Chapter 5

Artificial Neural Networks for EHL Film Thickness Predictions

5.1 Introduction

Tribodynamic modelling generally employs analytical equations for the prediction of film thickness in elastohydrodynamic contacts; chosen due to their timely solution. Whilst computationally efficient, these do not achieve the accuracy of the full numerical solution outside the bounds of the data used to generate the analytical equations. In the context of dynamic simulation, a full numerical solution at each time step of a system level model would, however, yield excessive computation time. This has led to the emerging use of data driven solutions, such as machine learning, in the field of tribology. These can achieve accuracy much closer to the numerical solution, whilst significantly improving computational time.

This chapter details the development of an Artificial Neural Network (ANN) for prediction of central film thickness at the roller-race conjunction. ANNs are trained using data generated by the numerical EHL solution, with the data set constrained to realistic operating conditions using the Greenwood regimes of lubrication. Multiple ANNs are compared to find the optimum structure, accounting for training time and accuracy. The ANN is then deployed explicitly, using the boundary conditions of a simple bearing model to test the film thickness accuracy and speed of solution. The trained ANN is then deployed implicitly in the system level FMBD introduced in Chapter 4, replacing the analytical film equations in the model.

The aim of this chapter is to improve the accuracy of the central film thickness estimation, while maintaining a timely solution in the context of a full dynamic solution. This workflow employed is not only relevant to roller bearings; it can be applied to a

wide range of contacts and different sources of training data depending on the modelling requirements.

5.2 Numerical vs Analytical Film Thickness Estimations at High Entrainment Velocities

Two main approaches exist for determination of the complex non-linear problem of film thickness in lubricated contacts. The first approach involves employing numerical methods [49], where systems of partial differential equations are formulated to describe the state of the contact and then solved iteratively [36]. While this method yields accurate results and is applicable to a wide range of operating conditions, it is computationally intensive due to its iterative nature. The second approach involves developing regressed analytical equations from experimental or numerical studies which can be used for specific lubrication regimes. These equations offer quick estimates of key parameters, such as central [52] and minimum film thickness [132]. However, whilst more computationally efficient than the full numerical solution, this approach has limitations.

The applicability of regressed equations is often limited to the range of data used for their development. There is also a requirement for extensive effort in collecting experimental or numerical data to develop them. The entrainment velocities considered in this work (up to $32\ m/s$) exceed the typical range over which the regressed equations are experimentally derived. Whilst it is possible to exceed the range of input data, it must be done with caution [98].

Figure 5.1 shows a comparison between the central film thickness calculated using the numerical method (see Section 3.3), and the analytical equation (Equation 3.15) across a speed range of 1 000 - 21 000 rpm. The bearing used for this comparison is the same as in Section 4.3. Geometry is detailed in Table 4.1, with rheological and material properties detailed in Table 4.2.

It is shown that as rotational speed and hence entrainment velocity increase, the film thickness prediction of the numerical and analytical calculations diverge. At 21 000 rpm, entrainment velocities of 30.7 m/s leads to a 22.6% difference between the methods, with the analytical equation overestimating the film thickness.

The implementation of ANNs within tribology is one way to overcome the computational expense of the full numerical solution and this limited validity of the analytical approach. Overcoming the above stated discrepancy is not the only advantage of a numerical EHL informed ANN. Since an ANN can be trained using a wide range of

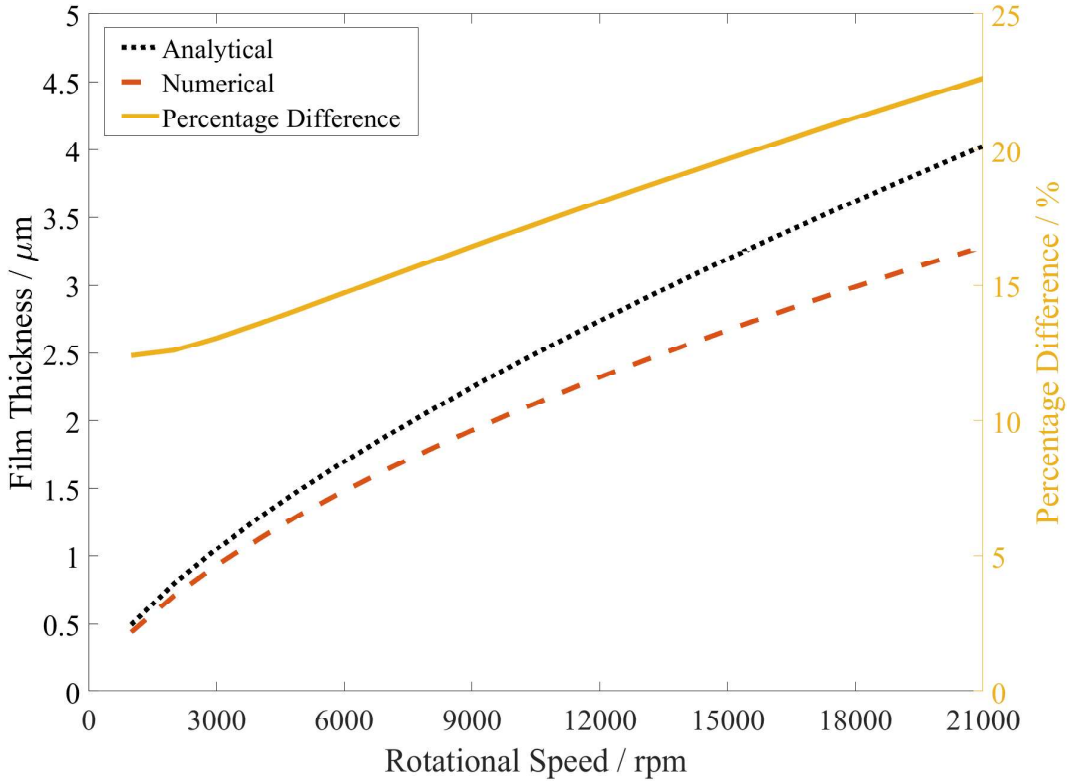

Figure 5.1 Central film thickness - EHL vs analytical.

input data, the effects of other tribological phenomena such as starvation and thermal effects can also be considered. Furthermore, the training of such ANNs is not bound to only numerical inputs; the results of experimental testing could be used to generate an experimentally validated ANN.

5.3 ANN Fundamentals

An Artifical Neural Network (ANN) is a computational model that is inspired by the biological neural networks present in the natural brain [133]. ANNs are a subset of machine learning (ML) that can be trained using supervised, unsupervised, or reinforcement learning techniques. In supervised learning, ANNs are particularly useful for regression tasks, where they can model complex non-linear relationships between inputs and outputs. ANNs compare their outputs with target values during training and, due to their structure, can adapted for a wide range of applications. The goal of this training is to minimize the error, and to improve the ability of the network to generalize and make accurate predictions for new, unseen data.

ANNs consist of a set of interconnected processing elements known as neurons. These are represented computationally as nodes, and the terms are often used inter-

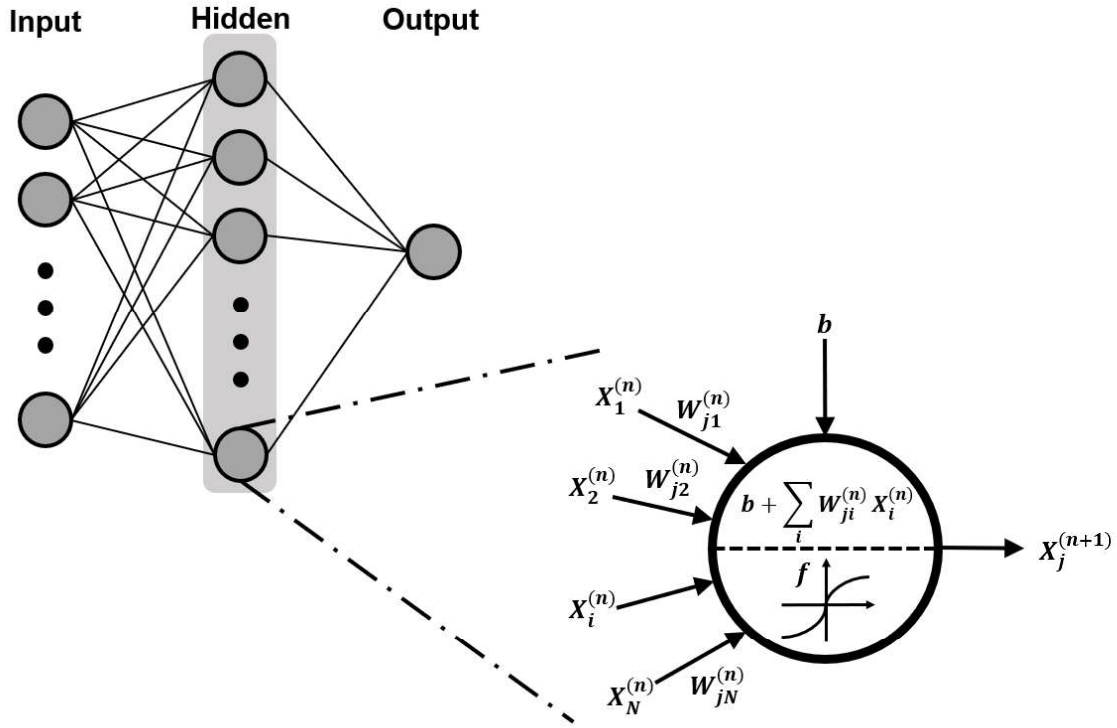

Figure 5.2 ANN schematic

changeably. These elements have the ability to adapt to input data for the purpose of solving complex non-linear functions. The neurons are organized into three main layers, shown in Figure 5.2: Input layer; Hidden layer(s) and Output layer. The adaptation is performed using weighted connections that link each neuron layer. These weightings are adjusted during the learning process, and determine the strength of the connections between neurons.

