf{u}_k}\Delta\mathbf{u}_k\Big|_{\mathbf{x}_k,\mathbf{u}_k} \le 0,\end{aligned} \tag{5}$$

where matrices $\mathcal{H}_k$ and $\mathcal{T}$ are typically ad-hoc approximations of the block-diagonal Hessian of the Lagrange function $\mathcal{L}$ associated to the Nonlinear Program (4), i.e. [17]

$$\nabla^2\mathcal{L} \approx \begin{bmatrix}\mathcal{H}_0 & & & & \\ & \mathcal{H}_1 & & & \\ & & \ddots & & \\ & & & \mathcal{H}_{N-1} & \\ & & & & \mathcal{T}\end{bmatrix}. \tag{6}$$

Functions $\mathbf{f}_k$ are the gradient of the stage cost function $L(\mathbf{x}_k, \mathbf{u}_k)$ at the different stages k, and function $\mathbf{f}_N$ is the gradient of the terminal cost function φ. Highly efficient methods are available to solve QP in the specific form (5) in real time [18], [19]. Generic QP solvers capable of tackling QP problems in a more general form are typically significantly slower, and not adequate for a real-time deployment of NMPC.

Fatigue is typically accounted for in the NMPC scheme in the form of penalties in the cost function. E.g. for introducing the fatigue of the tower base resulting from the tower fore-aft motion, it is very common to introduce a (weighted) term $\dot{x}^2$ in the stage cost L, where $\dot{x}$ is the tower top velocity [20]. While this approach is very well suited for real-time NMPC, and allows one to cast the problem in the specific structure (4), the use of a simple quadratic function as a fatigue estimation is debatable.

Assuming the fully observable state sequence $\{\mathbf{x}_0, \ldots, \mathbf{x}_N\}$, the WT damage progression over a prediction horizon $k = \{0, \ldots, N\}$ needs to be computed as

$$D_N = D_0 + f_\mathrm{t}(\mathbf{x}_0, \ldots, \mathbf{x}_N), \tag{7a}$$
$$D_0 = D(t_i), \tag{7b}$$

where D_0 is the initial damage value at physical time t_i and f_t an adequate function that computes an approximate damage accumulation from the state sequence $\{\mathbf{x}_0, \ldots, \mathbf{x}_N\}$.

Unfortunately, if one introduces D_N from Equation (7) in the cost function of Problem (4), then the sparsity pattern of the Hessian (6) loses its block-diagonal structure because the function f_t depends on the entire state sequence $\{\mathbf{x}_0, \ldots, \mathbf{x}_N\}$. Losing the block-diagonal structure of the Hessian (6) typically yields a dramatic increase of the computational time required to solve the QP in Problem (5).

In order to address this problem we propose to use a feature extraction in the form

$$\phi(\dot{\mathbf{x}}(\cdot), t) = \int_0^t \theta(\tau)\dot{x}(\tau)d\tau, \tag{8}$$

where $\theta(\cdot)$ is a set of ad-hoc basis functions. Equivalently, the feature functions can be described as:

$$\dot{\phi}(\dot{\mathbf{x}}(\cdot), t) = \theta(t)\dot{x}(t), \quad \phi(\dot{\mathbf{x}}(\cdot), 0) = 0 \tag{9}$$

We then propose to use the ANN to approximates the fatigue progression as:

$$\dot{D}(t) = \mathrm{ANN}\left(\phi(\dot{\mathbf{x}}(\cdot), t),\, \dot{\phi}(\dot{\mathbf{x}}(\cdot), t)\right), \quad D(0) = 0 \tag{10}$$

In order to use this fatigue approximation in the NMPC scheme, we can introduce the dynamics (9)-(10) in the continuous model underlying the NMPC scheme, and the time derivative of the fatigue $\dot{D}(t)$ as a Lagrange term [17] in the cost of the NMPC scheme.

In practice, the continuous dynamics (9)-(10) are treated via the numerical methods used to discretise the continous WT model forming the discrete Problem (4), and the the continuous Lagrange cost term $\dot{D}(t)$ is introduced in the stage cost L.

