SÃO PAULO STATE UNIVERSITY
SCHOOL OF ENGINEERING OF ILHA SOLTEIRA

NEURAL NETWORKS: ENHANCING INTELLIGENT SYSTEMS WITH DEEP LEARNING

Gabriel D. Silva
Student
Douglas D. Bueno
Advisor

UNESP
Ilha Solteira – SP
2023

RESEARCH REPORT

The present report approaches a way to improve smart systems. Through artificial intelligence applied in the mechanical engineering field, it provides a consistent algorithm that can reads data, trains the machine and provides results about the situation and what to do with it. It will be studied two cases, one of them using machine learning classical techniques to determine the forces applied to a unnamed aerial vehicle and other using deep learning techniques like neural networks in the structural health monitoring area. Complete after the research is done.
Keywords: machine learning, structural health monitoring, unnamed aerial vehicle

LIST OF FIGURES

Figure 2.1 SHM and Human Nervous System Analogy 3
Figure 2.2 Quadcopter dynamic scheme 7
Figure 2.3 Visual Representation of a MLP 10
Figure 2.4 Loss Function for Linear Regression 11
Figure 2.5 Gradient Descent Process 12

Figure 3.1 Data and neural networks relation 16
Figure 3.2 Model in production 16
Figure 3.3 Schematic model of the neural network for the normalized data . . 17
Figure 3.4 Schematic model of the neural network for the norms 17

Figure 4.1 Test loss for NN 1 18
Figure 4.2 Test loss for NN 2 18
Figure 4.3 Trajectory of the VANT 19
Figure 4.4 Forces normalized obtained from the neural network 1 20
Figure 4.5 Forces denormalized obtained from the combination of neural network 1 and 2 21

LIST OF TABLES

LIST OF ACRONYM

AI Artificial Intelligence
ANN Artificial Neural Network

BD Big Data

CNN Convolutional Neural Network

DL Deep Learning

FEM Finite Element Method

IoT Internet of Things

MAE Mean Absolute Error
MEMS Micro Electromechanical Systems
ML Machine Learning
MLP Multilayer Perceptron
MSE Mean Squared Error

NN Neural Network

RNN Recurrent Neural Network

SGD Stochastic Gradient Descent
SHM Structural Health Monitoring

UAV Unnamed Aerial Vehicle

CONTENTS

List of Figures *ii*

List of Tables *iii*

List of Acronym *iv*

1 Introduction **1**

1.1 Motivation 2

1.2 Objective 2

2 Literature Review **3**

2.1 Structural Health Monitoring 3

2.1.1 Definition 3

2.1.2 Brief History 4

2.1.3 Main Techniques 4

2.1.4 Railway Cracks 5

2.2 Unnamed Aerial Vehicle Control 6

2.2.1 Usage 6

2.2.2 Control Equations 7

2.2.3 Control Algorithm 7

2.3 Artificial Neural Networks 8

2.3.1 Deep Learning 8

2.3.2 Neural Networks Models 9

2.3.3 Loss Function 10

2.3.4 Optimizer 11

3 Methodology **14**

3.1 Softwares 14

3.1.1 MATLAB® 14

3.1.2 PyTorch 14

3.2 Neural Network for Unnmaed Aerial Vehicle Control 14

3.2.1 Data Generation 14

3.2.2 Neural Networks Overview 15

3.2.3 Neural Network Modeling 15

3.3 Neural Network for SHM 16

4 Results and Discussion **18**

5 Conclusion **22**

Bibliography **23**

1 INTRODUCTION

The use of Artificial Intelligence (AI) is very present nowadays [44, 62, 64]. This area of statistics neither is new nor started just now with the autonomous cars and voice assistants [55], but it is clear that in the last years it has been increasingly gaining more popularity. This happens mainly because of the advances that the World Wide Web has been had over the years [45, 16], since dial-up internet connection, back in the eighties, until now, with broadband internet and smartphones equipped with 5G connection. Another factor is that in the past, the cost to get a large capacity of storage memory was significantly more expensive than it is now, what makes today cheaper and easy to get memory to store information [30]. With the amount of data available, the evolution of internet and storage capacity, now it is not difficult to obtain, keep and analyze databases to make decisions [23].

AI application is everywhere and today, more than ever, it is easy to realize that. Either to get multimedia recommendations on streaming platforms, like occurs at Netflix, YouTube, Spotify, and so many others platforms [15], or to make predictions on the financial market and sports betting [54, 41, 35], AI is there behind the scenes making all the magic happen. Evidently there is nothing really magical about them, it is pure mathematics combined with a programming language that produces the algorithm capable of doing those things [31, 4, 66, 67]. The launch of ChatGPT–3, and shortly thereafter ChatGPT–4, has shown the power of those technologies and how they can change the way people do things [11, 12, 48, 7].

Getting into the smart systems application, the use of AI is widely used to Structural Health Monitoring (SHM), which is heavily used in the aerospace and civil fields, [6, 86]. The level and the complexity of the AI to be applied to monitor the structure, whether is going to use Deep Learning (DL) and Neural Network (NN) or simpler methods of Machine Learning (ML) like regressions, is determined by the problem itself and the results desired [25]. In some cases, the standards methods use numerical techniques and they may not be feasible, especially when there is a huge data to be analyzed. Thus, taking the AI road is an alternative to get the needed results for the monitoring in a more practical way [74, 77].

Still in this context, but in the field of Unnamed Aerial Vehicle (UAV), the use of AI can be combined to integrate UAV through wireless communication networks [43] what can be useful in the agriculture sphere [2] with technologies like Internet of Things (IoT) [82, 81]. Also, the use of the AI can be subtle, such as the use of a built-in MATLAB® function to make a simple NN to determine the final pose of a UAV based on the initial pose and the forces applied on it [28], or can be more sophisticated, like the use of ML and DL algorithms to predict materials properties, design new materials, discover new mechanisms and control real dynamic systems [33, 3].

It is clear, therefore, that AI can transit into different fields, such as entertainment, business, health care, marketing, financial, agriculture, engineering, among others [68, 87, 20, 83, 53, 59, 29]. The use of the Big Data (BD) can not only make it clear the scenario to be studied, but also to support making strategical decisions [36, 42]. The internet and hardware improvement [8], alongside the facility to storage data with accessible costs, encourages the AI use due to the benefits it can provide.

1.1 Motivation

With the 4.0 industry, the engineering evolution is growing bigger every year [52]. Solving engineering problems the traditional way may not be the best solution due to its non-triviality to complex systems and their mathematical modeling. With the amount of available data and power processing, many tasks may now be done with the AI aid [61].

In the SHM field, detecting failures and monitoring the structure is vital to prevent damages. Installing sensors in the structure and send signals to a central processing is a common way to provide predictive maintenance for structures [38, 56]. Accidents with wagons can be avoided by detecting cracks in railways. AI can be used to determine the failures and possible damages to determine if the railway is able to keep working.

For the control engineering area, a control system to UAVs are necessary to get them to do their tasks properly. A traditional *white box* method is certainly a rock-solid but not a trivial way to do that task. With a *black box* approach, it is possible to determine the control forces by having only the initial pose and the desired trajectory, instead of modeling mathematically from the scratch [47, 85]. This way, for determined situations, all the mathematical modeling complexity can be replaced to an AI system that can predict the control forces by giving easy-to-get information.

The use of AI techniques as a different approach to the traditional engineering problems can decrease efforts to complex systems and facilitate problems resolution, providing low-costs and faster solutions.

1.2 Objective

To develop two AI algorithms based on NN to apply in smart systems. The main goals are: (i) to determine the control forces used to move an UAV based on its initial pose and the desired trajectory; and (ii) to detect railways cracks through piezoelectric signal for SHM.

2 LITERATURE REVIEW

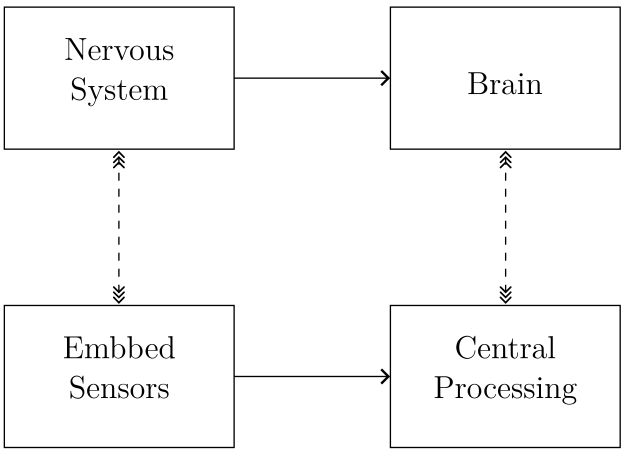
This chapter deals with the history, the main concepts and some practical cases of SHM inside the industry and academic area, besides showing how it may be used in the railway crack detection context. Next, in the dynamic field, it will be studied the main mechanical concepts to get the necessary understanding to an UAV motion as well the basics to know how an UAV can be controlled. Then, it will be shown the mathematics behind the algorithms of deep learning that will be implemented in the Chapter 4. Finally, the way how the algorithms are going to be implemented and the tools necessary to achieve the desired NN.

2.1 Structural Health Monitoring

2.1.1 Definition

According to Balageas et al. [9], the SHM main purpose is to provide, during the life of a structure, a diagnosis of: the state of the constituent material; the different parts of the structure; and the full assembly of each part that makes the structure as a whole. It is an improved way to make non-destructive evaluation. It can be applied in several areas such as civil infrastructure, like bridges and buildings; aerospace, like airplanes and spaceships; and mechanical, like machines.

Figure 2.1. SHM and Human Nervous System Analogy. The nerve endings are responsible to pass the information to the brain; as the sensors embedded in the structure play the role of the nerve endings, the central processing does the brain role.



Source: prepared by the author.

Furthermore, it also can be associated as an analogy to the human nervous system. Just like the sensors send a signal to the central processor, the human senses send a signal to the brain to make the recognition of what is happening, as shown in the Fig. 2.1.

