



DEVELOPMENT OF A METHODOLOGY TO SIMPLIFY DESIGN LOAD COLLECTIVES FOR OPTIMIZING THE PITCH BEARING LOAD DISTRIBUTION BY STIFFENING PLATES

COURSE OF STUDY:

SUBMITTED TO:

SUBMITTED BY :

MATRICULATION-NR.:

DURATION :

EXTERNAL SUPERVISOR:

INTERNAL SUPERVISOR:

CHAIR:

SUBMISSION DATE:

COMPUTATIONAL SCIENCE AND ENGINEERING

STIFTUNGSLEHRSTUHL FÜR WINDENERGIETECHNIK

FAKULTÄT DER UNIVERSITÄT ROSTOCK

SAI DEV LAKKAKULA

217100215

20 WEEKS

M.Sc. AMIN LORIEMI (CWD, RWTH AACHEN)

PROF. DR. RER. NAT. HABIL. UWE RITSCHEL

CHAIR OF WIND ENERGY TECHNOLOGY

4 NOVEMBER 2020

Abstract

The pitch bearing has a crucial role in adjusting the movement of the rotor blade connected to the rotor hub. By adjusting the pitch angle, the aerodynamic properties of the rotor are adapted according to the turbine control, which enables the power of the wind turbine to be regulated. Thereby pitch bearings are subjected to substantial aerodynamic loads transferred from the blade to the nacelle. For the design of the pitch adjustment system, with the pitch bearing as the central component, the evaluation of load collectives is necessary. These collectives are usually derived from high-resolution time series of a multi-body system (MBS) load calculation. Former studies have shown, that by shape optimizing surrounding structures of a pitch bearing, damaging effects like contact ellipse truncation can be significantly reduced and a better internal load distribution achieved. Taking high-resolution time series as a basis for such a shape optimization results in high computational costs, which may be unnecessary for finding an optimum solution.

In this thesis, dynamic load collectives acting on the pitch bearing are being analyzed and transformed into a collective of fewer considered loads, such that the excluded loads do not result in a significant error regarding the calculated internal loading of the bearing. After reducing the load collective, a suitable step size must be determined for each of the remaining loads, such that a simplification into discrete load steps approximately resembles the original continuous time series. Using FEM calculated bearing loads characteristics, machine learning models determine the critical loads occurring over the entire time series. With that the computational cost of optimizing the pitch bearing arrangement on basis of a load collective can be reduced without losing relevant information for optimization.

Acknowledgment

Foremost, I am thankful to Univ. Prof. Dr. Ing. Georg Jacobs , head of Chair for Wind Power Drives, RWTH Aachen for providing me an opportunity to be the part of this esteemed project. I wish to express my deepest gratitude to Amin Loriemi M.Sc. for enthusiasm and immense knowledge, endless patience, and continuous support in completing my Master Thesis at Chair for Wind Power Drives. It was a good learning experience under guidance. Research exposure with you helped me in mastering many new skills and helped me work on exciting projects during my thesis at CWD.

My sincere thanks to the university supervisor, Prof. Dr. rer. nat. habil. Uwe Ritschel, head of Chair for Wind Turbine Technology for supporting me to research at CWD, RWTH Aachen. His guidelines and regular updates on the thesis are valuable in completion of the thesis.

I am deeply indebted to my parents Srinivas Lakkakula and Premalatha Lakkakula, for their love and support throughout my life. Thank you both for helping and believing in me to focus on my studies during hard times. I would also like to extend my gratitude to my friends Sai Prakash Modali and Sahith Shanagonda in motivating me scientifically and standing by me during the sleepless nights in completion of this thesis.

Contents

Abstract	i
Acknowledgment	ii
List of Figures	v
List of Tables	vii
Nomenclature	viii
1 Introduction	1
1.1 Classification of Wind Turbines	1
1.1.1 Vertical Axis Wind Turbine	1
1.1.2 Horizontal Axis Wind Turbine	2
2 State of Art	4
2.1 Tower	5
2.2 Rotor	6
2.3 Pitch System	7
2.4 Pitch Bearing Design	8
2.4.1 Rolling Element-Raceway contact	9
2.4.2 Bearing Fatigue Life	10
2.4.3 Bearing Static Capacity	11
2.5 Pitch Bearing Failure	12
2.6 Bridging FEM with Machine Learning	14
2.6.1 Machine Learning	15
2.6.2 Linear Regression	17
2.6.3 Polynomial Regression	17
2.6.4 Random Forest Regressor	18
2.6.5 MLP Regressor	19

3 Problem statement	21
4 Method	22
4.1 Finite Element Calculation	22
4.1.1 Rolling Element modeling	23
4.1.2 Load Discretization Methodology	24
4.1.3 Calculation and Database Generation	26
4.2 Approximation using Machine Learning	28
4.2.1 Identifying Critical Loads	29
5 Results and Discussion	30
6 Conclusion	38
A Appendix	42

List of Figures

1	Types of Vertical axis wind turbine[1]	2
2	Variation of Windspeed with increase in height[2]	5
3	Single and double row slewing bearing with polyamide spacers [3]	7
4	Two elastic elastic bodies under compressive load Q [4]	9
5	Fretting wear on pitch bearing raceway[5]	13
6	Crack initiation at bolt thread holes[5]	14
7	Machine Learning workflow	16
8	Decision tree data splitting at nodes	18
9	MLP regressor with one hidden layer	20
10	Flow chart of the Methodology	22
11	Pitch bearing force and bending moment notation[6]	24
12	Axial force rounding	25
13	Bending moment rounding	25
14	Variation of unique load cases with change in force and moment step size	26
15	Critical contact force for ellipse truncation[7]	26
16	Contact force and Truncation factor distribution over bearing circumference	27
17	Maximum force learning curve of Random forest regression	30
18	Maximum force learning curve of MLP regression	31
19	Load combination sample for design loads	31
20	Maximum force approximation by all regressors	32
21	Truncation distribution by polynomial and MLP regression	33
22	Truncation distribution by random forest and MLP regression	33
23	Truncation distribution by linear and MLP regression	34
24	Maximum force distribution random forest and MLP regression	34
25	Standard deviation distribution by random forest and MLP regression	35
26	Maximum force distribution polynomial and MLP regression	35

27	Standard deviation distribution by polynomial and MLP regression	36
28	Maximum force distribution linear and MLP regression	36
29	Standard deviation distribution linear and MLP regression	37
30	Learning curve of Polynomial regressor for Mean and Maximum force . .	46
31	Learning curve of Polynomial regressor for Standard deviation and Truncation factor	46
32	Learning curve of linear regressor for Mean and Maximum force	46
33	Learning curve of linear regressor for Standard deviation and Truncation factor	47
34	Learning curve of Random Forest for Mean and Maximum force	47
35	Learning curve of Random Forest for Standard deviation and Truncation factor	47

List of Tables

1	Pitch bearing design specifications[7]	23
2	Bearing load case and parameters variation	28
3	Hyperparameter tuning of the regression models	29
4	Regression models prediction error	30
5	MLP regressor approximation error with change in training data size . . .	37

Nomenclature

Acronyms

- Abaqus/CAE** Complete Abaqus Environment
ANSI American National Standards Institute
ANN Artificial Neural Network
CWD Chair for Windpower Drives
FEM Finite Element Method
FVM Finite Volume Method
HAWT Horizontal Axis Wind Turbine
HB Brinell Hardness
IPC Individual Pitch Control
ISO International Organisation for Standardization
MATLAB Matrix Laboratory
MBS Multi-body System
ML Machine Learning
MLP Multi-Layer Perceptron Regressor
NREL National Renewable Energy Laboratory
PCA Principal Component Analysis
RWTH Rheinisch-Westfälische Technische Hochschule
SF Static Load Factor
SSR Sum of Squared Residuals
SVM Support Vector Machines
VAWT Vertical Axis Wind Turbine

Symbols

ρ	curvature of a plane
θ	amplitude of oscillation
r	radius of curvature
m	mass of air moving
a	semimajor axis of contact ellipse
b	semiminor axis of contact ellipse
α_o	contact angle of rolling element
k	von karman constant
v_w	velociiy of wind
z	height above ground level
z_0	surface roughness length
A_{con}	surface contact area
A_s	swept area of blades
E_{wind}	wind energy
C	dynamic eqivalent load
C_a	basic dynamic axial load rating
L_n	dynamic equivalent
F_a, F_z	axial load on pitch bearing
M_x, M_y	Bending moments
Q	compressive load
P_{ea}	dynamic equivalent axial load
P	eqivalent load
P_{wind}	power of passing wind
Z	Number of rolling element rows
Z_r	Number of rolling elements per row
U^*	friction velocity

Units

k	kilo
M	Mega
m	meter
mm	millimeter
N	Newton
W	watt
Pa	Pascal
kg	kilogram
J	Joules

1 Introduction

The key factor for well-being and development of quality of life is adequate access to energy sources like electricity, gas, etc. Providing and gaining access to this proper energy in remote areas is still challenging in the modern world. Modern-day and past energy sources are dominated by fossil fuels (coal oil and gas). Extensive usage of these resources led to a rise in carbon dioxide and greenhouse gas levels, impacting the environment substantially.

The primary energy consumption of the world is increasing at a high rate every year. In the year 2018, world energy consumption grew at a rate of 2.9% of the previous year [8], which is double-fold compared to the average last decade increase rate of 1.5%. Renewables power contributed a 14.5% increase rate of consumption. Wind energy has accounted for 50% of the renewables in the last few years [8], followed by solar energy generation 24%, which has significantly increased production rate than previous years. An increase in demand for Renewable Energy provoked Wind Turbine Technology innovations, as wind energy notably one of its significant proportions.

Climate change, health consequences of air pollution, and decreasing carbon-based resources are concerns that produce and use alternative energy. Some alternatives for carbon-based resources are geothermal, biomass, hydroelectric, and wind energy. The evolution of wind technology in past years is outstanding. Extensive research and technological advancements paved the path to high capacity wind turbines, increased installations at reduced costs than before. From a 22 kW rated first ever installed three-bladed in 1982 [9] to a 10 MW rated offshore turbine [10] expected in 2020, wind turbines' capacity has increased consistently. The current cumulative installed wind power capacity is close to 651 GW, an increase of 10 per cent compared to 2018 [11].

Modern wind turbines standing typically at the height of at least 80 m, generate electricity when the wind speeds reach cut-in speeds 3-5 m/s. They produce useful energy over 90% of the time in a given year and are supplied to end-user through power grids and operators.

1.1 Classification of Wind Turbines

Wind turbines are classified based on the rotational axis orientation as Vertical and Horizontal axis wind turbines. Each has its advantages and disadvantages over the other one.

1.1.1 Vertical Axis Wind Turbine

Vertical axis wind turbine (VAWT) has its rotational axis perpendicular to the ground. Small wind farms hailing in residential areas prefer installing VAWT. These turbines are capable of utilizing wind energy coming from all directions, and they are even sometimes powered by the wind from top to bottom, which makes them more versatile in installing at sites with high turbulent wind conditions. Vertical axis wind turbines are further classified into three types, namely Darrieus Turbine, Giromill turbine, and Savonius rotor

- Darrieus turbine, named after French engineer Darrieus observed forces acting on

an airfoil profile when placed in an air stream. The Forces are of varying directions and intensity, making the profile rotate and generate the torque. However, this turbine bank on lift forces, is not self-starting; auxiliary motor primes the rotor initially as they require enormous initial wind power.

- Giromill turbine is a rotor consist of a vertical airfoils attached to vertical axis extracting energy on the same principle as the Darrieus turbine. However, the blade angles can be controlled throughout the rotation for maximizing lift.
- Savonius rotor is a drag driven type. This turbine relies on the principle of differential drag used in anemometers. The efforts exerted on the hollow body faces by air stream are of different intensity, making them generate torque. Savonius turbines cannot produce more power than the former two types of turbines, making it less efficient than Lift-driven turbines.

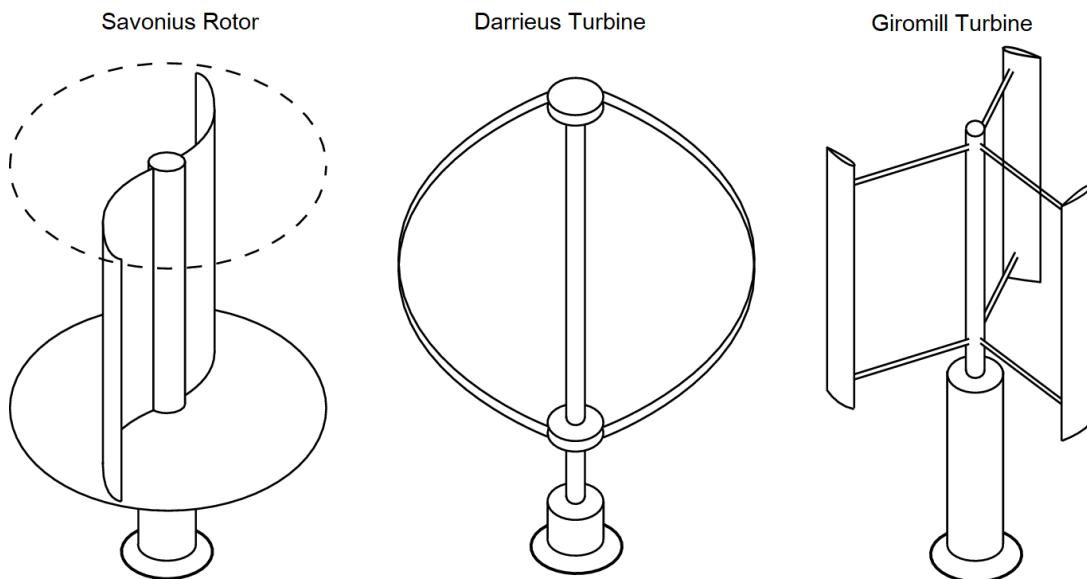


Figure 1: Types of Vertical axis wind turbine[1]

Vertical axis wind turbines are quieter compared to other types, their cost of manufacturing is low and can be transported easily to the installation site. They can be installed at shallow heights and need less ground area with capability of working even under turbulent conditions. The main drawback of VAWT is that the blades are very vulnerable to fatigue loads and less efficient compared to HWAT, making them less reliable than the Horizontal axis wind turbine (HWAT).

1.1.2 Horizontal Axis Wind Turbine

On the other hand, a Horizontal axis wind turbine (HAWT) is the one with a rotational axis horizontal and parallel to the ground. Even though they require a mechanism in orienting blades, most modern wind farms widely install HWAT's. Wind from single direction power HWAT, which makes these wind turbines unsuitable in places with turbulent winds. Though it is much more substantial and inefficient in rough conditions, it

is the preferred turbine type because of its position above ground and ability to produce more electricity from a given amount of wind favoring greater efficiencies.

The initial concept of the Horizontal Axis wind turbine is dated back to the 1930s in Yalta. An HWAT similar to the one today is built with a capacity of 100kW with a 32m tower height. Lift force acting on blades by wind flowing on turbine blades generate useful torque for power generation. These wind turbines are further classified into upwind and downwind type. An Upwind turbine has its rotor blades standing in front of the tower and nacelle. In contrast, the blades are positioned behind the nacelle and tower in case of a Downwind turbine.

Most modern wind turbines prefer employing a 3-rotor blades to 2-rotor blades concept, as designing dynamic properties for a 3-rotor blade turbine is convenient and secure than a 2-rotor blade turbine. Decreasing a rotor blade to 2 reduces the cost of manufacturing and weight mounted. Instead, this leads to an intricate designing with a teetering hub to overcome sudden shocks. Another disadvantage is that a 2-rotor blade requires high rotational speed to generate the same power output as a 3-rotor blade.

Higher power output ranging from 2-8 MW, greater efficiencies of up to 40-50%, matured research, and development in HWAT, making them more reliable and versatile operational wind speeds at extreme altitudes are the significant advantages of HAWT's over VAWT's. On the other hand, the sheer size of HAWT's makes it challenging to maintain and logistic problems while installing. Large ground area requirements, far installation sites, and environmental impacts are some other drawbacks of HAWT's.

2 State of Art

As commercial wind turbines are mostly horizontal axis wind turbines, making proper design considerations is crucial in availing most from the wind energy. Let us assume m (kg) mass of air with a velocity v_w (m/s) before passing through swept area of wind turbine blades A_s (m^2). The kinetic energy (J) of moving air is expressed[12] as

$$E_{wind} = \frac{1}{2}mv_w^2 \quad (1)$$

If the distance travelled by the moving mass of air in a short time interval dt is S (in m), the power (in W) associated with the wind passing through the swept area is calculated as

$$S = v_w dt \quad (2)$$

$$P_{wind} = \frac{E_{wind}}{dt} = \frac{1}{2} \frac{dm}{dt} v_w^2 \quad (3)$$

If we consider wind as a fluid there is change in both density and velocity of the air, which leads in change in mass of air. For this reason, *M.Reccab Ochieng and N.Frederick*[13] formulated using a factor of $\frac{2}{3}$ to calculate kinetic energy of air instead of a factor $\frac{1}{2}$. In this thesis mass of air is assumed constant and factor of $\frac{1}{2}$ is used. Mass flow rate $\frac{dm}{dt}$ at any instance is given by ρAv_w . Now the power expression (3) changes to

$$P_{wind} = \frac{1}{2} \rho A v_w^3 \quad (4)$$

From the above equation (4), it is evident that wind speed is a critical factor in wind turbine power. However, wind speeds at the hub are under the massive influence of height at which it is positioned since wind speed varies logarithmically with altitude[14]. According to Log law (equation(5)), wind speed increases logarithmically with an increase in height above the ground in the Prandtl layer. The gradient at which the wind speed increases is high initially and declines after a further height increase. For suppose if the given terrain has a surface roughness length of z_0 along with von Karman constant of k and a friction velocity of U^* , then the velocity profile $U(z)$ at any given height z from ground level is approximated as

$$U(z) = \frac{U^*}{k} \ln \left(\frac{z}{z_0} \right), z \geq z_0 \quad (5)$$

Despite enormous wind energy available, energy extracted by wind turbines is limited. Betz theory proposed by German physicist Albert Betz in 1919, explains that the maximum theoretical efficiency for a wind turbine is 59.3% [15], which means wind turbines can utilize at most 59.3% of the wind kinetic energy to spin the turbine. This limit is well known as the Betz limit. In practice, modern wind turbines have a maximum efficiency of 35-45%.

