Addis Ababa University

Addis Ababa Institute of Technology

School of Electrical & Computer Engineering

Design of Internal Model Control for Wastewater Treatment Plant Using Artificial Neural Network: Case Study Kality Wastewater Treatment Plant

BY

Haileyesus Simegn

ID.No: GSE/9273/10

Name of The Advisor

Dereje Shiferaw (Ph.D.)

A Thesis Submitted to the School of Graduate Studies of Addis Ababa University in Partial Fulfillment of the Requirements for the Degree of Masters of Science in Control Engineering

November 2021

Addis Ababa, Ethiopia

Approval

The undersigned certify that they have read and hereby recommend to the Addis Ababa University to accept the Thesis submitted by Haileyesus Simegn, and entitled "**Design of Internal Model Control for Wastewater Treatment Plant Using Artificial Neural Network: Case Study Kality Wastewater Treatment Plant**", in partial fulfillment of the Requirements for the Degree of Masters of Science Control Engineering.

Dereje Shiferaw (Ph.D)	________________	________________
Name of Supervisor	Signature	Date
Lebsework Negash (Ph.D)	________________	________________
Name of External Examiner	Signature	Date
Nebiyu T. (MSc)	________________	________________
Name of Internal Examiner	Signature	Date
Bisrat Derebssa(Ph.D)	________________	________________
Name of Head of Department	Signature	Date

Declaration

I, the undersigned, declare that this thesis is my original work and has not been presented for a degree in this or other universities, and all sources of materials used for this thesis work have been fully acknowledged.

Haileyesus Simegn	________________	________________
Student Name	Signature	Date

Place: Addis Ababa Institute of Technology, Addis Ababa University, Addis Ababa, Ethiopia.
This Thesis has been submitted for examination with my approval as a university advisor

Dereje Shiferaw (Ph.D)	________________	________________
Advisor's Name	Signature	Date

Acknowledgment

First and foremost, I'd like to express my gratitude to the Almighty GOD. Then, I'd like to express my heartfelt gratitude to my advisor, Dr. Dereje Shiferaw, for his motivation, guidance, supervision, and moral support throughout this research work. Finally, I'd like to thank my family and friends for your inspiration and moral support during times of need and difficulty.

Abstract

High non-linearity of wastewater treatment processes and the influent wastewater dynamics dependency on the season and surrounding lifestyle, it is still an active research area that requires a new control methodology to optimize and improve the current controller performance. Hence, this research is designed to develop an artificial neural network plant and controller models for the Kality wastewater treatment plant using feedforward backpropagation algorithms. Using the Levenberg Marquard learning algorithm, the plant model was developed to estimate the dissolved oxygen in the biological reactor with six input variables. After several ANN architectures are tested, the best result is obtained with three layers of 43 neurons each, a sigmoid activation function on the hidden layers, and a linear activation function in the output node. Similarly, an inverse ANN (internal model controller) model is designed to control the oxygen transfer coefficient with six input variables. The final optimal inverse neural network model structure is three layers with 47 neurons per layer, a sigmoid activation function on the hidden layers, and a linear activation function in the output node with the Levenberg Marquard learning algorithm. The simulation results also demonstrated that the performance of an ANN internal model controller outperforms that of a traditional PI controller in both transient and steady-state conditions. In the case of the PI controller, the plant response takes 19.4 seconds to reach a steady-state, whereas the ANN internal model controller requires only 16 seconds. This result shows that the ANN controller improves settling time by 21.25 percent. When the rise time of the plant response is evaluated, the ANN internal model controller improves the rise time by 67% compared with the PI controller. Since, ANN plant and controller models developed using process state variables, which provide a significant opportunity for disturbance rejection, transient and steady-state performance improvement.

KEYWORDS: *Artificial Neural Network, Feedforward Backpropagation Algorithm, Internal Model Control,* **Levenberg**-*Marquardt Algorithm,* ***PI****, Wastewater Treatment Plant.*

Contents **Page**

List of Figures

List of Tables

List of Acronyms and Abbreviation

AI	Artificial Intelligence
ANN	Artificial Neural Network
BOD	Biological Oxygen Demand
BOD5	Biological Oxygen Demand Five
BPA	Backpropagation Algorithm
BSM	Benchmark Simulation Model
COD	Chemical Oxygen Demand
DO	Dissolved Oxygen
Doi	Dissolved Oxygen Input
DSS	Dissolved Suspended Solid
ELU	Exponential Linear Unit
F/M	Food to Mass Ratio
HRT	Hydraulic Retention Time
IMC	Internal Model Control
MATLAB	Matrix Laboratory
mg	Milligram
MLSS	Mixed Liquor Suspended Solids
MSE	Mean Square Error
N/A	Not Available
NN	Neural Network
OTC	Oxygen Transfer Coefficient
PI	Proportional Integral
PID	Proportional Integral Derivative
SPSS	Statistical Package for the Social Sciences
SRT	Solids Retention Time
TP	Total Phosphorous
TSS	Total Suspended Solids

List of Symbols

b_j	Bias
d	Disturbance
G(s)	Actual Plant Model
$G_m(s)$	Nominal Plant Model
H	Hydrogen
H_2O	Water
J	Loss Function
k	Slop
N	Nitrogen
N(s)	Measurement Noise
NH_3	Ammonium
NO_2	Nitrate
NO_3	Nitrite
O_2	Oxygen
P	Plant
R	Set point
R^2	Coefficient of Determination
U(s)	Control Signal
w_{ij}	Synaptic Weight
xi	Input
y_i	Output
$\varphi(.)$	Activation Function
α	Preceding Weight Rate
η	Learning Rate
ε	Error
Σ	Summation

CHAPTER ONE: INTRODUCTION

1. Introduction

According to [1], 44.7 percent of Ethiopian households get their drinking water from unsafe wells, springs, and surface water. This is partly due to unplanned urbanization and scattered rural communities. The wastewater produced from unplanned industrialization and urbanization contaminated surface and groundwater throughout the country. Wastewater is usually classified based on the source emanate that is wastewater emanates from an urban area called sewage water whereas wastewater emanates from an industrial activity called industrial wastewater. In addition to the above source of wastewater, there is also rainwater storm which plays a key factor in the quality of wastewater significantly. Inadequate wastewater treatment and disposal systems in the country create severe environmental, health, and economic problems for both rural and urban residents. Miss management of wastewater systems at the country level, specifically in urban areas, has a serious negative impact on the long-term survival of all living things in the urban center and their neighborhoods. The aforementioned factors, as evidenced by the authors' experience in Addis Ababa City, prompted me to seek an alternative more efficient, effective, and least-cost wastewater treatment plant control method.

Several factors influence the effectiveness and efficiency of wastewater treatment plants and hence different variables in the activated sludge process are controlled based on the treatment plant objectives to generate the required effluent water quality. The dissolved oxygen concentration in the biological wastewater treatment tank is one of the major essential elements that must be managed to optimize wastewater treatment plant performance. Other factors such as mixed liquor suspended solids (MLSS), oxygen transfer coefficient, hydraulic retention time (HRT), solids retention time (SRT), and food to mass ratio (F/M) are also managed in addition to dissolved oxygen concentration to improve treatment performance. Different wastewater treatment plants used different criteria to evaluate the quality of effluent water, such as the amount of oxygen required by microorganisms to degrade organic materials (BOD), biological oxygen demand five (BOD5), total suspended solids (TSS), total phosphorous (TP), and chemical oxygen demand (COD) and others [2].

Wastewater treatment plant operators prioritized new, practically implementable, and reliable control strategies for wastewater treatment plants to improve current wastewater treatment plant performance and energy efficiency while ensuring process stability due to the continuous rise in wastewater treatment costs over time[3]. Researchers like [4] describe wastewater treatment systems as non-linear and dynamic systems that necessitate reliable and robust control strategies to achieve the desired efficient water quality. Hence, it necessitates robust control to improve the performance fluctuation of wastewater treatment plants.

The needed dissolved oxygen level in the biological wastewater treatment plant, according to the researcher [5] can be managed using a self-organized radial base function neural network model predictive control to satisfy effluent water quality while using the least amount of energy. The author employs a self-organized radial base function to forecast responses across defined time horizons and then applies these projected values to an empirical optimization process that minimizes the cost function in the region of the optimal control signal at each sampling interval. One of the challenges revealed in this research is that the suggested control approach works only for limited prediction and control horizons, and above the predictive horizons, integrated absolute errors dramatically rise. The second concern is the control approach does not account for nitrate accumulation in the biological treatment tank, which harms the performance of wastewater treatment plants. Other researchers, such as [6] suggested that PID-based hierarchical control for nitrogen removal, with the ratio of nitrogen removed from the slugging process to energy required as a control variable and dissolved oxygen as a manipulated variable. These forms of control necessitate the use of a plant model for parameter tuning, which makes it difficult to describe deterministically the nonlinear features of the bioreactor. It also faces tuning complications when using a PI controller with fixed parameters. Recent studies, such as [7] use a feedforward neural network to create an internal model control for linear and non-linear systems, including minimum phase and non-minimum-phase and get very good set-point tracking results. The High-frequency component in the inverse model is practically solved using a low pass filter.

The artificial neural network is distinguished by the following unique characteristics: it learns from experience, generalization based on training, real time implementation, nonlinear system modeling, and estimation capability [7]. These characteristics serve as a foundation for developing a biological wastewater treatment and its corresponding internal model controller models for this research work. Artificial neural network models were developed based on the data gathered from the Kality wastewater treatment plant. After model development, the ANN-

IMC performance was evaluated using mean square error (MSE) performance evaluation criteria. This chapter introduces the overall idea of the research which is further elaborated in subsequent chapters. Under this chapter very important topics such as the research aims, the objective of the study, research question, and the scope of the study were elaborated one after another keeping logical flow.

2. Statement of the Problem

Many scholars have proposed various dissolved oxygen concentration control methods up to this point; however, due to the high non-linearity of wastewater treatment processes and the influent wastewater dynamics relies on the season and surrounding lifestyle, as stated in works of [4], it is still an active research area which requires a new control methodology to optimize and improve the current control performance. The researchers, such as [6] suggested proportional integral derivative (PID) based hierarchical control for nitrogen removal from wastewater, with the ratio of nitrogen removed from the slugging process to energy required as a control variable and dissolved oxygen as a manipulated variable. These forms of control necessitate a plant model for parameter tuning. It also faces tuning complications when using a PI controller with fixed parameters. The optimal dissolved oxygen in the biological wastewater treatment, according to the researcher [5] can be controlled using self-organized radial base function neural network model predictive control to satisfy effluent water quality while using the least amount of energy. One of the challenges revealed in this research is that the suggested control approach works only for limited prediction and control horizons, and above the predictive horizons, integrated absolute errors dramatically increase. The second concern is the control approach does not account for nitrate accumulation in the biological treatment tank, which harms the wastewater treatment plant performance.

The promising characteristics of artificial neural networks, such as learning from experience, generalization based on training, real-time implementation, nonlinear system modeling, and estimation capability, act as a foundation for modeling nonlinear systems [7]. Hence, this research was designed to develop an artificial neural network plant and controller models for the Kality wastewater treatment plant using feedforward backpropagation algorithms. A trained artificial neural network plant model estimates the dissolved oxygen content for new data. The amount of oxygen transfer rate was manipulated till the set point dissolved oxygen level was achieved.

3. Research Objective

3.1 General Objective

The general objective of this research is to develop plant and internal model control models using an artificial neural network for the Kality wastewater treatment plant and hence improve plant performance with minimum energy requirement.

3.2 Specific Objectives

The specific objectives of this research are:

i. To develop an artificial neural network biological wastewater treatment plant model using feedforward and backpropagation algorithm

ii. To design an artificial neural network controller based on internal model control principles with feedforward and back propagation algorithm

iii. To compare the artificial neural network controller performance with PI controller

4. Research Question

1) What are the factors that affect dissolved oxygen concentration in biological wastewater treatment plant?
2) How to capture wastewater treatment plant processes' nonlinearity in the artificial neural network models?
3) What are the deficiencies of the conventional PI controller in biological wastewater treatment plant dissolved oxygen control?

5. Significance of the Research

The artificial neural network internal model control is a relatively new method that enables overcoming the constraints of classic proportional integral (PI) controllers such as tuning and plant model requirements. As a result, the data-driven ANN–IMC controller for wastewater treatment plants enhances controller performance in both transient and steady-state circumstances. Furthermore, water pollution and soil contamination concern in developing countries like Ethiopia rapidly increases over time due to the country's rapid urbanization and industrial growth at this time. Hence, this research arrives at the right time for wastewater treatment operators, municipal administrators, the industrial sector, and future researchers. In addition to the foregoing, it has also a positive benefit for health, economic, land, and potable water pollution reduction.

6. Scope of the Research

The Kality wastewater treatment plant is located at Akaki Kality Sub-City, Addis Ababa, Ethiopia. The wastewater treatment operations are made up of a succession of separate activities that lead to the ultimate effluent output point via physical, biological, and chemical treatment. The goal of this research was to create an artificial neural network plant model for a wastewater treatment plant, as well as an artificial neural network internal model controller for a biological wastewater treatment bioreactor. The processes' performance was enhanced by adjusting the quantity of oxygen supplied while controlling the amount of dissolved oxygen in the aerobic reactor to the appropriate level. Historical operational data from 2016 to 2018 was collected by accounting temporal and seasonal dynamics of influent wastewater.

7. Limitations of the Research

The quality of the research output is strongly dependent on the availability and quality of data collected from wastewater treatment plants. Many factors influence operational historical data availability and quality, including laboratory test equipment calibration, data organization, and storage system, laboratory test completeness and stakeholder awareness, and other environmental factors. The researcher was compelled to fill the missing data by interpolation due to a lack of complete data.

8. Organization of the Thesis

This research is divided into five chapters, which are presented sequentially to maintain a logical flow of the contents. The first chapter explains the broad concept of the research, which is expanded in future chapters. The second chapter provides a theoretical and empirical literature review of internal model control, wastewater treatment processes, artificial neural network types, elements, application, and especially control application on the wastewater treatment process, which is covered in this section. There is also a brief introduction to the Kality Wastewater

Treatment Plant, and a brief description of the influence of biological wastewater treatment process performance. The third chapter is titled under materials and methods, and it contains a brief explanation of the research methods, software deployed for this research, data gathering procedures, data organization and model development, and data analysis approach. The neural network training, validation, and testing process, as well as the data classification ratio, are described here. Chapter four covers the results and discussion section, which includes a summary of the simulation results in the form of tables, graphs, and charts. Important information is extracted, interpreted, and summarized in the discussion section. The suggested ANN-IMC controller performance is compared with the PI controller both in steady-state and transient situations. Finally, chapter five focuses on conclusion research findings and making relevant recommendations based on the findings, as well as suggesting future research areas that are not addressed in this study.

CHAPTER TWO: LITERATURE REVIEW

2.1 General Overview of Wastewater Treatment Processes

A municipal wastewater treatment plant generally consists of three types of treatment: primary, secondary, and tertiary treatment. Since the characteristics and content of wastewater components are strongly related to the lifestyle of the region, the sequence of treatments utilized is generally site-specific. The primary goal of primary treatment is to eliminate the majority of the suspended particles found in raw wastewater. Screening, de-gritting, sedimentation, and other site-specific activities may be included in primary treatment. Following primary treatment, the majority of the suspended particles are removed, and the wastewater is transferred to secondary treatment, which is targeted at removing soluble organic matter. Secondary treatments are often biological methods that remove biodegradable organic materials. The biological treatment is divided into anaerobic and aerobic treatment portions containing various microorganisms. Microorganisms in the anaerobic process do not require oxygen to function, but in the aerobic process, microorganisms require oxygen to respire and break down organic materials in the wastewater. As a result, controlling the proper quantity of oxygen in aerobic treatment has a dual benefit: it improves plant performance while also optimizing energy requirements. Finally, the secondary treatment effluent is directed to the final tertiary treatments, which may involve disinfection and other activities targeted at eliminating any remaining bacteria that were not removed after the secondary treatment [8].

2.2 Kality Wastewater Treatment Plant Overview

Kality wastewater treatment plant is located in Addis Ababa, Ethiopia, at 8^0 55` 11.93`` N and $45^0$ 55` 20.13``S. Based on its design capacity, the treatment plant treats 7,600 cubic meters of wastewater using 3,500 kg biochemical oxygen demand per day. It is simple earthen basins where wastewater is treated by removing particulate matter and biologically decomposing settled solids.
The treatment plant is consists of several treatment phases, including screening and degritting channels, distribution cell, stabilization pond, and treated water recirculation. According to [9], the screening and degritting channels are used to treat the effluent of 200,000 equivalent inhabitants. This piece of equipment is made up of two canals that operate in parallel and can be isolated using a coffer dam. Each channel has an inclined screen (i.e. 65 degrees) with a width of

2.5m and a free space of 25mm between bars. Each degritting channel is 10.5m long and the channel's outlet is fitted with a linear weir, allowing a constant degritting speed to be maintained. A distribution cell allows for the partial or complete feeding of all stabilized ponds. The totalizing flow meter measures the volume of water allowed to recirculate in each treatment path. The treatment plant has two rows of paralleled biological treatment, each with four stabilization ponds, the main characteristics of which are depicted in Table 2.1. The treatment plant is equipped with three archimedes screws, each of which can raise an incoming flow of 80l/s and has the following specifications: max first phase recycling capacity is 13,824 m^3/day, 800mm diameter, 51 rpm, 7.5 and 10 hours power motor. At maximum flow rate, the hydraulic retention time of the wastewater in the stabilization pond was approximately 30 days, and the effluent from the ponds flowed by gravity and was eventually discharged to the Little Akaki River.

