Spring 2013

Modeling and optimization of wastewater treatment process with a data-driven approach

Xiupeng Wei
University of Iowa

Recommended Citation

Wei, Xiupeng. "Modeling and optimization of wastewater treatment process with a data-driven approach." PhD (Doctor of Philosophy) thesis, University of Iowa, 2013.
http://ir.uiowa.edu/etd/2659.

MODELING AND OPTIMIZATION OF WASTEWATER TREATMENT PROCESS WITH A DATA-DRIVEN APPROACH

by

Xiupeng Wei

An Abstract

Of a thesis submitted in partial fulfillment of the
requirements for the Doctor of Philosophy degree
in Industrial Engineering in
the Graduate College of
The University of Iowa

May 2013

Thesis Supervisor: Professor Andrew Kusiak

ABSTRACT

The primary objective of this research is to model and optimize wastewater treatment process in a wastewater treatment plant (WWTP). As the treatment process is complex, its operations pose challenges. Traditional physics-based and mathematical-models have limitations in predicting the behavior of the wastewater process and optimization of its operations.

Automated control and information technology enables continuous collection of data. The collected data contains process information allowing to predict and optimize the process.

Although the data offered by the WWTP is plentiful, it has not been fully used to extract meaningful information to improve performance of the plant. A data-driven approach is promising in identifying useful patterns and models using algorithms versed in statistics and computational intelligence. Successful data-mining applications have been reported in business, manufacturing, science, and engineering.

The focus of this research is to model and optimize the wastewater treatment process and ultimately improve efficiency of WWTPs. To maintain the effluent quality, the influent flow rate, the influent pollutants including the total suspended solids (TSS) and CBOD, are predicted in short-term and long-term to provide information to efficiently operate the treatment process. To reduce energy consumption and improve energy efficiency, the process of biogas production, activated sludge process and pumping station are modeled and optimized with evolutionary computation algorithms.

Modeling and optimization of wastewater treatment processes faces three major challenges. The first one is related to the data. As wastewater treatment includes physical, chemical, and biological processes, and instruments collecting large volumes of data. Many variables in the dataset are strongly coupled. The data is noisy, uncertain, and incomplete. Therefore, several preprocessing algorithms should be used to preprocess the data, reduce its dimensionality, and determine import variables. The second challenge is in the temporal nature of the process. Different data-mining algorithms are used to obtain accurate models. The last challenge is the optimization of the process models. As the models are usually highly nonlinear and dynamic, novel evolutionary computational algorithms are used.

This research addresses these three challenges. The major contribution of this research is in modeling and optimizing the wastewater treatment process with a data-driven approach. The process model built is then optimized with evolutionary computational algorithms to find the optimal solutions for improving process efficiency and reducing energy consumption.

Abstract Approved: ______________________________________

Thesis Supervisor

____

Title and Department

____

Date

MODELING AND OPTIMIZATION OF WASTEWATER TREATMENT PROCESS WITH A DATA-DRIVEN APPROACH

by

Xiupeng Wei

A thesis submitted in partial fulfillment of the
requirements for the Doctor of
Philosophy degree in Industrial Engineering in
the Graduate College of
The University of Iowa

May 2013

Thesis Supervisor: Professor Andrew Kusiak

Graduate College
The University of Iowa
Iowa City, Iowa

CERTIFICATE OF APPROVAL

PH.D. THESIS

This is to certify that the Ph.D. thesis of

Xiupeng Wei

has been approved by the Examining Committee
for the thesis requirement for the Doctor of Philosophy
degree in Industrial Engineering at the May 2013 graduation.

Thesis Committee: ______________________________
Andrew Kusiak, Thesis Supervisor

____
Yong Chen

____
Pavlo A. Krokhmal

____
Pablo M. Carrica

____
M. Asghar Bhatti

To My Parents and Family

The important thing in life is to have a great aim, and the determination to attain it.

Goethe

ACKNOWLEDGMENTS

I am very grateful to my advisor, Professor Andrew Kusiak, for his guidance and devotion to the research throughout my Ph.D. studies. I really appreciate his advice on research my career, and life.