The right combination of the rotor diameter and hub height ratios at a specified wind park location enhances the turbine's power output. Turbine components like the tower, rotor

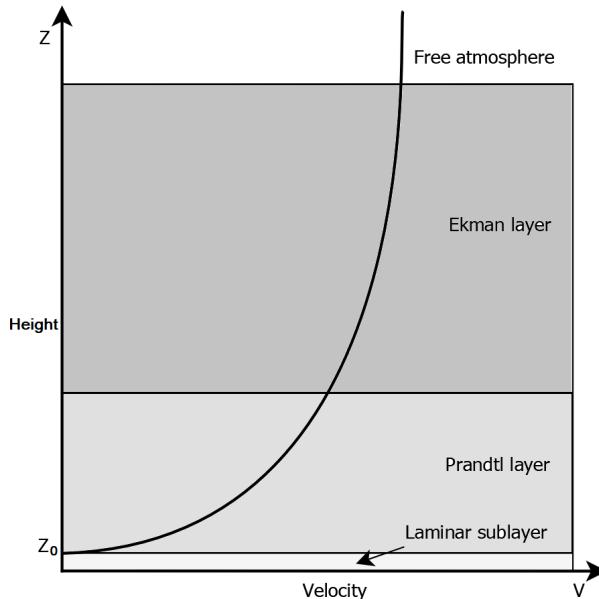


Figure 2: Variation of Windspeed with increase in height[2]

blades, and tower base must be designed carefully to provide enough strength alongside reducing effects like noises, vibrations, and structural deflections during turbulence and high wind speed conditions. The wind turbine's optimal functioning is hugely dependent on vital components like the rotor, gearbox, tower, yaw, and pitch systems. Below is a detailed description of the turbine components mentioned previously.

2.1 Tower

The tower is an essential component that affects the turbines's wind energy availability alongside carrying rotor and nacelle. Many tower designs are developed during the evolution of wind turbines, the terrain at the wind park location impacts the wind characteristics and behavior at any given time. The presence of mountains, hills, buildings, and forest terrain may change the wind conditions to turbulent, causing variable wind speeds at the ground level. For this reason, modern tower height varies as a function of wind park location, enabling adequate wind availability for blades to generate more electric energy. Tower height may vary depending on the wind park location, and increased tower height should practically make a trade-off between increased cost of manufacturing and power output.

A few of the Design considerations for an optimal tower are as follows [16]

- Tower mass can be reduced by manipulating diameter and thickness along the length, which decreases manufacturing cost.
- Fatigue strength and adequate stiffness of all components, including support structure, must be checked along with maximum deflection of blades and tower such that no blade makes contact with the tower in extreme load conditions
- Aerodynamic forces, inertial loading at the top of the tower as well as gravitational loads are to be considered in tower strength.

- Minimum tower height of 1.5 times rotor diameter is maintained in the design.

Some of the tower types developed for HAWT are explained below.

1. **Tubular Tower:** Steel tubular sections of length 20-30 m are bolted together with flanges on either side of the tubular section. The top and bottom-most flanges are connected to the nacelle and foundation, respectively. All the sections are of tapered shape, meaning that the tower's diameter decreases gradually along its length from bottom to top; this specifically strengthens and reduces the tower's mass.
Proven its advantages, this is the most used type in today's wind market. This type is more robust as it is made of rolled steel plates and protected parts as closed parts. Cost may increase further if any anti-rust are coatings required to sustain in unstable humid conditions.
2. **Concrete Tower:** Similar to tubular tower sections, precast concrete segments are assembled at the site. Precasting reduces the additional costs of production at the installation site. Concrete towers add extra strength compared to other tower types, increasing the chances of high capacity turbine installations. Tower sections are of length no longer than 4-5 m, as any further increase in sectional length is problematic for transportation.
3. **Lattice Tower:** Lattice towers are the most preferred ones previously, mainly suitable for turbines with less than 1 MW power rating. The cost of manufacturing is almost half of the tubular towers for the same stiffness and strength. Tower buckling is overcome by including additional strut elements in the tower. The main disadvantage of this type is aesthetic issues, namely visual appearance.

2.2 Rotor

The rotor is the rotating part of a turbine converting wind energy into torque. It comprises two main components, the rotor blades and the hub. The interaction between the rotor and the incoming wind is crucial for the turbine's energy generation. Hub is the component that connects the rotor blades to the main shaft, which is connected to the drivetrain of the wind turbine. The Hub should also withstand all the loads acting on blades; thus, they are commonly made of welded steel or cast iron [17].

There are many hub design concepts in use; each concept has a specific advantage over the other design. The main designs include teetering, rigid, and hinged. A rigid rotor hub is the one design where all the possible degrees of motion of the blades are removed. However, rigid rotors with varying blade pitch are not included in this concept. A teetering hub is a design in which the hub has a few degrees of freedom to move forward and backward, thus can withstand the sudden shock and dynamic loads acting on the blades. Rotor blades are hinged to the rotor hub in the hinged rotor concept. The hinge mechanism makes the design more complicated, as an additional mechanism needs to be provided to prevent the hinged blades from flopping during low wind conditions [18].

The surface area swept by the rotor blades during their rotation is called the swept area. The power output of a wind turbine is directly related to the area swept by the blades,

as expressed in the power equation (3). Few aspects that impact the blades' performance are blade aerodynamics, tip-speed ratio, angle of attack, twist, and blade pitch angles. The choice of the blade material mainly depends on the design of the blade opted. Some of the often used blade materials are wood laminates, glass-reinforced plastic, carbon fiber-reinforced plastics [17].

2.3 Pitch System

Pitch bearing, often called slewing bearing or blade bearing, is a principle component in changing the blade angle of attack and maintaining lift acting. It also dissipates the loads induced by stresses acting on blades to the hub of the turbine via raceways and bolted joints. Pitch bearings are bolted on one side to the blade root and on the other side to the rotor hub. Typical pitch bearing includes the following individual components

- Inner and Outer raceways
- Bolts to fastening hub and blades
- Cages or separators
- Integral seals

Irregular rotation of the blade and high magnitude moments on bearing complicates their design. Unlike conventional bearings, pitch bearings are oscillatory in motion, which means they do not make full turns; instead, they rotate back and forth over an angle. They are generally switched via a hydraulic or electric motor and gearing connection on one of the raceways. Collective pitch is a method performing simultaneous pitching of all blades, whereas in individual pitch control (IPC) each blade is individually pitched. Theoretical analysis suggests that IPC reduces loads on each blade better than Collective pitch control [19]. Any case as a vital component, pitch bearings must ensure adjusting blades under any harsh condition expecting a service life of 20 years for each pitch bearing once mounted.

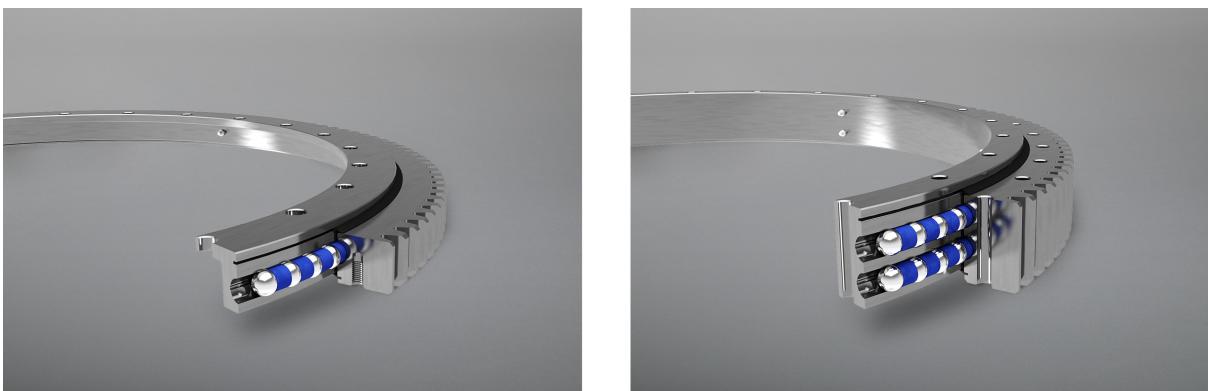


Figure 3: Single and double row slewing bearing with polyamide spacers [3]

Type and number of bearing elements classify the kind of pitch bearings in use, single or double row four-point contact spherical ball bearings with internal or external gearing

are the most used nowadays. Some wind turbines employ pitch bearings with single or double row roller elements. At an increased production cost, a double row has increased fatigue life along with lower ball loads and hertzian stresses than a single row four-point contact bearing [4]. The bearing material and amount of lubrication depend on factors like Hertz contact stresses, modes of failures, providing different manufacturers' different possibilities. Typical core hardness values of inner and outer raceways should fall in the hardness range of 250 to 300 Brinell hardness (HB), providing them adequate core yield and fatigue strength.

2.4 Pitch Bearing Design

Designing pitch bearings require a thorough knowledge of magnitudes and varying effects of loads acting. Pitch bearings are subjected to various loading's during their regular operation; some include blade weights, the centrifugal force of rotating blade, aerodynamics forces namely thrust. Below is a detailed explanation of the types of loads and their effect on the pitch system

- **Aerodynamic Loads:** These loads contribute significantly to total loading in the pitch system and induce aerodynamic moments about radial and pitch axis. Calculating the magnitude of these loads is challenging, as they vary along the length of the blade as a function of blade profile
- **Blade Weight:** Blade mass distributed along the blade length cause varying force and moment about the pitch axis due to gravity during its rotational and stationary position
- **Inertial Loads:** As we know, the rotating body experiences centrifugal forces along its rotating axis, centrifugal forces on the blade act along the pitch axis, inducing forces and moments in pitch bearing

Bearings employed for pitch bearings are Thrust type, which makes them support for axial loads as well. The loads acting in pitch bearings are categorized into three main types, namely axial, radial, and bending moments. A dynamic equivalent load is required in life estimation for a bearing subjected to the three loads. Furthermore, a bearing subjected to time-varying loads and speeds needs an equivalent load for analysis. At any instance, more than one-half of the total rolling elements are in loading condition; however, load distribution in them differs considerably over the bearing circumference. According to the National Renewable Energy Laboratory (NREL), the following design considerations [4] should be addressed for a proper designing of pitch bearing system:

- Bearing fatigue life (rolling contact fatigue)
- Bearing static capacity
- Adequate case depth and core hardness
- Friction torque
- Adequate lubrication

2.4.1 Rolling Element-Raceway contact

Assume two rigid solid elastic bodies with radii r_1 and r_2 making a point or line contact with each other under no load. If a load is applied on them, the solid bodies undergo elastic deformation resulting in an increased contact area. This theory is proposed by Heinrich Hertz in the year 1882, commonly known as the Hertzian contact theory. The validity of Hertzian contact theory is solely based on the following assumptions:

- The two bodies are homogeneous, elastic and do not make contact area exceeding their dimensions
- Compressive forces developed in bodies act in a direction normal to the contact point on the surface
- The contact surfaces are frictionless
- The strains developed are small and within the elastic limit

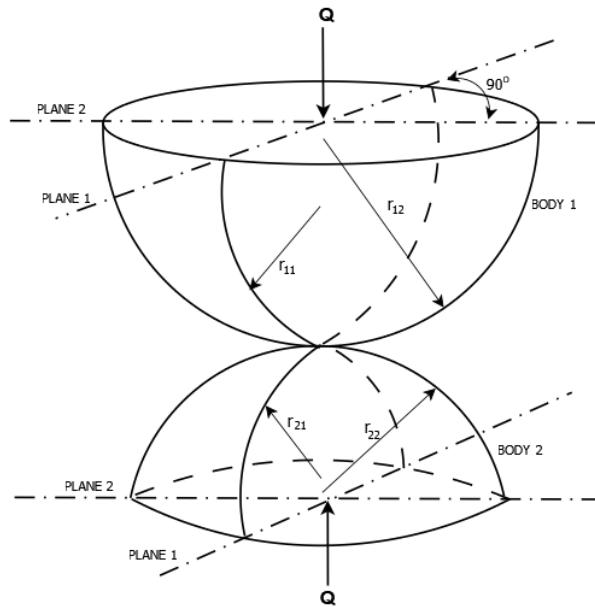


Figure 4: Two elastic elastic bodies under compressive load Q [4]

Let Plane 1 and Plane 2 be orthogonal planes in two solids respectively such that principle radius of curvatures in two orthogonal planes are r_{11}, r_{12}, r_{21} and r_{22} . The corresponding curvatures in the orthogonal planes are $\rho_{11}, \rho_{12}, \rho_{21}$ and ρ_{22} then curvature sum, curvature difference of the bodies in the orthogonal planes are given by $\sum \rho$ and $F(\rho)$ [4]. The two bodies can be considered as two ball elements in contact or a ball- raceway contact in a ball bearing.

$$\sum \rho = \rho_{11} + \rho_{12} + \rho_{21} + \rho_{22} \quad (6)$$

$$F(\rho) = \frac{\rho_{11} - \rho_{12} + \rho_{21} - \rho_{22}}{\sum \rho} \quad (7)$$

where:

$$\begin{aligned}\rho_{11} &= \rho_{12} = 2/d_{ball} \\ \rho_{12} &= -1/r_{raceway} \\ \rho_{22} &= \pm 1/\left(\frac{a_0}{\cos \alpha} \mp r_{raceway}\right) \\ \alpha &= \text{contact angle} \\ a_0 &= \text{mean distance between ball and bearing centres}\end{aligned}$$

In case of ball bearings, the force acting on ball element create an elliptical contact and a rectangular contact in roller bearings at their interface with raceway. The upper and lower signs in the curvature ρ_{22} refers to the inner and outer raceways. The semi-major a and minor axis b dimensions of contact ellipse formed are calculated using the formula below

$$a = 0.0236 a^* \left(\frac{Q}{\Sigma \rho} \right)^{1/3} \quad (8)$$

$$b = 0.0236 b^* \left(\frac{Q}{\Sigma \rho} \right)^{1/3} \quad (9)$$

where:

a^* and b^* are variables dependent on $F(\rho)$

2.4.2 Bearing Fatigue Life

Bearing fatigue life is defined as number of revolutions made by bearing raceways relative to another before any first evidence of fatigue failure appears on raceways or rolling elements. Typical bearing life is represented as L_n and measured in millions of revolutions, but as pitch bearings are cyclic oscillatory type; pitch the bearing life is measured in millions of oscillations. Many efforts are put into developing rolling bearing standards; most engineering rolling bearings calculations are based on ISO 281:2007 standard. As pitch bearings oscillate about a specific angle the above mentioned bearing standard do not produce optimal results that fit well for wind turbines. However, the modified ISO 281:2007 standard with certain assumptions is valid for oscillating bearings. According to ISO 281, assuming dynamic load rating as C , equivalent load P , and the roller geometry dependent exponent p , the bearing life L_{10} of rolling bearings with 90% reliability is calculated in the equation below.

$$L_{10} = \left(\frac{C}{P} \right)^p \quad (10)$$

Technical report by NREL ‘*Wind Turbine Design guidelines: Yaw and Pitch Rolling Bearing Life*’ obtain bearing load characteristics based on ANSI [9] and ISO 281[4]. Following is a detailed approach by NREL for fatigue life calculation of spherical pitch bearings. Using load rating standards, the basic dynamic axial load rating C_a is formulated as below.

$$C_a = f_{cm} (i \cos \alpha)^{0.7} Z_r^{2/3} D^{1.8} \tan \alpha, \quad \text{when } D \leq 25 \text{ mm} \quad (11)$$

$$C_a = 3.647 f_{cm} (i \cos \alpha)^{0.7} Z_r^{2/3} D^{1.4} \tan \alpha, \quad \text{when } D \leq 25 \text{ mm} \quad (12)$$

Where f_{cm} is the conformity factor, α is bearing contact angle, i is the number of rows, Z_r number of balls per row and, D is ball diameter. A uniformly distributed load that replaces the actual load conditions such that the bearing has the same life can be summoned as Dynamic equivalent load. Thrust bearings are under complex axial (F_a), radial load (F_r) and bending moments (M), and they can be replaced by a single dynamic equivalent axial load P_{ea} . NREL estimated the dynamic equivalent axial load as follows.

$$P_{ea} = 0.75 F_r + F_a + \frac{2M}{d_m} \quad (13)$$

Equation(11),(12) and (14) are valid only in estimating thrust bearing's life, whereas pitch bearings do cyclic oscillations under varying (non-constant) load and speeds. Assuming the variable load conditions of pitch bearings in a duty cycle can be discretized into set of n instances, in each instance k , bearing is subject to dynamic equivalent load P_{eak} with oscillation speed of N_k , oscillation amplitude θ_k , and an operating duration t_k . The dynamic equivalent axial load P_{ea} and basic dynamic axial load rating $C_{a,osc}$ for an oscillating pitch bearing over the duty cycle are calculated as follows.

$$P_{ea} = \left(\frac{\sum_{k=1}^n P_{eak}^p N_k t_k \theta_k^x}{\sum_{k=1}^n N_k t_k \theta_k^x} \right)^{1/p} \quad (14)$$

$$C_{a,osc} = C_a \left(\frac{180^\circ}{\theta} \right)^{1/p} \quad (15)$$

where:

$p = 3$ for ball bearings

$p = 10/3$ for roller bearings

θ = amplitude of oscillation

By considering dynamic equivalent axial load P_{ea} and basic dynamic axial load rating $C_{a,osc}$ from the equations (14) and (15) respectively, a 90% reliability fatigue life for an oscillating pitch ball bearing can be estimated as

$$L_{10} = \left(\frac{C_{a,osc}}{P_{ea}} \right)^3, \quad L_{10} : \text{millions of oscillations} \quad (16)$$

2.4.3 Bearing Static Capacity

For a pitch bearing subjected to axial (F_a), radial load (F_r) and bending moments (M), US Renewable Energy Laboratory proposed the following empirical formula(17) to estimate maximum load Q_{max} acting in it under static condition.

$$Q_{max} = \left(\frac{2F_r}{Z \cos \alpha_o} + \frac{F_a}{Z \sin \alpha_o} + \frac{4M}{DZ \sin \alpha_o} \right) \quad (17)$$

$$S_{max} = \frac{t Q_{max}}{A_{con}} \quad (18)$$

where:

- α_o : the contact angle
- Z : number of rolling element rows
- t : Bearing element dependent factor
- A_{con} : surface contact area (Ellipse or Rectangle)

As discussed earlier in section (2.4.1) according to Hertz contact theory, load applied on two solid elastic bodies of different radii which are in contact creates a small contact area causing elastic deformation on the surfaces no matter how low the applied load is. The stress normal to the roller element-raceway contact developed due to this load is called Hertz stress. Hertz contact stress is always compressive, meaning contact surfaces do not adhere together, creating no tensile stresses. Hertz contact stress can be calculated as the ratio of applied load to the contact area between contact surfaces. Maximum hertz contact stress S_{max} developed in a thrust bearing is calculated by equation (18). Contact area A_{con} depends on the type of rolling element in bearing; more about contact area is explained in section(2.4.1).

