POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

EXECUTIVE SUMMARY OF THE THESIS

Down-scaling of a Wind Turbine Test-Bench and Optimal Sensor Placement

TESI MAGISTRALE IN AERONAUTICAL ENGINEERING

AUTHOR: GREGORY COTTONE

ADVISOR: PIERANGELO MASARATI

ACADEMIC YEAR: 2021-2022

1. Introduction

This thesis aims to address the challenge of testing wind turbine structural integrity over their lifespan. Creating scaled test benches is essential due to the impracticality of replicating the full-scale turbines. However, scaling introduces non-linearity in load estimation, necessitating a method for accurate load estimation. Current practices rely on costly, intrusive physical measurements, prompting the exploration of a virtual sensing strategy. The Kalman filter, specifically the Augmented Extended Kalman Filter (AEKF), is proposed for state-input-parameter estimation. Virtual prototypes, coupled with strain measurements, help reduce development costs. Additionally, optimal sensor placement is crucial for stable estimation, and an Optimal Sensor Placement (OSP) strategy is introduced for joint estimation, building upon prior work.

2. Augmented Extended Kalman Filter

It is possible to estimate the states $\boldsymbol{x}$ and the unknown inputs $\boldsymbol{u}^{UK}$ simultaneously using a Kalman filter, simply considering the inputs as if they were new states and adding them to the state vector itself, so generating the augmented state vector, [2], [4], [5], [18], [19]:

$$\boldsymbol{x}^* = \begin{bmatrix} \boldsymbol{x} \\ \boldsymbol{u}^{UK} \end{bmatrix}$$

In order to define the Augmented Kalman Filter, the dynamics of the input has also to be modeled, so a zeroth-order random walk model for the vector of unknown inputs is introduced:

$$\boldsymbol{u}_{k+1}^{UK} = \boldsymbol{u}_k^{UK} + \boldsymbol{w}_k^{UK}$$

where $\boldsymbol{w}_k^{UK}$ k is a white, zero mean, uncorrelated random process with the associate covariance matrix $\boldsymbol{Q}_u$. The selection of $\boldsymbol{Q}_u$ is done using the same approach used by Cumbo et al. [4]:

$$\boldsymbol{Q}_u = (\Delta t \cdot \bar{\omega}_u \cdot \bar{a}_u)^2$$

Where $\bar{\omega}_u$ and $\bar{a}_u$ represent indicative values chosen by the user and have the same order of magnitude of the input frequency and amplitude respectively. Clearly, the matrices that characterize the dynamic system in the form of a discretized state will be modified as follows:

$$\boldsymbol{A}^* = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B}_u \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix}$$

$$\boldsymbol{H}^* = [\boldsymbol{C} \quad \boldsymbol{D}]$$

Usually the output does not depend on the input, so $\boldsymbol{D} = \boldsymbol{0}$.

$$\begin{cases} \begin{bmatrix} \boldsymbol{x}_{k+1} \\ \boldsymbol{u}_{k+1}^{UK} \end{bmatrix} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B}_u \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}_k \\ \boldsymbol{u}_k^{UK} \end{bmatrix} + \begin{bmatrix} \boldsymbol{w}_k \\ \boldsymbol{w}_k^{UK} \end{bmatrix} \\ \boldsymbol{y}_k = [\boldsymbol{C} \quad \boldsymbol{D}] \begin{bmatrix} \boldsymbol{x}_k \\ \boldsymbol{u}_k^{UK} \end{bmatrix} + \boldsymbol{v}_k \end{cases}$$

Where $\boldsymbol{w}_k$ is the Gaussian process noise and $\boldsymbol{v}_k$ is the Gaussian measurement noise. Written in compact form:

$$\begin{cases} \boldsymbol{x}_{k+1}^* = \boldsymbol{A}^* \boldsymbol{x}_k^* + \boldsymbol{w}_k^* \\ \boldsymbol{y}_k = \boldsymbol{H}^* \boldsymbol{x}_k^* + \boldsymbol{v}_k \end{cases}$$