The following section provides descriptions of the common terms that will be referenced throughout this chapter.

5.3.1 Network Architecture

- **Input layer**: Input data is assigned to input nodes in the first layer of the network. In the case of the film thickness estimation, nine input variables and hence nine nodes are required.

- **Hidden layer(s)**: These layers are located between the input and output layer. There can be multiple hidden layers, each consisting of multiple nodes. Multi-layer networks enable the resolution of non-linear problems, whereas single layer networks (no hidden layers) are limited to linear problems [134].

- **Output layer**: The output layer represents the final computed results. For the film thickness estimation, the output consists of a single node corresponding to the predicted central film thickness.

The type of ANN used in this study is called a multi-layer feedforward backpropogation neural network. Forward propogation through the network is used to make predictions based on the input data. Backpropogation is then performed to calculate prediction error and modify the network to minimize this. This process is repeated until the desired prediction accuracy is achieved.

5.3.2 Forward Propogation

Input data propagates through the neural network in the following manner:

1. **Weights**: The input data to the nodes is multiplied by corresponding weights, and a bias term is added (see Figure 5.2). The general equation for ANNs is:

$$z = f(b + \sum_i W_{ji}^{(n)} X_i^{(n)}) \tag{5.1}$$

   where z represents the activated output of the neuron. W_{ji} is the weight connecting the i-th neuron of the previous layer to the j-th neuron of the current layer in the n-th layer. $X_i^{(n)}$ is the input value to the neuron from the i-th neuron in the previous layer. The bias is represented by b, and f represents the activation function applied to the weighted sum.

2. **Activation function**: The weighted sum ($b + \sum_i W_{ji}^{(n)} X_i^{(n)}$) is passed through an activation function, f, which introduces non-linearity to the system to enable the learning of complex patterns.

3. **Bias**: The bias term, b, is able to shift the activation function's output. It ensures that a neuron can still activate even in the case where all input values are zero.

4. **Output generation**: Data is propagated through the system until it reaches the final prediction at the output layer.

5.3.3 Backpropagation

The learning process of an ANN is achieved by adjusting the weights in the network. This is done using a process called backpropagation [135]:

1. **Loss function**: During training, the target output of the ANN is known. The predicted output of the ANN is therefore measured against this using a loss function. In the case of ANNs used for regression, mean squared error (MSE) is used.

2. **Backpropogation**: This error is backpropogated through the system to determine the contribution of each weight to that error.

3. **Optimization**: Optimization algorithms, such as Levenberg-Marquardt [136] [137] [138], are used to update the weights and biases to reduce the loss.

4. **Epochs**: The above process repeats over multiple cycles, known as epochs. This is performed until the desired MSE is achieved.

5. **Overfitting**: This is the phenomenon whereby an ANN becomes too specialised at learning the training data, and as a result performs poorly with new, unseen data. This occurs when the network extensively adjusts its internal parameters to fit noise or outliers in the training set [140]. It is therefore necessary to limit the number of epochs once sufficient performance is achieved.

5.4 Methodology

This section details the following methodology:

1. Generating training data for the ANN using the numerical EHL method, and constraining this input data to a range valid for machine element contacts.

2. Evaluating the best ANN structure for the central film thickness estimation.

3. Testing the ANN using by calculating bearing film thickness explicitly based on kinematic condition obtained from a dynamic bearing model.

4. Embedding the ANN within a FMBD model to calculate and implicitly consider the film thickness within the bearing at each time step of the simulation.

The workflow describing the EHL data generation, variable constraints, training methodology and structure evaluation is presented in Figure 5.3. This workflow resulted in a structurally optimised trained ANN that could be used for the explicit and implicit modelling tests.

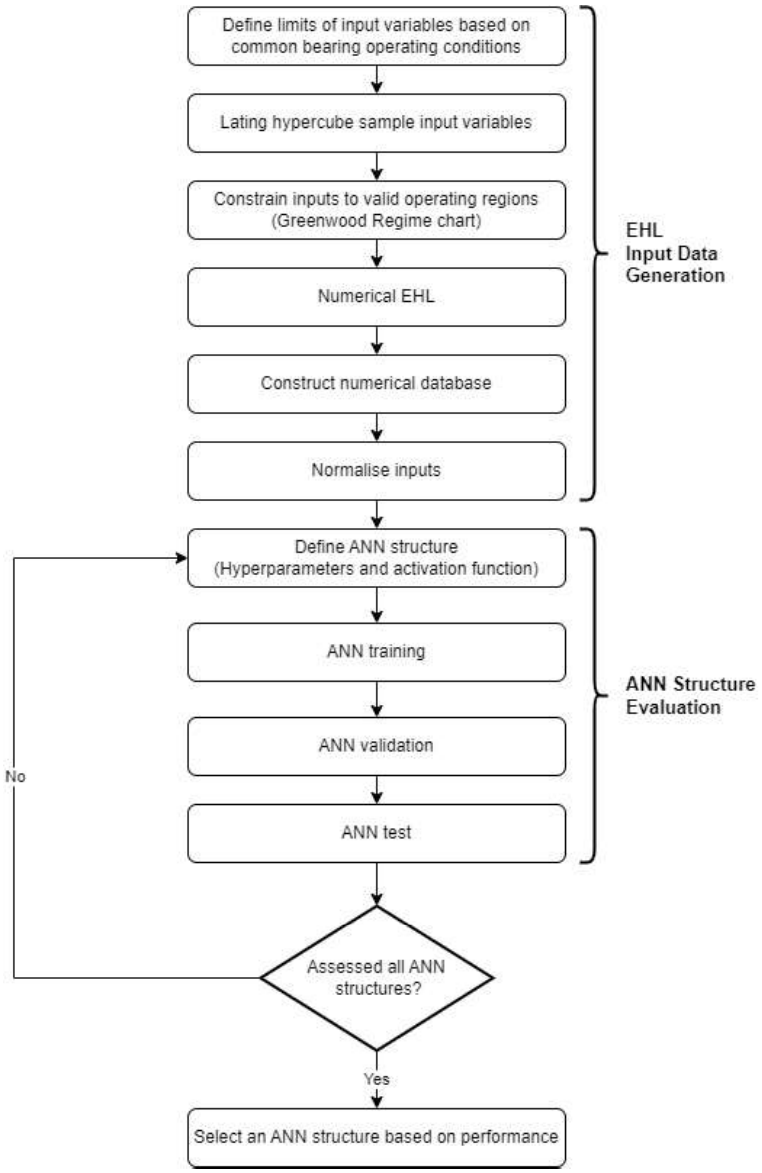

Figure 5.3 ANN data flow, models and training flowchart

5.4.1 EHL Input Data Generation

Training an ANN requires an often comprehensive data set. For this study, the training data was generated using the numerical EHL model presented in 3.3. This approach was selected due to the large size of the dataset required for training, and the relatively low resource-intensive nature to generate this. It is important to note that the training data could also be obtained from experimental work, which would further enhance the applicability of this approach for future studies.

Sampling Input Variables

Due to the large design space covered by the high number and range of input parameters, a robust sampling technique must be chosen to create the training data set. In traditional random sampling, each of the parameters is randomly sampled within its defined range. This may lead to insufficient coverage of the parameter space and simple bias, as it lacks a systematic approach to ensure even distribution [144].

The Latin Hypercube Sampling (LHS) method was utilised by Marian et al. [92], and was chosen for this study. It is a statistical method used to efficiently sample a high-dimensional parameter space, such as that required for the central film thickness calculation. LHS is derived from Latin Hypercube Design (LHD), where each parameter's range is divided into equal intervals along each dimension. Each interval is then randomly assigned to a unique position within its corresponding dimension. The process results in a matrix, where each row represents a combination of parameter values. Contrary to the random sampling method, LHS ensures that each interval is sampled exactly once per dimension, preventing clustering and improving representation across the space [144]. This ensures lower computational effort required for ANN training, despite the high number of input variables and value ranges.

