Norwegian University
of Life Sciences

Master's Thesis 2023   30 ECTS
Faculty of Science and Technology

A Supportive Framework for the Development of a Digital Twin for Wind Turbines Using Open-Source Software

Tiril Malmedal
Mechanics and Process Technology

A Supportive Framework for the Development of a Digital Twin for Wind Turbines Using Open-Source Software

May 15, 2023

Abstract

The world is facing a global climate crisis. Renewable energy is one of the big solutions, nevertheless, there are technological challenges. Wind power is an important part of the renewable energy system. With the digitalization of industry, smart monitoring and operation is an important step towards efficient use of resources. Thus, Digital Twins (DT) should be applied to enhance power output.

Digital Twins for energy systems combine many fields of study, such as smart monitoring, big data technology, and advanced physical modeling. Frameworks for the structure of Digital Twins are many, but there are few standardized methods based on the experience of such developed Digital Twins.

An integrative review on the topic of Digital Twins with the goal of creating a conceptual development framework for DTs with open-source software is performed. However, the framework is yet to be tested experimentally but is nevertheless an important contribution toward the understanding of DT technology development.

The result of the review is a seven-step framework identifying potential components and methods needed to create a fully developed DT for the aerodynamics of a wind turbine. Suggested steps are Assessment, Create, Communicate, Aggregate, Analyze, Insight, and Act. The goal is that the framework can stimulate more research on digital twins for small-scale wind power. Thus, making small-scale wind power more accessible and affordable.

Acknowledgements

The period has been many things; interesting, fun, frustrating, and all-consuming at times. My hope and motivation have been that my work and time used on this paper will be helpful for others to take the next step and make a digital twin prototype and that it can be further improved. In the end, I have learned a great deal about the writing process and the vast digital twin domain. It has been exceedingly exciting seeing the thesis form the final product.

I want to thank my supervisor, Muhammad Salman Siddiqui, for providing the intriguing topic and the supporting, valuable, and helpful counseling throughout the writing process.

Thank you to Jotne Connect for an insightful workshop that contributed to a further understanding of data modeling and standards in actual use cases and for the time and interest.

My fellow students at the reading hall must not be forgotten, making all the long days at school social and memorable. I am grateful for all the support from my family and friends throughout this work, and a special thanks to my roommate for being my guardian angel.

Ås, May 2023
Tiril Malmedal

List of Figures

List of Tables

Abbreviations

Table 1: Abbreviations used in the thesis.

5G NG-RAN	5G Next Generation-Radio Access Network
AOA	Angle of Attack
API	Application Programming Interface
BEM	Blade Element Momentum
CAD	Computer-Aided Design
CFD	Computational Fluid Dynamics
DDM	Data-driven Modeling
DT	Digital Twin
DTOP	Digital Twin Operational Platform
FEM	Finite Element Metod
FNN	Feed-forward Neural Network
FVM	Finite Volume Method
HPC	High Preformance Computing
IoT	Internet of Things
LSTM	Long Short-Term Memory
MCU	Microcontroller Unit
OSS	Open-Source Software
OWT	Offshore Wind Turbine
PBM	Physics-based Modeling
PCB	Printet Circuit Board
PLM	Product Lifecycle Management
RANS	Reynolds Average Navier Stokes
REST	Representational State Transfer
SRF	Single Reference Frame
SST	Shear Stress Transport
SSWT	Small-Scale Wind Turbine
STEP	Standard for The Exchange of Product Model Data
SWT	Small Wind Turbine
TSR	Tip Speed Ratio

Contents

Chapter 1

Introduction

Climate change poses a threat to life on earth, and the need for near-time actions is crucial[1]. The path towards a low carbon society are long, and renewable energy is pointed out as one of the major contributions towards this goal [1]. In contrast to non-renewable energy which is derived from static stores of energy, renewable energy is derived from renewable sources of energy flows, wind being one of the major renewable categories [2]. In IPCC's Mitigation Climate Change report [1], wind and solar energy are identified as the technologies with the highest potential to reduce greenhouse gas emissions with the lowest cost[1]. According to the strategy of Norwegian University of Life Sciences (NMBU) for 2019-2023[3] we need international collaboration within sustainable technology research to reach the United Nations sustainable development goals.

Wind technology is the harvesting of kinetic energy from wind by turbines [2]. Consequently, wind turbines are only able to operate under windy conditions, and are therefore a volatile, yet never ending, energy source [2]. In Norway, wind and other energy sources are growing. The installed Solar PV power has grown exponentially, with approximately 2400 installed systems in 2021[4], while the industrial wind also grow concurrently with lower installation costs[5]. Small-scale systems of wind technology and PV can complement each other, as they produce energy for different operating conditions. In 2013 the Norwegian government opened for small-scale energy production without the need for a concession to encourage small-scale wind[6]. As a further means to increase interest in small-scale wind, new and efficient digital technology can make it easier to implement and operate such energy systems.

The term Digital Twin (DT) was first used in the work on Information Mirroring Model for Product Lifecycle Management in 2002, and was later adopted by NASA in 2010[7]. In short terms, a Digital Twin is a digital system mirroring a physical system [8]. In the later years, Digital Twin concept and technology has been used in the context of Industry 4.0 - the digitalization of industry [8]. Creating a digital twin for a wind turbine could be beneficial for a number of reasons. Before deploying the turbine, a standalone DT can support an assessment of the local environment and help ensure correct power production and virtualize the specific case[9][10]. During wind turbine operation, a DT can contribute by monitoring the aerodynamic properties of the wind turbine in real-time and visualizing the turbine while automatically providing critical information without the need for physical inspection of the turbine [10]. Algorithms trained on the system's data could help predict when the turbine needs maintenance, and not risk failure leading to considerable damage.

The development of digital twin technology depends on a collaborative effort from many different stakeholders, such as industry and academia. In a review of digital twins in the wind energy sector[10], the authors recommend that academia and research institutes make the development of both virtual and predictive twins usable for society through open-source software. In [11] the author calls for collaboration in building machine learning models like open-source software, and refers to Python programming language as a good example of how a big community can further develop a tool, adding new features and bug fixing after its release[11]. A fully capable digital twin with high-fidelity modeling may take years to develop due to it's complexity[12]. If the digital twin software and framework are an open resource for everyone to develop, the time will be reduced in comparison with every firm making their proprietary digital twin from scratch. It is therefore, interesting to use technology that is accessible to any interested party.

Interoperable Europe emphasizes the importance of interoperability in the EU public sector to enable the exchange and usage of information across borders. Using open-source software (OSS) and sharing work on web-based databases will help enable collaboration independent of location [13]. To help this development, the EU created Joinup as a platform to openly share and access interoperable and free open-source software[13]. Despite this effort, an article title search for “wind turbine,” “digital twin,” and “open-source”, using the search engines Google Scholar and Scopus, give 0 and 2 results, respectively. This indicates that few published works directly focus on open-source software for developing digital twins for a wind energy system. Thus, investigating the existing literature regarding open-source software for modeling of wind turbine technology and digital twins will address this gap.

1.1 Objective

The challenges related to interoperability for digital twins and the definition above have led to the research objective and research questions presented below. Through a literature review, this thesis will try to identify the different steps to make a DT for wind energy and identify the open-source software and tools that can contribute to enabling collaborative digital twin development.

1.1.1 Primary objective

The main goal of this thesis is to propose a framework for developing a high-fidelity digital twin for a roof-mounted wind turbine utilizing open-source software.

1.1.2 Research questions:

1. What are the components of a high-fidelity digital twin for a wind turbine?
2. What open-source software can be used to realize the components of the Digital Twin?

1.1.3 Thesis Contribution

1. A framework for developing a digital twin for the aerodynamics of wind turbines based on a literature review.
2. Show a range of open-source software as enabling tools for development through findings in a selection of literature.

1.2 Thesis Structure

The thesis is further structured in four sections; Section 2 , present relevant topics for defining digital twins, and identifies the development steps. Section 3 describes the methods used to collect information for the thesis. Section 4 presents the findings obtained through an integrative literature review. Section 5 summarizes the findings and provides concluding remarks.

Chapter 2

Theory

This chapter aims to create a foundational understanding of the Digital Twin framework to support the results from the literature review in Chapter 4. It is divided into 9 sections, where Section 2.1 and 2.2 describe the Digital Twin concept, and Section 2.3 to 2.9 describes the steps for developing a DT.

2.1 Digital Twin

2.1.1 Concept and Definitions

Established definitions related to the Industry 4.0 and Digital Twin are hard to identify as many fields of applications exist. There exist different requirements and expectations of what a digital model should be capable of, before being classified as a digital twin. Common criteria are real-time data exchange between the physical and digital system components and the ability to analyze and use the data to optimize the digital and physical systems. Some terms are often used to define the different nuances of digital models;

Digital Model The digital model is defined as a virtualization of the physical asset that can accurately describe its predefined set of behaviors. The digital model has no data exchange with the physical counterpart, and new information needs to be added manually[14]. It is also mentioned that there are no requirements for the model fidelity.

Digital Shadow A digital Shadow accurately depicts the physical asset or process with automated near-time data flow from the physical system and has the capability to evaluate the data from a complete database[15][14].

Digital Sibling Digital Sibling is a term used when talking about risk assessment and "what-if" analysis. In [10] it is defined as the offline mode of the Digital Twin running scenario analysis.

Digital Thread The Digital Thread concerns the product information gathered by the Digital Twin throughout the product's lifecycle, which is connected to Product Lifecycle Management (PLM). The Digital Thread can be used in developing the next generation of the product, holding information about the physical asset performance and design[10].

Digital Twin

In [10], the definition of a digital twin is presented by consensus among many authors and industrial partners and reads as follows:

"A digital twin is defined as a virtual representation of a physical asset or a process enabled through data and simulators for real-time prediction, optimization, monitoring, control, and informed decision-making."

This definition's phrasing implies that the physical asset is not a part of the digital twin system. In [16], however, the physical space is included in the Digital Twin, with key features like data acquisition, pre-processing, and actuators.

2.2 Digital Twin Framework

A Digital Twin framework is the representation of the Digital Twin's components and structure. In this thesis, different DT frameworks are compared to make a supportive overview of the development methods of such DTs. Different approaches are used in building DT frameworks, and three separate frameworks are described in the following sections. In addition, a DT can be described by its capabilities in a hierarchical leveled system. This system is described in the final subsection.

Dimension-based

The architecture of a digital twin can be described through a dimensional model, and this has been done in many different ways. Three examples of various dimension model are shown in Figure 2.1, where the architecture of the DT concept is divided into a) three dimensions, b) four dimensions, and c) five dimensions. Though a bit different, all three variations have some core components, being the physical system, the data system, and communication between the different dimensions, allowing the data and information to flow between every part of the Digital Twin system. The three-dimensional framework does not contain a virtual component and does not fully represent a DT defined as being a virtual replica.

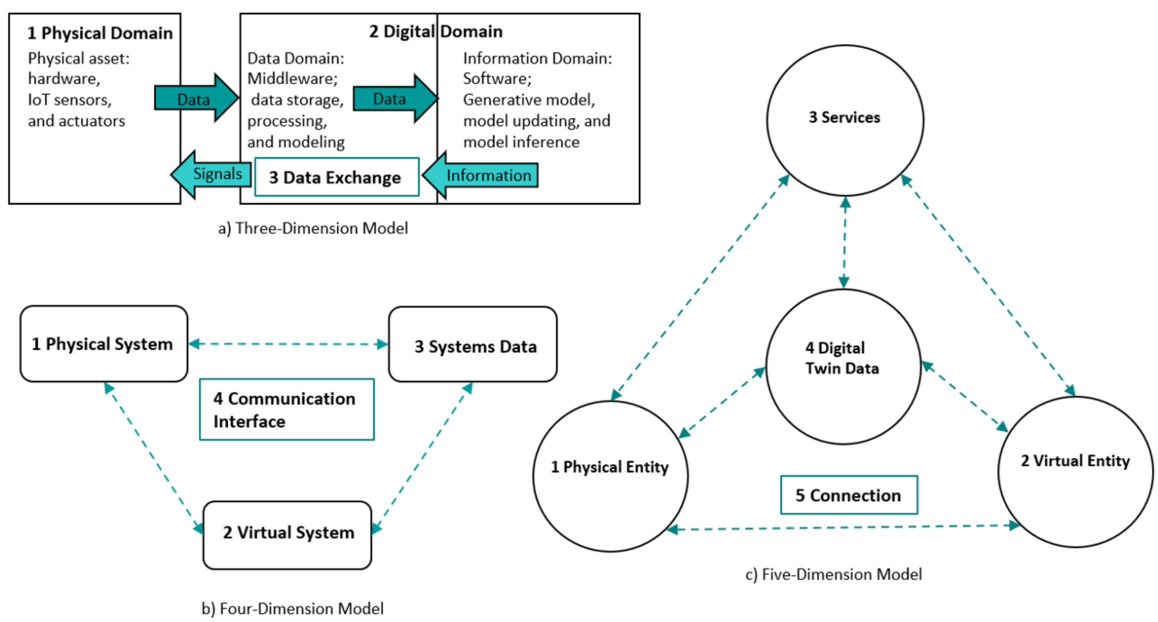

Figure 2.1: Three different dimension models for digital twin architecture, adapted from a) the three-dimensional model in [8, p. 17], b) the four-dimensional model in [8, p. 4], and c) the five-dimensional model in [17, p. 5].

The Physical System represents the physical assets or technologies that are subject to being modeled in the DT. The Virtual System is the real-time model and simulation of the assets in PS. Here, the DT

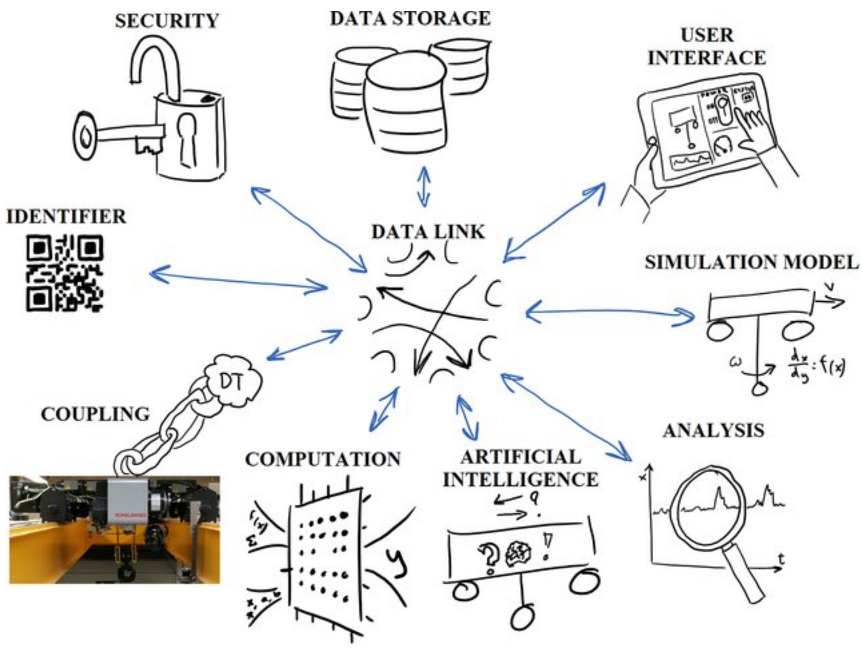

Figure 2.2: Conceptual figure of the Feature-Based DT Framework reused from [12].