Therefore, the augmented covariance matrix $\boldsymbol{Q}^*$ will be written as:

$$\boldsymbol{Q}^* = \begin{bmatrix} \boldsymbol{Q} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Q}_u \end{bmatrix}$$

Where $\boldsymbol{Q}_u \gg \boldsymbol{Q}$.
Some important observations coming from Capalbo et al. [2] need to be made:

- To have the maximum theoretical observability, a number of strain sensors must be used at least equal to the total number of quantities to be estimated (inputs and parameters).
- Mass parameters cannot be observed in static conditions, therefore in the estimation, for example, of the mass of a body it is necessary to use a dynamic input. Moreover, the input must be characterized by a frequency more or less similar to the dynamic excitation frequency of the component.

The pseudo-algorithm of the Augmented Extended Kalman Filter for state-input estimation is shown below:

Algorithm Augmented Extended Kalman Filter - Input estimation

1: Starting from the system of implicit non linear equations expressed in the GGL formulation:

$$\begin{cases} \frac{1}{h}(\boldsymbol{q}_{k+1} - \boldsymbol{q}_k) - \boldsymbol{v}_{k+1} + \boldsymbol{B}(\boldsymbol{q}_{k+1})^T \boldsymbol{\mu}_{k+1} = 0 \\ \boldsymbol{M}(\boldsymbol{q}_{k+1}) \frac{1}{h}(\boldsymbol{v}_{k+1} - \boldsymbol{v}_k) + \boldsymbol{f}(\boldsymbol{q}_{k+1}, \boldsymbol{v}_{k+1}, \boldsymbol{u}_{k+1}) - \boldsymbol{B}(\boldsymbol{q}_{k+1})^T \boldsymbol{\lambda}_{k+1} = 0 \\ \boldsymbol{B}(\boldsymbol{q}_{k+1}) \boldsymbol{v}_{k+1} = 0 \\ \boldsymbol{\phi}(\boldsymbol{q}_{k+1}) = 0 \\ \boldsymbol{u}_{k+1}^{UK} = \boldsymbol{u}_k^{UK} \end{cases} \tag{1.47}$$

where the input vector is split in two subvectors, one defining the known input vector, $\boldsymbol{u}_{k+1}^K$ and another one defining the unknown input vector $\boldsymbol{u}_{k+1}^{UK}$, which is the one we need to estimate:

$$\boldsymbol{u}_{k+1} = \begin{bmatrix} \boldsymbol{u}_{k+1}^K \\ \boldsymbol{u}_{k+1}^{UK} \end{bmatrix} \tag{1.48}$$

for simplicity we define $\boldsymbol{u}_{k+1}^{UK} = \hat{\boldsymbol{u}}_{k+1}^-$.
The 1.47 is written in compact form as:

$$\boldsymbol{g}_d(\hat{\boldsymbol{x}}_{k+1}^-, \hat{\boldsymbol{x}}_k^+, \hat{\boldsymbol{u}}_{k+1}^-) = 0 \tag{1.49}$$

so we find an explicit formulation in the form of:

$$\hat{\boldsymbol{x}}_{k+1}^- = \boldsymbol{f}_d(\hat{\boldsymbol{x}}_k^+, \hat{\boldsymbol{u}}_{k+1}^-) \tag{1.50}$$

where:

$$\hat{\boldsymbol{x}}_{k+1}^- = \begin{bmatrix} \boldsymbol{q}_{k+1} \\ \boldsymbol{v}_{k+1} \end{bmatrix} \tag{1.51}$$

$$\hat{\boldsymbol{x}}_k^+ = \begin{bmatrix} \boldsymbol{q}_k \\ \boldsymbol{v}_k \end{bmatrix} \tag{1.52}$$

