|  |  |
| --- | --- |
| CONTACT Software GmbH - OWL Maschinenbau | **Logo_Uni_1** |

Bachelorarbeit - Thesis

**Kurzzeitprognose der Turmschwingungskinematik von Onshore-Windenergieanlagen**

Autor: Zelgai Nemati

Matrikelnummer: 4516359

Studiengang: B. Sc. Systems Engineering

Erstprüfer: Prof. Dr. Klaus-Dieter Thoben

Zweitprüfer: M. Sc. Andreas Haselsteiner

Betreuer Contact Software: Dr. Nicole Göckel

Dr. Thomas Dickopf

Abgabedatum: 24.02.2022

Inhaltsverzeichnis

[1 Einleitung 1](#_Toc96881860)

[2 Zielstellung 2](#_Toc96881861)

[3 Stand der Forschung 3](#_Toc96881862)

[3.1 Turmschwingungskinematik 3](#_Toc96881863)

[3.2 Prognosemodelle 4](#_Toc96881864)

[3.2.1 Autoregression (AR) 5](#_Toc96881865)

[3.2.2 Moving-Average (MA) 7](#_Toc96881866)

[3.2.3 ARIMA 8](#_Toc96881867)

[3.2.4 SARIMA 10](#_Toc96881868)

[3.2.5 Prophet (Neuronales Netzwerk) 11](#_Toc96881869)

[4 Senvion Windkraftanlage 12](#_Toc96881870)

[5 Flucto Sensorbox 13](#_Toc96881871)

[6 Datenpipeline 14](#_Toc96881872)

[7 Kurzzeitprognose der Turmschwingungskinematik 15](#_Toc96881873)

[7.1 ARIMA 15](#_Toc96881874)

[7.1.1 Umsetzung 15](#_Toc96881875)

[7.1.2 Auswertung 15](#_Toc96881876)

[7.2 SARIMA 15](#_Toc96881877)

[7.2.1 Umsetzung 16](#_Toc96881878)

[7.2.2 Auswertung 16](#_Toc96881879)

[7.3 Prophet (Neuronales Netzwerk) 16](#_Toc96881880)

[7.3.1 Umsetzung 16](#_Toc96881881)

[7.3.2 Auswertung 16](#_Toc96881882)

[7.4 Vergleich der Prognosemodellergebnisse 17](#_Toc96881883)

[8 Contact Elements Integration 18](#_Toc96881884)

[8.1 Konfiguration des Asset-Dashboard 18](#_Toc96881885)

[8.2 Automatisierung Jobs und Vorlagen 18](#_Toc96881886)

[8.3 Implementierung der Datenpipeline 18](#_Toc96881887)

[8.4 Datenvisualisierung 18](#_Toc96881888)

[8.5 Echtzeitprognose der Turmschwingungskinematik 19](#_Toc96881889)

[8.6 Erweiterungsmöglichkeiten 19](#_Toc96881890)

[9 Fazit 20](#_Toc96881891)

[10 Literatur 21](#_Toc96881892)

**Abkürzungen und Formelzeichen**

**Abkürzungen**

**Formelzeichen**

|  |  |  |
| --- | --- | --- |
| Zeichen | Einheit | Erklärung |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Einleitung

Onshore-Windenergie liefert schon heute einen substanziellen Teil des Energiemixes und hat in den letzten zehn Jahren erhebliche Fortschritte gemacht. Beispiele sind größere und zuverlässigere Turbinen, steigende Bauhöhen und größere Rotorblattdurchmesser. Aufgrund dieses technologischen Fortschritts und der optimierten Skalierung konnten zwischen den Jahren 2010 und 2019 die Stromgestehungskosten um 39% (von 0,086 USD/kWh auf 0,053 USD/kWh) und die Installationskosten um 24% gesenkt werden [Ire20]. Diese Entwicklung führt zu einer gesteigerten Wettbewerbsfähigkeit der Windenergie Branche und somit haben 75% aller im Jahr 2019 in Auftrag gegebenen Windprojekte niedrigere Stromgestehungskosten als die billigste fossile Energiequelle [Ire20]. Die Windenergie ist auf dem Weg eine tragende Säule des zukünftigen grünen Energiemixes zu werden [Ire20].

