Energy Informatics

RESEARCH Open Access

Predictive digital twin for offshore wind farms

Amirashkan Haghshenas[1*], Agus Hasan[2], Ottar Osen[2] and Egil Tennfjord Mikalsen[1]

*Correspondence:
amirashkan@osc.no

[1] Offshore Simulator Centre AS, Ålesund, Norway
[2] Department of ICT and Natural Sciences, Norwegian University of Science and Technology, Ålesund, Norway

Abstract

As wind turbines continue to grow in size, they are increasingly being deployed offshore. This causes operation and maintenance of wind turbines becoming more challenging. Digitalization is a key enabling technology to manage wind farms in hostile environments and potentially increasing safety and reducing operational and maintenance costs. Digital infrastructure based on Industry 4.0 concept, such as digital twin, enables data collection, visualization, and analysis of wind power analytic at either individual turbine or wind farm level. In this paper, the concept of predictive digital twin for wind farm applications is introduced and demonstrated. To this end, a digital twin platform based on Unity3D for visualization and OPC Unified Architecture (OPC-UA) for data communication is developed. The platform is completed with the Prophet prediction algorithm to detect potential failure of wind turbine components in the near future and presented in augmented reality to enhance user experience. The presentation is intuitive and easy to use. The limitations of the platform include a lack of support for specific features like electronic signature, enhanced failover, and historical data sources. Simulation results based on the Hywind Tampen floating wind farm configuration show our proposed platform has promising potentials for offshore wind farm applications.

Keywords: Digital twin, Wind energy, Predictive maintenance

Introduction

Wind power is becoming increasingly popular across the world as it plays a vital role in both sustainable and emission-free energy production, making it a perfect energy resource for reducing carbon footprint and global warming. Wind turbines with modern technologies are complex machines combining aerodynamics, mechanics, and electrical with advanced control systems. They continue to grow in size, and they are increasingly being deployed offshore in hostile and operationally demanding environments. To ensure the systems are safe, profitable, and cost-effective, it is imperative to implement a well-organized operation and maintenance strategy based on a digital solution (Garlick et al. 2009). The ongoing global digital revolution, sparked by the Industry 4.0 initiative, has brought new concepts and emerging technologies to the fore that can help these missions to be accomplished. One of the core concepts of Industry 4.0 is digital twin, which can be defined as a digital representation of a physical asset. A digital twin is intended to accurately represent a physical object, based on data and simulation, that can be used for forecasting, monitoring, controlling, and optimizing through the entire lifespan of the asset. Many applications of digital twins have already been developed, including for power generation, manufacturing and processes, building structures, meteorology, healthcare systems, education systems, automotive industries, and urban planning (Rasheed et al. 2020). This paper introduces and demonstrates the concept of predictive digital twin for wind farm operation and predictive maintenance. To this end, we develop a digital twin platform based on Unity3D for visualization and OPC Unified Architecture (OPC-UA) for data communication. Specifically, our proposed digital platform is used to provide predictive information regarding the potential failures of wind turbine components.

Motivation and scope

The wind industry is looking for a way to increase its energy production as the demand for renewable energy develops. One way to boost the energy output is by increasing the size of the rotor blades. The rising size of the blades can put more strain on the turbine's construction and other components. Lightning strikes, blade icing, material or power regulator failure, damage from external objects, and poor design are all contributing to blade failure, which can result in costly repairs and income loss if the turbine is standstill (Tavner et al. 2013). Furthermore, the generator, gearbox, and bearing are also prone to failure. The main causes for the generator failure can be attributed to wind loads, weather conditions, manufacturing or design flaws, incorrect installation, lubricant contamination, and insufficient electrical insulation. Based on historical data and research, bearings and gears account for the majority of the gearbox failures (Elasha et al. 2019). Unclean lubricant, inaccurate bearing settings, temperature and vibration variations, and inappropriate maintenance are just a few of the variables that might cause failure. In general, wind turbine failures can be divided into two categories: external and internal, as shown in Fig. 1. Electrical failures mostly are caused by moisture and temperature inside the converter enclosure. This environmental condition creates a seasonal conversion climate. Short circuits caused by condensation is also one of the most common electrical failures. This usually happens after a scheduled or unplanned shutdown resulting in damage to the components, necessitating replacement, and reducing the wind turbine's lifetime. Mechanical failures inside the nacelle largely occur due to temperature problems, moisture reaction with metal parts that weaken and degrade mechanical elements, problems with the hydraulic and cooling system, blade icing, and erosion.

Due to the nature of the offshore environment, operation and maintenance of wind farms can be difficult and expensive. Thus, there is an incentive to plan operation and maintenance in safer and smarter ways. Digital twins can be viewed as an enabling technology for intelligent wind farm operation. A digital twin can be defined as a virtual model designed to accurately reflect a physical asset (Jones et al. 2020; Liu et al. 2021). Unlike the current practice, which is based on the Supervisory Control and Data Acquisition (SCADA) system, digital twins can be used for prediction and forecasting (Dai et al. 2018). Remark that digital twins are implemented in a software, for which algorithms that can be used for prediction and forecasting are written based on Machine Learning (ML) or Artificial Intelligence (AI). In our case, we use the Prophet algorithm,

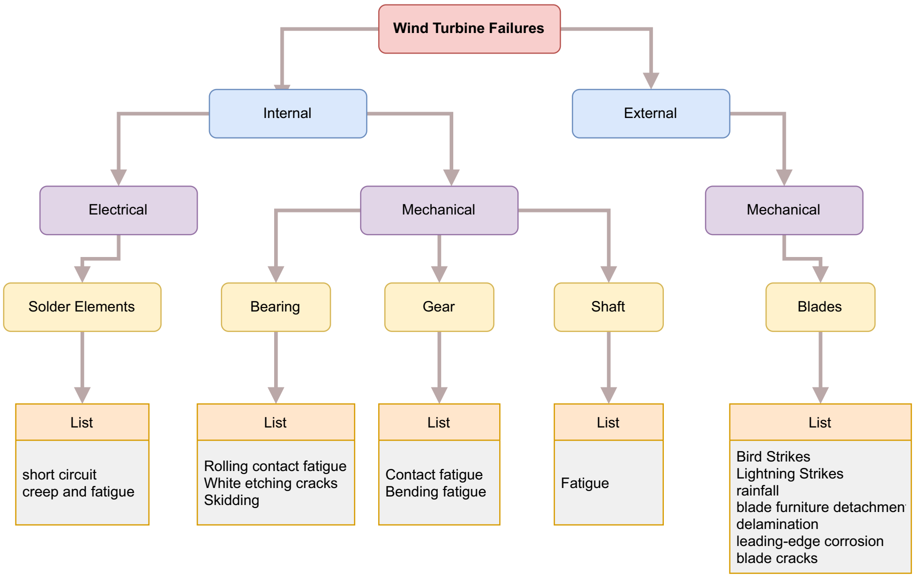


Fig. 1 Different types of wind turbine failures

which is a type of ML algorithm. There are many types of digital twin, e.g., monitoring digital twin, imaginary digital twin, prescriptive digital twin, and predictive digital twin (Verdouw et al. 2021). The objective of this paper is to design and demonstrate a predictive digital twin platform for offshore wind farms based on Unity3D and OPC-UA, which can be used to predict abnormalities and possible failures in each individual wind turbine. Due to the vastness of the subject and the considerable variety of subtopics, we narrow the subject down to mechanical component failures. Having said that, the predictive digital twin platform can also be implemented for electrical or control system failure prediction. One of the most crucial components of rotary equipment is the bearing. The key point to monitor the bearing effectively is the accurate degradation process prediction, which helps to prevent total failures and reduce maintenance costs. Therefore, the case example selected in this paper is about bearing failures since they are a major source of unscheduled maintenance, repairs, and replacements, resulting in energy production downtime (Dong et al. 2014).

Literature review

Digital twin has various aspects and comes with different definitions. Boschert and Rosen (2016) define a digital twin as a description of the physical and functional characteristics of a component, a product, or a system that includes more or less all information that can be useful throughout its entire life-cycle. Fuller et al. (2020) describe a digital twin as an integration of data between physical and virtual machines in either direction with ease. Glaessgen and Stargel (2012) express that in order to accurately reflect the life of a physical asset, a digital twin utilizes the best available physical models, sensor updates, fleet history, etc., to create an integrated multi-physics, multiscale, and probabilistic simulation. Finally, according to Verdouw et al. (2015), a digital twin is a digital representation of an object with a unique identification that can be trusted, is of integrity, is immediately available, and can serve its intended purpose. According to all mentioned definitions, it can be said that a digital twin provides predictability, control, monitoring, and optimization of physical assets by utilizing data and simulations during the entire life-cycle of the assets. The aforementioned definitions implicitly underline the importance of communication between the physical asset and its digital twin. The OPC-UA is an industrial machine-to-machine communication developed by the OPC Foundation (Mühlbauer et al. 2020). The OPC-UA is based on commonly used communication standards like the Hypertext Transfer Protocol (HTTP). Thus, it can be used in different operating systems. Because of its flexibility, the OPC-UA has been considered as a pillar of representing semantic digital twins (Perzylo et al. 2019). For this reason, we use OPC-UA as the communication protocol in our digital twin platform.

Digital twin of wind farms is beneficial for monitoring and operating individual wind turbines remotely (Pimenta et al. 2020). They enable cost-effective maintenance and ensure greater reliability of the components used to convert wind energy into electricity (Moghadam et al. 2021). Oñederra et al. (2019) outlined development of a digital twin for a medium voltage cable prototype in a wind farm which can be used to simulate its behavior and increase its lifespan in order to accomplish preventative maintenance. In their work, a hybrid model of a dynamic medium voltage cable model and an interpolation technique was created in OpenModelica. Furthermore, as part of a predictive maintenance plan, Sivalingam et al. (2018) provided a unique approach for predicting the Remaining Useful Life (RUL) of an offshore wind turbine in a digital twin shell by monitoring the turbine conditions. Wang et al. (2021) summarized recent work regarding reliability of offshore wind turbine structures and reviewed some possible damages/failure. Moreover, they proposed a digital twin concept to monitor offshore wind turbine support structures as a solution to some problematic challenges. Botz et al. (2019) conducted research to apply digital twin framework by gathering vital data from attentively chosen spots of hybrid wind turbine structure for updating material models of the wind turbine in order to improve maintenance and operating parameters and extend the turbine's useful life. In another publication, Kooning et al. (2021) presented a summary of recent research on modelling methodologies to build a digital twin for a wind turbine by considering the components, aerodynamics, structural and mechanics, power electronic converters, pitch and yaw systems. Furthermore, Pimenta et al. (2020) created a digital twin by using SCADA to create a feasible trustworthy numerical model of a floating wind turbine.