According to ISO 76 standard, a bearing's static load capacity is the amount of load bearing can withstand before undergoing a permanent deformation of 0.01% of rolling element diameter at the center of most heavily loaded rolling element-raceway contact. Stress experienced under load C_o , is experimentally found out to be equal to maximum Hertz contact stress.

- 4200 MPa for all ball bearings
- 4000 MPa for all roller bearings

Bearing experiences static loads when it is still and non-rotating or under shocks. Bearings designed without considering static load capacity fail prematurely by undergoing permanent deformation and compromising its function. Since static load capacity is measured and evaluated in a non-moving state of bearing, it is not recommended for estimating bearing life. However, to protect the rolling bearing from sudden static loads, it is designed considering certain safety margin called static load factor (SF). The static load factor is ratio of the maximum allowable load in the rolling element to the actual load. As the maximum permissible load in a bearing is associated with maximum hertz stress; the static load factor for spherical bearings can be approximated as

$$SF = \left(\frac{4200}{S_{max}} \right)^3 \quad (19)$$

2.5 Pitch Bearing Failure

Wind turbines usually work in complex and changeable environments with condensed water formed by temperature change, salt in the sea breeze, acid rain, corrosive gas, and lubricant acidizing, which cause corrosion and numerous possible damage mechanism in pitch bearings. Some of the significant pitch bearing failure mechanisms are as follows.

1. **Contact Ellipse Truncation:** In pitch bearings with ball rolling elements, the surface contact between the ball element and the raceways is an ellipse. The contact angle between given by the manufacturer is called mounted contact angle. But under loading conditions, the contact between the ball elements and raceways increases; this increased angle is called free contact angle. When the loads are extreme and the free contact angle is correspondingly high, high contact angles can result in an contact ellipse which is truncated by the raceway edge, leading to high stresses at the raceway edge. Some of the ways to prevent ellipse truncation are increased mounted contact angle, use of stiffener plates[20]. Increasing mounted contact angle helps rolling elements to address more stresses than before.

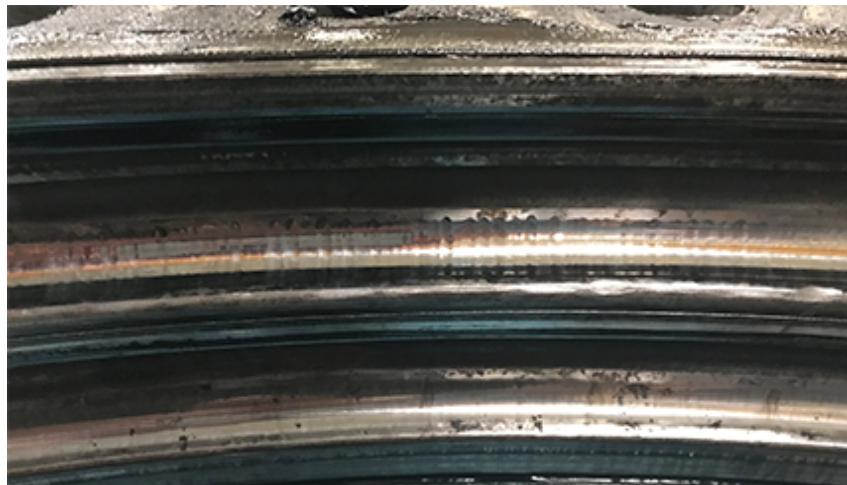


Figure 5: Fretting wear on pitch bearing raceway[5]

2. **Fretting Wear:** In some operating conditions, the lubrication between the contact surfaces is squeezed out, resulting in metal to metal contact. When two metal surfaces in contact undergo sliding movement, mechanical damage on the surfaces occurs called fretting wear. If the sliding between the two surfaces is repetitive, the wear is transformed into hollow spots and dents. This type of failure commonly occurs at the raceway and rolling element interface in pitch bearings. Possible causes for the fretting wear are small-amplitude vibrations and oscillations, inadequate lubrication. Measures to prevent this failure are preloading the components to prevent vibrations, adequate lubrication. Fretting wear is escalated into fretting corrosion when the non-lubricated surfaces adhere to one other breaking the natural oxide layer and forming welds. The figure(5) shows fretting wear occurred on raceway of a pitch bearing.
3. **Bearing Ring Cracks:** Bearing rings have holes to employ the grease fittings or handling the features like bolts. The presence of holes in any structure acts as stress risers, increasing the bearing ring's nominal stress. Cracks are initiated in these areas, failing the bearings. Crack initiation commonly starts at thread holes and can propagated through the whole ring cross-section.



Figure 6: Crack initiation at bolt thread holes[5]

2.6 Bridging FEM with Machine Learning

A physical system's life expectancy can be enhanced by optimizing its components' performance under real-time loads. For example, in wind turbines a shape optimized stiffening plate is used to enhance the pitch bearing's life expectancy. A detailed analysis of its functioning can optimize the performance of a component. Finite element and volume analysis are some of the modern methods employed in analyzing any system or its critical components. Modeling and analyzing a physical system with FEM or FVM includes extensive use of software tools that implement them.

Any change in design, size, functioning, loading of a component or system requires additional time for solving the system again, either a week or months, depending on system complexity. Besides, varying the system parameters also increases the duration of the analysis. Hence it is evident that this approach of estimating system and component characteristics are at the expense of increased time and cost. However, modern advancements made possibility in computational tools like Machine learning, Artificial Neural Networks, which save computational time and cost if appropriately implemented.

Incorporating an ML or ANN algorithm in analysis with the existing Finite element methods can be useful if an appropriate approach for blending the methods is designed. Several researchers from different fields have tested the effectiveness of this methodology.

- *R Lostado et al.* in the research “*Combining regression trees and the finite element method to define stress models of highly non-linear mechanical systems [21]*”, combined regression trees with Finite Element Methods to predict the behavior of a complex non-linear Vehicle Axle system. Initially, non-linear FEM maps generated are used as input to the decision tree regression model, predicting the vehicle axle behavior. It is observed that the decision tree model predicted the contact pressures in the system with high accuracy even under operating conditions, which are not considered in the FEM simulation
- In the research “*An Integrated Machine Learning and Finite Element Analysis Framework, applied to Composite Substructures including Damage [22]*”, *T.H.E Guikers* developed an ANN and FEM framework to establish a flexible approach

to substructure homogenization. That is, this method allows a homogenous substructuring of the given model regardless of the domain. Firstly data is generated through FEM simulations on the chosen substructure by varying all the dependent parameters (e.g., applied loads). The response of varying parameters is used as input data in training the ANN model, which predicts the chosen substructure's mechanical behavior as a function of independent parameters. After that, the trained ANN model is coupled into the FEM software package Abaqus/CAE as a user material subroutine(UMAT), which is then used for substructure homogenization in FEM calculations. The coupled ANN and FEM approach has proven its higher accuracy and efficiency capabilities alongside a decrease in labour and input model preparation when complex models are tested.

- An ANN–FEM coupled model to predict maximum pressure in journal bearings is developed by *Sunil Kumar, Dr. Vijay Kumar, and Dr. Anoop Kumar Singh* in the research article “*Prediction of maximum pressure of journal bearing using ANN with multiple input parameters [23]*”. FEM analysis is performed on a lubricated hybrid journal bearing, and the results are used as training and testing set for a feed-forward propagation ANN model. The external loads and rotational speeds are used as features (input parameters) for performance predictions within and outside the prescribed range of training data. The ANN model’s predictions are in close agreement with FEM analysis as the prediction error is around 0.5%. It is concluded that the coupled model saves much time as modeling and parameter tuning of FEM analysis for each load case is avoided.

This integrated approach of using Machine Learning with Numerical methods is not only limited to Mechanical or Electrical systems but also adapted in biomechanics and medical sciences to study human body parts under disease or ill conditions as discussed below

- In the study “*Bridging Finite Element and Machine Learning Modeling: Stress Prediction of Arterial Walls in Atherosclerosis [24]*”, *Ali Madani et al.* generated a database of 12,172 2D-finite element parametric calculations for various arterial geometries. The hyperelastic behavior of the arterial wall is extracted in the finite element calculations. This database is used to train Neural network models to predict the von Mises stress, which indicates the risk of plaque rupture (in case of atherosclerosis). Trained deep learning models can accurately predict the max von Mises stress within 9.86% error on a held-out test set and concluded that the deep learning models could potentially be utilized to develop insights on the nature of the disease itself.

The above research shows that bridging machine learning and deep learning models into numerical methods with an appropriate technique provides control and edge on the cost function and analysis time.

2.6.1 Machine Learning

Machine learning is a science of programming an inductive model that attains learning ability from different data kinds without any supervision. Learning implies detecting

underlying patterns, relationships from the given limited input data. Machine learning models are vastly used in various life problems like Healthcare and Medical diagnosis, Banking, Email Intelligence, Evaluation and, Assessment. Because of present computing technologies, it is possible to automatically produce highly accurate models that can analyze bigger and complex data at higher speeds.

Machine learning can be classified as supervised and unsupervised learning. Supervised learning predicts output variables using the labeled input data, whereas unsupervised learning draws inferences without any labeled input. The following flowchart depicts the crucial steps in Machine learning workflow on any given data.

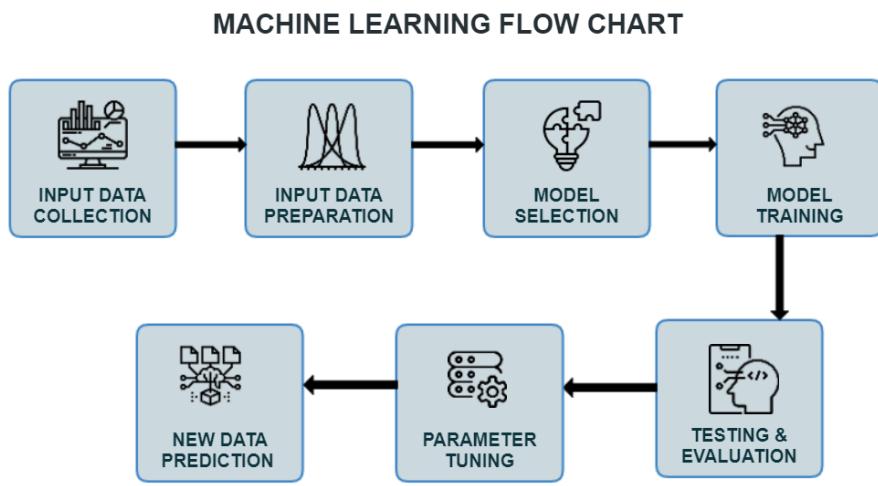


Figure 7: Machine Learning workflow

Data preparation is the initial step in machine learning. In this step, the raw data is normalized, scaled, or de-duplicated, removing error or biases present in the given data. This step provides an idea of patterns and outliers in the data. **Choosing a right model** is a key step in ensuring that the required goal is achieved. There are many models in use that serve multiple purposes ; commonly used machine learning algorithms are linear and logistic regression, Random Forest, Support Vector Machines (SVM), Principal Component Analysis (PCA), and Neural Networks.

Model training, a portion of previously prepared data called Training data set, is passed to train the model in this step. The prediction capability of the model is incrementally improved over each cycle. The model achieves this performance by updating the weights and biases of it by itself. In supervised learning, the model is trained over the labeled training data, whereas unsupervised learning model draws inferences by itself. To check how well the Machine Learning model is trained on the training data error called **Training error** is analyzed. The initial training data is again passed to the model to predict the output. The training error is calculated by comparing actual output with the predicted output of the training data. **Testing Model** is the step performed after the model is trained on the given data. The performance of the trained model is evaluated by testing it on an unused Test data set. The error measured on the testing data is called as **Validation error**.

Once the model has been evaluated, input **Parameter tuning** has to be performed. Trial and error techniques such as Increasing input datasets and varying numbers of training

cycles are conducted to acquire more accurate training and prediction models. The model is ready to predict if the above process is completed. **Prediction** capabilities of machine learning extend into various fields like medicine, image recognition, engineering, and astronomy.

2.6.2 Linear Regression

Regression is a statistical method of establishing relationships between two or more variables. It must be identified based on the logical content considerations, which are dependent and independent variables. A regression function maps features or variables to other variables sufficiently based on the mathematical dependence between them. The dependent features Y are dependent variables or outputs; independent features X are called independent variables or inputs or predictors. Regression problems usually have one or more continuous and unbounded dependent variables. The inputs, however, can be continuous, discrete, or even categorical data. The regression models used in this research are from the python **scikit-learn** [25] library described in the sections below.

Linear regression is one of the essential, simplest, and widely used regression techniques across many research fields. This technique establishes a linear relationship between two or more independent and dependent variables. If X is a vector of independent variables and a dependent variable Y , the linear relationship between Y and X is given by the regression equation as below.

$$Y = \beta_0 + \beta_1 x_1 + \cdots + \beta_r x_r + \varepsilon \quad (20)$$

where:

- $X = (x_1, x_2, x_3, \dots, x_n)$ is independent vector
- $(\beta_1, \beta_2, \beta_3, \dots, \beta_n)$ are regression coefficients
- ε is the error of the regression function

The linear regressor during its training, calculates the regression weights $b_0, b_1, b_2, \dots, b_r$ such that dependencies of output variable on the input variables are sufficiently defined. The linear regression function $f(X)$ is given as $f(X) = b_0 + b_1 x_1 + \cdots + b_r x_r$. Suppose the input dataset has 'n' observations. In that case, the regressor estimates the predicted response $f(X_i)$ for each observation X_i such that the difference between predicted and actual response $f(X_i) - Y_i$ called residual is minimal. Best residuals are obtained by minimizing the sum of squared residuals (SSR) of all the observations[25].

2.6.3 Polynomial Regression

In general, polynomial regression is an extension of linear regression; this can be achieved by adding the linear model's polynomial features. For suppose a linear regressor model with two independent variables x_1, x_2 is given by equation (21)

$$Y = w_0 + w_1 x_1 + w_2 x_2 \quad (21)$$

Now the above linear model can be converted into a second-order polynomial model with two independent variables by adding appropriate coefficients as

$$Y = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1 x_2 + w_4 x_1^2 + w_5 x_2^2 \quad (22)$$

The above equation is still a linear model if the variables $x_1, x_2, x_1x_2, x_1^2, x_2^2$ are considered as variables z_1, z_2, z_3, z_4, z_5 respectively. Now the linear model with new variables is given by the equation (23)

$$Y = w_0 + w_1z_1 + w_2z_2 + w_3z_3 + w_4z_4 + w_5z_5 \quad (23)$$

A polynomial regression model is transformed into a linear regression in machine learning and is trained on nonlinear functions. It is observed that this approach generates a faster and more efficient model.

2.6.4 Random Forest Regressor

Random forest regression is a supervised learning technique of approximating data by boot-strap aggregation or bagging methods, dividing the dataset into different sample sets and predicting the outputs. As the name explains, a random forest regressor is a collection of multiple regression trees. It uses multiple regression trees' output in determining the final output rather than relying on individual regression trees. A decision tree is a non-parametric model capable of performing both classification and regression. It is called a decision tree because the model completely works by making decisions in a stepwise manner [26]. The steps where decision making is performed are called nodes.