2: Knowing $\hat{\boldsymbol{x}}_k^+$ and $\hat{\boldsymbol{u}}_{k+1}^-$, compute a linearization of the state matrix A:

$$\boldsymbol{A} = \left.\frac{\partial \boldsymbol{f}_d(\hat{\boldsymbol{x}}_k^+, \hat{\boldsymbol{u}}_{k+1}^-)}{\partial \boldsymbol{x}}\right|_{\hat{\boldsymbol{x}}_k^+} \tag{1.53}$$

3: Compute also a linearization of the system with respect to the unknown input vector in the "a priori" input prediction of the current step $\hat{\boldsymbol{u}}_{k+1}^-$:

$$\boldsymbol{B}_u = \left.\frac{\partial \boldsymbol{f}_d(\hat{\boldsymbol{x}}_k^+, \hat{\boldsymbol{u}}_{k+1}^-)}{\partial \boldsymbol{u}^{UK}}\right|_{\hat{\boldsymbol{u}}_{k+1}^-} \tag{1.54}$$

4: Find the "a priori" state prediction solving the implicit system of equations 1.47, obtaining the "a priori" state prediction $\hat{\boldsymbol{x}}_{k+1}^-$.

5: Assemble the augmented matrices:

$$\boldsymbol{A}^* = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B}_u \\ \boldsymbol{0} & \boldsymbol{I} \end{bmatrix} \tag{1.55}$$

$$\boldsymbol{H}^* = [\boldsymbol{C} \quad \boldsymbol{0}] \tag{1.56}$$

6: Compute the "a priori" state error covariance matrix:

$$\boldsymbol{P}_{k+1}^- = \boldsymbol{A}^* \, \boldsymbol{P}_k^+ \, \boldsymbol{A}^{*T} + \boldsymbol{Q}^* \tag{1.57}$$

7: Compute the Kalman gain:

$$\boldsymbol{K}_{k+1} = \boldsymbol{P}_{k+1}^- \, \boldsymbol{H}^{*T} \, \left(\boldsymbol{H}^* \, \boldsymbol{P}_{k+1}^- \, \boldsymbol{H}^{*T} + \boldsymbol{R}\right)^{-1} \tag{1.58}$$

8: Compute the "a posteriori" augmented states using the available measurements:

$$\begin{bmatrix} \hat{\boldsymbol{x}}_{k+1}^+ \\ \hat{\boldsymbol{u}}_{k+1}^+ \end{bmatrix} = \begin{bmatrix} \hat{\boldsymbol{x}}_{k+1}^- \\ \hat{\boldsymbol{u}}_{k+1}^- \end{bmatrix} + \boldsymbol{K}_{k+1}(\boldsymbol{y}_{k+1} - \boldsymbol{h}_d(\hat{\boldsymbol{x}}_{k+1}^-)) \tag{1.59}$$

9: Compute the "a posteriori" state error covariance matrix:

$$\boldsymbol{P}_{k+1}^+ = (\boldsymbol{I} - \boldsymbol{K}_{k+1}\boldsymbol{H}^*) \, \boldsymbol{P}_{k+1}^- \tag{1.60}$$

3. Test Bench Down Scaling

Starting from a test bench model of a certain size associated with the 3.4 MW wind turbine, we want to create a scaled test bench model, i.e. of reduced size, so that it allows to reproduce the same KPIs, like e.g. displacements of the ball bearing using a particular estimated load.
A flexible model of the Wind Turbine Test Bench embedded in a full non linear framework to capture the non-linear body motion is used in combination with strain measurements to retrieve the loads transmitted from the blade by means of a state-input estimator. The

model is composed by a flexible hub modeled as a finite element model, a rigid blade connected to the hub through a pitch ball bearing bearing.