IV. Frequency data projection

In this section, the WT sensor data is projected into a time-frequency domain to decouple the data from the original state sequence. Moreover, a frequency selection algorithm is presented to reduce the state space dimension while maintaining the most relevant frequency features.

A. Time-frequency domain feature extraction

In this paper, we will investigate the use of $\cos(\cdot)$ and $\sin(\cdot)$ as basis functions to build the feature function $\theta(\cdot)$ from Equation (10), i.e.

$$\begin{aligned}\theta(\tau) \quad = \quad & [\cos(\omega_1\tau), \sin(\omega_1\tau), \ldots, \\ & \cos(\omega_n\tau), \sin(\omega_n\tau)] \in \mathbb{R}^{2n},\end{aligned} \tag{11}$$

where $\{\omega_1, \ldots, \omega n\}$ rad/s are the radial frequency used to build the basis functions. The most relevant frequencies for the signal fall in the range $[0.1,\ 0.5]$ Hz. We therefore select the frequencies ω_i in the range $[0.6, 3.15]$ rad/s. In order to avoid adding a large number of states in the NMPC model (see Section III), it is beneficial to select as few basis functions as possible, and therefore as few frequencies ω_i as possible. It is therefore useful to select the frequencies ω_i carefully.

B. Automatic frequency selection

In this section we propose a heuristic algorithm to select the frequencies ω_i automatically. To select the most relevant frequencies, an algorithm is developed to generate a grid of n frequencies. The algorithm is based on identifying the p larger peaks in the power spectrum density of the load signal, and attributes to each peak a number of n frequencies ω_i. The total number of p peaks to extract is defined before executing the algorithm.

The pseudo-code is included in Algorithm 1. First, the frequency response is represented as the upper envelope function, shown in Figure 2(a). From this function, the peaks are extracted automatically. After the envelope peaks are extracted, a grid of n sampling points is automatically generated between each peak and the boundary valley points. Figure 2(b) and (c) shows the grid selection results for 3 peaks with 128 and 24 sampling points respectively.

Algorithm 1: Proposed frequency selection algorithm

```
function Frequency_Selection (|P1(f)|, f, p, n);
  Input  : Power density spectrum |P1(f)|, frequency f,
           number of peaks p and sampling points n
  Output: f_W(|P1(f)|, f, p, n)
Envelope function generation;
Extraction of p maximum peaks;
for i ← 1 to p do
    Find neighbourhood boundary valleys v_l and v_r;
    for j ← 1 to n do
        j-samples grid generation for the frequency
        interval [v_l − i − v_r];
    end
end
```

In Section VI, the results are analysed for 5 different ANNs. Each one is trained with $f_W = [128, 64, 32, 16, 8]$, obtained for 3 peaks using Algorithm 1. The results are compared for each f_W against a validation set. The selected frequencies are used in Equation (10) to obtain the time-frequency response of the temporal data.

V. Fatigue estimation using ANNs

In this section the ANN training and validation methodology is presented along with the data processing procedure and ANN structure.

A. Data processing

To train and evaluate the ANNs the temporal and time-frequency data has to be correctly pre-processed to obtain training ($\mathcal{G} \in \mathbb{R}^{tr \times r}$) and testing ($\mathcal{S} \in \mathbb{R}^{s \times r}$) sets, where tr represents the number of training samples, s the number of testing samples and r the number of wind cases used for the training and testing of the ANN. The general overview of this process is shown in Figure 3.