The LHD is a $n_\mathrm{s} \times n_\mathrm{f}$ matrix, where n_s and n_f represent the number of simulations the number of factors respectively. LHS enhances LHD by introducing a randomization component. The randomly selected samples within each interval undergo permutation, ensuring that samples are not biased by the order of selection.

LHS elements are generated by subtracting a random number between zero and one $Z_\mathrm{r} \in [0,1]$ from each LHD element $x_{ij,\mathrm{LHD}}$. This is then divided by the number of test points [145]:

$$x_{ij,\mathrm{LHS}} = \frac{x_{ij,\mathrm{LHD}} - Z_\mathrm{r}}{n_\mathrm{s}} \tag{5.2}$$

This equation rescales the LHD values to a range between 0 and 1. By subtracting a random number between 0 and 1 and dividing by the total number of sample points, the

resulting Latin hypercube samples are spread evenly across the interval (0,1) for each parameter. This is important, because it allows the Latin hypercube samples to be easily transformed to any desired range or distribution. This transformation to the design space is done using the limits of the tribological parameters in Table 5.2.

The quality of the test field (freedom of correlation and uniform distribution) can be assessed based on the distances between data points [146]. The MaxiMin criterion in the MATLAB® Statistics and Machine Learning toolbox was used to optimise the LHS. This maximises the the minimum distance between individual test points such that the LHS test field is uniformly distributed:

$$\text{MaxiMin} = \left[\sum_{1 \leq i < j \leq n_1} d\left(x_i, x_j\right)^{-\xi} \right]^{-\frac{1}{\xi}} \tag{5.3}$$

where d represents all distances in the test field, and subscripts i and j are indexes for the parameter and sample point respectively. ξ represents the application dependant factor which determines the degree of importance assigned to the distances [145].

Constraining the Input Data Bounds

The performance of ANNs is heavily reliant upon the quality of the data set provided for training. To construct a training database, Marian et al. [92] utilised a Finite Element Method (FEM) solver for film thickness calculations. The database covered a very large range of lubricant and material properties for relatively low entrainment speed conditions ($< 0.4\ m/s$ for the 2D line contact studies). Contact conditions for some combinations of these input parameters exceed realistic conditions within common machine elements, including bearings. To further improve upon this methodology, the input data range required constraining.

The Greenwood Regime chart [147] was used for this purpose. The regions of the chart, as shown in Figure 5.4, are:

- Isoviscous Rigid (IR)
- Isoviscous Elastic (IE)
- Piezoviscous Rigid (PR)
- Piezoviscous Elastic (PE)

The bounds indicate the transition between the lubrication regimes, which are classified based on material, rheological and geometric properties. To find which

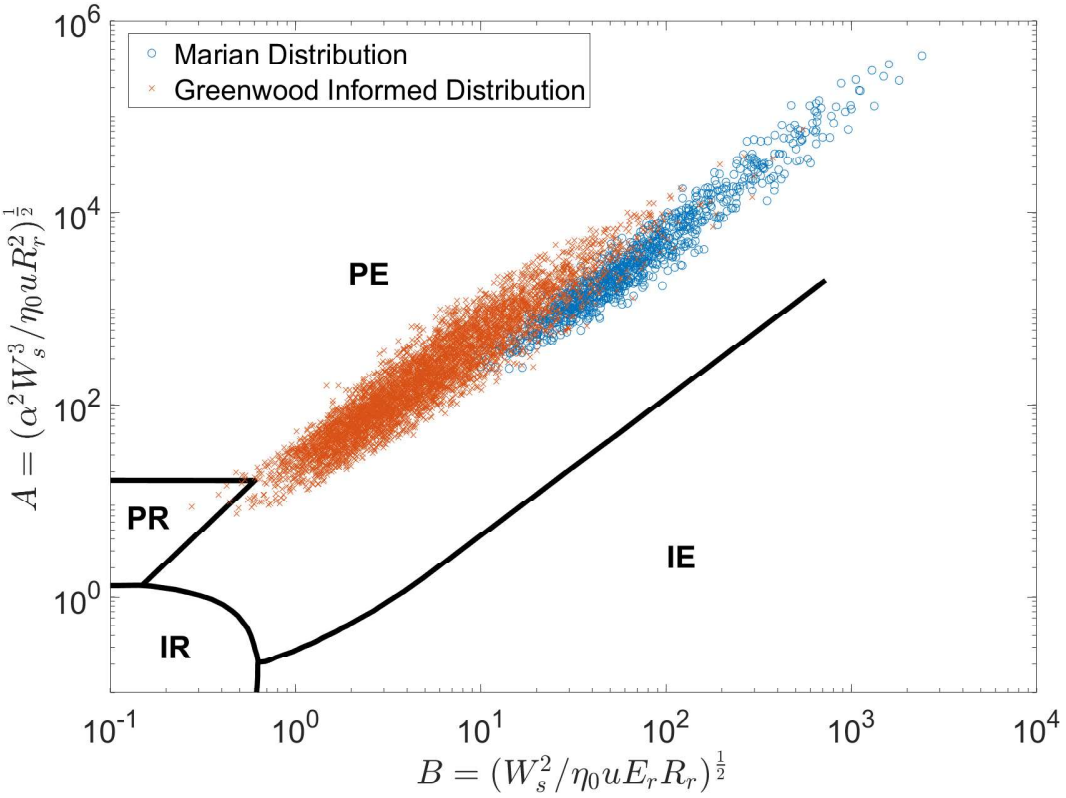

Figure 5.4 Greenwood informed training data vs Marian et al. [92].

regime a contact is operating within, the dimensionless elasticity (G_e) and viscosity (G_v) parameters can be calculated:

$$G_e = \left(\frac{\alpha^2 W_i^3}{\eta_0 u R_r^2}\right)^{\frac{1}{2}} \tag{5.4}$$

$$G_v = \left(\frac{W_i^2}{\eta_0 u E_r R_r}\right)^{\frac{1}{2}} \tag{5.5}$$

The PE region signifies an EHL contact, where pressures are high enough to elastically deform the material and increase the viscosity of the lubricant. The IR region relates to the hydrodynamic regime of lubrication, where the contact load does not deform the surface and viscosity remains constant. Since these investigations are focussed on improving the EHL film thickness estimation, the training data set was required to fall within the PE region of the Greenwood plot.

The initial range of each parameter is shown in Table 5.1. A training data set was then generated using these limits. The input data was then constrained further to ensure Hertzian pressures, P, were between 300 MPa and 3.5 GPa, as well as redistributing any

Table 5.1 Range of ANN film thickness calculation parameters

Parameter	Unit	Minimum	Maximum
Load	N	150	5000
Entrainment Velocity	m/s	0.6	30
Reduced Radius	m	0.0001	0.02
Reduced Elastic Modulus	GPa	200	250
Pressure-Viscosity Coefficient	GPa^{-1}	10	30
Lubricant Viscosity	$Pa.s$	0.0005	0.1
Lubricant Density	kg/m^3	7750	8050
Poisson's Ratio	–	0.3	0.35
Contact Length	m	0.001	0.050

points that fell outside of the PE and PR regions. A flowchart to explain the process of constraining the input variables is shown in Figure 5.5.

A comparison of an the unconstrained and constrained input variables used for the training data is shown in Figure 5.6. It is shown that for the same number of data points (5 000), the constrained data cloud is concentrated over a smaller region of the chart. This improved the point density in regions of interest, increasing the likelihood that the training data more closely matches the test data.

Constructing the Numerical Database

The 1D EHL model presented in Section 3.3 was used to generate the numerical database for training the ANN. The variables corresponding to each constrained data point in Figure 5.6 were used as inputs to the calculation. The target output of central film thickness was calculated.

5.4.2 ANN Structure Evaluation

The general structure of the ANN is described in the following format, as per [80]:

$$N_{in} - [N_{h1} - N_{h2} - N_{h3}]_t - N_{\text{out}} \tag{5.6}$$

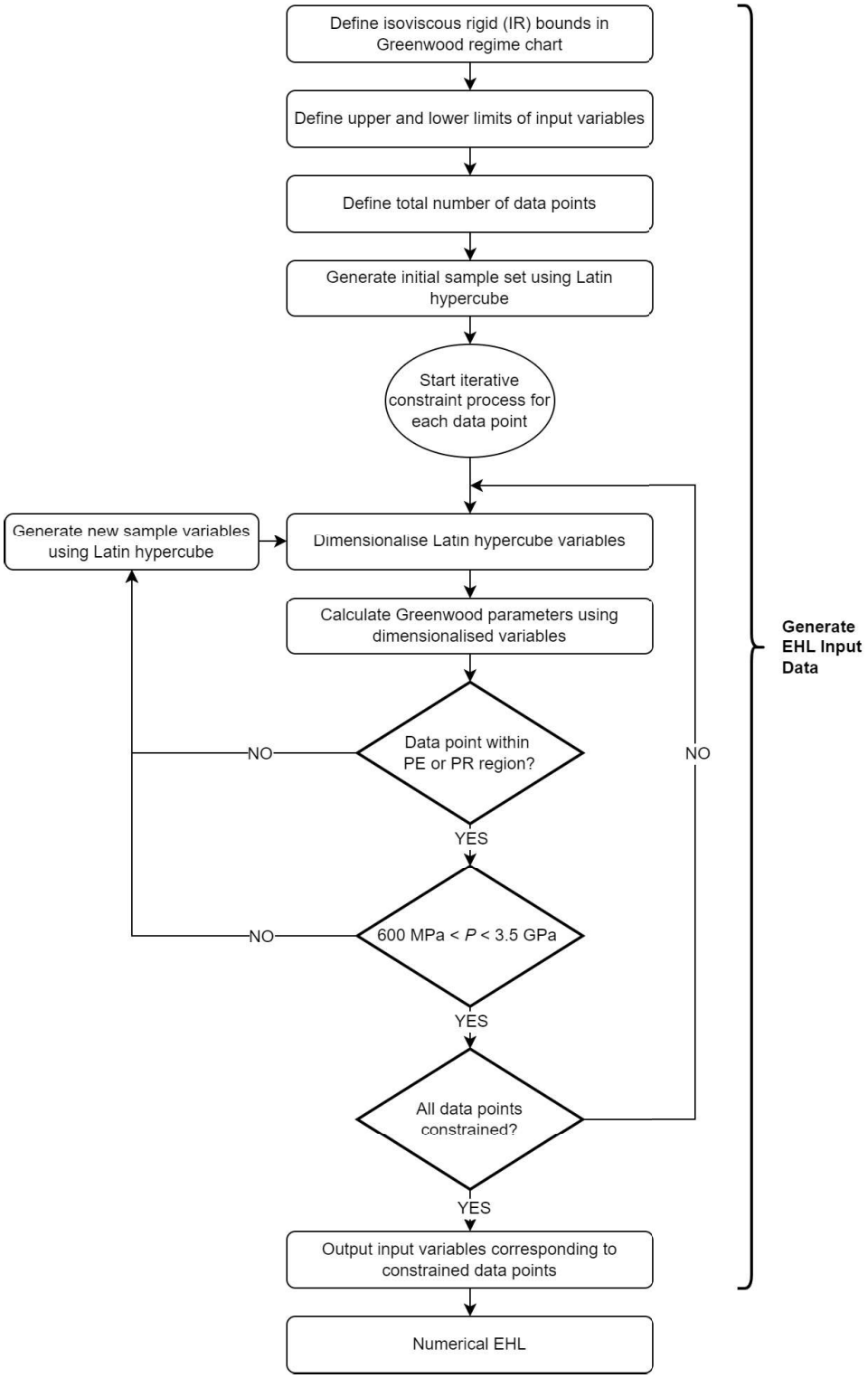

Figure 5.5 Workflow to constrain training input data using Greenwood regimes

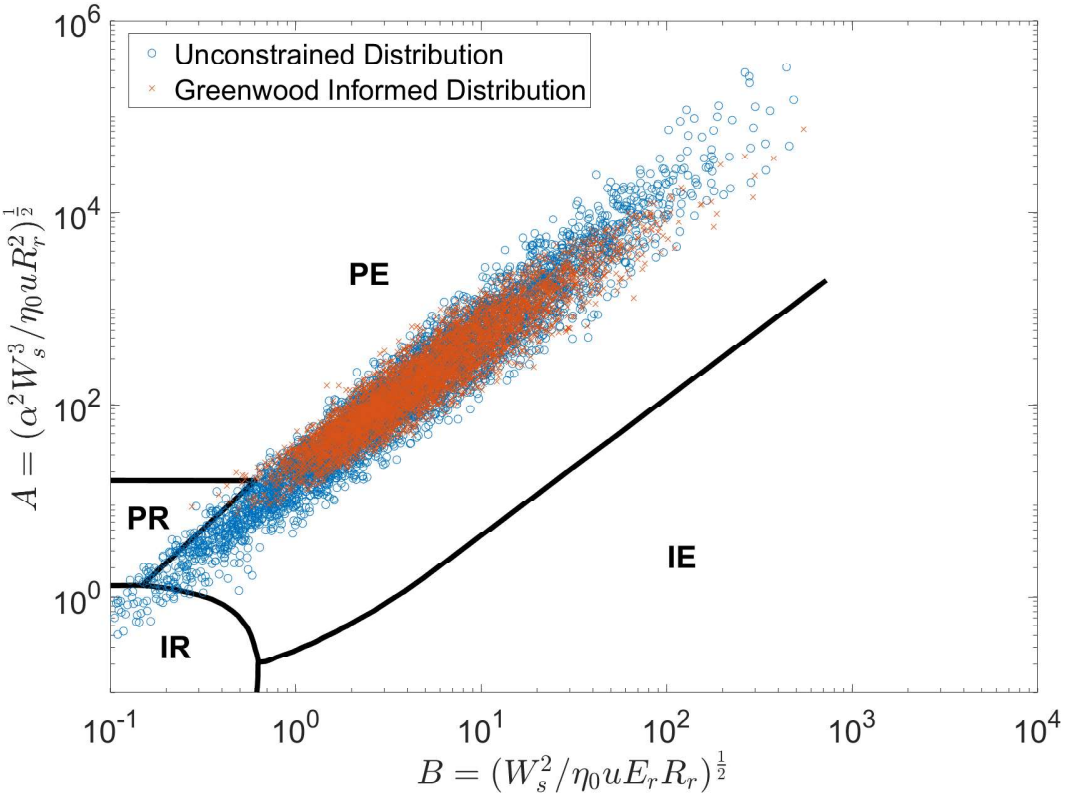

Figure 5.6 Greenwood informed training data vs unconstrained data.

The number of neurons in each layer is denoted by N, with the input and output layers indicated by the subscripts *in* and *out*, respectively. Subscripts $h1$, $h2$, and $h3$ denote the hidden layers, with t being the total number of hidden layers. A graphic representation of the structure used for the film thickness estimations is show in Figure 5.7. As the structural complexity of ANNs increases, the training time increases due to the greater number of neurons and layers. Implementations of ANN in the field of tribology, specifically film thickness predictions, are typically limited to between one and three hidden layers [139].

The structure of an ANN affects both its training time and prediction accuracy. To evaluate the performance of different ANN structures, and hence select an appropriate structure for this application, a sensitivity study was performed. The study comprised of over 500 different ANN structures. The input data range remained constant across all structures, while the variables listed in Table 5.2 were adjusted. This involved varying the hyperparameters: the number of hidden layers varied from one to four, and the number of neurons from 10 to 20. Three activation functions: Hyperbolic tangent, Logistic sigmoid and Rectilinear were evaluated. The wall time for each training data point generation was recorded, as well as the total training time of each ANN structure.

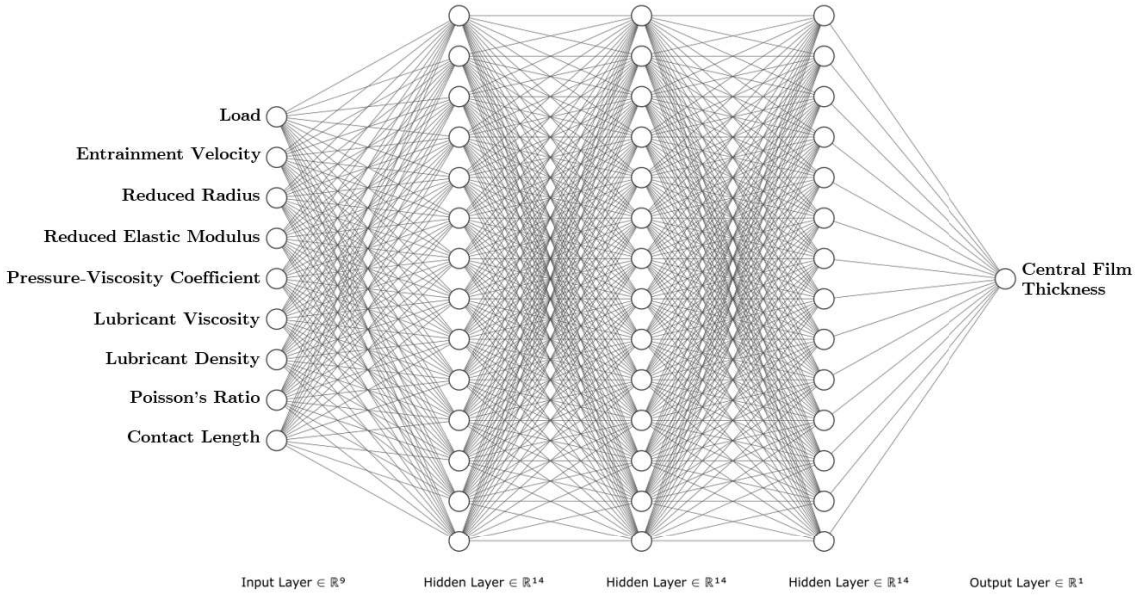

Figure 5.7 ANN structure to predict EHL central film thickness $(9-[14-14-14]_3-1)$

Table 5.2 Sensitivity study of ANN structure

Variable	**Value**
Number of training data points	600 - 1500
Number of hidden layers, t	1 - 4
Number of neurons, N	10 - 20
Activation function type	Hyperbolic tangent (Tanh), Logistic sigmoid (LogSig), Rectilinear (ReLU)

Selection of the final structure to be used for film thickness estimation in the bearing models was based on total training time, coefficient of determination (R^2), and the potential for the ANN to overfit. R^2 (Equation 5.12) was evaluated for the test data sets, since this gives an accurate indication of the networks ability to predict film thickness with unseen input variables.