Applying predictive methodologies into a digital twin platform to estimate failure probability provides the ability to schedule on-time maintenance for reducing repair time and unplanned maintenance, as well as organize proper spare parts to mitigate inventory costs. The prediction of wind turbine failures is mostly conducted based on the time scales of short-, medium-, and long-term (Foley et al. 2012). Long-term prediction focuses on estimating the variables over time periods of days, whereas medium-term prediction examines the variables at hourly intervals, and short-term forecasting seeks to estimate the values at 10-s or 10-min intervals. Different prediction methods have been used to forecast wind turbine components, which can be divided into four categories: statistical models, physics-based methods, data mining algorithms, and hybrid models (Kusiak et al. 2013). Statistical forecasting approaches are popular because of their objective in analyzing data and identifying patterns that can be used for future forecasts. An example of statistical approaches is the Auto Regressive Integrated Moving Average (ARIMA). The ARIMA is a statistical method that uses previous values to describe the values of a time series. This method is built on two basic characteristics of past values and errors. Furthermore, the method utilizes historical data to determine the performance of the model by using estimated errors (Menculini et al. 2021). Physics-based methodologies are based on the physical models that define and represent the behavior of the variables providing more reliable predictions for the future trend of the model. In general, the nonlinear part of the data is considered as numerical methods concentrate on capturing the linear data. Data mining methodology is used to model both linearity and non-linearity of the data. Data mining models are created through heuristics and calculations. It first searches for patterns or trends among the provided data to create a model. The Neural Network (NN), Support Vector Machine (SVM), K-Nearest Neighbour (KNN), and tree-based regression algorithms are examples of data mining methods used for prediction. The SVM, in particular, is a prominent supervised learning technique that is utilized to solve both classification and regression issues. However, it is mostly applied in machine learning for classification problems. This algorithm aims to construct the best line or decision boundary in n-dimensional space that can be used to categorize the data easily in the future, which is known as a hyperplane, to easily place the new points in the correct category. The SVM creates the hyperplane by choosing extreme points. As a result, extremes are called support vectors, and the algorithm that employs them is known as a Support Vector Machine (Jose et al. 2013). The Auto-Encoder Neural Network (AENN) is a type of unsupervised Artificial Neural Networks (ANN) to rebuild and decode the data from the compact encoded model into a representation close to the original input. It is basically designed to minimize data dimensionality by learning how to disregard data noise (Ren et al. 2018).

The aforementioned methods above are, however, prone to large trend errors when there is a change in trend near the cutoff period and they fail to capture any seasonality, which underlies the idea of the Prophet prediction algorithm (Taylor and Letham 2017). The Prophet fits non-linear trends with annual, weekly, and daily seasonality, as well as holiday impacts, to forecast time series data (Jana et al. 2022). It is an open-source software developed by Facebook's Core Data Science team. It is used for time series forecasting with substantial seasonal consequences and chronological data from several seasons. Generally, Prophet can handle outliers and missing data and is robust to changes in the trend.

Contribution of this paper

The novelty of this paper is the development of a predictive digital twin architecture and software for wind farm applications based on OPC-UA and Unity3D. Thus, the contribution is within the area of system engineering and software development (please refer to Fig. 5). All codes are available in this link: https://github.com/hamirashkan/Predictive_DigitalTwin_WindFarm. The OPC-UA is a secure and reliable mechanism for information exchange between systems (Stojanovic et al. 2021), while the Unity3D is one of the most popular game engines in the gaming industry. The proposed digital twin platform is further combined with the Prophet algorithm to predict the component failures in individual wind turbines.

The contributions of this paper can be summarized as follow:

- Development of a bi-directional digital twin platform architecture for monitoring and controlling of wind farms based on Unity3D and OPC-UA. To the best of our knowledge, this is the first time such architecture is proposed, developed, and demonstrated for wind farm applications.
- Development of a failure prediction method based on the Prophet algorithm for component failure prediction in individual wind turbines. To the best of our knowledge, this is the first time the algorithm is used for prediction in wind farm applications.
- Development of augmented reality for interactive visualization in the predictive digital twin platform.

Outline of this paper

This paper is divided into eight sections. Section 1 provides the introduction. Section 2–5 describe the platform development and its related algorithms for prediction and visualization tools. Section 6–7 discuss the experimental setup and results. Finally Section 8 is the conclusion. The complete outline is given below:

- Introduction, which includes motivation & scope, literature review, contribution of this paper, and outline of this paper.
- Failure prediction in wind turbines, which includes failure prediction procedure, data processing & cleaning, and modelling & forecasting.
- Digital twin platform development, which includes communication architecture, data source, and visualization interface.
- Predictive twin algorithms, which includes algorithms for temperature and vibration failure prediction.
- Augmented reality for interactive visualization.
- Experimental setup, which includes data availability and simulation setup.
- Result and discussion.
- Conclusion.

Failure prediction in wind turbine

Among possible sources of failures in Fig. 1, bearing is one of the wind turbine components that is prone to failure (Liu and Zhang 2020). Thus, we use it as a use case. Nonetheless, the method presented in this paper can be extended easily for other sources of failures. Bearing health status can be measured and monitored based on different variables and parameters. However, temperature and vibration are the two most notable indicators of whether bearings are functioning normally or not. Various methodologies have been used to evaluate the rolling element of the RUL. Current technologies such as machine learning and artificial intelligence provide accurate results of prediction and are being used to monitor and predict asset condition and behavior.

Bearing failure prediction procedure

The normal procedure for bearing health prediction can be seen from Fig. 2 and is elaborated as follows (Jin et al. 2021):

1. Obtaining data from the asset which can be done via various methods such as the SCADA system.
2. Cleaning and processing the acquired raw data for dealing with missing values, dividing the dataset into dependent and independent variables, and splitting the dataset into training and testing sets by applying data processing methodologies to achieve an understandable format.
3. Defining a performance index based on the related parameters and factors using methods discussed in the previous section and setting a threshold to find the outliers.
4. Building a model to train the data, which is mostly implemented by machine learning methods using the healthy part of the data or the healthy index in the previous step as the training set and train the model to recognize the normal trend of the working condition, and then test the model on the rest of the dataset to check the accuracy of the built model and finally find the best model.
5. Forecasting the future trend of the data based on the built model using machine learning algorithms.
6. Recognizing, extracting, and evaluating the outliers and irregularities in order to discover the possible failure of the working conditions.
7. Decision making and solving the problem either by humans or by employing artificial intelligence to change the potential failure condition into a healthy state.

Data processing and cleaning

Data cleaning is required in order to build a model representing the health status of the wind turbine. The wind turbine system performance is based on sensors that collect data from the SCADA system. Accordingly, there could be outliers in data or, if the sensors malfunction, no data is produced. These errors can occur if sensors are not calibrated

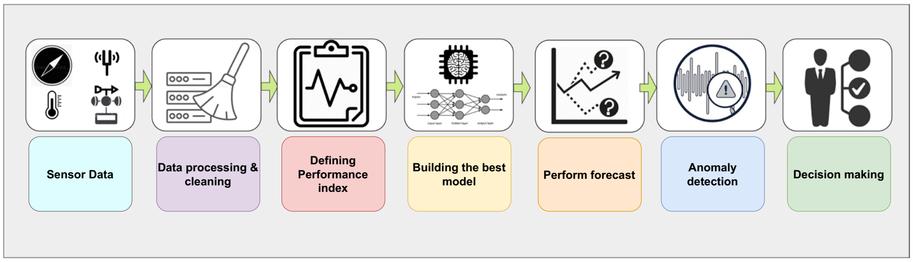


Fig. 2 A typical failure prediction procedure

or if they degrade over time, yielding outliers in the SCADA system. Criteria considered and used for primary data cleaning are as follows:

$$\begin{cases} \text{Delete } x_i, & \mathit{if}\ x_i \in \text{timeout} \vee \text{sensor off} \\ x_i = x_{i+1}\ \mathit{or}\ x_{i-1}, & \mathit{if}\ x_i \in \text{loss data} \\ x_i = 0, & \mathit{if}\ x_i < 0\ \mathit{or}\ x_i \text{ is null} \\ x_i, & \mathit{for} \text{ others} \end{cases} \tag{1}$$

Signal analysis and performance index

Vibration monitoring and analysis in rotating elements provide critical information concerning anomalies occurring inside the machinery's internal structure, as well as the ability to plan maintenance actions. Vibration measurement and interpretation are both parts of the vibration analysis. The information gained from the vibration signals is used to predict failures, increase asset usage and efficiency, extend the life of assets, and lower maintenance costs related to the asset health condition. As long as a machine is in good condition, its vibration spectrum is modest and steady. When problems occur and parts of the machine's dynamic processes change, the vibration spectrum changes as well. Furthermore, there are potential high or low-temperature issues that result in failures in bearing. High temperature leads to a decrease in lubricant viscosity, fatigue, seals drying and cracking, and weakening retainer and cage. Low temperature causes raise the lubricant viscosity, skidding, and increasing torque. Data on the temperature time series of the wind turbine bearing have a firm correlation with their historical values as well as with other related external variables, such as active power, gearbox bearing and oil temperature, hub temperature, ambient temperature, rotor RPM, etc. Therefore, the prediction of temperature changes is essential for overheating warnings. The vibration signals are first gathered in the time domain using a vibration sensor, then applying vibration processing methods and factors to extract useful information from vibration signals in order to forecast the potential failures. There are four indicators/performance indices that can be used to detect bearing failures: Root Mean Square (RMS), Kurtosis, Skewness, and performance index. While the first three are well-known statistical indicators (e.g., see Lin and Ye (2019); Eftekharnejad et al. (2011)), the performance index can efficiently be used to monitor the healthy status of the component and help to find proper thresholds in order to extract the outliers and avoid future failures. Finding a correlation between parameters is a way of finding a health index. For example, for bearing condition monitoring it can be useful to find the correlation between factors concluded in temperature rising such as gearbox bearing and oil temperature, output power, rotor RPM, ambient temperature, hub temperature, or correlation between wind speed and output energy. Defining the performance index based on the correlation of multi-dimensional variables into one-dimensional can also be useful, as can be seen from Fig. 3. In this paper, we define the performance index as the average value of the hub, shaft bearing, and gearbox bearing temperature.