To obtain the training and testing sets, first the instantaneous damage data $\dot{D}$ is obtained off-line from the RFC algorithm, then it is filtered using a simple moving average

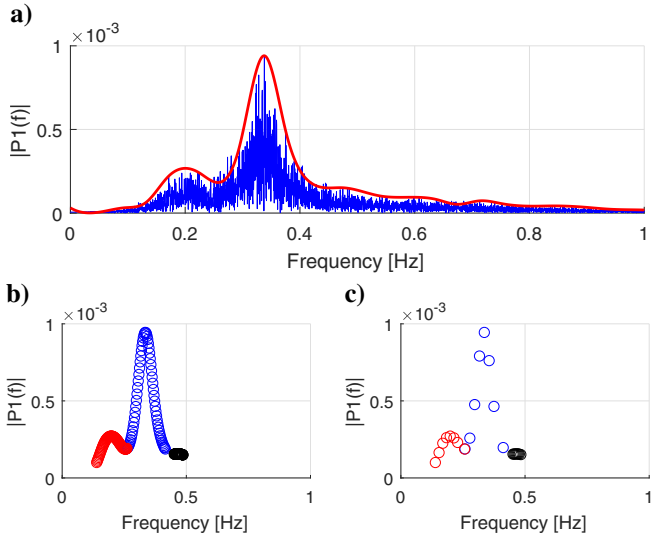

Fig. 2. Frequency domain of $\dot{x}$ with upper envelope function (a), 128 sampling points selection grid (b) and 24 sampling points selection grid (c) for 3 peaks (red, blue and black in the bottom figures)

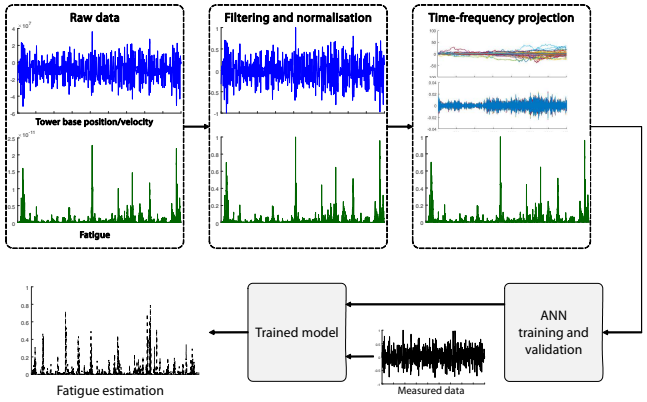

Fig. 3. Data pre-processing, ANN training and fatigue estimation methodology

(SMA) approach to reduce the noise originated by the RFC algorithm. The filtering stage is needed due to possible numerical errors resulting from the computation $\dot{D}$ in order to avoid false trends when training the ANNs.

Once the damage data is filtered, the signals are scaled. Since fatigue data is several orders of magnitude lower than the tower position and velocity data, it is necessary to standardise the variables so they can be compared when training the ANN. In particular, the tower base velocity is restricted to $\dot{x} \in [-1, 1]$ and the instantaneous damage data to $\dot{D} \in [0, 1]$. In the scope of this work, the scaling is performed taking into consideration the maximum and minimum values of all the $\dot{x}$ and $\dot{D}$ datasets respectively. When implementing the proposed methodology in a closed-loop scheme, the physical limits of the signals will be considered when performing the data scaling.

Finally, when the signals are scaled, the data is split between training $\mathcal{G}$ and testing sets $\mathcal{S}$. In Section VI a 80-20% data split is done to train and validate the ANNs.

B. ANN structure

ANNs can be considered as black-box models. Nevertheless, it is possible to configure their internal structure to test different simulation cases. ANNs are organised in different layers: input, hidden and output. The optimal tuning of the ANN structure is out of the scope of this paper. A standard trial-and-error procedure based on several simulation cases has been carried out to select the most suitable number of hidden layers and neurons. While in Section VI, only the results for one ANN structure are shown, during the development of this work multiple simulations have been studied until obtaining the most satisfactory results.

The proposed ANN has two inputs: the time-frequency projection ϕ_h and its derivative $\dot{\phi}_h$, represented by Equation (10). The network has only one output, the expected instantaneous damage $\hat{\dot{D}}$. The designed ANN consists of 3 hidden layers with 4, 18 and 8 neurons respectively. The values of the weights connecting the input, hidden and output layers are set using the training set $\mathcal{G}$ and the training considerations in Section VI-A. The activation functions for the hidden layers are *tansig*, *logsig* and *tansig* respectively.