I would also like to thank Professor Yong Chen, Professor Pavlo A. Krokhmal, Professor Pablo M. Carrica and Professor M. Asghar Bhatti for their willingness serve on my thesis defense committee and review my Ph.D research. I appreciate their valuable comments.

I thank all current members of the Intelligent System Laboratory and those who have graduated. Their encouragement, support and sharing happy and not so happy times make me enjoy the days in the laboratory and the College.

At last and most importantly, I would like to express my sincere and deep love to my parents, my family and friends, who spiritually and physically supported me in pursuing the Ph.D. degree program. They are my motivation to explore this world.

ABSTRACT

The primary objective of this research is to model and optimize wastewater treatment process in a wastewater treatment plant (WWTP). As the treatment process is complex, its operations pose challenges. Traditional physics-based and mathematical-models have limitations in predicting the behavior of the wastewater process and optimization of its operations.

Automated control and information technology enables continuous collection of data. The collected data contains process information allowing to predict and optimize the process.

Although the data offered by the WWTP is plentiful, it has not been fully used to extract meaningful information to improve performance of the plant. A data-driven approach is promising in identifying useful patterns and models using algorithms versed in statistics and computational intelligence. Successful data-mining applications have been reported in business, manufacturing, science, and engineering.

The focus of this research is to model and optimize the wastewater treatment process and ultimately improve efficiency of WWTPs. To maintain the effluent quality, the influent flow rate, the influent pollutants including the total suspended solids (TSS) and CBOD, are predicted in short-term and long-term to provide information to efficiently operate the treatment process. To reduce energy consumption and improve energy efficiency, the process of biogas production, activated sludge process and pumping station are modeled and optimized with evolutionary computation algorithms.

Modeling and optimization of wastewater treatment processes faces three major challenges. The first one is related to the data. As wastewater treatment includes physical, chemical, and biological processes, and instruments collecting large volumes of data. Many variables in the dataset are strongly coupled. The data is noisy, uncertain, and incomplete. Therefore, several preprocessing algorithms should be used to preprocess the data, reduce its dimensionality, and determine import variables. The second challenge is in the temporal nature of the process. Different data-mining algorithms are used to obtain accurate models. The last challenge is the optimization of the process models. As the models are usually highly nonlinear and dynamic, novel evolutionary computational algorithms are used.

This research addresses these three challenges. The major contribution of this research is in modeling and optimizing the wastewater treatment process with a data-driven approach. The process model built is then optimized with evolutionary computational algorithms to find the optimal solutions for improving process efficiency and reducing energy consumption.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Motivation

To protect clean water, wastewater needs to be treated before discharge back to the nature. Wastewater treatment plants (WWTPs) involve several different processes to treat wastewater at different stages.