Modeling and forecasting

The Prophet is chosen as the prediction solution in this paper. The reason is not only because it can handle large trend errors and take into account seasonality, but also it

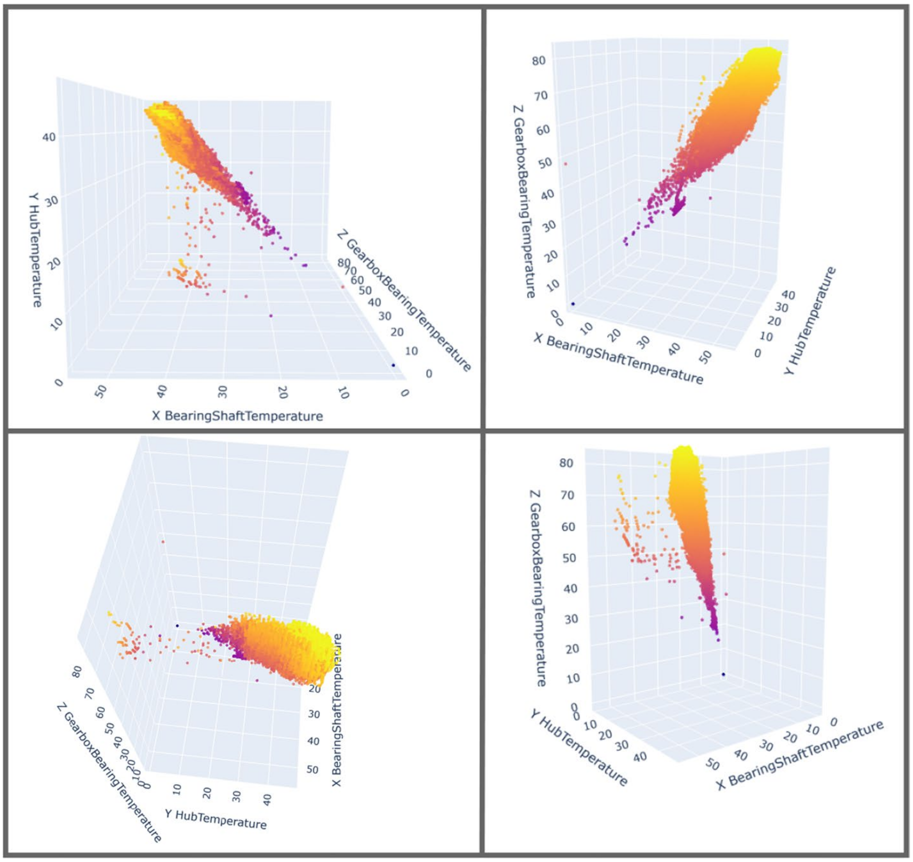


Fig. 3 Strong relation between the hub, shaft bearing, and gearbox bearing temperature

includes parameters that can be adjusted without any knowledge of the underlying model. Furthermore, it has been designed using three-part decomposable time series models of trend, seasonality, and maintenance. All of which makes this algorithm an ideal prediction solution for maintenance operations and failure forecasting. The model is defined as follows:

$$y(t) = g(t) + s(t) + h(t) + \epsilon(t) \tag{2}$$

where $g(t)$ reflects the non-periodic trends in the value of the time series, $s(t)$ indicates the periodic variations such as weekly and yearly seasonality, and $h(t)$ refers to the maintenance schedule. The error term $\epsilon(t)$ is used to refer to any characteristic modifications, which are not captured by the model, and is modelled as Gaussian distribution. In this paper, the trend $g(t)$ is modelled as a linear trend with change points. The reason is because our problem does not exhibit saturation growth. The formula is given by

$$g(t) = (k + \boldsymbol{a}(t)^\intercal \boldsymbol{\delta})t + (m + \boldsymbol{a}(t)^\intercal \boldsymbol{\gamma}) \tag{3}$$

where $\boldsymbol{a}(t) \in \{0, 1\}$, $k \sim \text{Normal}(0, 5)$ is the growth rate, $\boldsymbol{\delta} \sim \text{Exponential}(0, 5)$ is the rate adjustment, $m \sim \text{Normal}(0, 5)$ is the offset parameter, and $\boldsymbol{\gamma}$ is an arbitrary continuous function. If the problem exhibits saturation growth, the model can be replaced by the logistic model. The seasonality component $s(t)$ is chosen based on a standard Fourier series as follows

$$s(t) = \sum_{n=1}^{N} \left(a_n \cos\left(\frac{2\pi nt}{P}\right) + b_n \sin\left(\frac{2\pi nt}{P}\right)\right) \tag{4}$$

where, based on experiment, in our case $N = 24$ and $P = 1$. In this paper, we do not consider the term $h(t)$ as it is not relevant for the simulation.

Digital twin platform development

In this section, we describe the communication architecture, data source, and visualization tools that are used to develop the digital twin platform. The software and hardware used to build the platform are open source. From software development perspectives, the functional and non-functional requirements for the digital twin platform includes:

- Functional requirement
  - The functionality to switch between various data sources (cf. Figs. 3 and 10). This requirement is to ensure the platform can accommodate different data sources, either static data, live data, simulated data, or historical data.
  - Support for both 2D (dashboard), 3D (Unity), and augmented reality user interface (cf. Figs. 5, 7, and 14). This requirement is to ensure the platform can be presented in different visualization types.
  - Role-based user functionality (modes) (cf. Figs. 7 and 9). This requirement is to accommodate different user inputs and configuration. For example, the turbine specification can be defined manually by the users.
  - Support for Functional Mockup Unit (FMU) and Functional Mockup Interface (FMI) (cf. Fig. 10). This requirement is to ensure the platform can accommodate co-simulation based on the FMU/FMI standards.
- Non-functional requirement
  - Support real-time data acquisition (cf. Figs. 3 and 10). This requirement is to ensure that the platform enables real-time data streams from the real asset to the digital model.
  - Based on OPC-UA (cf. Fig. 5). This requirement is to ensure the platform is secure, reliable, and follows the International Electrotechnical Committee standard (IEC 62541).
  - Protect sensitive data (secure) and reliable [OPC-UA is a secure and reliable mechanism for information exchange between systems (Stojanovic et al. 2021)].
  - Compatible with existing operating systems (Unity3D can be installed in Windows, macOS, and Linux). This requirement is to ensure the platform can be used in different system environments.

We use the Hywind Tampen floating wind farm configuration as a case study. The reason for using the Hywind Tampen wind farm is not only because it became the

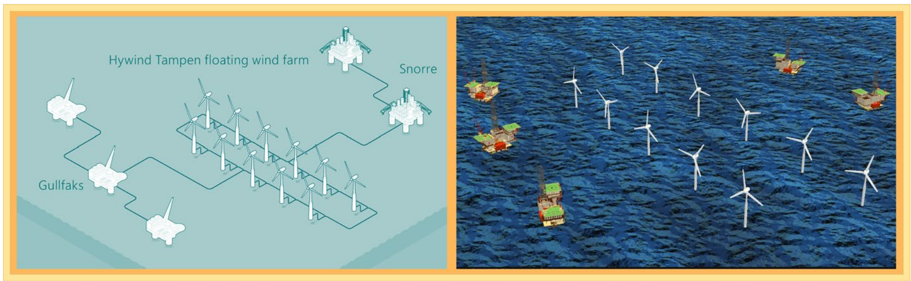


Fig. 4 The left figure shows the Hywind Tampen project developed by Equinor ASA, while the right figure shows its digital twin representation in Unity3D

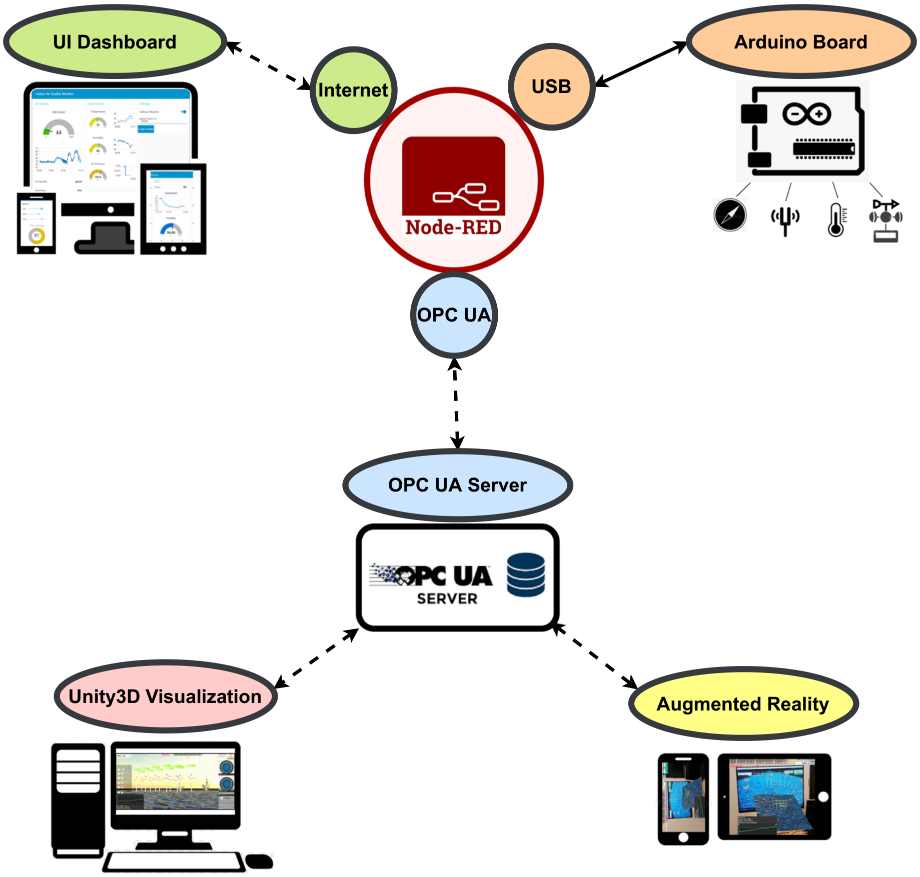


Fig. 5 Communication architecture based on the OPC-UA for connecting digital twin components

symbol for energy transition in the North Sea, but also it became a test bed for further development of floating wind, installation methods, and simplified mooring system. The Hywind Tampen is a 94.6 MW floating wind farm developed by Equinor ASA (see Fig. 4) and is designed to provide power for the Snorre and the Gullfaks offshore oil and gas platforms located at the Norwegian Continental Shelf (NCS). The aim of the wind farm is to eliminate 200,000 tons of CO_2 and 1000 tons of NOx emissions per year. Upon its completion, the Hywind Tampen is the world's biggest floating wind farm and the first wind farm to supply electricity for offshore oil and gas platforms (Tenggren et al. 2020).