Heutzutage werden die meisten Offshore Windenergieanlagen komponentenweise installiert und um die Kosten weiterhin zu senken, ist es insbesondere notwendig, den Installationsprozess zu optimieren. Die Montage der Rotorblätter stellt dabei die größte Herausforderung dar, denn hier ist hohe Präzision und Sorgfalt erforderlich, um das Blattende in die Rotornabe einzusetzen [San20, San21]. Der Wind übt Lasten auf die mechanischen Strukturen der Windkraftanlage aus und die daraus resultierenden Relativbewegungen zwischen Turm und den Rotorblättern erschweren die Blattmontage [San20, San21]. Überschreitet die Relativbewegung einen definierten Schwellenwert, kann die Installation nicht mehr durchgeführt werden, da Schäden beim Montagevorgang zu erwarten sind und es kommt zu einer kostspieligen Verzögerung [San20, San21].

Deshalb werden aktuell Wetterlimits zur Planung solcher Installationsmaßnahmen verwendet, wobei eine direktere limitierende Größe als das Wetter, eine Prognose der Turmschwingung wäre. Jedoch ist bislang, aus wissenschaftlicher Perspektive, nicht beantwortet, welche Modelltypen am besten geeignet sind, mit welcher Genauigkeit sich eine solche Prognose der Turmschwingungskinematik verwirklichen lässt und wie diese Erkenntnisse optimal in der Praxis genutzt werden können z.B. in Form einer Echtzeit IoT Anwendung.

# Zielstellung

Aktuell werden wie beschrieben Wetterlimits bei der Planung und Durchführung von Installationsvorgängen verwendet, da bei einer zu hohen Relativbewegung zwischen Turm und Rotorblatt eine Installation nicht möglich ist und es zu Schäden an den Komponenten kommen kann. Eine direktere limitierende Größe als das Wetter, wäre eine Prognose der Turmschwingung (in Abbildung 2-1 blau markiert). Deswegen soll im Rahmen dieser Abschlussarbeit die Forschungsfrage beantwortet werden, wie genau die Schwingungskinematik eines Windenergieanlagen-Turms für die nächsten Sekunden und Minuten vorhergesagt werden kann.

Es sollen drei verschiedene Prognosemodelle (ARIMA, SARIMA, Prophet) aufgestellt werden, welche auf GitHub unter einer MIT Lizenz verfügbar sind und im Kapitel 3.2 genauer thematisiert werden. Die Genauigkeit dieser Modelle wird dann für verschiedene Zeiträume getestet, sodass eine fundierte Beantwortung der wissenschaftlichen Fragestellung sich aus diesen Ergebnissen ableitet. Diese optimierten Prognosemodelle sollen dann im letzten Schritt in der Contact Elements for IoT Plattform entwickelt werden. Des Weiteren sollen Dashboard Widgets entwickelt werden, welche die Ergebnisse dieser Prognosemodelle visualisieren. Durch Oberflächenkonfiguration soll es ebenfalls möglich sein diese Modelle auf andere Datensätze und Fragestellungen zu adaptieren.

Diagram, engineering drawing

Description automatically generated

**Abbildung 2-1** Prognose der Turmschwingungskinematik

# Stand der Forschung

## Turmschwingungskinematik

Die Analyse der Turmschwingungskinematik ist notwendig, um den Installationsprozess von Windenergieanlagen zu optimieren. Im Jahr 2020 wurden im Rahmen einer Messkampagne Daten zur Turmkinematik, während der Installation von Offshore Windenergieanlagen, erhoben [San21]. Die Auswertung dieser Positionsdaten hat ergeben, dass die Schwingungskinematik des Turms eine besondere Charakteristik aufweist, welche der nachfolgenden Abbildung 3-1 zu entnehmen ist [San21].

Die Turmschwingungskinematik wird maßgeblich vom Wind und anderen Umweltparametern beeinflusst, die in der Literatur häufig als „zufällige“ Störgrößen aufgefasst werden. Im Rahmen dieser Abschlussarbeit wird von einem stationären Datensatz ausgegangen, da der Erwartungswert und die Varianz nicht zeitabhängig sind und keine Saisonalität vorliegt [Neu11]. Das ist durch die Frequenz und zeitliche Spanne (maximal 24 Stunden) der Messreihen zu begründen.