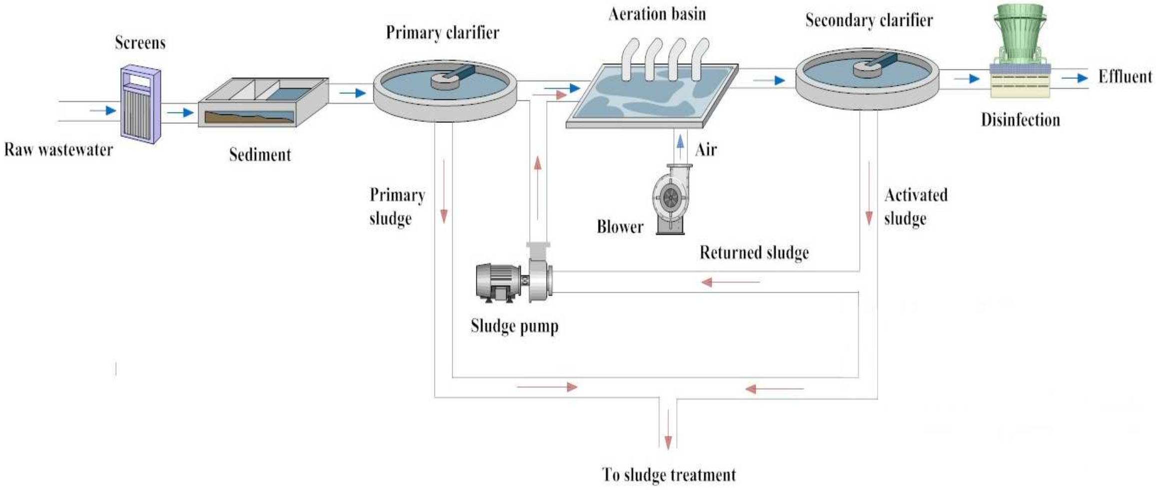


Figure 1.1. Flow schematic diagram of a typical WWTP

A flow diagram of a typical WWTP process is shown in Figure 1.1. The collected wastewater enters the plant and passes through bar screens. The large items such as rags, sticks are screened and are disposed later. After screening, the influent wastewater enters a wet well and then being pumped to primary clarifiers. After maintaining a retention time of 1 to 2 hours, the scum floats to the surface where it is removed by a skimmer. Then the wastewater is delivered to aeration tanks by intermediate pumps. Process air is provided by single-stage, centrifugal blowers to and around the aeration tanks. During normal operation partial of the sludge from the secondary clarifiers, called returned activated sludge (RSL), enters into aeration tanks through sludge pumps. When the RSL and the wastewater are mixed together, microorganisms in activated sludge use the oxygen provided by the fine bubble diffusers located on the bottom of the aeration basins to break down the organic matters. The remaining sludge from the secondary clarifiers and the sludge from the primary clarifiers are pumped to the anaerobic digesters to produce biogas. The liquid from the secondary clarifiers flows to the chlorine contact tanks where chlorine is injected into the flow to kill most bacteria, and then the final effluent is discharged to the river.

During the whole process, physical, chemical and biological sub-processed involved. The process is highly nonlinear and dynamics. The WWTPs are controlled by experience and some small scale experimental results. Therefore, the plants are not well optimally operated. The energy consumed by raw wastewater boosting pumps and air blowers are partially wasted. Heavy rainfall may overwhelm the plant, causing spills and overflows due to unaccurate estimation of plant influent flow based on experience.

Therefore, modeling and optimization of wastewater treatment process has been an interest of industries and researchers. However, it is difficult to use traditional methods to perform this task due to the complex and nonlinear nature of the process, such as physical and mathematical based models.

With the development of the information technology and automated instruments, large volume process data is recorded in WWTPs. This enables another approach, data-driven approach, to model and optimize the process. A data-driven approach is a promising method for finding useful information through the data. It is the process of finding patterns by algorithms versed on the crossroads of statistics and computational intelligence. Successful data-mining applications have been reported in business and marketing, manufacturing, science and engineering.

With the data-driven approach, the treatment process can be accurately represented by models without solving complex physical and mathematical equations. The models can be used to predict the behavior of the plant and be solved with evolutionary algorithms to find the optimal control settings to save energy and improve energy efficiency.

1.2 Research objectives

The primary goal of this research is to provide a systematic data-driven approach to model and optimize the wastewater treatment process. The goal can be achieved with the following objectives:

1) Forecast the plant influent flow based on a novel way to provide useful influent flow information to plant management.
2) Predict the total suspended solids in wastewater to provide information to select chemical and biological control strategy.
3) Predict CBOD in wastewater.
4) Model and optimize the wastewater boosting process to reduce energy consumption by pumps
5) Model and optimize the activated sludge process to improve the energy efficiency
6) Model and optimize the sludge digestion process to maximize the biogas production.

To the author’s knowledge, there’s no existing or completed project that has accomplished the above objectives. In this research, the six objectives are accomplished with data-mining techniques and evolutionary algorithms developed here. The model and methods developed in this thesis can be extended to other industrial process problems.

In Chapter 2, the plant influent flow at a WWTP is predicted with two data-driven neural networks. To satisfy the spatial and temporal characteristics of the influent flow, rainfall data collected at 6 tipping buckets, radar data measured by a radar station and historical influent data are used as model inputs. The static MLP neural network provides good prediction accuracy up to 150 min ahead. To extend the time horizon of predictions, to 300 min, a dynamic neural network with an online corrector is proposed.