Systems Data is defined as the entity that stores data, both from the DT model and other data sources. The five dimension model [18] also contains a Services component, which includes the functions for both the users of the system and internal functions to support the DT.

Feature-based

To develop a Feature-based Digital Twin Framework, authors in [12] have identified ten separate features in Digital Twins, listed below:

1. **Data Link:** A hub for all information related to the DT.
2. **Coupling:** Two-way interface between the physical product and the Digital Twin. (Here, the physical asset is viewed as not a part of the DT.)
3. **Identifier:** Physical identifiers serve as gateways between the physical assets and their DT. Digital identifiers connect the DTs to the network (Internet).
4. **Security:** Appropriate security level should be embedded in the DT by design, based on a security risk analysis done in advance of DT development.
5. **Data Storage:** The feature stores large amounts of data in a way that enables fast and easy access and communication through the Data Link.
6. **User Interface:** The tool that allows the human user to interact with the DT.
7. **Simulation:** Modeling the physical asset or system in a way that describes its visual, graphical, and/or numerical essence.
8. **Analysis:** Analyses on monitoring or simulation data, e.g., for correlation or sensitivity, provides the user or AI with information supporting decision-making.
9. **Artificial Intelligence:** Making autonomous decisions based on data and the information provided by analysis.
10. **Computation:** Generates data by solving mathematical tasks through either local edge computing or global data processing hubs.

Entity-based

An Entity-based DT framework from the standard ISO23247 is mapped to DT-implemented technologies in manufacturing in the work of Ferko et.al [**ferko˙supporting˙nodate**]. This framework builds on an architecture of entities, sub-entities, and functional entities in hierarchical order, as shown in Figure 2.3. The physical system, consisting of the Observable Manufacturing Elements, lies outside the DT framework, as in the aforementioned framework. In the DT framework, there are four entities; Device Communication Entity, Digital Twin Entity, User Entity, and Cross-System Entity. The Functional Entities in the Cross-System Entity take care of the data management, like translating and security measures, and therefore span the other entities. The all-embracing data and security entities

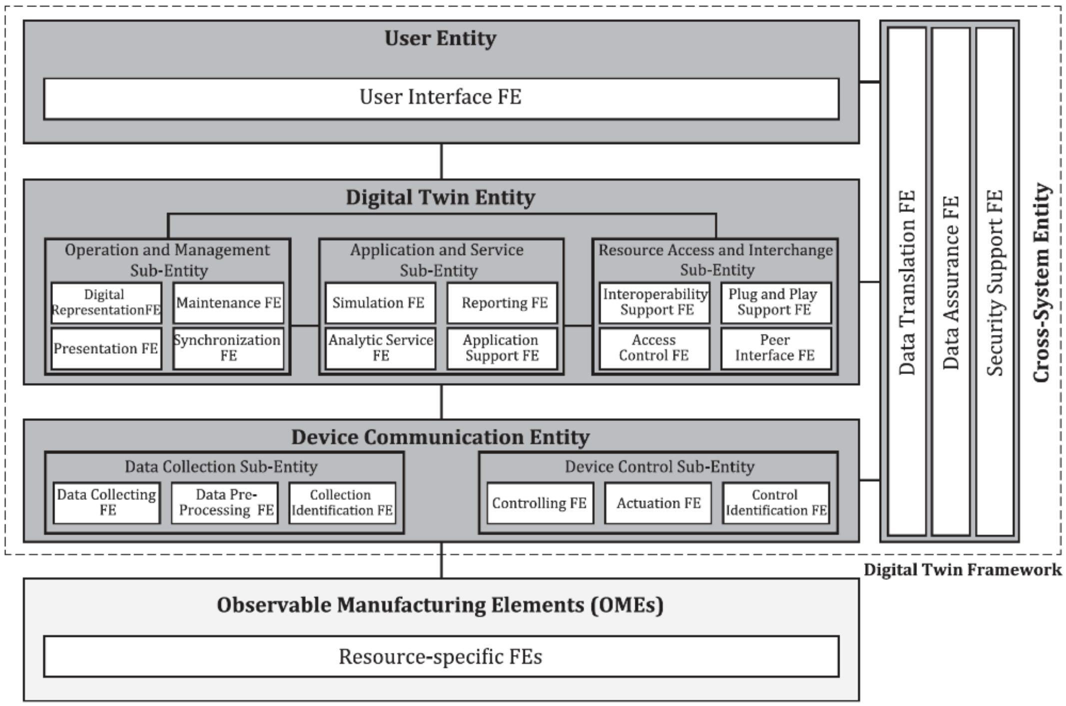

Figure 2.3: Figure redistributed from [19], with showing the Entity-based Digital Twin Framework from the standard for Digital Twins in manufacturing ISO23247. With Permission to use:(Figur 3 - Entity-based digital twin reference model for manufacturing fra ISO 23247-2:2021 Automation systems and integration — Digital twin framework for manufacturing — Part 2: Reference architecture er gjengitt av [Tiril Malmedal] til bruk i oppgaven [A Supportive Framework for the Development of a Digital Twin for Wind Turbines Using Open-Source Software] med tillatelse fra Standard Online AS i mai 2023. Standard Online er ikke ansvarlig for eventuelle feil i gjengitt materiale. Se www.standard.no)

Capability levels

In [20], the digital twin results and the developer's capabilities are required to ensure acceptable quality. The requirements for the digital twin regard sufficiently good quality for both the data, computation model(s), the presentation of results in the user interface, and the functionality of the digital twin throughout its lifetime. In DNVs recommended practice *Qualification and assurance of digital twins*[20] Functional Elements in Digital Twin's capability are divided into 6 distinct levels, from 0 to 5, described in table 2.1 and related to a Wind Turbines potential DT capabilities.

Table 2.1: The table relates general, and shortened, definitions for capability levels from DNVs report[20] to a Wind Turbine specific DT.

Capability Level	FE functions DNV [20]	Functions related to WT
0 Standalone	Description of Physical System without being connected to real-time data. Data modeling for contextualizing and structure	Preview on the WT and the surroundings before the installation. Opportunity to simulate WT in local environment on different locations, utilizing height and weather data.
1 Descriptive	Describe the physical systems state using real-time sensor data and historical data for states back in time. Notify user of events.	Live and historical sensor data for velocity, pressure, temperature, vibration, power generation, and humidity/density. Real-time presentation of the turbine with notifications of critical conditions or changes.
2 Diagnostic	Supports condition monitoring and troubleshooting with diagnostic information	Indicators give vital information about the turbine without physical inspection. Remote inspection of turbine fault.
3 Predictive	Predicts future states or performance, supports prognostic capabilities.	Prediction of wind conditions, power generation, and time left to component failure.
4 Prescriptive	Provide prescriptions/recommendations based on what if/risk analysis and uncertainty quantification.	Recommendations for example when to start the turbine or carry out maintenance.
5 Autonomous	Closed control loop where the DT replaces the user where decisions and executive control actions on the system are made by the autonomously DT.	Control loop with Power optimization, (Maximum Power Point Tracker)

Since each of the levels builds on the preceding, these levels can be a good tool for defining and building the DT more complex step by step. Using this way of thought, the first step of designing the Digital Twin will be to create a CAD model of the physical asset (here the wind turbine) that is unconnected, and from there on add functionalities and connections until the digital twin is in the wanted capability level.

In the next section, suggested development steps are explained with inspiration from the defined components from the frameworks and capability level view. Additional information on relevant topics is provided as well in the.

2.3 Assessment

To begin with, an assessment of wind resources and location can be made using digital tools. This step will support an early-stage system design phase, where the information on the wind energy system already accumulates even before the wind turbine is physically made and installed. The use of visualizing tools for the knowledge gained through assessments will, in this stage, contribute to the making of a standalone DT already.

2.3.1 Wind resource assessment

A wind resource assessment aims to map the average wind speed and direction in the chosen location. The evaluation can extract information from historical weather data to calculate a wind energy system's average annual power output. Meteorological institutes have weather databases for most places, and these can be used to map the typical wind speed and direction for a specific location. If the source data isn't sufficient for the particular site, adjustment tools for data regarding height adjustment and other local conditions are available[2].

2.3.2 Site Assessment

An assessment of the place, current buildings, and the wind turbine can replace expensive and time-consuming experimental testing. Incorporating the mapped wind resources and CAD models into an analysis software can give insight into system behaviour, expected power output, and the visual aspect of the wind energy system.

2.4 Create

As identified through the comparison of the dimension-based frameworks, the data system is one of the three core components. It follows that a Wind Turbine with a Digital Twin must be equipped with sensors, communicating devices, and actuators. In creating the physical system, the implementation of sensors is essential for data gathering towards a real-time monitoring system, and a descriptive twin.

From the assessment phase, the wind resources are determined, and proper wind energy harvesting technology can be chosen. The following section describes some wind turbine types.

2.4.1 Wind Turbines

Classifying Wind Turbines can be ground in rotor size or rated power, the axis of rotation, if the turbine has fixed or variable speed, and the forces that drive the rotor, for example. The technology of a WT is divided in three ways in the book of Kishore et al.[21], and is based on the following;

1. Orientation of the axis of rotation - vertical or horizontal.
2. Component of aerodynamic forces that powers the wind turbine - lift or drag. (Difference between the lift force and drag force will be described in Section 2.6.)
3. Energy-generating capacity - micro, small, medium, or large.

Horizontal Axis Wind Turbine (HAWT)

The HAWTs axis rotates parallel to the incoming airstream. The most commonly used HAWTs are the upwind (blades in front of the rotor) two- and three-bladed turbines[2]. These turbines need to face the wind to function and are, therefore, dependent on a yaw system to maintain correct orientation. For big WTs, an active yaw system such as an electric motor is needed, but for smaller turbines, it is often sufficient with a fan tail that positions itself along the wind direction[2][21]. The efficiency of these lift-type turbines lay around 45-50% [21].

Vertical Axis Wind Turbine (VAWT)

The VAWT rotates perpendicular to the wind flow and does not need any yaw mechanism as it can exploit wind from all directions[2]. The vertical axis also allows the generator and gearbox to be placed on the ground. Some common small-scale VAWTs are (i) Cup anemometer, (ii) Savonius rotor, (iii) Darrieus rotor, and (iv) Musgrove rotor. The two first are driven by drag force, and the latter two are driven by lift force. [2].

- Efficiency below 40%, Savonious below 25% [21]
- Lift and drag type

Table 2.2: The table show different classifications of Wind Turbines

Reference	Twidell & Weir		Kishore et al.		Salih et al.	
Scale/Class	Diameter (m)	Power (kW)	Diameter (m)	Power (kW)	Diameter (m)	Power (kW)
Micro	n/a	n/a	0 - 0,10	n/a	3 <	0,05 – 2
Small	6.4-20	10-100	0,10 - 1	n/a	3 - 12	2 – 40
Medium	20-64	100-1000	1 – 5	n/a	12 -45	40 – 999
Large	64-160	1000-6000	>5	n/a	>46	>1000

Small Wind Turbines (SWTs)

In [21] classifies the size of WTs as micro, small, medium, and large. However, different authors operate with a different number of classes, and as Table 2.4.1 shows, their definition varies a lot. [2].

However, the European standard SWT NEK IEC 61400-2:2013 considers a HAWT for small-scale WT if the power rating is 50kW or below[9]. This seems like a more reasonable classification, as it is close to two of the findings in the table.

The three most essential components of SSWT are (i) Rotor, (ii) Transmission mechanism, and (iii) Generator[21]. In addition, the tail and radial bearing (for the passive yaw system), the tower, and breaking mechanism.

2.4.2 Sensors

Smart monitoring combines information obtained from sensors over time with up-to-date analysis- and representation technology. E.g. the WINDMIL-project executed at ETH Zürich [22]. Here, affordable sensor technology combined with both "Data-driven Simulation (Inverse Engineering)" and "Physics-Based Simulation (Forward Engineering)" are used to aid WT operators in monitoring, inspecting, and life-cycle assessment of the WTs. Some examples of sensors are presented in Section 4.

Condition-based maintenance

[23] explains the different maintenance modes for both preventive and corrective maintenance. In preventative maintenance, there is either scheduled maintenance based on pre-determined dates or condition-based maintenance. Typical components that are exposed to failures are the blades, nacelle/drivetrain, hydraulic/pneumatic system, tower, sensors, and electrical system [24]. In addition, sensors that should monitor the state of important components could also fail, leading to a cease of critical information stream. Therefore, condition-based maintenance and a robust monitoring system are crucial elements to prevent unexpected failure and expensive accidents. For a small-scale WT, the aerodynamics of the blades are of big importance regarding the efficiency of the total system[21]. Optimizing the performance and detecting failures before it considerably affects the system will be beneficial.

2.5 Communicate

Communication is the transfer and exchange of data between the different components of the Digital Twin system. This includes communication between the different DT domains and between components within each domain. E.g., between the physical and the digital domain and between the data storage and the simulation models. This requires a good communication interface that allows the data to flow quickly between the entities and to be easily accessible and interpretable.