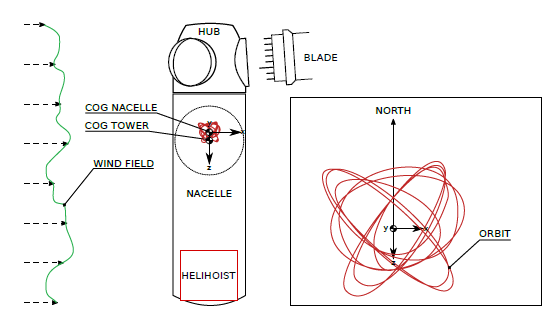


Abbildung 3‑1 Auswertung der Installationsdaten vom Offshore Windpark:

„Trianel Windpark Borkum 2“ [San21]

## Prognosemodelle

Um eine fundierte Auswahl der zu implementierenden Prognosemodelle zu gewährleisten, werden im Folgenden konkrete Kriterien aus der Charakteristik der Turmschwingungskinematik und der vorliegenden Problemstellung hergeleitet.

**Kriterium 1**

Wie bereits im Kapitel 3.1 beschrieben wurde, wird davon ausgegangen, dass es sich bei den Umwelteinflüssen um zufällige Störgrößen handelt. Deshalb sollte ein gewähltes Prognosemodell dazu in der Lage sein sich zeitlich verändernde Eigenschaften des Schwingungssignals berücksichtigen zu können.

**Kriterium 2**

Da es sich bei den Umwelteinflüssen um zufällige Störgrößen handelt, kann keine Aussage über das auftretende Rauschen der Zeitreihendaten getroffen werden. Deshalb sollte das gewählte Prognosemodell keine besonderen Anforderungen an das Rauschverhältnis stellen

**Kriterium 3**

Nachdem das Modell validiert und die notwendigen Prognoseparameter optimiert wurden, sollte die Berechnung sehr schnell sein, speziell vor dem Hintergrund der Contact Elements for IoT Integration.

**Kriterium 4**

Das gewählte Prognosemodell sollte erfahrungsgemäß präzise Ergebnisse mit einer hohen Genauigkeit, für die Strukturkinematikprognose liefern.

### Autoregression (AR)

Autoregressive Prognosemodelle, abgekürzt AR-Prognosemodelle, stellen die einfachste Schätzmethode dar und können nur auf stationäre Datensätze angewandt werden [Neu11, Sch01]. AR-Modelle prognostizieren zukünftige Parameterwerte, anhand der vergangenen Werte, welche im Fachjargon auch als „Lags“ bezeichnet werden [Neu11, Sch01]. Diese Abhängigkeit zu vergangenen Werten (bzw. der Einfluss der vergangenen auf zukünftige Werte) kann der nachfolgenden Abbildung 3-2 entnommen werden.

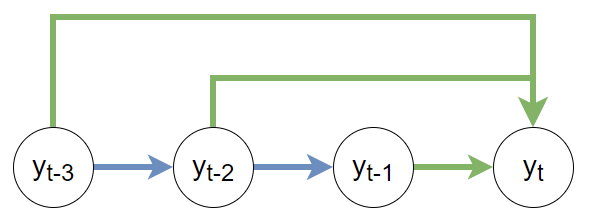


Abbildung 3‑2 AR-Modell Funktionsweise

Ein AR Modell erster Ordnung bzw*. AR(1)* Modell wird durch die folgende Formel beschrieben:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | (3-1) |  |

Für dieses Autoregressionsmodell mit dem Grad 1, entspricht der Parameterwert zum Zeitpunkt *t*, der Summe aus dem Mittelwert, dem gewichteten Parameterwert zum Zeitpunkt *t-1* und einem Fehlerwert *e*, welcher in der einschlägigen Literatur auch als „White-Noise“ bezeichnet wird [Neu11, Sch01].

Diese mathematische Definition des eingeführten Prognosemodelltyps, lässt sich auch auf beliebig viele Lags erweitern und ist somit nicht wie in Formel 3-1 auf einen Lag beschränkt. Eine allgemeine Definition für *i* Lags, ist der Formel 3-2 zu entnehmen [Neu11, Sch01]:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | (3-2) |  |

Bei der Auswahl des AR Modells bzw. bei der Bestimmung des optimalen Komplexitätsgrads gibt es einen Grundsatz. Dieser Grundsatz besagt, dass wenn zwei Prognosemodelle die beinahe gleiche Genauigkeit der Prognose hervorbringen, das Modell mit dem niedrigeren Grad bzw. der niedrigeren Komplexitätsstufe verwendet wird [Neu11, Sch01].