In Chapter 3, data-mining algorithms are applied to predict total suspended solids (TSS) in wastewater. Numerous scenarios involving carbonaceous biochemical oxygen demand (CBOD) and influent flow rate are investigated to construct the TSS time-series. The multi-layered perceptron (MLP) model performed best among the five different data-mining models that are derived for predicting TSS. The accuracy of the predictions is improved further by an iterative construction of MLP algorithm models.

In Chapter 4, numerous models predicting carbonaceous biochemical oxygen demand (CBOD) are presented. The performance of individual seasonal models is found to be better for fall and winter seasons, when the CBOD values were high. For low CBOD values, the modified seasonal models are found most accurate. Predictions for up to five days ahead are performed.

In Chapter 5, a data-driven approach is presented to model and optimize wastewater pumping process to reduce pumping energy cost. Data-mining algorithm, multilayer perceptron neural network, is used to build the pumping energy model. The optimization problem formulated by integrating the model is solved by the proposed two level integration algorithm to find optimal pump configurations and pump speed settings. Significant energy reduction is observed when the pumping station running under optimized optimal settings.

To save energy while maintaining effluent quality, a data-driven approach for optimization of energy efficiency of the activated sludge process is presented in Chapter 6. A dataset from a wastewater treatment plant is used to formulate the objectives of the model. The optimal concentrations of dissolved oxygen that would minimize energy consumption and effluent pollutants are determined with an evolutionary computational algorithm. Three scenarios with different preference between energy savings and effluent quality are investigated.

In Chapter 7, optimization of biogas production process in a wastewater treatment plant is presented. The process model is developed using routinely collected data categorized as controllable and uncontrollable variables. A multi-layer perceptron neural network is applied to construct the optimization model. Optimizing single variable and all variables are both investigated. An evolutionary algorithm is used to solve the formulated problem.

Chapter 8 presents the conclusions and future research.

CHAPTER 2

SHORT-TERM FORECASTING OF INLUENT FLOW RATE

2.1 Introduction

The influent flow to a wastewater treatment plant (WWTP) has a significant impact on the energy consumption and treatment process [1]. To maintain the required water level in a wet well, the number of raw wastewater pumps should be arranged based on the quantity of coming influent flow. Optimal arrangement and scheduling of pumping system can greatly reduce electricity usage. The pollutants, such as total suspended solids (TSS) and biochemical oxygen demand (BOD) in the wastewater are also correlated to the influent flow [2]. The treatment process should be adjusted accordingly to the pollutants concentrations in the influent. For example, high BOD concentration requires longer aeration time and supply of more oxygen [3]. Thus, it is important to predict the influent flow at future time horizons in order to well manage the plant and control the effluent quality.

Accurate prediction of the influent flow, however, is still a challenge in wastewater industry. A WWTP usually receives wastewater from municipal sewers and storm waters from areas around the plant [4]. The quantity of the generated wastewater or precipitation may vary across different areas. In fact, to account for the influent flow to a WWTP, spatial and temporal correlations should be considered.

Several studies have focused on developing models to predict the influent flow [5-10]. Hernebring *et al.* [11] presented an online system for short-term sewer flow forecasts optimizing the effects of the receiving wastewater. A more complex phenomenological model has been built in [12] based on one year of full-scale WWTP influent data .It included diurnal phenomena, a weekend effect, seasonal phenomena and holiday periods. Carstensen *et al.* [13] reported prediction results of hydraulic load for urban storm control of a WWTP. Three models, a simple regression model, an adaptive grey-box model and a complex hydrological and full dynamic wave model, represented three different levels of complexity and showed different ability to predict water loads one hour ahead. Though these models have taken into account temporal correlations of the influent flow, however, they have ignored the spatial feature of the influent flow.

The wastewater processing industry has used physics-based deterministic models to estimate the influent flow. Online sensors have been used to provide flow information at sub-pumping stations. Based on the empirical data, such as the distance between the sub-station and the WWTP, the sewer piping size, the influent flow could be roughly estimated and calibrated by the historical data to improve the estimation accuracy [14]. Such simple models did not fully consider temporal correlations of the influent flow. In case of large rainfalls or lack of sensors covering large areas, the predicted influent flow may have carried a significant error.