VI. Results

The ANN from Section V is trained and evaluated for fatigue estimation using the FAST temporal data from Figure 1. MATLAB R2016b (64 bits) is used to train and test the model. The ANN structure is defined in Section V-B.

A. Training and estimation considerations

The training process has been carried out on a PC Intel Core i5-6500 @ 3.20 GHz with 16 GB of RAM with training and validation sets of 80% and 20% of the data respectively.

As mentioned in Section IV-B, a selection of frequencies is used in order to study the effect of reducing the number of frequencies to perform the state projection. In particular, the set frequencies is obtained from Algorithm 1 are $f_W = [128, 64, 32, 16, 8]$ for a total of 3 peaks in the envelope function. For each value in f_W, a different ANN is trained.

To analyse the results, the ANN estimation is compared to a test baseline, obtained with the RFC algorithm with the following parameters: $m = 4$, $\sigma_u = 20 \times 10^{\frac{\log(15.117)}{m}}$. The results are shown and discussed in the following section.

B. Fatigue estimation evaluation

Figure 4 shows the accumulated damage D_{Acc} obtained from the RFC algorithm, the most common tower fatigue estimation used in the literature, which is the L^2 norm of the tower bending velocity $||\dot{x}||^2$ [21], and the estimated $\hat{D}_{Acc}$ for the different trained ANNs. As shown, all of the selected frequency points provide satisfactory results with minor disparities with respect to the test data. Meanwhile, using the L^2 norm of $\dot{x}$ does not guarantee good results when there are extreme loads that cause big jumps in the fatigue, like between $t = 2980s$ and $t = 3090s$. To quantify the improvement of the proposed methodology versus the classic baseline approach, Figure 5 shows three metrics to compare the results with the test set from the RFC: L^1, L^2 and L^∞ norms.

Figure 5 clearly shows how the proposed strategy beats the classic baseline approach in all of the analysed metrics. Regarding the ANN methodology, the best results are obtained when the input signal uses dense frequency grids. Major differences between the test data and the estimations are observed in Figure 4 at the last stages of the simulation. However, these differences are a consequence of the accumulated error during the simulation. The use of additional features (i.e. $min - max$ functions, other state projection functions, etc.) or improved ANN structures could solve these issues by enriching the input dataset when training the ANNs.

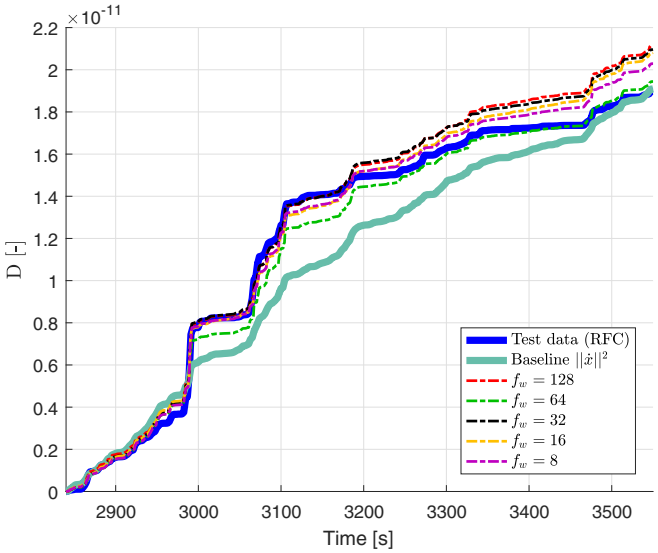

Fig. 4. Accumulated damage D_{Acc} validation set, baseline fatigue $||\dot{x}||^2$ and estimation $\hat{D}_{Acc}(f_W)$

Of special interest is that in some cases, lower values of f_W perform better than the higher ones. This is the case of $f_W = 64$ and $f_W = 8$. The reason behind this is because the selection Algorithm 1 can designate frequencies that are more relevant in some cases, even with lower frequency sampling points. To solve this, better frequency selection procedures will be implemented in the future.