Bei dieser Abwägung zwischen einer möglichst niedrigen Komplexität und einer möglichst hohen Genauigkeit des Prognosemodells und der sich daraus ableitenden Fragestellung, welche Lags mit in die Prognose einbezogen werden sollen, wird die partielle Autokorrelationsfunktion (abgekürzt PACF) herangezogen [Neu11, Sch01]. Diese Funktion gibt an, welche Lags einen signifikanten direkten Einfluss (in Abbildung 3-2 grün markiert) auf den aktuellen Parameterwert haben und in das Prognosemodell mit einfließen sollten. So lassen sich auch Lags identifizieren, die einen geringen direkten Einfluss auf den zu prognostizierenden Wert haben und man kann somit die Komplexität des Prognosemodells reduzieren, ohne einen bedeutenden Genauigkeitsverlust hinzunehmen [Neu11, Sch01].

Der folgenden Abbildung 3-3 ist eine bespielhafte PACF zu entnehmen.

TAUSCHEN, BESCHREIBUNG UND ERLÄUTERUNG DER ABBILDUNG

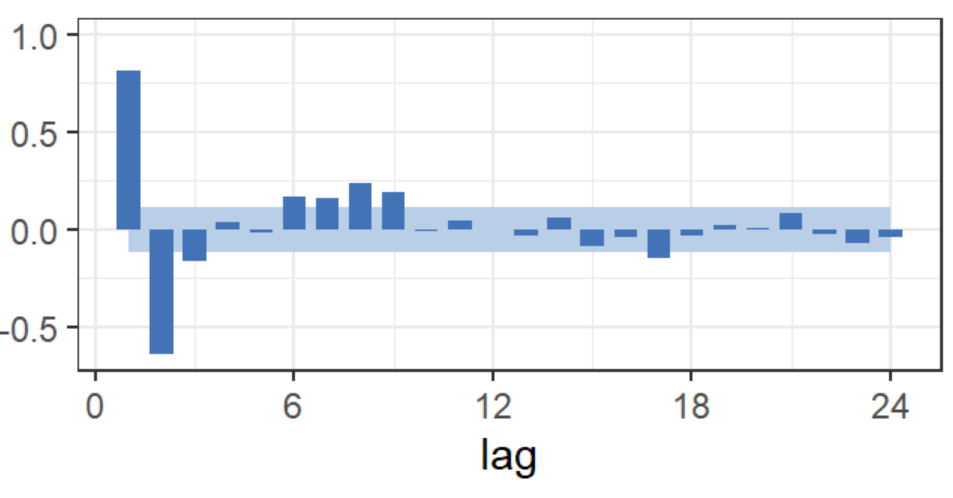


Abbildung 3‑2 AR-Modell Funktionsweise

Im Rahmen dieser Abschlussarbeit wird kein AR Prognosemodell verwendet, da das ARIMA Modell (welches im Kapitel 3.2.3 theoretisch eingeführt wird) auf diesem aufbaut und erhebliche Vorteile gegenüber diesem Modelltyp hat.

### Moving-Average (MA)

Moving-Average Prognosemodelle, abgekürzt MA-Prognosemodelle, sind ebenso wie AR-Modelle nicht sehr komplex und können ebenfalls nur auf stationäre Datensätze angewandt werden [Neu11, Sch01]. MA-Modelle prognostizieren nicht anhand der vergangenen Parameterwerte (so wie es AR-Modelle machen), sie prognostizieren anhand der vergangenen Fehlerwerte [Neu11, Sch01]. Dazu kann man sich zwei Graphen vorstellen. Einer bildet den tatsächlichen Verlauf ab und der andere bildet die Prognose dieser Werte ab. Nun wird für die Lags die Differenz aus Prognosewert und tatsächlichem Wert berechnet. Man erhält den Fehlerwert für den betrachteten Lag. Für den ersten Wert einer Zeitreihe wird als Prognosewert der Durchschnitt aller Parameterwerte genommen. Dieses Vorgehen ist der folgenden Abbildung 3-3 zu entnehmen, wo die tatsächlichen Werte in blau, die Prognosewerte in Rot und die errechneten Fehlerwerte in orange dargestellt sind.

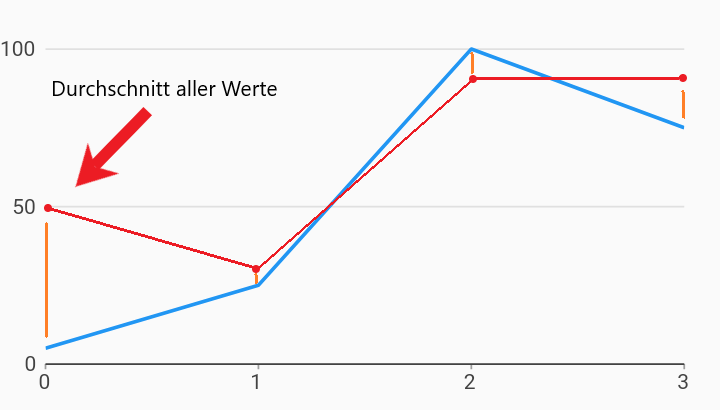


Abbildung 3‑3 MA Bestimmung des initialen Prognosewerts und der folgenden Fehlerwerte