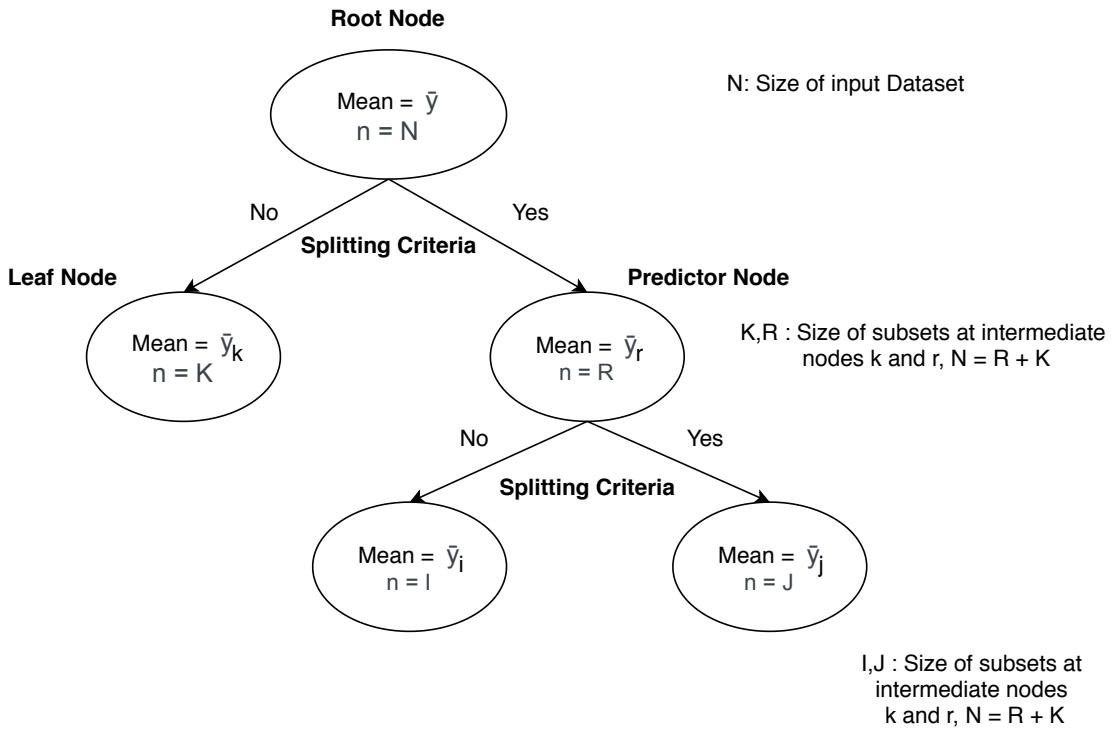


Figure 8: Decision tree data splitting at nodes

Regression decision tree or regression tree is a stepwise splitting model made of nodes where the splitting of the dataset into smaller subsets is recursively performed to predict the target value. The splitting of the dataset at nodes in regression trees is usually based on the Least Squares method [26]. The first node where the splitting starts is called a root node. Using the least-squares method, the dataset or subset at a node is split as

follows: Firstly, all the ‘n’ independent values in the dataset or subset are sorted, (n-1) possible ways to split the dataset into two are examined. For each of n-1 possible splits, the sum of squares of dependent variable about the mean of the dependent variable also called sum of squared error of the dataset is calculated.

The split that yields the least sum of squares is chosen as the criteria for splitting at that particular node. Splitting each node into two children nodes is performed until the child node does not have less data size than the defined threshold size. If the child node has dataset size of less than threshold it is called a *leaf* node. Once the *leaf* node is reached the dependent values for the whole independent values subset at that node are replaced by a mean dependent value. All the nodes where the dataset split is performed are called intermediate, or predictor or decision nodes and the nodes which cannot be further subdivided are leaf nodes. If the given dataset has multiple features, then the splitting at the nodes is performed based on the features one by one.

Suppose $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^N$ be the dataset, $x^{(i)} \in X$ and $y^{(i)} \in Y$, the main aim of the regression tree to construct a function $f : X \rightarrow Y$ such that the error $\sum_i |f(x^{(i)}) - y^{(i)}|^2$ is minimum. This is achieved by splitting the dataset into fewer subset in a step-wise manner as mentioned above and finding the output. If k is intermediate node with subset K , the mean $\bar{y}(k)$ of dependent variable and the sum of squared error $r(k)$ are given by equation[26] below

$$\bar{y}(k) = \frac{1}{N(k)} \sum_{x^{(i)} \in k} y^{(i)}, \quad r(k) = \frac{1}{N(k)} \sum_{x^{(i)} \in k} (y^{(i)} - \bar{y}(k))^2 \quad (24)$$

The initial dataset is split into multiple subsets in a random forest regressor and fed to multiple regression trees as input. The most common output value of the regression trees is considered a random forest regressor’s output. If there are multiple outputs to predict then a multiple output regression model is used. In multi-output random forest regression, each output variable is treated as a separate regression problem.

2.6.5 MLP Regressor

Multi-Layer Perceptron or MLP regressor is a supervised machine learning regressor (Neural network model). Unlike other regression models, MLP regressor consists of three layers: an input layer, output and a hidden layer. The number of hidden layers are variable and are defined by the user. Each layer consists of a certain number of nodes. The input variables or features are called input nodes, and the output variables are represented as output nodes, whereas the nodes in the hidden layer are called neurons. MLP regressor constructs an approximation function $f : X^m \rightarrow Y^o$ by training on a dataset initially, here $X = x_1, x_2, \dots, x_m$ are the features or independent variables or input variables where as $Y = y_1, y_2, \dots, y_o$ are the output variables or dependent variables. Suppose the regressor has a single hidden layer with ‘n’ neurons. Each neuron in the hidden layer transforms the values from the previous layer with a weighted linear summation $w_{1,i}x_1 + w_{2,i}x_2 + \dots + w_{m,i}x_m$, where ‘i’ being index of the neuron. Activation

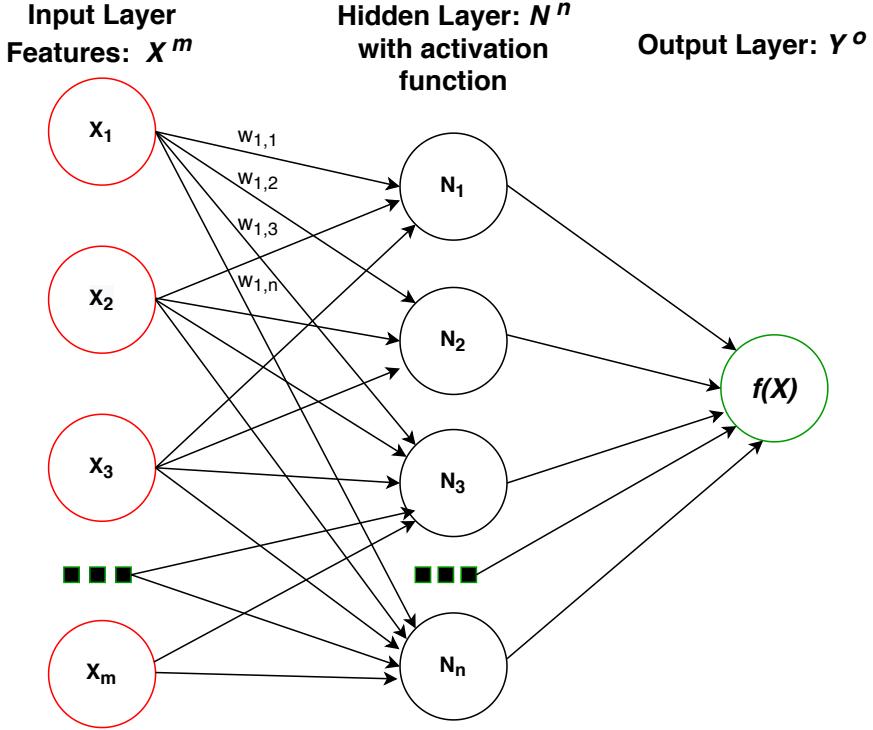


Figure 9: MLP regressor with one hidden layer

function decides, whether a weighted sum value at a neuron N_i should be further activated (passed) to the next layers or not by adding bias with it. The output layer receives the values from the last hidden layer and transforms them into output values.

There are many activation functions in the neural networks; however, MLP regressor mostly uses nonlinear functions like sigmoid or logistic, hyperbolic tangent, Rectified linear unit [25]. As a result, the hidden layers are nonlinear, making the regressor model to predict nonlinear relationships between inputs and outputs. Suppose Y^i and $f(X)^i$ are the actual and initial predicted data of ‘ i_{th} ’ output variable during the MLP regressor training. Each neuron in the hidden layer has its own set of weights; once the initial set of weights are generated, the regressor adjusts weights so that the error $E = \frac{1}{2} \sum_{i=1}^p \|f(X)^i - Y^i\|^2$ is minimized.

The weight’s adjustment in MLP regressor during training is performed using the back-propagation method. Because E is calculated by the extended network exclusively through composition of the node functions, it is a continuous and differentiable function of the l weights w_1, w_2, \dots, w_ℓ in the network. We can thus minimize E by using an iterative process of gradient descent, for which we need to calculate the gradient

$$\nabla E = \left(\frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, \dots, \frac{\partial E}{\partial w_\ell} \right) \quad (25)$$

By analyzing the effect of individual gradients $\frac{\partial E}{\partial w_\ell}$ on the error gradient ∇E , the weights of particular output are adjusted to attain minimum of the error function, $\nabla E = 0$.

3 Problem statement

Pitch bearing experiences thousands of different loads during its lifetime. Static load capacity and dynamic equivalent loads are essential in designing a pitch bearing, mentioned in section (2.4). Dynamic equivalent loads and static load capacities can be derived from design loads using powerful tools like MATLAB and Microsoft Excel, which can further be used in designing a pitch bearing through fatigue analysis. Through this method of transforming the design loads into dynamic equivalent loads, the time for designing pitch bearing is highly reduced as only algebraic equations are used for necessary calculations.

As mentioned earlier service life of pitch bearings can be increased by including a stiffening plate. A stiffening plate is usually positioned on the blade side of the pitch bearing. It prevents ovalization, ellipse truncation [20] in the bearing and its rolling elements by local stiffening of the bearing rings. However, the methods mentioned above of pitch bearing designing is not suitable for designing a stiffening plate, as identifying the forces causing ellipse truncation directly from time series loads is not possible. Designing a stiffening plate includes its shape optimization over all the critical loads acting on pitch bearings over its design.

A study conducted by *Amin Loriemi et al.*[7], RWTH Aachen suggests a shape optimized stiffening plate reduces the magnitude of loads acting in pitch bearing rolling elements. A substructured pitch bearing and stiffening plate FEM model developed by them [7] requires a maximum of 1 to 2 minutes to calculate the pitch bearing load distribution for a single load case. Since all the loads acting on pitch bearing are not responsible for lifetime reduction, the loads causing it needs to be classified or separated from the time series design loads of pitch bearing, which can be used for stiffening plate shape optimization. Classifying these critical or crucial loads from only FEM calculations of pitch bearings is not cost-effective, and requires enormous time.

Once these loads are acquired, shape optimization becomes easy, as the number of loads in studying the pitch bearing characteristics are finite. For this a method that can classify and predict the pitch bearing load characteristics over the entire design loads based on a few FEM calculations is preferred. This thesis aims to develop such a method to reduce the pitch bearing design loads to a smaller load collective for optimizing stiffening plates. The design loads in this thesis are opted from the research of *Amin Loriemi et al.*[7], RWTH Aachen, which are approximately 3.6 million load combinations from multibody simulations.

The modern approach to bridge the FEM model with the Machine learning discussed in the section (2.6) is opted into this research, as it has proven advantage of accuracy and a decrease in the computation costs. A machine learning model is developed such that it predicts the pitch bearing load characteristics based on few FEM calculations. The main objective is to develop a methodology to reduce design loads acting in pitch bearings to a smaller load collective using a combined Finite element and Machine learning approach for stiffening plates optimization.

4 Method

Firstly, the pitch bearing design loads (time continuous) obtained from multibody system simulations are discretized into fewer load cases. The load discretization is performed using a method called aggregation. The bearing behavior for discretized loads collective is calculated using the finite element method model of the pitch bearing. The bearing response obtained before are further used as training data for a machine-learning algorithm, which further approximates the bearing behavior for the entire time-continuous design loads. The approximated pitch bearing response is analyzed to obtain the critical loads from pitch bearing's design loads. The flowchart (10) depicts the workflow carried in obtaining the critical loads.

Workflow of Critical Load Collective Identification from Design Loads

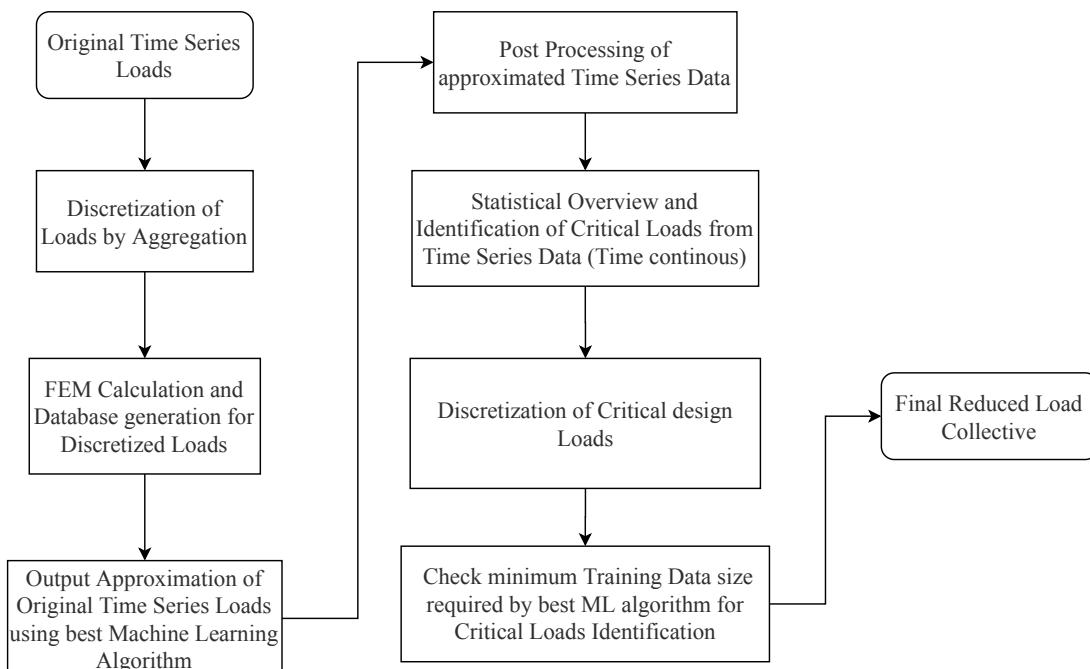


Figure 10: Flow chart of the Methodology

Several machine learning algorithms are used in approximating the pitch bearing behavior for the design loads. The best performing ML algorithm output is chosen as the bearing response for the design loads.

4.1 Finite Element Calculation

FEM model of pitch bearing arrangement developed by Mr.Amin Loremi M.Sc at Chair for Wind Power Drives, RWTH Aachen, is used as the base model in this thesis. The pitch bearing is a double row four-point contact slewing bearing; this design provides exceptional load capacities, with raceways allowing rolling elements to carry loads from any direction. The rolling elements are spherical ball elements with an osculation ratio of 0.965, i.e., the ratio of raceway curvature radius to ball radius. Each row comprises

114 ball elements evenly spaced at a constant angle with adjacent elements between the raceways. In this bearing model, the inner and outer raceways are connected to the rotor hub and blade-root, respectively, with bolted joints. Few design properties of the pitch bearing are presented in the table(1).

Specification	Design Value
Bearing inner radius	1350 mm
Bearing outer radius	1600 mm
Ball element radius	35 mm
Raceway curvature radius	36.27 mm

Table 1: Pitch bearing design specifications[7]

The FEM model consists of 4 parts, namely the rotor hub and 3 blade roots, treating each part as a substructure of the model. All substructural degrees of freedom except that needed to connect this part to the global model is eliminated. All the three-pitch bearing's outer raceway is included in the rotor hub, whereas the inner raceway is included with the blade root.

4.1.1 Rolling Element modeling

Based on Hertzian elastic contact theory between solids, each ball element is modelled as non-linear traction springs mounted between the centre of curvature. Following are few researches conducted which prove the correctness and capability of this approach in modeling bearing elements

- *Alain Daidie et al.* modelled ball and raceway contact as an assembly of two rigid beam elements, two rigid shell elements, and a non-linear spring investigating loads along with contact angles in Slew Ball Bearing. Non-linear spring is assembled in between rigid beams and shell element to ensure singular deformation of spring element. This research depicts the variation in parameters like contact angle, stiffness of rings, and supporting structures play a crucial role in design. It suggests that a non-linear spring model can be extended to industrial applications, as the results of the parametric analysis are encouraging.
- *X H Gao et al.* in the article “modeling of ball–raceway contacts in a slewing bearing with non-linear springs”[27], concluded that if contacts and non-linear springs have the same load-deformation characteristics, the ball-raceway connection can be replaced with a non-linear spring. The spring stiffness is calculated using Hertz theory. The load and deformation characteristics of spring elements are similar to the previously mentioned models.

A kinematic coupled spring elements model developed by *Amin Loreimi* [7] based on the researches mentioned above is used in this thesis. The nonlinear stiffness curve was thereby obtained by high fidelity contact simulation.

4.1.2 Load Discretization Methodology

This section explains the approach for reducing load collective size of any time series data. A step-wise load aggregation method is implemented in this research as it provides control over the final load-collective size. Aggregation is a technique of reducing massive time-series data points to fewer summary data points, these summary points can explain the load variation patterns over the time series.

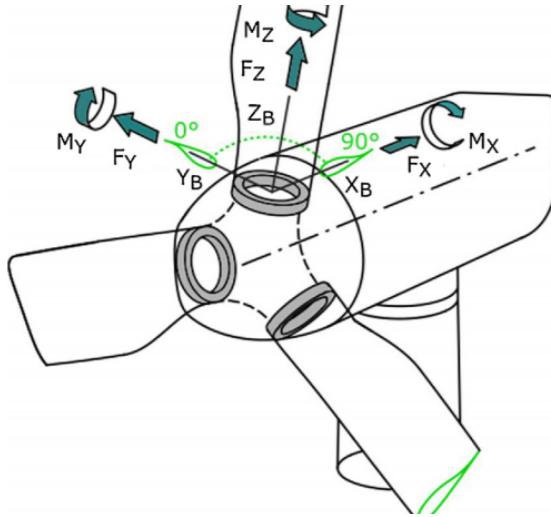


Figure 11: Pitch bearing force and bending moment notation[6]

Assuming R as rounding vector for load reduction and $r_1, r_2, r_3, \dots, r_n$ are ‘n’ load steps of a rounding vector R , a step-size ‘ h ’ is chosen such that all the loads in the rounding vector are uniformly spaced. Each load step r_i obtained by adding constant step length h to the load step before it, i.e., $r_i = r_{i-1} + h = r_{i+1} - h$. All the loads in time series are rounded to a load just greater in the rounding vector. For example the loads within the range of r_i and r_{i+1} are rounded to r_{i+1} . Since bending moments and axial force magnitudes varying by a factor of at least 10, it is recommended to use two different rounding vectors, to minimize the calculation time in the MATLAB

$$R = r_1, r_2, r_3, \dots, r_n, \quad R = f(h) \quad (26)$$

$$r_i \in [r_1, r_n], \quad r_i = r_{i-1} + h \quad (27)$$

where:

r_1, r_n : minimum and maximum loads in R

h : step-size of rounding vector

The figure(12) and (13) explains how the forces and bending moments between two step loads are rounded to a load just greater in the rounding vector. From the aggregated load collective data, it can be observed that the same force and bending moment load combinations occur multiple times over the given time series. A Duplicate check algorithm generates unique load collectives by omitting all the duplicate load combinations.