Moreover, the results in Figure 4 indicate that low sampling values are useful to estimate the tower fatigue. This is beneficial for deploying real-time NMPC schemes with fatigue reduction in the objective function: a good prediction with a low selection of frequencies means that the fatigue penalty in Problem (4) will hold a smaller state space size and thus, the computation time to solve the QPs will not increase dramatically.

VII. Conclusions

A methodology to estimate fatigue in WTs using ANNs has been developed. The evaluation of the proposed strategy shows good performance for different trained ANNs. The sensor data has been projected in a time-frequency domain, aiming at eliminating the state dependency of the

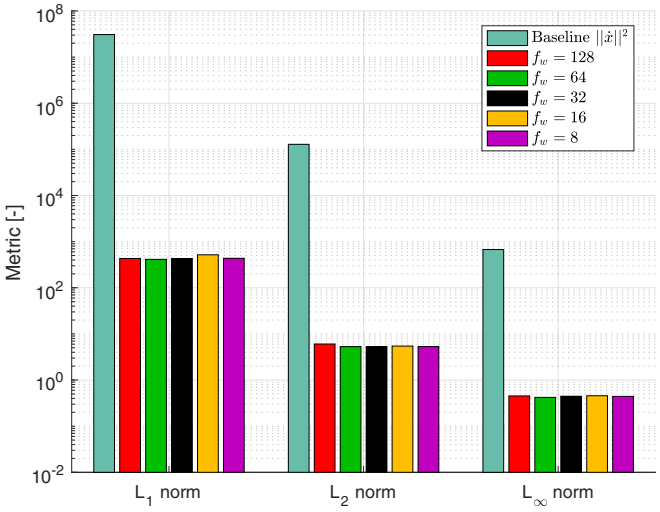

Fig. 5. Metrics for the baseline validation set, baseline fatigue $||\ddot{x}||^2$ and estimation $\hat{D}_{Acc}(f_W)$

fatigue when deploying control schemes. Using the proposed time-frequency projection of the temporal data, the damage estimation is decoupled from the original state and input trajectories. This projection, while increasing the size of the state space, preserves the structure of the problem, allowing to maintain sparse block-diagram Hessian matrices in the real-time NMPC problem formulation.

While the proposed feature projection has shown promising results, it remains to be seen if the frequency domain is the most appropriate feature for fatigue prediction. Other features will be studied in the future, always maintaining the structure for real-time implementation in mind.

In Section IV-B, an algorithm to automatically select the frequencies for the feature projection has been presented. This approach has been shown to be useful to select a reduced amount of frequencies when training the ANNs. Future work includes the improvement of the algorithm, specifically to better select the most relevant frequencies even with low selection values.

To test the ANNs, long predictions have been investigated. However, when implementing the proposed methodology in real-time NMPC schemes, the estimation has to be done over short-time prediction horizons (≈10-20 seconds), it remains to be seen if the ANN approach has advantages over other fatigue estimation approaches. Future work will aim to test the proposed method in a real-time NMPC scheme with fatigue reduction capabilities and compare it with other fatigue estimation solutions.

ACKNOWLEDGMENT

This research has been funded by the German Ministry of Economic Affairs and Energy under the grant 0324125A.

REFERENCES

[1] "Global wind energy council," http://gwec.net/global-figures/wind-in-numbers/.

[2] K. Selvam, S. Kanev, J. W. van Wingerden, T. van Engelen, and M. Verhaegen, "Feedback–feedforward individual pitch control for wind turbine load reduction," *International Journal of Robust and Nonlinear Control*, vol. 19, no. 1, pp. 72–91, 2009.

[3] D. Song, J. Yang, M. Dong, and Y. H. Joo, "Model predictive control with finite control set for variable-speed wind turbines," *Energy*, vol. 126, pp. 564–572, 2017.