2.1.2 Brief History

The SHM development began back in the 20th century and it has been coupled with the evolution of the digital computing hardware, what allowed the costs of the applied techniques less expensive over time.

It all starts back in the early 1970s and 1980s. The oil industry tried to develop vibration-based damage identification methods for offshore platforms by simulating damage scenarios, examining the changes in the resonant frequencies and correlating them with those measured on a platform. In the same period, the aerospace community studied vibration-based damage identification along with the development of the space shuttle. From that, it was developed the shuttle modal inspection system which aimed to identify fatigue damage in components like fuselage panels and control surfaces. The system was so successful that all orbiter vehicles had been periodically subjected to this test. Also, the civil engineering community studied vibration-based damage evaluation of bridge structures and buildings in the late 1980s [24].

From the late 1990s to the early 2000s, Sohn et al. [75] showed the evolution of the techniques used in SHM, analyzing mainly the following factors: the operational evaluation; data acquisition and cleansing; feature extraction; and statistical modeling for feature discrimination. He also verified that the statistical patter recognition had not been embraced by the researchers to be more often used in such matter.

Nowadays, in order to contour inherent issues of SHM methods, as large computational effort and hand-crafted work that results in poor classification performance, many deep learning techniques have been used, such as Convolutional Neural Network (CNN) [5].

2.1.3 Main Techniques

Accelerometers

The use of accelerometers is consolidated in the engineering community to be used in several areas and it is present in the people daily life in things like game consoles, smartphones, and tablets.

Micro Electromechanical Systems (MEMS) sensor have several applications in measuring linear acceleration or angular motion along axis as an input to control a system. MEMS accelerometer sensors often measure the movement of a mass with a position measuring interface circuit that is converted into a digital electrical signal by an analog-to-digital converter for digital processing [18].

In SHM situation, the accelerometers are in the MEMS. The MEMS are, then, embedded in the structure and can provide information about the structure by detecting low-amplitude and low-frequency vibrations that are not always viable with the conventional low-cost sensor boards [69].

There are many others sensors used in vibration-base techniques like velocity and displacement sensors, however the accelerometers are widely used for this purpose.

Graphical Inspection

The use of digital cameras to detect any kind of irregularity in the surface is also a way to monitor the structure, mainly in the surface areas. The camera itself may be static in a strategical position that allow it to provide good images to be analyzed or can be embedded in the structure itself or in an UAV that will surround it.

To automate and improve the accuracy of the damage detection, image processing techniques are employed, that being a non-conventional approach [72]. In the civil engineering context, it is commonly utilized computer vision to damage detection [26] and also UAV integrated in the same local as the structures for SHM [71].

Many of the images obtained can have their not only the images improved by AI, but also the analyses can take advantage of it.

Piezoelectric Materials

When dealing with acoustic-based techniques, the use of piezoelectric materials as sensor is a great choice due to its ability to respond to stimuli, incorporation, and compatibility with construction materials. Beyond that, these materials are relatively cost-efficient and can sense vibrations in the structures they are installed [37].

Piezoelectric materials and their main property were discovered back in 1880 [17]. The phenomenon is that by the application of pressure in those kinds of materials in the correct direction, it is observed the production of a potential difference and consequently an electrical charge. Examples of materials that are piezoelectric are quartz, zinc, sodium chlorate, tourmaline, calamine, topaz, tartaric acid, cane sugar, and others [13].

The application of these materials in SHM is basically to install the piezoelectric sensor in the structure intend to be monitored and through the tension or compression in it done, a sign will be sent to the central system by the potential differential. The signal indicates that something not usual is happening in the structure. Of course there are levels of the signals and each case must be evaluated in its context. In the last years the use of piezoelectric materials has been capable to identify failures, like the presence of delamination damage, as long as the piezoelectric sensors are close to the damage [49].

2.1.4 Railway Cracks

Train is one of the most used means of transportation around the world, either to transport people or groceries, therefore, there are inherent problems in the attached to it. One of the most common problems is the crack on the railway track, mainly due to the expansion and contraction caused by the heat and to constant pressure because of the wagon.

The crack in a railway is considerable problem because it may cause fatal accidents since the wagon is able to leave the railway. In this scenario, many methods are used to detect the crack or to foresee it before any misfortune happen. Karthick and Ramalingam [39] proposed a system to identify the cracks and prevent the accidents. One of its advantages is that if some crack is detected on the track, the train starts to slow and stop before it passes by there. Other method includes the use of sensor coupled in the track that allow to detect the crack and send a signal to the command center through IoT [70].

The use of piezoelectric materials for SHM is very common, as seen in the Section 2.1.3. Loveday [46] presents a system where piezoelectric transducer are installed along the railway track. They receive an electrical wave and send it, then, a signal to the receiver, making it possible, also through Finite Element Method (FEM), to detect any inconsistency that should not be there.

There are, hence, lots of methods that can be used to detect and prevent accidents in railway tracks. Putting they together and optimize them with SHM techniques are an efficient way of improve the railway ecosystem.

2.2 Unnamed Aerial Vehicle Control

2.2.1 Usage

An UAV has several applications, going from the simplest to the most sophisticated. It can be used since for entertainment, like toys; commercially, to record big shows in arenas; surveillance, to monitor places; and also in engineering, aiding in various context to improve some processing.

Due to its portability and autonomy, it can be used to facilitate the delivery o medicines. In this sense, UAV can be used for transportation of medical goods in critical times, where other means of transportation may not be feasible. In the final of 2019, COVID-19 pandemics spread throughout the world, making it difficult to deliver patients their needed medicines [65, 51]. Besides, risks are inherent to the transportation and in come countries, like the USA, UAV usage may be restricted [80]. A strategical way to use them is also welcome.

In the agriculture context, in order to boost the productivity, UAV can be used to remotely sense the farming, obtaining information on the state of the fields with non-contact procedures, like nutrient evaluation and soil monitoring; or even for aerial spraying, using pesticide to prevent damages in the plantation [21].

The main reason for its adoptions is the mobility, low maintenance costs, hovering capacity, ease of deployment, etc. It is widely used for the civil infrastructure, gathering photographs faster than satellite imagery and with better quality. Combining those benefits with AI can be a powerful tool for the future [73].

2.2.2 Control Equations

Considering the UAV a quadcopter, as the Fig. 2.2 shows, Geronel et al. [28], based on the work of Fossen [27], described the equation of motion for a quadcopter with a payload as being:

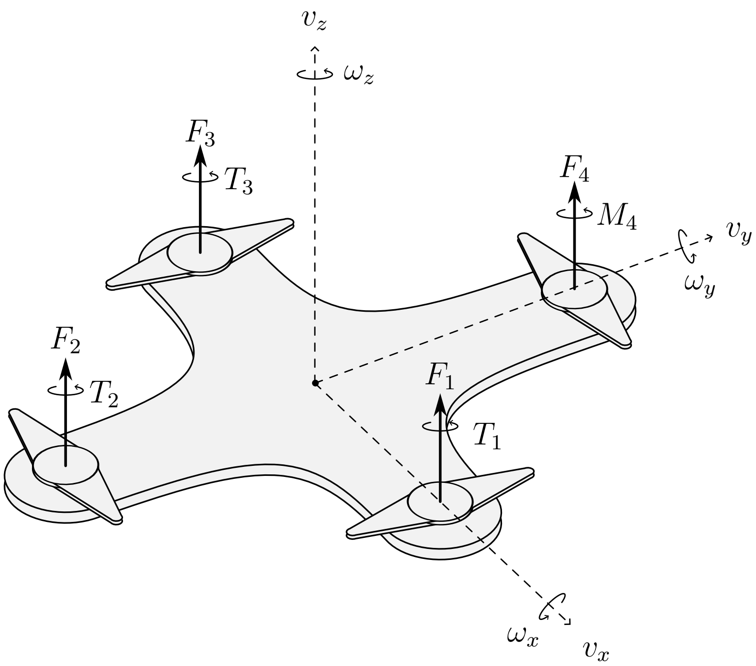
$$\mathbf{M}_{\eta_c}(\eta_c)\ddot{\eta}_c + \mathbf{C}_{\eta_c}(\nu,\eta_c)\dot{\eta}_c + \mathbf{g}_{\eta_c}(\eta_c) + \mathbf{K}_{\eta_c}(\eta_c)\eta_c = \tau_{\eta_c}(\eta_c) + \mathbf{F}_d \tag{2.1}$$

where $\mathbf{M}_{\eta_c}(\eta_c)$ is the inertial matrix; $\mathbf{C}_{\eta_c}(\nu,\eta_c)$ is the Coriolis matrix; $\mathbf{g}_{\eta_c}(\eta_c)$ is the gravitational vector; $\mathbf{K}_{\eta_c}(\eta_c)$ is the stiffness matrix; $\tau_{\eta_c}(\eta_c)$ is the control torque; $\mathbf{F}_d$ is the gust vector; and ν is the velocity generalized coordinate in the body-frame. The Eq. (2.1) can be represented in the state space form as:

$$\dot{x}_s = \mathbf{A}_c x_s(t) + \mathbf{B}(u)(t) + \mathbf{X} \tag{2.2}$$

where $x_s = \left\{\dot{\eta}_c \quad \eta_c\right\}^\top$ is the state vector; $\mathbf{B}(u)(t)$ is the input vector; $\mathbf{X}$ is the state vector of gravity; and $\mathbf{A}_c$ and $\mathbf{B}$ are the dynamic and input matrices, respectively.

Figure 2.2. Quadcopter dynamic scheme. F_i and T_i, $(i = 1, 2, 3, 4)$, are the forces and the torque applied in the propeller, respectively. ω_j and v_j, $(j = x, y, z)$, are the momentum and the velocities applied in the UAV, respectively. The payload is not represented in the figure.



Source: prepared by the author.

All non-explicit matrices and the development of the equations are shown in the Geronel et al. [28] work.