2.5.1 Standards and Security

Cyber-security is crucial, and the security aspect should be incorporated in the design of a DT[12]. This will be a natural part of the development of communication interfaces in the DT system, as this will cover

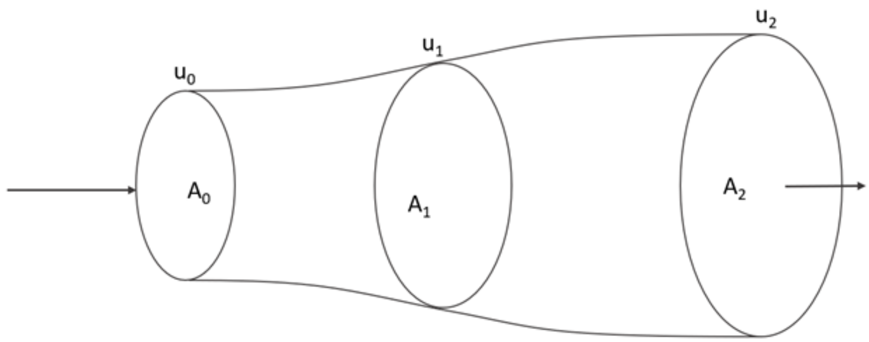

Figure 2.4: The figure shows the air column used in linear momentum theory.

all data transferring. In addition, standards can increase interoperability, and the entity-based framework is also based on the standard for digital twins in manufacturing systems, ISO23247[**ferko˙supporting˙nodate**].

2.6 Aggregate

A wind energy system is complex, as it combines the fields of aerodynamic forces, mechanical loads, electro-mechanical power systems, and advanced material technology. Making a Digital Model of a wind energy system. Consequently, a digital model that aggregates all these aspects with high accuracy is a model with great complexity. To tackle the aspect of high-fidelity modeling, especially regarding the dynamic behavior of a wind turbine and the interacting flow, different complexities for modeling should be evaluated in advance. For establishing a basic understanding of how to model a wind turbine's behavior, the fundamental principles of wind energy harvesting technology will be presented below.

2.6.1 Linear Momentum

The Lanchester-Betz-Zhukowsky theory involves calculating the kinetic energy in the wind based on a simple model of an air column[2], as sketched in Figure2.4. The mass of the air column can be expressed as

$$m = \rho A_0 u_0 \tag{2.1}$$

, where ρ is the air density, A_0 is the crossectional area of the column, and u_0 is the and the kinetic energy as

$$KE = 1/2 m u_0^2 \tag{2.2}$$

And the power potential in the moving wind at speed u_0^2 can be expressed as

$$P_0 = \frac{1}{2}(\rho A_1 u_0) u_0^2 = \frac{1}{2}\rho A_1 u_0^3 \tag{2.3}$$

The power coefficient C_P for a wind turbine can be explained as the relationship between the available power in the airstream for a crossectional swiped area A and the actual extracted wind power P_T for the wind turbine as follows

$$P_T = C_P P_0 \tag{2.4}$$

Betz deduces a maximum limit for the power coefficient

$$C_P^{max} = 16/27 = 0.59 \quad (2.5)$$

In practice, the maximum power coefficient of wind turbines is approximately 0.40[2], and according to [21] it is even lower for SWTs.

2.6.2 Blade element

The aerodynamic principle of wind turbines is to utilize *the lift* and *drag* forces. To explain how these forces work, the blades can be divided into cross-sections called *airfoils*.

Drag and lift force

The drag force D and the lift force L can be expressed with equation 2.6.2 and equation 2.6.2, respectively. The forces working on an airfoil are illustrated in 2.5.

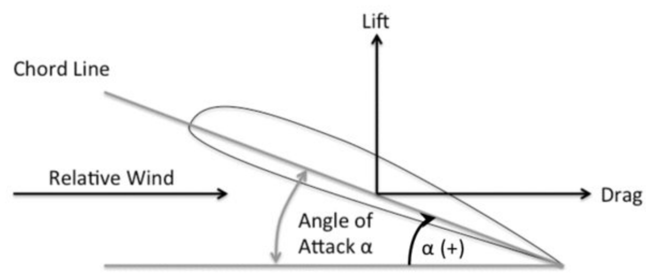

Figure 2.5: The figure shows the cross-section of a wind turbine blade rotating perpendicular to the incoming wind, with an angle of attack between the chord line and the relative wind. The directions of the lift and drag forces are shown. Figure obtained from [25] with licence CC BY 4.0.

$$D = C_D \frac{\rho}{2} A u_\infty^2 \quad (2.6)$$

$$L = C_L \frac{\rho}{2} A u_\infty^2 \quad (2.7)$$

, where A is the area swiped by the rotor with a radius equal to the turbine blades length, ρ is the density of the air, u_∞ is the upstream wind speed from afar, and C_D and C_L are proportional constants called drag and lift coefficients, respectively[21].

For a turbine that utilizes the lift force, the lift-to-drag ratio should be the highest possible, regulated by the angle of attack, α. Consequently, the angle of attack is an essential variable for optimal wind energy conversion. The turbines driven by lift forces can have higher rotational speed than the wind, unlike the drag turbines limited to moving as fast as the wind flow[2]. For these turbines, the tips speed ratio λ is defined as

$$\lambda = \frac{\omega R}{u_0} \quad (2.8)$$

, where ω is the rotational speed, R is the blade radii and u_0

The resulting wind power P from the turning torque produced by the blades can be integrated over the length of the blade and as follows:

$$P = \int_{r_h}^{r_t} \frac{1}{2} B \rho u_{rel}^2 (C_L sin\phi - C_D cos\phi) c \omega r dr \quad (2.9)$$

, where r_h and r_t denoted the hub and tip radii, respectively. B represents the number of blades, u_{rel} is the relative wind speed, ϕ is the inflow angle, c is the length between the blade's edges, and dr is the thickness of the blade element along its length. As u_{rel} and ϕ vary for the blade sections, the loads and power production is usually solved using computers and numerical simulations by discretization and iterative calculations[2].

2.6.3 Modeling

To simulate and analyze a wind turbine's aerodynamic performance, experimental, analytical, and numerical methods can be used[10]. Experimental methods consist of wind tunnel testing, and the Blade Element Momentum method is an analytical method using physics-based modeling. The numerical approach can be done with Computational Fluid Dynamics (CFD)[26].

CFD numerically solves the flow variables pressure, p, and velocity, v, with high resolution and accuracy. CFD software can often visualize the results in a fast and intuitive way. The fidelity of the simulation of a wind turbine system may depend on the solving algorithm's ability to describe the laminar to turbulent flow[21]. Different turbulence models will be discussed in Section 4.

2.7 Analyze

The analysis step may consist of different types of analyses, dependent on the capability and goal of the digital twin. As a feature in the FBDTF, it should validate the monitoring data, and this is the minimum requirement. Furthermore, the analysis of the simulation data should make it interpretable for human and AI decision-makers. It is key to obtaining insight into the connected physical and digital systems by extracting useful information for the supervising entity from incomprehensible big data sets. The interpretability for users can be obtained through Data-Driven Modeling such as Artificial Intelligence and Machine Learning. These methods can also be the fulfilling components for prediction and optimization. Examples from the literature will be presented in Section 4.

2.8 Insight

The next feature should be a way of utilizing the analyzed information for the user. Visualizing methods that provide a real-time picture of the physical system will finally realize the digital system as a virtual representation and enable monitoring. Still, according to the definition defined in Section 2.1, the control of the physical system must be enabled.

2.9 Act

In the final step, tools that allow the supervising entities to perform remote actions on the physical asset should be implemented. In the digital part of the system, this is provided through a user interface, for example through web applications. Now, the Digital Twin should be able to accurately describe the wind turbine's aerodynamic behavior through simulations and monitoring data in a real-time virtual model, and provide other services like predictions, model optimization, and autonomous control.

2.9.1 User Interface

Provide easy access and interpretation for the human mind. The user interface is presented in all three framework types (Services in the dimension-based framework). It is the enabling tool for human

Integrating AI in the DT enables autonomous decision-making, and the DT can perform its tasks without a supervising user executing every task.

Chapter 3

Method

3.1 Literature search

The main collection of data for the thesis is done through an integrative review. The literature review was chosen because of an aim to map the possibilities and challenges of developing a digital twin for a wind turbine. With the integrative review, exploring different ideas and experiences from others can be combined into a new framework. The choice of search database has mainly been Scopus due to the amount of literature available and the "advanced search" feature, which contributes to the efficient literature search. Google Scholar and Research Gate have been used for additional purposes.

Topics that were reviewed are:

- Digital twin frameworks
- Small wind turbine
- Digital Twins for wind energy
- Open-source software for Digital Twins
- Simulation methods for the aerodynamics of a wind turbine
- Sensor technology for Digital Twins

These topics were believed to cover the scope of the research questions and to give valuable information on the development stages for a Digital Twin for the aerodynamic performance of a small roof-mounted wind turbine.

During the work, the author experienced that the scope and number of topics ranged too broad, and this caused challenges in extracting useful information. The author's coverage of the available literature is not guaranteed, as many of the findings were new information and demanded the acquiring of much new knowledge. Often the search entries gave either too broad or too narrow results. Narrowing down publishing years until recent years might have made it hard to find Digital Twin-related theory related to Small-scale wind turbines, as the offshore wind sector with large turbines is the present focus. Additionally, the literature on especially Digital Twins, but also wind turbines has inconsistency in definitions and fundamental theories. This made it even more time demanding to collect and extract valuable and accurate information.

A better study of the field of digital twins in advance of the thesis could have led to a better understanding and more concise research questions on the topic. Also, a systematic or semi-systematic literature review may contribute to a better discussion of different methods by limiting the review's range. However, this study is important because it provides a useful insight into the different necessary elements when developing a DT. To the authors best knowledge, such an assessment is novel, and will therefore contribute important first insights into creating a common framework for developing digital twins by open-source software.

3.2 Workshop

The other source of information during this work was through a Digital Twin workshop at Jotne Connect (then Jotne EPM Technology). The company develops Digital Twin software that complies with the STEP and ISO23247 Standards for Digital Twins. The intention of attending the workshop was to gain a better understanding of the practical use of a Digital Twin, as literature only contributes to theoretical knowledge. The workshop presented use cases in various industries, a go-through of the semantics for data management using EXPRESS language, and some hands-on exercises on simulation data management.

At that point, the author experienced a lack of foundational knowledge to fully make use of the workshop and to obtain the expected insights. Nevertheless, an understanding of the need to establish open standards in the Digital Twin field has grown more extensive during the work with the thesis and even so, with the workshop in mind. The attendance also gave an introduction to data modeling and semantics for storing and simulating data, which is a very important aspect not only in the context of Digital Twin but in the entire domain of digitalization.

Chapter 4

Results and Discussion

This chapter presents the different steps of developing a digital twin based on the high-order framework shown in Figure 4.1; *Assessment*, *Create*, *Communicate*, *Aggregate*, *Analyse*, *Insight*, and *Act*. Literature on potential methodology and software is presented in every step, attempting to unite the available information in one DT development strategy. Section **??** summarizes the framework, extracting the key findings from the presented literature.

The frameworks presented in 2 all describe the DT system based on its functions. The FBDTF in [12] sticks out with its Data Link feature, which reduces the number of communication interfaces. The framework in Figure 4.1 visualizes the main development steps for a DT for roof-mounted SWT and is based on the frameworks and DT capability levels presented in Section 2.

Combining the star structure in the FBDTF and the dimensions and entities in the other frameworks might possibly lead to more efficient communication as the number of interfaces is reduced. Hence, the information flow is not restricted to the predefined interfaces between each element, as all information is accessible through one hub.

4.1 Assessment

Each of the capability levels presented above builds on the preceding. These levels can be a good tool for defining and building the DT more complex step by step. Using this way of thought, the first step of designing the Digital Twin will be to create a CAD model of the physical asset (here the wind turbine) that is unconnected, and from there on, add functionalities and connections until the digital twin has acceptable capability level for its specific use. For its purpose, the model in the assessment step does not have to be of high accuracy, as it will not give any real-time predictions and is not critical to the operation of the WT in the future. Therefore, information about the modeling strategy will not be presented here, but in Section 4.4

The development of an unconnected virtual model (Standalone DT) may not be able to predict the aerodynamic performance of the WT precisely. Still, it can provide other values to the user. Developing a Standalone DT for a house using Unity Game Engine was demonstrated in the work of Elfarri[27]. The author shows the many opportunities the model provides for both realtors and homebuyers. Extending these findings to a WT, vendors, and private persons considering installing wind power systems can have potential benefits in utilizing a game engine in a planning phase, as it will aid in informed decision-making with realistic demonstrations. For example, game engines provide insights into the WT's impact on its surroundings before the physical creating stage. This advantage is emphasized in the work of Sørensen et al.[28], where Unreal Engine 5 (UE5) is used to simulate two HAWT types in a realistic environment. The simulation gave information on the visual impact and the power production of the WTs. The authors also mention how the engine contributes to the optimization of the WT's position in the current area, the prediction of power production, and the quick implementation and development using *plugins*. The

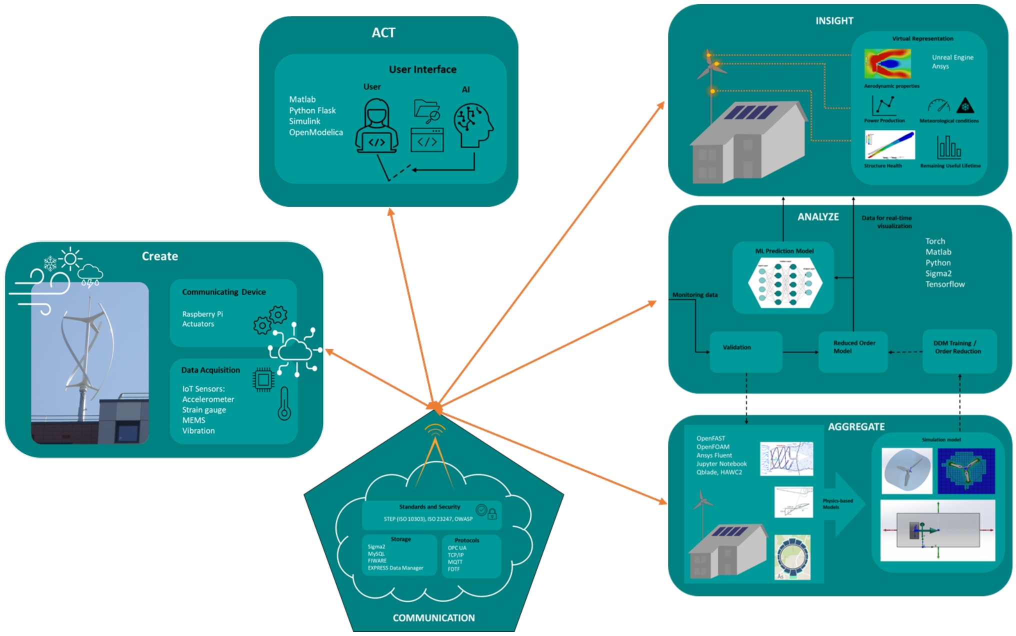

Figure 4.1: The figure depicts the suggested DT Development Framework with the steps from Create to Act. The Assessment step has been incorporated in the segment for the Aggregate step, as the model can be further developed there.