[4] S. Gros and A. Schild, "Real-time economic nonlinear model predictive control for wind turbine control," *International Journal of Control*, pp. 1–14, 2017.

[5] J. B. Rawlings, D. Angeli, and C. N. Bates, "Fundamentals of economic model predictive control," in *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*. IEEE, 2012, pp. 3851–3861.

[6] P. Gebraad, J. J. Thomas, A. Ning, P. Fleming, and K. Dykes, "Maximization of the annual energy production of wind power plants by optimization of layout and yaw-based wake control," *Wind Energy*, vol. 20, no. 1, pp. 97–107, 2017.

[7] S. Sichilalu, T. Mathaba, and X. Xia, "Optimal control of a wind–pv-hybrid powered heat pump water heater," *Applied energy*, vol. 185, pp. 1173–1184, 2017.

[8] K. Hammerum, P. Brath, and N. K. Poulsen, "A fatigue approach to wind turbine control," in *Journal of Physics: Conference Series*, vol. 75, no. 1. IOP Publishing, 2007, p. 012081.

[9] J. de Jesus Barradas-Berglind, R. Wisniewski, and M. Soltani, "Fatigue damage estimation and data-based control for wind turbines," *IET Control Theory & Applications*, vol. 9, no. 7, pp. 1042–1050, 2015.

[10] S. D. Downing and D. Socie, "Simple rainflow counting algorithms," *International journal of fatigue*, vol. 4, no. 1, pp. 31–40, 1982.

[11] J. Klemenc and M. Fajdiga, "A neural network approach to the simulation of load histories by considering the influence of a sequence of rainflow load cycles," *International journal of fatigue*, vol. 24, no. 11, pp. 1109–1125, 2002.

[12] J. M. Jonkman and M. L. Buhl Jr, "Fast user's guide-updated august 2005," National Renewable Energy Laboratory (NREL), Golden, CO., Tech. Rep., 2005.

[13] I. E. Commission *et al.*, "Iec 61400-1: Wind turbines part 1: Design requirements," *International Electrotechnical Commission*, 2005.

[14] G. Schott, B. Donat, and M. Schaper, "The consecutive wöhler curve approach to damage accumulation," *Fatigue & Fracture of Engineering Materials & Structures*, vol. 19, no. 2-3, pp. 373–385, 1996.

[15] M. Zanon, S. Gros, and M. Diehl, "Indefinite Linear MPC and Approximated Economic MPC for Nonlinear Systems," *Journal of Process Control*, vol. 24, pp. 1273–1281, 2014.

[16] B. Houska, H. J. Ferreau, and M. Diehl, "An Auto-Generated Real-Time Iteration Algorithm for Nonlinear MPC in the Microsecond Range," *Automatica*, vol. 47, no. 10, pp. 2279–2285, 2011.

[17] L. T. Biegler, *Nonlinear programming: concepts, algorithms, and applications to chemical processes*. Siam, 2010, vol. 10.

[18] M. Diehl, H. Bock, and J. Schlöder, "A real-time iteration scheme for nonlinear optimization in optimal feedback control," *SIAM Journal on Control and Optimization*, vol. 43, no. 5, pp. 1714–1736, 2005. [Online]. Available: http://epubs.siam.org/sicon/resource/1/sjcodc/v43/i5/p1714_s1

[19] A. Domahidi and J. Perez, "FORCES Professional," embotech GmbH (http://embotech.com/FORCES-Pro), July 2014.

[20] S. Raach, D. Schlipf, F. Sandner, D. Matha, and P. W. Cheng, "Nonlinear model predictive control of floating wind turbines with individual pitch control," in *American Control Conference (ACC), 2014*. IEEE, 2014, pp. 4434–4439.

[21] M. Blackwell, O. R. Tutty, E. Rogers, and R. Sandberg, "Computational fluid dynamics based iterative learning control for smart rotor enabled fatigue load reduction in wind turbines," in *American Control Conference (ACC), 2014*. IEEE, 2014, pp. 4446–4451.