Wie bereits bei dem AR Modell, kann ein MA Modell beliebig viele Lags zur Prognose verwenden und es kann somit ein beliebig hoher grad gewählt werden. Ein *MA(i)* Modell, wird durch die folgende Formel 3-3 beschrieben [Neu11, Sch01]:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | (3-3) |  |

Im Rahmen dieser Abschlussarbeit wird ebenfalls kein MA Prognosemodell verwendet, da das ARIMA Modell (welches im Kapitel 3.2.3 theoretisch eingeführt wird) auf diesem aufbaut und erhebliche Vorteile gegenüber diesem Modelltyp hat.

### ARIMA

Autoregressive Integrated Moving Average Prognosemodelle, abgekürzt ARIMA-Prognosemodelle, sind eine Kombination der beiden schon thematisierten Modelltypen [Sia18, Zha01]. AR-Modelle und MA-Modelle stehen in einem besonderen Zusammenhang, der im Folgenden aus der Formel 3-1, welche ein *AR(1)* Modell beschreibt, mathematisch hergeleitet wird:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | (3-1) |  |
|  | Ausschreiben des blau markierten Parameters |  |  |
|  |  |  |  |
|  | Ausschreiben des blau markierten Parameters |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  | (3-4) |  |

Die hergeleitete Formel 3-4 entspricht einem *MA (∞)* Modell, was der Formel 3-3 entnommen werden kann. Somit besteht der Zusammenhang *AR (1) = MA (∞)*.

Diese beschriebene Beziehung der beiden Modelltypen (AR und MA) ermöglicht es einem Prognosemodelle mit erhöhter Genauigkeit und verringerter Komplexität zu implementieren, indem man die einzelnen Terme eines AR und MA Modells in einem gemeinsamen ARIMA Modell kombiniert [Ade14, Sia18, Zha01]. Diese Kombination der AR und MA Terme kann der folgenden Formel 3-5 beispielhaft entnommen werden [Ade14, Zha01]:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | (3-5) |  |

Anders als bei den bisher behandelten Prognosemodellen, gibt es beim ARIMA Modell drei Grad-Parameter, welche in der einschlägigen Literatur *p*, *d* und *q* genannt werden [Ade14, Sia18, Zha01]. Die Notation für diesen Modelltyp kann der folgenden Formel 3-6 entnommen werden:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | (3-5) |  |

Der Parameter *p* beschreibt den Grad des AR-Teils, der Parameter *d* beschreibt den Grad des I-Teils und der Parameter *q* beschreibt den Grad des MA-Teils [Ade14, Sia18, Zha01].

Somit wird ein *ARIMA(1, 0, 1)* Modell, mit einem AR-Teil, keinem I-Teil und einem MA-Teil, durch die folgende Funktion 3-6 beschrieben:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | (3-6) |  |

Die thematisierte Komplexitätsreduktion führt dazu, dass nach der Validierung der Prognoseparameter eine schnelle Kalkulation der Prognose möglich ist [Sia18, Zha01]. Ein weiterer entscheidender Vorteil von ARIMA und SARIMA (siehe Kapitel 3.2.4) Modellen ist, dass sie sowohl auf stationäre als auch auf nicht stationäre Datensätze anwendbar sind, welches durch eine Integration der Zeitreihe ermöglicht wird (dafür steht das *I* im Namen des Modeltyps) [Sia18, Zha01]. Außerdem berechnen beide Modelle die Prognose auf Grundlage der Vergangenheitswerte (und der vergangenen Fehlerwerte) und gewichten die zeitlich näherliegenden Werte stärker als Werte, die lange in der Vergangenheit zurück liegen. Dadurch ist sichergestellt, dass die Modelle auf sich ändernde Eigenschaften des Schwingungssignals optimal reagieren können [Sia18].

Im Kapitel 6.1.1, indem die praktische Umsetzung des ARIMA Modells thematisiert wird, wird ein bereits bestehender Algorithmus verwendet, um das optimale ARIMA Modell, passend zur jeweiligen Fragestellung und der zu analysierenden Zeitreihe, zu identifizieren.