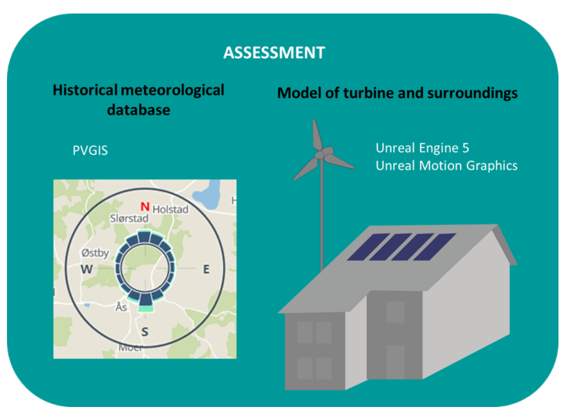

Figure 4.2: In the assessment step, an unconnected virtual model with optional fidelity is made based on resource and site assessments. Pictures from windy.app and needpix.com.

plugins in UE5 are collections of code, data, and visual assets[28] and act as enabling tools for an easy DT development process, as they can be reused and support the quick implementation of functions. Still, they also limit the DT abilities to the existing plugin futures. To develop case-specific components, C++ coding, and other methods are available in UE5[28].

In a feasibility study on building-mounted wind turbines, Arteaga-López et al.[9] use CFD software to perform an analysis of the mapped wind resources for an urban environment. The proposed method for the development of building-mounted wind turbines projects is summarized in eight steps; The two first steps form the evaluation of the site and feasibility and wind resources, and the following four consist of modeling and simulating the different parts in the physical system using CFD analysis, and the two final steps are economic evaluation and system installation. SOLIDWORKS Flow Simulation software was used for the CFD simulations in the modeling steps.

The analysis of the flow surrounding the buildings revealed a turbulence effect and suggested the height for the SWT to be above this "bubble"(21m a.g.l.) to harness the full wind speed. Simulating a 13.5m hub height WT on top of a chosen building showed a damaging turbulence area close behind the rotor. On the other hand, simulating a 21m hub height WT, this turbulence was not present[9].

These findings are interesting in the DT context, demonstrating how the site and wind turbine CFD assessment can help reveal local conditions through visualizing tools. It gave insight into the flow characteristics in the specific site and how the hub height of the WT is of great significance in accordance with obtaining optimal wind harvest and the longest possible WT lifetime.

4.1.1 Wind resource data

Together with turbine characteristics, wind data is necessary for power generation prediction. In [28], historical wind data is collected from the European Commission's *Photovoltaic Geographical Information System* (PVGIS) (online tool primarily used to predict PV-system performance based on location). In the use case presented in [9], a governmental meteorological center aided the wind resource assessment to find the average wind speeds by regional zones. Adjustment techniques for the wind data height was used, as the data was logged for a different height than the current WT's hub height. Some methods for estimating wind speeds at higher altitudes are available. In [9], two methods are mentioned, *the Hellmann exponential law* and *the logarithmic wind profile law.*

The Hellmann exponential law can be expressed as follows:

$$\frac{v}{v_0} = (\frac{H}{H_0})^{\alpha} \tag{4.1}$$

The logarithmic law can be expressed as follows:

$$\frac{v}{v_0} = \frac{ln(\frac{H}{z_0})}{ln(\frac{H_0}{z_0})} \tag{4.2}$$

, where v and v_0 is the speed at heights H and H_0, respectively. α is defined as the friction coefficient and z_0 is the roughness coefficient length. More information about α and z_0 can be found in [29].

Other computational tools for wind assessment are mentioned in [2]; the Numerical Objective Analysis Boundary Layer (NOABL) and Wind Atlas Analysis and Application Program (WAsP). NOABL models the air mass moving across complex terrain but does not consider local disturbances like vegetation and buildings, and therefore is bad-suited for the purpose of SWT in urbanized areas. The WAsP modeling involves removing local effects from reliable wind data from one place to generalize it and then adding the local topography and obstructions of another relevant site for wind power production[2].

The Create step can be initialized when the assessment has provided enough information to evaluate and decide what turbine type and equipment to go for.

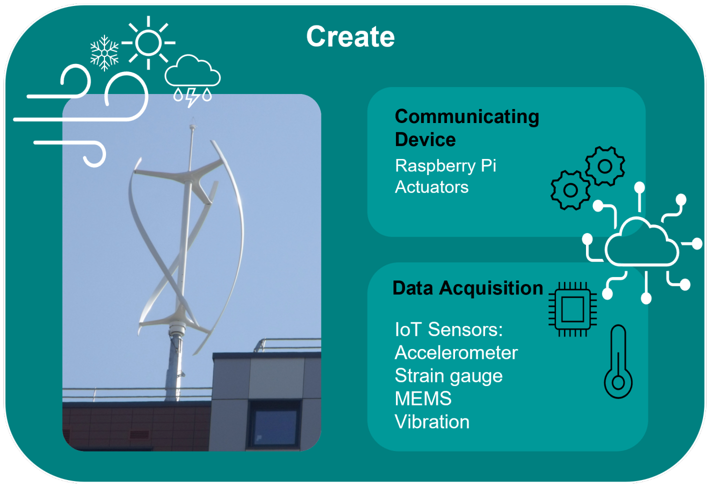

Figure 4.3: The Create step consists of the installation of physical object. (A picture of a Vertical Axis Wind Turbine on the new Aston University Student Accommodation by ell brown is licensed under CC BY 2.0.)

4.2 Create

The second stage is creating the physical system, where the installation of the wind turbine, its sensors, and actuators is performed. Based on the assessment results, the wind turbine can be placed in the best-suited location, and the mapped wind resources may help to find the correct turbine regarding power rating and rated speed. Describing the installation of wind turbines and actuators is beyond the scope of this thesis, but some relevant sensor technology for smart monitoring will be mentioned beneath, as they are essential for the data acquisition.

4.2.1 Sensor Technology

Suitable sensor technology was defined as the first Key Enabling technology (KET) for realizing the development of a Human-Cyber-Physical System by Chen et al. [30]. The importance of non-destructive structural health monitoring has been underlined in several studies [31][32], [30][23]. A survey on smart inspection of WTs performed by Dimitrova et al.[31] divides the inspection technology into three; surface inspection (e.g. Li-DAR), subsurface inspection (e.g. 3D Laser-Scanning), and defect detection with sensors. Commonly used damage detection methods are compared, showing that drone inspection is preferred on an overall basis, but sensor inspection is the only standard method that does not require human resources[31].

Sensors must continuously collect data to support the real-time simulations in a digital twin and provide accurate information by monitoring values for flow dynamics and meteorological conditions. As structural damage also impacts aerodynamic performance, sensors for structural monitoring should be applied. This will add valuable information for PLM since structural monitoring enables condition-based maintenance.

For aerodynamic monitoring, data acquisition for wind speed and direction and meteorological data like air temperature, humidity, and pressure data should be performed. These values can be measured from the nacelle with an anemometer, wind vane, and temperature, humidity, and pressure sensors. Furthermore, temperature and humidity sensors can contribute to detecting ice formation, and if possible, sensors can be placed on the blades. To support the control of the mechanical system (e.g., active yaw system), insight into upwind conditions such as wind speed can be obtained using Li-DAR technology[31].

For structural monitoring, accelerometers, strain gauges, vibration sensors, and Micro-Electro-Mechanical sensors (MEMS) can be mounted on both the support structure and blades[14]. The accelerometers are mentioned as one of the most promising approaches for remote sensing technologies in [30]. The design and implementation of structural monitoring of an offshore wind farm are presented in Hines et al. [33], including the use of various accelerometers, both wired and wireless. For the wireless accelerometer, a gateway transceiver is mounted. This method is applied on large-scale WTs and might not be directly transferable to SWTs due to the potentially negative impact on aerodynamic performance if the geometry of the sensors is too big and lumpy. Hence, finding suitable sensor technology for the specific wind turbine and location (blade or structure) is crucial. A work on SWTs done by Cheung [24] uses an accelerometer and microcontroller unit (MCU) for blade condition monitoring towards condition-based maintenance. The three-axial accelerometer measures the frequency of the blade to monitor it's natural vibration and oscillation and detect abnormal values to determine when maintenance should be performed.

Sensor installment
The correct location for the sensors can be found executing FEM analysis as investigated in [34]. The experiments utilized several sensor types to develop detection methods for typical blade defects; force sensors, accelerometers, and strain gauges. Although the experimental setup with wired sensors is not directly applicable to a real scenario, the work provides fault detection and optimal sensor placement techniques.

4.3 Communication

When the physical system is created, sensor data can be communicated through wireless communication technology for temporary or long-term storage. Communication between the system's components can use the protocols presented in Table 4.1.

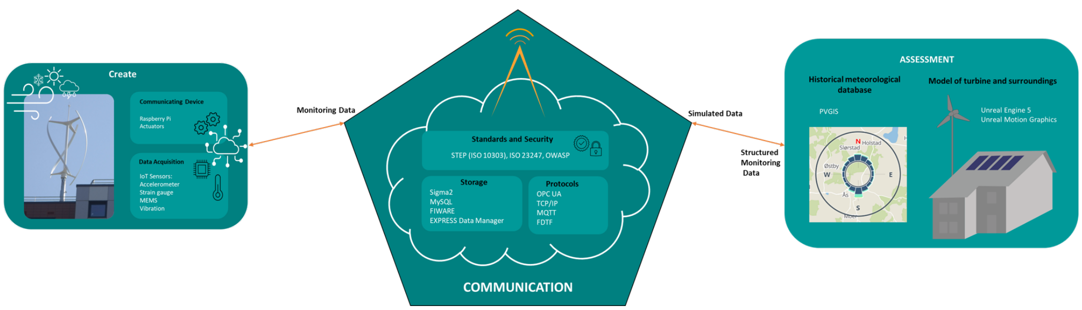

Figure 4.4: The tools applied in the communication step allows data transferring between the physical and digital physical system.

4.3.1 Communication protocols

Communication protocols are tools for allowing the supervising entity (human or AI algorithm) to communicate with the IoT sensors. Both communication and data representation protocols are necessary. The authors in [8] list protocols considered relevant, which are divided into protocols for communication and data representation in [35]. This is shown in Table 4.1. In [36] the interoperability between DTDL and OPC UA is demonstrated. The MQTT protocol is used in an experiment with the micro computer Raspberry Pi as the communicating hardware[37].

4.3.2 Standards and security

It is stated that different stakeholders not utilizing the same standards and data modeling formats hinder interoperability, collaboration, and efficient DT development[10]. For this reason, relevant standards and security measures should be incorporated into the Digital Twins Communication step.

Table 4.1: List of main data representation and communication protocols used for transfer and exchange of data. The table is adapted from [35]

Type	Abbreviation	Term	Description
40.12Data Rep.	DTDL	Digital Twin Definition Language	Open-source, describing DTs with sets of metamodel classes [38]
	-	FIWARE	Framework of open-source platforms enabling data management and sharing between systems[39].
	OPC UA	Open Platform Communication Unified Architecture	Widely adopted standard for information flow and semantically data modeling[36][40].
	FDTF	Feature-Based DT Framework	Each DT feature communicates through one Data Link, reducing interfaces[12].
40.12Comm	CoAP	Constrained Application Protocol	Compatible with web-based application and HTTP data transmission.[35]
	MQTT	Message Queuing Telemetry Transport	Utilise a publish-subscribe protocol for rapid communication[37].
	Modbus TCP/IP	Transmission Control Protocol / Internet Protocol	Designed for communication in the industrial field between PLCs, using port 502 data transfer.[35]
	URLLC	Ultra Reliable Low Latency Communication	Reliable and low-latency data exchange based on 5G wireless protocol[35].

On the topic of securing data, there exist many resources for securing methods. An example is the open-source community-lead foundation OWASP, which has created a general model for measuring and improving the security of a software system [41]. More relevant Digital Twin and IT security standards are referenced in the DNV report[20, p. 9] mentioned earlier.

Though there are standards available and in development [10], the best method for obtaining good data management and cyber-security has to be mapped for each specific case based on an evaluation of the security threats[12].

The STEP Standard (ISO 10303: STandard for the Exchange of Product model data) is an International Standard for the representation of product information and for the exchange of product data[10]. On ISO's website, it is stated that the "*objective is to provide a neutral mechanism capable of describing products throughout their life cycle*"[42]. As part of the STEP standard development, the EXPRESS Language was and has been approved as an international standard (ISO 10303-11) by the International Organization of Standardization in 1994 [43]. The purpose of the language was to provide a full range of capabilities needed to represent data, independent of the implemented technology, to enable data interoperability and integration. The two core capabilities of EXPRESS are human readability through a graphical layout with EXPRESS-G, and computer processability, through a lexical form. The core data functionalities are data exchange, data sharing, and data archiving, covering some main challenges of information technology[43].

Standard languages like the EXPRESS language provides a basis for reliable archiving, with an information model for data semantics so that future users can analyze and understand the content of the archived data sets. The "core language data model" for establishing the semantics of the data holds five key elements[43];

1. **Entity data types:** represent information on objects that are of interest to the user, e.g. product, organization, or person.

2. **Attributes:** Represent properties of the information e.g. name, length, and startDate.
3. **Constraints:** represent uniqueness, cardinality, or the values of attributes.
4. **Subtyping:** The subtype inherits the attributes and constraints of one or more other types, and possesses additional attributes and constraints.
5. **Schemas:** Gather elements into logical collections. The schema contains entities that in its turn contain attributes, constraints, and/or subtypes.

EXPRESS Data Manager (EPM) is a data managing tool using the EXPRESS language. The software is commercial, but it enables data modeling that can comply with standards like STEP, PLC, and ISO 23247. [43]

In [19] the adaption of the entity-based ISO 23247 DT framework in the manufacturing industry (shown in Figure 2.3 in Section 2.2) is mapped towards each functional entity. An overview of the technologies that are used in each entity is given. Of the 14 cases analyzed, only two are realizing the security functional element with either C-based algorithms or a protocol named Oauth2[19].