The following sections describe the evaluation process of each network structure.

Training, Validation and Test Datasets

The dataset was first divided into three sets: the training set, the validation set, and the test set, each containing 70 %, 15 % and 15 % of the training data respectively:

1. **Training Set**: The training set is the portion of the dataset used to train the ANN, containing the input data and the corresponding output data. As aforementioned, the ANN adjusts the internal parameters based on this data to learn the underlying patterns.

2. **Validation Set**: The validation set is used to tune the performance of the ANN during the training process. It is an independent dataset that the network has not seen before, allowing for the evaluation of its generalization capabilities. The network's performance on the validation set is monitored during training to make decisions on adjusting hyperparameters (number of hidden layers, neurons per hidden layer, activation functions), or stopping the training process to prevent overfitting.

3. **Test Set**: The test set is a completely independent dataset that is not used during training or validation. It is used to assess the final performance and generalization ability of the trained ANN. By evaluating the network on unseen data, the test set provides an unbiased estimate of the model's performance in actual use.

For this sensitivity study, the size of the training data set was varied (600, 1000, 2000 and 5000) to observe the its effect on the quality of the predictions. A limit of 1000 epochs was also implemented, restricting the ANN to 1000 full iterations through the entire training set. This was found to be sufficient to improve accuracy while preventing overfitting.

Data Normalization

The input and target parameters were normalized using the min-max normalisation function:

$$\tilde{x} = \frac{x - x_{\min}}{x_{\max} - x_{\min}}(u - l) + u \tag{5.7}$$

where u, and l represent the upper and lower normalized unit values of 1 and -1 respectively. The dimensional target input value is denoted by x, and the final normalised input or output parameter of the ANN is denoted by $\tilde{x}$. $x_{\max}$ and $x_{\min}$ are retained to dimensionalise the output variable after the prediction.

Activation Functions

As suggested in [92], four suitable activation functions for the hidden layers were selected for testing. These are mathematical functions that are applied to the output of

each neuron in a layer of the neural network. They introduce non-linearity which allows the network to learn complex input-output relationships. Activation functions help determine the output of a neuron based on the weighted sum of its inputs. A description of each function is provided below:

- **Sigmoid (logistic)**: This function transforms the input values into a range between 0 and 1. It has continuously differentiable smooth S-shaped curve and is given by the following formula [141]:

$$\log\mathrm{sig}(x) = \frac{1}{1+e^{-x}} \tag{5.8}$$

  Sigmoid functions may suffer from the "vanishing gradient" problem where the partial derivative reaches zero [142], leading to slower convergence during training.

- **ReLU (Rectified Linear Unit)**: This function outputs the input value directly if it is positive, and zero otherwise. The mathematical definition is:

$$\mathrm{ReLU} = \begin{cases} x, & x \geq 0 \\ 0, & x \leq 0 \end{cases} \tag{5.9}$$

  The gradient is 1 when the neuron is activated, and zero when it is deactivated. This function is computationally efficient and addresses the vanishing gradient problem to an extent [142].

- **Tanh (Hyperbolic Tangent)**: The hyperbolic tangent or tanh function is defined as:

$$\tanh(x) = \frac{2}{1+e^{-2x}} - 1 \tag{5.10}$$

  The formulation and behaviour is very similar to sigmoid. It produces values which range from -1 to 1, having a centred mean around zero.

- **Linear**: A simple linear activation was used on the output.

Evaluating the Network Performance

During backpropagation of the ANN, the Mean Squared Error (MSE) was used to evaluate the network's performance:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(t_i - y_i)^2 \tag{5.11}$$

where t_i and y_i are the target and predicted value respectively. The total number of training points being trained, validated or tested is denoted by N.

To assess the goodness of fit of the ANN, the statistical metric R^2, known as the coefficient of determination, was used. This measures the proportion of variance in the dependant variable (film thickness) that is predictable from the input variables (Table 5.1) in the model. This value ranges from 0 to 1, with a higher value indicating the best fit of the model to the data. This was post-processed after training and is calculated as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(t_i - y_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} \tag{5.12}$$

where $\overline{y}$ is the mean of the target sample. The numerator of the fraction, $\sum_{i=1}^{N}(t_i - y_i)^2$, represents the sum of squared residuals, which quantifies the variation in the target variable that is not explained by the model. The denominator, $\sum_{i=1}^{N}(y_i - \bar{y})^2$, is the total sum of squares, which captures the total variation in the target variable [92].

Preventing Overfitting

Early stopping and regularisation was used to prevent statistical overfitting during training [143]. Early stopping halts the training process before the model reaches the maximum number of epochs. This is done by monitoring the performance (MSE (Equation 5.11)) of the network against the validation set during training. Once the performance reaches a plateau, or begins to degrade, the training is stopped early. Regularisation adds additional constraints to the learning process. It modifies the performance criteria by accounting for the change in mean square of the network weights and biases (Mean Squared Weight (MSW)). This is calculated in Eq. 5.13:

$$MSW = \frac{1}{N}\sum_{j=1}^{N} w_j^2 \tag{5.13}$$

where w_j is the individual weight value associated with the j-th neuron or connection in the network.

By applying an adjustment factor, denoted as γ', the weights and biases can be reduced during propagation (Eq. 5.14), thus mitigating the risk of overfitting and improving the network's generalization capability.

Table 5.3 Roller Bearing Parameters

Parameter	Value
Inner race diameter	31.5 *mm*
Roller diameter	7.5 *mm*
Roller length	15 *mm*
Number of rollers	12
Radial interference	5 μm
Young's modulus	218 *GPa*
Poisson's ratio	0.33

Table 5.4 Operating Conditions

Parameter	Value
Radial force	6000 *N*
Rotational velocity	10 000 *rpm*

$$MSE_{reg} = \gamma' * MSW + (1 - \gamma') * MSE \tag{5.14}$$

5.4.3 Explicit Bearing Film Thickness Predictions

After identifying a suitable training data size and structure, an ANN was trained to estimate the EHL central film thickness. These results could then be compared to the analytical (Equation 4.11) and numerical (Section 3.3) methods for calculating the film thickness under realistic bearing operating conditions.

The FMBD model used in Chapter 4 was used for this study. The shaft was modelled as a rigid body, and loading in the radial direction was purely static to remove the influence of additional dynamic effects. The shaft was constrained to one rotational and two lateral degrees of freedom. Bearing properties and operating conditions are shown in Table 5.3 and Table 5.4 respectively.

The bearing was modelled as dry, without the influence of the EHL film at the roller-race contacts. The kinematic and dynamic results necessary for the film thickness estimation were extracted from an individual roller at each time step of the simulation.

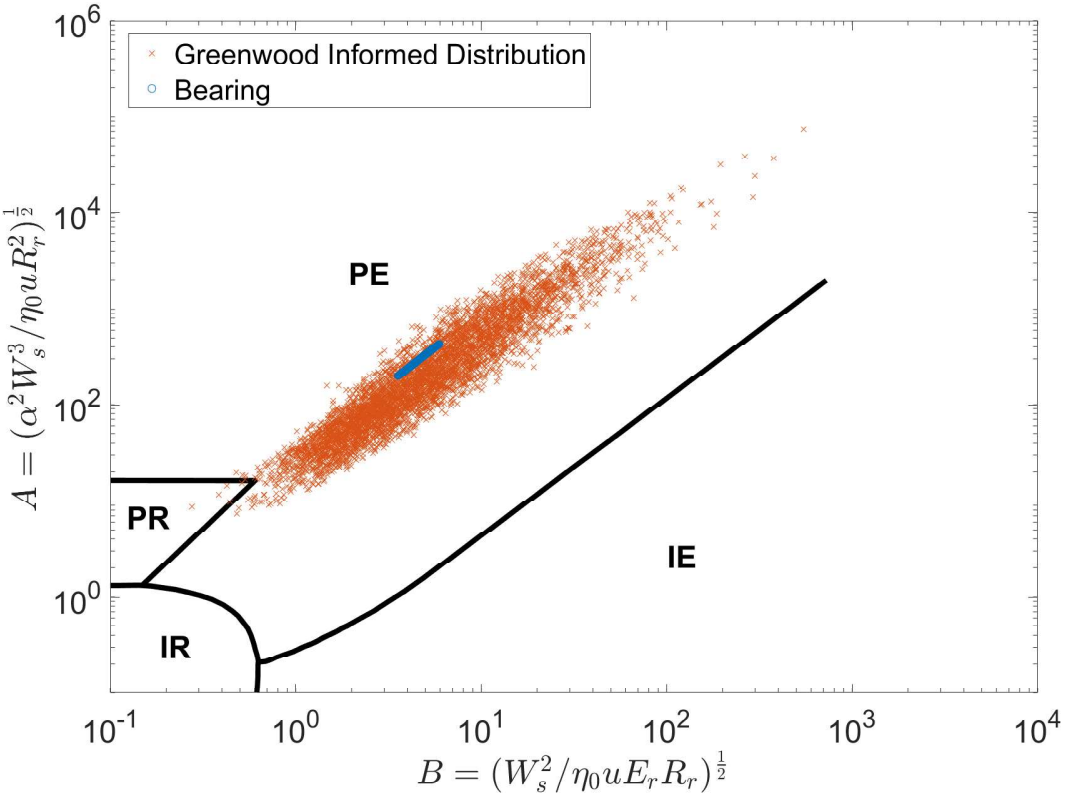

Figure 5.8 Bearing operating conditions vs Greenwood informed training data.