### SARIMA

Seasonal Autoregressive Integrated Moving Average Prognosemodelle, abgekürzt SARIMA-Prognosemodelle, sind eine Weiterentwicklung von ARIMA-Modellen [Che06, Fan16, Liu21, Nob01]. SARIMA Prognosemodelle beziehen zusätzliche saisonale Parameter zur Prognose mit ein und können somit für bestimmte Zeitreihen optimierte Ergebnisse hervorbringen, weshalb sie sich einer hohen Beliebtheit erfreuen [Che06, Liu21]. Besonders vor dem Hintergrund, der im Rahmen dieser Abschlussarbeit zu behandelnden Fragestellung, bei der die Turmkinematik durch Umweltparameter maßgeblich beeinflusst wird, sind durch das SARIMA Modell Prognoseergebnisse mit einer hohen Genauigkeit zu erwarten, da diese einer Saisonalität unterliegen.

Die Notation für diesen Modelltyp kann der folgenden Formel 3-8 entnommen werden [Che06, Fan16, Liu21]:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | (3-5) |  |

Das *P* steht für die Anzahl der saisonalen AR Terme, *D* für die Anzahl der saisonalen I Terme, *Q* für die Anzahl der Saisonalen MA Terme und *s* für die Länge der Saisonalität [Che06, Fan16, Liu21]. Der folgenden Formel 3-9 kann eine allgemeine Definition des SARIMA Modells entnommen werden [Fan16]:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | (3-5) |  |

Somit kann ein ARIMA(**1**,0,**1**)(**2**,0,**1**)6 Modell durch die folgende Formel beschrieben werden:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | (3-6) |  |

Die Vorteile des ARIMA Prognosemodells, welche im Kapitel 3.2.3 thematisiert wurden, gelten ebenso im Kontext des SARIMA Modells. Außerdem sind wie beschrieben durch die saisonale Komponente bessere Ergebnisse zu erwarten. Diese Vermutung wird dann im Kapitel 6.2.2 abschließend beantwortet.

Im Kapitel 6.2.1, indem die praktische Umsetzung des SARIMA Modells thematisiert wird, wird ein bereits bestehender Algorithmus verwendet, um das optimale SARIMA Modell, passend zur jeweiligen Fragestellung und der zu analysierenden Zeitreihe, zu identifizieren.

### Prophet (Neuronales Netzwerk)

Um die Eingangs formulierte Fragestellung, wie genau die Turmkinematik für die nächsten Sekunden und Minuten vorhergesagt werden kann, fundiert beantworten zu können, sollen im Rahmen dieser Arbeit nicht nur statistische Methoden, sondern auch Neuronale Netze zur Kurzzeitprognose der Turmschwingungskinematik verwendet werden, da diese eventuell bessere Ergebnisse hervorbringen.

Prophet ist ein vom Unternehmen Facebook entwickeltes Zeitreihen-Prognose-Tool bzw. Framework, welches open-source ist und mithilfe der Programmiersprachen R und Python benutzt werden kann [Dar21, Tay17, Toh20]. Eine detaillierte Dokumentation wurde vom Großkonzern erstellt und ist unter folgendem Link aufrufbar:

**https://facebook.github.io/prophet/docs/quick\_start.html#python-api**

Die Analyse bzw. Prognose durch Prophet basiert auf einem neuronalen Netz, welches aus drei Grundfunktionen zusammengesetzt ist [Dar21, Toh20]. Dies kann der folgenden Formel 3-4 entnommen werden:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  | (3-4) |  |

Der grundlegende Trend einer Zeitreihe wird durch g(t) modelliert, die mögliche Saisonalität einer Zeitreihe wird durch s(t) dargestellt und h(t) steht für mögliche Lücken im Datensatz bzw. in der zu analysierenden Zeitreihe [Dar21, Toh20]. Dadurch ist es möglich das initial für wirtschaftswissenschaftliche Fragestellungen entwickelte Netz (ähnlich wie beim ARIMA Modell) auf diverse Fragestellungen zu adaptieren. [Tay17]

Im Rahmen dieser Arbeit wird Prophet gegenüber anderen Neuronalen Netzwerken bevorzugt, da es hoch konfigurierbar ist und keine besonderen Anforderungen an den Datensatz in Bezug auf Stationarität stellt [Dar21, Tay17, Toh20]. Ein weiteres ausschlaggebendes Argument ist, dass es mithilfe von Prophet möglich ist die Prognose in Echtzeit zu testen und optimieren, sodass es besonders in IoT Szenarien zu besseren Ergebnissen führt. [Tay17]