4.3.3 Storage

In a DT for a tidal turbine, the data hosting platform MySQL Server is used as an open-source database for the necessary management of data [16]. Another open-source platform is FIWARE, listed in Table 4.1, which is an industrial data platform for smart systems integration. Sigma2 data services for Norwegian researchers and scientists offer resources for data storage and sharing, in addition to other services such as, high-performance Computation (HPC), data management plan DMP, and Sensitive Data Sevices[44].

4.4 Aggregate

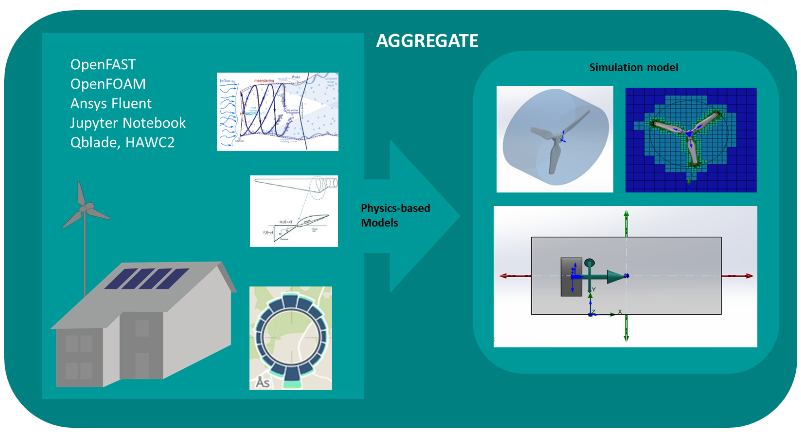

Figure 4.5: The Aggregate step connects the physical models of reality into a Pictures licenced CC by 4.0 licence

The main goal for the DT is aggregating models for the aerodynamic performance of the specific wind turbine within a particular location with high accuracy. Empirical physical equations, numerical simulations, and experimental flow characteristics are aggregated in one multiscale model. For the flow characteristic, the wind and site assessment executed in Section 4.1 comes in handy, as it can tell what model for flow dynamics is needed.

Because the aim of making the DT is to create the real-time virtual model of the physical system (here, the aerodynamic performance of the WT) as accurately as possible, the following subsections attempt to

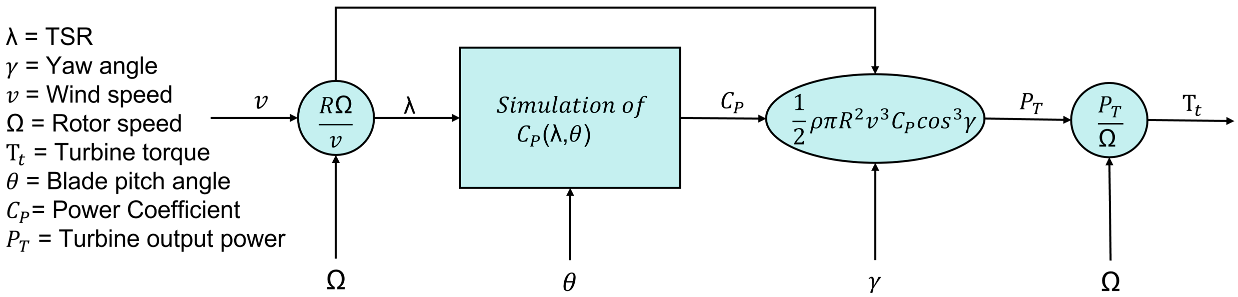

Figure 4.6: Adapted simple digital model for WT aerodynamics presented by [14].

summarize the potential modeling techniques for the rotor's and surrounding flow's aerodynamics found in the literature.

4.4.1 Aerodynamic performance

The review of De Kooning et al. [14] discusses enabling modeling tools for a DT framework that simulates both the aerodynamic and mechanical performance of a wind turbine, suggesting different levels of complexity and potential software. To begin with, the aerodynamic performance can be simulated in a simple digital model as shown in Figure 4.6, where the equations for Tip Speed Ratio, λ, power output, P_t, and turbine torque, T_t, is calculated with effective wind speed, v, rotational speed, Ω, and the yaw angle, γ, as inputs. These input data come from the monitoring system on the turbine, both for the rotor and the electrical mechanical components. In addition, the power coefficient C_P is simulated as a function of λ and the blade pitch angle θ, where the simulation provides insight into the maximal power point for the specific wind turbine. The model is a low-fidelity model where potential methods for increasing complexity and higher accuracy for different aspects are suggested.

The aspects for potential improvement are (i) temperature-varying air density, (ii) aging effects and ice formation impact on power coefficient, (iii) non-constant turbine torque, (iv) aero-elastic effects and turbine behavior, and (v) wake effect in a wind farm[14].

The incorporation of BEM and CFD simulation is proposed to take aspects (iv) and (v) regarding aerodynamic and wake effects into account. The wake effect is especially important for turbine location energy systems with several turbines, as the wake from one turbine might reduce the operating conditions of another.

BEM

This Blade Element Momentum method calculates the aerodynamic force on each blade element and the accumulated loads on each blade. Due to the rapid calculations that come with the low complexity and inexpensive computations, blade element momentum method is a widely used tool[45][21]. Several codes that utilize BEM for simulating WT aerodynamics are mentioned in literature [45] [21] [14] These are OpenFAST, QBlade, FLEX5, Bladed, Phatas, Gast and HAWC2.

Despite being a popular tool, it has shortcomings as the physics cannot describe every operating scenario of a WT[46][**kishore˙wind˙2018**]. In [46], it is stated that a 2D airfoil BEM simulation will under-predict the aerodynamic loads and power production because it does not consider the effects of rotational augmentation. Correction models for specific conditions such as tip vortexes, dynamic stalls, and disturbed flow have been developed, and they can make a sufficiently good model for a particular case. Still, they do not make the BEM model applicable to WT simulations on a general basis, as the correction models only count in a range of values. In addition, the momentum method becomes invalid for high solidity WTs and high tip speed ratios[46]. Since these are the characteristics of lift-based VAWTs like Savonius and Darrieus rotors, other methods are required to obtain sufficiently good simulations.

CFD

With Computational Fluid Dynamics, the flow surrounding the wind turbine can be numerically calculated. (and describe the interface between solid and fluid). According to the review of Thé and Yu[46] there are three main CFD techniques; The Reynolds Averaged Navier-Stokes (RANS) method, the Large Eddy Simulation (LES), and the Hybrid RANS-LES method (HRLM), where the most commonly used method is that of RANS. CFD solves nonlinear partial differential equations for a finite number of volume elements in space and time to find the velocity and pressure fields around the turbine rotor[21]l. This method is often called the Finite Volume Method (FVM).

The process of Computational Fluid Dynamics can be divided into three steps[21]:

1. **Pre-processing:** Defining the geometry, computational domain, and mesh.
2. **Solving:** In the second step, the flow physics model and material properties are chosen before solving the mathematical model.
3. **Post-processing:** The third step is to analyze the results concerning the accuracy and then visualize them to emphasize the key findings.

The two first steps will be described in the following subsections. The post-processing in the third step will be covered in the sections for analysis 4.5 and insight 4.6, as they describe analyzing and virtualization, respectively.

Pre-processing

Geometry While creating the geometry, some considerations should be made regarding the complexity of the model to minimize the computational power needed in the solving process and, at the same time make it realistic enough. For dimensionality, the 2D simulations with RANS methods will over-predict the aerodynamic performance due to the inability to simulate blade tip vortices[46]. Hence, 3D modeling is often preferred for better results. As the 3D simulations demand a lot higher level of computational power, 2.5D methods have been investigated as well. In [47] the performance of a SWT was tested using 2D, 2.5D, and 3D CFD simulation cases. The study showed that all three models were acceptable for the blade tip, but considering the flow dynamics near the hub, 2.5D or 3D simulations are needed.

An evaluation of what components to include should be performed as well. For example, it will be sufficient only to model the rotor without the support structure if the wake effect is not of importance[21]. Another method to cut down on the computational domain is to take advantage of periodicity. For a three-bladed HAWT, the rotor has a periodicity of 120, which in theory, makes it possible to analyze only a third of the entire rotor. Doing this will also lower the effort in meshing[21]. On the other hand, the periodicity does not necessarily count for the fluid dynamics, as both the wake and inflow might be asymmetric and ununiform for transient flow[46]. Modeling a turbine blade can be done utilizing the 2D cross-section airfoil data for known WT types and assemble them in a 3D CAD program like for instance, the NREL5MW is modeled for a numerical investigation by Siddiqui et al. [26]. The Master thesis of Erlandsen and Sæther[48] explores parametric modeling strategy and digital twin development using 3D CAD models with a focus on the ability to understand and adopt the model for other engineers. The study resulted in a recommendation of *Resilient Modeling Strategy* (RMS) for building a component, and the authors refer to the best practice on the website for RMS. The authors recommend Inventor software for parametric modeling.

In addition to the turbine model, a surrounding computational domain with boundary conditions needs to be defined. In [46], "Best practice guidelines" gives detailed recommendations on the size and shape of the domain related to the rotor diameter, D, to minimize computational costs, but maintain a sufficient distance to avoid interference with the domain boundaries. The inlet and outlet boundaries are placed with a minimum of 2D upwind and 10D downwind relative to the rotor.

An essential part of the pre-processing is to generate a mesh for the calculations in the processing step. Different mesh types and methods can be applied and these are described in detail in [21]. In addition, automatic meshing is incorporated in some CFD software, like SOLIDWORKS used in the study described under Assessment[9]. Even so, it is important to evaluate the mesh to ensure the correct refinement and grid-independent results before the actual processing. This is called grid convergence analysis[46]. The same study also provides the initial

For the rotational movement three techniques for rotational motion Single Reference Frame (SRF), Multiple Reference Frame (MRF), and Sliding Mesh Interface(SMI) are presented in [26] and [46]. SRF and MRF are used for steady-state solutions, and the governing equations need additional terms for the rotational motion[47] . SRF can only be used if every component rotates around the same axis with. In MRF, the mesh is divided into subdomains separating rotating and fixed parts, e.g. rotor and support structure. The SMI is a rotating cylindrical mesh that solves for transient states[46], but has been proven to have a high computational cost.

Processing

Before solving, the flow characteristics and turbulence model must be chosen. The characteristics and wind direction and speed should be available from the preceding development modules. The local wind speed and directions have been found through the assessment of wind resources in the first development stage. In addition, sensor data from the physical turbine is available after the creation phase.

Flow physics From the lowest to the highest fidelity, [14] mentions potential wake models called the Jensen model (only based on wind and rotor speed), the NREL tools FLORIS and FAST.FARM, and Large Eddy Simulation (LES). Another middle-fidelity method is the Reynold Averaged Navier-Stokes(RANS) method.

Reynold-Averaged Navier-Stokes (RANS) In CFD equations that describe the physics of Newtonian fluid are solved, and in[46], three potential turbulence correction models for RANS are discussed:

1. The Spalart Allmaras (SA) model: This one-equation model is reported to have good performance for wind speeds up to 7m/s for HAWT. On the other hand, the model does not provide good results for the pressure distribution on the blade surface and the low-speed shaft torque at stall conditions for wind speeds from 10m/s[46]. Furthermore, it is not appropriate for near-wake velocity profiles for VAWTs.

2. The $\kappa - \epsilon$ models: The two-equation model is widely used due to its simplicity and robustness. It comes in many variants and is considered good for turbulent flow in HAWT applications but deviates from the experimental data at low TSR for VAWTs. The two transport equations solve the kinetic energy κ and the turbulence dissipation rate ϵ [49].

3. The SST (Shear Stress Transport) $\kappa-\omega$ model: This model is also based on two transport equations and. The principle is to combine the $\kappa-\epsilon$ model for the free stream and $\kappa-\omega$ model for the near-wall flow. [47] used the model for parametric analysis.

In [9], the flow pattern was simulated in SOLIDWORKS with turbulence model k-ϵ. Most common CFD software has the option of using RANS turbulence models. For example, the same aforementioned model was used in OpenFOAM to simulate the aerodynamic performance of a SWT in regard to different angles of attacks in(AOAs) [49].

Though widely used to predict key performance parameters, the RANS methods have some downsides. According to [46] the most significant weakness is the incapability of capturing unsteady flow characteristics, where massive flow separation and vortex shedding are pointed out. These lacks can be covered by Large Eddy Simulations (LES).

Large Eddy Simulation (LES) LES is the most accurate turbulence model for solving Navier-Stokes equations but it comes with a high computational cost that is still too demanding for most computers[46][21]. To make a compromise between the efficient but sometimes inaccurate RANS method and the accurate but time- and processor-demanding LES method, the Hybrid RANS-LES Method (HRLM) has been developed[46].

Hybrid RANS-LES Method The HRLM combines the simulation of attached boundary layers with the RANS method and the separated flows with the LES method, where the computational complexity meets in the middle of RANS and LES. The method is mentioned as an improvement to the $\kappa - \epsilon$-method in [49], as it will better calculate the wake and near-wall flow. The work of Thé et al.[46] provides a best practice guidelines for employing HRLM.

Solving the mathematical model The next steps toward solving the computations are setting up temporal and spatial discretization and solving algorithms for appropriate iteration time steps and coupling pressure and velocity. The best practice of Thé Yu[46] and the book of Kishore et al.[21] provide more information on this. The number of processors should be evaluated concerning the number of nodes in the mesh as well, where a rule of thumb may be 50.000-100.000 nodes per core[21].

Real-time simulations

For enabling real-time prediction DTs with numerical high-fidelity simulations, computational resources that make the simulations and analyses as fast as possible[10] is a prerequisite. Since an ordinary computer lacks the computational power to perform HPC tasks, other ways need to be found. Computational fluid analysis can be both time-consuming and demand a lot of processor force. Therefore it may not be fitted to be used in a real-time model, since this will delay the results, stopping us from achieving a realistic and precise digital twin.

For high-performance computations, the use of supercomputers to obtain good results in a reasonable amount of time is used by researchers. The Norwegian resource Sigma2 has several supercomputers for HPC and additional data management services for research and sharing of data for high-complexity cases.

Lowering computational power

A method for reducing computational power and enabling real-time simulations is to make a multi-layer model where the different levels have different time steps. The method is suggested by De Kooning et al. and it is based upon a compromise between real-time simulations with low-fidelity models, and high-fidelity models with accurate results but not real-time. The layers are ordered in hierarchical order from models with low fidelity and small time-step that give real-time simulations, to high fidelity and big time-step simulations to update the low-level models for a more realistic virtual model. The high-level computations can utilize cloud-based resources for high-performance computations.