Table 2.1: Stabilization pond characteristics

Stabilization pond	Capacity (m^3)	Retention time @ 7500^3/day	Average height(m)	Area M^2
Facultative pond	95,149	12.7	2.04	46,493
Maturation pond	44,176	5.9	0.95	46,577
Polishing pond I	44,586	5.9	1.02	43,563
Polishing pond II	44,237	5.9	0.95	46,482
Total	228,148	30		183,115

2.3 Fundamental Biological Wastewater Treatment

The goal of biological wastewater treatment is to eliminate biodegradable organic materials from wastewater. A wide range of microorganisms present in wastewater will interact with organic molecules, utilizing them as either energy sources or as a material source for the production of new cellular components. Without treatment, organic waste would wind up in receiving bodies of water, where bacteria would proliferate unchecked, killing many aquatic creatures and transmitting diseases. Because individual measurements of organic components in wastewater are impracticable, the concentration of organic matter is measured based on a feature that all organic matter has in common, such as the ability to be oxidized and the presence of organic carbon. Measurements of biological oxygen demand and chemical oxygen demand are commonly used to indicate the organic matter in wastewater based on these common properties. The overall content of organic materials in wastewater is measured by chemical oxygen demand. The oxygen required by microorganisms as they remove organic materials from wastewater is referred to as biological oxygen demand. Biological oxygen demand is always lower than chemical oxygen demand for two reasons: first, some of the organic matter in wastewater may not be biodegradable, and second, biological oxygen demand only measures the oxygen consumption associated with substrate degradation, whereas part of the substrate is converted to new microorganisms [9].

Aerobic, anaerobic, and biological nutrient removal processes are the three primary components of biological waste treatment systems. Aerobic treatment happens in the presence of free molecular oxygen, with the primary goal of removing organic materials. In contrast to the aerobic process, the anaerobic process occurs in the absence of oxygen. The biological nutrient removal process is intended to remove nitrogen and phosphorus from wastewater [8] and [10]. Biological nutrient removal occurs by the oxidation of ammonia to nitrite and subsequently to nitrate, a process known as nitrification, which is followed by the conversion of nitrate to nitrogen gaseous, a process is known as de-nitrification. Bacteria, such as those from the genus Nitrosomonas, convert ammonia to nitrites as shown in the following chemical reaction [11].

$$2NH_4 + N + 3O_2 \; = \; 2NO_2 - N + 4H+ + 2H_2O + Energy \tag{2.1}$$

The oxidation of nitrites into nitrates occurs by the action of bacteria, such as those from the genus nitrobacteria, expressed by:

$$2NO_2 --N + O_2 = \; 2NO_3 \; - N + Energy \tag{2.2}$$

2.4 Factor Affecting Oxygen Transfer Rate

According to [12] the contaminants in wastewater decreases the rate of oxygen transfer and the solubility of oxygen. The oxygen transfer rate of aeration equipment to be installed in a wastewater treatment plant is commonly determined in various operating circumstances. As a result, it is critical to be able to define the elements that impact the oxygen transfer rate to estimate the transfer rate under operational circumstances based on the results of tests conducted under standardized conditions. Many factors have a significant impact on oxygen transfer rates such as when the temperature rises, the saturation of concentration decreases, implying a decrease in the oxygen transfer rate. Similarly, the impact of altitude is evident in the oxygen saturation concentration. The higher the altitude, the lower the air pressure and, consequently, the saturation concentration is lower. Under steady-state circumstances, the higher the dissolved oxygen concentration maintained in the reactor, the lower the oxygen transfer rate, and the aerator properties and reactor geometry also impact the oxygen transfer rate.

2.5 Internal Model Control

Based on the [13] investigation, Internal model control (IMC) is a common model-based control strategy in industry processes control because of its simple structure, fine disturbance rejection capabilities, and robustness. IMC may be used with both linear and nonlinear systems, and its design and implementation for an open-loop stable system are straightforward. The modified IMC structure and mathematical derivation according to [14] research work was illustrated in the subsequent section.

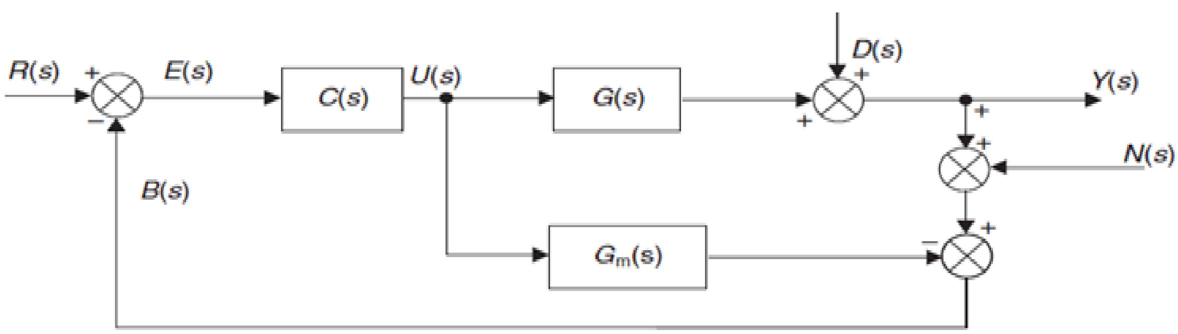


Figure 2. 1: IMC structure

Figure 2.1 noted that R(s) is the desired value, U(s) is the controlled signal feed to both the actual plant and its nominal model, G(s) is the actual plant, and $G_M(s)$ is the nominal plant model, D(s) is the unknown disturbance input, N(s) is the measurement noise, Y(s) is the output, and B(s) is the difference between the actual and the nominal model plant output.

$B(s) = (G(s) - G_M(s))\ U(s) + N(s) + D(s)$ (2.3)

If N(s) and D(s) are both zero, then B(s) represents the difference between plant and model output. If $G(s) = G_M(s)$, B(s) measures just the disturbance and measurement noise. If $G(s) = G_M(s)$ and N(s) & D(s) are both zero, then B(s) is also zero, indicating that the control system is in an open loop. This is the case when there is no ambiguity. However, if G(s) differs from G_M(s), and N(s) and D(s) are non-zero, then B(s) is non-zero; B(s) represents the processes' uncertainty.

$$Y(s) = G(s)\,U(s) + D(s) \tag{2.4}$$

$$B(s) = Y(s) + N(s) - G_M(s)\,U(s) \tag{2.5}$$

$$U(s) = C(s)\,R(s) - B(s) \tag{2.6}$$

When we remove B(s) and U(s) from equations (2.5) and (2.6), we get the following result:

$$Y\,(s) = \frac{G(s)C(s)R(s)}{1+G(s)C(s)-Gm(s)} + \frac{(1-Gm(s)C(s))D(s)}{1+G(s)C(s)-Gm(s)} - \frac{G(s)C(s)N(s)}{1+G(s)C(s)-Gm(s)} \tag{2.7}$$

S(s) is the sensitivity function that connects Y(s) and D(s) when R(s) = N(s) = 0.

$$S(s) = \frac{Y(s)}{D(s)} = \frac{(1-Gm(s)C(s))}{1+G(s)C(s)-Gm(s)} \tag{2.8}$$

Then complementary sensitivity function can be computed from the sensitivity function as follows.

$$\mathrm{T(s)} = 1 - \mathrm{S(s)} = \frac{\mathrm{G(s)C(s))}}{1+\mathrm{G(s)C(s)}-\mathrm{Gm(s)}} \tag{2.9}$$

As a result, when the measurement noise N(s) is equal to zero, the output function is as follows:

$$\mathrm{Y(s)} = \mathrm{T(s)}R(s) + S(s)D(s) \tag{2.10}$$

If T(s) is one when S(s) equals zero, there is perfect set-point tracking, and if T(s) is zero, there is perfect disturbance rejection. Both criteria are satisfied when the plant transfer function equals the plant model transfer function and the controller function equals the inverse of the plant model transfer function.

2.6 Internal Model Control Property

The following internal model controller attributes are described in the work of [13]. The first IMC property is dual stability: suppose that both the plant and the controller are input-output stable and that the model accurately describes the plant. The closed-loop system will then be stable in terms of input-output. Perfect control, the second IMC attribute, is achieved when the inverse of the operator defining the plant model exists, is used as the controller, and the closed-loop system is input-output stable with this controller. The final property is zero offsets: assume that the inverse of the steady-state model operator exists, that the steady-state controller operator is equal to this, and that the closed-loop system with this controller is input-output stable. Then, for asymptotically constant inputs, offset-free control is achieved.

2.7 Internal Model Control with Artificial Neural Network

One of the most ideal model-based controllers that can be readily constructed using a neural network is the internal model control. Two neural network models, one for the plant model and one for the controller, should be created. The discrepancy between the plant output and the model output, as illustrated in Figure 2.2, is sent back into the control system. The plant and controller are considered to be input-output stables in this technique. The neural emulator (M) for plant (P) is sought first, followed by the neural controller (C) that has been taught to act as the plant's inverse, and lastly, the general structure is examined with filter (F) to overcome high-frequency component from inverse model [15].

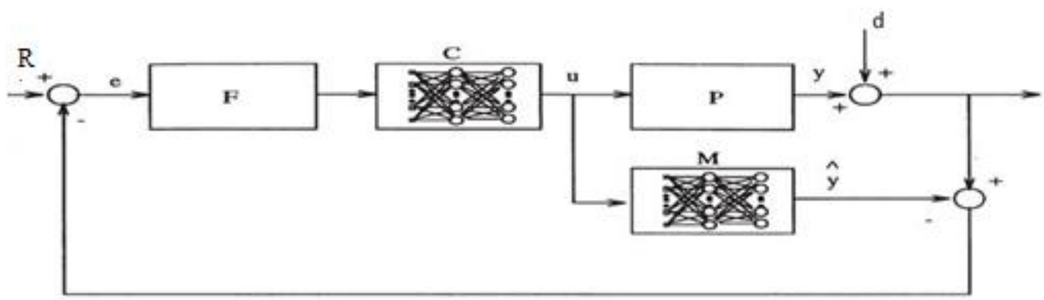


Figure 2.2: ANN internal model control scheme

In Figure 2.2, R is set-point (desired value), u is the neural controller output signal that is provided to both the plant (P) and the neural plant model (M), y is the plant output, and y^ is the neural network plant model output, and d is the unknown disturbance input.

2.8 Benchmark Simulation Model

A benchmark simulation model (BSM), according to [16], is a simulation environment that defines a plant layout, a simulation model, influent loads, test procedures, and evaluation criteria. Compromises were sought for each of these items to combine clarity with realism and accepted standards. Once the simulation code has been validated by the user, any control strategy can be used, and performance can be evaluated using a predefined set of criteria. The plant model is made up of five activated sludge reaction tanks, two tanks dedicated to the anoxic treatment, and three aerobic tanks. A secondary settler follows the active sludge reactor. To put the benchmark to the test, a basic control strategy is proposed that aims to control the dissolved oxygen in the 5^{th}

compartment by manipulating the oxygen transfer coefficient and the nitrate level in the second anoxic compartment by manipulating the internal recycle flow rate, as shown in Figure 2.3.

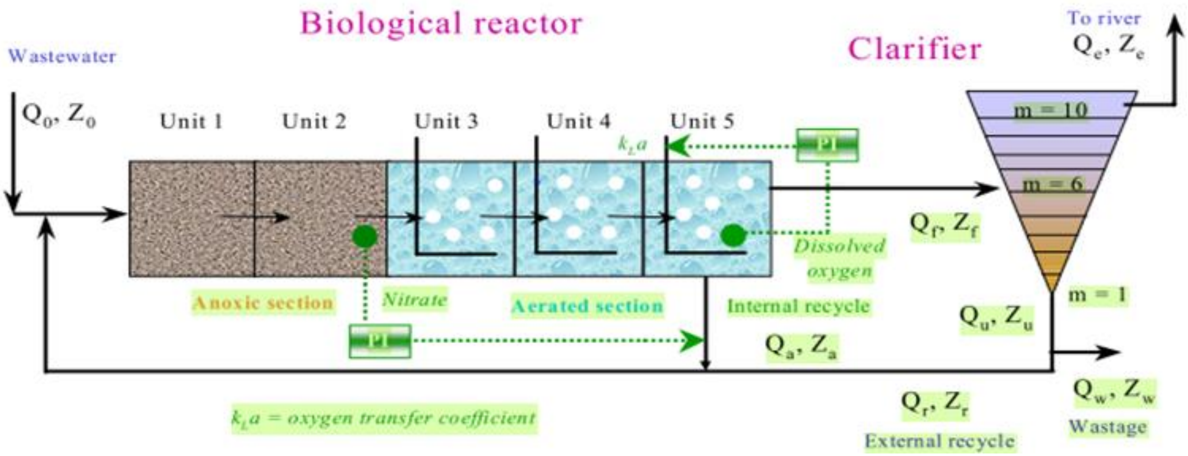


Figure 2.3 General overviews of BSM[16]

BSM identifies soluble inert organic matter, biodegradable substrate, active heterotrophic biomass, active autotrophic biomass, particulate inert organic matter, particulate product arising from biomass decay, oxygen, nitrate, nitrite nitrogen, nitrogen, soluble biodegradable nitrogen, particulate biodegradable organic nitrogen, and alkaline as operational variables. In addition to these, BSM also identified and suggested the following control variables for the benchmark plant's deployment of a new control strategy.

- Internal flow recirculation rate
- Return sludge flow rate
- Wastage flow rate
- Anoxic/aerobic volume
- Aeration intensity
- External carbon source flow rate at each thank
- Influent distribution by use of step feed
- Distribution of internal flow recirculation
- Distribution of internal sludge flows

2.9 Machine Learning

Machine learning is a subset of artificial intelligence that deals with the development of mathematical models to aid in the understanding of data. Learning enters the fray, as described in [16] when we offer these models adjustable parameters that may be changed to observed data; in this manner, the program can be said to be "learning" from the data. Once fitted to previously seen data, these models may be used to estimate and comprehend features of newly observed data.

2.9.1 Supervised Machine Learning

Supervised learning entails modeling the link between measured data characteristics and a label associated with the data; once this model is established, it can be used to apply labels to fresh, unknown data. This is further classified into classification tasks and regression tasks: in classification, the labels are discrete categories, whereas, in regression, the labels are continuous variables [16].

2.9.2 Unsupervised Machine Learning

Unsupervised learning is the process of modeling the features of a dataset without using labels, and it is frequently referred to as "letting the datasets speak for themselves". Clustering and dimensional reduction are performed with this method. Clustering techniques find discrete groupings of data, whereas dimensional reduction algorithms look for more concise representations of the data [16].

2.9.3 Reinforcement Machine Learning

Reinforcement learning is the process of learning an input-output mapping through ongoing contact with the environment to reduce a scalar index of performance. Reinforcement learning is intermediate between supervised and uncontrolled learning. When just partial labels are available, semi-supervised learning approaches are frequently helpful.

2.9.4 Offline Learning

After all training data subsets have been presented to the network; the parameters of the offline learning model are updated regularly using epoch or batch techniques. Because the discrepancy between the network response and the intended value is computed and accumulates until the final data point is given into the network, this training approach necessitates at least one training session for model parameter changes [17]. Because offline learning weight updates occur after the network learns all sorts of data characteristics and more accurately depict system behavior in test subsets, they are preferred to instantaneous model parameter changes based on single samples. However, because model parameter changes take a long time before all data points are available in the network; this form of learning is not suited for real-time applications.

2.9.5 Online Learning

Online learning model parameter changes, in contrast to offline learning, are based on the per-epoch approach once each sample is submitted to the network. This form of learning is appropriate for systems with dynamic and changing system behavior [17]. Online learning is appropriate for real-time learning systems; nevertheless, model parameter updates per sample without a thorough understanding of system behavior render the learning system unworkable in the actual world.

2.10 Artificial Neural Network

According to[18] definition, a neural network is a massively parallel distributed processor made up of basic processing units that has a natural proclivity for storing and making available experience information. It is similar to the brain in two ways: first, the network acquires knowledge from its surroundings through learning processes, and second, inter-neuron connection strengths, known as synaptic weights, are utilized to retain the learned knowledge. The adaptive nature of these networks is a key aspect, with ‘learning by example' replacing programming in problem-solving. A learning algorithm is a technique utilized to execute the learning process, and its role is to alter the synaptic weights of the network in an ordered way to achieve the desired design aim. This characteristic makes such computational models particularly desirable in application fields where there is little or no understanding of the problem to be solved and training data is easily available. In general, neural networks may be viewed as a "black box" device that receives inputs and generates results.

Most research classifies neural networks as feedforward and backward (recurrent) neural networks; however, some material includes meshes and hybrid forms of networks. In a feed forward neural network, signals flow from the input neuron to the hidden neuron and then to the output neuron solely in the forward direction, but in a recurrent neural network, signals can flow both forward and backward direction.

2.10.1 Forward Neural Network

The common forward and recurrent neural network models are listed in the works of [19] as follows.

- Adeline and Madeline
- Perceptron
- Backpropagation network
- Radial base function
- General regression network
- Learning vector quantization
- Probabilistic neural network
- Fuzzy neural network

2.10.2 Recurrent Neural Network

- Hopfield network
- Boltzmann machine
- Kohonen's self-organizing feature map
- Recirculation network
- Adaptive resonance theory

2.11 Artificial Neural Network Elements

Feature: A feature is a system learning input variable or an independent variable. A basic machine learning project may employ a single feature, whereas a more complex machine learning project may employ millions of features. The features are chosen based on their significance and independence in terms of information content, as well as their influence on projected output variables. The second ANN element is a label that is a variable predicted by the artificial neural model utilizing feature values and suitable neural nets are referred to as labels. The label may represent the future dissolved oxygen output of anaerobic thank or oxygen transfer coefficient, or anything else. The third ANN element is biased. Bias is input to the node in addition to features, bj in the following equation, is the node's internal threshold. This is a randomly chosen value that governs the activation node[20].

$$Y_i = \sum W_{ij}X_i + b_j \tag{2.11}$$

Where Y_i is a node's output, W_{ij} is a model parameter between nodes i and j, X_i is the input features, and b_j is biased to node j. The degree of the bias b_j and the input characteristics influence the total activation node. When b_j is big or positive, the node has a strong bias, which prevents node-firing. When bj is 0 or negative, the node has a low bias, no node-firing.

Model parameter (weight) is the fourth ANN component: The weights factors W_{1j}, W_{2j}, and W_{nj} connect feature one to the next layer of all nodes. This is analogous to biological neurons' varied synaptic strengths. Weights are changed at varying frequencies depending on the learning method and model assessment criteria used until the error function reaches a predefined value. Every input (x_1, x_2... x_n) is multiplied by the weight factor (W_{1j}, W_{2j}...W_{nj}). (W_{ij} x_i) tends to excite the node if the weight factor is positive. If the weight factor is negative, the node is inhibited by (W_{ij} x_i). Equation 2.12 demonstrates the relation between weight, inputs, and outputs of the ANN model.

$$Yi=\sum W_{ij}X_i+b_j \tag{2.12}$$

Where Y_i is a node's output, W_{ij} is a model parameter shared by nodes i and j, X_i is an input feature, and bj is biased toward node j. Model parameters begin with a starting value and are modified until they reach their optimal value. Weight initialization approaches based on statistical distribution data and activation functions implemented in the network were proposed by various researchers[21].