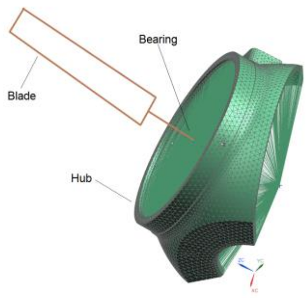


Figure 1: Flexible Multibody model of the system

The steps to follow for the estimation are:

1. Simulate the full-scale test bench model by applying a known static or dynamic load and collect sensor measurements.
2. Create a test bench model that is scaled, i.e. of smaller size with respect to the previous model.
3. Apply the AEKF algorithm to the scaled model to estimate the input forces needed to have the same displacement state measured in the previous simulation of the full scale model.

In the following figure we can observe the trend of the forces estimated for the scaled test bench, in the case of measurements with and without noise compared with the forces applied in the original test bench. These estimated loads thus have to be applied to the scaled test bench to obtain the same KPIs as the full-scale testbench, like e.g. bearing displacements.

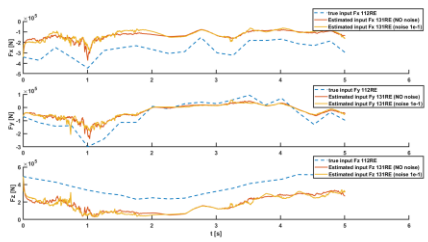


Figure 2: Estimated input forces of scaled test bench model

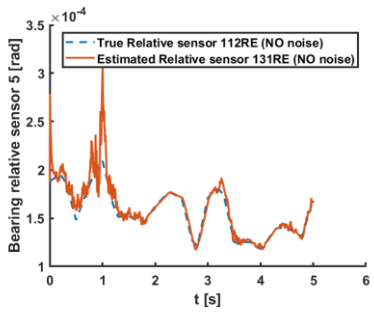


Figure 3: Estimated state of the relative displacement sensor of the bearing

Note that the 112RE is the full-scale testbench and the 131RE is the scaled testbench. In the following figures we observe the measurements of one of the different strain sensors placed in the hub, the comparison between that obtained in the original non-scaled system and those estimated for the scaled system show that the latter have an absolute value smaller magnitude, which is what we expected since the forces estimated for the scaled system are also lower in magnitude.

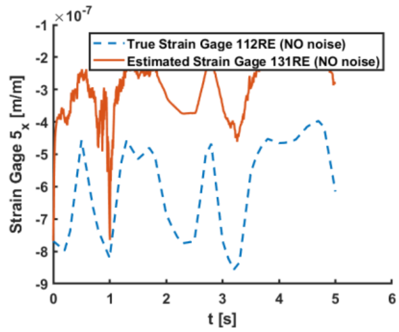


Figure 4: Estimated state of a strain gage measurement taken from the hub

In the following plot it is possible to observe the trend of the ratio between the dynamic load applied in the original test bench and that estimated in the scaled test bench, and as was expected, this ratio is highly non-linear, thus confirming that it is not possible to reduce the magnitude of the loads by the same percentage of reduction as the test bench and thus showcasing the need for an estimator as proposed in this thesis.

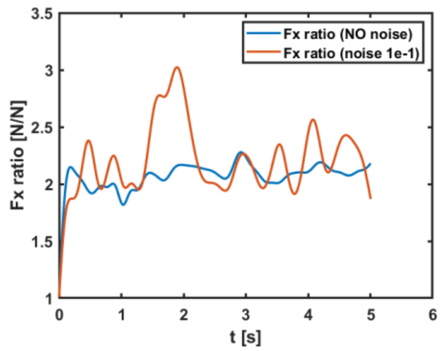


Figure 5: Fx ratio

4. Optimal Sensor Placement

An Optimal Sensor Placement strategy aims to select the optimal position of the sensors needed to solve an input and parameter estimation problem, however, this thesis focuses mainly on optimizing the positioning of strain sensors, called Strain Gauges, [6], [22], [4]. Before proceeding with the description of the method, an important assumption must be made, that is, in the case of estimating the input, the knowledge of the direction of the input and the point of application of the same, are known, in any case, in the event that it is not possible to know the direction of the load, it would be sufficient to estimate the three Cartesian components of the load and obtain the direction of the resulting load by means of a simple vector operation. An outline of the first part of the method, the Training, is represented in the figure below. The training aims to reduce the number of sensors while keeping a large enough number before applying the second step of the method, that is the Steady State Error Covariance metric.