2.2.3 Control Algorithm

Geronel et al. [28] developed a MATLAB® algorithm to control the quadrotor, as a *white box* method. It controls the UAV in three different trajectories: rectangular, circular and linear. Given τ as the input vector, which represents the position controller $U_1(t)$ and the attitude controller $U_2(t)$, $U_3(t)$, $U_4(t)$, it is able to give a complete overview of the quadrotor's motion. The algorithm provides the state space vector $\mathbf{x}_s$ with the quadrotor position and angles, as their derivatives.

$$\tau = \left\{ U_1 \quad U_2 \quad U_3 \quad U_4 \right\}^\top \tag{2.3}$$

$$\mathbf{x}_s = \left\{ x \quad y \quad z \quad \phi \quad \theta \quad \psi \quad \dot{x} \quad \dot{y} \quad \dot{z} \quad \dot{\phi} \quad \dot{\theta} \quad \dot{\psi} \right\}^\top \tag{2.4}$$

2.3 Artificial Neural Networks

2.3.1 Deep Learning

The concepts of deep learning studied in this section is going to be based on the work of Goodfellow et al. [31], Haykin [34] and the documentation of PyTorch[1], TensorFlow[2] and MATLAB® [3].

There are several definitions of AI [84], but the computer scientist McCarthy [50] defines it as "the science and engineering of making intelligent machines, especially intelligent computer programs". He also states that "it is related to the similar task of using computers to understand human intelligence, but AI does not have to confine itself to methods that are biologically observable".

The big area of study is the AI and it includes several branches like fuzzy logics, robotics and machine learning. The later one, in turn, is another field with also some branches and one of them is the deep learning. However, all the three terms can be interchangeable in the major context.

The deep learning history goes back to the 1940s and it had several names over the years. It was called by *cybernetics* (1940s–1960s), *connectionism* (1980s–1990s), and from 2006 until now is known as *deep learning*. The DL models were engineered systems inspired by the biological brain and they were denominated Artificial Neural Network (ANN). One of the motivations of the neural perspective was to understand that the brain provides a proof by example that intelligent behavior is possible and try to reverse engineer the computation principals behind the brain, duplicating its functionality. Today it goes beyond the neuroscientist perspective and it is more of general principle of learning multiple levels of composition.

DL dwells in the programming sphere. The approach, however, it is not like the

[1] https://pytorch.org/docs/stable/index.html
[2] https://www.tensorflow.org/api_docs
[3] https://www.mathworks.com/help/matlab/

traditional programming scripts and models. To automate stuff, there are three main parts: (i) the input data, (ii) the rule (function) and (iii) the output data. In oth types there are two of three parts available, but different ones for each other. In the traditional programming, there is the input data and the rule, for the algorithm output the data. For deep learning, there is the input data and the output data, for the algorithm provides the rule. A good analogy is cooking: in the traditional programming context, one has the ingredients and the recipe to make the main course; in the deep learning context, one has the ingredients and the main course to discover the recipe.

2.3.2 Neural Networks Models

A ANN is machine learning a model that simulate a biological NN to make a machine learns as the human being learns. ANN are the heart of DL and there are several models of them, each one most suitable for different kind of problems. Some of them are Multilayer Perceptron (MLP), CNN, Recurrent Neural Network (RNN), among others.

Multi-layer Perceptron

A MLP is a important class of NN. It consists of a set of sensorial units that compose the *input layer*; one or more *hidden layers*; and an *output layer*. The input signal propagates forward through the network, layer by layer. They are used to solving complex problems, with the supervised training with the *error back-propagation* algorithm.

The learning by back-propagation consists of two steps through the layers of the perceptron: a forward pass (propagation) and a backward pass (back-propagation). In the forward pass, an input vector is applied to the sensorial nodes of the network and it propagates through the network, layer by layer; in this step the weights are fixed. During the backward pass, the weights are fit accordingly through a loss function (see Section 2.3.3). This error signal is propagated through the network in the opposite direction of the synaptic connections. The weights are adjusted to make that the network output gets closer of the wanted output. Fig. 2.3 represents a MLP.

The three main features of the MLP are:

- Non-linear activation function. It is commonly used a smooth non-linear activation function, like rectifier function (ReLU):

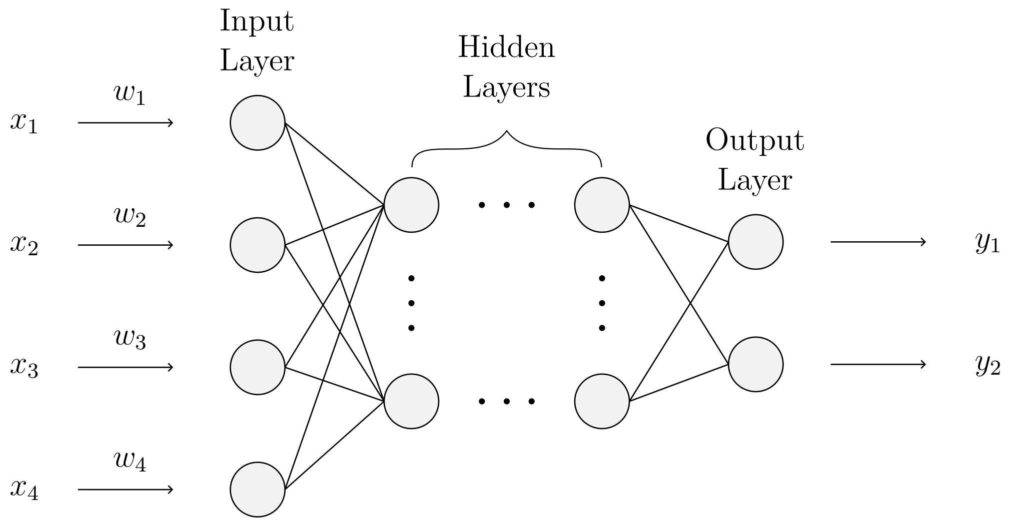
$$y_j = \begin{cases} x, & \text{if } x > 0, \\ 0 & \text{otherwise.} \end{cases} \tag{2.5}$$

  Or sigmoid function:

$$y_j = \frac{1}{1 + \exp(-v_j)} \tag{2.6}$$

  where v_j is the weighted sum of all input layers with their respective weights of the

Figure 2.3. Visual Representation of a MLP. The input vector x_i is given in the input layer with i assuming any integer, just as the output data y_i. The weights w_{x_i} are specific for each input data. Input and output data can have multiple entries.



Source: prepared by the author.

j neuron; and y_j is the output of the neuron.

- Hidden layers. They allow the network to learn complex tasks, extracting progressively the most significantly features of the input vector.

- Connectivity. High level of connectivity, determined by the network synapses.

These features, plus the ability to learn from the experience of the training, that makes the MLP so powerful, however, they are also responsible for its deficiency. First, the non-linearity and the high connectivity makes hard the theoretical analysis of an MLP; second, the hidden layers make it more difficult to visualize the learning processing. The learning process is harder because the search must be conducted in a much bigger space of possible functions.

2.3.3 Loss Function

The *loss function*, also called *cost function* or *error function*, is the one used measure the error between the predicted output of an algorithm and the real target output. There are several loss functions suitable to different kind of situation. For each distributed data there is one that fits better. Many kinds of them are available and must be analyzed the most proper one to each case. The choice of what loss function should be picked will depend on not only the data and its pattern, but also the computational processing and the cost attached to it.

Regression

Although regression problems do not require DL to create a satisfactory model, naturally it is possible to do so. For regression problems, common loss functions adopted are the Mean Absolute Error (MAE) and Mean Squared Error (MSE) [14]:

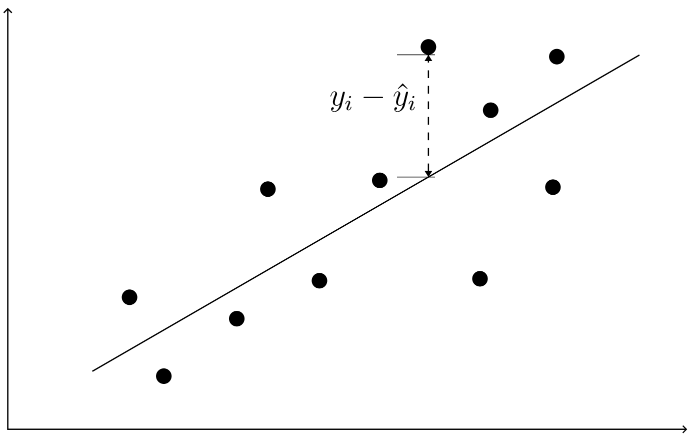
$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{2.7}$$

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2.8}$$

where n is the sample size; y_i is the predicted output; and $\hat{y}_i$ is the real target.

Fig. 2.4 shows a linear data and how the domain of the loss function is obtained for a linear regression.

Figure 2.4. Loss Function for Linear Regression. The loss function take all the distances between the predicted and the target value to verify if the model is in the right path. The lower the distance, the better the model. The y-axis represents the output data and the x-axis represents the input data.



Source: prepared by the author.

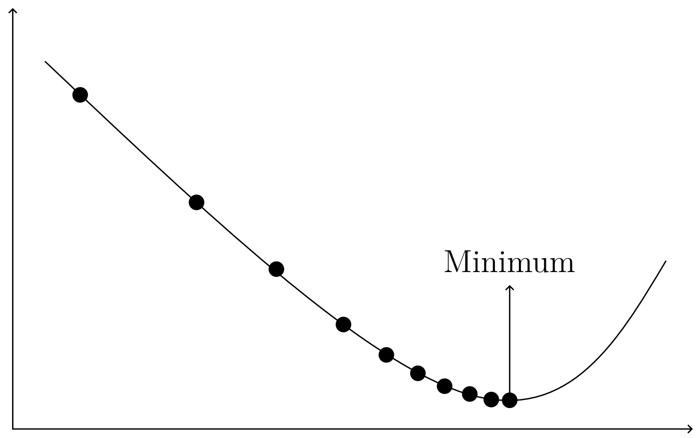
2.3.4 Optimizer

The optimizer is an algorithm that updates the model in response to the output of the loss function, that is, it aids to minimize the loss function. As the loss function minimizes, the model is getting closer to the target values and, hence, closer to the real pattern.