The activation function is the last ANN component: An activation function is a crucial component of an artificial neural network that has a significant impact on the network's performance. The activation function is a node's processing unit that takes distinct weighted inputs from different nodes and turns them into the activation function's output range. A nonlinear activation function was added to the network to represent nonlinearity in the input-output relationships. There are several types of activation functions available for various network topologies and applications. The following are the detailed activation function lists and their properties as mentioned in the works of [22].

Linear activation function: it has a straight-line equation, the activation is proportionate to the input, and it is provided by:

$$f(x_i) = k\, x_i \tag{2.13}$$

Where xi is the input of the activation function f() on the i^{th} channel and k is a constant, f(xi) is the output of the net. The derivative of xi is k. That is, every change in inputs only generates a constant rate of change in output. The linear activation function solely represents the linear input-output relationship. If the activation functions of all layers are linear, the entire network is equal to a single layer with a linear activation function.

The sigmoid function is a nonlinear activation function whose output is typically between zero and one. Although it is monotonic and its output is very steep, small changes in the input result in large changes in output when the input is close to 0, but towards either end of it, the output tends to be 0 or 1, respond very little to the input, that is to say, the gradient at this region is going to be very small, even close to 0, this is referred to as "vanishing gradients".

$$f(x_i) = e^{xi} / (1+ e^{xi}) \tag{2.14}$$

The **tangent hyperbolic** (tanh) function is likewise nonlinear, but unlike the sigmoid function, its output is zero-centered, and it squashes a real-valued integer to the range [-1, 1], so there's no need to worry about activation blowing out. Its gradient is greater than the sigmoid function; yet, it suffers from the problem of vanishing gradients.

$$\tanh xi = \frac{\sinh xi}{\cosh xi} = \frac{e^{xi} - e^{-xi}}{e^{xi} + e^{-xi}} \tag{2.15}$$

The ReLu activation function has gained popularity in recent years; it is typically used in the hidden layers of ANNs, particularly in virtually all convolutional neural networks or deep learning. ReLu appears to be linear, but it is nonlinear since the output is 0 when x_i is less than 0, and it avoids and corrects the vanishing gradient problem. It is, nevertheless, less computationally costly than sigmoid and tanh. Because ReLu has no upper limit, it has the potential to blowout the activation. In addition to the problems mentioned above, if the ReLu gradient is 0 for negative inputs, the weights will not be adjusted during descent, and the neurons in that state will stop responding to variations in inputs, becoming stuck in a perpetually inactive state which is known as the "Dying ReLu" problem.

$$f(x_i) = \max(0, x_i) = \begin{cases} x_i, & x_i > 0 \\ 0, & x_i < 0 \end{cases} \tag{2.16}$$

Another form of activation function based on ReLu is the exponential linear unit (ELU). It speeds up learning and solves the vanishing gradient problem. Because the activation of negative inputs is extremely near to zero, and their gradients are similar to the natural gradient.

$$f(x_i) = \begin{cases} x_i, & xi > 0 \\ \alpha_i(e^{xi} - 1), & x_i \leq 0 \end{cases} \tag{2.17}$$

2.12 Backpropagation Algorithm

Forward and backpropagation techniques are used consecutively in the training of multilayer perceptron networks. Forward propagation begins its executions by inserting a sample from the training set into the network inputs and continues layer by layer until the appropriate outputs are produced. Forward propagation seeks simply to extract the predicted value from the network, taking into consideration only the present network parameters and node biases, which will remain unchanged during the execution of this step. Following that, the network output node outputs are compared to the corresponding available intended value to generate the error signal. The error signal generated in the output nodes is then backpropagated layer by layer until the final layer parameter and bias change occurs. Backpropagation, unlike the feedforward method, alters (updates) the network parameters and biases of all network nodes. In general, the application of the forward and backward phases successively allows the network parameters and node biases to be automatically modified in each iteration, resulting in a steady reduction in the total of the errors created by the network replies for the desired responses [11].

2.13 Application of Neural Network

2.13.1 Classification

Classification is a supervised learning approach that can predict the label for unpredictably labeled data. We wanted to categorize unlabeled data points using the supplied set of label inputs features points. Let it be the (x, y) location of points on the plane illustrated in Figure 2.4 given the two-dimensional data collection. The color of the points represents the labels for the two classifications for each point. Based on the attributes and labels provided, we can build a model capable of recognizing the new point that should be labeled ‘blue' or ‘red.'

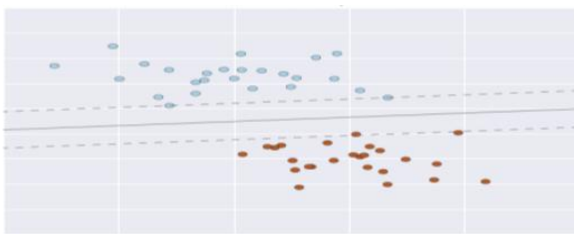


Figure 2.4: Simple classification model (adapted from [16])

2.13.2 Regression

We may estimate points using a simple linear regression model based on continuous label data. Each data point in a two-dimensional data collection is defined by two characteristics. As illustrated in Figure 2.5, the color of each point indicated each label. In basic linear regression, the label is considered as a third special dimension, and therefore we may map a plane with data points

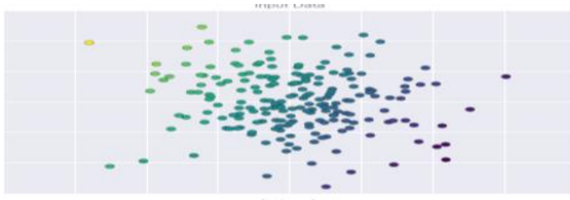


Figure 2.5: Simple data set for regression (adapted from [16])

2.13.3 Clustering

Clustering is an unsupervised learning method that attempts to build a model to allocate data to discrete groups without the use of previous knowledge data points.

Figure 2.6: Clustering (adapted from [16])

2.13.4 System Identification and Control

The capacity of multi-layer perceptron neural network approximation and dynamic system control make neural networks attractive options for modeling non-linear systems and corresponding non-linear controller design. The artificial neural network succeeds in system identification and controller design [23]. These are:

- Non-linear internal model control
- Fixed stabilizer controller
- Adaptive inverse model control
- Model predictive model
- Model reference control
- Neural adaptive feedback linearization
- Stable direct adaptive control

2.13.5 Optimization

Optimization is the process of determining the least value of a loss or cost function given a set of constraints, or the highest value of a profit or gain function given a set of constraints. So optimization is a well-established area in economics and mathematics, but ANNs, such as the Hopfield network, have been proven to be more effective at handling complex and nonlinear optimization issues.

2.14 Artificial Neural Network Optimization Techniques

2.14.1 Gradient Descent

The gradient decent technique of weighting [24] states that the update is determined by the previous weight, learning rate, and gradient of the loss function. The learning rate in gradient descent is constant, and convergence is dependent on the initial condition. Gradient descent optimization is prone to delayed convergence and stacking in the loss function's local minimum value. Including momentum in the weight update equation; the problem of stacking in local minimum can be alleviated.

$$W_{ji(t+1)} = W_{ji(t)} - n\, \partial J/\partial W_{ji(t)} + \alpha[W_{ji(t)} - W_{ji(t-1)}] \quad (2.18)$$

Where $w_{ji}(t)$ is the previous weight value, n is constant learning rate and $\partial J/\partial\ w_{ji}(t)$ is the gradient of the loss function, $\alpha[w_{ji}(t)-w_{ji}(t-1)]$ is the momentum term with α value in the between of [0.5, 0.9].

2.14.2 Quasi-Newton Method

The quasi-Newton method replaces the real inversion of the hessian matrix with an approximation of the inversion of the Hessian matrix. This method's weight update iterates as follows [25].

$W_{ji(n+1)} = W_{ji(n)+}ndn$ where d_n=Bngd (2.19)

Where B_n is the approximation of inverse Hessian matrix which updates from iteration to iteration and n is found by a line search. The logic of this method is that two consecutive iterations (x_n ,x_{n+1}) and gradient $\Delta\ f(xn), \Delta f(xn+1)$ contain Hessian information.

$$\Delta f(xn+1) - \Delta\ f(xn) \approx \quad H(xn)\ \ (xn+1-xn) \tag{2.20}$$

2.14.3 Conjugate Gradient Method

The Conjugate-gradient technique eliminates the need for Hessian matrix inversion by employing a direction vector that is a linear combination of previous direction vectors and the current negative gradient vector. The conjugate-gradient approach decreases oscillatory behavior in the minimal search while reinforcing weight modification based on past successful path choices. Only the quadratic cost function is convergent for the conjugate-gradient technique.

$W_{n+1} = W_n$ +nP(n) (2.21)

The direction vector at the n^{th} iteration of the conjugate-gradient technique algorithm is denoted by P(n). The network's weight vector is then updated by making a linear combination of the previous weight vector and the current direction vector.

2.15 Weight Initialization Technique

Weight initialization is critical in the training of artificial neural networks to attain a global optimum point in a short iteration cycle. Various scholars suggested various weight initialization strategies for various data kinds and activation functions. Initialization approaches that are appropriate for one type of data and activation function may not be appropriate for another. Table

2.2 summarizes the weight initialization techniques described in works of [26] where f_{inputs} and $f_{outputs}$ are the numbers of input and output weight connections respectively.

Table 2.2: Weight initialization method (adapted from [26]).

Standard Normal Distribution mean : 0 standard deviation : 1		
Xavier Initialization		
Activation function	Uniform distribution [-r, r]	Normal distribution
sigmoid	$r = \sqrt{\frac{6}{n_{inputs} + n_{outputs}}}$	$\sigma = \sqrt{\frac{6}{n_{inputs} + n_{outputs}}}$
Hyperbolic tangent	$r = \sqrt[4]{\frac{6}{n_{inputs} + n_{outputs}}}$	$\sigma = \sqrt[4]{\frac{6}{n_{inputs} + n_{outputs}}}$
He Initialization		
ReLU	$r = \sqrt{2}\sqrt{\frac{6}{n_{inputs} + n_{outputs}}}$	$\sigma = \sqrt{2}\sqrt{\frac{6}{n_{inputs} + n_{outputs}}}$

2.16 Wastewater Treatment Process Control Strategy Review

Researchers like [3] describe wastewater treatment systems as non-linear and dynamic systems that necessitate reliable and robust control strategies to achieve the desired objectives. In addition to system complexity, seasonal fluctuations are frequent and dynamic. Based on the treatment objectives, the performance of wastewater treatment plants is defined by several criteria such as effluent quality, economic cost, and greenhouse gas emissions. Wastewater treatment performances are mainly affected by large disturbances and uncertainties related to the influent wastewater's composition. Over the years, different pieces of literature have described a wide range of different control strategies for wastewater treatment plants. Among the most commonly used control strategies are dissolved oxygen cascade control, ammonium-based supervisory control (feedback, and feedforward–feedback control), advanced single input single output and multiple inputs multiple output control, and aerobic volume control, which had been extensively researched.

The desired dissolved oxygen level in the biological wastewater treatment plant, according to the researcher [4], can be managed using a self-organized radial base function neural network model predictive control to satisfy effluent water quality while using the least amount of energy. The author employs a self-organized radial base function to forecast future responses across defined time horizons and then applies these projected values to an empirical optimization process that minimizes the cost function in the region of the optimal control signal at each sampling interval. One of the challenges revealed in this research is that the suggested control approach works only for limited prediction and control horizons, and above the predictive horizons, integrated absolute errors dramatically rise. The second concern is the control approach does not account for nitrate accumulation in the biological treatment tank, which harms the performance of biological wastewater treatment plants. Other researchers such as [27], use a neural network as an internal model of the process and change the dilution rate to achieve the control objective. This control strategy provides several options for adjusting the control law using the following parameters: the prediction horizon, the control horizon, the error weights, and the command. The controller adjusts the dilution rate and forces the dissolved oxygen concentration to stay within the imposed set-point. This has a positive effect on the substrate concentration, which is kept within the legal limits.

Researchers such as [28] investigated the proportional-integral, fuzzy-proportional integral, model predictive control, and fuzzy logic control application for the wastewater treatment plant. The plant's performance is measured using the operational cost index and the effluent quality index. Other scholars including [29] similarly investigate the model predictive control, PID, and data-driven approach control for wastewater treatment plants application and made comparisons among them. Based on predictions of future wastewater system behavior, model predictive control performs superior control in optimizing nitrogen removal. The multivariable configuration significantly improves the performance of PID control in dissolved oxygen and nitrate control. When compared to the default PI simulation proposed by BSM, the data-driven approach produced comparable results. Among the three control structures under consideration, none can claim to be the best all-around control strategies for wastewater systems, as each has some limitations, as well as significant advantages in terms of different application and performance parameters evaluation.

Researchers [5] on the other hand proposed PID hierarchical control for nitrogen removal from wastewater, with the ratio of nitrogen removed from the slugging process to energy required as a control variable and dissolved oxygen as a manipulated variable. This form of control necessitates the use of a plant model for parameter tuning, which makes it difficult to describe deterministically the nonlinear features of the bioreactor. Due to the variation and high uncertainty of the parameters, tuning in a wastewater treatment plant is a difficult task as noted from the work of [30]. The multivariable wastewater treatment plant employs an auto-tuning method for decentralized PI controllers based on an adaptive interaction algorithm. The decentralized PI with an approximate Frechet tuning algorithm was found to be an appealing tuning task for multivariable wastewater treatment plants, with improvements in energy consumption and effluent standards. However, the adaptive model is proved to yield optimal controller parameters where the controller gains are not fixed.

A data driven adaptive optimal control was proposed to optimize the operating points of the dissolved oxygen concentration and the nitrate level in a wastewater treatment plant, and its performance is compared to the performance of a PID controller with fixed operating points under the full range of operating conditions. The simulation result shows that a data driven adaptive optimal controller can significantly reduce energy consumption [29]. The optimal control approach is motivated by the need to compromise between treatment efficiency and operational costs, according to [31]. Two distinct control problems are addressed, each with a different scope and method, but both within the context of optimality. Based on the simulation result, the effectiveness of a control scheme is almost entirely determined by the cost function choice. Dissolved oxygen regulation and sludge recycling policy are two separate control issues that have been considered. Given the inherently decoupled nature of the two problems, two separate solutions based on differing approaches have been developed, but all within the context of performance index optimization. In this sense, the resulting control laws represent optimal trade-offs between control precision and energy expenditure.

The model predictive control strategy was also assessed by some scholars to control the dissolved oxygen in the aerated reactors and the nitrate level in the anoxic compartments as stated in the works of [32]. The airflow rate in the aerated reactors and the internal recycle flow are the variables that are manipulated. The simulation results show that the model predictive control strategy outperforms traditional PI control in terms of overshoot and time response. A combined feedback feedforward model predictive control strategy scheme for disturbance rejection was also proposed and demonstrated to be effective. Despite the above significant contribution, the success of the model predictive control strategy is highly dependent on a reliable process model, that is, a model that accurately approximates the process dynamics, which is a difficult job for highly nonlinear and dynamic systems like wastewater treatment process.

Other researchers [33] come up with a self-organizing sliding-mode controller for wastewater treatment processes to achieve acceptable and stable control performance. The proposed controller is novel in terms of a practical control strategy, combining the benefits of sliding-mode control and fuzzy-neural networks to suppress disturbances and uncertainties, and the self-organizing mechanism, based on tracking error and structure complexity was alleviated the chattering phenomenon of the self-organizing sliding-mode controller and improve performance.

For the present work, the promising characteristics of artificial neural networks, such as learning from experience, generalization based on training, real-time implementation, nonlinear system modeling, and the internal model control easily implementable with ANN leads to develop biological wastewater treatment plant and its corresponding internal model controller using artificial neural networks. As a result, this model development method enables us to overcome the previous research limitation such as tuning difficulties, minimizing computation time, complex mathematical model requirements, and dependency on sensor performance. The simulation result also demonstrated that the performance of an ANN internal model controller outperforms that of a traditional PI controller in both transient and steady-state conditions as well as it has better disturbance rejection capability compared with previous studies. Even though the artificial neural internal model controller has a slightly higher overshoot than the conventional PI controller, the other performance indicators improved significantly.

CHAPTER THREE: MATERIALS AND METHODS

3.1 Material

3.1.1 Data Collection

The required data for this research work was gathered from the Kality wastewater treatment plant, which is located in Akaki Kality Sub-City, Addis Ababa, Ethiopia, from 2016 to 2018 for three consecutive years to capture seasonal variation and time series trend. Operational variables such as dissolved oxygen, ammonia, nitrite, nitrate, oxygen transfer coefficient, biochemical oxygen demand, biological oxygen demand, temperature, ph, and dissolved suspended solids were collected from treatment plant laboratory tests. These raw data should be subjected to data preparation operations such as filling missing values, data transformation, and outlier detection performed before model development. Following data preparation, factor analysis was deployed to identify input factors that have a substantial effect on the output variable for both plant and controller neural network models. Finally, six operational variables were selected as model variables. Then the processed data was split into three subsets: training, testing, and validation sets based on a random division approach until all subsets exhibit comparable statistical features.

Table 3.1: Model variables

No	Variables	Designated Nomenclature	Unit	No	Variables	Designated Nomenclature	Unit
1	Output dissolved oxygen	DO	mg/l	4	Input dissolved oxygen	Do_i	mg/l
2	Ammonia	NH_3	mg/l	5	Oxygen transfer coefficient	OTC	m3/day
3	Nitrite	NO_3	mg/l	6	Nitrate	NO_2	mg/l

3.1.2 Software

The MATLAB neural network toolbox and MATLAB script file were used to create the neural network script code, configure the network, select the training algorithm, initialize the weights and biases, train and validate the network, and evaluate the model prediction performance. MATLAB was also used for numerical statistical analysis and graphical depiction of trained network model prediction performance. Outlier identification was performed using the SPSS software platform, which provides sophisticated statistical analysis and a vast library of machine learning algorithms. Mendeley and Microsoft Office in addition to the aforementioned software tool deployed for citation; and data gathering, organization, and storage, and document preparation, respectively.