Communication architecture

Figure 5 shows the communication architecture of our proposed digital twin platform. The OPC-UA is used as the communication backbone to realize horizontal and vertical communication between subsystems in the field layer and the entities of the upper layers. The OPC-UA is a key component of Industry 4.0, which enables devices and cyber-security systems to be accessed in a common way and data can be exchanged across them in a similar manner regardless of the manufacturer. In this paper, authenticated communication has been used to provide connection between server and clients. The OPC-UA servers can be created through the UaExpert application or other platforms offering this service and all clients can be connected to the available servers from different devices.

The platform is also utilizing Node-RED, which is an open-source Application Programming Interface (API) platform developed by IBM's Emerging Technology Services team, providing a broad range of online services for connecting physical and digital assets. The Node-RED and serial port data can be synchronized with the OPC-UA Server namespaces. Sensor data is connected to the digital platform by using the OPC-UA and serial communication blocks from the Node-RED. Local sensors are connected to the digital platform via Arduino UNO WiFi Rev.2 board which is an IoT hardware used for creating sensor networks. It transmits the sensor data through serial communication which is accessible through cloud platforms and WiFi devices. In this paper, all sensors are connected to the Arduino board, and they send the data to the PC via serial communication port and consequently, can be transferred through the Node-RED platform by adding a serial port block.

Data source

The digital twin platform uses a variety of data sources, which enable different options depending on the application and requirement, as described in Fig. 6. The data can be classified as:

- Static data: which includes data created by the user in the 3D platform to conduct various experiments and what-if scenarios.
- Live data: which includes data received from the physical asset through sensors like wind speed, direction, and temperature.
- Historical data: which are used for simulating semi-realistic scenarios to conduct prediction and processing procedures.
- Simulated data: which are created with physics-based software such as Matlab and imported with the FMI plug-in into the system from more complex physical models.

Visualization

The 3D visualization is implemented in Unity3D, an interactable open-source platform that allows users to drag-and-drop assets from an inventory to the scene and set up various scenarios by changing internal and external factors. We use Unity3D

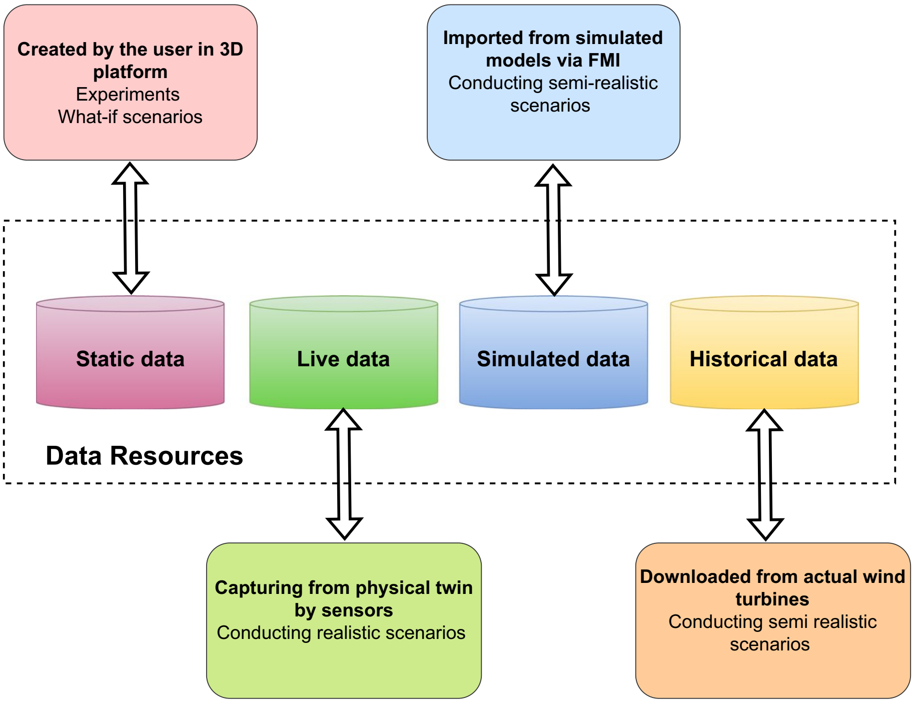
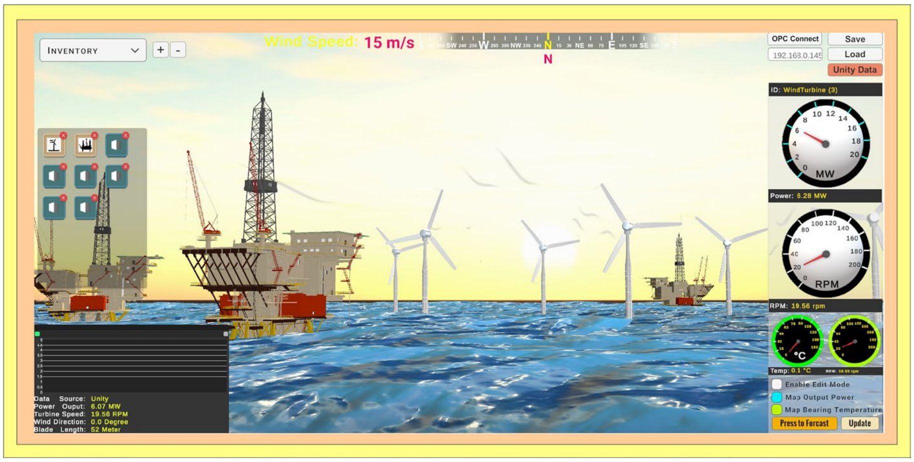

Created by the user in 3D platform
Experiments
What-if scenarios
Imported from simulated models via FMI
Conducting semi-realistic scenarios
Static data
Live data
Simulated data
Historical data
Data Resources
Capturing from physical twin by sensors
Conducting realistic scenarios
Downloaded from actual wind turbines
Conducting semi realistic scenarios

Fig. 6 Different data resources and their use for conducting diverse scenarios



192.168.0.145

ID: WindTurbine (3)

Power: 6.28 MW

RPM: 19.56 rpm

Temp: 0.1 °C

Data Source: Unity
Power Ouput: 6.07 MW
Turbine Speed: 19.56 RPM
Wind Direction: 0.0 Degree
Blade Length: 52 Meter

Fig. 7 Control and visualization of desired scenario in Unity3D platform including asset inventory and UI indicators

instead of Unreal Engine because it is widely regarded as the most accessible game development platform due to its usage of the C# programming language. Unreal Engine, on the other hand, is written in C++, which is a more difficult language to master. Except for the wind turbine and oil rig, which are free 3D models, the setting was entirely developed from scratch. A realistic ocean has been created using water textures and shaders to simulate waves, foam, and movement in response to wind speed and direction, which can be enhanced in the future by adding hydrodynamic models. The scene contains wind turbines and oil rigs to mimic the Hywind Tampen floating wind farm. Users are able to access the user interface in two different modes: operator mode and editor mode, based on their permission levels. Operators can control and set the wind farm and wind condition by using the available panels, sliders, input fields, and buttons as shown in Fig. 7.

The user interface contains widgets utilized for the dashboard such as sliders, input fields, buttons, toggles, charts, etc, while the 3D models used in the inventory panel are wind turbines and oil rigs, as can be seen from Fig. 8.

Users can adjust the turbine geometry as well as the wind speed and the wind direction to conduct interactive simulation, as can be shown in Fig. 9. Furthermore, the users have the ability to switch instantly among four separate data resources. Each data resource can be accessed through a different switch button, as can be seen from Fig. 10. The first mode is Unity3D static data, which contains user-defined parameters in Unity3D that can be utilized and adjusted directly from the user interface and editor to determine the desired output and situation. The second mode is the FMU data, which is the simulated data imported from Matlab Simulink or other simulation applications using the FMI plugin that allows the users to conduct more complicated experiments based on the complex imported models and consequently visualize in the Unity3D platform. The third and most practical technique is the OPC-UA mode, which provides a two-way communication data transferring from the physical asset to the digital asset, allowing the user to conduct diverse experiments and what-if scenarios based on the real-time data and give the command to the physical asset simultaneously that is presenting the main concept of the digital twin. The last mode is the actual data related to actual wind farms' historical data imported from the CSV files to operate semi-realistic scenarios and investigations.

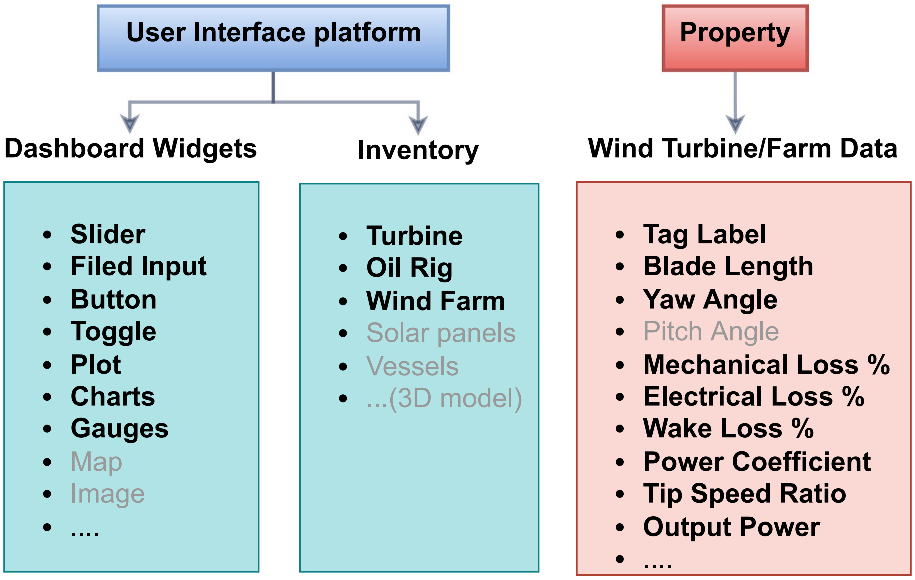
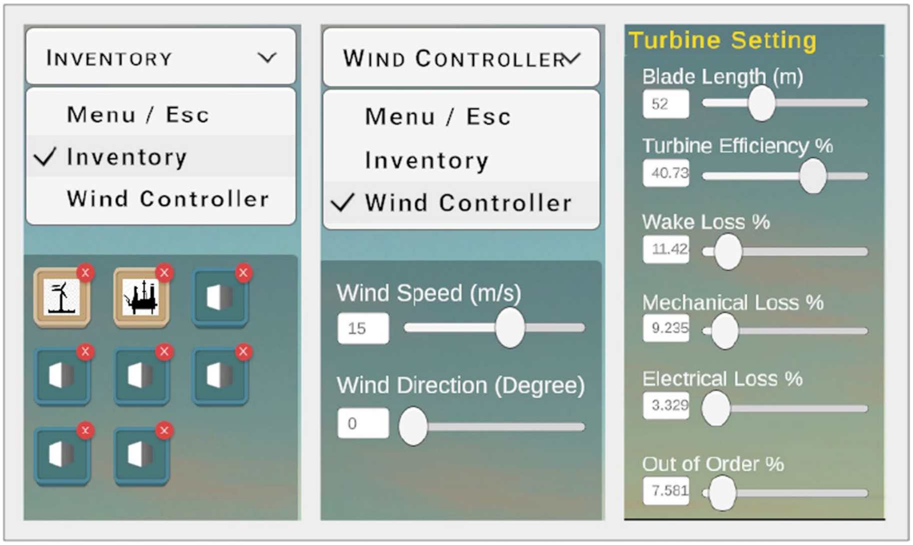