4.4.2 Summarising remarks

The techniques that enable real-time simulations with adequate quality for DTs make out a comprehensive topic. Trying to summarize the essence, some aspects can be pointed out: The historically much-used BEM is computationally affordable but is not able to simulate dynamic effects in a realistic manner. The more computationally expensive CFD method using RANS models for simulating aerodynamic effects from the interaction between blade surface and fluids can give good results. A higher price in simulation time must be paid to get the most realistic results, addressing an outage of sufficient computational power. The same counts for the LES model, which describe the wake accurately, but with higher computational complexity than RANS modeling. To get better results faster, HPC resources are crucial.

Alternatively, different methods using data-driven modeling can ease the computational power by replacing the most complex physics modeling with machine learning to find physical relations with lower computational costs. This topic has been placed in the next section, as it can be considered a data analysis.

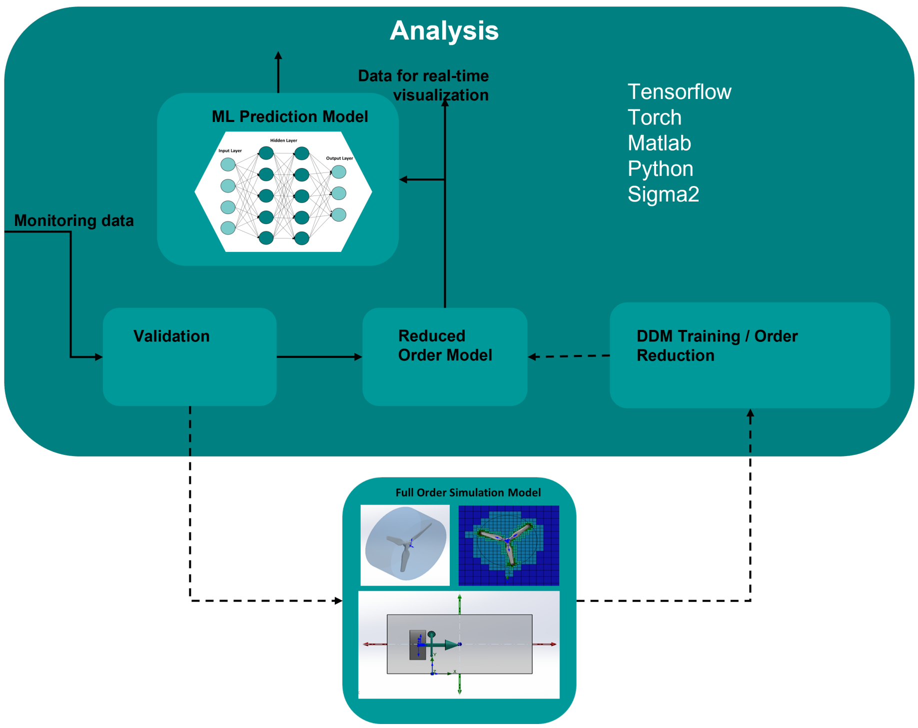

Figure 4.7: The figure show schematics of a potential Analysis method using Hybrid Analysis and Modeling. The dotted lines represent data transfer in the time-demanding numerical simulation using Full Order Models, and the solid lines show the real-time simulation loop.

4.5 Analyze

Analysis should be performed to validate the monitor and simulation data quality and to process the data for visualization[12]. It follows that the analysis step is equally important as the choice of modeling technique in the last step, as it is of great significance for the quality of the virtual real-time model. In addition to the post-processing of simulation results, using Data-Driven Modeling (DDM) like machine learning (ML) can simulate physical relations either entirely or partly, whereas the second alternative refers to Hybrid Analysis and Modeling (HAM).

4.5.1 Simulation Post-processing

Validation methods will, in most cases, consist of comparing the simulation results with experimental data and evaluating the error[46]. Often, Root Mean Square Error is used to compare the errors. The simulations made by using the chosen level of fidelity in the aggregate phase then need to be analyzed to extract useful information for the DT user[**empty citation**]. To the author's best knowledge, none of the reviewed articles included concrete methods.

4.5.2 Alternative analysis for modeling

Data-driven modeling (DDM)

Data-driven modeling is a way of utilizing artificial intelligence and machine learning to learn the "full physics"[10], and it has been gaining popularity with the arrival of easily applicable open-source machine learning libraries like Tensorflow, Torch, and OpenAI[50]. Autiosalo et al.[12] distinguish AI from ML by defining AI as the decision-making tool and ML as the algorithms that process data for AI or a user. Connecting it to the DT development stages, the AI constitutes the Act-step with the human user, enabling automated DT control. At the same time, The ML algorithms are a part of the Analysis step, processing and finding interesting correlations or information from vast datasets.

ML algorithms can enhance the DT with predictive capabilities. In the work of [51], a wind power prediction model was developed by combining Principal Component Analysis (PCA) and deep learning algorithms. PCA was used on large data sets of wind data, and the results are used as input data for the open-source TensorFlow deep learning framework. Keras API is used with TensorFlow to enable the method to be deployed in many other platforms[51]. The results were validated and compared to other state-of-the-art techniques, showing that the method had lower errors and better visualization for wind farm data.

Another area of application for ML is the concept of "Virtual Sensors"[52], where machine learning finds relationships from simple sensor time-series data to replace too complex or even non-existing monitoring methods for wake effects, blade root bending moment, and blade tip-tower clearance. The study showed that a lagged-FNN approach with low computational cost worked well for non-predictive purposes, and that LSTM (Long Short-Term Memory) could provide signal forecasting seconds ahead of real-time. This method might solve the problem with the inaccuracy of sensor monitoring pointed out in the survey of inspection methods by Dimitrova et al. [31], or substitute costly numerical wake simulations.

The DT model should be able to be reused in another case to support the collaborative effect that is demanded, as stated earlier. As the machine learning algorithms are specialized for their training data, they might be too restricted for adaptation in other cases. This is where the hybridization of PBM and DDM becomes relevant.

Hybrid Analysis and Modeling (HAM)

Physics-based modeling and Data-driven modeling has been two separate categories. A comprehensive overview of DT technology within the wind energy industry was recently published by Stadtman et al.[10]. The article addresses the experiences and challenges with DT technology from both industrial and researchers' points of view while comparing state-of-the-art DT technologies and modeling techniques. Stadtman et al. conclude that neither Physics-based Modeling (PBM) or Data-driven Modeling (DDM) is ideal for DT context, pointing out challenges like too time-consuming and computationally demanding simulations for PBM and the lack of insight in DDMs due to black-box methods. Hybrid Analysis and Modeling (HAM) is proposed as the new ideal for modeling to tackle these challenges. The concept of HAM is to use PBM as a basis to utilize its robust, generalizable, and understandable method and DDMs capable of fast computations for real-time predictions and model updating[10].

Surrogate models In the range of HAM methods, surrogate models are mentioned in many works [8][53][54] [14]. The complexity of high-fidelity physics-based simulations makes it a time-demanding task. A surrogate model can be the solution to get fast simulations for real-time prediction.

Jorgensen et al. propose a method for a [55] surrogate model that can be described as a fast simulator using real-time observation data from the physical system and the output data from a slower numerical analysis in the virtual system to provide diagnosis and predictions for the Digital twin to act upon. Here the surrogate model is doing the so-called "fast twinning" while the numerical model forms a "slow twinning" system together with model validation and updating.

While defining a surrogate model as *"a computationally efficient tool to estimate the main features of the output of a more complex simulator, given some inputs"*[55], Jorgensen et al. also divides the surrogate models into three categories;

1. Data-driven emulators: Uses statistical or mathematical methods
2. Reduced-order models (ROM): Reducing the dimensions in governing equations with orthonormal vector basis
3. Multi-fidelity based surrogates: Reduction of the numerical resolution, relaxation of tolerance, and removal of physics effects.

The author further develop a surrogate model based on the first category with Gaussian Process (GP) surrogate model for an OWT support structure. It is presented as a fit model for many applications in the offshore engineering space, including FEM simulations[55].

The reduced-order model (ROM) method is based on reducing the complexity of the model of a system and describing it with fewer parameters, still obtaining an acceptable level of accuracy in the analysis results. In [56] a ROM for turbulent flow surrounding a WT is obtained using Proper Orthogonal Decomposition (POD). The work utilizes the open-source tools OpenFOAM and Python, applying FVM on the Full Order Model (FOM) and reducing this model with POD.

To analyze an airfoil, Tsiolakis et al.[56] have used a "hybrid projection-based proper orthogonal decomposition (POD) strategy applied to turbulent flow problems." OpenFOAM has been used for the purpose of calculating the reduced operators in the "offline phase." For the construction of the reduced basis and solution of the "online phase," Python was used[56]. The results showed relative errors under 5% and 2% for pressure and velocity, respectively, which shows that the method gives good approximations with reduced order of fidelity and can be of interest for DT implementation.

The term "Big data cybernetics" is defined as a physics-based model that is updated using data-driven modeling in a control loop, making a hybrid method utilizing both PBM, DDM, Big data and control theory in[50][53]. It is discussed as a enabling tool for prediction and what-if scenario. Prediction methods must be implemented in the analysis to give a basis for automated control.

Two different frameworks of HAM, Physics-Guided Machine Learning (PGML) and Interface Learning (IL), are tested in [57]. The PGML uses physics-based features at intermediate layers of the neural network to reduce the uncertainty of predictions and is demonstrated for predicting aerodynamic forces. Compared to a standard ML method, the PGML uncertainty is considerably lower for AOAs between -10 and +12 degrees. This supports the choice of HAM instead of only data-driven modeling as a way of enabling predictive features in the DT.

The proposed IL framework[57] is a coupling system for a Full Order Model (FOM) and a Reduced Order Model (ROM), where the coupling method is presented in three different ways. The technique for making the ROM matrix with orthonormal bases of a reduced subspace is Proper Orthogonal Decomposition (POD)[57]. The three coupling systems are tested in a demonstration using LSTM as ML architecture, and results show that the method called "Uplifted Prolongation Interface" gives the most accurate values compared to the FOM due to its ability in error corrections.

[58] Physics-based Output Prediction Model (p-bOPM) using ROM for the floater on a Floating Offshore WT (FOWT), Ansys Twin Builder and Modelica language modeling, FAST code and Simulink for real-time power prediction with environmental data from a Real-Time Digital Simulator working as a sensor, communicating through TCP/IP. Modelica, FAST and Simulink are all open source, but Ansys Twin Builder is proprietary software. However, Ansys did release open-source PyAnsys software on GitHub in October 2021[59], enabling developers to use Python packages for a small sample of Ansys software. Packages for several software were promised, and in August 2022 the package for Ansys Fluent, *PyFluent*, was released[60].

4.5.3 Summarising remarks

Validation of results for evaluating model performance and updating the model if needed must be done to obtain DT quality. Using DDM instead of PBM will lower the demand of computational power multiple times, allowing real-time simulation and prediction DTs. On the other hand, ML demands big datasets for training algorithms on specific cases and is not as generalizable as PBM. Combining PBM and DDM can utilize the strength of both in a HAM-approach. Different methods for both areas need to be chosen for the specific task, e.g. simulation in real-time, prediction models and high-fidelity FOMs supporting surrogate models with reduced complexity.

4.6 Insight

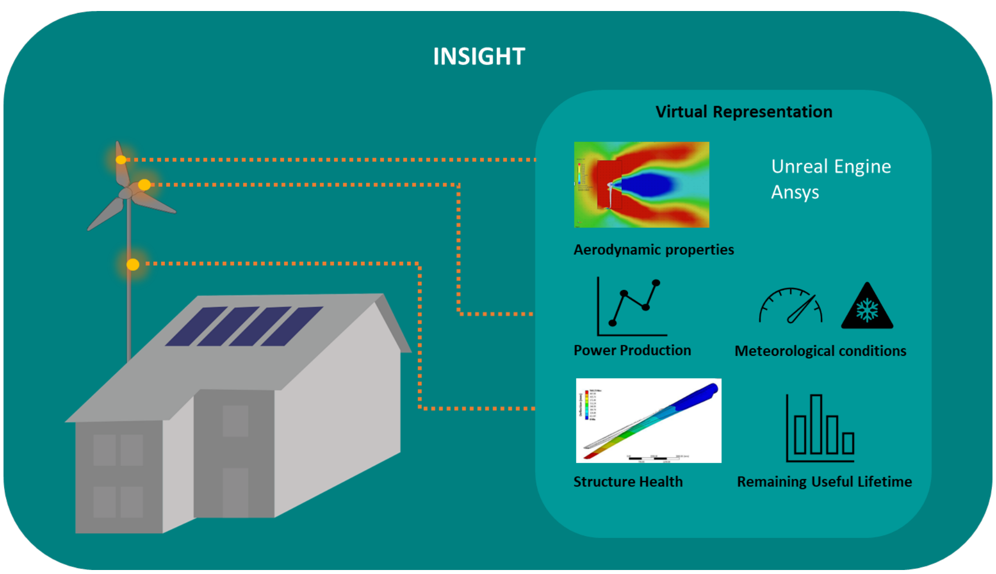

Figure 4.8: The figure depicts real-time virtualization of the physical system, where the left part could be a live 3D model using a game engine, and the technical data can be represented in a monitoring screen in a web application.

4.6.1 Visualization

Fully updating the virtual model, the simulation and analysis results can be added to the virtual model from the chosen platform in the assessment step. Not many articles regarding the exact methods of virtualizing the analysis results are found, maybe because it is already incorporated in many analysis software. Some extracted information regarding flow visualization and game engine are presented below.

Flow and vortex

Given the ability to compute real-time analysis with supercomputers and hybrid methods or DDM, the simulation of flow and vortices can be visualized in the live DT virtual model.

Starting with a basic method of visualizing the performance is to show the calculated values for e.g., power production in figures like diagrams that are continuously updated. In that way both the real-time data and historical data are clearly interpretable for the human user. According to Sørensen et al. [28] Unreal Engine is capable of showing power production.

Based on examples from articles, a variety of methods to present flow characteristics from the numerical analysis can be used; 3D ISO surfaces or contour plots picturing vortices, contour plots for velocity, contour plots for pressure, and streamlines and flow patterns for the direction, where streamlines can show separation occurrence along the blade[46][47]. The ability to apply these methods for a live DT must be considered, as they are used for simulating a scenario for a given simulation run and not for a continuous live model.