3.2 Methods

Model, according to [33] classification, may be divided into two types: black-box model and white-box model. A black-box model is the most basic type of mathematical model: it is based solely on observation, may have some predictive power, but does not explain. A white box or knowledge base model, on the other hand, is the outcome of any investigation of a physical, chemical, biological, or any other phenomenon that generates the quantity to be represented. These events are represented by equations that are based on theoretical information that was accessible at the time the model was created. As a result, the white-box paradigm necessitates theory, whereas the black-box approach necessitates measurement. The majority of neural network applications available now are black-box models. The creation of an artificial neural network model entails several phases, beginning with input selection and ending with model calibration.

3.3 ANN Model Development Conceptual Framework

The first stage in the artificial neural network model creation process, according to [34], is the selection of model output (i.e. the variable to be predicted) and model input variables from the potential variable list. The unprocessed input raw data, as well as the model output data, should be passed under data transformation, missing value filling, outlier identification before applying for model building. Then the whole datasets are split into three data subsets (i.e. training, testing, and validation). Following this, the model architecture type was chosen. The ANN model was built after the raw data has been preprocessed and the model architecture was determined. Through constant trial-and-error, the developed model was optimized by selecting a suitable number of layers, nodes, and activation functions. Finally, the model was validated using validation datasets. If the validation yields a better match with the observed datasets, the model was ready for implementation. In general, ANN model development processes necessitate sequential activities such as input variables selection, data preprocessing, data division, model architecture selection, model structure determination, optimization, and model performance assessment and model evaluation. Figure 3.1 shows sequential activities which are illustrated in detail in the subsequent section.

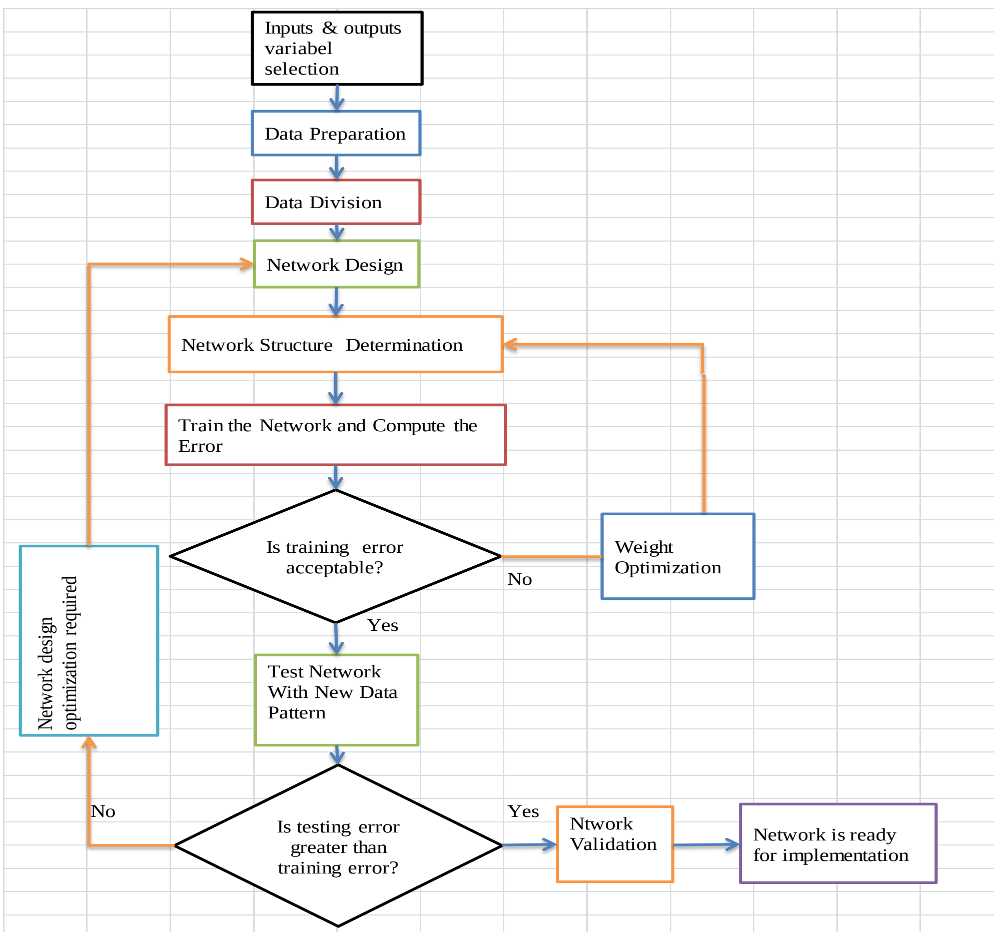


Figure 3.1: ANN model development process conceptual framework (adapted from [19])

3.3.1 Input Selection

Typically, the construction of an artificial neural network model begins with the selection of acceptable inputs. Input selection should be in an ideal condition, free of both too few and too many model inputs, which are both undesirable. Omission of one or more essential inputs results in an ill-trained model from an input-output connection that cannot achieve the network's desired generalization capabilities. Several variables in any plant or process often contained linked information; hence, including too many variables with related information may result in input redundancy. Many issues might arise when using an artificial neural network with input redundancy. The first significant issue is the increased possibility of overfitting as a result of a high number of inputs resulting from a big size of network with associated model parameters that must be trained. The second issue caused by duplicated input is that network training may become stuck in local minima, impeding the process of identifying the ideal weight points.

[34] Provides a list of several approaches that are required to detect the link between inputs and outputs and therefore help us to pick the correct inputs variables for the artificial neural network model. In general, input-output identification strategies may be divided into two types: model-based approaches and model free approaches. The model based method is based on the creation of multiple artificial neural networks with various inputs to select the best inputs from a pool of potential inputs. The model-free method, on the other hand, does not require trained artificial neural network models to identify candidate inputs. Ad-hoc is a model free method in which model developers pick inputs on arbitrary bases depending on domain expertise. The second model free input-output selection approach investigates the strength of the link between input outputs using statistical metrics. In general, the choice of analytical approaches is determined by the features of the process's input-output connection. If the model plant's input-output characteristics are linear, the correlation may provide enough information to pick an appropriate input-output pair. If, on the other hand, the neural network model non-linear system has a high likelihood of non-linear input-output coupling statically dependent measure such mutual information approaches offer adequate information for correct candidate input selection. Since the model based input selection method is more time consuming than model free techniques, and because the wastewater treatment process is highly non-linear and dynamic, factor analysis is chosen and used for present work to identify relevant input variables for both artificial neural

network models. Operational variables such as input and output dissolved oxygen, ammonia, nitrite, nitrate, oxygen transfer coefficient were selected for model input variables.

3.3.2 Data Preprocessing

Because all algorithms are not capable of resolving difficulties with missing data, additional characteristics, or de-normalized values, the goal of data pre-processing jobs is to prepare the data as best as possible for learning algorithms[35]. Following a thorough examination of the raw data, the following data preparation activities were completed in this study.

Filling missing values: Except for a few algorithms, the majority of training algorithms do not function well with missing values. Some numbers may be incorrect owing to a variety of factors, including random error, systematic error, and sensor noise. The following methods can be used to handle the missing value: get rid of the instance if there is sufficient data and just a few non-relevant instances have some missing values, it is safe to delete these instances. Remove the following attribute: when the majority of the values are absent, the values are consistent, or the property is closely associated with another attribute, eliminating it makes sense. N/A: assign a special value sometimes a value is absent for legitimate reasons, such as the value being out of scope or the discrete attribute value not being specified, and forecast the value based on other characteristics.

Outlier removal: Some data values may be substantially different from the bulk of data values and impact the entire training process in many ways. These outliers are detected and removed from training datasets.

Data transformation strategies reduce the datasets to a format that a learning algorithm expects as its input, and may even help the system learning speed and performance. Making the data in the range of normalization scales generally between 0 and 1 is one of the data transformation tasks.

$$x = \frac{xi}{xmax} \tag{3.1}$$

3.3.3 Data Division

The second job in the artificial neural network model building procedure was to divide the given data sets into three groups. The initial data set is the training data set, which is used to estimate the unknown connection parameters between various nodes by backpropagating the error term from the output layer to each hidden layer and all the way to the input layer. The second data subset is the testing set, the primary goal of which is to identify when to terminate training before the network overfits. The last data subset is the validation data set, which is used to assess the generalization capabilities of a trained network for unexpected system characteristics. All three sub-datasets should be representative of all data patterns in the available data sets for the artificial neural networks to be properly trained with appropriate network architecture. To create the finest artificial neural network models, training, testing, and validation data sets should all have the same statistical characteristics.

Artificial neural network data division methods are categorized under two major ways in the work of [34]. The unsupervised method, the first method, does not explicitly examine the statistical characteristics of data sets. The majority of this method is unable to determine if the best feasible model has been built or whether model performance based on the validation set is reflective of model performance. In contrast to the first technique, the supervised approach, the second technique, considers the statistical characteristics of the data set while splitting the entire data set into various subsets. The trial-and-error technique is the most often utilized method from the supervised approach by many researchers. This work also chose the trial-and-error approach for data division.

3.3.4 Model Architecture Selection

The primary goal of neural network design selection is to define network type and information flow in the network. Traditionally, neural networks are divided into feedforward and backward (recurrent) neural networks. In a feedforward neural network, information flows solely forward from the input neuron to the hidden neuron and then to the output neuron, but in a recurrent neural network, information flows both forward and backward possible. There are various types of feedforward neural networks available in the literature, but the most common forward neural networks include Adeline and Madeline, Perceptron, Backpropagation networks, Radial base functions, General regression networks, Learning vector quantization, Probabilistic neural networks, and Fuzzy neural networks. On the other hand, recurrent network models include well-known neural networks such as the Hopfield network, Boltzmann machine, Kohonen's self-organizing feature map, Recirculation network, Adaptive resonance theory, and Bi-directional associative network types, as stated in the works of [19]. In addition to the above-mentioned conventional categorization, other researchers, such as [34] include hybrid forms of neural model designs for systems with very complicated, non-linear, and dynamic processes, as well as numerous variables involved.

A previous study on wastewater treatment plant modeling and control utilizing multi-layer neural network architecture and results demonstrated that the plant and its controller were adequately represented by ANN models. As a result, this study employs a multi-layer perceptron design from the feedforward network architecture. The network has a high degree of connection, the amount of which is controlled by the network's synaptic weights. The general multi-layer perceptron design and information flow from the input layer to the output layer are depicted in Figure 3.2.

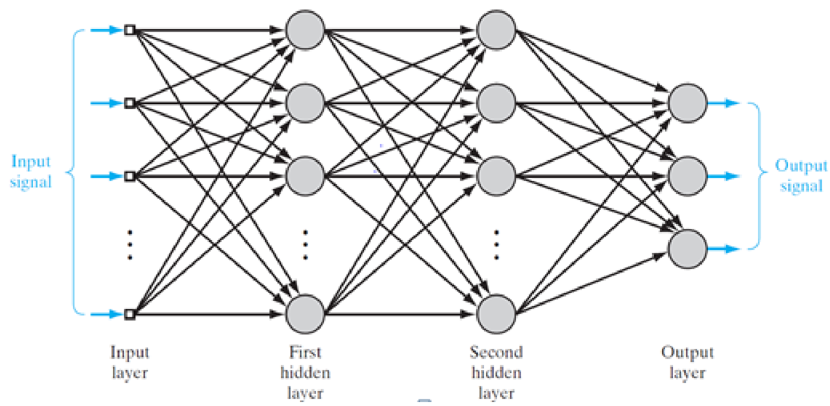


Figure 3.2: Two hidden layer multi-layer perceptron

In figure 3.2, input signals travel through the network in the forward direction (from the input node to the output node) and emerge as an output signal at the network's output node. The second signal, which originates at the output layer and travels backward (layer by layer) through the network, is referred to as an error signal. The backpropagation training technique updates network parameters based on the error signal. The input layer is in charge of receiving data from the outside world. These inputs are typically normalized within the bounds of activation function limit values. Hidden layers are made up of neurons that are in charge of extracting patterns related to the system being studied. A multilayer perceptron's hidden neurons are programmed to conduct two computations: The first job is to compute the function signal that appears at each neuron's output, which is represented as a continuous nonlinear function of the input signal and the synaptic weights associated with that neuron. The second objective is to compute an estimate of the gradient vector, which is required for the network's backward pass. The output layer is in charge of creating and displaying the final network outputs that come from the processing conducted by the neurons in the preceding levels.

3.3.5 Model Structure Selection

Different challenges are involved in artificial neural network structure selection, such as determining the best number of hidden layers and associated nodes, selecting information processing units, and balancing model complexity and model generalization ability. Different model structure selection approaches are described in depth in the works of [34]. These approaches are divided into three categories: global, ad hoc, and stepwise. The stepwise model structure selection method was chosen in this study because it provides model developers has two options: start with the simplest artificial neural network structure and progress to a complex network step by step, or start with a complex network structure and minimize the network step by step to reach an optimal network structure.

A pruning/removing technique begins with a sophisticated artificial neural network topology and is adequate to capture system dynamics and inherent system properties. Then, one by one, network parameters and related nodes depending on magnitude are eliminated until network performance dramatically deteriorates. Constructive algorithms, on the other hand, start with a simple artificial neural network structure and add a hidden layer and associated nodes one at a time, training a network until model performance improvement ceases. In this study, both techniques were used. Aside from determining the optimum hidden layer and related nodes, the model structure selection procedure necessitated the selection of an appropriate activation function for signal processing and captures system non-linearity at each node. Since there are no negative input variables, the sigmoid activation function is used in this study.

3.3.6 Model Training and Optimization

Local optimization techniques such as backpropagation, conjugate gradient, and newton methods have piqued the interest of many researchers. The main problem with these local optimization approaches is that when the error surface is uneven, there is a risk of getting stuck in local minimum spots. However, even if there is a local minimum trapping problem, they are computationally efficient. The remaining deterministic approach, on the other hand, has modest proportions that are subject to global optimization approaches such as genetic algorithms. Unlike the deterministic method, the statistical approach seeks to get the distribution of model parameters rather than a single parameter vector. This method is typically used when the model developer wants to account for parameter uncertainty throughout the training process.

As stated in [36] the backpropagation algorithm (BPA) is a generalized delta rule for nonlinear activation functions. BPA employs optimization techniques based on gradient descent. J may define the loss function as follows:

$$J = \sum_{j=1}^{m}(d_i - y_i)2 \tag{3.2}$$

Where di is the intended value and yi is the model's projected value.

$$\frac{\partial J}{\partial w_{ji}} = \frac{1}{m}\sum_{j=1}^{m}\frac{\partial}{\partial w_{ji}}(d_i - y_i)2 \tag{3.3}$$

$$\frac{\partial J}{\partial w_{ji}} = \frac{1}{m}\sum_{j=1}^{m}\frac{\partial}{\partial y_j}\left(d_i - y_j\right)2\ \frac{\partial y_j}{\partial w_{ji}} \tag{3.4}$$

$$\frac{\partial J}{\partial w_{ji}} = \frac{-2}{m}\sum_{j=1}^{m}(d_i - y_i)\ \frac{\partial y_j}{\partial w_{ji}} \tag{3.5}$$

If the activation function is sigmoid, neuron output y_j is denoted by f. (s) and the total of all characteristics is multiplied by the weight factors, the resultant equation becomes.

$$\frac{\partial f}{\partial s} = f(s) - f(s)2 \quad = f(s)[1 - f(s)] \tag{3.6}$$

$$\frac{\partial y_j}{\partial s_j} = y_j[1 - y_j] \tag{3.7}$$

$$S_j = \sum_{i=0}^{n} w_{ij}\, x_i \tag{3.8}$$

$$\frac{\partial y_j}{\partial w_{ji}} = \frac{\partial y_j}{\partial s_j} \quad \frac{\partial s_j}{\partial w_{ji}} \tag{3.9}$$

$$\frac{\partial s_j}{\partial w_{ji}} = \sum_{i=0}^{n}\frac{\partial w_{ji}}{\partial w_{ji}}\, x_i \quad = \quad x_i \tag{3.10}$$

$$\frac{\partial y_j}{\partial w_{ji}} = \frac{\partial y_j}{\partial s_j} \quad \frac{\partial s_j}{\partial w_{ji}} = \ y_j\left[1 - y_j\right] x_i \tag{3.11}$$

$$\frac{\partial J}{\partial w_{ij}} = -\frac{2}{m}\sum_{j=1}^{m}\left(d_j - y_j\right)y_j\left(1 - y_j\right)x_i \tag{3.12}$$

$$\frac{\partial J}{\partial w_{ij}} = -\frac{2}{m}\sum_{j=1}^{m}\delta_j x_i \tag{3.13}$$

Where $\delta_j = \left(d_j - y_j\right)y_j(1 - y_j)$

The weight increase is defined by in the gradient descent as follows:

$$\Delta w_{ji} = -\mu\frac{\partial J}{\partial w_{ji}} \tag{3.14}$$

Where, $\Delta w_{ji} = \frac{2\mu}{m}\sum_{j=1}^{m}\delta_j x_i$

$$\Delta w_{ji} = n\sum_{j=1}^{m}\delta_j x_i \tag{3.15}$$

Where $n = \frac{2\mu}{m}$ is learning rate

Finally, the BPA new weight update is defined as follows, depending on the previous weight and gradient of the loss function:

$$w_{ji(t+1)} = w_{ji(t)} - n\frac{\partial J}{\partial w_{ji(t)}} \tag{3.16}$$

Backpropagation weight vector changes are proportional to the error's gradient; the relative changes that must occur in different weights when a training sample (or set of samples) is presented but does not specify the exact magnitudes of the desired weight changes. The amount of the change is determined by the right selection of the learning rate n. A large number of n indicates quick learning, but the weight may swing, whereas low values indicate sluggish learning[20].

Backpropagation leads the weights in a neural network to a local minimum of the mean square error, which may differ significantly from the global minimum that corresponds to the best weight choice. This is especially noticeable when the error surface is extremely uneven and has a large number of local minima. The weight changes in the i^{th} iteration of the backpropagation algorithm are determined by the weight changes in the (i-1) iteration that came before it. As a result, the inclusion of past weights in the weight update method has a similar impact as an average gradient in overcoming the backpropagation difficulties of trapping at a local minimum point before reaching a global optimum point. Backpropagation algorithm with momentum term defined as follows [22].

$$w_{ji(t+1)} = w_{ji(t)} - n\frac{\partial J}{\partial w_{ji(t)}} + \alpha[w_{ji(t)} - w_{ji(t-1)}] \tag{3.17}$$

The optimal value of α minimizes the number of iterations required for convergence. A score near 0 indicates that the past has little influence on the weight change, whereas a value close to 1 indicates that the present error has little influence on the weight change. The value is often in the range of [0.5, 0.9].