Few advantages of the implemented load reduction method are, large load combinations are reduced to a few thousand depending on the chosen step size of rounding vector, thus

cost and calculation time of analysis is noticeably minimized. As mentioned in section (3), the 3.6 million design loads from time series data are used in this analysis. The figure(14) depicts how the total load collective are reduced to fewer unique load combinations with a variation in axial force and moment step sizes.

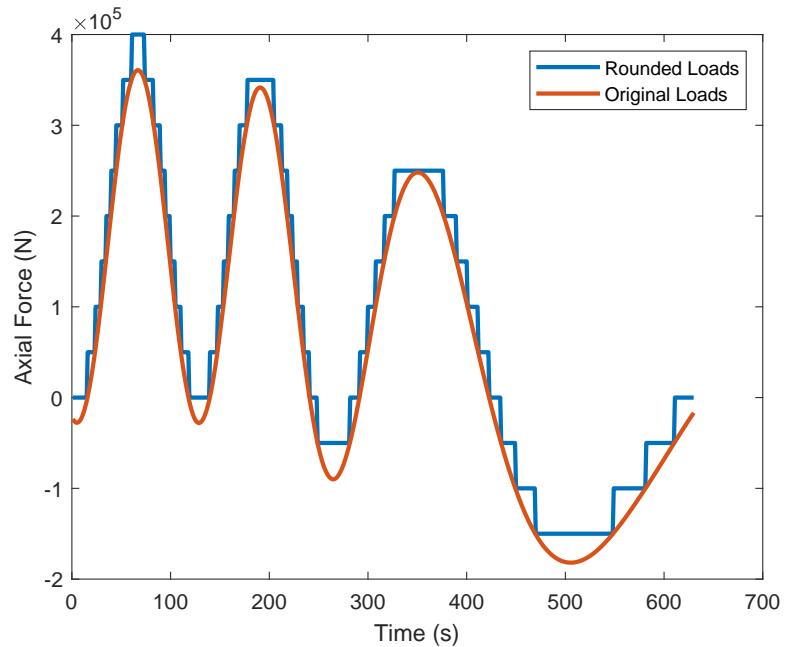


Figure 12: Axial force rounding

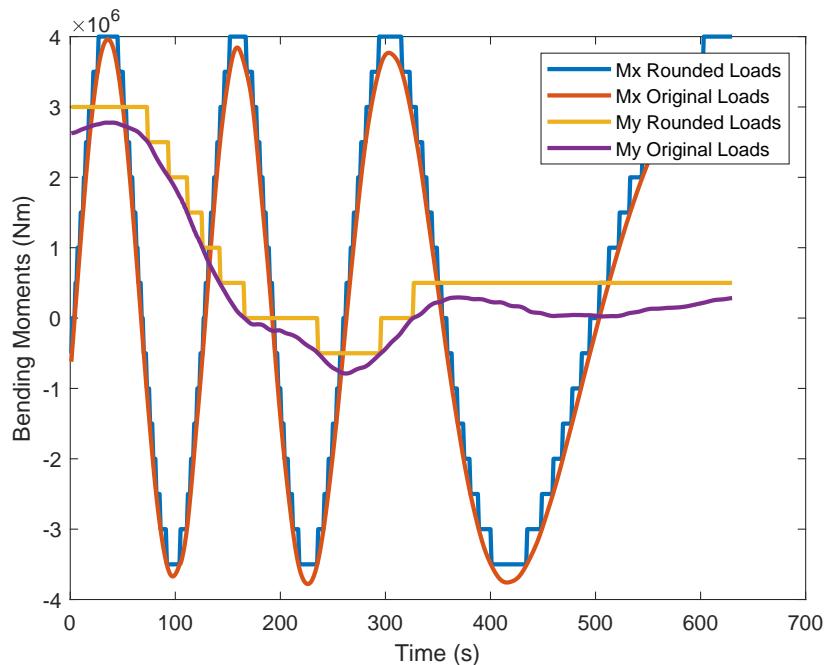


Figure 13: Bending moment rounding

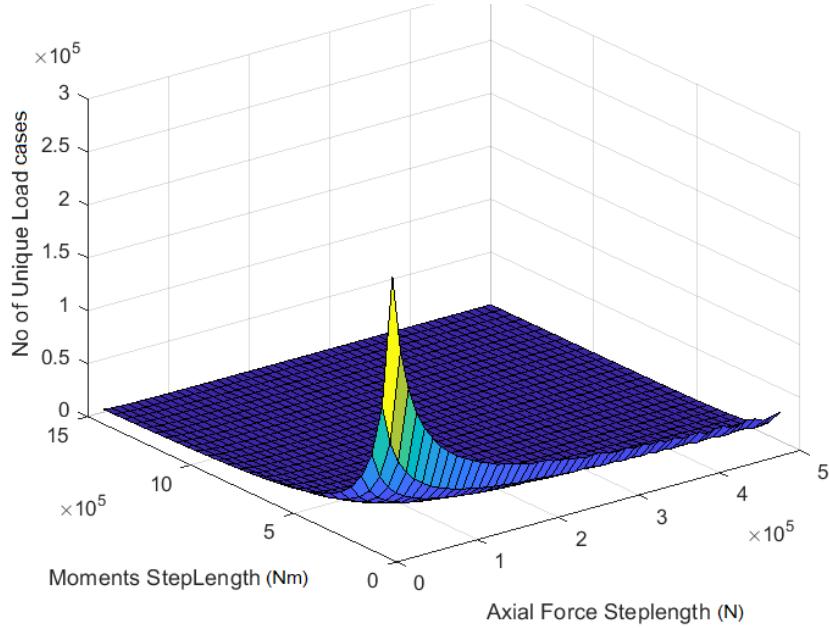


Figure 14: Variation of unique load cases with change in force and moment step size

4.1.3 Calculation and Database Generation

Using a step sizes 50 kN and 500 kNm for force, moments in the earlier mentioned load reduction methodology, the total design loads are reduced to 7229 unique load cases. The pitch bearing FEM model's structural response for each of the 7229 load cases is calculated in Abaqus/CAE. The structural deformation and the compressive force acting in each of 228 ball elements when under the loads are extracted using Python. The extracted bearing element data describes the load characteristics of the bearing and its elements. The truncation factor is the ratio of actual force acting to the minimum force required to cause contact ellipse truncation. Figure(15) depicts the relationship between contact force and contact angle of the rolling element in pitch bearing. At the given contact angle if the contact force acting on the rolling element is greater than the $f_{truncation}$ (truncation factor > 1), the contact ellipse is truncated by the raceway edge.

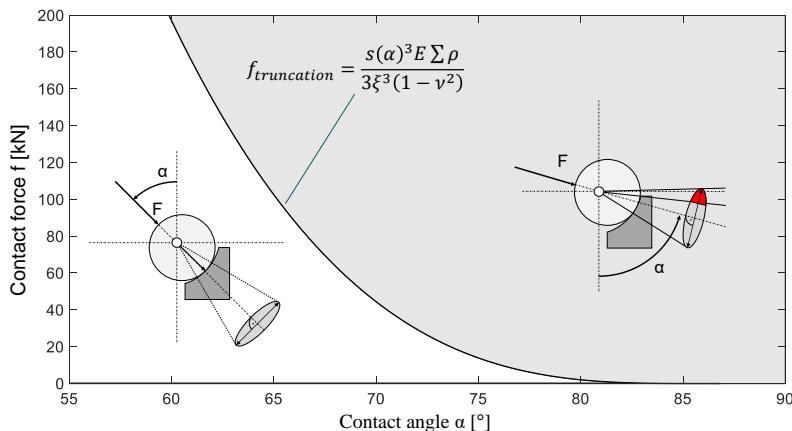


Figure 15: Critical contact force for ellipse truncation[7]

where:

E : Young modulus of raceway and ball material

ν : Poisson's ratio of the raceway and ball material

F : Contact force at ball and raceway contact

α : Contact angle of ball element

$\sum \rho$: Curvature sum of contact bodies in orthogonal planes

Assuming A, B, C, and D are contact points of two parallel ball elements with the bearing raceway in the bearing rows, as shown in the figure(16). The contact force and truncation parameter of double row bearing elements at the contact points over the raceway circumference (for all elements) for a discrete axial force and bending moments load case (500 kN , 3.5 MNm, 9.5 MNm) are shown in the figure(16). The compressive contact force and the truncation factor in the pitch bearing's ball elements are the two crucial factors that can conclude whether a specific ball element is subjected to ellipse truncation. Based on this, contact forces and truncation factor are the parameters which are analyzed over design loads to separate the critical load cases.

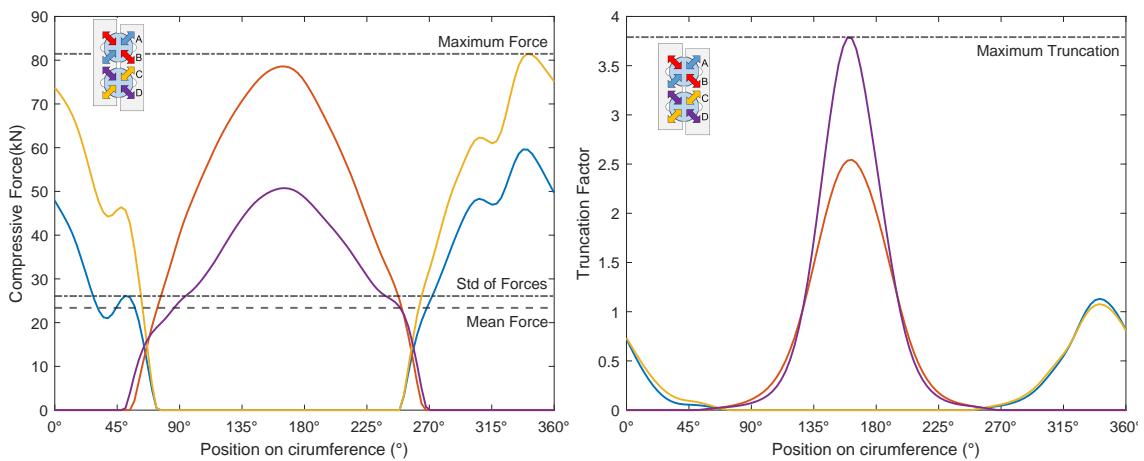


Figure 16: Contact force and Truncation factor distribution over bearing circumference

Thus, the parameters like maximum force, mean force, the standard deviation of forces, and the ellipse truncation factor for all the ball elements (as shown in the figure(16)) are calculated for all the 7229 discretized loads. These parameters define the severity of the contact force and ellipse truncation. The table(2) shows few of the bearing load parameters data calculated.

Axial Force F_z (kN)	Moment M_x (kNm)	Moment M_y (kNm)	Mean Force (kN)	Maximum Force (kN)	S.Deviation (kN)	Truncation Factor
500	3500	9500	23.38	81.45	26.08	3.787
100	-500	2500	6.96	18.99	7.01	0.107
100	-500	3000	8.14	22.96	8.29	0.15
100	-500	3500	9.3	26.96	9.56	0.20
100	5500	5000	17.81	59.37	19.49	0.915
100	5500	5500	18.54	62.96	20.34	1.036
100	5500	6000	19.3	66.51	21.24	1.227
400	2000	-3500	10.47	30.32	10.88	0.35
400	2000	-3000	9.49	26.91	9.8	0.28
400	2000	-2000	7.68	21.09	7.81	0.17

Table 2: Bearing load case and parameters variation

4.2 Approximation using Machine Learning

Instead of determining the bearing structural behavior through only FEM calculations, a new approach of approximating its structural response using machine learning regressors is implemented. Thus, total design loads are rounded and discretized so that bearing structural response in all load conditions is framed in the FEM calculations performed. Machine learning algorithms used for approximation are Linear, Polynomial, Random forest, and MLP Regressors (Artifical Neural Network). The bearing characteristics database generated for 7229 load cases during FEM postprocessing is the regression models' initial training dataset.

The regressor models are trained and tested by treating the loads in the training dataset as features and corresponding bearing charecteristics as output variables. Before training the model on the training dataset, input features (loads) are scaled using '*MinMaxScaler*'. Performing scaling removes the bias and error present in the input features, helping the regression model predict the patterns and relations between features and the output variables.

The trained regressor model performance is tested using the training dataset. Model accuracy or error in output data prediction is calculated by comparing model-predicted output with the testing dataset's corresponding output variables. If the model accuracy is not satisfactory, hyperparameters tuning is performed to increase its accuracy.

Hyperparameters are the properties of the model which govern the entire training process. These values are tunable and set before the model learning process begins. Hyperparameters can have direct control and impact on the training behavior of the machine learning algorithms. Some examples of the hyperparameters are learning rate, hidden layers, activation functions. The table(3) shows the best hyperparameter combinations of all the four regression models.

Hyperparameter	Linear Regressor	Polynomial Regressor	Random Forest Regressor	MLP Regressor (Neural Network)
<i>degree</i>	-	4	-	-
<i>activation function</i>	-	-	-	rectified linear unit function (relu)
<i>n-jobs / order /solver</i>	2 or more	F	2 or more	stochastic gradient based optimizer (adam)
<i>n_estimators</i>	-	-	100 - 500	-
<i>max_iterations</i>	-	-	-	till convergence
<i>hidden_layer_sizes</i>	-	-	-	Layers : 2 or 3 size: (100,50,25) or (50,200,25)

Table 3: Hyperparameter tuning of the regression models

The best value of the hyperparameter ‘*Degree*’ in the polynomial regression model is 5. It is observed that any other degree than ‘5’ is not suited for this study; this is concluded by checking the model’s inability to predict outputs at load cases with lower magnitudes. As we need the model to predict 3.6 million load cases, it is suggested to use parallel programming in all the regression models.

The best hidden layer topology for MLP regressor is obtained by evaluating the hidden layers size and number of neurons in the hidden layer. By varying hidden layer size from 2 to 5 layers with corresponding number of neurons from 25 - 200 in each hidden layer the prediction capability of MLP regressor is tested. And the topology with least error values is shown in the table(3).

4.2.1 Identifying Critical Loads

After hyperparameter tuning is performed, the bearing characteristics over the whole design loads are predicted using the trained regression models. Since stiffening plates are primarily designed to decrease the force acting on the pitch bearings which is responsible for ellipse truncation, it is suggested to identify these complete ellipse truncation causing load cases from the design loads. The predicted maximum contact force data is evaluated, and a maximum force threshold is chosen. The chosen maximum force threshold is so that the bearing experiences a maximum contact force above this threshold for 25% of its lifetime. In this analysis, a load case is considered critical if it produces a truncation factor above one or contact force above the threshold value (55 kN) in the bearing ball elements. Approximation performance of each regression model is analyzed individually as well as also by comparing with the other regression models.

Once the best regression model is identified based on its performance, the critical load cases are identified from design load cases and separated using MATLAB. The critical load cases identified in the time series data are again aggregated to fewer load cases. A new load and bearing parameters database for the aggregated critical loads is extracted from the 7729 load cases database generated before. This critical load database is further used to check the ability of best machine learning model for critical load identification with minimum training data size.

5 Results and Discussion

Pitch bearing FEM model is a non-linear model; the regression models are expected to predict the nonlinear system response. Expect in the linear regression model's case, the regression model's approximation capability has significantly increased by tuning the model's hyperparameters and their prediction accuracy's (mean absolute error) are shown in the table(4).

Regression Model	Mean Force	Maximum Force	Standard Deviation	Truncation Factor
Linear Regression	32%	37%	35%	147%
Polynomial Regression	8%	8.5%	7.5%	15.5%
Random Forest Regression	1.5%	2%	0.5%	3.5%
MLP Regression	1%	1.2%	0.65%	4%

Table 4: Regression models prediction error

It is quite apparent that the linear regression fails to predict these non-linear pitch bearing parameters, which is due to the reason that a linear regression does not include nonlinearities in it. The polynomial regressor predicts force parameters with a mean absolute error of 8% and truncation factor with an error of up to 15.5%. Simultaneously both the MLP and random forest regressors performed quite exceptionally in the approximating the output's which can be seen in the table(4).

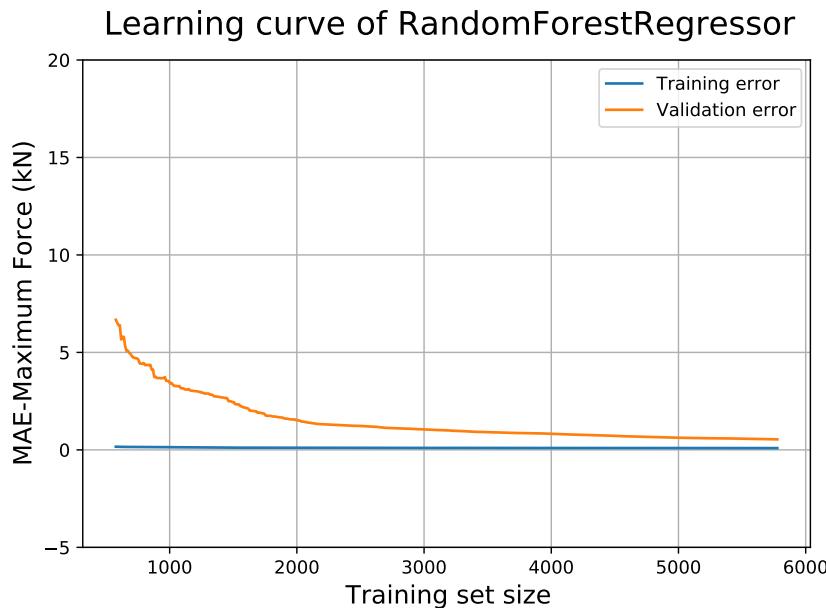


Figure 17: Maximum force learning curve of Random forest regression

By analyzing the training and validation error it is observed that the training data size of 7229 load cases is more than sufficient for the nonlinear regression models. The figures

(17) & (18) show that using training data as less as 2000 to 3000 load cases, the random forest and MLP regressors can approximate the output (maximum force) with an error within the error tolerance. This gives a hint that the bearing parameters over the entire design load series can be approximated using more less FEM calculations (less database size) than we have done in this analysis.