Fig. 8 The components used to implement the User Interface and configuration settings



Wind Speed (m/s) 15

Wind Direction (Degree) 0

Blade Length (m) 52

Turbine Efficiency % 40.73

Wake Loss % 11.42

Mechanical Loss % 9.235

Electrical Loss % 3.329

Out of Order % 7.581

Fig. 9 User ability panels. The left panel is inventory which is used for adding new assets to the scene, the middle panel is the wind controller used for setting the wind conditions, and the right panel is the turbine modification setting which is used to change the turbine's parameters

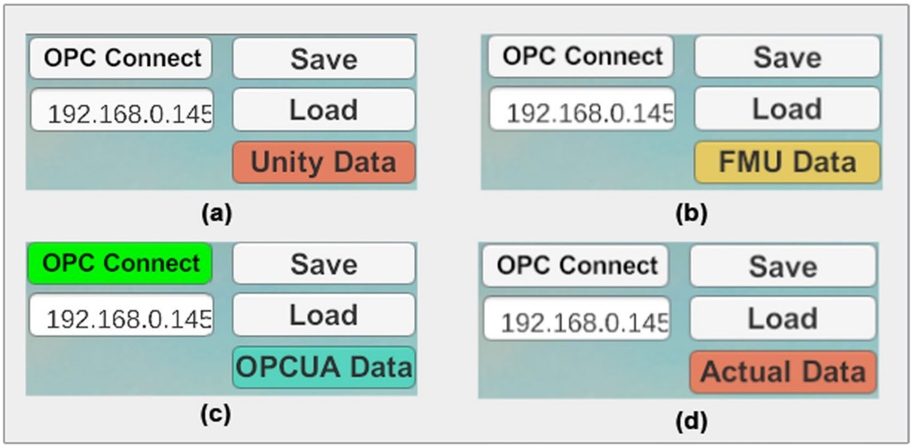
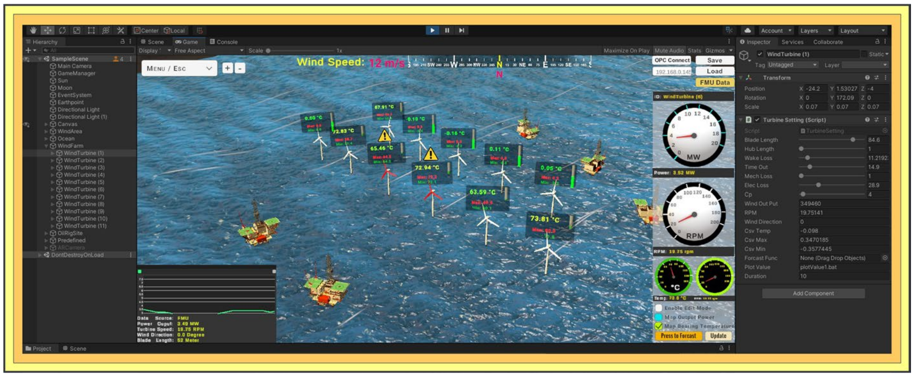


Fig. 10 Data resource switch buttons are shown in the figure for (**a**) Unity data (**b**) FMU data (**c**) OPC-UA (**d**) Live (actual) data

The editor mode provides more configurations and abilities to change the parameters which can give a higher level of access to the user based on the hierarchy. The available editor mode can be seen in Fig. 11.

To represent output power of each wind turbine, a bar indicator of the turbine has been designed to show the current power generation in a green to red colour gradient showing the minimum to maximum power. The bearing vibration and temperature are also shown on small panels on the top of each turbine, showing the current value and the minimum and maximum range of the values, which can be seen from Fig. 12.

In the 3D visualization, the condition indicator panel located on the top of each wind turbine includes the temperature or vibration values, a small cylinder bar shows the value and the colour transitioning from green to red represents the minimum to



Data Source: FMU
Power Ouput: 3.49 MW
Turbine Speed: 19.75 RPM
Wind Direction: 0.0 Degree
Blade Length: 52 Meter

□ Enable Edit Mode
□ Map Output Power
☑ Map Bearing Temperature
Press to Forcast
Update

☑ WindTurbine (1) Static
Tag Untagged Layer

Transform
Position X -24.2 Y 1.53027 Z -4
Rotation X 0 Y 172.09 Z 0
Scale X 0.07 Y 0.07 Z 0.07

☑ Turbine Setting (Script)
Script TurbineSetting
Blade Length 84.6
Hub Length 1
Wake Loss 11.2192
Time Out 14.9
Mech Loss 1
Elec Loss 28.9
Cp 4
Wind Out Put 349460
RPM 19.75141
Wind Direction 0
Csv Temp -0.098
Csv Max 0.3470185
Csv Min -0.3577445
Forcast Func None (Drag Drop Objects)
Plot Value plotValue1.bat
Duration 10

Fig. 11 The editor mode of the user interface used to access more configuration settings based on the user access

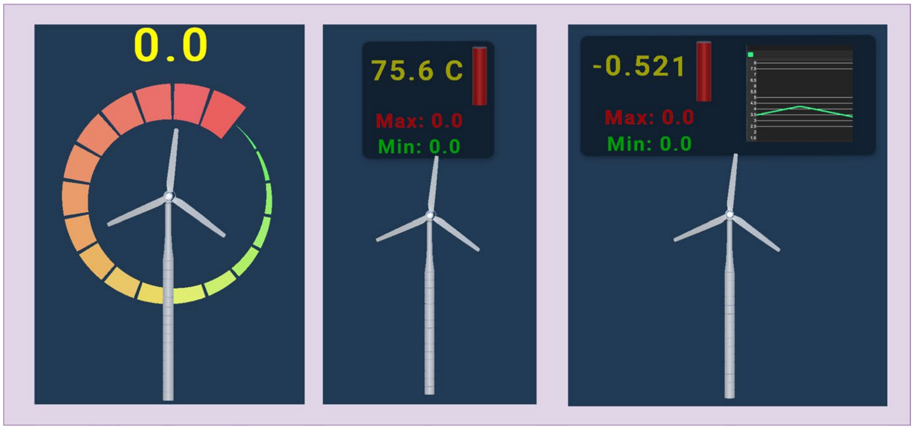


Fig. 12 The left figure shows the power output indicator, the middle one shows the bearing temperature indicator, and the right one shows the bearing vibration indicator

maximum value, respectively. If the temperature or vibration value exceeds the minimal/maximal thresholds, then the turbine blades colour turns to red in case of passing the maximum value and an alarm sign will appear on the top of the turbine, or blue if it goes beneath the minimum value, as can be seen in the Fig. 13.

Predictive twin algorithms

In this section, we describe the algorithm for bearing failure prediction based on temperature and vibration data. The algorithm follows the procedure presented in the previous section. To this end, the Prophet prediction algorithm is used to forecast the bearing condition.

Temperature failure prediction

Algorithm 1 uses historical temperature data to predict the future trend, and hence bearing failure in the future. First, the data is cleaned according to the rules presented in the previous section. Afterward, a performance index is defined from the historical dataset. In this case, the performance index is computed as the average of the gearbox bearing temperature, bearing shaft temperature, and the hub temperature. Once the

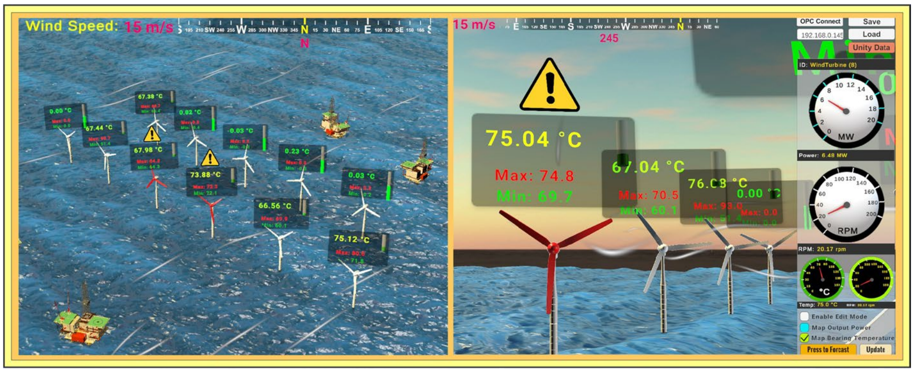


Fig. 13 Condition monitoring of the turbine. If the bearing temperature or vibration exceeds the healthy boundary, the turbine colour turns to red with a warning sign on top of it

performance index is defined, a machine learning method is used to build a model for prediction. The Prophet is used to forecast the future trend of the temperature. Once an anomaly is detected, the algorithm can inform the operator to perform preventive maintenance.

```
Data: Turbine dataset, temperature correlated variables
Result: Predicting the future trend
start: import and read the csv file
foreach $x_i$ captured by sensors do
    if $x_i \in$ timeout then
        Delete
    end
    if $x_i \in$ loss data then
        $x_i = x_{i+1}$ or $x_{i-1}$
    end
    if $x_i <$ 0 or $x_i$ is null then
        $x_i = 0$
    end
end
    • find the correlation rate of variables in the dataset.
    • define the performance index based on the most correlated variables to bearing temperature.
foreach $C_i$ in $Dataset.correlation()$ do
    if $corr(C_i\ ,\ C_{BearingTemp}) > 80\%$ then
        $indx_i = C_i$
    end
end
$Performance\ Index = (\frac{1}{n}\sum_{i=1}^{n} indx_i).normalize$
    • train the model based on a performance index of the 80% of the historical healthy data.
    • post-processing and test the model on the rest 20 % of the historical data.
    • forecasting the future trend using Prophet.
if trend is abnormal then
    anomaly detection.
    performing decision-making and preventive maintenance.
end
if trend is healthy then
    updating the performance index boundary and aggregate with the healthy historical data.
    $NewHealthyBoundary \leftarrow HistoricalData + RealTimeData$
end
end
```