In a multilayer perceptron network, there are three typical forms of weight updates under the generalized delta rule. The first is pre-pattern or stochastic gradient descent techniques, in which the learning weights vary after each sample presentation. Because the weights must be adjusted after each sample presentation, this form of training is more costly than per-epoch or Bach method training. Another issue with per-pattern learning is that the network may only learn to create an output that is near to the intended output for the current pattern, without learning anything about the whole training set. The second method is batch-mode learning, in which weights are adjusted only after all inputs and goals have been given to the network; this method may be used with both static and dynamic networks. Parallelism in batch-mode training can be used to minimize training time, as opposed to per-pattern training, which cannot be parallelized in this way. The third type of weight update method, mini-batch, is intermediate between instantaneous and batch updates. The entire data set is split into batches and sent into the network one at a time. The output error is computed and accumulated for each sample input in each batch until the last input's elements are provided into the network. Then, at the end of each batch, the weight is updated. This sort of weight update technique has about the same training speed as the stochastic method and the same stability property as the batch method. This method of weight updates was used in this study.

3.3.7 Model Evaluation

As a prerequisite for developing an artificial neural network model, it is necessary to first pick the appropriate model architecture, structure, data division, and training method. Following completion of the aforementioned required tasks, the ANN model developer always develops performance evaluation criteria to fulfill the model's stated objectives. The literature provides a variety of performance evaluation measures dependent on the model goals. ANN model performance evaluation criteria that are suitable for one or more objectives may not be suitable for others. As a result, the researcher must select appropriate techniques based on stated objectives.

The mean squared error was the most often used error function in neural network model optimization. It is a network performance function that evaluates network performance by attempting to minimize the sum of mean squared errors. The MSE is an example of supervised training, as is the perceptron learning rule. There is a source input and a target output in the network. As each input is applied to the network, the network output is compared to the target. The error is determined by the difference between the target and network outputs. The network's performance is calculated by averaging the total number of errors. The goal of network training was to lower the MSE to the smallest possible value. The MSE was chosen for the present study as a performance assessment due to its applicability to ANN network optimization, and to provide a consistent evaluation platform with BSM.

CHAPTER FOUR: RESULTS AND DISCUSSION

4.1 Introduction

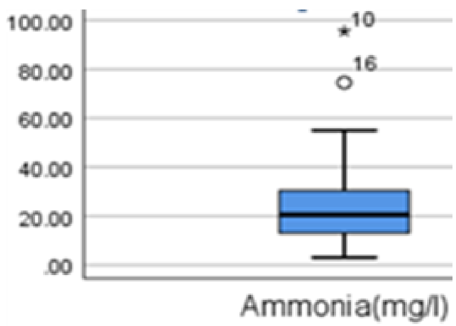
For modeling, simulation, and analysis of both the plant and controller models, MATLAB software was utilized. The overall controller design aims are to improve set-point tracking, reduce settling time, overshoot, and rising time, and steady-state error. The primary control aim for this research endeavor is to keep the dissolved oxygen concentration in the aerobic treatment at 2 $g.m^{-3}$ and therefore improve the wastewater treatment process's performance [16]. The performance of the suggested controller is improved by performing a series of training until the best ANN configuration is found. Before using the data for artificial neural network model creation, input data preparation such as outlier identification, missing data filling, normalization, and input data statistics analysis were performed in this chapter. In addition to these activities, this chapter contains the training processes, final selected model parameters, simulation results, and subsequently performance comparison of PI and ANN internal model controller.

4.2 Input Data Pre-Processing

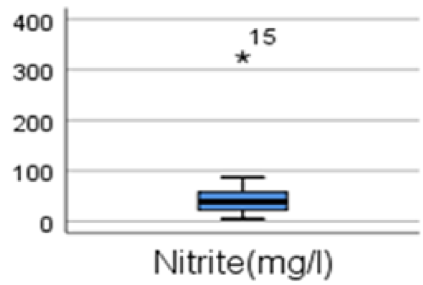
The aim of data pre-processing was to create a suitable neural network's training dataset, which reflects the relationship between network inputs and outputs and makes it much more suited for the network. Outlier detection, data transformation, graphical analysis, and basic statistical description were the primary tasks completed in the data preparation phase.

4.2.1 Input and Output Data Outliers Test

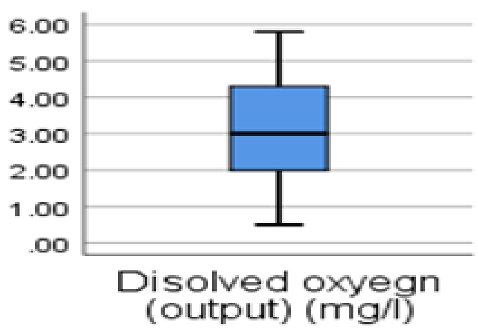
Extremes and outliers are identified and removed from training and validation datasets in this study. SPSS is a software package that uses a box plot to identify extremes and outliers in data sets. Tukey hinges define the box plot's boundaries. A line inside the box denotes the median. The length of a box is an interquartile range calculated using Tukey's method. As shown in figures 4.1 of A and B, input data sets such as ammonia and nitrite contain extreme data values at cases 10 and 15 respectively. All of these values are greater than three times the interquartile range from the box's end. Similarly, figures A and E demonstrated that outliers in the ammonia and nitrate datasets, respectively, at cases 16 and 17. All identified extremes and outliers are eliminated from the training and validation dataset during the plant and controller model development.



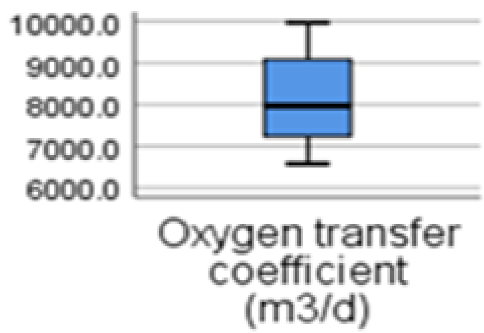
A



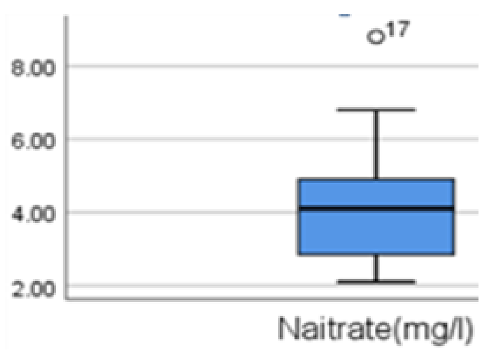
B



C



D

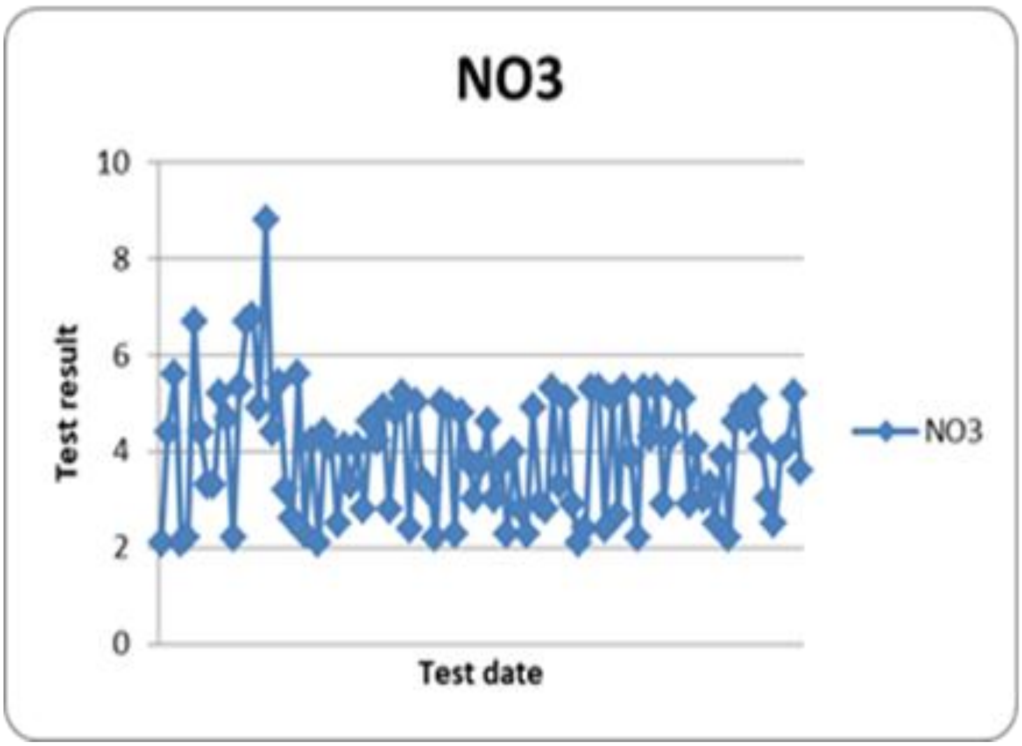


E

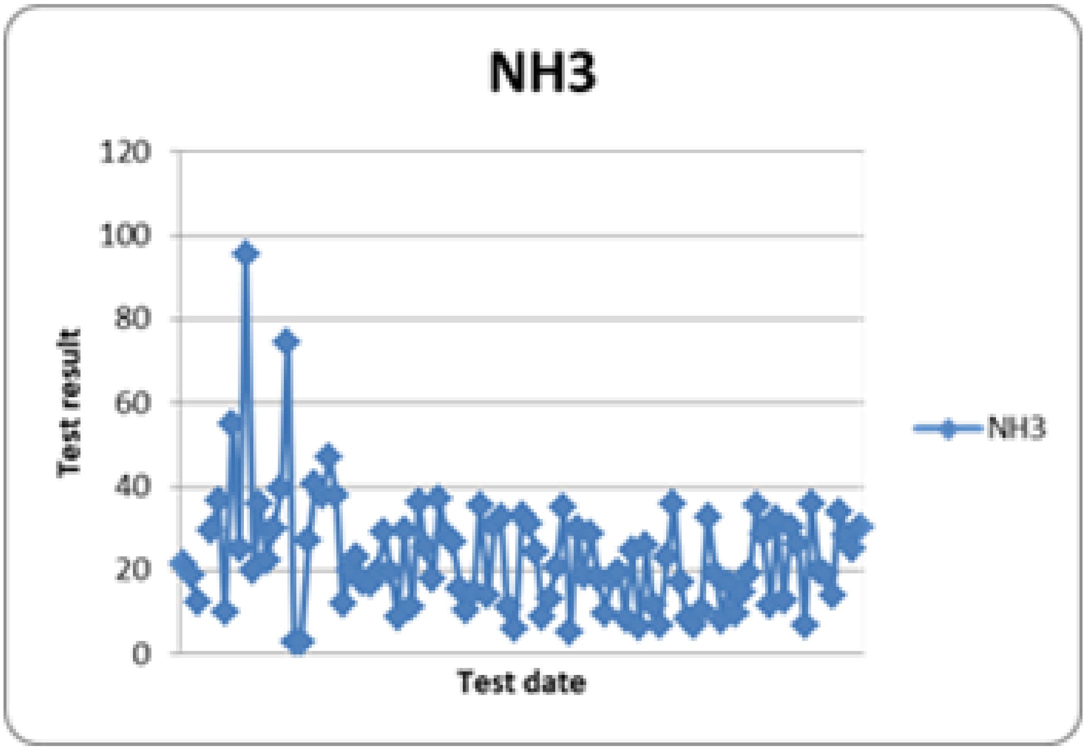
Figure 4.1: Input-output data outliers test result

4.2.2 Input Data Graphical Representation

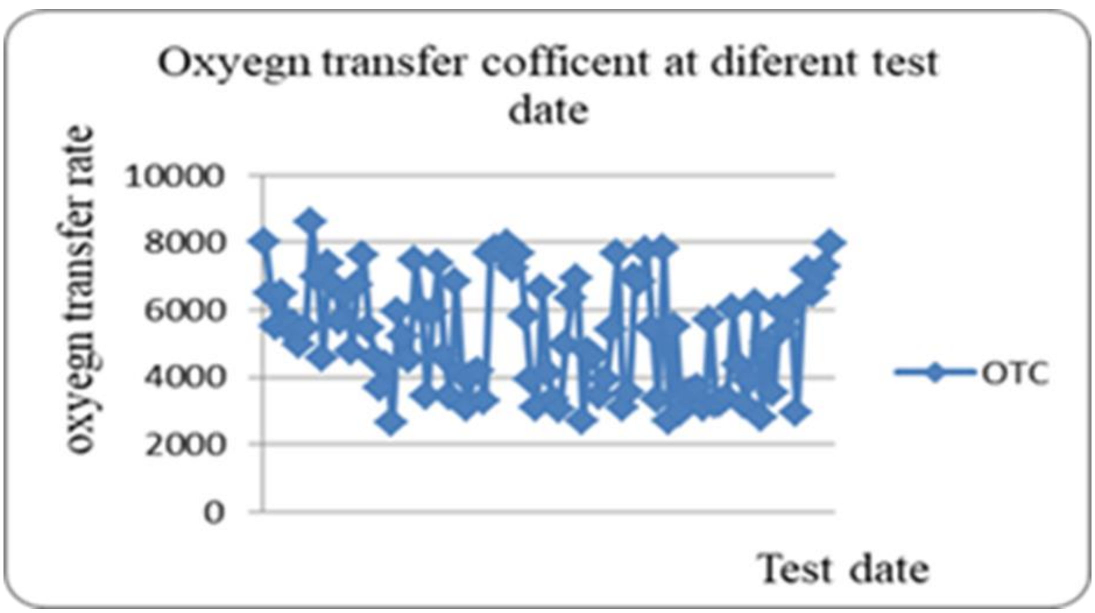
The raw operational laboratory test data collected from the Kality wastewater treatment plant for the six operational and wastewater quality parameters such as nitrate (NO_2), nitrite (NO_3), ammonium (NH_3), input dissolved oxygen (Doi), and output dissolved oxygen (DO)), and oxygen transfer coefficient (OTC) are directly plotted using Excel software. Figure 4.2 shows that training data are highly variable across the different tests. High data variability over the different seasons, in addition to sampling test variation, is prominently observed from dataset graphic representation. The dynamic and nonlinear nature of wastewater treatment plants makes conventional PI controllers less efficient and difficult to control. With the inherent nonlinear input-output relationship capturing capability and generalization based on sample training, an artificial neural network internal model controller can easily control these high variability input-output dynamics.



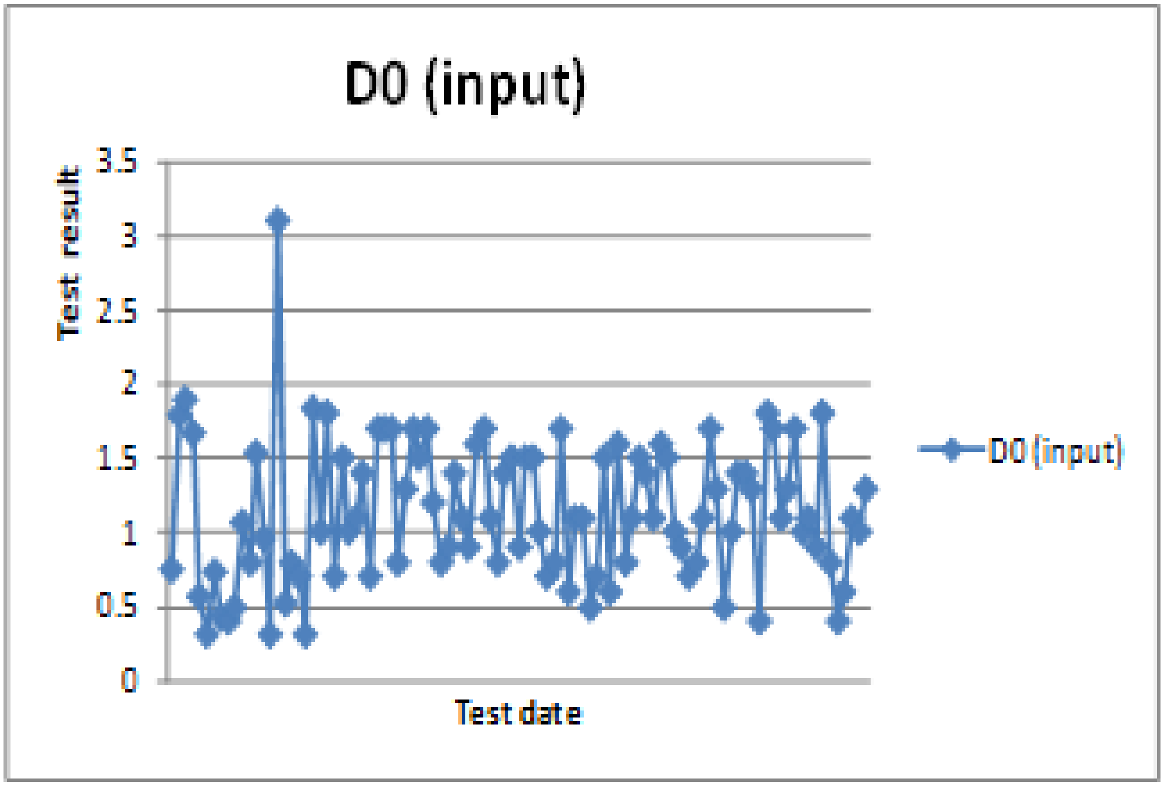
A


NH3
120
100
80
60
40
20
0
Test result
Test date
NH3

B


Oxyegn transfer cofficent at diferent test date
10000
8000
6000
4000
2000
0
oxyegn transfer rate
Test date
OTC

C



D

Figure 4.2: Input and output variables seasonal characteristics

4.2.3 Input and Output Data Preliminary Statistics

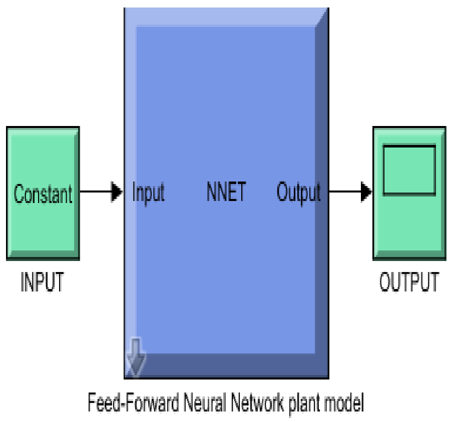
Table 4.1 shows the basic statistics descriptors for the input and target variables in raw operational data. The oxygen transfer coefficient, as shown in Table 4.1, is highly dispersed or spread out when compared to other input variables. Whereas, the input dissolved oxygen dataset, on the other hand, has less variability. Since all of the skewness test results are positive, the distribution is positively skewed, but the degree of skewness is not statistically significant. Similarly, dissolved oxygen (input and output), ammonia, nitrate, and oxygen transfer coefficient kurtosis test result value less than three, thus classified as platykurtic, whereas ammonia and nitrite kurtosis test result value greater than three and classified as leptokurtic.