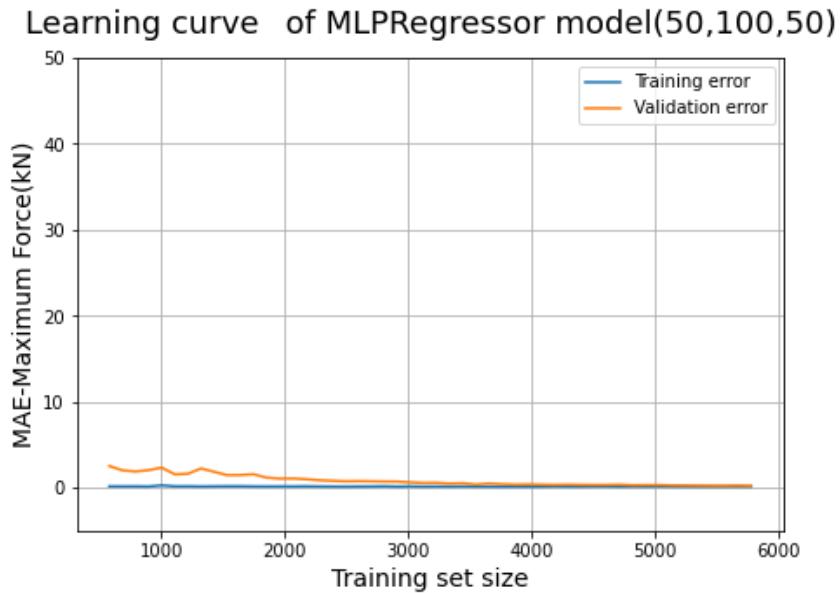


Figure 18: Maximum force learning curve of MLP regression

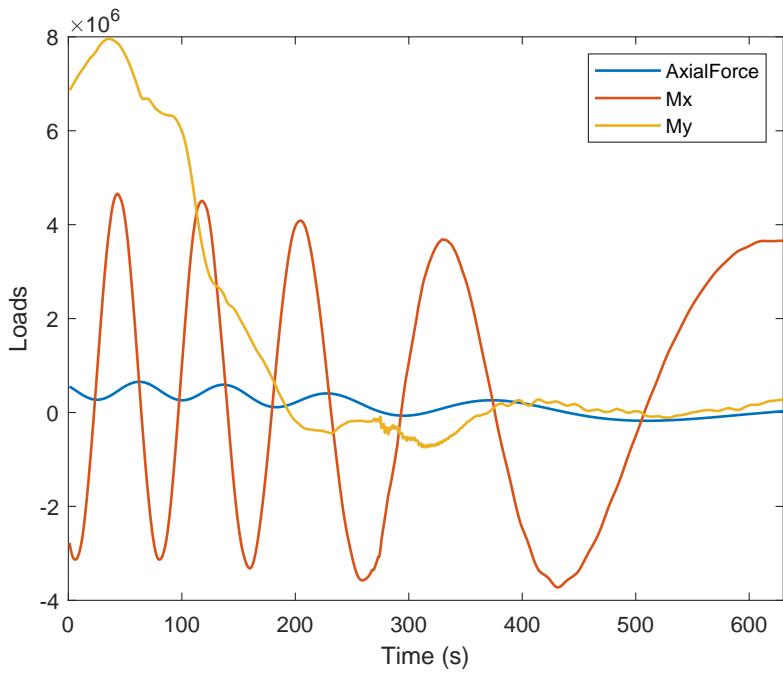


Figure 19: Load combination sample for design loads

The maximum force figure(20) for the load combinations in the figure(19) shows that a small change in axial force (F_z) and bending moment (M_y) causes high fluctuations in maximum force on the pitch bearing ball elements. From the approximated maximum

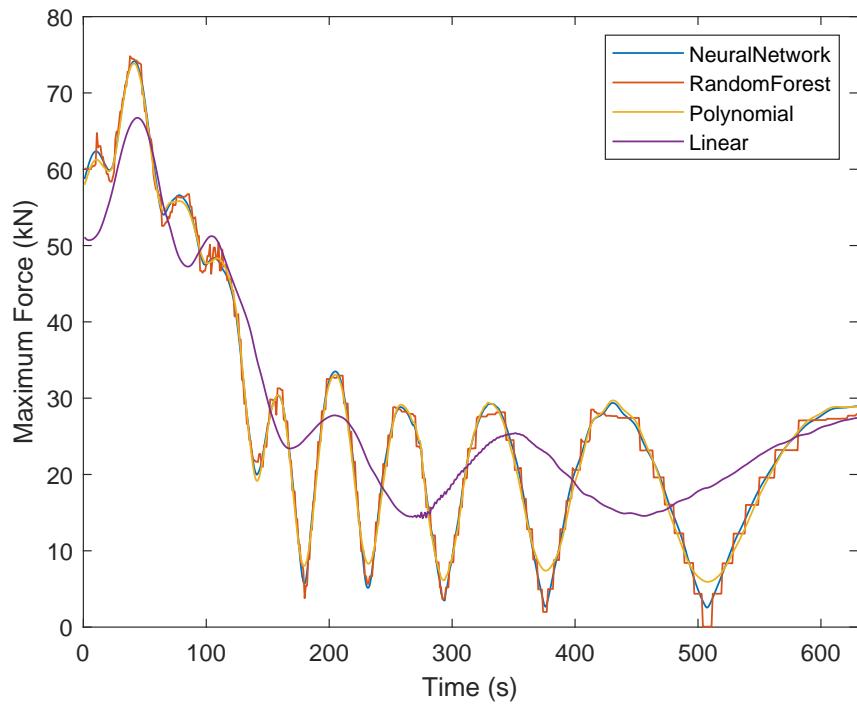


Figure 20: Maximum force approximation by all regressors

force figure(20) it is evident that the linear regressor completely fails to approximate the outputs, as compared to other regression models. Where as, polynomial regressor lacks in identifying the change patterns and approximate the outputs of highly fluctuating loads especially at low load magnitudes with the required accuracy as seen in the figure(20). The reason for this can be that polynomial regressor fails to approximate outputs for features that are outliers from the training data, meaning these load cases are new to the polynomial regressor, and it is unable to estimate output for these loads. A Polynomial regressor model with a higher degree (degree > 5) can predict the output data for these load cases; however, this type of model (degree > 5) has more prediction error than the lower degree models.

On the other hand, the random forest regressor has a step approximation curve rather than smooth; this can be because of the fact that random forest regressor approximates outputs using the mean value of outputs in the training data as explained in section(2.6.4) before. The MLP regressor (Neural network), approximates a new output for any slight change in the input loads with an error shown in the table(4), it's high accuracy is because of its error reduction capability by back propagation. The design series loads in this thesis are Weibull distributed; thus, the output parameters obtained over the design loads seem Weibull distributed with different shape and scale parameters which can be seen in the following figures.

The following figures in this section are the distribution figures approximated by the regression models for the entire time series design loads. The truncation factor figures (21) and (22) depict that data obtained by MLP regressor (Neural network) is more continuous than the random forest and polynomial regressors. The data approximated by polynomial regressor is shown in figure(21). Even though numerous load cases are causing a truncation greater than '3' as seen in the discretized training data, polynomial

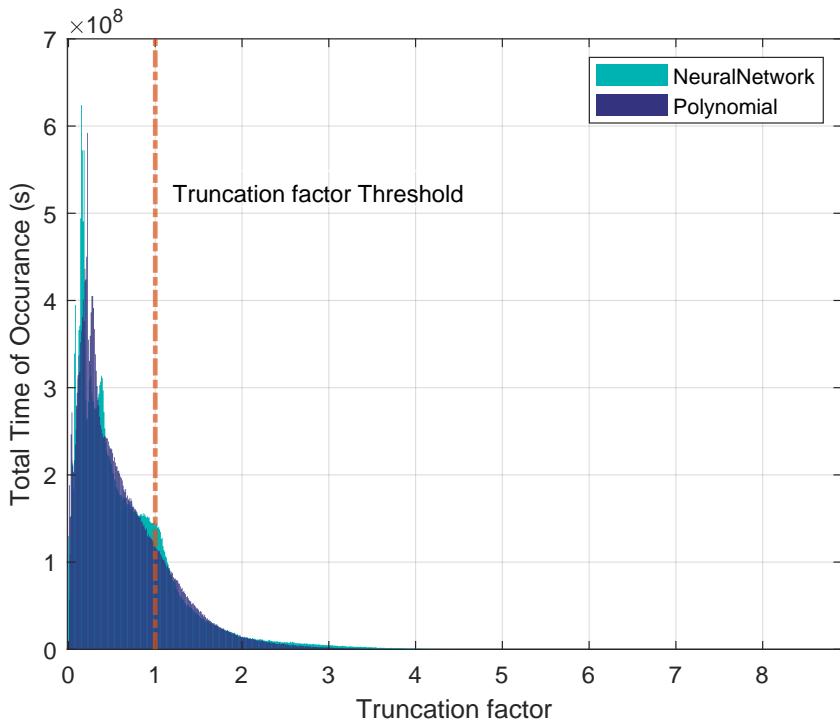


Figure 21: Truncation distribution by polynomial and MLP regression

regression did not identify these load cases in the time continuous design loads. This shows the inability of polynomial regressor in predicting the truncation factor. On the other hand, as random forest uses mean outputs from training data, approximations made by it are discrete and discontinuous.

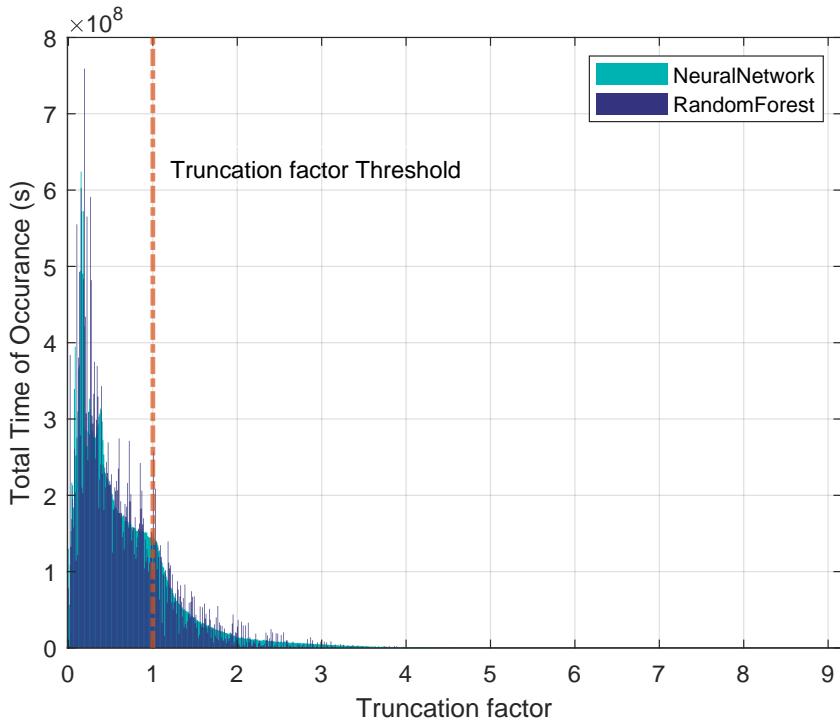


Figure 22: Truncation distribution by random forest and MLP regression

The truncation factor approximation by linear regressor (figure(23)) has a very high spike

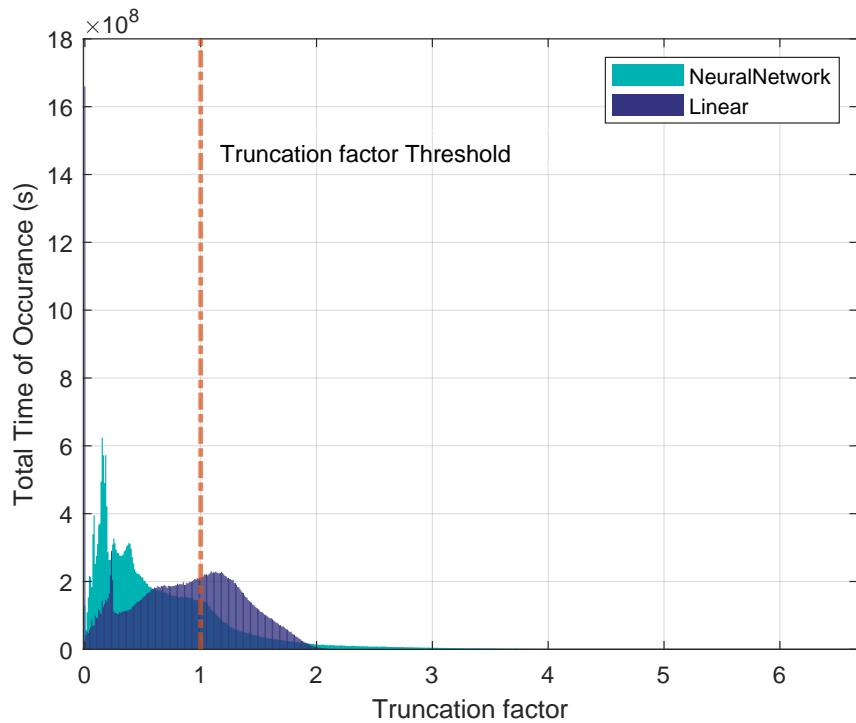


Figure 23: Truncation distribution by linear and MLP regression

at ‘0’, and also shows that half of the pitch bearing design loads cause a truncation factor greater than the threshold. Since the truncation factor by definition is mostly greater than zero by some value, the data predicted by linear regressor in this scenario cannot be the true data, this explains that critical load cases causing truncation factor above threshold as approximated by linear regressor are not the actual critical loads.

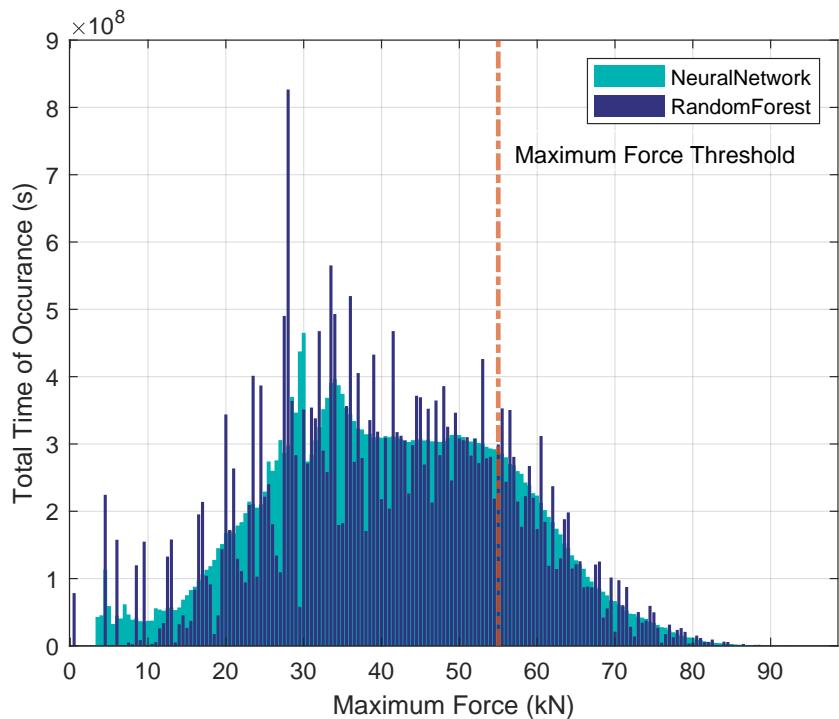


Figure 24: Maximum force distribution random forest and MLP regression

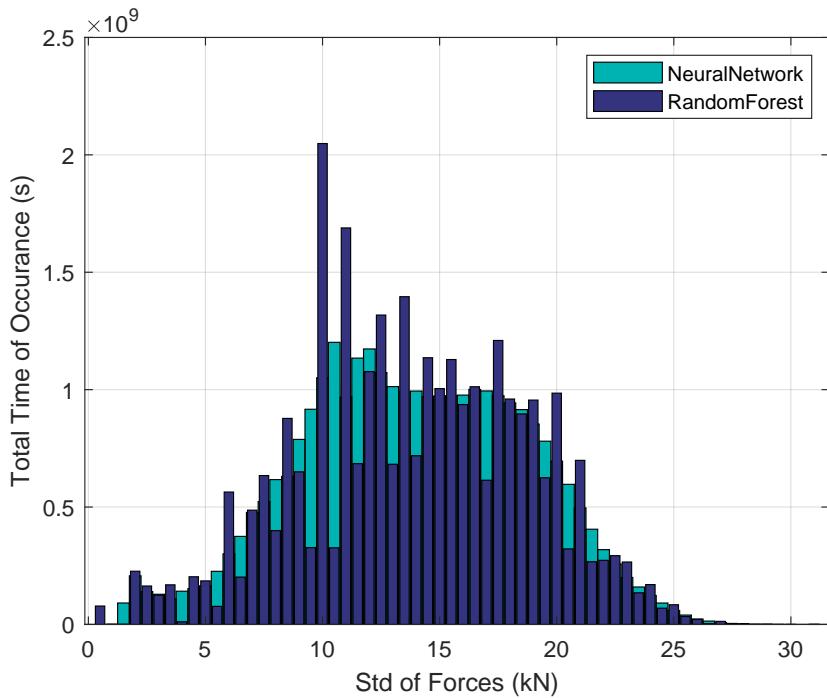


Figure 25: Standard deviation distribution by random forest and MLP regression

Figures (24) and (25) show that random forest approximated outputs are entirely discontinuous and not weibull distributed compared to MLP regressor. At the same time, the maximum force approximation figure(26) and standard deviation figure(27) over the entire design loads gives a clear idea that polynomial regressor fails to predict the low contact force data such as below 10 kN when compared to the MLP regressor. This means that these load cases are wrongly classified and may be included in the critical loads.