In this work, short-term prediction (300 min ahead) of the influent flow of a WWTP is presented. To take account of the spatial-temporal characteristics of the influent flow, rainfall data measured at different tipping buckets, radar reflectivity data covering the entire area handled by the WWTP, and the historical influent data to the plant are used to build the prediction models. The rainfall data provided by tipping buckets offers valuable precipitation measurements. Weather radar provides spatial-temporal data covering large area including the places not covered by the tipping buckets. The high frequency of radar data makes them useful to forecast the rainfall several hours ahead. The historical influent time series data contains temporal influent information used to predict the influent flow.

Neural networks (NNs) are used to build prediction models in the research reported in this research. Successful applications of NNs have been reported in literature [15-20]. Kriger and Tzoneva [21] developed a NN model to predict the chemical oxygen demand of the influent. A three-layer feed forward NN has been applied to the effluent BOD [22]. The NN models provided satisfactory prediction results.

The remainder of the chapter is organized as follows. Section 2.2 describes the data collection, preparation and preprocessing as well as the metrics used to evaluate accuracy of models. Section 2.3 presents a static multi-layer perceptron (MLP) neural network which is employed to build prediction model of the influent flow. In Section 2.4, a data-driven dynamic neural network is proposed to solve the time lag problem appearing in the models by the static MLP neural network. The neural network structure and the computational results are discussed.

2.2 Data collection and processing

2.2.1 Data cleaning

The plant influent flow data and other data not specified are collected at the Wastewater Reclamation Facility (WRF), located in Des Moines, Iowa, United States. WRF operates a 97 million gallon per day (MGD) regional wastewater treatment plant in southeast Des Moines, Iowa. The peak influent flow rate can be as high as 200 MGD. The plant was mainly constructed in the mid 1980s to treat municipal wastewater and storm water from the greater Des Moines metropolitan area. The activated sludge process is used to biologically remove organics in the water.

To build the influent flow prediction model of WRF, the model inputs include historical influent data, rainfall data and radar reflectivity data. The influent flow data is collected at 15-s intervals at WRF. It is preprocessed to 15-min to have the same frequency as the rainfall data.

The rainfall data was measured at six tipping buckets (blue icons in Figure 2.1) in the vicinity of WRF (red icon in Figure 2.1). As WRF receives wastewater from a large area, including rainfall data in the model inputs satisfies the spatial characteristic of the influent flow. Figure 2.2 shows the difference of rainfall rates at these tipping buckets at certain times. It illustrates that the rainfall is location dependent and may be very despite the proximity of the tipping buckets. It indicates the importance of rainfall data to the influent flow prediction model.