Algorithm 1: Bearing temperature prediction procedure

Vibration failure prediction

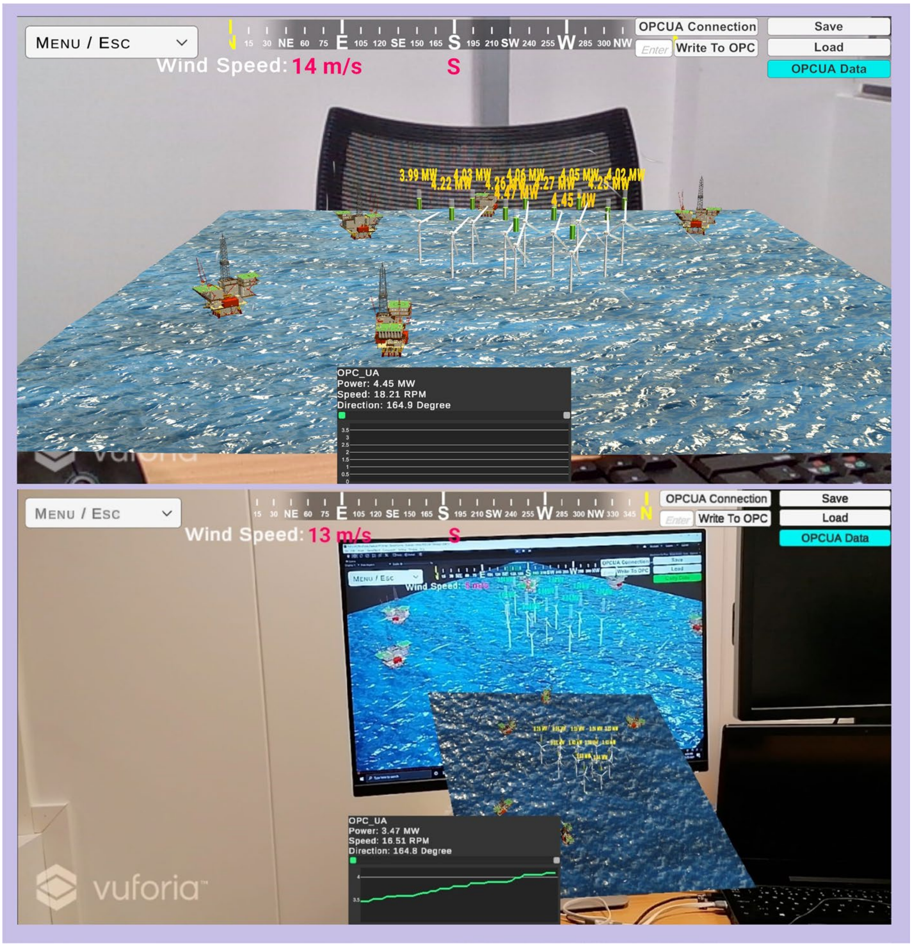
The vibration failure prediction is done in the same way as the temperature prediction. The only difference is when determining the performance index, instead of taking an average of temperatures inside the nacelle, the algorithm uses indicator such as the RMS, Skewness, and Kurtosis, described in the previous section, as can be seen from Algorithm 2. In this case, as we will see in the next section, the RMS provides faster detection.

```
Data: Bearing vibration dataset
Result: Predicting the anomalies and future trend
start: import and read the csv file
foreach variable captured by sensors do
    if $x_i \in$ timeout then
        Delete
    end
    if $x_i \in$ loss data then
        $x_i = x_{i+1}$ or $x_{i-1}$
    end
    if $x_i <$ 0 or $x_i$ is null then
        $x_i = 0$
    end
end
foreach dataset captured in 10 minutes do
    calcualte RMS, Kurtosis and Skewness for N sample points captured in 10 minutes.
    RMS = $\sqrt{\frac{1}{N}\sum_{i=1}^{N} X_i^2}$
    Kurtosis = $\frac{1}{N}\sum_{i=1}^{N}(\frac{X_i - m}{\sigma})^4$
    Skewness = $\frac{1}{N}\sum_{i=1}^{N}(\frac{X_i - m}{\sigma})^3$
end
    • extract the healthy data and define a normal trend for each calculated feature over two weeks
      (RMS, Kurtosis, Skewness).
    • fit the model based on the 80% of the historical healthy data for each feature (RMS, Kurtosis,
      Skewness).
    • post processing and test the model on the rest 20 % of the historical data.
    • forecasting the future trend using Prophet.
if trend is abnormal then
    anomaly detection.
    performing decision-making and preventive maintenance.
end
if trend is healthy then
    updating the normal trend and threshold and aggregate with the healthy historical data.
    $NewHealthyThreshold \leftarrow HistoricalData + RealTimeData$
end
end
```

Algorithm 2: Bearing vibration prediction procedure

Augmented reality for interactive visualization

The proposed solution is equipped with augmented reality in order to improve and facilitate user interaction and abilities. The concept of augmented reality involves blending the digital world with the physical world. This is accomplished by using specialized software to visualize and combine them together, providing a platform for which users can interact more easily. This option enables users to directly access their digital assets via smartphones for obtaining useful data or adjusting physical assets using technologies such as IoT without requiring special hardware or tools. The PTC Vuforia plugin for Unity3D has been used to implement augmented reality. To activate the augmented reality, users need to hold their smartphone or tablet and focus on the image target, which can be a graphic, QR code, or 3D object. Furthermore, a 3D model pops up on the mobile devices and provides augmented reality interaction. The augmented reality platform works simultaneously with other visualization platforms. The implemented augmented reality can be seen in Fig. 14.



OPC_UA
Power: 4.45 MW
Speed: 18.21 RPM
Direction: 164.9 Degree

OPC_UA
Power: 3.47 MW
Speed: 16.51 RPM
Direction: 164.8 Degree

Fig. 14 The augmented reality-based visualization of a digital twin of a wind farm

Experimental setup

Data availability

As mentioned in the previous section, there are several data sources that we can use in the digital twin platform. For condition monitoring of the wind farm, real-time data can be obtained directly from the sensors. For forecasting, since we want to test the predictive algorithm, the temperature and vibration data are obtained from Kaggle, an online data repository to test machine learning algorithms. The temperature data used to find the performance index and to predict the failure are obtained from a real data set from a wind farm located in Gansu Province in China (Zhang 2003). In this case, a SCADA system measures 21 parameters at 10-min intervals. The power rate of each wind turbine is 1800 KW. The key parameters considered in this study are active power, ambient temperature, bearing shaft temperature, gearbox bearing temperature, gearbox oil temperature, generator RPM, generator winding temperature, hub temperature, main box temperature, rotor RPM, and wind speed. The machine learning model will be trained using algorithms to build a relation between the inputs and outputs. Consequently, the quality of the data needs to be tested to ensure that the model accurately represents the system condition. Anomalies in data must be removed from the model to prevent the model from interpreting system performance incorrectly. The vibration data is obtained from NASA dataset (NASA 2022). The dataset consists of four individual files with each containing 20,480 data points with a sample rate set at 20 kHz. According to data gathered after 1 week, a defect in the outer race of bearing 1 was observed at the end of the failure test.

Simulation setup

Real-time data which include wind speed and temperature are collected from the local sensors connected to the Arduino board. The board is connected to a PC by serial ports to send the measured data to the system. The received real-time data is transferred to Unity3D through the OPC-UA protocol using Node-RED. In the Node-RED, a serial port block is added to receive the collected data from the Arduino board and send it to the OPC-UA client block which is connected to the main OPC-UA server. This data can be transferred and used by other OPC-UA clients. There are two clients employed to transfer data among the available platforms. The first one is the client made in Node-RED used to receive the sensor data and communicate to the 2D GUI dashboard. The second one is the client created in C# in Unity3D used for communication to the 3D visualization and Augmented reality platforms. The schematic diagram of the experimental setup is presented in Fig. 15.

Simulation of the real-world wind turbine scenarios is done through the wind energy-related functions in the Unity3D C# scripting. The data is received by the OPC-UA client in Unity3D and it is injected into the defined wind energy functions and converted to the variables which are used for configuration and adjusting the desired scenario. To provide the natural wind farm scenario, the historical data is read from the CSV files to make the artificial presentation of the bearing temperature and vibration of the wind turbines. Each wind turbine has its own temperature and vibration changing per time