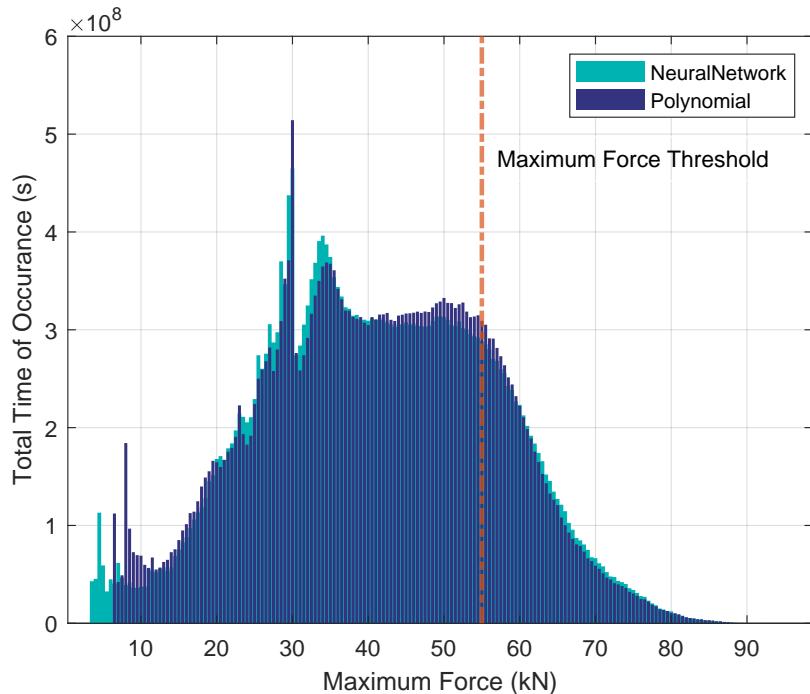


Figure 26: Maximum force distribution polynomial and MLP regression

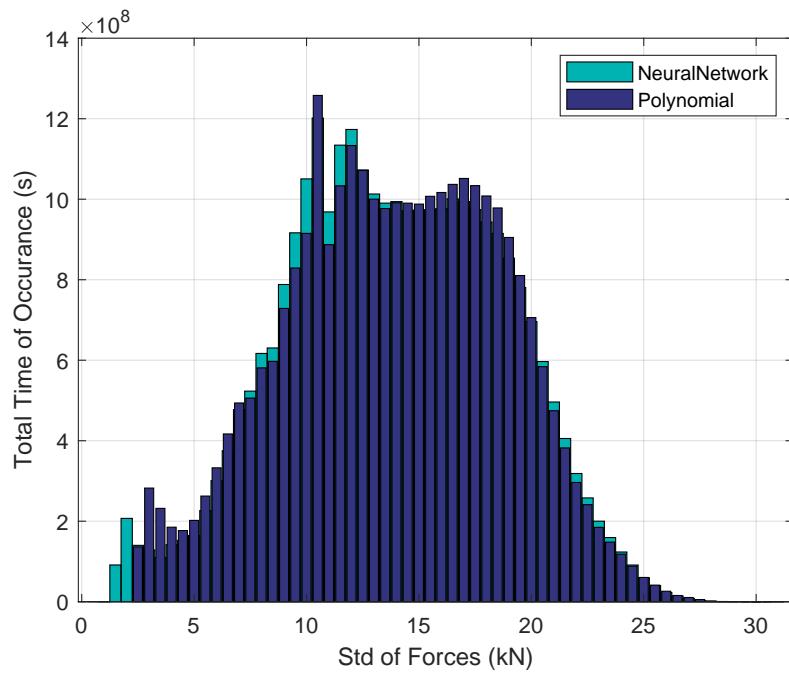


Figure 27: Standard deviation distribution by polynomial and MLP regression

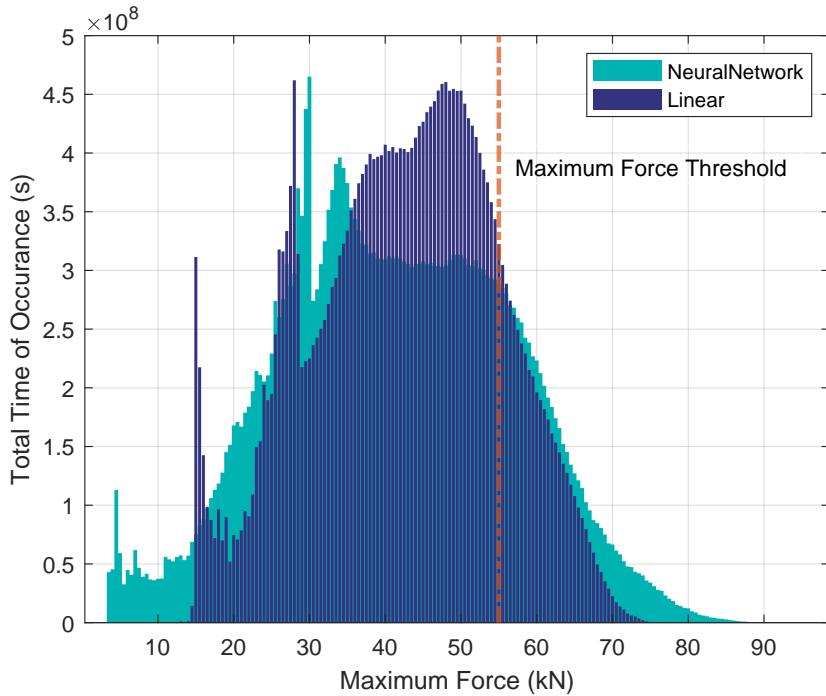


Figure 28: Maximum force distribution linear and MLP regression

Figures (28) and (29) are the comparison graphs of outputs by linear and MLP regressor. Even though there are many low and high maximum force causing load cases as approximated by other regressors, linear regressor did not detect and approximate outputs in these scenarios. These wrong predictions prevail even when the training data size to the linear regressor is increased. Increasing training data size usually increases chances for the regressor to detect relationships between input and outputs, which is not occurring in linear regressor approximation, implying that linear regressor is not suitable for approximating bearing parameters. Since maximum force and truncation factor are vital factors

for identifying critical loads, based on the above discussion and regressor's performance, it is suggestable to use MLP regressor (Neural network) as best regressor, to identify the time continuous critical loads from design loads.

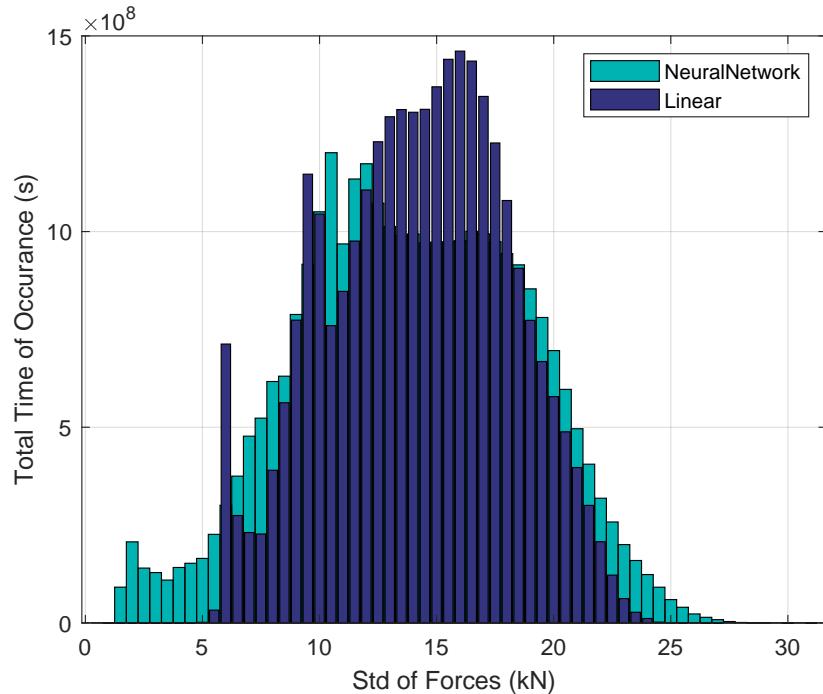


Figure 29: Standard deviation distribution linear and MLP regression

The critical loads (time series) identified from 3.6 Million design loads by MLP regressor are about 330 thousand approximately. These obtained load cases cause either an ellipse truncation of above '1' or cause a maximum contact force above maximum force threshold (55 kN) in ball elements of pitch bearing. These load cases are further reduced into fewer hundred discretized loads using aggregation methods. Using the critical loads identified before as training data for the MLP regressor, its ability to predict bearing characteristics when under fewer loads is tested. It is observed that a maximum step size for the approximating bearing characteristics over design loads with permissible error limits is (axial force: 200 kN, bending moment: 500 kNm).

The following table(5) depicts the performance of MLP regressor with varying training database size. Suppose an error tolerance or threshold of 10% is chosen, it is evident from the table(5) that the minimum training data size required by the best performing MLP regressor for identifying the critical loads from design loads is at least 317 discrete load cases obtained through aggregation.

Load step size for aggregation (force, moment)	Training Database size (No.of Load cases)	Mean Force	Maximum Force	Standard Deviation	Truncation Factor
100 kN, 500 kNm	847	1%	1.2%	1%	3.5%
150 kN, 500 kNm	441	1.5%	10%	1.5%	4%
200 kN, 500 kNm	317	3%	10%	1.5%	7%
100 kN, 1000 kNm	269	10%	15%	2%	40%

Table 5: MLP regressor approximation error with change in training data size

6 Conclusion

A new coupled FEM-Machine learning method for identifying a reduced bearing load collective from a massive set of design loads is developed. This reduced load collective can further be used for shape optimization of the stiffening plate of pitch bearing. This method is more cost-effective and time-efficient than the standard FEM method in critical load identification.

The developed algorithms establish the relationship between bearing loads and the bearing characteristics using the training data (FEM database). These relations are further used to approximate or predict the bearing characteristics for any loads rather than calculating the characteristics every time for the applied load as done in the standard FEM method. It is observed that the relationship between loads acting and the resulting bearing characteristics are nonlinear. Few factors which affect the accuracy of the predicted bearing characteristics are as follows.

- The chosen step size for aggregation or rounding of the bearing load combinations significantly influences the approximated bearing characteristics; the discretized load cases act as the machine learning model's training points.
- FEM database generated for discretized design loads should include bearing structural response in all load conditions so that regressor can establish relationships between variables in all conditions during its training.
- Parametrization of the pitch bearing characteristics is also a critical aspect in this prediction method, as the number of parameters deciding the critical loads varies.

The bearing load characteristics in this analysis are parameterized into mainly four parameters, namely mean, the maximum, standard deviation of contact forces and truncation factor. The approximated results show that few design load cases are causing a maximum contact force and truncation factor in ball elements up to 100 kN and '7' respectively, which can severely affect the bearing life. Hence such critical loads are classified from the design loads mainly based on these two parameters. The parameter thresholds chosen for the critical load classification are 55 kN for maximum force and '1' for truncation factor. The load cases influencing the service life of rotor blade bearings are identified from the design loads and discretized into a few hundred load cases. Through this analysis, it is observed that there are almost 1/10th of the design load cases which can cause failure in the pitch bearing by ellipse truncation.

MLP regressor (ANN) outperformed all the other developed regression models in this study. MLP regression model is capable of approximating the time series bearing characteristics even with training data size as less as a few hundred, thus achieving the main objective of this analysis. The maximum load step sizes for aggregation (force, moment) using this model with an acceptable approximation error of 10% are (200 kN, 500 kNm).

The mean absolute error of MLP regressor approximated data is within the tolerance limit; this can be because of error reduction capability of it through backpropagation, which is not employed in other models. Contrary to this linear, polynomial, and random

forest regression models failed to approximate or predict the outputs (bearing parameters) in many aspects. Polynomial regressor did not predict the output in highly fluctuating load scenarios. On the other hand, random forest by its nature is using the mean of the training data outputs to approximate the design load outputs rather than approximating the actual outputs.

As further work, the reduced loads identified by the developed new method can be used in shape optimization of the stiffening plates, thus reducing the calculations and cost of analysis. In future, this coupled FEM-Machine learning can be used in designing any component of wind turbines whose designing requires conducting a large number of FEM calculations. Before implementing this method, the three earlier mentioned factors need to be addressed.

References

- [1] Schaffarczyk A.P. *Introduction to Wind Turbine Aerodynamics*. Springer International Publishing, February 2020. ISBN 978-3-030-41028-52.
 - [2] Marcel Lesieur. *Basic Fluid Dynamics*. In: *Turbulence in Fluids. Fluid Mechanics and its Applications*. Springer, Dordrech. ISBN 978-1-4020-6435-7. URL https://doi.org/10.1007/978-1-4020-6435-7_2.
 - [3] WMH Herion Antriebstechnik GmbH. Slewing Bearings Catalog, 2002. URL <https://www.nrel.gov/docs/fy02osti/26645.pdf>.
 - [4] T. Harris and J.H. Rumbarger. Wind turbine design guideline DG 03: Yaw and Pitch Rolling Bearing Life, 2009. URL <https://www.nrel.gov/docs/fy10osti/42362.pdf>.
 - [5] Corey D.Bayles. Extend wind turbine life with pitch bearing upgrades. URL <https://www.kaydonbearings.com>.
 - [6] Matthias Stammller, Andreas Reuter, and Gerhard Poll. Cycle counting of roller bearing oscillations – case study of wind turbine individual pitching system, June, 2019.
 - [7] A Loriemi, G Jacobs, D Bosse, and M Zweiffel. Improvement of the pitch bearing load distribution by shape-optimized stiffening plates, September 2020.
 - [8] BP plc. Statistical review of world energy, 68thedition, 2019.
 - [9] EL Petersen. *1999 European Wind Energy Conference : wind energy for the next millennium*. James & James Ltd, March 1999. ISBN 9781902916002.
 - [10] IRENA (2019). Future of wind: Deployment, investment, technology, grid integration and socio-economic aspects, 2019 edition. URL <http://www.irena.org/publications>.
 - [11] GWEC. Global wind report 2019.
 - [12] Trevor M. Letcher. *Wind Energy Engineering : A Handbook for Onshore and Offshore Wind Turbines*. ACADEMIC PRESS, 2017. ISBN 978-0-12-809451-8.
 - [13] M.Reccab Ochieng and N.Frederick. Physical formulation of the expression of wind power.
 - [14] Ameya Sathe and Wim Bierbooms. Influence of different wind profiles due to varying atmospheric stability on the fatigue life of wind turbines, June, 2019.
 - [15] Kuik, Okulov, and G.A.M. The betz-joukowsky limit for the maximum power coefficient of wind turbines, 2009. URL <http://isjaee.hydrogen.ru/?pid=1845>.
 - [16] Eric Lantz, Owen Roberts, Jake Nunemaker, Edgar DeMeo, Katherine Dykes, and George Scott. Increasing Wind Turbine Tower Heights: Opportunities and Challenges, 2019. URL <https://www.nrel.gov/docs/fy19osti/73629.pdf>.
-

- [17] L. Fingersh, M. Hand, and A. Laxson. Wind Turbine Design Cost and Scaling Model, December, 2006. URL <https://www.osti.gov/>.
- [18] J. Cotrell. The Mechanical Design, Analysis, and Testing of a Two-Bladed Wind Turbine Hub, 2002. URL <https://www.nrel.gov/docs/fy02osti/26645.pdf>.
- [19] Matthias Stammller, Philipp Thomas, and Fabian Schwack. Effect of load reduction mechanisms on load and blade bearing movements of wind turbines, November, 2019. URL <https://doi.org/10.1002/we.2428>.
- [20] Matthias Stammller, Sebastian Baust, Andreas Reuter, and Gerhard Poll. Load distribution in a roller-type rotor blade bearing, November, 2019. URL doi:10.1088/1742-6596/1037/4/042016.
- [21] R Lostado, F J Martínez de Pisón, A Pernía, F Alba, and J Blancos. Combining regression trees and the finite element method to define stress models of highly non-linear mechanical systems, April 2019.
- [22] T.H.E. Gulikers. An integrated machine learning and finite element analysis framework, applied to composite substructures including damage, December, 2018.
- [23] Sunil Kumar, Dr.Vijay Kumar, and Dr.Anoop KumarSingh. Prediction of maximum pressure of journal bearing using ANN with multiple input parameters, May 2020.
- [24] Ali Madani, Ahmed Bakhaty, Jiwon Kim, Yara Mubarak, and Mohammad Mofrad. Bridging finite element and machine learning modeling: Stress prediction of arterial walls in atherosclerosis, March 2019.
- [25] P Pedregosa et al. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.
- [26] Breiman L. *Classification and Regression Trees*. Chapman & Hall/CRC, October 2017. ISBN 9781315139470. URL <https://doi.org/10.1201/9781315139470>.
- [27] X H Gao, X D Huang, H Wang, and J Chen. Modelling of ball–raceway contacts in a slewing bearing with non-linear springs, September, 2010.