# Senvion Windkraftanlage

Nennleistung: 3,37 MW

Nabenhöhe 128 m

Rotordurchmesser 104 m

Rotorblattlänge 50,8 m

Anlaufgeschwindigkeit 3,5 m/s

Abschaltgeschwindigkeit 25m / s

Forschungsplattform zur Entwicklung und Erprobung praxistauglicher Lösungen in der Anlagentechnik

Liefert Messdaten für verbesserungen in Konstruktion, Werkstoffwahl, Fertigung und Steuerung von WEA

Erprobung neuartiger Sensoren zur Überwachung von Getriebe, Triebstrang und Rotorblatt

# Flucto Sensorbox

Grundlegender Aufbau

Welche Daten werden ermittelt (was ist IMU)

In welchem Format liegen die Daten dann vor?

# Datenpipeline

Wie werden die Daten verarbeitet?

Was machen die einzelnen Arbeitsschritte?

FlowChart und Beschreibung der einzelnen Schritte.

# Kurzzeitprognose der Turmschwingungskinematik

Kurz auf die Grundlagenkapitel eingehen und sagen, dass diese verwendet werden.

Auf die folgende Gliederung der Kapitel eingehen und wie die Genauigkeit der Modelle im Auswertungskapitel gemessen wird.

## ARIMA

Kurz darauf eingehen, in welchem Kapitel die Grundlagen dieses Modells erklärt wurden.

### Umsetzung

Wie ist die Umsetzung (am besten mit einem Flow-Chart und einer textuellen Beschreibung)?

Wo habe ich mich inspiriert?

Wo kann man den Code und den HTML Export finden?

### Auswertung

Wie genau ist das Modell für einen gegebenen Zeitraum (also die erste Minute)?

Wie lange hat das Training und die anschließende Prognose gedauert?

Verweis auf den abschließenden Vergleich der Modelle.

## SARIMA

Kurz darauf eingehen, in welchem Kapitel die Grundlagen dieses Modells erklärt wurden.

### Umsetzung

Wie ist die Umsetzung (am besten mit einem Flow-Chart und einer textuellen Beschreibung)?

Wo habe ich mich inspiriert?

Wo kann man den Code und den HTML Export finden?

### Auswertung

Wie genau ist das Modell für einen gegebenen Zeitraum (also die erste Minute)?

Wie lange hat das Training und die anschließende Prognose gedauert?

Verweis auf den abschließenden Vergleich der Modelle.

## Prophet (Neuronales Netzwerk)

Kurz darauf eingehen, in welchem Kapitel die Grundlagen dieses Modells erklärt wurden.

### Umsetzung

Wie ist die Umsetzung (am besten mit einem Flow-Chart und einer textuellen Beschreibung)?

Wo habe ich mich inspiriert?

Wo kann man den Code und den HTML Export finden?

### Auswertung

Wie genau ist das Modell für einen gegebenen Zeitraum (also die erste Minute)?

Wie lange hat das Training und die anschließende Prognose gedauert?

Verweis auf den abschließenden Vergleich der Modelle.

## Vergleich der Prognosemodellergebnisse

Auf das RMSE Modell eingehen und was es aussagt.

Kurz auf die Rechenzeit der verschiedenen Modelle eingehen.

Welches Modell ist für welchen Use-Case gedacht?

# Contact Elements Integration

Was ist Contact Elements?

Was ist das Ziel von Contact Software und dieser Integration?

Kurze Beschreibung der folgenden Kapitel

## Konfiguration des Asset-Dashboard

Was ist ein Asset?

Was für Möglichkeiten hat man bei der Konfiguration?

Wie ist das Asset konfiguriert?

## Automatisierung Jobs und Vorlagen

Was ist die Automatisierung und in welchem Rahmen wurde diese entwickelt?

Welche Möglichkeiten hat man hier eine Automatisierung zu gestalten?

Wie sieht es mit der Wiederverwendbarkeit der Subtemplates aus?

## Implementierung der Datenpipeline

Wie wird Kapitel 6 in CE4IoT umgesetzt (genau die parallelen aufzeigen)?

## Datenvisualisierung

Welche Standardfunktionalität gibt es?

Warum reicht das nicht aus und wie kann man diese Möglichkeiten erweitern?

Wo kann ich die Erweiterung finden?

Wie kann ich die Erweiterung verwenden?

Wie sieht die Erweiterung aus?