Condition data

Other variables that describe the system conditions can be visualized as well to ease the understanding of a human user. Typically this is wind direction and weather conditions, and temperature indicators. Again the game engines are brought up for virtualization of surrounding environment. The thesis written by Elfarri[27] shows how a game engine (Unity Game Engine) can be an excellent tool for obtaining realistic real-time visualization, even visualizing conditions a human can't see, like temperature and air quality using a particle approach. Implementing this virtual reality to an operating wind turbine, visualizing humidity and temperature might give a quick indicator of critical conditions like the potential for ice formation.

4.7 Act

From the visualization provided by the insight platform, a human user gains knowledge for decision-making and planning future actions based on a predictive or prescriptive DT. On this, the user should be able to make actions for the system state through an interface. To obtain an autonomous DT, another supervising entity like AI is needed to close the control loop with prediction model-based decision-making. As shown in Figure 4.9, the human user should be able to disconnect the artificial intelligence as a supervisor, while acting on the predictions analysis.

Cybernetics

The figure shows a control loop, which can be modeled and executed quite easily in software like OpenModelica[61] or Simulink[62]. Python Flask and Simulink are connected through Matlab to create a PI pitch control in [62]. Simulink has readily understandable block features for an easy implementation. Python Flask is also used for coding a web-based interface.

User Interfaces

In [63], a browser-based program for digital twins is developed using Python as a programming language and the Python flask package. The web interface is suggested to be designed with HTML5, CSS, and

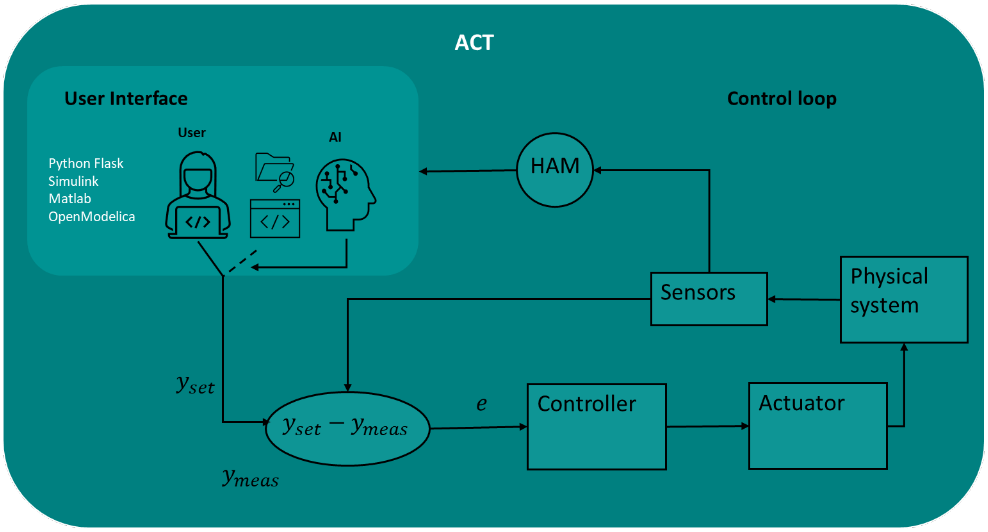

Figure 4.9: Conceptual sketch of how a control loop can be implemented to obtain autonomous DT. The user can connect and disconnect the AI, which will act on prediction analysis and control algorithms when connected.

JavaScript. The CAD model of the physical system is generated in AutoDesk Fusion 360. The data model is built up in a three layer-structure where the foundation layer is made out of the IoT components like the physical sensors, devices, and hardware like sensors and actuators. The second interface layer connects the user to data via the network and is made out of user device(s) and the user access via a web server. The third layer is the cloud computing layer, which consists of data storage and High-Performance Computing.

The authors in [63] have a set of requirements that are listed under:

1. Provide access for users
2. Monitor the state of the physical twin
3. Modifications to various aspects of the physical twin
4. Perform simulations to aid in decision-making
5. Centralize results to aid in decision-making
6. Evolve digital twin to ensure accurate comparability

They have concluded that the DTOP-Cristallo answers to requirements 1, 4, 5, and 6., excluding requirements 2 and 3 which are considered IoT-layer requirements. As the code is openly shared, a further adaptation toward a specific use case, e.g. a wind turbine, could be developed.

Chapter 5

Conclusion

In the Introduction, the primary objective and research questions were defined:

Primary objective

The main goal of this thesis is to propose a framework for developing a high-fidelity digital twin for a roof-mounted wind turbine utilizing open-source software.

Research Questions

1. **What are the components of a high-fidelity digital twin for a wind turbine?**
2. **What open-source software can be used to realize the components of the Digital Twin?**

Following has been found in the thesis:

The analysis of available literature on seven suggested steps of development has mapped the key methods and software possible to realize a high-fidelity digital twin for the aerodynamics of a wind turbine. This has been presented in 4. Key findings are;

- For accurate and realistic simulation in real-time, the lack of computational power to do so has been underlined in the literature. Alternative methods were mapped, and the Hybrid Analysis Modeling (HAM) approach has experimentally been a good solution.
- Realizing tools for high-fidelity simulations are supercomputers with High-Performance Computation and machine learning with the ability to analyze big data.
- Data modeling and cyber security are of high importance as Digital Twin Systems are built around the exchange and flow of data. Methods of gaining the correct level of security must be evaluated for each case based on existing standards and solutions.

For each of the seven steps, the components (answering research question one) and methods (answering research question two) have been summarized as follows:

Assessment

1. Main components: Unconnected digital model.
2. Method: Map potential for power production and visualize system. For example using a game engine for a virtually good location and CFD software for aerodynamic characteristics for the best site concerning power output and turbine performance

Create

1. Main components: (Physical system), Data acquisition, e.g., IoT sensors
2. Method: Choice of correct sensor type and placement, with respect to desired parameters for simulations and monitoring.

Communicate

1. Main components: Security system, data modeling and exchange, storage.
2. Method: Security as part of DT design, protocols for interoperable data semantics and transfer, easy-to-access storage, and storage for big data.

Aggregate

1. Main components: Models for simulating the wind turbine's behavior (e.g. aerodynamic performance, power generation, structural loads), the surrounding environment (e.g. flow dynamics and meteorological conditions), and computational software.
2. Method: For a WT aerodynamics focus, the DT should have high-fidelity CFD modeling of the rotor-fluid interaction and lower fidelity can be applied for electrical and mechanical systems. The need of High-Performance Computation (HPC) resources should be addressed, e.g., Sigma2.

Analyze

1. Main components: Data validation, model validation, and data-driven modeling.
2. Method: Validation analyses for the quality of both monitoring data and simulation models. Data-driven modeling can support the simulations defined in Aggregate to obtain both accurate, generalizable, and real-time results. In addition, machine learning involving neural networks can make use of big data for predictions and what-if scenarios.

Insight

1. Main components: Real-time virtual model, and visualization of technical data, predictions, and scenarios
2. Method: A virtual model updated with real-time information can show the user's preferred parameters in, for example, game engines.

Act

1. Main components: User interface
2. Method: A web-based user interface allowing the user to access data and models from the components in Communication, Aggregate, Analyze, and Insight steps, and to perform actions on the physical system through the DT's communication and actuators.

5.1 Concluding Remarks

The present framework has not been tested and is only a conceptual framework at the time. In addition, the amount of theoretical DT frameworks available is big, but not supported by a similar amount of shared experiences. Therefore, the next step to strengthen the present framework's applicability is the development and experimental-based evaluation of an open-source software-based DT. During a potential development using the suggested OSS, the interoperability of the software and necessary security measures should be evaluated in the early stages.

With the current technology, HPC resources are expensive and DT development is complicated. This causes too big of an inconvenience for vendors and private energy system owners. Therefore, the adaptation of the present framework is more relevant to research and development projects.

References

[1] IPCC. "Summary for Policymakers". In: *Climate Change 2022: Impacts, Adaptation, and Vulnerability. Contribution of Working Group II to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. Ed. by H. O. Pörtner et al. In Press. Cambridge, UK: Cambridge University Press, 2022, In Press.

[2] John Twidell and Tony Weir. "Renewable Energy Resources". In: *Renewable Energy Resources*. Third Edition. new York: Routledge, 2015, pp. 234–323.

[3] NMBU. *Strategi 2019 til 2023*. Strategi 2019-2023 Norges miljø- og biovitenskapelige universitet. Apr. 20, 2018. URL: https://www.nmbu.no/om/strategi/2019-2023 (visited on 05/01/2023).

[4] Arne Sandbakken. *Solkraft i sterk vekst*. EnergiPluss. Apr. 8, 2022. URL: https://energipluss.com/solkraft-i-sterk-vekst/ (visited on 05/15/2023).

[5] Statkraft. *Vindkraft — Statkraft*. URL: https://www.statkraft.com/var-virksomhet/vindkraft/ (visited on 05/15/2023).

[6] Kommunal- og moderniseringsdepartementet Olje- og energidepartementet. *Veileder for kommunal behandling av mindre vindkraftanlegg[H-2343 B]*. 2015. URL: https://www.regjeringen.no/contentassets/cc4c0065a1c24ff3882d13778e7343ea/veileder_kommunal_behandling_mindre_vindkraftanlegg.pdf.

[7] Michael Grieves. *Origins of the Digital Twin Concept*. Aug. 31, 2016. DOI: 10.13140/RG.2.2.26367.61609.

[8] Zhihan Lv and Elena Fersman, eds. *Digital Twins: Basics and Applications*. Cham: Springer International Publishing, 2022. ISBN: 978-3-031-11400-7 978-3-031-11401-4. DOI: 10.1007/978-3-031-11401-4. URL: https://link.springer.com/10.1007/978-3-031-11401-4 (visited on 01/12/2023).

[9] Ernesto Arteaga-López, Cesar Ángeles-Camacho, and Francisco Bañuelos-Ruedas. "Advanced methodology for feasibility studies on building-mounted wind turbines installation in urban environment: Applying CFD analysis". In: *Energy* 167 (Jan. 15, 2019), pp. 181–188. ISSN: 0360-5442. DOI: 10.1016/j.energy.2018.10.191. URL: https://www.sciencedirect.com/science/article/pii/S0360544218321881 (visited on 03/16/2023).

[10] Florian Stadtman et al. *Digital Twins in Wind Energy: Emerging Technologies and Industry-Informed Future Directions*. Apr. 16, 2023. DOI: https://doi.org/10.48550/arXiv.2304.11405. arXiv: 2304.11405[cs]. URL: http://arxiv.org/abs/2304.11405 (visited on 04/26/2023).

[11] Colin Raffel. "Building Machine Learning Models Like Open Source Software". In: *Communications of the ACM* 66.2 (Feb. 2023), pp. 38–40. ISSN: 0001-0782, 1557-7317. DOI: 10.1145/3545111. URL: https://dl.acm.org/doi/10.1145/3545111 (visited on 04/18/2023).

[12] Juuso Autiosalo et al. "A Feature-Based Framework for Structuring Industrial Digital Twins". In: *IEEE Access* 8 (2020). Conference Name: IEEE Access, pp. 1193–1208. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2019.2950507.

[13] Interopre. *About Joinup — Joinup*. May 11, 2023. URL: https://joinup.ec.europa.eu/collection/joinup/about (visited on 05/12/2023).

[14] Jeroen D. M. De Kooning et al. "Digital Twins for Wind Energy Conversion Systems: A Literature Review of Potential Modelling Techniques Focused on Model Fidelity and Computational Load". In: *Processes* 9.12 (Dec. 2021). Number: 12 Publisher: Multidisciplinary Digital Publishing Institute, p. 2224. ISSN: 2227-9717. DOI: 10.3390/pr9122224. URL: https://www.mdpi.com/2227-9717/9/12/2224 (visited on 01/09/2023).

[15] Ziqi Huang et al. "A Survey on AI-Driven Digital Twins in Industry 4.0: Smart Manufacturing and Advanced Robotics". In: *Sensors* 21.19 (Jan. 2021). Number: 19 Publisher: Multidisciplinary Digital Publishing Institute, p. 6340. ISSN: 1424-8220. DOI: 10.3390/s21196340. URL: https://www.mdpi.com/1424-8220/21/19/6340 (visited on 01/17/2023).

[16] Yu Cao et al. "Digital twin real time monitoring method of turbine blade performance based on numerical simulation". In: *Ocean Engineering* 263 (Nov. 1, 2022), p. 112347. ISSN: 0029-8018. DOI: 10.1016/j.oceaneng.2022.112347. URL: https://www.sciencedirect.com/science/article/pii/S0029801822016419 (visited on 01/11/2023).

[17] Qinglin Qi et al. "Enabling technologies and tools for digital twin". In: *Journal of Manufacturing Systems*. Digital Twin towards Smart Manufacturing and Industry 4.0 58 (Jan. 1, 2021), pp. 3–21. ISSN: 0278-6125. DOI: 10.1016/j.jmsy.2019.10.001. URL: https://www.sciencedirect.com/science/article/pii/S027861251930086X (visited on 01/11/2023).

[18] Chunlong Wu et al. "Conceptual digital twin modeling based on an integrated five-dimensional framework and TRIZ function model". In: *Journal of Manufacturing Systems*. Digital Twin towards Smart Manufacturing and Industry 4.0 58 (Jan. 1, 2021), pp. 79–93. ISSN: 0278-6125. DOI: 10.1016/j.jmsy.2020.07.006. URL: https://www.sciencedirect.com/science/article/pii/S0278612520301163 (visited on 01/11/2023).

[19] Enxhi Ferko, Alessio Bucaioni, and Moris Behnam. "Supporting Technical Adaptation and Implementation of Digital Twins in Manufacturing". In: *ITNG 2023 20th International Conference on Information Technology-New Generations*. Ed. by Shahram Latifi. Advances in Intelligent Systems and Computing. Cham: Springer International Publishing, 2023, pp. 181–189. ISBN: 978-3-031-28332-1. DOI: 10.1007/978-3-031-28332-1_21.

[20] DNV AS. *Preview: DNV-RP-A204 Qualification and assurance of digital twins*. DNV. 2020. URL: https://www.dnv.com/Default (visited on 03/07/2023).

[21] Ravi Kishore, Colin Stewart, and Shashank Priya. *Wind Energy Harvesting : Micro-to-Small Scale Turbines*. De Gruyter Textbook. Berlin: De Gruyter, 2018. ISBN: 978-1-61451-565-4. URL: https://search.ebscohost.com/login.aspx?direct=true&db=e000xww&AN=1800858&site=ehost-live (visited on 02/23/2023).