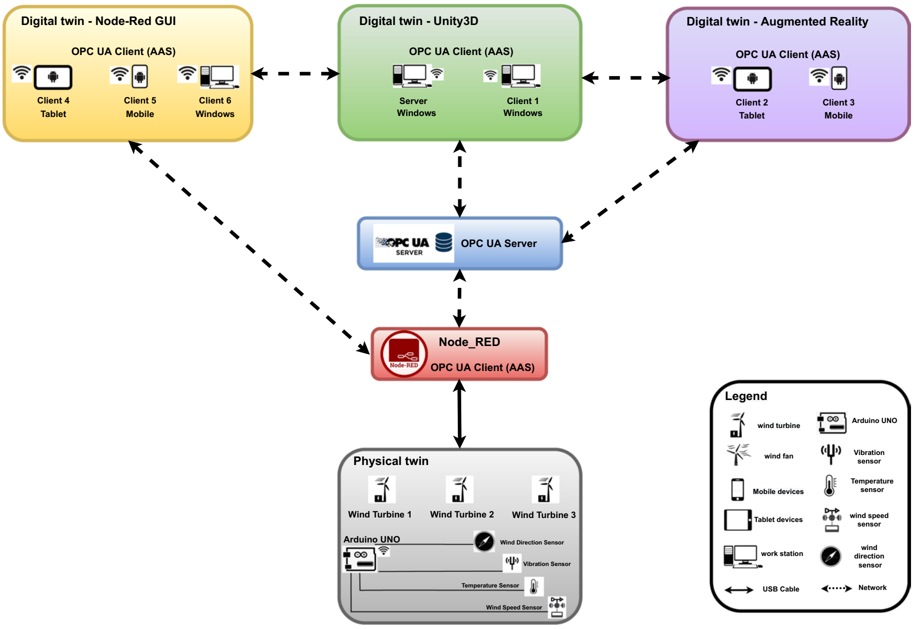


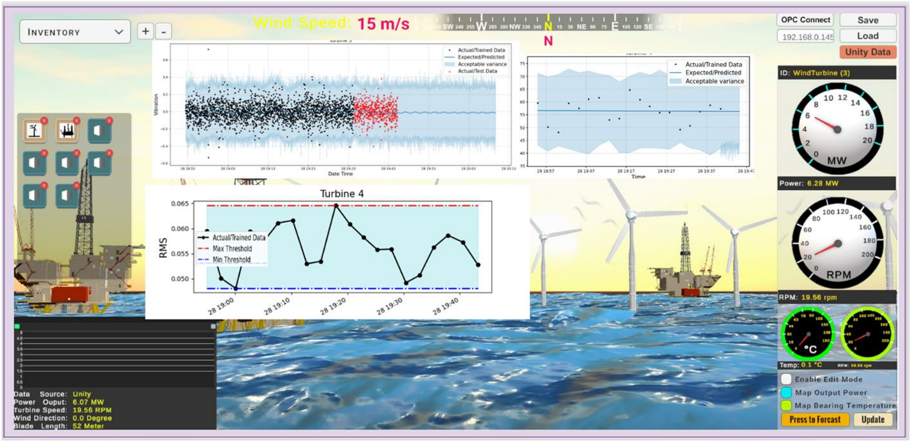
Fig. 15 The schematic diagram of the experiment

interval set by the user. By starting the Unity3D, the system starts to read the bearing temperature and vibration data from the CSV files and sets it for each turbine. Moreover, a realistic system will be provided to apply different what-if scenarios and prediction methods showing the wind turbine bearing condition and other desired information.

Result and discussion

To predict the bearing condition of each individual wind turbine in the current scenario running in Unity3D, a forecast button is available on each wind turbine setting panel, which can be pressed by the user to start predicting the bearing temperature or vibration by considering the current situation and historical data. By pressing the forecast button, Unity3D triggers the Python script which runs the prediction methods for that turbine and starts to train the machine learning model based on the data collected on each turbine and represents the output forecast in a chart, as can be seen from Fig. 16, along with the healthy thresholds. All the prediction calculations and output charts are presented by Python scripts.

Observing the result of the vibration predictions, it is apparent that either the RMS, Kurtosis, or Skewness show an abnormal state before a serious bearing problem occurs. The RMS function appears to be the most accurate and sensitive. It indicates bearing anomaly for more than one day before the actual failure, while the Kurtosis and Skewness reveal the abnormal distribution in less than one day, as can be seen from Fig. 17. Therefore, the RMS function has been selected as the main vibration performance index in this paper. The process is first by calculating the RMS value for a specific time interval of the historical healthy data, then find the acceptable variance and set the minimum and maximum thresholds. Afterward, the real-time data collected from the sensors are directly used by prediction algorithms to be evaluated against the healthy data and check if they are within the healthy thresholds. The RMS of data was calculated by using a percentage interval which was set at 5% of the length of the whole sample. Consequently, regardless of the size of the data set, the system calculates the RMS value of the bearing vibration data at any time interval, regardless of how much data is available.



Wind Speed: 15 m/s

192.168.0.145

ID: WindTurbine (3)

Power: 6.28 MW

RPM: 19.56 rpm

Temp: 0.1 °C

□ Enable Edit Mode
□ Map Output Power
□ Map Bearing Temperature

Data Source: Unity
Power Ouput: 6.07 MW
Turbine Speed: 19.56 RPM
Wind Direction: 0.0 Degree
Blade Length: 52 Meter

Fig. 16 The system predicts the bearing condition based on the collected data from each wind turbine once the forecast button is pressed, and displays the results on the screen

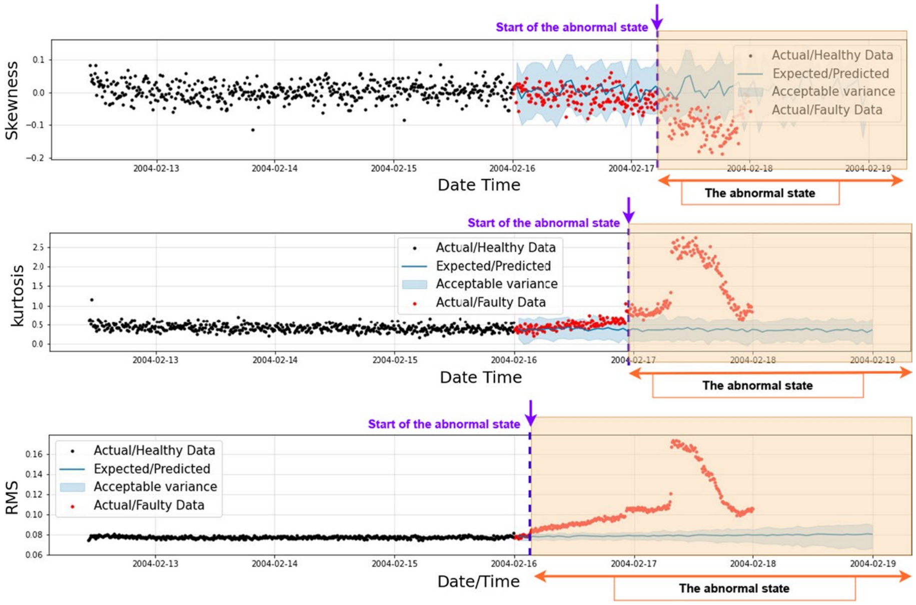


Fig. 17 The comparison between the RMS, Kurtosis, and Skewness

For the temperature prediction, the performance index was selected based on the values that are most correlated, which are the gearbox bearing temperature, bearing shaft temperature, and hub temperature. Each of these temperatures correlates with the variables of the wind energy system. The gearbox bearing temperature, for example, correlates strongly with the power output, wind speed, rotor speed, and oil temperature, as can be seen from Fig. 18. Basically, the system computes the average of the correlated values per day, clears the outliers, normalizes the result, and assigns that value to represent a healthy index. It is beneficial to take the daily average temperature since the method works better for a daily interval and produces more useful results than conducting the forecast in smaller time intervals. The prediction result is shown in Fig. 19. As can be seen, the future trend (blue line) is upward, and the actual values are increasing towards the end of the simulation, indicating that the forecast was correct, and that the failure can be predicted before it occurred.

The main purpose of this paper is to create a predictive digital twin platform that can be used for real-time predictions and to schedule adequate maintenance to mitigate risks and costs associated with the downtime. The implemented prediction techniques are based on the best available solutions and methodology, and they work well in the proposed predictive digital twin platform. However, all the prediction methods can be improved in the future by including more robust solutions.

Conclusion

In this paper, a predictive digital twin platform for wind farms is proposed. The striking feature of the digital twin platform compared to the SCADA system is the ability to predict failure. In this case, we have added the Prophet prediction algorithm for wind turbine component failures. The platform allows users to collect, visualize, and analyze data in real-time to improve predictive capabilities, enable better decision

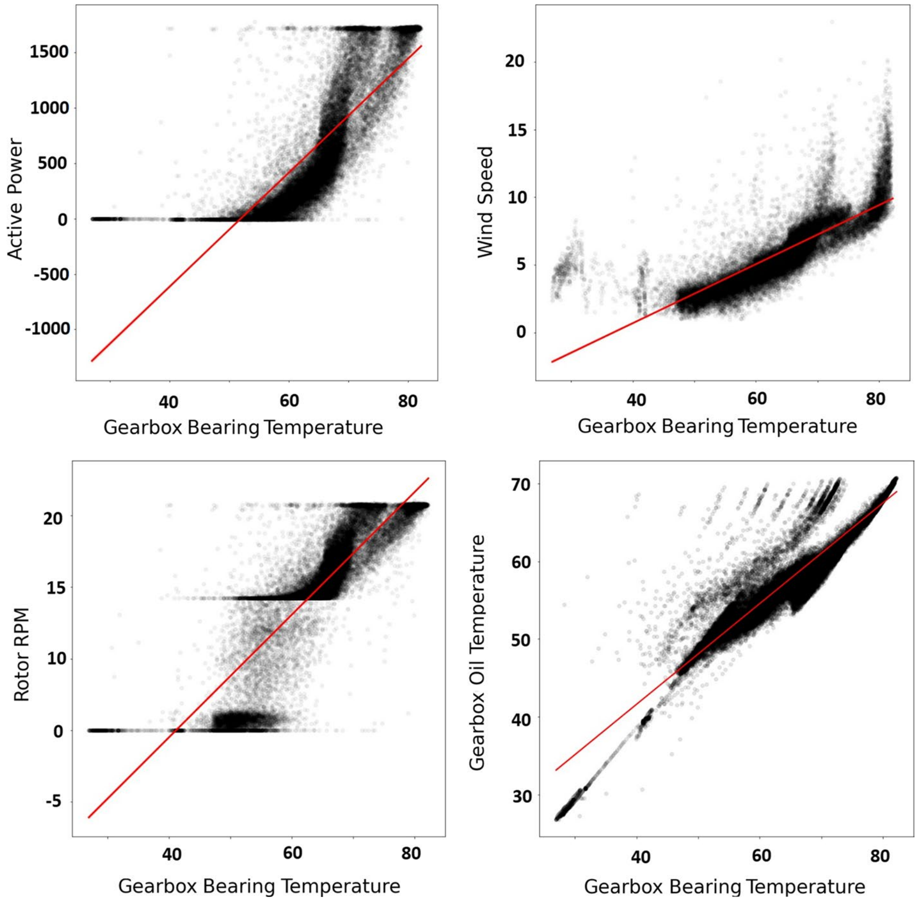


Fig. 18 The correlation diagram of the gearbox bearing temperature to the most related parameters

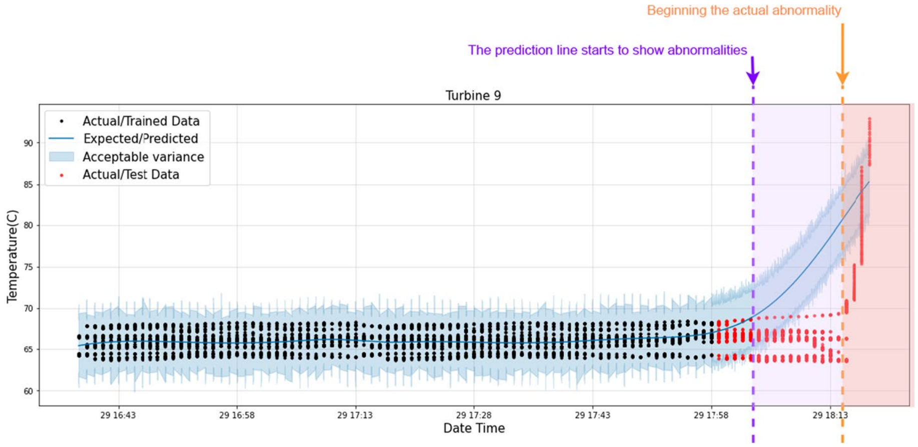


Fig. 19 The failure prediction based on temperature measurement

making, reduce potential failures, and improve reliability. The proposed platform is based on the OPC-UA and Unity3D and commences by collecting real-time data from sensors. Different result presentations are offered through 2D and 3D visualization, and augmented reality, which can be chosen depending on the desired objectives and requirements. To test the digital twin platform, we consider failure predictions of the wind turbine bearings. To this end, we use the configuration of the Hywind Tampen floating wind farm. In this case, the platform, which is equipped with the Prophet prediction algorithm, utilizes vibration and temperature data to monitor and predict the failures. The prediction method yields decent results by employing performance indices obtained from experiments and research studies.