Gradient Descent

The *gradient descent* is one of the main algorithm [57] that optimizes the model and many important ones are based on it, like the Stochastic Gradient Descent (SGD). The goal is to get the minimum, as the error (loss) between the predicted and the target data is null. This would mean that the model fits to the pattern of the data.

Figure 2.5. Gradient Descent Process. In this case, the loss function (yellow curve) can be represented in a two-axes plan. Depending on the data, it is not possible to represent graphically due to its multi dimension. Each point represents the learning step. When the gradient descent reaches the minimum of the loss function, it means that the model may be accurate. Note that the gradient descent can reach a local minimum of the function and not the global minimum necessarily. The y-axis represents the loss function and the x-axis represents the weight values.



Source: prepared by the author.

The gradient descent is a powerful algorithm that reduces the loss function, minimizing the error between the predicted value and the target value.

Since the gradient of a function gives the direction of the steepest ascent of a function and it is orthogonal to the surface at a determined point, it seems reasonable that moving in the perpendicular direction gives the maximum increase of the function [76]. On the other hand, the negative of the gradient may be used to find the opposite, that is, the steepest descent of the function, or the minimum decrease. If the steps given to the direction of the negative gradient of the function are small, there is a good chance to get minimum value of the function. However, if the steps are too long, the chance to pass by the minimum value is high [58]. These steps are called *learning rate* and should be chosen wisely.

This way, let $\mathbf{x}$ be the entry vector with the predicted data and L the loss function adopted for some deep learning model, and ϵ the learning rate, the gradient descent is:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \epsilon \nabla L(\mathbf{x}_t) \tag{2.9}$$

In determined cases, it is possible to avoid running the iterative algorithm and just go directly to the critical point by solving $\nabla L(\mathbf{x}_t) = 0$ for $\mathbf{x}$.

Stochastic Gradient Descent

As seen, gradient descent is a powerful tool to minimize the loss function, however, for large data, the cost of operation is very high and its use is not feasible. The main idea of SGD is that the gradient is an expectation. Later, the data is divided in subsets, also called *mini-batch* and then the gradient is performed over them. The mini-batch size is chosen to be a relatively small numbers of examples. The data inside each subset may be considered redundant, that is why it uses one single value of the subset to compute the gradient descent. This way, the process is considerable better for computational resources.

The SGD can be written as:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \frac{\epsilon}{m} \sum_{i=1}^{m} \nabla L(\mathbf{x}_t; p^{(i)}, q^{(i)}) \tag{2.10}$$

where m is the mini-batch size; and $\nabla L(\mathbf{x}; p^{(i)}, q^{(i)})$ is the gradient of the loss function with respect to the parameter vector $\mathbf{x}$ for the i^{th} example $(p^{(i)}, q^{(i)})$ in the mini-batch.

Yet, nowadays, with the amount of data, many techniques are still applied in SGD as creating an automatic adaptive learning rates which achieve the optimal rate of convergence [19] and the momentum technique to improve it [78].

Adam

Adam is an algorithm for first-order gradient-based optimization of stochastic objective functions, like the loss function, as seen in the Section 2.3.3. It is based on adaptive estimates of low-order moments and computationally efficient, requiring little computational memory. Adam is a strategical choice when using large data or parameters and with very noisy/sparse gradients [40].

Kingma and Ba [40] showed that the algorithm can be implemented as it follows:

Algorithm 1 Adam Algorithm. Good default setting are $\alpha = 0.001,\ \beta_1 = 0.9,\ \beta_2 = 0.999$ and $\epsilon = 10^{-8}$. Operations on vectors are element-wise.

Require: α: stepsize
Require: $\beta_1,\ \beta_2\ \in [0, 1)$: exponential decay rates for the moment estimates
Require: $f(\theta)$: loss function
Require: θ_0: initial parameter

```
  m_0 ← 0
  v_0 ← 0
  t ← 0
  while θ_t not converged do
      t ← t + 1
      g_t ← ∇_θ f_t(θ_{t−1})
      m_t ← b_1 · m_{t−1} + (1 − β_1) · g_t
      v_t ← β_2 · v_{t−1} + (1 − β_2) · g_t^2
      m̂_t ← m_t/(1 − β_1^t)
      v̂_t ← v_t/(1 − β_2^t)
      θ_t ← θ_{t−1} − α · m̂_t/(√v̂_t + ε)
  end while
      return θ_t
```

3 METHODOLOGY

The code implementation will be pragmatical and the lines of the code will not be fully explained. The frameworks methods will not be explained either, but their documentation are reasonably comprehensive with previous programming knowledge, especially in Python and MATLAB® , and they are going to be linked whenever possible.

While the engineering goal of AI is to solve real-world problems using it as an equipment, the scientific goal is to determine which ideas explain the various sorts of intelligence [84] and the current objective is to use AI from the engineering perspective.

3.1 Softwares

3.1.1 MATLAB®

The standard in the engineering industry, MATLAB® is a powerful toolbox that can be used to several activities. Since applications in fields like medicine and biology [22], like image processing, until the most complex problems in engineering [10], that involve matrix operation and control simulation, MATLAB® has lots of tools that aid to solve problems.

Geronel et al. [28] used MATLAB® to develop their algorithm, hence the data generation will be done with it.

3.1.2 PyTorch

A framework is a group of libraries for a programming language that implements a lot of tools to facilitate some tasks. There is a lot of deep learning ones available and the most popular ones are TensorFlow [1] and PyTorch [60]. While the first one was developed by Google and released in 2015 the second one was developed by Meta (Facebook), although it is now under the Linux Foundation umbrella, and released in 2016, being both open-source. Many companies, like Uber [32] and Tesla [63], use PyTorch in their AI team, while companies like Coca-Cola uses TensorFlow [79]. This means that both are trustful frameworks to rely upon their built-in functions. PyTorch will be used to develop all the NN.

3.2 Neural Network for Unnmaed Aerial Vehicle Control

3.2.1 Data Generation

Since the script of Geronel et al. [28] provides the control torque τ as input and the state-space $\mathbf{x}_s$ as output vector through dynamic and control equations, the NN goal developed is to go in the opposite direction, as a inverse function: take $\mathbf{x}_s$ as the input vector and predict the τ_η vector as output. Modifications in the script are minimal. The time is a discrete vector with 200 s and step 0.01, therefore the time vector has 1×20001

dimension. The “extra” value of time is the zero value.

The output vector $\mathbf{T}$ has 20001×4 dimension the and the input vector $\mathbf{X}_s$ has 20001×12 dimension:

$$\mathbf{T} = \begin{bmatrix} U_1 & U_2 & U_3 & U_4 \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{3.1}$$

$$\mathbf{X}_s = \begin{bmatrix} x & y & z & \phi & \theta & \psi & \dot{x} & \dot{y} & \dot{z} & \dot{\phi} & \dot{\theta} & \dot{\psi} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{3.2}$$

Circular trajectory was arbitrary selected as starting point. By a loop, it was generated 1000 different trajectories changing the position by increasing 1/600 for each loop.

This way, it was generated one thousand input and output vector. Both were stored in a MATLAB® variable and exported through the `.mat` extension to be used with TensorFlow inside Python environment.

3.2.2 Neural Networks Overview

The first approach for the NN is to use the raw data, both for input and output and do the training. Even though it works, it does not give the proper result. Therefore, the preprocessing of the data is mandatory to get the best results. This way, all input and output data were normalized in order to get them all standardized.

The normalization is in L2 form, from the *sklearn.preprocessing.normalize* function, applied in each matrix column. From the trained NN, all input data should be normalized and naturally the output also will be normalized. However, the control forces (output data) can not be normalized to be useful, but there is no “denormalized” correspondent matrix to the output data from the NN.

To solve this problem, a second NN was created to be able to denormalize the output data. When preprocessing the data, as the normalization is done, both norms of the input and output data are stored and the second NN is made from them. The Fig. 3.1 show how the data and the NNs are related for the training. When the training and the validation is done, the ready-to-use model will perform as shown in the Fig. 3.2 scheme.

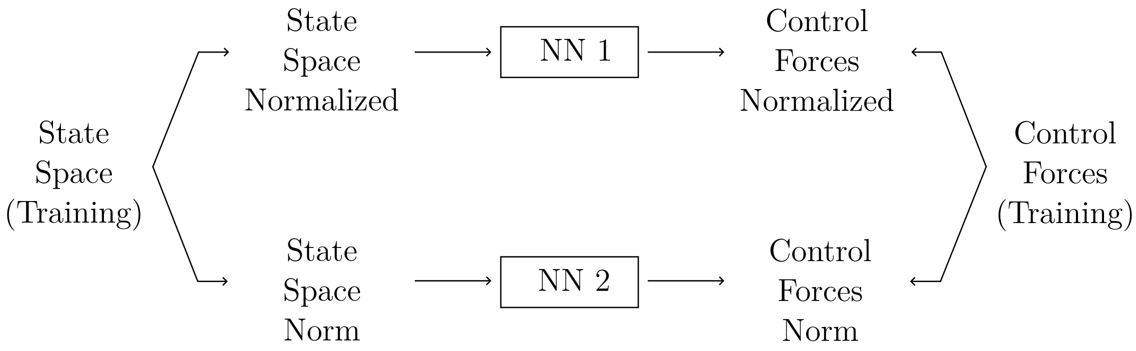
3.2.3 Neural Network Modeling

The NN 1 is responsible for, from the normalized state space, to return the normalized control forces, as shown in the Fig. 3.3. The problem is considered as a regression problem, as the Eq. (3.3) shows. Input and output data are all matrices.

$$f\left(x, y, \ldots, \dot{\theta}, \dot{\psi}\right) = \langle U_1, U_2, U_3, U_4 \rangle \tag{3.3a}$$

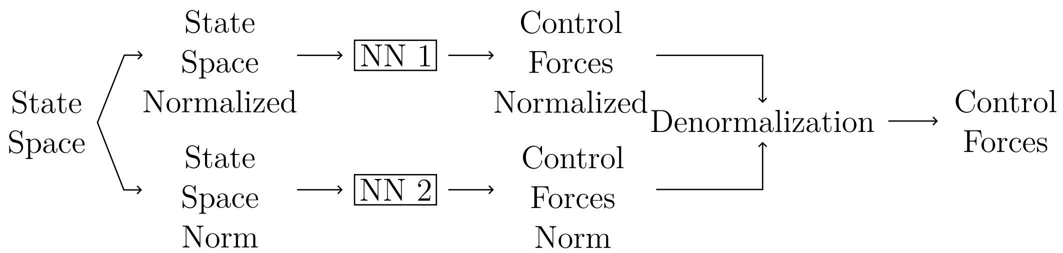
$$f(\mathbf{X}_s) = \mathbf{T} \tag{3.3b}$$

Figure 3.1. Data and neural networks relation. The training data in the extremes are the ones generated by the white box parametric model.