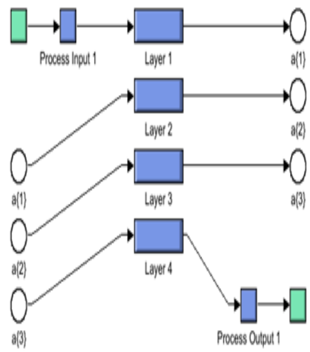
Table 4.1: Basic input and output statistics

		Dissolved oxygen(input)(mg/l)	Ammonia(mg/l)	Nitrate(mg/l)	Nitrite(mg/l)	Oxygen transfer coefficient(m3/d)	Dissolved oxygen(output)(mg/l)
N	Valid	99	98	99	99	99	99
	Missing	0	1	0	0	0	0
Mean		1.1377	23.3041	3.9 459	43.03	8142.462	3.1140
Median		1.1000	20.5000	4.1000	39.00	7969.400	3.0000
Mode		1.10	3.00[a]	4.10[a]	56	6570.9[a]	2.70
Std. Deviation		.47788	14.0786	1.29511	35.347	998.9955	1.44282
Variance		.228	198.208	1.677	1249.438	997992.00	2.082
Skewness		.606	1.911	.588	5.209	.138	.093
Kurtosis		1.367	7.324	.687	41.087	-1.350	-.991

4.3 Artificial Neural Network Plant Model

The biological reactor of a wastewater treatment plant's forward ANN model was built with a multi-layer perceptron that was trained with ammonia, nitrate, nitrite, input dissolved oxygen, oxygen transfer coefficient, and one lag output dissolved oxygen as input variables while output dissolved oxygen as output variable using the Levenberg Marquard learning algorithm. Levenberg Marquardt algorithm has the fastest convergence when the training data is small and medium in size like this study. Using a random division function, the input and output datasets were divided into 80 and 20 percent for training and validation datasets, respectively. Both sets will be normalized by dividing each value by the maximum value of the dataset. The training dataset is used to estimate unknown connection parameters between various nodes by backpropagating the error term from the output layer to each hidden layer and the input layer, whereas the validation dataset evaluates a trained network's generalization capabilities for unexpected system properties. Several ANN architectures were tested, and the best results were obtained with three layers of 43 neurons each, a sigmoid activation function on the hidden layers, and a linear activation function in the output node. When the validation errors increased and the maximum failed to reach six, the training was terminated. Finally, the weights and biases with the lowest validation error are returned. At an epoch of 11, the best validation performance achieves a mean square error of 0.057484. The ANN forward plant model, the number of layers, and the number of neurons deployed in the model are depicted in figures 4.3, 4.4, and 4.5, respectively.





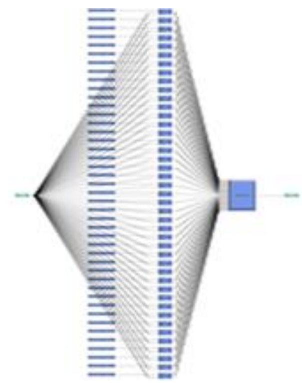


Figure 4. 3 ANN Plant Model   Figure 4. 4 Layer structure   Figure 4. 5 Number of neurons

Figure 4.6 shows that the mean percentage prediction deviation is -0.63 percent (under-predicted). As a result, it is possible to conclude that the ANN plant model is quite efficient and accurately predicts dissolved oxygen.

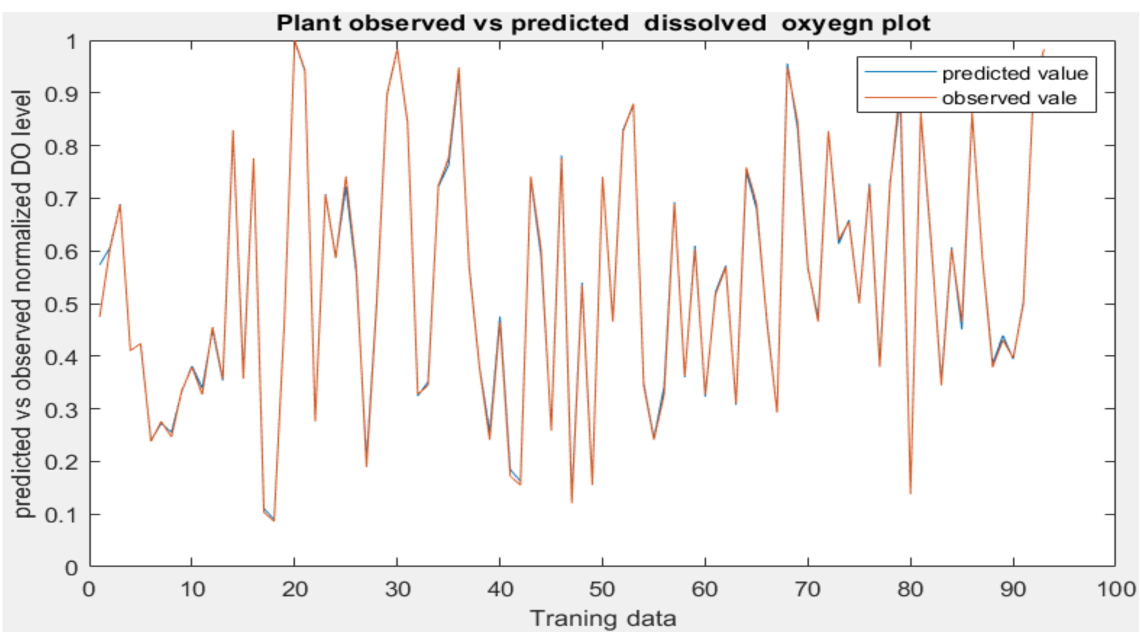


Figure 4.6: ANN plant model prediction vs actual data trend

Artificial neural network training generates various plots to demonstrate various aspects of training, such as the amount of dependent variable variation explained by explanatory variables (regression plot), training performance (performance plot), error dispersion (error histogram), and training state. The ANN regression plots describe the situation of neural network behaviors with training and validation datasets. The regression plots in Figure 4.7 show the relationship between the output and the target values. If the training is excellent, the network output and target will be the same. For example, in this model, the total model output approximates 0.95 * target value + 0.028 with a coefficient of determination of 0.99896, indicating an excellent data match in the input-output ANN plant model. The performance plot, similar to the regression plot, shows the value of the performance function versus the number of iterations. The performance plot of the training and validation states is shown in Figure 4.8. This graph shows that until the best training point, there is no discernible difference between the train and validation curves. This implies that there are no major problems with the network training. The MSE initially decreases as the number of training epochs increases. The validation performance reached the minimum value at iteration 11, and the training continued until iteration 13 when it stopped. Furthermore, Figure 4.8 shows that the best validation performance is attained at epoch 11 with a mean square error of 0.057484.

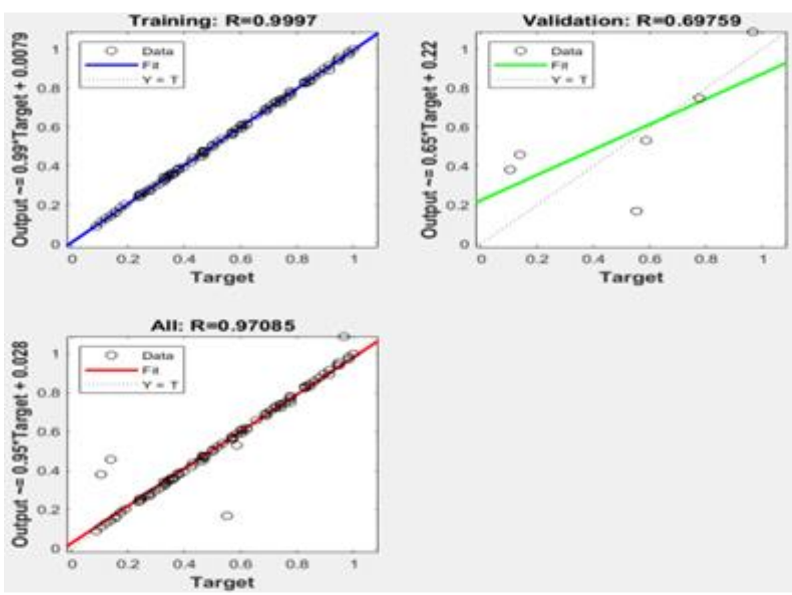


Figure 4.7: Regression plot

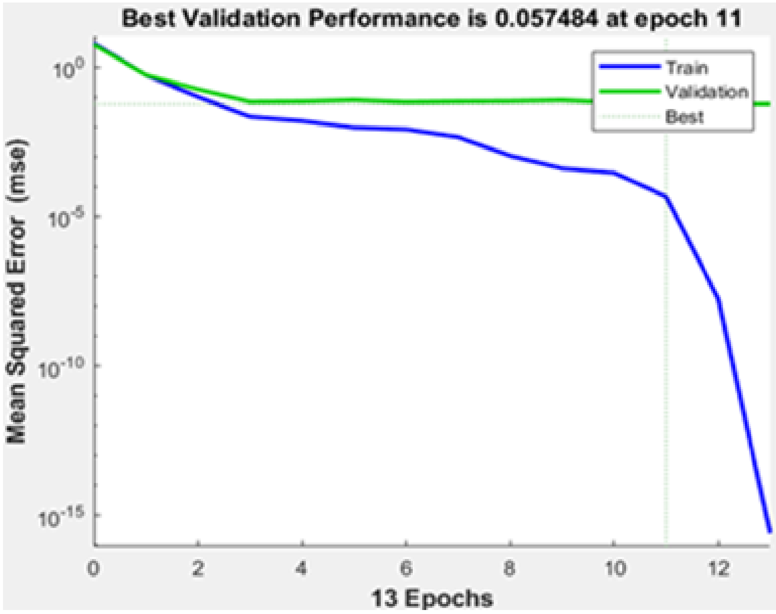


Figure 4.8: Performance plot

Error histograms depict the distribution of neural network errors on training instances. The error histogram shows the differences between the target and predicted values after training a feedforward neural network. In general, a normal distribution centered at 0 is expected for each output variable. Since the error values indicate how predicted values differ from target values, they can be negative or positive. The zero error line corresponds to the error axis's zero error value. Figure 4.9 demonstrates good error histogram characteristics; greater error is concentrated at the center and disappears as one move away from the center in both directions.

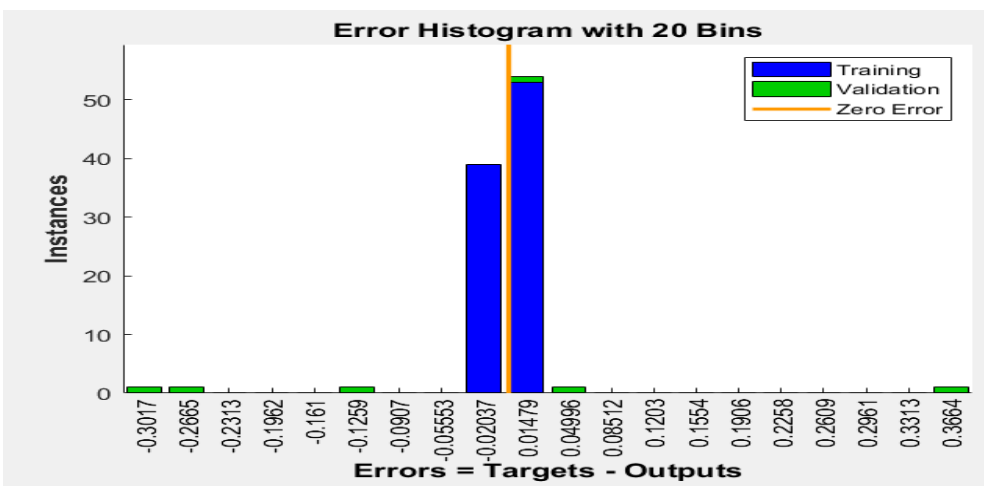


Figure 4.9: Error histogram

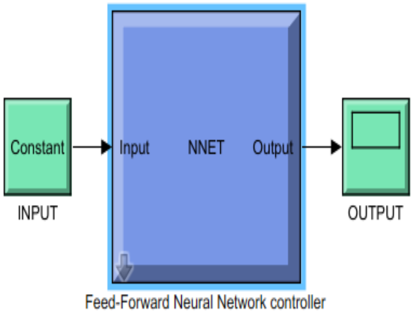
4.4 Artificial Neural Network Plant Model Final Parameters

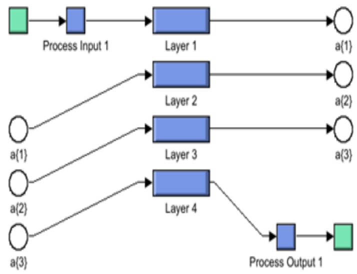
The final parameters of the artificial neural network plant model are summarized based on the results obtained above.

- Three hidden layers, 43 neurons per layer
- Learning rate, 0.0003
- Loss function, Mean squared error
- Optimization algorithm, Levenberg-Marquardt
- Best epoch, 11 with the mean square error of 0.057484
- Data division, Random function with 80% for training and 20% for validation
- Activation functions, sigmoid activation function for hidden layers, linear activation function for the output layer
- Overfitting prevention, after six consecutive increases in a validation error, training is halted.

4.5 Artificial Neural Network Internal Model Controller Model

The inverse model of the biological reactor of the wastewater treatment plant is built using an artificial neural network multilayer perceptron. The network was trained with ammonia, nitrate, nitrite, input dissolved oxygen, and output dissolved oxygen and one lag oxygen transfer coefficient as input variables and oxygen transfer coefficient as output variable using the Levenberg Marquard learning algorithm. Using a random division function, the input and output data sets are divided into 85 percent and 15 percent for training and validation sets, respectively. The training dataset is used to estimate unknown connection parameters between different nodes by backpropagating the error term from the output layer to each hidden layer and the input layer, whereas the validation dataset is used to evaluate a trained network's generalization capabilities for unexpected system properties. Several ANN architectures were tested, and the best results were obtained with three layers of 47 neurons, a sigmoid activation function on the hidden layers, and a linear activation function in the output node. When the validation errors increased and the maximum failed to reach six, the training was terminated. Finally, the weights and biases with the lowest validation error are returned. The best validation performance achieves a mean square error of 0.005874. Figures 4.10, 4.11, and 4.12 depict the ANN controller model, the number of layers, and the number of neurons deployed in the model, respectively.





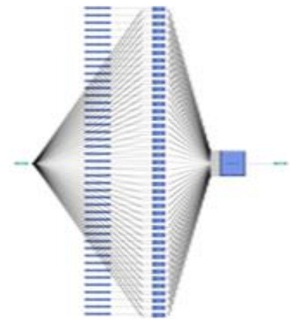


Figure 4. 10 ANN controller model Figure 4. 11 Layer structure Figure 4.12 Number of neurons

Figure 4.13 show that the mean percentage prediction deviation is 0.36 percent. As a result, the model is quite efficient in modeling the inverse model of the wastewater treatment plant's biological reactor. This result granted that the ANN controller accurately predicts the oxygen transfer coefficient based on the dissolved oxygen level in the biological reactor of the biological wastewater treatment plant.

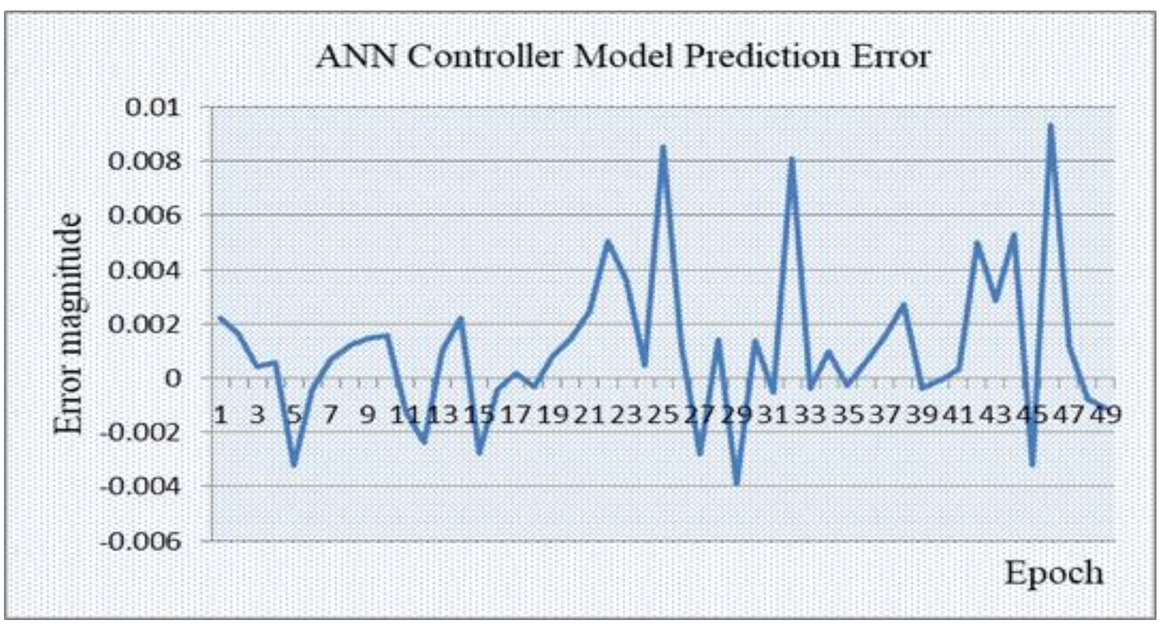


Figure 4.13: ANN-IMC prediction error

The ANN regression plots explain the neural network behaviors with training and validation data. The regression plots are shown in Figure 4.14, and they show the relationship between the output and the target values. With a 0.908871 coefficient of determination, the total model output in this model approximates 0.72 * target value + 0.022; this indicates a better data fit between model input and output data. The performance plot, in addition to the regression plot, displays the value of the performance function versus the number of iterations. It plots the results of training and validation tests. The performance plot of the training and validation states is shown in Figure 4.15. The validation performance reached the minimum value at iteration 4, and the training continued until iteration 10 when it ceased. Furthermore, figure 4.15 shows that the best validation performance is attained with MSE 0.0058744. The performance plot illustrated that there is no substantial difference between the train and validation curves until the best training point(i.e at epoch 4). This implies that there are no significant issues with the network training. When the number of training epochs increases, the MSE initially decreases, but it may begin to increase on the validation data set as the network overfit the training data.