These results include roller load, contact entrainment velocity, and reduced radius of the contact between the roller and inner-race.

The loading pattern is cyclic in nature as the roller enters and exits the most highly loaded region of the bearing, corresponding to the radial force vector applied to the inner race. Sufficient preload ensures constant contact between elements and raceways so that the regime does not deviate from EHL. The contact reduced radius (3.03 mm) and entrainment speed (15.7 m/s) do not change throughout the orbit as they are a function of bearing geometry and constant operating speed. The normal contact load fluctuation is shown in Figure 5.9.

The operating conditions of the bearing were within the range of validity of the training data set. This is demonstrated in Figure 5.8 whereby the Greenwood parameters for the bearing operating points are calculated and overlayed on the training data cloud.

The structure chosen for this study was based on the conclusions drawn in Section 5.5.1. A structure of 3 hidden layers with 14 neurons per layer was selected ($9 - [14 - 14 - 14]_3 - 1$). Logistic sigmoid was selected as the activation function for the hidden layers. This structure is represented graphically in Figure 5.7.

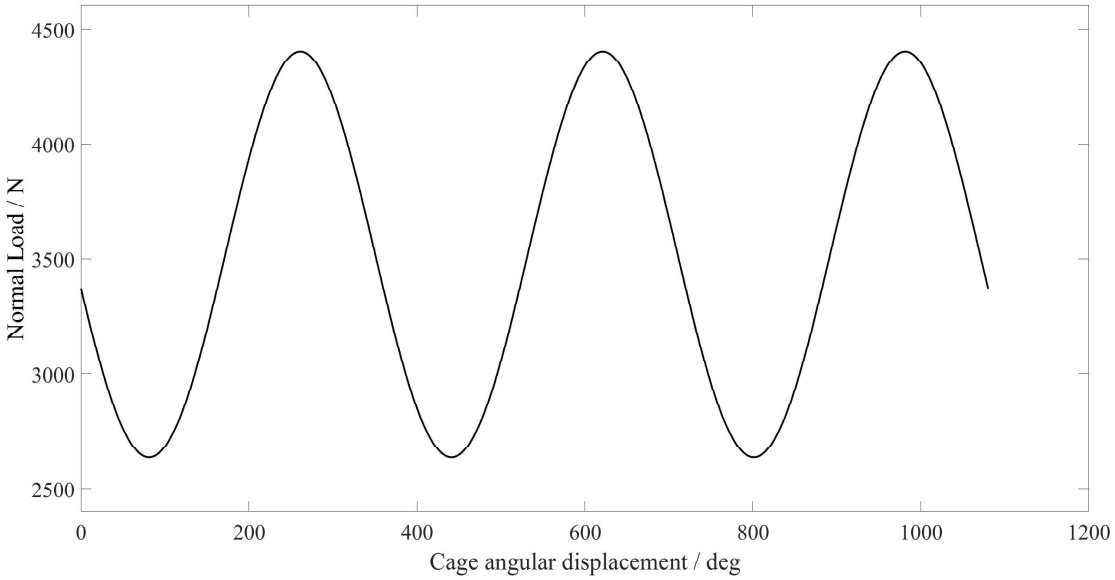

Figure 5.9 Normal contact load vs cage angular displacement

5.4.4 Implicit Bearing Film Thickness Predictions

Same model as in Section 4.3, but with ANN instead of the analytical formulae. TODO.

5.5 Results and Discussion

The following data was obtained using consumer grade hardware with the following specifications: Intel® Core™ i7-9750H CPU 6 cores @ 2.60GHz, 32GB RAM; GPU: NVDIA GeForce GTX 1650. identical hardware was used for both the full numerical and the ANN solutions to provide performance comparisons and assess the suitability of ANNs for film thickness calculations in FMBD solvers.

5.5.1 ANN Structure Evaluation

The 1D EHL model presented in Section 3.3 was used to generate the numerical database for training the ANN. Each numerical solution and hence training point took an average of 5.88 s to compute. The construction of the entire database on a single core therefore has a wall time of between 58.8 *min* and 489 *min* for 600 and 5000 points respectively. This wall time is noted for baseline comparisons, and can be significantly improved if parallelisation across multiple cores is utilised.

Tables 5.5 - 5.7 present the R^2 values obtained from 600 training data points, considering different activation functions described in Section 5.4.2. The number of layers and neurons was varied for each activation function. Among the activation functions

Table 5.5 R^2 performance of ANN structures using 600 data points and a LogSig activation function

Activation Function: LogSig												
R^2		Number of Neurons										
		10	11	12	13	14	15	16	17	18	19	20
Number of Layers	1	0.997	0.995	0.998	0.998	0.999	0.999	0.999	0.998	0.998	0.995	0.999
	2	0.999	0.999	0.999	0.997	1.000	0.996	0.998	0.998	0.997	0.997	0.995
	3	1.000	0.998	0.998	0.999	0.997	0.997	0.997	0.998	0.993	0.995	0.996
	4	0.997	0.999	0.997	0.997	0.999	0.998	0.993	0.989	0.989	0.975	0.990

Key
R^2 [-]
0.990
0.993
0.997
1.000

tested, the rectilinear (ReLU) function consistently underperformed when compared to the logistic sigmoid (LogSig) and hyperbolic tangent (Tanh) functions across all network structures. Similar to the work of Marian et al. [139], the optimum layers was found to be between two and three, however the activation function was shown to be the dominating determinate of R^2 performance.

The LogSig activation function demonstrated the best R^2 performance, even for lower complexity structures. This was therefore selected for each hidden layer to assess the influence of training data quantity R^2 performance. Tables 5.8 - 5.11 present the training times for networks with varying numbers of training points and structural configurations. The results indicate that training time increases with the number of layers and neurons in the network. However, training time exhibits a much stronger positive correlation with the number of data points rather than with the complexity of the network structure.

Figure 5.10 illustrates the influence of the number of training points on the R^2 value when using the LogSig activation function. The results indicate that 600 data points are sufficient to train an ANN to the accuracy required for the central film thickness prediction ($R^2 = 0.99791$). Increasing the dataset to 5 000 data points resulted in an R^2 value of 1, however this came with significant time cost for structures with higher complexity; reaching 600 s for 4 layers of 20 neurons and a total of 1000 epochs. Combined with a data generation time of 489 min, this represents a computationally inefficient trade-off between accuracy and training cost. It is therefore concluded that 1000 - 2000 training data points are sufficient to achieve accurately trained neural networks for predicting the central film thickness while maintaining computational efficiency.

The results of this study guided the decision for the network structure to be used for the bearing models. A structure of 3 hidden layers with 14 neurons per layer was selected $(9 - [14 - 14 - 14]_3 - 1)$. Logistic sigmoid was selected as the activation function for the hidden layers. This was trained using 2000 data points.