Source: prepared by the author.

Figure 3.2. Model in production. The scheme shows how the process returns the control forces from the state space.



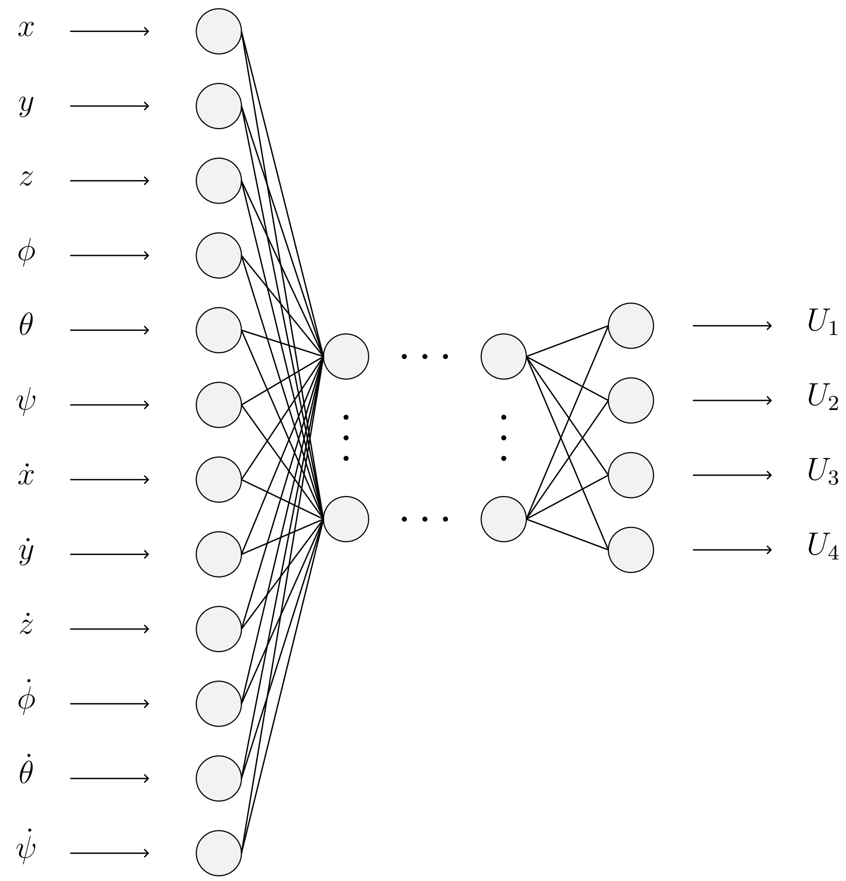
Source: prepared by the author.

Characteristics of the NN 1 are provided in the Section 3.2.3.

TABELA AQUI

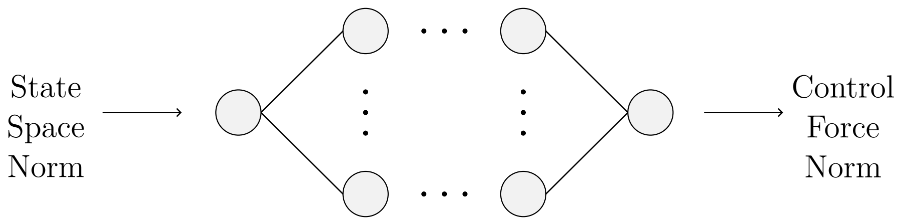
3.3 Neural Network for SHM

Figure 3.3. Schematic model of the neural network for the normalized data. The input layer receives every element of the $\mathbf{x}_s$ and returns every element of τ. There are two hidden layers, each one with 64 neurons.



Source: prepared by the author.

Figure 3.4. Schematic model of the neural network for the norms. The input data receives the state space norm and return the correspondent force control norm. There are two hidden layers, each one with 64 neurons.



Source: prepared by the author.

4 RESULTS AND DISCUSSION

This chapter shows the results obtained with the NN modeling. For both cases (SHM and UAV) the first approach is the model itself, showing the metrics and evaluating them. The second one is a comparison from the true label and the predict label for a random sample. Finally, a discussion is made based on the obtained results.