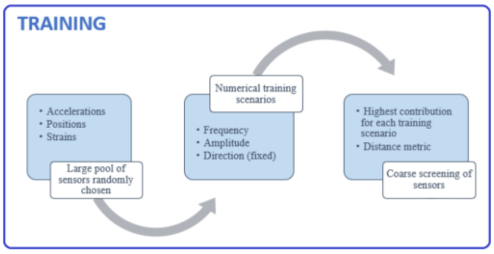


Figure 6: Training of the system for the first coarse screening, [6]

So we start from the generation of a pool of potential sensors, in particular we select a set of elements in the finite element model. Then the algorithm starts with the creation of the output matrix $\boldsymbol{H}$ by placing a Strain Gauge sensor in the outer surface of each element contained in the starting element set. At this point we proceed with the Coarse Screening, which consists in simulating the system with a training input (usually sinusoidal), the measurements of the initial pool of sensors are collected and those with the lowest signal-to-noise ratio are removed, i.e. those less sensitive to training input. A further removal of sensors takes place on the basis of the proximity between them, in particular, all those nearby sensors are removed that create clusters in the surface of the model. In the end, a set of sensors that represent the basis on which the Steady State Error Covariance metric begins to work is obtained.

3.2. OSP - Steady State Error Covariance Metric

Starting from the pool of sensors obtained after coarse screening, this algorithm is based on the solution of the Continuous Algebraic Riccati Equation (CARE) for different sensor configurations, [6]:

$$\boldsymbol{F}^*\boldsymbol{P} + \boldsymbol{P}\boldsymbol{F}^{*T} + \boldsymbol{Q}^* - (\boldsymbol{P}\boldsymbol{H}^{*T}\boldsymbol{R}^{-1})\boldsymbol{R}(\boldsymbol{P}\boldsymbol{H}^{*T}\boldsymbol{R}^{-1})^T = 0$$

Where $\boldsymbol{F}^*$ represents the augmented linearized state matrix, while $\boldsymbol{H}^*$ represents the augmented measurement matrix. The term $\boldsymbol{Q}^*$ represents the plant covariance matrix expressing the uncertainty in the augmented model, that is, the uncertainty in the finite element model, and in the input and parameter dynamical model. An important assumption is that $\boldsymbol{Q} \ll \boldsymbol{Q}_u, \boldsymbol{Q}_p$, this is because we assume that the finite element structural model is orders of magnitude more accurate than the zeroth-order hold model of the input and parameter. So the following approximation of $\boldsymbol{Q}^*$ is proposed:

$$\boldsymbol{Q}^* = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{Q}_u & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{Q}_p \end{bmatrix}$$

$\boldsymbol{Q}_u$ and $\boldsymbol{Q}_p$ are calculated on the basis of the type of estimate we want to make, in fact they take on a different form if the goal is to estimate only the input, only the parameters or the set of inputs and parameters.
Finally, $\boldsymbol{R}$ represents the covariance matrix of the measurements.

3.2.1. State - Input Augmentation

Starting from the state representation of the linear dynamic system under analysis, augmenting the state vector $\boldsymbol{x}$ with the input $\boldsymbol{u}$ means creating the following state vector:

$$\boldsymbol{x}^* = \begin{bmatrix} \boldsymbol{q} \\ \dot{\boldsymbol{q}} \\ \boldsymbol{u} \end{bmatrix}$$

Consequently the new system becomes:

$$\begin{cases} \dot{\boldsymbol{x}}^* = \boldsymbol{F}^*\boldsymbol{x}^* \\ \boldsymbol{y} = \boldsymbol{H}^*\boldsymbol{x}^* \end{cases}$$