Table 5.6 R^2 performance of ANN structures using 600 data points and a Tanh activation function

Activation Function: Tanh												
R^2		Number of Neurons										
		10	11	12	13	14	15	16	17	18	19	20
Number of Layers	1	0.998	0.999	0.995	0.998	0.998	0.994	0.998	0.999	0.998	0.998	0.998
	2	0.999	0.998	0.998	0.999	0.996	0.998	0.996	0.996	0.996	0.993	0.994
	3	0.999	0.992	0.996	0.995	0.992	0.989	0.989	0.985	0.994	0.970	0.984
	4	0.997	0.997	0.992	0.994	0.991	0.983	0.982	0.978	0.990	0.989	0.983

Key
R^2 [-]
0.990
0.993
0.997
1.000

Table 5.7 R^2 performance of ANN structures using 600 data points and a Tanh activation function

Activation Function: ReLU												
R^2		Number of Neurons										
		10	11	12	13	14	15	16	17	18	19	20
Number of Layers	1	0.985	0.991	0.994	0.993	0.993	0.994	0.992	0.994	0.995	0.987	0.990
	2	0.993	0.996	0.995	0.993	0.992	0.984	0.989	0.993	0.992	0.993	0.992
	3	0.986	0.983	0.996	0.988	0.984	0.980	0.992	0.992	0.988	0.986	0.988
	4	0.981	0.990	0.975	0.928	0.951	0.986	0.985	0.986	0.986	0.982	0.985

Key
R^2 [-]
0.990
0.993
0.997
1.000

Table 5.8 Training time of ANN structures with LogSig activation function and 600 data points

600 Data Points, Activation Function: LogSig												
Training Time [s]		Number of Neurons										
		10	11	12	13	14	15	16	17	18	19	20
Number of Layers	1	0.76	0.96	1.16	1.01	1.49	1.30	1.41	1.72	1.12	0.90	1.13
	2	1.01	2.93	1.94	1.43	1.88	1.80	1.56	1.80	1.33	2.51	1.77
	3	2.01	1.29	2.58	2.22	2.27	1.52	1.67	2.26	2.06	1.94	3.57
	4	1.64	5.97	2.21	2.79	6.70	2.66	4.36	2.51	2.76	2.61	2.90

Key
Time [s]
0.00
10.00
100.00
600.00

Table 5.9 Training time of ANN structures with LogSig activation function and 1000 data points

1000 Data Points, Activation Function: LogSig												
Training Time [s]		Number of Neurons										
		10	11	12	13	14	15	16	17	18	19	20
Number of Layers	1	1.00	1.10	1.36	1.11	1.89	2.20	2.09	1.01	1.58	2.83	1.10
	2	3.25	1.66	2.33	2.72	4.00	3.07	2.35	2.81	5.35	1.97	3.77
	3	2.74	6.35	4.76	5.58	4.17	2.36	4.88	7.56	7.18	6.55	5.18
	4	2.59	2.29	8.12	6.47	7.17	3.81	7.70	11.46	19.01	12.14	8.92

Key
Time [s]
0.00
10.00
100.00
600.00

Table 5.10 Training time of ANN structures with LogSig activation function and 2000 data points

2000 Data Points, Activation Function: LogSig												
Training Time [s]		Number of Neurons										
		10	11	12	13	14	15	16	17	18	19	20
Number of Layers	1	0.90	2.47	1.70	1.74	4.38	2.61	2.49	1.81	4.54	2.58	2.67
	2	6.06	7.19	10.66	13.38	6.20	6.54	6.76	14.61	16.32	7.86	20.32
	3	10.72	10.45	4.55	5.19	21.54	20.00	20.00	25.44	27.50	11.00	19.71
	4	11.40	18.79	16.06	6.69	41.55	30.72	28.04	67.86	14.78	62.17	57.13

Key
Time [s]
0.00
10.00
100.00
600.00

Table 5.11 Training time of ANN structures with LogSig activation function and 5000 data points

5000 Data Points, Activation Function: LogSig											
Training Time [s]		Number of Neurons									
		10	11	12	13	14	15	16	17	18	19	20
Number of Layers	1	1.74	4.06	3.12	3.24	3.82	3.78	2.11	11.73	3.45	14.25	12.07
	2	10.86	22.86	37.50	30.99	20.02	19.95	38.09	11.02	48.89	134.95	37.88
	3	14.53	33.38	34.62	116.81	61.86	57.12	140.49	124.75	228.28	71.22	85.55
	4	94.66	36.71	139.61	70.20	13.33	229.45	88.24	155.37	217.48	500.54	600.01

Key
Time [s]
0.00
10.00
100.00
600.00

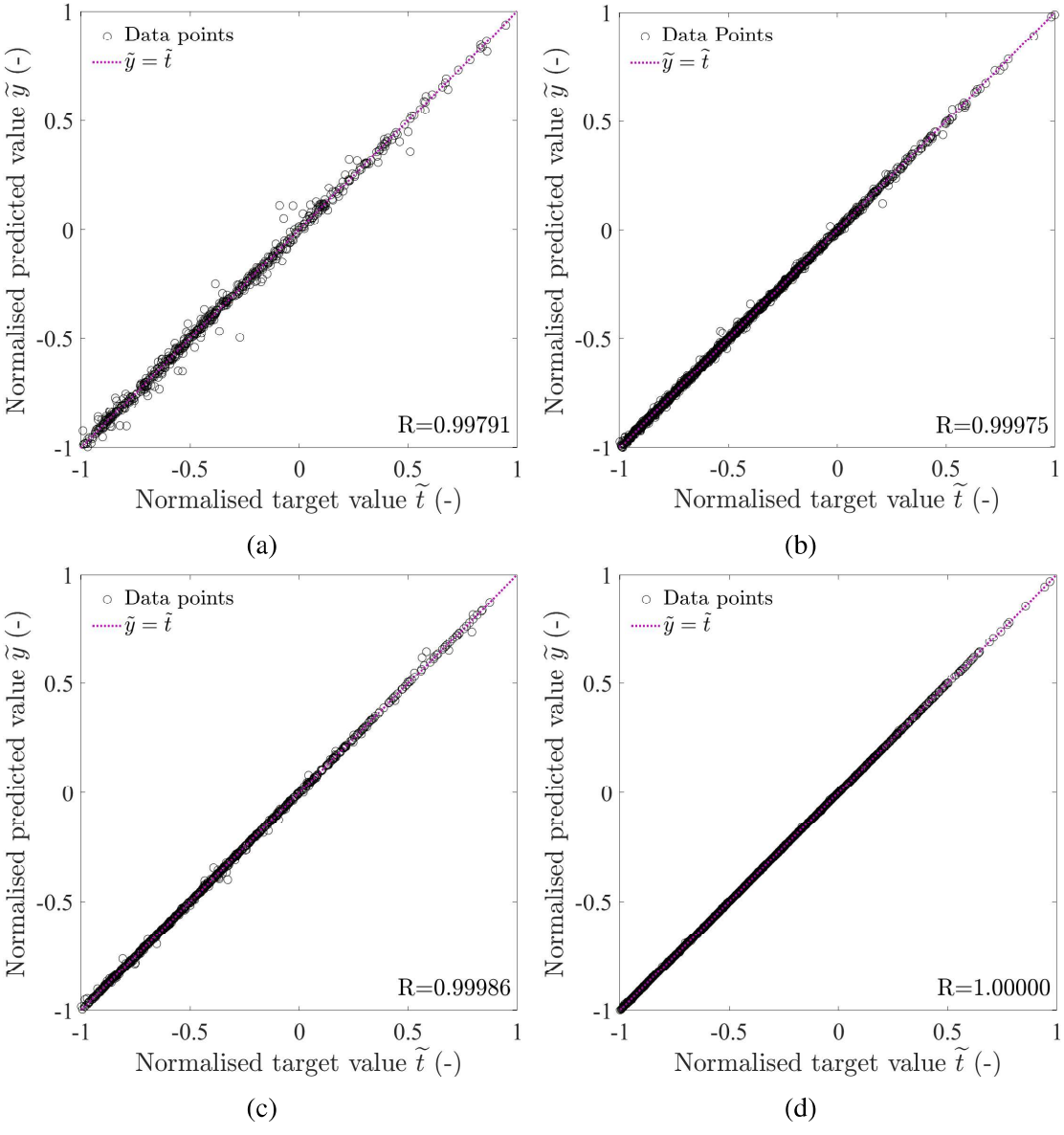

Figure 5.10 R^2 performance of ANN structure (N=14, t=3) using a Logistic Sigmoid activation function at each hidden layer: a) 600 points, b) 1000 points, c) 2000 points, d) 5000 points

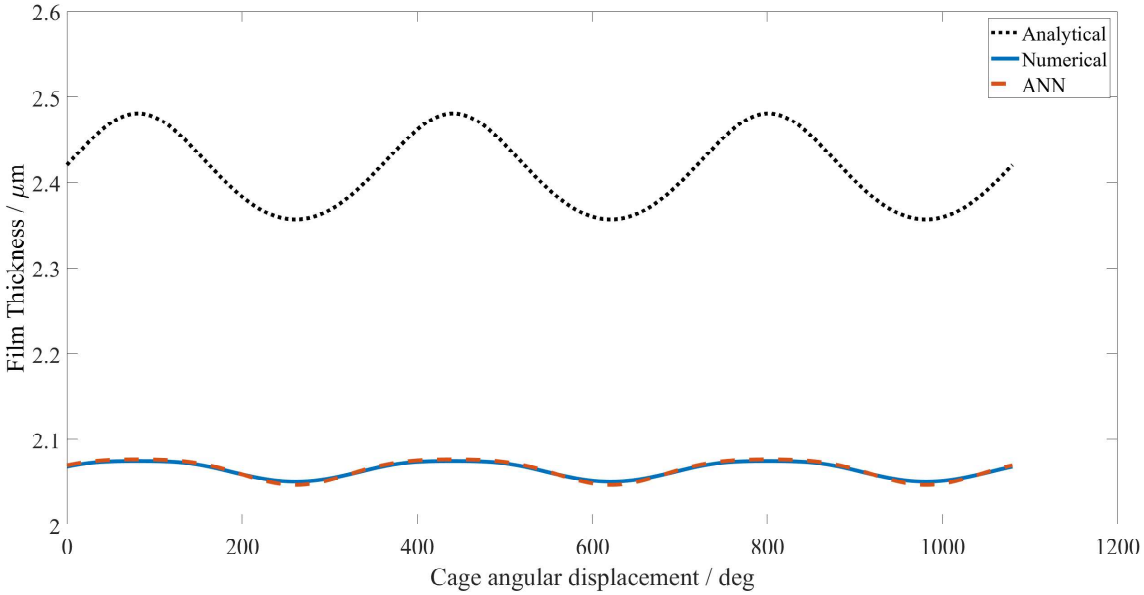

Figure 5.11 ANN, numerical and analytical central film thickness comparisons.