Figure 4.1. Test loss for NN 1

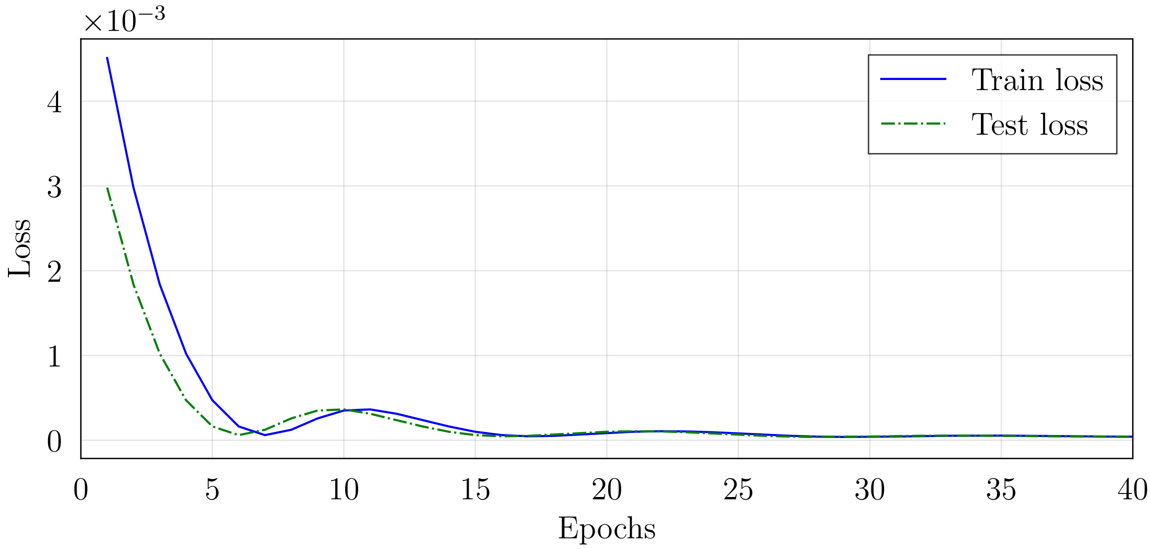


Figure 4.2. Test loss for NN 2

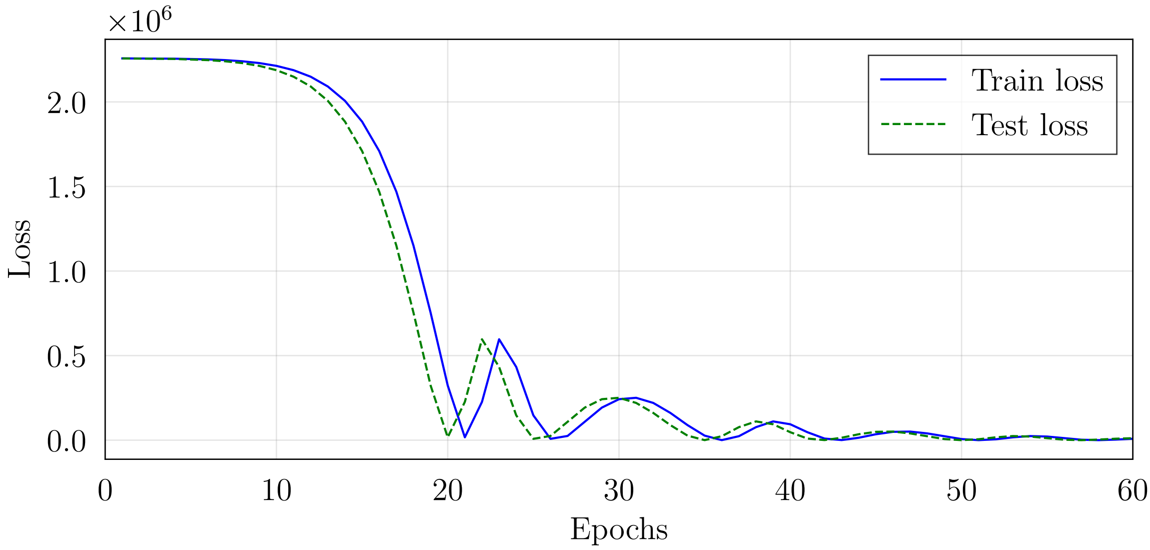


Figure 4.3. Trajectory of the VANT

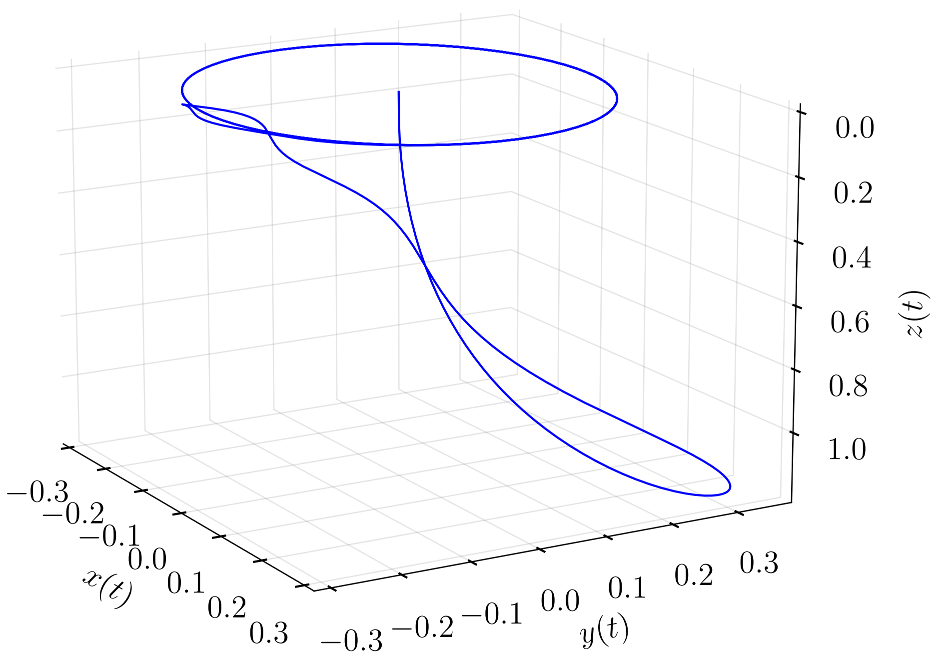
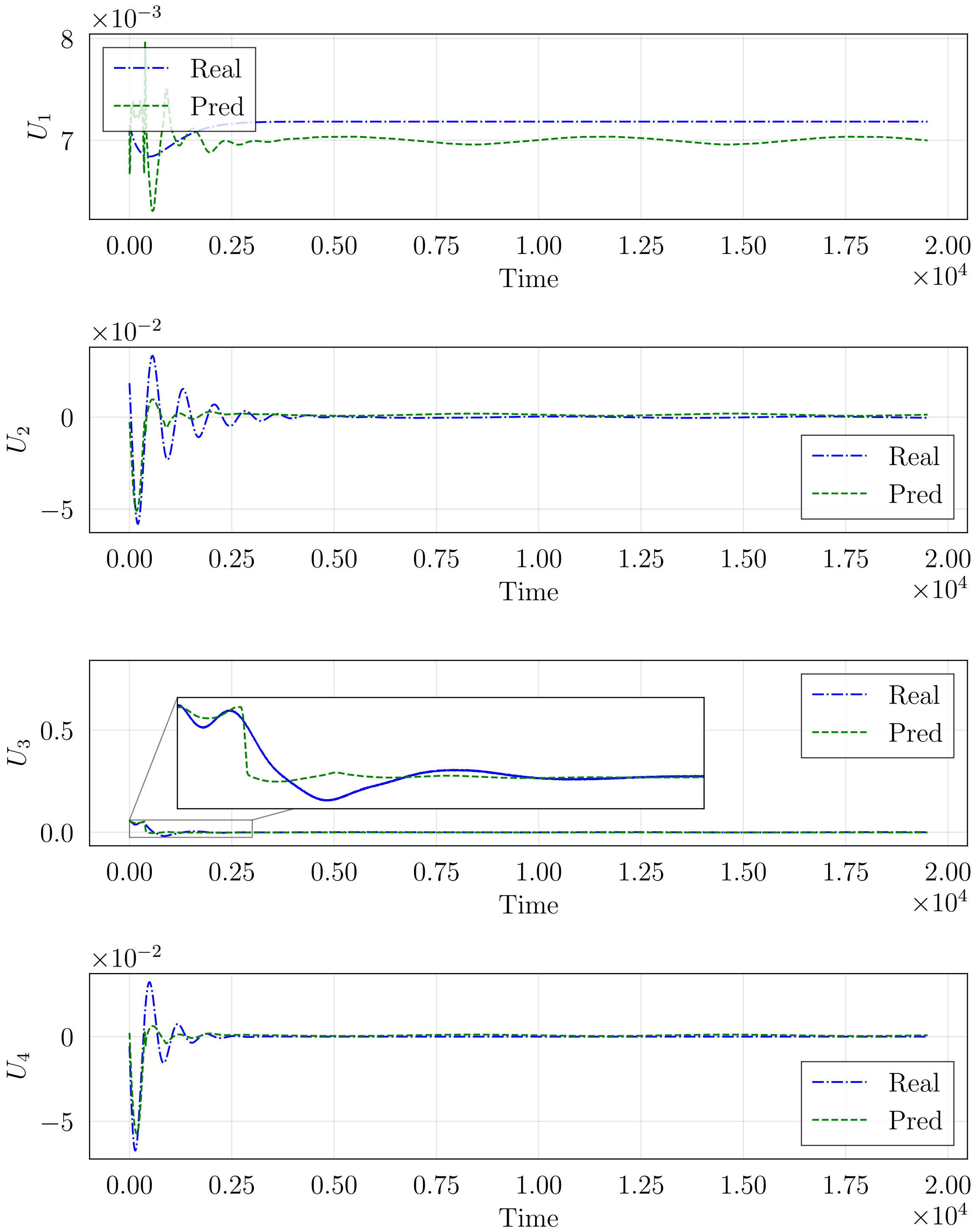
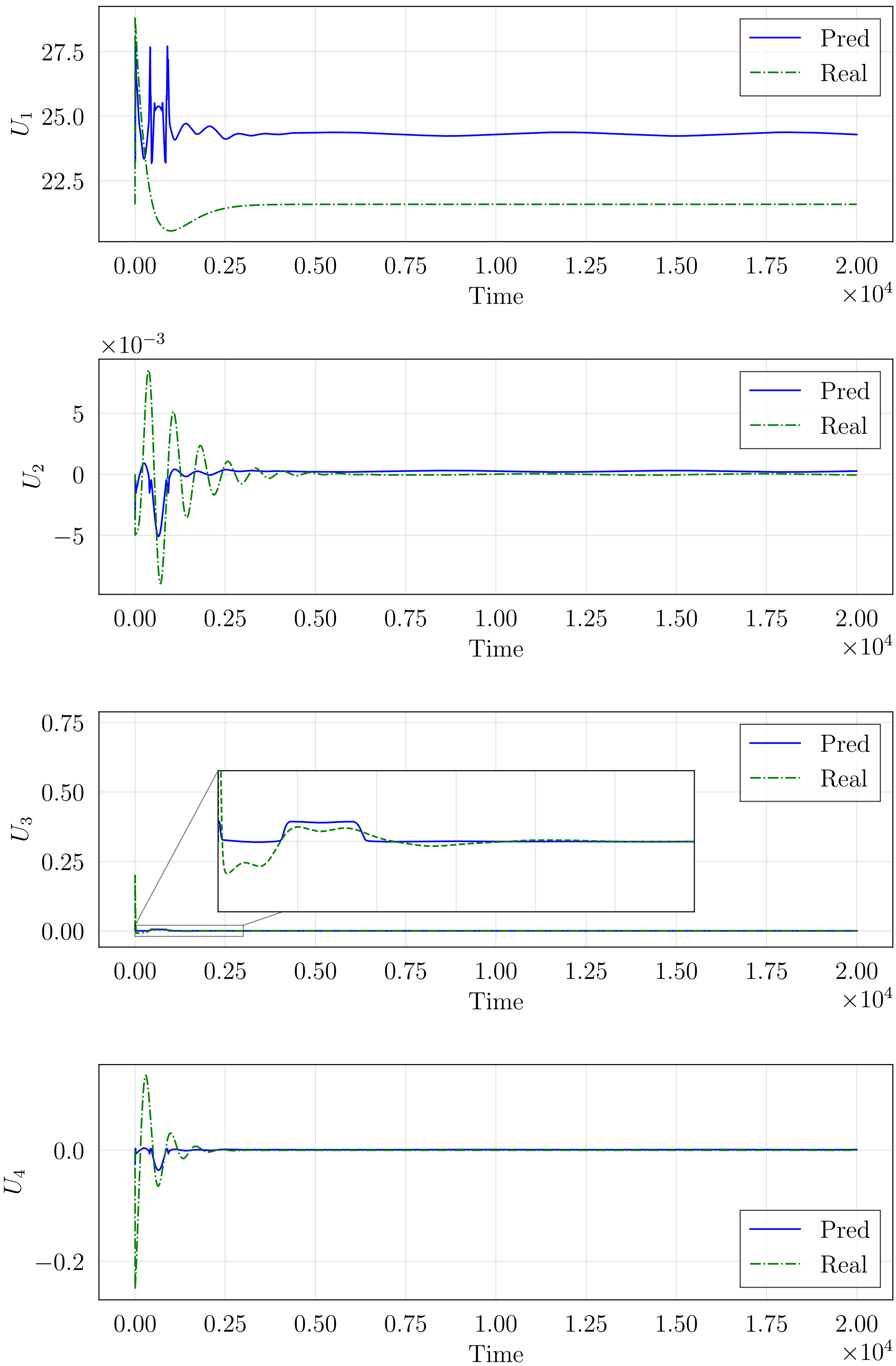


Figure 4.4. Forces normalized obtained from the neural network 1. Comparison is made from the results predicted from the neural network and the normalized preprocessing data, both for input and output data.



Source: prepared by the author.

Figure 4.5. Forces denormalized obtained from the combination of neural network 1 and 2. Comparison is made from the results predicted from the combination of both neural network and the raw data got from the white box parametric model.



Source: prepared by the author.

5 CONCLUSION

BIBLIOGRAPHY

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng. TensorFlow: A system for large-scale machine learning. *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, pages 265–283, 2016.

[2] S. Ahirwar, R. Swarnkar, S. Bhukya, and G. Namwade. Application of Drone in Agriculture. *International Journal of Current Microbiology and Applied Sciences*, 8(01): 2500–2505, Jan. 2019. ISSN 23197692, 23197706. doi: 10.20546/ijcmas.2019.801.264.