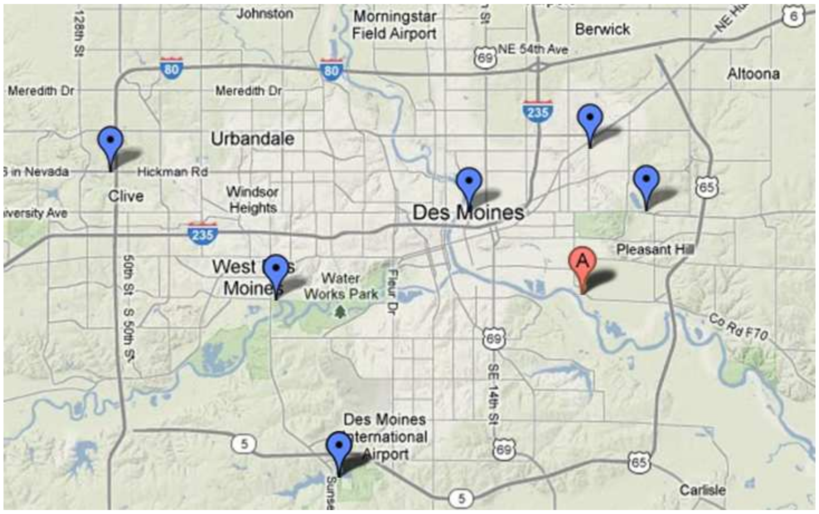


Figure 2.1. Location of tipping buckets and WRF

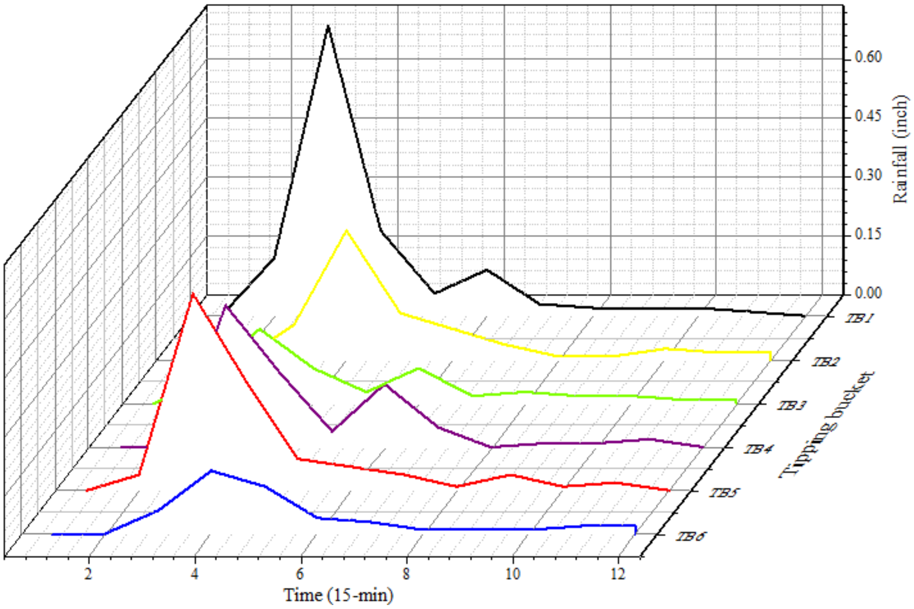


Figure 2.2. Rainfall at six tipping buckets

The rainfall graphs in Figure 2.2 illustrate the runoffs at several locations rather than completely reflect the precipitation at entire area covered by WRF. Therefore, the radar reflectivity data is proposed to provide additional input to for influent flow prediction. The NEXRAD-II radar data used in this research is from the weather station KDMX in Des Moines, Iowa, approximately 32 km from WRF. KDMX uses Doppler WSR-88D radar to collect high resolution data for each full 360 degree scan every 5-min with a range of 230km and a spatial resolution of about 1 km by 1 km. The radar reflectivity data has been collected at 1, 2, 3, and 4 km constant altitude plan position indication height (CAPPI). As shown in Figure 2.3, reflectivity may be quite different at different heights at the same scanning time. Terrain and flocks of birds may result in errors of radar readings. In addition, reflectivity at one height may not be able to fully describe the storm because it occurring at different heights. To deal with these issues, it is necessary to use radar reflectivity data from different CAPPIs.

The radar reflectivity data at nine grid points surrounding each tipping bucket is selected and averaged with the center data to be the reflectivity for that tipping bucket. Null values are treated as missing values and are filled by the reflectivity at the surrounding gird points. The NEXRAD radar data was collected at 5-min intervals. It has been processed to 15-min by averaging 3 radar data reflectivity values.

Table 2.1 summarizes the dataset used in this research. In addition to 4 historical influent flow inputs at 15, 30, 45 and 60 min ahead, 6 rainfall and 24 radar reflectivity inputs provide the temporal and spatial features into the model. The data was collected from January 1, 2007 through March 31, 2008. The data from January 1, 2007 through November 1, 2007 containing 32,697 data points is used for train neural networks. The remaining 11,071 data points and is used to test the performance of the built models.