Since the platform is developed based on the OPC-UA, it can be adopted and integrated directly by energy companies in their existing system. Indeed, the Norwegian energy company Equinor ASA, for example, has been looking into this solution (see funding information). Furthermore, currently the platform is used in teaching and research at the Department of ICT and Natural Sciences NTNU. The proposed framework can be enhanced in terms of efficiency and robustness by adding and implementing calculations for other components of the wind turbine. Limitations of the platform include: (i) a lack of support for specific features like Electronic Signature, Enhanced Failover, and historical data sources, and (ii) The Unity 3D has poor source control integration and large team tools. Improving the forecast results can be done by updating the prediction algorithms and finding a good performance index based on the correlations of the variables. There is room to improve its capability by fine-tuning different parameters in the framework to obtain optimal parameters for various settings. Furthermore, it is possible to design a customized interface to improve the quality of the visualization and to provide a better user experience.

Abbreviations

OPC-UA	Open platform communications-unified architecture
SCADA	Supervisory control and data acquisition
RUL	Remaining useful life
ARIMA	Auto regressive integrated moving average
NN	Neural network
SVM	Support vector machine
KNN	K-nearest neighbor
AENN	Auto encoder neural network
ANN	Artificial neural network
RMS	Root mean square
NCS	Norwegian continental shelf
GUI	Graphical user interface
API	Application programming interface
CSV	Comma-separated values
FMU	Functional mockup unit
FMI	Functional mockup interface
RPM	Rotation per minute
IoT	Internet of things

Acknowledgements

Not applicable.

Author contributions

AH (Amir) performed data analysis and was a major contributor in writing the manuscript, AH (Agus) proposed the idea, analyzed the data, and interpreted the results, OO proposed the method, and EM analyzed the results. All authors read and approved the final manuscript.

Funding

Open access funding provided by Norwegian University of Science and Technology. This work is partially supported by Equinor ASA.

Availibility of data and materials

The datasets used and/or analysed during the current study are available from the corresponding author on reasonable request.

Declarations

Ethics approval and consent to participate
Not applicable.

Consent for publication
Not applicable.

Competing interests
The authors declare that they have no competing interests.

Received: 14 October 2022 Accepted: 17 January 2023
Published online: 25 January 2023

References

Boschert S, Rosen R (2016) Digital twin—the simulation aspect. In: Hehenberger P, Bradley D. (eds.) Mechatronic Futures, pp. 59–74. Chap. 8

Botz M, Raith M, Emiroglu A, Wondra B, Grosse C (2019) Advances in engineering materials, structures and systems: Innovations, mechanics and applications. In: Zingoni A. (ed.) International Conference on Structural Engineering, Mechanics and Computation (SEMC 2019), pp. 1971–1975. Chap. 8

Dai J, Yang W, Cao J, Liu D, Long X (2018) Ageing assessment of a wind turbine over time by interpreting wind farm scada data. Renewable Energy 116:199–208

Dong S, Yin S, Tang B, Chen L, Luo T (2014) Bearing degradation process prediction based on the support vector machine and markov model. Shock Vib 2014:1–15

Eftekharnejad B, Carrasco MR, Charnley B, Mba D (2011) The application of spectral kurtosis on acoustic emission and vibrations from a defective bearing. Mech Syst Signal Process 25(1):266–284

Elasha F, Shanbr S, Li X, Mba D (2019) Prognosis of a wind turbine gearbox bearing using supervised machine learning. Sensors 19(4):1–17

Foley AM, Leahy PG, Marvuglia A, McKeogh EJ (2012) Current methods and advances in forecasting of wind power generation. Renewable Energy 37:1–8

Fuller A, Fan Z, Day C, Barlow C (2020) Digital twin: enabling technologies, challenges and open research. IEEE Access 8:108952–108971

Garlick W, Dixon R, Watson S (2009) A model-based approach to wind turbine condition monitoring using scada data. 20th International Conference on Systems Engineering, Coventry, UK . Coventry University

Glaessgen E, Stargel D (2012) The digital twin paradigm for future NASA and us air force vehicles. 53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference, Hawai, USA. NASA

Jana RK, Ghosh I, Wallinc MW (2022) Taming energy and electronic waste generation in bitcoin mining: insights from facebook prophet and deep neural network. Technol Forecast Soc Chang 178:121584

Jin X, Chen Y, Wang L, Han H, Chen P (2021) Failure prediction, monitoring and diagnosis methods for slewing bearings of large-scale wind turbine: a review. Measurement 172:108855

Jones D, Snider C, Nassehi A, Yon J, Hicks B (2020) Characterising the digital twin: a systematic literature review. CIRP J Manuf Sci Technol 29:36–52

Jose C, Goyal P, Aggrwal P, Varma M (2013) Local deep kernel learning for efficient non-linear svm prediction. 30th International Conference on Machine Learning, Georgia, USA

Kooning J, Stockman K, De-Maeyer J, Jarquin-Laguna A, Vandevelde L (2021) Digital twins for wind energy conversion systems: a literature review of potential modelling techniques focused on model fidelity and computational load. Processes 9:1–26

Kusiak A, Zhang Z, Verma A (2013) Prediction, operations, and condition monitoring in wind energy. Energy 60:1–12

Lin HC, Ye YC (2019) Reviews of bearing vibration measurement using fast fourier transform and enhanced fast fourier transform algorithms. Adv Mech Eng 11(1):1–12

Liu Z, Zhang L (2020) A review of failure modes, condition monitoring and fault diagnosis methods for large-scale wind turbine bearings. Measurement 149:107002

Liu M, Fang S, Dong H, Xu C (2021) Review of digital twin about concepts, technologies, and industrial applications. J Manuf Syst 58:346–361

Menculini L, Marini A, Proietti M, Garinei A, Bozza A, Moretti C, Marconi M (2021) Comparing prophet and deep learning to arima in forecasting wholesale food prices. Forecasting 3:644–662

Moghadam FK, Rebouças GFS, Nejad AR (2021) Digital twin modeling for predictive maintenance of gearboxes in floating offshore wind turbine drivetrains. Forsch Ingenieurwes 85:273–286

Mühlbauer N, Kirdan E, Pahl M-O, Carle G (2020) Open-source opc ua security and scalability. 2020 25th IEEE International Conference on Emerging Technologies and Factory Automation, Vienna, Austria. IEEE

NASA: NASA Bearing Failure Datasets, (2022)

Oñederra O, Asensio FJ, Eguia P, Perea E, Pujana A, Martinez L (2019) Mv cable modeling for application in the digital twin of a wind farm. 2019 International Conference on Clean Electrical Power (ICCEP), Otranto, Italy. IEEE

Perzylo A, Profanter S, Rickert M, Knoll A (2019) Opc ua nodeset ontologies as a pillar of representing semantic digital twins of manufacturing resources. 2019 24th IEEE International Conference on Emerging Technologies and Factory Automation, Zaragoza, Spain. IEEE

Pimenta F, Pacheco J, Branco C, Teixeira C, Magalhães F (2020) Development of a digital twin of an onshore wind turbine using monitoring data. J Phys Conf Ser 1618:1–11

Rasheed A, San O, Kvamsdal T (2020) Digital twin: values, challenges and enablers from a modeling perspective. IEEE Access 8:21980–22012

Ren L, Sun Y, Cui J, Zhang L (2018) Bearing remaining useful life prediction based on deep autoencoder and deep neural networks. J Manuf Syst 48:71–77

Sivalingam K, Sepulveda M, Spring M, Davies P (2018) A review and methodology development for remaining useful life prediction of offshore fixed and floating wind turbine power converter with digital twin technology perspective. 2nd International Conference on Green Energy and Applications (ICGEA), Singapore. IEEE

Stojanovic L, Usländer T, Volz F, Weißenbacher C, Müller J, Jacoby M, Bischoff T (2021) Methodology and tools for digital twin management–the FA3ST approach. IoT 2(4)

Tavner P, Greenwood D, Whittle M, Gindele R, Faulstich S, Hahn B (2013) Digital twin: values, challenges and enablers from a modeling perspective. Wind Energy 16:175–187

Taylor SJ, Letham B (2017) Forecasting at scale. Peer J 5:3190–2

Tenggren E, Olsen LE, Johansen DA, Bracchi T (2020) A numerical study on the effect of wind turbine wake meandering on the power production of hywind tampen. J Phys Conf Ser 1669:012026

Verdouw C, Beulens A, Reijers H, van der Vorst J (2015) A control model for object virtualization in supply chain management. Comput Ind 68:116–131

Verdouw C, Tekinerdogan B, Beulens A, Wolfert S (2021) Digital twins in smart farming. Agric Syst 189:1–19

Wang M, Wang C, Hnydiuk-Stefan A, Feng S, Atilla I, Li Z (2021) Recent progress on reliability analysis of offshore wind turbine support structures considering digital twin solutions. Ocean Eng 232:109168

Zhang GP (2003) Time series forecasting using a hybrid arima and neural network model. Neurocomputing 50(1):159–175

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.