Where:

$$\boldsymbol{F}^* = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} = \begin{bmatrix} \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ -\boldsymbol{M}^{-1}\boldsymbol{K} & -\boldsymbol{M}^{-1}\boldsymbol{C} & \boldsymbol{M}^{-1}\boldsymbol{S} \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{bmatrix}$$

$$\boldsymbol{H}^* = [\boldsymbol{H} \quad \boldsymbol{D}]$$

3.2.2. State - Young's Modulus Augmentation

In the case when the augmentation is done with the Young's modulus parameter, the dynamic system in state form is no longer linear, because inside the matrix $\boldsymbol{A}$ the stiffness matrix $\boldsymbol{K}$ is a function of Young's modulus, therefore consequently also the matrix $\boldsymbol{A}$, since in the augmented version, the parameter is seen by the Kalman filter as simply a new state, it means that $\boldsymbol{A}$ is dependent on the states (on the parameter) and therefore it is no longer linear, that is $\boldsymbol{A}$ is no longer constant but varies over time. The following system is still linear because it is not the augmented version:

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{A}(E)\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} \\ \boldsymbol{y} = \boldsymbol{H}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{u} \end{cases}$$

With:

$$A(E) = \begin{bmatrix} \mathbf{0} & \boldsymbol{I} \\ -\boldsymbol{M}^{-1}\boldsymbol{K}(E) & -\boldsymbol{M}^{-1}\boldsymbol{C} \end{bmatrix}$$

But if we augment the parameter E with the states we get the following non-linear system:

$$\begin{cases} \dot{\boldsymbol{x}}^* = \boldsymbol{A}^*(\boldsymbol{x}^*)\boldsymbol{x}^* + \boldsymbol{B}^* \\ \boldsymbol{y} = \boldsymbol{H}^*\boldsymbol{x}^* + \boldsymbol{D}\boldsymbol{u} \end{cases}$$

where:

$$\boldsymbol{x}^* = \begin{bmatrix} \boldsymbol{q} \\ \dot{\boldsymbol{q}} \\ E \end{bmatrix}$$

$$\boldsymbol{A}^*(\boldsymbol{x}^*) = \begin{bmatrix} \boldsymbol{A}(x^*) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}$$

$$\boldsymbol{B}^* = \begin{bmatrix} \boldsymbol{B} \\ \mathbf{0} \end{bmatrix}$$

$$\boldsymbol{H}^* = [\boldsymbol{H} \quad \boldsymbol{D}]$$

So all that remains is to linearize the system with respect to an augmented reference state:

$$\boldsymbol{F}^* = \frac{\partial(\boldsymbol{A}^*\boldsymbol{x}^* + \boldsymbol{B}^*\boldsymbol{u})}{\partial \boldsymbol{x}^*}$$

$$\boldsymbol{F}^* = \begin{bmatrix} \mathbf{0} & \boldsymbol{I} & \mathbf{0} \\ -\boldsymbol{M}^{-1}\boldsymbol{K} & -\boldsymbol{M}^{-1}\boldsymbol{C} & \boldsymbol{M}^{-1}\dfrac{\partial \boldsymbol{K}}{\partial E}\boldsymbol{q} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

$$\frac{\partial \boldsymbol{K}}{\partial E} = \boldsymbol{K}^E$$

$\boldsymbol{F}^*$ and $\boldsymbol{H}^*$ are the matrices that will actually be used in solving the CARE.