[22] *ERC StG WINDMIL*. URL: https://chatzi.ibk.ethz.ch/erc-stg-windmil.html (visited on 03/27/2023).

[23] Kyungil Kong et al. "Progress and Trends in Damage Detection Methods, Maintenance, and Data-driven Monitoring of Wind Turbine Blades – A Review". In: *Renewable Energy Focus* (Aug. 23, 2022). ISSN: 1755-0084. DOI: 10.1016/j.ref.2022.08.005. URL: https://www.sciencedirect.com/science/article/pii/S1755008422000606 (visited on 01/11/2023).

[24] Chun Kau Cheung. "Small Wind Turbine Blade Sensor for Condition Based Maintenance". PhD thesis. San Luis Obispo, California: California Polytechnic State University, Sept. 1, 2011. DOI: 10.15368/theses.2011.183. URL: http://digitalcommons.calpoly.edu/theses/628 (visited on 02/21/2023).

[25] Yehezkiel Tumewu, Crescenzo Petrone, and Mettupalayam Sivaselvan. "Numerical Simulation of the Influence of Platform Pitch Motion on Power Generation Steadiness in Floating Offshore Wind Turbines". In: *International Journal of Environmental Science & Sustainable Development*. 1 (July 1, 2017), p. 9. DOI: 10.21625/essd.v1i2.39.

[26] M. Salman Siddiqui et al. "Numerical investigation of modeling frameworks and geometric approximations on NREL 5MW wind turbine". In: *Renewable Energy* 132 (Mar. 1, 2019), pp. 1058–1075. ISSN: 0960-1481. DOI: 10.1016/j.renene.2018.07.062. URL: https://www.sciencedirect.com/science/article/pii/S0960148118308590 (visited on 01/17/2023).

[27] Elias Mohammed Elfarri. "Digital Twin of a Building Powered by Artificial Intelligence and Demonstrated in Virtual Reality". Accepted: 2022-08-26T17:19:52Z. Master thesis. NTNU, 2022. URL: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/3013924 (visited on 03/07/2023).

[28] Jonas Vedsted Sørensen, Zheng Ma, and Bo Nørregaard Jørgensen. "Potentials of game engines for wind power digital twin development: an investigation of the Unreal Engine". In: *Energy Informatics* 5.4 (Dec. 21, 2022), p. 39. ISSN: 2520-8942. DOI: 10.1186/s42162-022-00227-2. URL: https://doi.org/10.1186/s42162-022-00227-2 (visited on 05/04/2023).

[29] F. Bañuelos-Ruedas, C. Angeles-Camacho, and S. Rios-Marcuello. "Analysis and validation of the methodology used in the extrapolation of wind speed data at different heights". In: *Renewable and Sustainable Energy Reviews* 14.8 (Oct. 1, 2010), pp. 2383–2391. ISSN: 1364-0321. DOI: 10.1016/j.rser.2010.05.001. URL: https://www.sciencedirect.com/science/article/pii/S1364032110001309 (visited on 04/25/2023).

[30] Xiao Chen et al. "A Human-Cyber-Physical System toward Intelligent Wind Turbine Operation and Maintenance". In: *Sustainability* 13.2 (Jan. 2021). Number: 2 Publisher: Multidisciplinary Digital Publishing Institute, p. 561. ISSN: 2071-1050. DOI: 10.3390/su13020561. URL: https://www.mdpi.com/2071-1050/13/2/561 (visited on 05/04/2023).

[31] Mariya Dimitrova et al. "A Survey on Non-Destructive Smart Inspection of Wind Turbine Blades Based on Industry 4.0 Strategy". In: *Applied Mechanics* 3.4 (Dec. 2022). Number: 4 Publisher: Multidisciplinary Digital Publishing Institute, pp. 1299–1326. ISSN: 2673-3161. DOI: 10.3390/applmech3040075. URL: https://www.mdpi.com/2673-3161/3/4/75 (visited on 05/04/2023).

[32] Jiajun Xia and Guang Zou. "Operation and maintenance optimization of offshore wind farms based on digital twin: A review". In: *Ocean Engineering* 268 (Jan. 15, 2023), p. 113322. ISSN: 0029-8018. DOI: 10.1016/j.oceaneng.2022.113322. URL: https://www.sciencedirect.com/science/article/pii/S0029801822026051 (visited on 01/11/2023).

[33] Eric M. Hines et al. "Structural instrumentation and monitoring of the Block Island Offshore Wind Farm". In: *Renewable Energy* 202 (Jan. 1, 2023), pp. 1032–1045. ISSN: 0960-1481. DOI: 10.1016/j.renene.2022.11.115. URL: https://www.sciencedirect.com/science/article/pii/S0960148122017670 (visited on 05/04/2023).

[34] Yaowen Ou et al. "Vibration-based monitoring of a small-scale wind turbine blade under varying climate conditions. Part I: An experimental benchmark". In: *Structural Control and Health Monitoring* 28.6 (2021). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/stc.2660, e2660. ISSN: 1545-2263. DOI: 10.1002/stc.2660. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/stc.2660 (visited on 03/28/2023).

[35] Cheng Qian et al. "Digital Twin—Cyber Replica of Physical Things: Architecture, Applications and Future Research Directions". In: *Future Internet* 14.2 (Feb. 2022). Number: 2 Publisher: Multidisciplinary Digital Publishing Institute, p. 64. ISSN: 1999-5903. DOI: 10.3390/fi14020064. URL: https://www.mdpi.com/1999-5903/14/2/64 (visited on 04/27/2023).

[36] Salvatore Cavalieri and Salvatore Gambadoro. "Proposal of Mapping Digital Twins Definition Language to Open Platform Communications Unified Architecture". In: *Sensors* 23.4 (Jan. 2023). Number: 4 Publisher: Multidisciplinary Digital Publishing Institute, p. 2349. ISSN: 1424-8220. DOI: 10.3390/s23042349. URL: https://www.mdpi.com/1424-8220/23/4/2349 (visited on 04/27/2023).

[37] E.P. Hinchy, N.P. O'Dowd, and C.T. McCarthy. "Using open-source microcontrollers to enable digital twin communication for smart manufacturing". In: *Procedia Manufacturing* 38 (2019), pp. 1213–1219. ISSN: 23519789. DOI: 10.1016/j.promfg.2020.01.212. URL: https://linkinghub.elsevier.com/retrieve/pii/S2351978920302134 (visited on 04/21/2023).

[38] GitHub. *opendigitaltwins-dtdl/DTDL.v2.md at master · Azure/opendigitaltwins-dtdl · GitHub*. 2023. URL: https://github.com/Azure/opendigitaltwins-dtdl/blob/master/DTDL/v2/DTDL.v2.md (visited on 04/27/2023).

[39] Á. Alonso et al. "Industrial data space architecture implementation using fiware". In: *Sensors (Switzerland)* 18.7 (2018). ISSN: 1424-8220. DOI: 10.3390/s18072226.

[40] Wolfgang Mahnke, Stefan-Helmut Leitner, and Matthias Damm. *OPC unified architecture*. OCLC: ocn268784080. Berlin: Springer, 2009. 339 pp. ISBN: 978-3-540-68898-3 978-3-540-68899-0.

[41] OWASP. *The Model*. 2023. URL: https://owaspsamm.org/model// (visited on 05/14/2023).

[42] ISO. *ISO 10303-1:2021(en), Industrial automation systems and integration — Product data representation and exchange — Part 1: Overview and fundamental principles*. 2021. URL: https://www.iso.org/obp/ui/# (visited on 05/14/2023).

[43] LSC Group. *A White Paper on Enabling Open Information Systems*. 2010.

[44] Sigma2 AS. *Om Sigma2 — Sigma2*. Om Sigma2. 2023. URL: https://www.sigma2.no/om-sigma2 (visited on 05/12/2023).

[45] S. Perez-Becker et al. "Implementation and validation of an advanced wind energy controller in aero-servo-elastic simulations using the lifting line free vortex wake model". In: *Energies* 14.3 (2021). DOI: 10.3390/en14030783. URL: https://www.scopus.com/inward/record.uri?eid=2-s2.0-85104574312&doi=10.3390%2fen14030783&partnerID=40&md5=51ea75ea0818dacd277328d66e221b4d.

[46] Jesse Thé and Hesheng Yu. "A critical review on the simulations of wind turbine aerodynamics focusing on hybrid RANS-LES methods". In: *Energy* 138 (Nov. 1, 2017), pp. 257–289. ISSN: 0360-5442. DOI: 10.1016/j.energy.2017.07.028. URL: https://www.sciencedirect.com/science/article/pii/S0360544217312033 (visited on 05/01/2023).

[47] M.S. Siddiqui et al. "Parametric Analysis Using CFD to Study the Impact of Geometric and Numerical Modeling on the Performance of a Small Scale Horizontal Axis Wind Turbine". In: *Energies* 15.2 (2022). ISSN: 1996-1073. DOI: 10.3390/en15020505.

[48] Maja Gunvor Erlandsen and Morten Sæther. "Kartlegging av beste strategi for 3D modellering og oppbygging av digitale tvillinger gjennom testing av etablert teori". Accepted: 2020-08-18T08:47:59Z Journal Abbreviation: Exploration of strategies for 3D CAD modelling and construction of digital twins through testing established theory Publication Title: 87. Master thesis. Norwegian University of Life Sciences, Ås, 2020. URL: https://nmbu.brage.unit.no/nmbu-xmlui/handle/11250/2672729 (visited on 03/06/2023).

[49] T. Nikou et al. "Investigation of the Aerodynamic Characteristics of a Small Horizontal Wind Turbine Blade with Open-Source Software Using Advanced Turbulence Modelling". In: *Chemical Engineering Transactions* 88 (2021), pp. 895–900. ISSN: 2283-9216. DOI: 10.3303/CET2188149.

[50] Adil Rasheed, Omer San, and Trond Kvamsdal. "Digital Twin: Values, Challenges and Enablers From a Modeling Perspective". In: *IEEE Access* 8 (2020). Conference Name: IEEE Access, pp. 21980–22012. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.2970143.

[51] M. Khan, T. Liu, and F. Ullah. "A new hybrid approach to forecast wind power for large scale wind turbine data using deep learning with tensorflow framework and principal component analysis". In: *Energies* 12.11 (2019). ISSN: 1996-1073. DOI: 10.3390/en12122229.

[52] Nikolay Dimitrov and Tuhfe Göçmen. "Virtual sensors for wind turbines with machine learning-based time series models". In: *Wind Energy* 25.9 (2022). _eprint:https://onlinelibrary.wiley.com/doi/pdf/10.1002/we.2 pp. 1626–1645. ISSN: 1099-1824. DOI: 10.1002/we.2762. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/we.2762 (visited on 02/23/2023).

[53] O. San, A. Rasheed, and T. Kvamsdal. "Hybrid analysis and modeling, eclecticism, and multifidelity computing toward digital twin revolution". In: *GAMM Mitteilungen* 44.2 (2021). ISSN: 0936-7195. DOI: 10.1002/gamm.202100007.

[54] Pablo Noever-Castelos, David Melcher, and Claudio Balzani. "Model updating of a wind turbine blade finite element Timoshenko beam model with invertible neural networks". In: *Wind Energy Science* 7.2 (Mar. 16, 2022). Publisher: Copernicus GmbH, pp. 623–645. ISSN: 2366-7443. DOI: 10.5194/wes-7-623-2022. URL: https://wes.copernicus.org/articles/7/623/2022/ (visited on 05/04/2023).

[55] Jack Jorgensen et al. "Requirements for the application of the Digital Twin Paradigm to offshore wind turbine structures for uncertain fatigue analysis". In: *Computers in Industry* 145 (Feb. 1, 2023), p. 103806. ISSN: 0166-3615. DOI: 10.1016/j.compind.2022.103806. URL: https://www.sciencedirect.com/science/article/pii/S0166361522002020 (visited on 01/11/2023).

[56] V. Tsiolakis et al. "Reduced order models for finite-volume simulations of turbulent flow around wind-turbine blades". In: Journal of Physics: Conference Series. Vol. 2018. ISSN: 1742-6588 Issue: 1. 2021. DOI: 10.1088/1742-6596/2018/1/012042.

[57] S. Pawar et al. "Hybrid analysis and modeling for next generation of digital twins". In: Journal of Physics: Conference Series. Vol. 2018. ISSN: 1742-6588 Issue: 1. 2021. DOI: 10.1088/1742-6596/2018/1/012031.

[58] Changhyun Kim et al. "Design, Implementation, and Evaluation of an Output Prediction Model of the 10 MW Floating Offshore Wind Turbine for a Digital Twin". In: *Energies* 15.17 (Jan. 2022). Number: 17 Publisher: Multidisciplinary Digital Publishing Institute, p. 6329. ISSN: 1996-1073. DOI: 10.3390/en15176329. URL: https://www.mdpi.com/1996-1073/15/17/6329 (visited on 05/04/2023).

[59] Jorge Gil. *Ansys Gets Into Open Source With GitHub — Ansys.* 2023. URL: https://www.ansys.com/blog/ansys-gets-into-open-source-with-github (visited on 05/07/2023).

[60] Giovanni Petrone. *Providing Open-Source Access to Ansys Fluent with PyFluent.* 2023. URL: https://www.ansys.com/blog/open-source-access-to-fluent-with-pyfluent (visited on 05/07/2023).

[61] Fei Tao et al. "Digital twin modeling". In: *Journal of Manufacturing Systems* 64 (July 1, 2022), pp. 372–389. ISSN: 0278-6125. DOI: 10.1016/j.jmsy.2022.06.015. URL: https://www.sciencedirect.com/science/article/pii/S0278612522001108 (visited on 01/16/2023).

[62] N. Luo et al. "Development of small-scale wind energy systems adaptable to climatic conditions using chattering torque control - PI pitch control and CAES strategy". In: *Renewable Energy and Power Quality Journal* 1.15 (Apr. 2017), pp. 494–499. ISSN: 2172038X, 2172038X. DOI: 10.24084/repqj15.369. URL: http://www.icrepq.com/icrepq17/369-17-luo.pdf (visited on 05/01/2023).

[63] M.S. Bonney et al. "Development of a digital twin operational platform using Python Flask". In: *Data-Centric Engineering* 3.1 (2022). DOI: 10.1017/dce.2022.1.

Norges miljø- og biovitenskapelige universitet
Noregs miljø- og biovitskapelege universitet
Norwegian University of Life Sciences

Postboks 5003
NO-1432 Ås
Norway