5.5.2 Explicit Bearing Film Thickness Predictions

The performance of the selected ANN was compared the to the computed numerical and analytical results during several operating cycles of the bearing model presented in Section 5.4.3.

Figure 5.11 demonstrates that the ANN prediction of film thickness (orange) matches very closely with the numerical calculation (blue). The fluctuations arise due to the varying contact load, with a peak of 2.076 μm and 2.074 μm for the ANN and numerical model respectively. By comparison, the analytical equation overestimates the peak film thickness to be 2.481 μm.

Table 5.12 demonstrates the relative performance of the ANN for both computation time and MSE fr the analysis in Figure 5.11. The ANN demonstrates a $\sim$1 500 factor computation time reduction in comparison with the ANN, whilst maintaining excellent accuracy as shown by the MSE. The analytical solution is a factor of $\sim$75 faster than the ANN, but is significantly less accurate. As discussed in Section 5.2, this will increase with entrainment velocity and complexity of modelling requirements.

The ANN film thickness prediction also shows excellent agreement across the speed range from 1 000 - 21 000 *rpm*. Figure 5.12 demonstrates the accuracy of the ANN prediction, with an MSE of 3.18 $\times 10^{-4}$ μm^2 across the speed range. This proves the effectiveness of this solution across a broad operating range.

Table 5.12 Film thickness computation methodology performance relative to the numerical solution

Method	Time per point	MSE
	[s]	[μm^2]
Numerical	4.87E+00	-
Analytical	4.43E−05	1.24E−01
ANN	3.10E−03	3.89E−06

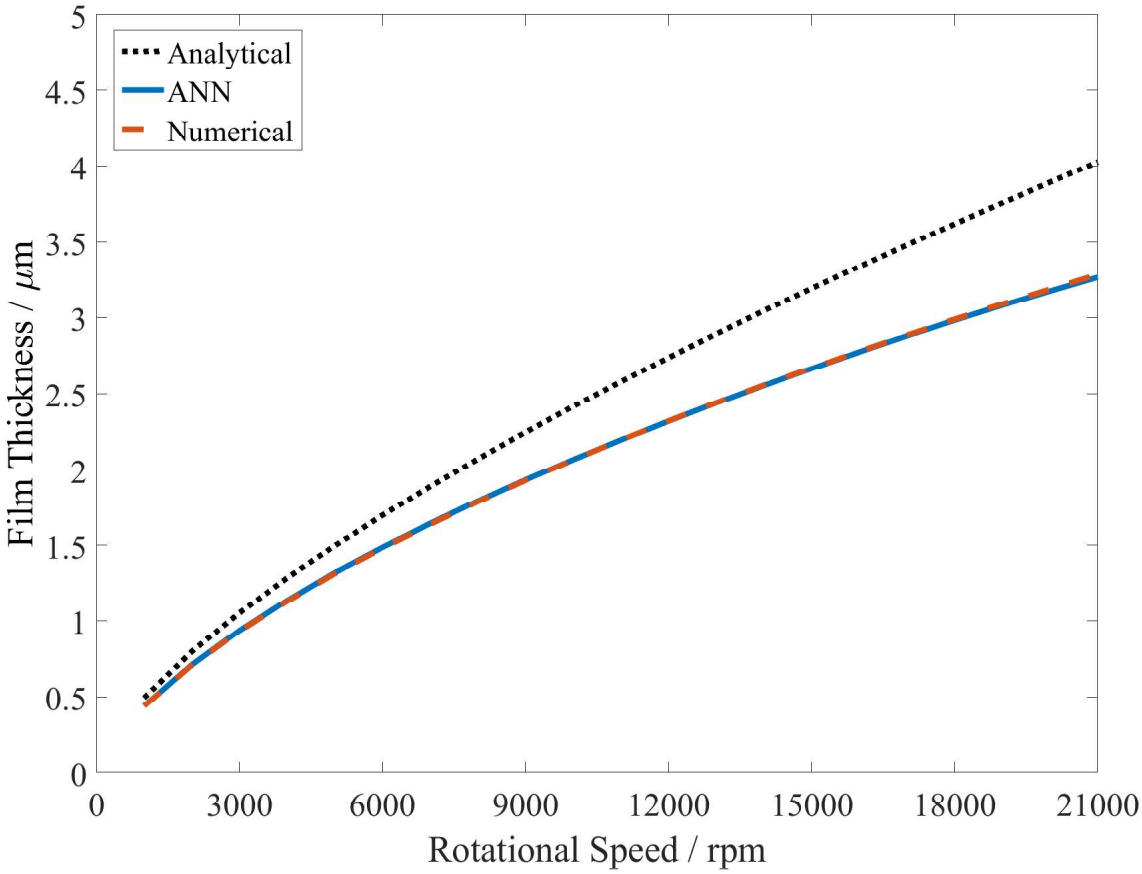

Figure 5.12 Central film thickness - EHL vs ANN vs analytical speed sweep. thickness comparisons at increasing rotational speeds

5.6 Conclusions

A tribological ANN has been trained and employed to calculate the central EHL film thickness at a roller-race conjunction. Different ANN structures were evaluated to find the optimum structure for the film thickness estimation. Numerically generated training data was sampled using Latin Hypercube Sampling (LHS), and constrained to realistic contact conditions using the Greenwood regime. This ensured a high quality dataset; essential for high prediction accuracy. The trained ANN was then used to calculate film thickness in a roller bearing using a simple dynamic bearing model under constant load. The ANN was then employed in an FMBD model, where the film thickness was estimated implicitly at each time step of a dynamic simulation and considered in the prevailing contact mechanics.

1. Each numerical solution to generate a training data point took 5.88 s. The single core wall time for 600 and 5 000 training points was 58.8 min and 489 min respectively. The numerical solution database could benefit from explicit parallelisation ie. use of multiple computational cores of the CPU.

2. 600 training points achieved sufficient coefficient of determination performance (R^2 = 0.99791) to accurately predict central film thickness. Whilst 5000 points resulted in an R^2 value of 1, training of the network and effort to generate this quantity of training data rendered it computationally inefficient.

3. An ANN structure with three hidden layers, each containing 14 neurons and using a logistic sigmoid activation function, was found to be optimal for the operating bounds investigated in this study when trained on 2,000 data points.

4. The analytical equations increasingly deviate from the numerical calculation at high entrainment velocities; up to 22.6 % at 21 000 rpm for the bearing examined in this study. At the same speed, the ANN had an error of 1.58 %. Across a speed range from 0 - 21 000 rpm, the MSE of the ANN was 3.18 $\times 10^{-4}$ μm^2.

5. For the bearing case study presented, the mean squared error (MSE) of the ANN film thickness prediction was 3.89 $\times 10^{-6}$ μm^2 when benchmarked against the numerical solution for a fluctuating contact load. This is a dramatic improvement over the MSE of the analytical model which was 1.24 $\times 10^{-1}$ μm^2.

6. The ANN was shown to be $\sim$1 500 times faster than the numerical solution with a very small margin of error. The ANN is a factor of $\sim$75 times slower than the analytical equation, but a factor of 3 $\times 10^4$ more accurate when comparing MSE performance against the numerical method.

This study has proven ANNs to be an accurate and computationally efficient method of calculating EHL film thickness. Despite having no physical understanding of the system, this data-driven solution has proved accurate and computationally efficient for the film thickness estimation. This computational efficiency and accuracy lends itself very well to tribological models in FMBD solvers. It must be noted that the ability of the ANN to extrapolate beyond the bounds of the training data set must be addressed. Excessive extrapolation may lead to instability in the dynamic solution. However, by selecting a sufficient design space and robust sampling method such as LHS, this risk can be mitigated.

Roller bearings are only one application of these ANNs. The use cases extend far beyond roller bearings, to key components in automotive, machining and other industrial applications where interactions between contiguous surfaces exist. Many of these machine elements operate within the regions that this ANN was trained to cover, making it a deployable solution across many applications with very little computational training effort.