[3] S. Assilian. *Artificial Intelligence in the Controle of Real Dynamic Systems.* PhD thesis, Queen Mary University of London, 1974.

[4] G. Aurélien. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow.* O'Reilly Media, Inc., 2022.

[5] O. Avci, O. Abdeljaber, S. Kiranyaz, and D. Inman. Structural Damage Detection in Real Time: Implementation of 1D Convolutional Neural Networks for SHM Applications. In C. Niezrecki, editor, *Structural Health Monitoring & Damage Detection, Volume 7*, pages 49–54. Springer International Publishing, Cham, 2017. ISBN 978-3-319-54108-2 978-3-319-54109-9. doi: 10.1007/978-3-319-54109-9_6.

[6] M. Azimi, A. Eslamlou, and G. Pekcan. Data-Driven Structural Health Monitoring and Damage Detection through Deep Learning: State-of-the-Art Review. *Sensors*, 20 (10):2778, May 2020. ISSN 1424-8220. doi: 10.3390/s20102778.

[7] D. Baidoo-Anu and L. Owusu Ansah. Education in the Era of Generative Artificial Intelligence (AI): Understanding the Potential Benefits of ChatGPT in Promoting Teaching and Learning. *SSRN Electronic Journal*, 2023. ISSN 1556-5068. doi: 10.2139/ssrn.4337484.

[8] T. Baji. Evolution of the GPU Device widely used in AI and Massive Parallel Processing. In *2018 IEEE 2nd Electron Devices Technology and Manufacturing Conference (EDTM)*, pages 7–9, Kobe, Mar. 2018. IEEE. ISBN 978-1-5386-3712-8. doi: 10.1109/EDTM.2018.8421507.

[9] D. Balageas, C.-P. Fritzen, and A. Güemes. *Structural Health Monitoring*, volume 90. John Wiley & Sons, 2010.

[10] R. K. Bansal, A. K. Goel, and M. K. Sharma. *MATLAB and Its Applications in Engineering (Based on MATLAB 7.5 (R2007b)). - Description Based on Print Version Record*. Dorling Kindersley, Delhi. ISBN 978-81-317-4207-5.

[11] S. S. Biswas. Potential Use of Chat GPT in Global Warming. *Annals of Biomedical Engineering*, Mar. 2023. ISSN 0090-6964, 1573-9686. doi: 10.1007/s10439-023-03171-8.

[12] S. S. Biswas. Role of Chat GPT in Public Health. *Annals of Biomedical Engineering*, Mar. 2023. ISSN 0090-6964, 1573-9686. doi: 10.1007/s10439-023-03172-7.

[13] C. Brown, R. Kell, R. Taylor, and L. Thomas. Piezoelectric Materials, A Review of Progress. *IRE Transactions on Component Parts*, 9(4):193–211, Dec. 1962. ISSN 0096-2422. doi: 10.1109/TCP.1962.1136768.

[14] W. d. O. Bussab and P. A. Morettin. *Estatística Básica*. Saraiva Uni, 2017.

[15] S. M. Chan-Olmsted. A Review of Artificial Intelligence Adoptions in the Media Industry. *International Journal on Media Management*, 21(3-4):193–215, Oct. 2019. ISSN 1424-1277. doi: 10.1080/14241277.2019.1695619.

[16] R. Cohen-Almagor. Internet History:. *International Journal of Technoethics*, 2(2): 45–64, Apr. 2011. ISSN 1947-3451, 1947-346X. doi: 10.4018/jte.2011040104.

[17] J. Curie and P. Curie. Développement par compression de l'électricité polaire dans les cristaux hémièdres à faces inclinées. *Bulletin de la Société minéralogique de France*, 3 (4):90–93, 1880. ISSN 0150-9640. doi: 10.3406/bulmi.1880.1564.

[18] M. Dadafshar. Accelerometer and Gyroscopes Sensors: Operation, Sensing, and Applications. *Maxim Integrated [online]*, 2014.

[19] C. Darken and J. E. Moody. Towards Faster Stochastic Gradient Search. 4, 1991.

[20] T. Davenport and R. Kalakota. The potential for artificial intelligence in healthcare. *Future Healthcare Journal*, 6(2):94–98, June 2019. ISSN 2514-6645, 2514-6653. doi: 10.7861/futurehosp.6-2-94.

[21] J. del Cerro, C. Cruz Ulloa, A. Barrientos, and J. de León Rivas. Unmanned Aerial Vehicles in Agriculture: A Survey. *Agronomy*, 11(2):203, Jan. 2021. ISSN 2073-4395. doi: 10.3390/agronomy11020203.

[22] O. Demirkaya, M. H. Asyali, and P. Sahoo. *Image Processing with MATLAB: Applications in Medicine and Biology*. CRC Press, Boca Raton, 2009. ISBN 978-0-8493-9246-7.

[23] Y. Duan, J. S. Edwards, and Y. K. Dwivedi. Artificial intelligence for decision making in the era of Big Data – evolution, challenges and research agenda. *International Journal of Information Management*, 48:63–71, Oct. 2019. ISSN 02684012. doi: 10.1016/j.ijinfomgt.2019.01.021.

[24] C. R. Farrar and K. Worden. An introduction to structural health monitoring. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 365(1851):303–315, Feb. 2007. ISSN 1364-503X, 1471-2962. doi: 10.1098/rsta.2006.1928.

[25] C. R. Farrar and K. Worden. *Structural Health Monitoring: A Machine Learning Perspective*. John Wiley & Sons, 2012.

[26] D. Feng and M. Q. Feng. Computer vision for SHM of civil infrastructure: From dynamic response measurement to damage detection – A review. *Engineering Structures*, 156:105–117, Feb. 2018. ISSN 01410296. doi: 10.1016/j.engstruct.2017.11.018.

[27] T. I. Fossen. *Guidance and Control of Ocean Vehicles*. Wiley, Chichester ; New York, 1994. ISBN 978-0-471-94113-2.

[28] R. S. Geronel, R. M. Botez, and D. D. Bueno. Dynamic responses due to the Dryden gust of an autonomous quadrotor UAV carrying a payload. *The Aeronautical Journal*, 127(1307):116–138, Jan. 2023. ISSN 0001-9240, 2059-6464. doi: 10.1017/aer.2022.35.

[29] S. Ghatrehsamani, G. Jha, W. Dutta, F. Molaei, F. Nazrul, M. Fortin, S. Bansal, U. Debangshi, and J. Neupane. Artificial Intelligence Tools and Techniques to Combat Herbicide Resistant Weeds—A Review. *Sustainability*, 15(3):1843, Jan. 2023. ISSN 2071-1050. doi: 10.3390/su15031843.

[30] K. Goda and M. Kitsuregawa. The History of Storage Systems. *Proceedings of the IEEE*, 100(Special Centennial Issue):1433–1440, May 2012. ISSN 0018-9219, 1558-2256. doi: 10.1109/JPROC.2012.2189787.

[31] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, 2016.

[32] Goodman. Uber AI Labs Open Sources Pyro, a Deep Probabilistic Programming Language. https://www.uber.com/en-GR/blog/pyro/, Nov. 2017.

[33] K. Guo, Z. Yang, C.-H. Yu, and M. J. Buehler. Artificial intelligence and machine learning in design of mechanical materials. *Materials Horizons*, 8(4):1153–1172, 2021. ISSN 2051-6347, 2051-6355. doi: 10.1039/D0MH01451F.

[34] S. S. Haykin. *Neural Networks: A Comprehensive Foundation*. Prentice Hall, Upper Saddle River, N.J, 2nd ed edition, 1999. ISBN 978-0-13-273350-2.

[35] O. Hubáček, G. Šourek, and F. Železný. Exploiting sports-betting market using machine learning. *International Journal of Forecasting*, 35(2):783–796, Apr. 2019. ISSN 01692070. doi: 10.1016/j.ijforecast.2019.01.001.

[36] S. Jeble, S. Kumari, and Y. Patil. Role of Big Data in Decision Making. *Operations and Supply Chain Management: An International Journal*, pages 36–44, Jan. 2018. ISSN 2579-9363. doi: 10.31387/oscm0300198.

[37] P. Jiao, K.-J. I. Egbe, Y. Xie, A. Matin Nazar, and A. H. Alavi. Piezoelectric Sensing Techniques in Structural Health Monitoring: A State-of-the-Art Review. *Sensors*, 20 (13):3730, July 2020. ISSN 1424-8220. doi: 10.3390/s20133730.

[38] G. C. Kahandawa, J. Epaarachchi, H. Wang, and K. T. Lau. Use of FBG Sensors for SHM in Aerospace Structures. *Photonic Sensors*, 2(3):203–214, Sept. 2012. ISSN 1674-9251, 2190-7439. doi: 10.1007/s13320-012-0065-4.

[39] N. Karthick and N. Ramalingam. Implementation of Railway Track Crack Detection and Protection. *International Journal Of Engineering And Computer Science*, June 2017. doi: 10.18535/ijecs/v6i5.47.

[40] D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization, Jan. 2017.

[41] A. Kollár. Betting models using AI: A review on ANN, SVM, and Markov Chain. Preprint, Open Science Framework, Mar. 2021.

[42] H. Kościelniak and A. Puto. BIG DATA in Decision Making Processes of Enterprises. *Procedia Computer Science*, 65:1052–1058, 2015. ISSN 18770509. doi: 10.1016/j.procs. 2015.09.053.

[43] M.-A. Lahmeri, M. A. Kishk, and M.-S. Alouini. Artificial Intelligence for UAV-Enabled Wireless Networks: A Survey. *IEEE Open Journal of the Communications Society*, 2:1015–1040, 2021. ISSN 2644-125X. doi: 10.1109/OJCOMS.2021.3075201.

[44] R. S. T. Lee. *Artificial Intelligence in Daily Life.* Springer Singapore, Singapore, 2020. ISBN 9789811576942 9789811576959. doi: 10.1007/978-981-15-7695-9.