## Echtzeitprognose der Turmschwingungskinematik

Auf Kapitel 8.2 eingehen und aufzeigen, wie das ARIMA Modell mit der zur Verfügung stehenden Funktionalität umgesetzt wurde?

## Erweiterungsmöglichkeiten

Wie kann man im IoT Kontext diese Prognose verbessern bzw. optimieren?

Wie kann man die Lösung auf andere Fragestellungen möglichst Ressourcenschonend adaptieren?

# Fazit

ffff

# Literatur

**[Ade14]** Adebiyi A. A., & Adewumi A. O., & Ayo C. K. (2014).

Stock Price Prediction Using the ARIMA Model.

*UKSim-AMSS 16th International Conference on Computer Modelling and Simulation*, 2014, pp. 106-112.

https://doi.org/10.1109/UKSim.2014.67

**[Che06]** Chen K. Y., & Wang C.H. (2006).

A hybrid SARIMA and support vector machines in forecasting the production values of the machinery industry in Taiwan.

*Expert Systems with Applications*, Volume 32, pp. 254-264.

https://doi.org/10.1016/j.eswa.2005.11.027

**[Dar21]** Daraghmeh M., & Agarwal A., & Manzano R., & Zaman M. (2021).

Time Series Forecasting using Facebook Prophet for Cloud Resource Management.

2021 IEEE International Conference on Communications Workshops (ICC Workshops), 2021, pp. 1-6.

https://doi.org/10.1109/ICCWorkshops50388.2021.9473607

**[Fan16]** Fang T., & Lahdelma R. (2016).

Evaluation of a multiple linear regression model and SARIMA model in forecasting heat demand for district heating system.

*Applied Energy*, Volume 179, pp. 544-552.

https://doi.org/10.1016/j.apenergy.2016.06.133

**[Ire20]** IRENA (2020).

Renewable Power Generation Costs in 2019.

*International Renewable Energy Agency*, pp. 46-60.

**[Liu21]** Liu S., & Jiao Y., & Sun Q., & Jiang J. (2021).

Estimation of Sea Level Change in South China Sea from Satellite Altimetry Data.

*Scientific Programming*, 2021, pp. 1-7.

https://doi.org/10.1155/2021/6618135

**[Neu11]** Neusser K. (2011). Zeitreihenanalyse in den Wirtschaftswissenschaften. Berlin: Springer-Verlag.

**[Nob01]** Nobre F. F., & Monteiro A. B., & Telles P. R., & Williamson G. D. (2001).

Dynamic linear model and SARIMA: a comparison of their forecasting performance in epidemiology.

*Statistics in medicine*, Volume 20, pp. 3051-3069.

https://doi.org/10.1002/sim.963

**[San20]** Sander A., & Haselsteiner A. F., & Barat K., & Janssen M., & Oelker S., Ohlendorf J., & Thoben K. (2020).

Relative Motion During Single Blade Installation: Measurements From the North Sea.

ASME 2020 39th International Conference on Ocean, Offshore and Arctic Engineering, Volume 9.

https://doi.org/10.1115/OMAE2020-18935

**[San21]** Sander A., & Haselsteiner A., & Holmann B. (2021).

Could mass eccentricity explain the formation of orbits in the wind turbines?

https://www.researchgate.net/publication/355582885

**[Sch01]** Schlittgen R., & Streitberg B. H. J. (2001).

Zeitreihenanalyse.

München: R. Oldenbourg Verlag.

**[Sia18]** Siami-Namini S., & Tavakoli N., & Namin A. S. (2018).

A Comparison of ARIMA and LSTM in Forecasting Time Series.

*2018 17th IEEE International Conference in Machine Learning and Applications (ICMLA)*, 2018, pp. 1394-1401.

https://www.doi.org/10.1109/ICMLA.2018.00227

**[Tay17]** Taylor S. J., & Letham B. (2017).

Forecasting at scale.

*PeerJ Preprints*.

https://doi.org/10.7287/peerj.preprints.3190v2

**[Toh20]** Toharudin T., & Pntoh R., & Caraka R., & Zahroh S., & Lee Y., & Chen R. (2020).

Employing long short-term memory and Facebook prophet model in air temperature forecasting.

*Communications in Statistics – Simulation and Cimputation.*

https://doi.org/10.1080/03610918.2020.1854302

**[Zha01]** Zhang G. P., (2001).

Time series forecasting using a hybrid ARIMA and neuronal network model.

*Neurocomputing*, Volume 50, pp. 159-175.

https://doi.org/10.1016/S0925-2312(01)00702-0