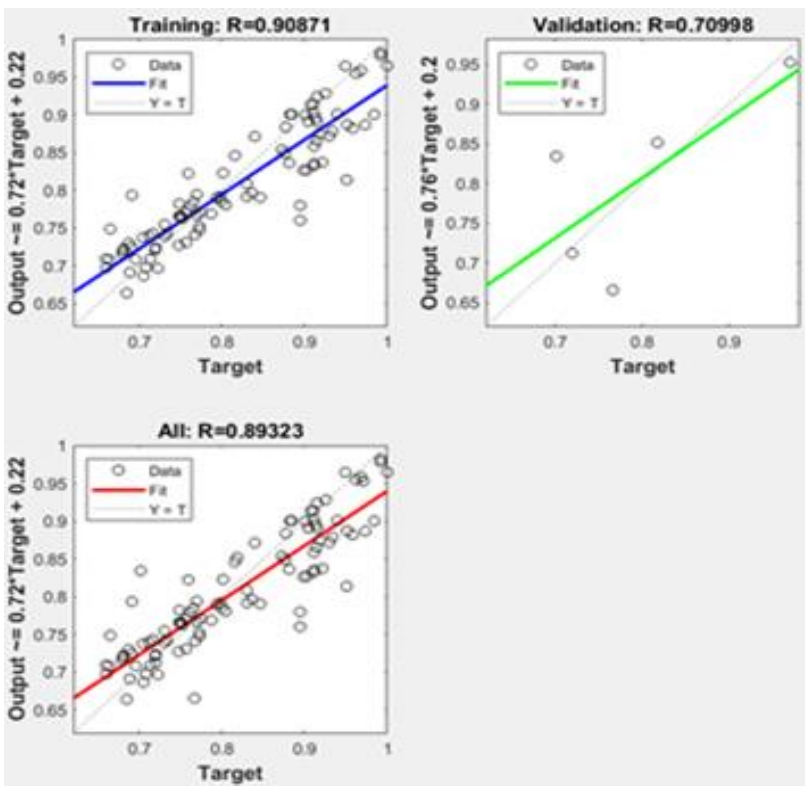


Figure 4. 13: Regression plot

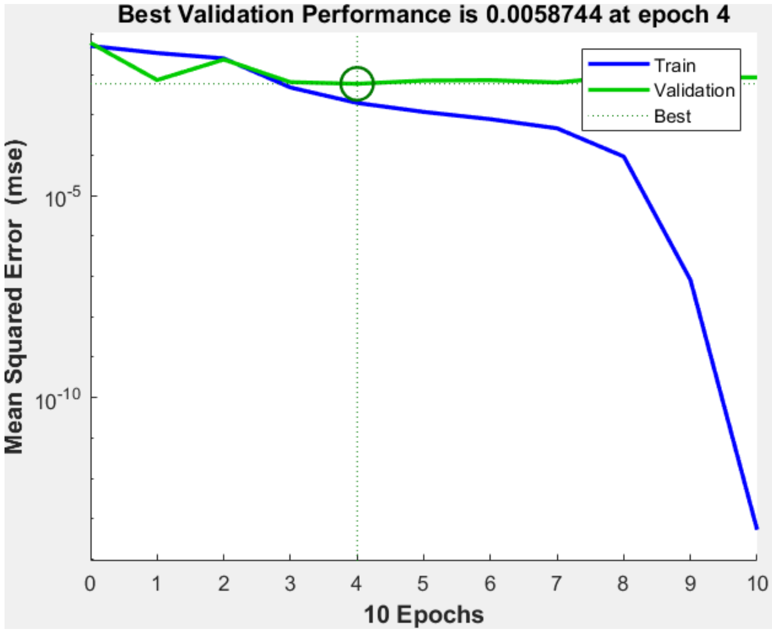


Figure 4. 14: Performance plot

Error histograms in Figure 4.15 illustrate the dispersion of neural network errors on training instances. The error histogram shows the differences between the target and predicted values after training a neural network. In general, a normal distribution centered at 0 is expected for each output variable. Since these error values indicate how predicted values differ from target values, they can be negative or positive. If the error distribution is more concentrated around the zero error line and decreases as we move away from the center, the training processes perform better as noted in Figure 4.15.

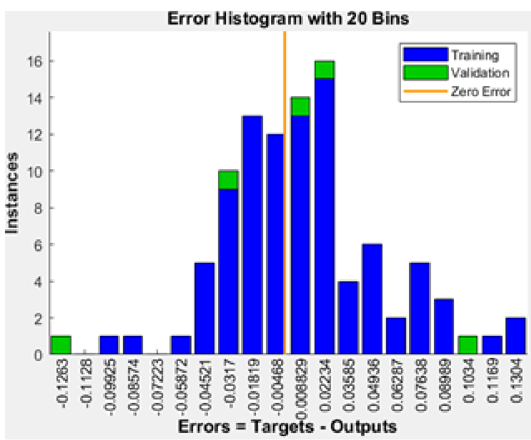


Figure 4. 15: Error histogram

4.6 Artificial Neural Network Internal Model Controller Final Structure

Based on the results obtained above, the final parameters of the ANN controller are summarized as follows.

- Hidden neurons, 47 per layer
- Hidden layers, 3
- Best epoch, 4 with the mean square error of 0.0058744
- Learning rate,0.0001
- Data division, Random function with 85% for training and 15% for validation
- Loss function, Mean squared error
- Optimization algorithm, Levenberg-Marquardt
- Activation functions, Sigmoid for hidden layers, linear for the output layer
- Overfitting prevention, early stopping after six training epochs without improvement in the loss of the validation set.

4.7 Artificial Neural Network Internal model controllers Implementation

This section demonstrates the use of direct inverse control strategies for set point tracking studies on a wastewater treatment plant. The control manipulated input, u (k), represents the oxygen transfer coefficient, while the controlled output, y(k+1), represents the dissolved oxygen concentration level. Y_{diff} is the difference between actual plant output (y(k+1)) and neural net forward model output ($\tilde{y}$(k+1)). The error (e) is generated from the reference set point and Y_{diff}, and then it passed through the low pass filter to remove noise. By trial and error, a low pass filter was selected to filter high-frequency signals while adjusting the controller sensitivity. The low pass filter output, along with the delayed plant and controller output, is injected into the neural net inverse model. Similarly, the neural net forward model receives delayed neural net inverse model output (oxygen transfer coefficient) and delayed plant output (dissolved oxygen level). For inverse models, one-step-ahead prediction and implementation are used, as is the method of training this inverse model. The control sampling time is chosen to be the same as the data acquisition sampling time of a single dimensionless time interval. Figure 4.16 depicts the implementation of the artificial neural network IMC strategy.

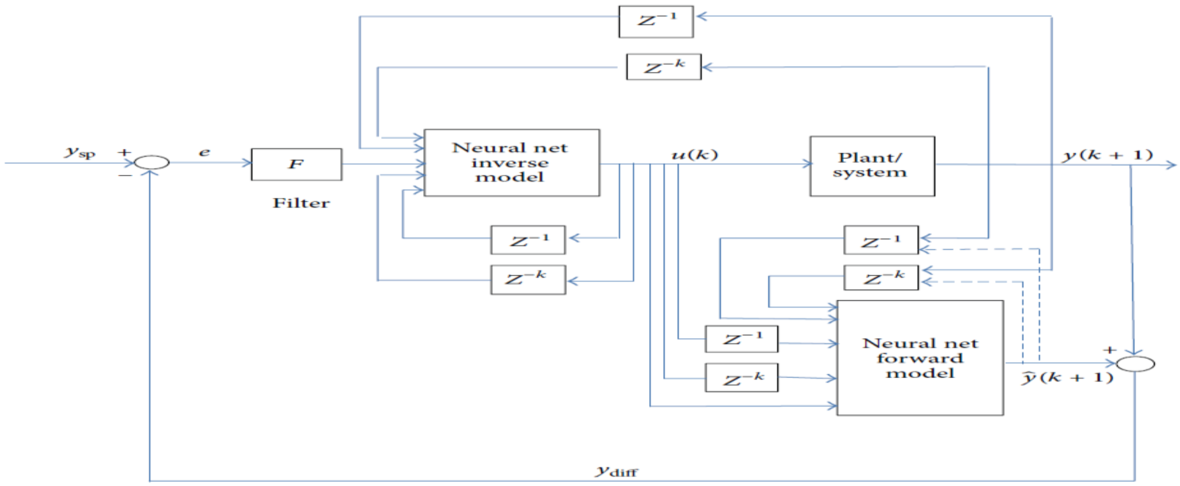


Figure 4. 16: Neural network in IMC implementation strategy (adapted from [37]

4.8 Artificial Neural Network Internal Model Control Simulation Result

Figure 4.17 shows the results for setpoint tracking performance. It is evaluated with the nominal steady-state value of 2g.m^{-3} for dissolved oxygen level in the biological reactor. After 12 seconds, the system response follows the setpoint trajectory very well, with only minor offsets.

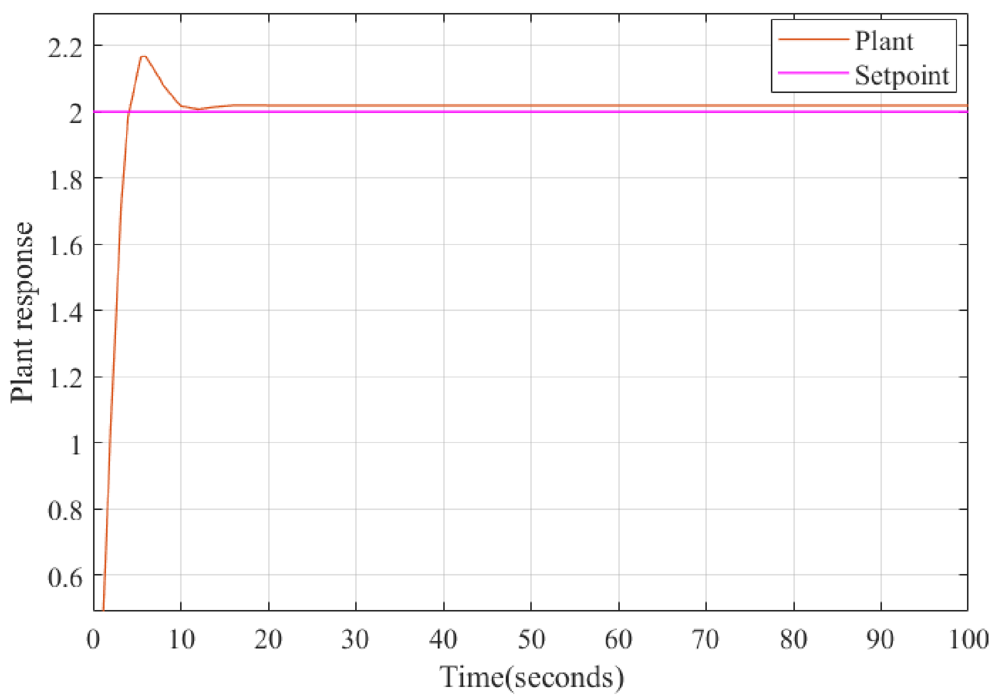


Figure 4. 17: Plant response with ANN-IMC

4.9 Artificial Neural Network Controllers Performance

The steady-state and transient characteristics such as delay time, rising time, peak time, settling time, overshoot, and steady-state error, were used to evaluate the performance of the ANN internal model controller. According to Figure 4.15, the best validation performance occurred at epoch 4, yet training continued until iteration 10 then it was terminated. As noted from this figure, the best validation performance score is 0.0058744 mean square error. Figure 4.18 shows that the wastewater treatment plant transient response takes a short transient time to reach its respective values compared with the PI controller. However, the ANN controller is not superior to the conventional PI controller in all control performance evaluation criteria; rather, it has some drawbacks such as high peak overshoot as shown in Figure 4.17. Even if this kind of shortcoming dipcated from simulation result, the ANN internal model controller provides a design option to model non-linear wastewater treatment process behavior based on input and output data only, eliminating the need for a plant mathematical model and its inverse. The plant response takes 1.75, 2.15, 5, and 16 seconds to reach the delay time, rising time, peak time, and settling respectively. In addition, the plant response trajectory commutes an 8.7% overshoot before coming back to a steady-state condition with a 1% steady-state error.

4.10 PI Controller Simulation Result

Based on trial and error techniques, a PI controller was designed and deployed for the same wastewater treatment plant used in the ANN-IMC case using Matlab Simulink tuner block. The following optimal PI controller parameters were obtained after tuning : ki = 0.161178250539944, KP = 0.01232. The PI control performances are also specified in terms of steady-state error and transient performance, such as delay time, rising time, peak time, settling time, overshoot, and steady-state error, as with the ANN-IMC controller. The simulation outputs and quantitative steady-state and transient performance are shown in Figure 4.18.

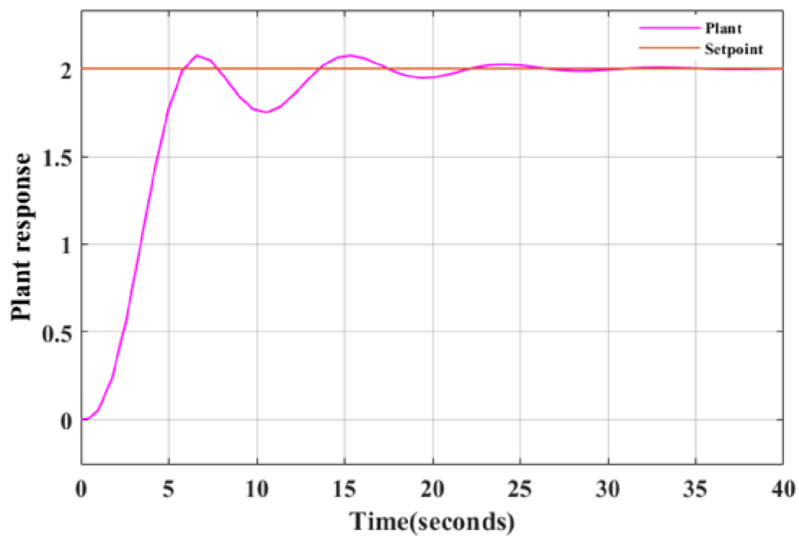


Figure 4. 18: Plant response with PI controller

Figure 4.18 demonstrated that the system response takes 3.4, 3.6, 6.83, and 19.4 seconds to reach the delay time, rising time, peak time, and settling time. In addition to this, the plant response trajectory commutes a 3.75% overshoot before returning to a steady-state with a 2.5% steady-state error.

4.11 Performance Comparison of ANN and PI Controller

The artificial neural network provides a tool for modeling the non-linear wastewater treatment plant and its corresponding controller without specifying any mathematical function, whereas the PI controller optimized its parameter by linearizing the real non-linear process which is prone to error. In terms of tracking, the ANN plant response behavior is depicted in Figure 4.17, where it is compared to the PI controller, ANN–IMC can track the set point (dissolved oxygen level) very well in a short transient time. As noted in Figures 4.17 and 4.18, the plant response takes 19.4 seconds to attain a steady-state for the case of the PI controller, whereas the ANN controller takes 16 seconds only. This demonstrates that the ANN controller outperforms the PI controller in terms of settling time by 3.4 seconds. As a result, we can conclude that ANN internal model control has the best performance: when settling time is taken into account, which represents a 21.25 percent improvement over the PI controller. When evaluating the rise time of the plant response for both controllers, the AAN-internal model controller improves rise time by 67%. This depicted in Table 4.2, the PI controller requires 3.6 seconds to reach its peak value, whereas the ANN controller requires only 2.15 seconds, a very significant improvement.

The disturbance rejection capability of the ANN-IMC and classical PI controllers was evaluated in addition to their transient performance. When a disturbance is applied to the plant, the plant response oscillation, and overshoot magnitude significantly increase in the case of the PI controller. Despite a slight overshoot, the plant response oscillation did not differ from the disturbance-free system when ANN-IMC was deployed. As a result, in terms of disturbance rejection, ANN-IMC outperforms the traditional PI controller.

When comparing ANN-IMC and PI controllers in terms of hardware and information requirements, the classical PI controller necessitates sensors to monitor dissolved oxygen levels in the biological reactor, which raises initial and maintenance costs and reduces system reliability. The artificial neural network, on the other hand, provides tools for developing an internal model controller without deploying a level sensor and/or a mathematical model of the plant. As a result, the artificial neural network has a high potential to improve the limitations of traditional PI controllers. In addition to this, the PI controller only requires information on the direction of change in gains between the control and manipulated variables, limiting control performance options. On the other hand, the artificial neural network plant and ANN-IMC models are developed based on process state variables. Hence, it opens up a significant opportunity for control performance enhancement.

The simulation results of the ANN-IMC show that it is not superior to the PI controller in all control performance indicators, rather it has some limitations in some performance parameters. The ANN-IMC has more overshoot than the PI controller, as shown in Table 4.2. Even though the ANN-IMC has a slightly higher overshoot than the conventional PI controller, the observed transient performance improvements compensate for this drawback and open up a significant opportunity for control performance enhancement.