3.2.3. OSP Algorithm

Then, once the case of interest has been defined and the appropriate $\boldsymbol{F}^*$ and $\boldsymbol{H}^*$ matrices have been selected based on the kind of augmentation is used, we proceed with the application of the OSP algorithm based on the solution of the CARE. The pseudo algorithm is shown below:

Algorithm Optimal sensor placement - pseudo algorithm,

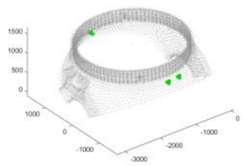
Execute the iterative process to find the $\bar{n}_s$ optimal sensors:
for $k = 0 : (n_s^0 - \bar{n}_s)$ **do**
  given the measurement matrix $\boldsymbol{H}^{n_s^0-k}$ evaluate CARE
  $\boldsymbol{P}_{uu}^k = care(\boldsymbol{H}^{n_s^0-k})$
  Iterate on the subset of sensors $n_s^0 - k - 1$:
  for $g = 1 : (n_s^0 - k - 1)$ **do**
    remove sensor g from $\boldsymbol{H}^{n_s^0-k}$ to obtain $\boldsymbol{H}^{n_s^0-k-1}$, with $g \notin (n_s^0 - k - 1)$
    $\boldsymbol{P}_{uu}^{k,g} = care(\boldsymbol{H}^{n_s^0-k-1})$
  end for
  Evaluate the sensor $\bar{g}$ which gives the lowest contirbution to the covariance matrix
  $\bar{g} = min(trace(\boldsymbol{P}_{uu}^k) - trace(\boldsymbol{P}_{uu}^{k,g}))$
  remove the sensor $\bar{g}$ from $\boldsymbol{H}^{n_s^0-k}$ to obtain $\boldsymbol{H}^{n_s^0-\bar{k}}$, with $\bar{k} = k + 1$
end for

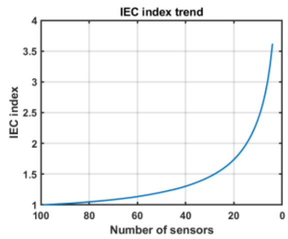
3.3. Hub Test Bench Numerical Testing

The numerical test of the OSP is performed on the hub of the Test Bench, where the load (the input) is applied directly in node ID 2 rather than in the blade.

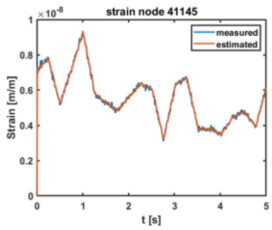
3.3.1. OSP – Input Estimation

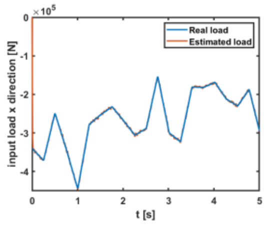
Once the parameters of interest have been estimated, we have a more accurate model of the system available, so it is now possible to estimate a generic input. The following figures show the results of the OSP for the estimation of the input.





The following figures show the results of the input estimate, in particular on the left we observe the estimate of one of the four strain sensors while in the figure on the right we observe the estimate of the input.

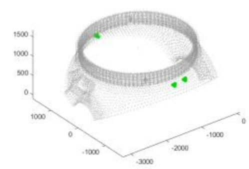


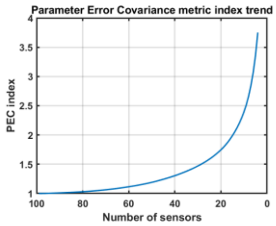


3.3.2. OSP - Young's Modulus Estimation

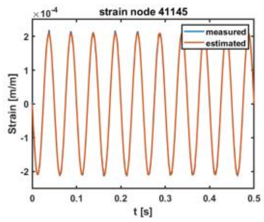
The training load for the coarse screening that is applied is a sinusoidal input with a frequency of 20 Hz and an amplitude equal to 1e10 Newton along the global X-direction, the signals generated in the training simulation are not affected by noise.