A Appendix

Algorithm 1: Linear Regression Algorithm

Input: Database with bearing forces and corresponding bearing parameters F_m , F_{max} , F_{sd} and μ_{max} (mean, maximum, standard deviation and truncation factor of forces in bearing ball elements)

Output: Regressor predicted bearing parameters F_m , F_{max} , F_{sd} and μ_{max} for entire pitch bearing design loads

```
# Importing the required libraries
import pandas as pd
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_absolute_error, mean_squared_error

#Importing the Input and force dataset whose output to be predicted
# as a pandas Dataframe
df_inp = pd.read_csv('InputData.csv', index_col=False)
df_out = pd.read_csv('OutputData.csv', index_col=False)

#Scaling the input independent variables (Force and Bending moments)
#Performing scaling remove errors and biases for input data
features = ['AxialForce', 'MomentX', 'MomentY']
target = ['MeanForce', 'MaxForce', 'Std_Force', 'Truncation']
scaler = MinMaxScaler()
scaler.fit(df_inp[features])
scaler.transform(df_inp[features])
scaler.transform(df_out[features])
X = df_inp[features], Y = df_inp[target]

#Dividing input data into two parts as training and testing dataset
xTrain, xTest, yTrain, yTest = train_test_split(X, Y, train_size=0.6)
reg = LinearRegression()
reg.fit(xTrain, yTrain)

#Predicting response for 'xTest' dataset by trained 'reg' model
#Calculating error values between 'pred_y_Test' and actual 'yTest'
pred_y_Test = reg.predict(xTest)
mae = mean_absolute_error(yTest, pred_y_Test) # mean absolute error
mse = mean_squared_error(yTest, pred_y_Test) # mean square error

#Parameter tuning is performed, if 'mae', 'mse' are very high
#Predicting response for 'df_out' dataset using trained model
X_out = df_out[features]
Y_out = reg.predict(X_out)
Y_out.to_csv(index=False) # saving predicted response as 'csv'
```

Algorithm 2: Polynomial Regression Algorithm

Input: Database with bearing forces and corresponding bearing parameters F_m , F_{max} , F_{sd} and μ_{max} (mean, maximum, standard deviation and truncation factor of forces in bearing ball elements)

Output: Polynomial regressor predicted bearing parameters F_m , F_{max} , F_{sd} and μ_{max} for entire pitch bearing design loads

```
# Importing the required libraries
import pandas as pd
from sklearn.linear_model import LinearRegression as Linear_reg
from sklearn.preprocessing import MinMaxScaler, PolynomialFeatures
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_absolute_error, mean_squared_error

#Importing Input and force dataset whose output to be predicted
df_inp = pd.read_csv('InputData.csv', index_col=False)
df_out = pd.read_csv('OutputData.csv', index_col=False)
degree = 5          # Defining the degree of polynomial regressor

#Scaling independent variables (Force and Bending moments)
#Performing scaling remove errors and biases for input data
features = ['AxialForce', 'MomentX', 'MomentY']
target = ['MeanForce', 'MaxForce', 'Std_Force', 'Truncation']
scaler = MinMaxScaler()
scaler.fit(df_inp[features])
scaler.transform(df_inp[features])
scaler.transform(df_out[features])
X = df_inp[features], Y = df_inp[target]

#Dividing input data into two parts as training & testing dataset
xTrain, xTest, yTrain, yTest = train_test_split(X, Y, train_size = 0.6)
poly_reg = make_pipeline(PolynomialFeatures(degree), Linear_reg())
poly_reg.fit(xTrain, yTrain)

#Predicting response for 'xTest' dataset by trained 'reg' model
#Calculating error values between 'pred_y_Test' and actual 'yTest'
pred_y_Test = poly_reg.predict(xTest)
mae = mean_absolute_error(yTest, pred_y_Test) # mean absolute error
mse = mean_squared_error(yTest, pred_y_Test) # mean square error

#Parameter tuning is performed, if 'mae', 'mse' are very high
#Predicting response for 'df_out' dataset using trained model
X_out = df_out[features]
Y_out = poly_reg.predict(X_out)
Y_out.to_csv(index=False) # saving predicted response as 'csv'
```

Algorithm 3: Random Forest Regression Algorithm

Input: Database with bearing forces and corresponding bearing parameters F_m , F_{max} , F_{sd} and μ_{max} (mean, maximum, standard deviation and truncation factor of forces in bearing ball elements)

Output: Regressor predicted bearing parameters F_m , F_{max} , F_{sd} and μ_{max} for entire pitch bearing design loads

```
# Importing the required libraries
import pandas as pd
from sklearn.ensemble import RandomForestRegressor
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_absolute_error, mean_squared_error

#Importing Input and force dataset whose output to be predicted
df_inp = pd.read_csv('InputData.csv', index_col=False)
df_out = pd.read_csv('OutputData.csv', index_col=False)
n_estimators = 100      #Number of trees in forest (hyperparameter)

#Scaling independent variables (Force and Bending moments)
#Performing scaling remove errors and biases for input data
features = ['AxialForce', 'MomentX', 'MomentY']
target = ['MeanForce', 'MaxForce', 'Std_Force', 'Truncation']
scaler = MinMaxScaler()
scaler.fit(df_inp[features])
scaler.transform(df_inp[features])
scaler.transform(df_out[features])
X = df_inp[features], Y = df_inp[target]

#Dividing input data into two parts as training & testing dataset
xTrain, xTest, yTrain, yTest = train_test_split(X, Y, train_size = 0.6)
rf_reg = RandomForestRegressor(n_estimators, random_state = 0)
rf_reg.fit(xTrain, yTrain)

#Predicting response for 'xTest' dataset by trained 'reg' model
#Calculating error values between 'pred_y_Test' and actual 'yTest'
pred_y_Test = rf_reg.predict(xTest)
mae = mean_absolute_error(yTest, pred_y_Test) # mean absolute error
mse = mean_squared_error(yTest, pred_y_Test) # mean square error

#Parameter tuning is performed, if 'mae', 'mse' are very high
#Predicting response for 'df_out' dataset using trained model
X_out = df_out[features]
Y_out = rf_reg.predict(X_out)
Y_out.to_csv(index=False) # saving predicted response as 'csv'
```

Algorithm 4: Multi Layer Perceptron Regression Algorithm

Input: Database with bearing forces and corresponding bearing parameters F_m , F_{max} , F_{sd} and μ_{max} (mean, maximum, standard deviation and truncation factor of forces in bearing ball elements)

Output: Regressor predicted bearing parameters F_m , F_{max} , F_{sd} and μ_{max} for entire pitch bearing design loads

```
# Importing the required libraries
import pandas as pd
from sklearn.neural_network import MLPRegressor
from sklearn.preprocessing import MinMaxScaler
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_absolute_error, mean_squared_error

#Importing Input and force dataset whose output to be predicted
df_inp = pd.read_csv('InputData.csv', index_col=False)
df_out = pd.read_csv('OutputData.csv', index_col=False)
hidden_layer_sizes=(50,200,25) #Number of neurons in hidden layers
max_iter = 1500 #Depends on convergence of model

#Scaling independent variables (Force and Bending moments)
#Performing scaling remove errors and biases for input data
features = ['AxialForce', 'MomentX', 'MomentY']
target = ['MeanForce', 'MaxForce', 'Std_Force', 'Truncation']
scaler = MinMaxScaler()
scaler.fit(df_inp[features])
scaler.transform(df_inp[features])
scaler.transform(df_out[features])
X = df_inp[features], Y = df_inp[target]

#Dividing input data into two parts as training & testing dataset
xTrain, xTest, yTrain, yTest = train_test_split(X, Y, train_size = 0.6)
mlp_reg = MLPRegressor(hidden_layer_sizes, max_iter)
mlp_reg.fit(xTrain, yTrain)

#Predicting response for 'xTest' dataset by trained 'reg' model
#Calculating error values between 'pred_y_Test' and actual 'yTest'
pred_y_Test = mlp_reg.predict(xTest)
mae = mean_absolute_error(yTest, pred_y_Test) # mean absolute error
mse = mean_squared_error(yTest, pred_y_Test) # mean square error

#Parameter tuning is performed, if 'mae', 'mse' are very high
#Predicting response for 'df_out' dataset using trained model
X_out = df_out[features]
Y_out = mlp_reg.predict(X_out)
Y_out.to_csv(index=False) # saving predicted response as 'csv'
```

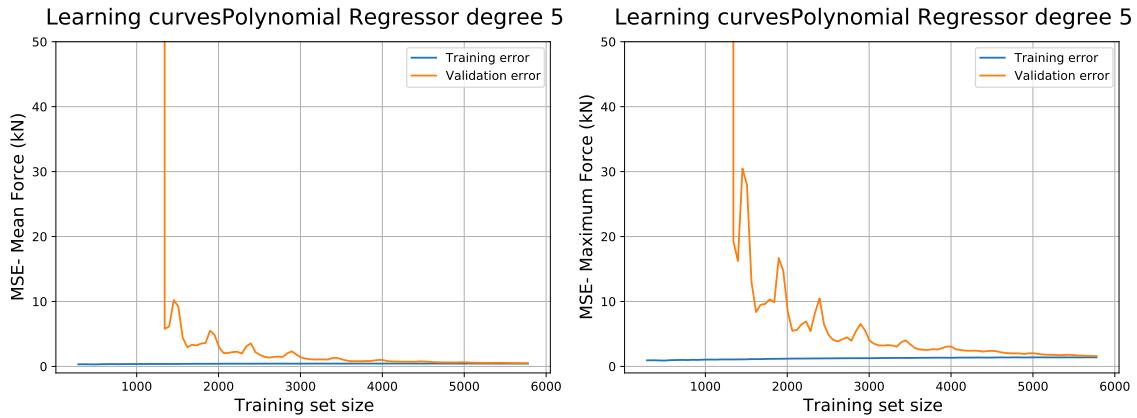


Figure 30: Learning curve of Polynomial regressor for Mean and Maximum force

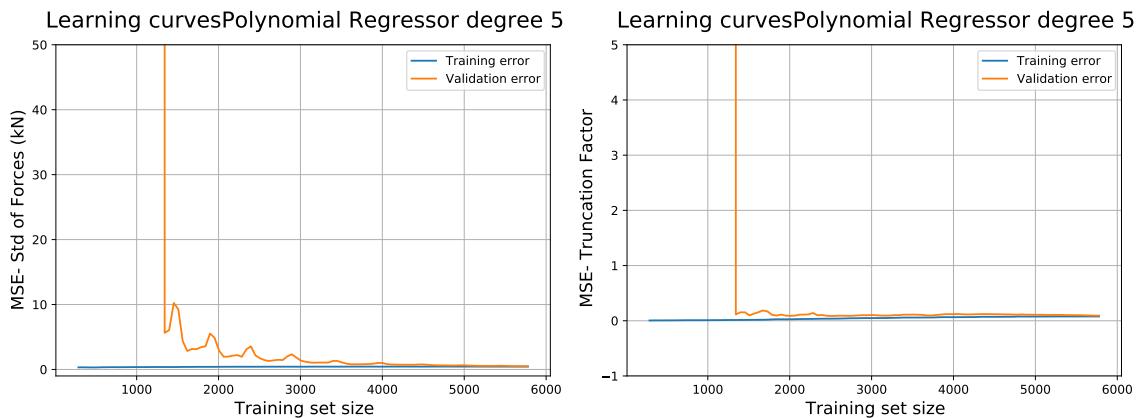


Figure 31: Learning curve of Polynomial regressor for Standard deviation and Truncation factor

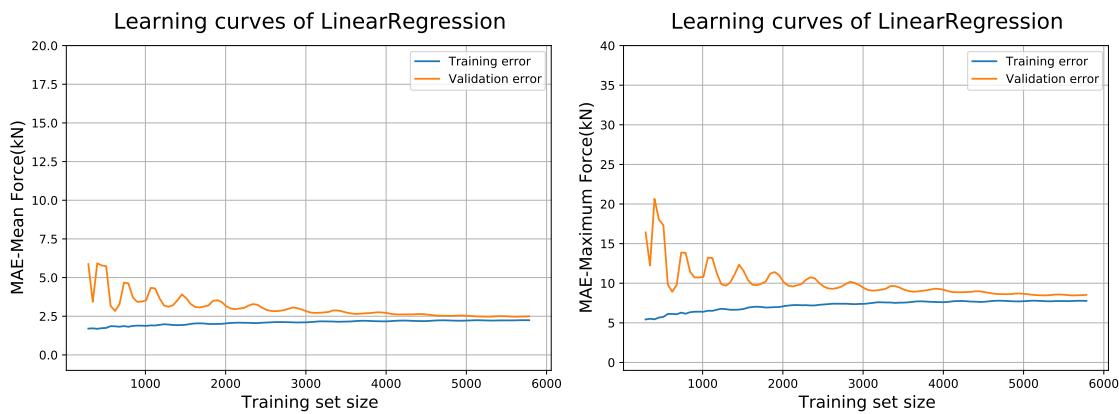


Figure 32: Learning curve of linear regressor for Mean and Maximum force

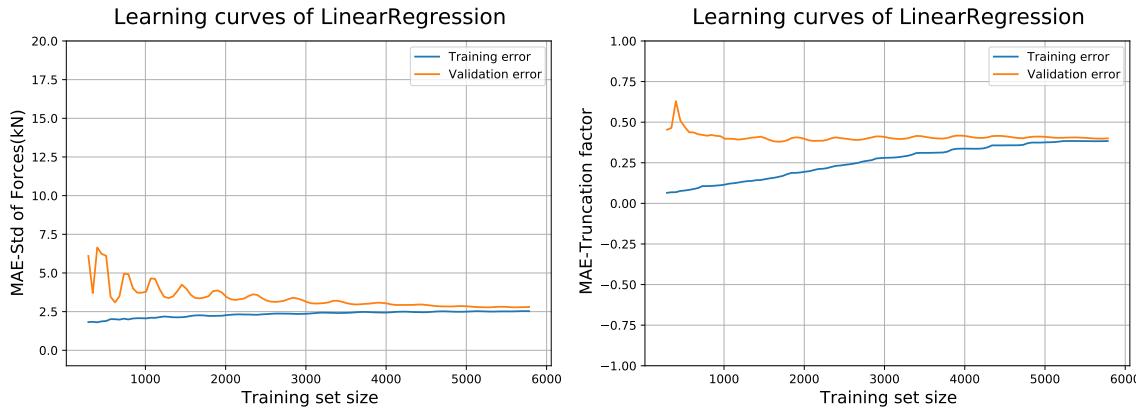


Figure 33: Learning curve of linear regressor for Standard deviation and Truncation factor

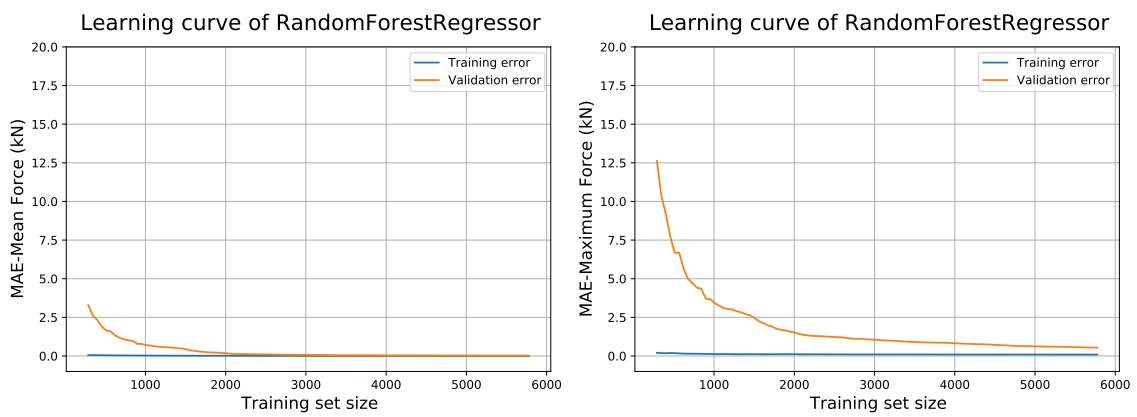


Figure 34: Learning curve of Random Forest for Mean and Maximum force

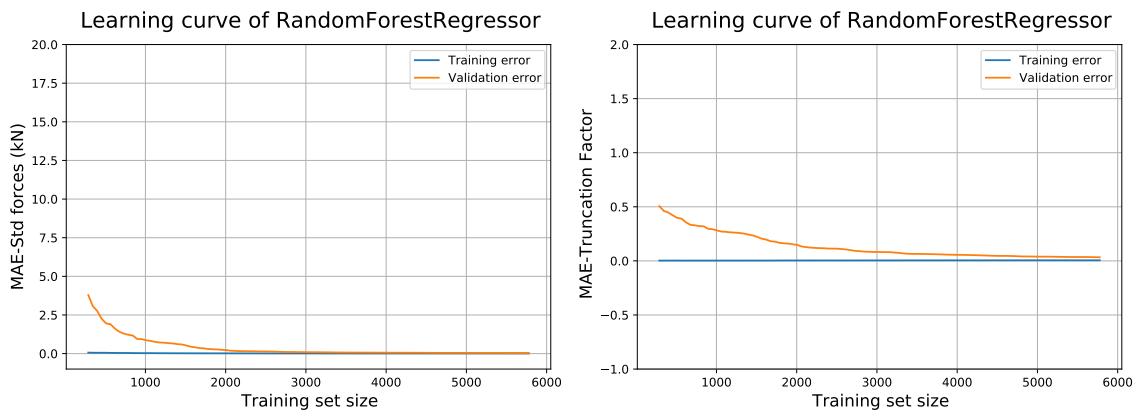


Figure 35: Learning curve of Random Forest for Standard deviation and Truncation factor

Declaration

I hereby declare that this research thesis is my own work and has not been presented anywhere else. Efforts are made to acknowledge and provide references to the work derived from other researchers for the literature.

Rostock, November 04, 2020



Sai dev Lakkakula