[45] B. M. Leiner, V. G. Cerf, D. D. Clark, R. E. Kahn, L. Kleinrock, D. C. Lynch, J. Postel, L. G. Roberts, and S. Wolff. A brief history of the internet. *ACM SIGCOMM Computer Communication Review*, 39(5):22–31, Oct. 2009. ISSN 0146-4833. doi: 10.1145/1629607.1629613.

[46] P. W. Loveday. Development of piezoelectric transducers for a railway integrity monitoring system. In S.-C. Liu, editor, *SPIE's 7th Annual International Symposium on Smart Structures and Materials*, pages 330–338, Newport Beach, CA, Apr. 2000. doi: 10.1117/12.383154.

[47] O. Loyola-González. Black-Box vs. White-Box: Understanding Their Advantages and Weaknesses From a Practical Point of View. *IEEE Access*, 7:154096–154113, 2019. ISSN 2169-3536. doi: 10.1109/ACCESS.2019.2949286.

[48] B. D. Lund and T. Wang. Chatting about ChatGPT: How may AI and GPT impact academia and libraries? *Library Hi Tech News*, Feb. 2023. ISSN 0741-9058, 0741-9058. doi: 10.1108/LHTN-01-2023-0009.

[49] C. E. B. Maio. *Técnicas para monitoramento de integridade estrutural usando sensores e atuadores piezoelétricos*. Mestrado em Dinâmica das Máquinas e Sistemas, Universidade de São Paulo, São Carlos, Mar. 2011.

[50] J. McCarthy. What Is Artificial Intelligence? *Stanford University*, 2007.

[51] D. McPhillips. Home delivery of medications can help improve access, especially when time is tight. *CNN Health*, Dec. 2022.

[52] B. Meindl, N. F. Ayala, J. Mendonça, and A. G. Frank. The four smarts of Industry 4.0: Evolution of ten years of research and future perspectives. *Technological Forecasting and Social Change*, 168:120784, July 2021. ISSN 00401625. doi: 10.1016/j.techfore.2021.120784.

[53] D. Mhlanga. Industry 4.0 in Finance: The Impact of Artificial Intelligence (AI) on Digital Financial Inclusion. *International Journal of Financial Studies*, 8(3):45, July 2020. ISSN 2227-7072. doi: 10.3390/ijfs8030045.

[54] C. Milana and A. Ashta. Artificial intelligence techniques in finance and financial markets: A survey of the literature. *Strategic Change*, 30(3):189–209, May 2021. ISSN 1086-1718, 1099-1697. doi: 10.1002/jsc.2403.

[55] N. Muthukrishnan, F. Maleki, K. Ovens, C. Reinhold, B. Forghani, and R. Forghani. Brief History of Artificial Intelligence. *Neuroimaging Clinics of North America*, 30(4): 393–399, Nov. 2020. ISSN 10525149. doi: 10.1016/j.nic.2020.07.004.

[56] T. Nagayama and B. F. Spencer. Structural Health Monitoring Using Smart Sensors. *Newmark Structural Engineering Laboratory Report Series 001*, Nov. 2007. ISSN 1940-9826.

[57] I. E. Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Number v. 87 in Applied Optimization. Kluwer Academic Publishers, Boston, 2004. ISBN 978-1-4020-7553-7.

[58] M. Nielsen. *Neural Networks and Deep Learning*, volume 25. Determination press San Francisco, CA, USA, 2015.

[59] A. Pannu. Artificial Intelligence and its Application in Different Areas. 4(10), 2015.

[60] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[61] D. T. Pham and P. T. N. Pham. Artificial intelligence in engineering. 1998.

[62] I. Poola. How Artificial Intelligence in Impacting Real life Everyday. *International Journal for Advance Research and Development*, 2(10):96–100, 2017.

[63] PyTorch. PyTorch at Tesla - Andrej Karpathy, Tesla, Nov. 2019.

[64] J. R. Rabunal and J. Dorado, editors. *Artificial Neural Networks in Real-Life Applications*. Idea Group Pub, Hershey PA, 2006. ISBN 978-1-59140-902-1 978-1-59140-903-8 978-1-59140-904-5.

[65] M. Ramakrishnan, P. G. Poojari, M. Rashid, S. Nair, V. Pulikkel Chandran, and G. Thunga. Impact of COVID-19 pandemic on medicine supply chain for patients with chronic diseases: Experiences of the community pharmacists. *Clinical Epidemiology and Global Health*, 20:101243, Mar. 2023. ISSN 22133984. doi: 10.1016/j.cegh.2023.101243.

[66] S. Raschka. *Python Machine Learning*. Packt Publishing Ltd, 2015.

[67] S. Raschka, Y. H. Liu, V. Mirjalili, and D. Dzhulgakov. *Machine Learning with PyTorch and Scikit-Learn: Develop Machine Learning and Deep Learning Models with Python*. Packt Publishing Ltd, 2022.

[68] J. L. Ruiz-Real, J. Uribe-Toril, J. A. Torres, and J. De Pablo. Artificial Intelligence in Business and Economics Research: Trends and Future. *Journal of Business Economics and Management*, 22(1):98–117, Oct. 2020. ISSN 1611-1699, 2029-4433. doi: 10.3846/jbem.2020.13641.

[69] A. Sabato, C. Niezrecki, and G. Fortino. Wireless MEMS-Based Accelerometer Sensor Boards for Structural Vibration Monitoring: A Review. *IEEE Sensors Journal*, 17 (2):226–235, Jan. 2017. ISSN 1530-437X, 1558-1748, 2379-9153. doi: 10.1109/JSEN.2016.2630008.

[70] S. Sakena Benazer, M. Sheik Dawood, S. Karthick Ramanathan, and G. Saranya. Efficient model for IoT based railway crack detection system. *Materials Today: Proceedings*, 45:2789–2792, 2021. ISSN 22147853. doi: 10.1016/j.matpr.2020.11.743.

[71] S. Sankarasrinivasan, E. Balasubramanian, K. Karthik, U. Chandrasekar, and R. Gupta. Health Monitoring of Civil Structures with Integrated UAV and Image Processing System. *Procedia Computer Science*, 54:508–515, 2015. ISSN 18770509. doi: 10.1016/j.procs.2015.06.058.

[72] A. Sharma and N. Mehta. Structural Health Monitoring Using Image Processing Techniques-A Review. Aug. 2016.

[73] M. Sivakumar and N. M. Tyj. A Literature Survey of Unmanned Aerial Vehicle Usage for Civil Applications. *Journal of Aerospace Technology and Management*, 13:e4021, 2021. ISSN 2175-9146. doi: 10.1590/jatm.v13.1233.

[74] K. Smarsly, K. Lehner, and D. Hartmann. Structural Health Monitoring based on Artificial Intelligence Techniques. In *Computing in Civil Engineering (2007)*, pages 111–118. 2007.

[75] H. Sohn, C. R. Farrar, F. Hemez, and J. Czarnecki. A Review of Structural Health Monitoring Literature 1996 – 200. *Los Alamos National Laboratory, USA*, 1:16, 2003.

[76] J. Stewart. *Calculus*. Cengage Learning, Boston, MA, USA, eighth edition edition, 2016. ISBN 978-1-285-74062-1 978-1-305-27176-0.

[77] L. Sun, Z. Shang, Y. Xia, S. Bhowmick, and S. Nagarajaiah. Review of Bridge Structural Health Monitoring Aided by Big Data and Artificial Intelligence: From Condition Assessment to Damage Detection. *Journal of Structural Engineering*, 146, 2020.

[78] I. Sutskever, J. Martens, G. Dahl, and G. Hinton. On the Importance of Initialization and Momentum in Deep Learning. pages 1139–1147, 2013.

[79] TensorFlow. The Coca-Cola Company using TensorFlow for digital marketing campaigns (TensorFlow Meets), Aug. 2018.

[80] C. A. Thiels, J. M. Aho, S. P. Zietlow, and D. H. Jenkins. Use of Unmanned Aerial Vehicles for Medical Product Transport. *Air Medical Journal*, 34(2):104–108, Mar. 2015. ISSN 1067991X. doi: 10.1016/j.amj.2014.10.011.

[81] A. Tzounis, N. Katsoulas, T. Bartzanas, and C. Kittas. Internet of Things in agriculture, recent advances and future challenges. *Biosystems Engineering*, 164: 31–48, Dec. 2017. ISSN 15375110. doi: 10.1016/j.biosystemseng.2017.09.007.

[82] C. Verdouw, S. Wolfert, and B. Tekinerdogan. Internet of Things in agriculture. *CABI Reviews*, 2016:1–12, Jan. 2016. ISSN 1749-8848. doi: 10.1079/PAVSNNR201611035.

[83] S. Verma, R. Sharma, S. Deb, and D. Maitra. Artificial intelligence in marketing: Systematic review and future research direction. *International Journal of Information Management Data Insights*, 1(1):100002, Apr. 2021. ISSN 26670968. doi: 10.1016/j.jjimei.2020.100002.

[84] P. H. Winston. *Artificial Intelligence*. Addison-Wesley Pub. Co, Reading, Mass, 3rd ed edition, 1992. ISBN 978-0-201-53377-4.

[85] Z. F. Wu, J. Li, M. Y. Cai, Y. Lin, and W. J. Zhang. On membership of black-box or white-box of artificial neural network models. In *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*, pages 1400–1404, June 2016. doi: 10.1109/ICIEA.2016.7603804.

[86] X. Ye, T. Jin, and C. Yun. A review on deep learning-based structural health monitoring of civil infrastructures. *Smart Structures and Systems*, 24(5):567–585, Nov. 2019. doi: 10.12989/SSS.2019.24.5.567.

[87] K.-H. Yu, A. L. Beam, and I. S. Kohane. Artificial intelligence in healthcare. *Nature Biomedical Engineering*, 2(10):719–731, Oct. 2018. ISSN 2157-846X. doi: 10.1038/s41551-018-0305-z.