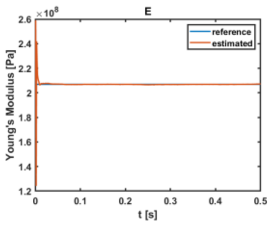
The number of sensors that is selected after the coarse screening is set to 100, it has been avoided to increase this number to avoid that the algorithm arrives at the solution in too long computational times. The two figures below show the results of the algorithm, in particular the image on the left shows where the 4 final sensors are located in the surface of the hub, while the image on the right shows the trend of the Parameter Error Covariance (PEC) metric used in the OSP:





It can be seen from the graph on the left that two clusters of two sensors each have been created, suggesting that some of them are redundant, in fact sensors close enough to each other will give the same information, obviously with the assumption that the strain field doesn't change too much throughout the structure/that region, this implies that the Young's modulus could be estimated using just two sensors. Furthermore, it can be seen from the graph on the trend of the metric that in passing from the initial 100 sensors obtained from the first coarse screening to the final 4 sensors of the OSP, the metric quadruples. At this point, the next step consists in simulating the dynamic system by applying an input equal to that used in the training scenario of the OSP and the measurements coming from the four optimal sensors previously obtained with the OSP are collected. Finally, the AEKF algorithm is used to estimate the strains in the elements of the OSP set and the Young's modulus parameter. The following figures show the results of the estimation of the parameter, in particular on the left we observe the estimate of one of the four strain sensors while in the figure on the right we observe the estimate of the Young's modulus parameter.





As can be seen from the figures, the estimate of the sensor coincides almost perfectly with the real one, while the estimate of the parameter converges to the real value almost instantaneously.

5. Conclusions

The thesis demonstrates the effectiveness of the Augmented Extended Kalman Filter (AEKF) in estimating loads, strain fields, and parameters in Wind Turbine Test Benches. This approach, based on flexible multibody models, offers a cost-efficient alternative to direct measurements. The AEKF's suitability for nonlinear systems like wind turbines and its ability to handle uncertainties make it a valuable tool for virtual sensing. The Optimal Sensor Placement (OSP) strategy enhances load-parameter estimation. Future research directions include applying AEKF to diverse mechanical systems, optimizing sensor placement, exploring alternative estimation techniques, extending virtual sensing to higher-level parameters, and conducting real-world validations for practical implementation.

References

[2] C. E. Capalbo, D. De Gregoriis, T. Tamarozzi, H. Devriendt, F. Naets, G. Carbone, and D. Mundo. Parameter, input and state estimation for linear structural dynamics using parametric model order reduction and augmented Kalman filtering. Submitting, 2022.

[4] R. Cumbo, T. Tamarozzi, K. Janssens, and W. Desmet. Kalman-based load identification and full-field estimation analysis on industrial test case. Mechanical Systems and Signal Processing, 117:771–785, 2019.

[5] R. Cumbo, T. Tamarozzi, P. Jiranek, W. Desmet, and P. Masarati. State and force estimation on a rotating helicopter blade through a kalman-based approach. Sensors, 20(15):4196, 2020.

[6] R. Cumbo, L. Mazzanti, T. Tamarozzi, P. Jiranek, W. Desmet, and F. Naets. Advanced optimal sensor placement for kalman-based multiple-input estimation. Mechanical Systems and Signal Processing, 160:107830, 2021.

[18] E. Risaliti, M. Vermaut, J. Croes, B. Cornelis, and W. Desmet. Force estimation on a mcpherson system by means of a state estimator and a multibody model. In proc. of the ECCOMAS Thematic Conference on Multibody Dynamics, 2017.

[19] E. Risaliti, T. Tamarozzi, M. Vermaut, B. Cornelis, and W. Desmet. Multibody model based estimation of multiple loads and strain field on a vehicle suspension system. Mechanical Systems and Signal Processing, 123:1–25, 2019.

[22] T. Tamarozzi, E. Risaliti, W. Rottiers, W. Desmet, et al. Noise, ill-conditioning and sensor placement analysis for force estimation through virtual sensing. In In International Conference on Noise and Vibration Engineering (ISMA2016),, pages 1741–1756. Katholieke Univ Leuven, Dept Werktuigkunde, 2016.