Table 4. 2: ANN-IMC and PI controller transient performance

Performance Indicators	ANN-IMC	PI Controller
Delay time (seconds)	1.75	3.4
Rise time (seconds)	2.15	3.6
Peak time (seconds)	5	6.83
Settling time (seconds)	16	19.4
Overshoot (%)	8.7	3.75
Steady state error (%)	1	2.5

CHAPTER FIVE: CONCLUSION AND RECOMMENDATION

5.1 Conclusion

The primary goals of this study were to develop an artificial neural network wastewater treatment plant model and its associated controller based on the internal model control principle. To complete these tasks, raw operational laboratory test data collected from the kality wastewater treatment plant on the six operational variables such as nitrate, nitrite, ammonium, dissolved oxygen (input and output), and oxygen transfer coefficient, for three consecutive years from 2016 to 2018 G.C.

An artificial neural network plant model was developed to estimate the dissolved oxygen level in the final aerobic treatment tank using ammonia, nitrate, nitrite, dissolved oxygen, oxygen transfer coefficient, and previous dissolved oxygen output as input variables and current dissolved oxygen level as output variables, with the Levenberg Marquard learning algorithm. Using a random division function, the input-output datasets were divided into 80 percent and 20 percent for training and validation datasets, respectively. Several ANN architectures were tested, and the best result was obtained with three layers with 43 neurons per layer, a sigmoid activation function on the hidden layers, and a linear activation function in the output node. The dynamic and non-linearity property of wastewater treatment processes, as expected from the complex nature of processes, requires a large network structure.

Similarly, the Inverse neural network internal model controller was designed to control the oxygen transfer coefficient using six operational parameters as input features such as ammonia, nitrate, nitrite, and input dissolved oxygen, output dissolved oxygen, and one lag oxygen transfer coefficient, and oxygen transfer coefficient as an output variable using Levenberg Marquard as the learning algorithm. In the inverse neural network model, the dataset was divided into 85 and 15 percent using a random division function for training and validation sets, respectively, as in the direct artificial neural network model. Following several trials and errors, the final optimal inverse neural network model structure was determined to be three layers with 47 neurons per layer, a sigmoid activation function on the hidden layers, and a linear activation function in the output node. Thus, for both models, the process property is perfectly captured and represents from the above input-output real operational data with a very small prediction deviation.

The regression result of all datasets found in the direct artificial neural network plant model is 0.97085, which indicates the existence of better data fit in the model. This implies that explanatory variables (ammonia, nitrate, nitrite, input dissolved oxygen, oxygen transfer coefficient, and one lag output dissolved oxygen) explain 97 percent of the variation in dependent variables (dissolved oxygen level). By analogy, explanatory variables (ammonia, nitrate, nitrite, input dissolved oxygen, output dissolved oxygen, and one lag oxygen transfer coefficient) explain 90% of the dependent variables (oxygen transfer coefficient) in an inverse neural network model.

The mean percentage prediction deviation for the artificial neural network plant model is approximately 0.63 percent (under-predict). Similarly, the inverse neural network model's (controllers) mean percentage prediction deviation is around 0.36 percent. Since these prediction deviations in both models are very small, we can conclude that both models are quite efficient in modeling the biological wastewater treatment plant and associated internal model controller. The validation performance plot of the direct artificial neural network (plant) model revealed that the best epoch is 11 with an MSE of 0.057484, but training continued until iteration 13 and then stopped. The MSE is very low and acceptable in comparison to the model performance requirement established during the training phase. Similarly, with an MSE of 0.0058744, the inverse neural network (controller) model achieved the best validation performance at epoch 4.

This study shows that artificial neural network internal model control outperforms conventional PI controllers for dissolved oxygen control in the biological wastewater treatment plant. The wastewater treatment plant is inherently nonlinear, so specifying a functional form, as is done with a purely mathematical model-based controller and conventional PI controllers that rely on plant linearization and level sensors, does not always produce a good result. As a result, this study develops an ANN biological wastewater treatment plant and its corresponding internal model controller model without complex mathematical models or level sensors. The simulation results demonstrated that ANNs are capable of accurately representing dynamic and complex nonlinear biological wastewater treatment plants.

The comparisons artificial neural network controller (based on the internal model control principles) and conventional PI controller, the simulation result indicates that ANN internal model controller simulation results demonstrated that the plant response is faster than the PI counterpart. As noted from the simulation result, the plant response takes 19.4 seconds to attain a steady-state condition for the PI controller, whereas the ANN internal model controller plant response takes 16 seconds only. This demonstrates that the ANN internal model controller outperforms the PI controller in terms of settling time by 3.4 seconds, which represents a 21.25 percent improvement. When evaluating the rise time of the plant response for both controllers, the AAN internal model controller improves rise time by 67%, which is a dramatic improvement. The conventional PI controller necessitates sensors to monitor dissolved oxygen levels in the biological reactor, which raises initial and maintenance costs and reduces system reliability. Furthermore, the PI controller parameter update is based on the direction of change in gain between the control and manipulated variables only without enough information about plant state, which limits control performance improvement. However, since the artificial neural network plan and the ANN-IMC models were developed based on process state variable, which opens up a significant opportunity for control performance improvement.

5.2 Recommendation

The simulation results show that an artificial neural network-based internal model controller for dissolved oxygen control in the biological wastewater treatment plant outperforms the conventional PI controller. Since the artificial neural network plant and comptroller models were developed using real data collected from the Kality wastewater treatment plant and the simulation results were very promising, it is recommended to the Kality wastewater treatment plant operator to implement on the actual operation of the existing plant and similar projects which found under the construction and planning phases. For future work, as shown by the simulation results, the proposed controller has a slightly higher overshoot than the conventional PI controller. As a result, there is still room for improvement by implementing different artificial neural network architecture and other intelligent controller design methods, such as a genetic algorithm.

References

[1] W. Advisory, “Draft situational analysis for Ethiopia integrated urban sanitation and hygiene strategy, pp.48-50.” 2014.

[2] M. H. M. C. M. van Loosdrecht and G. A. E. Damir, *Biological Wastewater Treatment Principles, Modelling, and Design*. IWA, 2008.

[3] K. Peter, “Computational Intelligence Techniques for Control and Optimization of Wastewater Treatment Plants pp. 1-5.” 2016.

[4] M. Gabriel, “Towards Data-Driven Modeling of Saulekilen Wastewater Treatment Plant based on Artificial Intelligence.” 2019.

[5] Q.-L. C. Hong-Gui Han, Jun-Fei Qiao, “Model predictive control of dissolved oxygen concentration based on self-organized RBF neural networks.” 2012.

[6] P. Siliva R., Ramon V., Mario F., “PI dissolved oxygen control wastewater treatment for plant-wide nitrogen removal efficiency, pp.1-6.” 2018.

[7] P. Automation and A. Vienna, “Radial Basis Function Networks for Internal M o d e l Control,” vol. 298, no. 1995, pp. 283–298.

[8] D. Dionisi, *Biological Wastewater Treatment Processes Mass and Heat Balances*. 2017.

[9] G. Z. R. Qasim, *Wastewater Treatment and Reuse Theory and Design Examples Volume 2: Post-Treatment, Reuse, and Disposal*. 2018.

[10] R. Qasim, *Wastewater Treatment and Reuse Theory and Design Examples Volume 1: Principles and Basic Treatment*.

[11] H. Of, B. W. T. D., and O. of activated S. S. S. Edition, and A. C. van H. and J. G. M. van der Lubbe, *Handbook of Biological Wastewater Treatment Design and Optimisation of activated Sludge Systems*, Second Edi. IWA, 2012.

[12] V.Marcos, *Basic Principles of Wastewater Treatment volume 2*, IWA. 2007.

[13] M. Haseebuddin, “Internal Model Control For Nonlinear Dynamic Plants Using U-Model,” no. May 2004.

[14] R. S.Burns, “Advanced control engineering books, A division of Reed Educational and Professional Publishing Ltd, pp. 347-358.” 2001.

[15] J. A. K. S. J. oos P. L. V. B. L. R. De Moor, *Artificial Neural Networks For Modelling And*

Control Of Non-Linear Systems. Kluwer Academic Publishers, 1996.

[16] J. Vanderplas, *Python Data Science Handbook*. 2017.

[17] F. Takenaka, T. Hashimoto, and T. Hongo, *Artificial Neural Networks A Practical Course*, vol. 50. 1954.

[18] Simon Haykin, *Neural networks, and learning*, vol. 127. 1992.

[19] B. K. Bose, "Modern Power Electronics and AC Drives," *Prentice-Hall*. p. 738, 2002.

[20] K. Mehrotra, K. M. Chilukuri, and S. Ranka, "Elements of Artificial Neural Networks (Complex Adaptive Systems)," p. 400, 1996.

[21] G. Montavon, "Introduction to Neural Networks," *Lect. Notes Phys.*, vol. 968, no. November, pp. 37–62, 2020.

[22] J. Feng and S. Lu, "Performance Analysis of Various Activation Functions in Artificial Neural Networks Performance Analysis of Various Activation Functions in Artificial Neural Networks," 2019.

[23] S. S and S. P, *Artificial Neural Networks – Architectures And Applications*. 2010.

[24] O. Mahmoud, F. Anwar, and M. J. E. Salami, "Learning Algorithm Effect On Multilayer Feed Forward Artificial Neural Network Performance," Vol. 2, No. 2, Pp. 188–199, 2007.

[25] D. Belegundu, *Optimization Concepts, and Applications In Engineering*. 2011.

[26] Y. Kwon, Y. Kwon, D. Chung, and M. Lim, "The Comparison of Performance According to Initialization Methods of Deep Neural Network for Malware Dataset," no. 4, pp. 57–62, 2019.

[27] S. Caraman, M. Sbarciog, and M. Barbu, "Predictive Control of a Wastewater Treatment Process The model of the wastewater treatment process," *Int. J. Comput. Commun. Control*, vol. 2, no. 2, pp. 132–142, 2007.

[28] E. S. S. Tejaswini, G. Uday Bhaskar Babu, and A. Seshagiri Rao, "Design and evaluation of advanced automatic control strategies in a total nitrogen removal activated sludge plant," vol. 35, 2021.

[29] J. F. Qiao, Y. C. Bo, W. Chai, and H. G. Han, "Adaptive optimal control for a wastewater treatment plant based on a data-driven method," vol. 67, 2013.

[30] S. I. Samsudin, M. F. Rahmat, N. A. Wahab, M. S. Gaya, and M. C. Razali, "Application of adaptive decentralized PI controller for activated sludge process," 2013.

[31] M.Stefano, "Optimal control of the activated sludge process," 2015.

[32] W. Shen, X. Chen, M. N. Pons, and J. P. Corriou, “Model predictive control for wastewater treatment process with feedforward compensation,” vol. 155, 2009.

[33] G.Dreyfus, *Neural Networks Methodology, and Applications*. 2018.

[34] H. R. Maier, A. Jain, G. C. Dandy, and K. P. Sudheer, “Methods used for the development of neural networks for the prediction of water resource variables in river systems: Current status and future directions,” 2010.

[35] Y. Sugomori, B. Kaluža, F. M. Soares, and A. M. F. Souza, *Deep Learning - Practical Neural Networks with Java*. 2017.

[36] R. S. Burns, “Advanced Control Engineering,” 2001.

[37] M. A. Hussain, J. Mohd Ali, and M. J. H. Khan, “Neural network inverse model control strategy: Discrete-time stability analysis for relative order two systems,” *Abstr. Appl. Anal.*, vol. 2014, 2014.

Appendix

Appendix A: Matlab Code for Plant Model

```
a= readmatrix('C:\Users\HP\Desktop\ANN-PLNAT1.csv');
 x= a(:,1:6);
 y = a (:,7);
 m = length(y);
 for i = 1:6
 x2(:,i) = (x(:,i)- min(x(:,i)))/(max(x(:,i))- min(x(:,i)));
  end
  X2
  xt = x2';
  yt= y';
  hiddenLayerSize  = [ 43 43 43 ];
  net =  feedforwardnet(hiddenLayerSize);
  net.trainFcn = 'trainlm';
  net.trainparam.lr = 0.003;
  net.trainParam.max_fail= 6;
  net.trainParam.mu=  0.0003;
  net.trainParam.mu_max=  100000000;
  net.divideParam.trainRatio = 80/100;
  net.divideParam.valRatio = 20/100;
  net.divideParam.testRatio = 0/100;
  [net, tr]= train(net, xt,yt);
  plotperf(tr);
  tr.best_epoch;
  yTrain = net(xt(:,tr.trainInd));
  yTrainTrue = yt(tr.trainInd);
  mse_traing= sqrt(mean(yTrain- yTrainTrue).^2);
  yval = net(xt(:,tr.valInd));
  yvalTrue = yt(tr.valInd);
```

```
mse_val= sqrt(mean(yval- yvalTrue).^2);
plot(yTrainTrue,yTrain,'x'); hold on;
plot(0:6, 0:6); hold off;
net = gensim(net,-1);
```

Appendix B: Training Data

DOi	NH3	NO3	NO2	OTC	DO
0.75	21.75	2.1	74	9185.7	2.75
1.79	18.75	4.4	35	7588.8	3.5
1.89	12.5	5.6	75	7648.9	3.99
1.67	NA	2.1	33	7127.4	2.38
0.57	29.5	2.2	56	9015.1	3.2
0.32	36.75	6.7	36	7309.5	2.46
0.74	10	4.4	68	6930	1.38
0.43	55	3.3	56	9463.3	1.6
0.4	25	3.3	43	7172.4	1.43
0.5	95.5	5.2	49	8743.8	1.94
1.07	20	4.7	55	9550.4	2.2
0.8	35.75	2.2	43	8273.4	1.9
1.54	22.5	5.34	67	6877.1	2.64
0.96	29.75	6.7	37	7541.8	2.07
0.32	39.5	6.8	325	7168.9	4.81
3.1	74.5	4.9	87	9081.1	2.07
0.52	3	8.8	9	7106.3	4.5
0.8	3	4.4	38	8335.7	0.6
0.7	27	5.4	25	7715.2	0.5
0.32	40.75	3.2	18	9968.8	2.7
1.84	38	2.6	5	6845.7	5.8
1	47	5.6	8	8153.2	5.47
1.8	38	2.3	7	7136.4	1.6
0.7	12	4.2	33	8129.1	4.1
1.5	20.1	2.1	68	7844.7	3.4
1	23.3	4.4	56	6768.9	4.3
1.1	17.3	4.1	51	7669.7	3.3
1.4	16.7	2.5	26	6990.9	1.1
0.7	19.7	4.1	13	9150.3	2.9
1.7	29	3.3	57	8916.3	5.2
1.7	19.5	4.1	63	9270.4	5.7
1.7	8.7	2.8	22	7567.2	4.9
0.8	29.5	4.6	71	7453.2	1.9
1.3	11.4	4.2	44	6786.3	2
1.7	36.3	4.9	60	9085.7	4.2
1.5	25.7	2.8	65	6611.4	4.5
1.7	17.9	4.8	56	7668.4	5.5
1.2	37	5.2	26	9088.8	3.3
0.8	28.3	2.4	65	7986.9	3.4
0.9	26.6	5	36	9479	2.2

1.4	15.6	3.4	21	7639.8	1.4
1.1	10.5	3.2	47	6570.9	2.7
0.9	14.8	2.2	10	8816	1
1.6	35.6	5	36	7017.8	0.9
1.7	13.9	4.9	14	7024.5	4.3
1.1	30.5	2.3	71	7629.3	3.5
0.8	32.6	4.8	24	9365.8	1.5
1.4	11.2	3.8	39	8273.8	4.5
1.5	6.1	3	32	7449.4	0.7
0.9	32.9	3.7	20	7459.3	3.1
1.5	31	4.6	50	9120.8	5.6
1.5	24.2	3	11	6862	0.9
1	8.7	3.7	59	9650.7	4.3
0.7	13.1	2.3	71	8981	2.7
0.8	20.9	4	20	9114.2	4.8
1.7	35	2.8	9	7096.6	5.1
0.6	5.4	2.3	28	8734.8	2
1.1	29.9	4.9	47	9305.5	1.4
1.1	18.6	2.9	32	7542.7	1.9
0.5	28.9	2.8	23	9225.1	4
0.7	18.5	5.3	67	9810.6	2.1
1.5	9.7	3.3	20	9080.5	3.5
0.6	17.8	5.1	53	6888	1.9
1.6	19.6	2.9	59	8916.7	3
0.8	8.3	2.1	66	9678.1	3.3
1.1	24.9	2.4	43	6576.6	1.8
1.5	6.5	5.3	25	7698.4	4.4
1.4	25.9	5.3	9	7199.2	4
1.1	11.6	2.4	46	7954.7	2.7
1.6	6.7	5.1	10	9702.7	1.7
1.5	23.1	2.7	5	8435.4	5.5
1	35.9	5.3	33	7056.9	4.9
0.9	17.2	3.9	23	7969.4	3.3
0.7	8.3	2.2	59	8369.7	2.7
0.8	6.9	5.3	63	9890.8	4.8
1.1	10.3	4.3	30	7173.2	3.6
1.7	32.7	5.3	73	9888.6	0.6
1.3	19.3	2.9	20	8784.9	3.8
0.5	7.9	4.3	71	9078.7	2.9
1	16.6	5.2	16	9105.8	4.2
1.4	9.6	5.1	46	8015.8	2.2
1.4	14.7	2.9	60	9119.3	4.2
1.3	19	4.1	58	6624.6	5.3

0.4	35.3	3	25	9590.5	0.8
1.8	28.6	3.3	13	8957.6	4.5
1.7	11.6	2.5	6	9484.4	5
1.1	32.3	3.9	34	7906.6	3.6
1.3	13	2.2	50	7483.4	2
1.7	30.4	4.6	39	7703.5	3.5
1	26.2	4.9	49	6776	2.7
1.1	6.9	4.6	48	8696.7	0.8
0.9	35.9	5.1	70	6823.2	5
1.8	19.8	4.1	18	7283.3	3.4
0.8	19.2	3	56	8998	2.2
0.4	13.9	2.5	53	8796.2	2.5
0.6	33.9	4	39	7476.9	2.3
1.1	28.5	4.1	18	7279.8	2.9
1	25.1	5.2	21	9046.5	5.3
1.3	30.3	3.6	41	7496.4	5.7

Source: Kality Wastewater Treatment

Appendix C: ANN –IMC Simulink Model