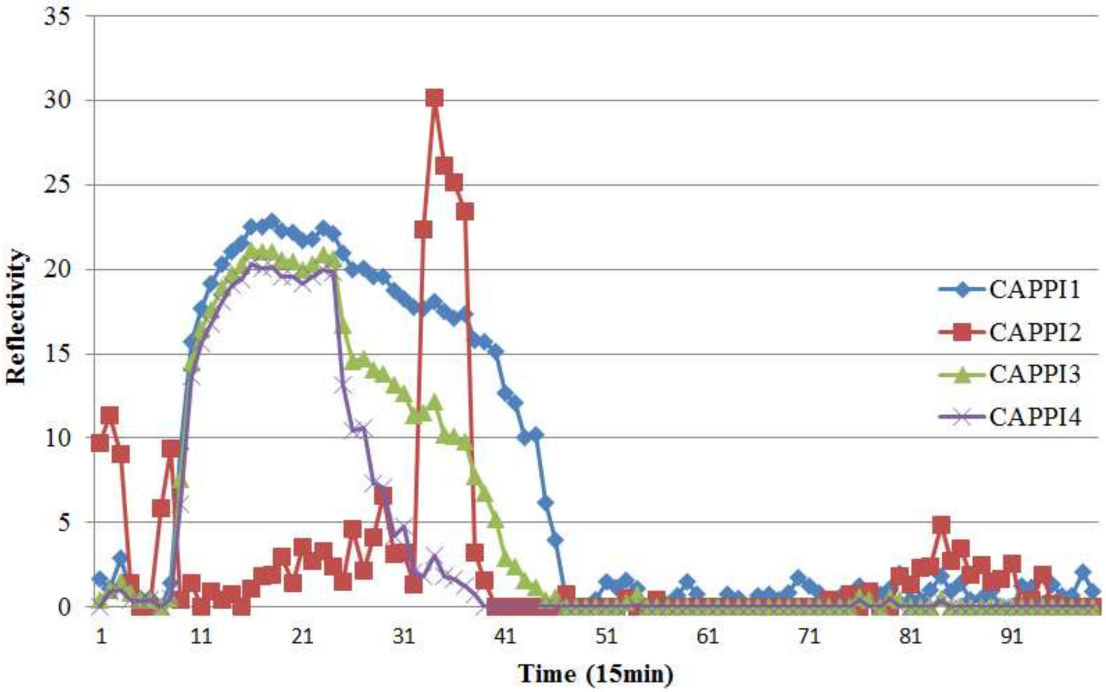


Figure 2.3. Radar reflectivity at different CAPPI

Table 2.1. The data set description

Inputs	Description	Unit
x_1-x_6	Rainfall at 6 tipping buckets	inch
x_7-x_{30}	Radar reflectivity at 6 tipping buckets at 4 CAPPI	number
x_{31}-x_{34}	Historical influent flow	MGD

2.2.2 Prediction accuracy metrics

Three commonly used metrics, the mean absolute error (MAE), mean squared error (MSE), and correlation coefficient R^2 are used to evaluate the performance of the prediction models (Eq. (2.1)-(2.3)).

$$MAE = \frac{1}{n}\sum_{i=1}^{n} | f_i - y_i | \tag{2.1}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n} | f_i - y_i |^2 \tag{2.2}$$

$$R^2 = 1 - \frac{\sum_i (f_i - y_i)^2}{\sum_i (f_i - y_i)^2 + \sum_i (f_i - \overline{y}_i)^2} \tag{2.3}$$

where f_i is the predicted value produced by the model, y_i is the observed value, $\overline{y}_i$ is the mean of the observed value, and n represents the number of test data points.

2.3 Modeling by static multi-layer perceptron neural network

To build the influent flow prediction model, a static multi-layer perceptron (MLP) neural network was developed. The MLP neural network is one of the most widely used network topologies after its introduction 1960 [23]. It has overcome limitations of the single-layer perceptron to handle model nonlinearity. Prediction and classification applications of MLP neural networks have been reported in science and engineering [24-28].

The structure of the MLP neural network used in the research reported in this research is shown in Figure 2.4. It is a supervised back-propagation network with three layers. Each layer has one or more neurons which are interconnected to each neuron of the previous and the next layers. The connection between two neurons is parameterized by a weight and a bias. Different activation functions, such as logistic, hyperbolic tangent, identity, sine and exponential, were selected for the hidden and output layers.

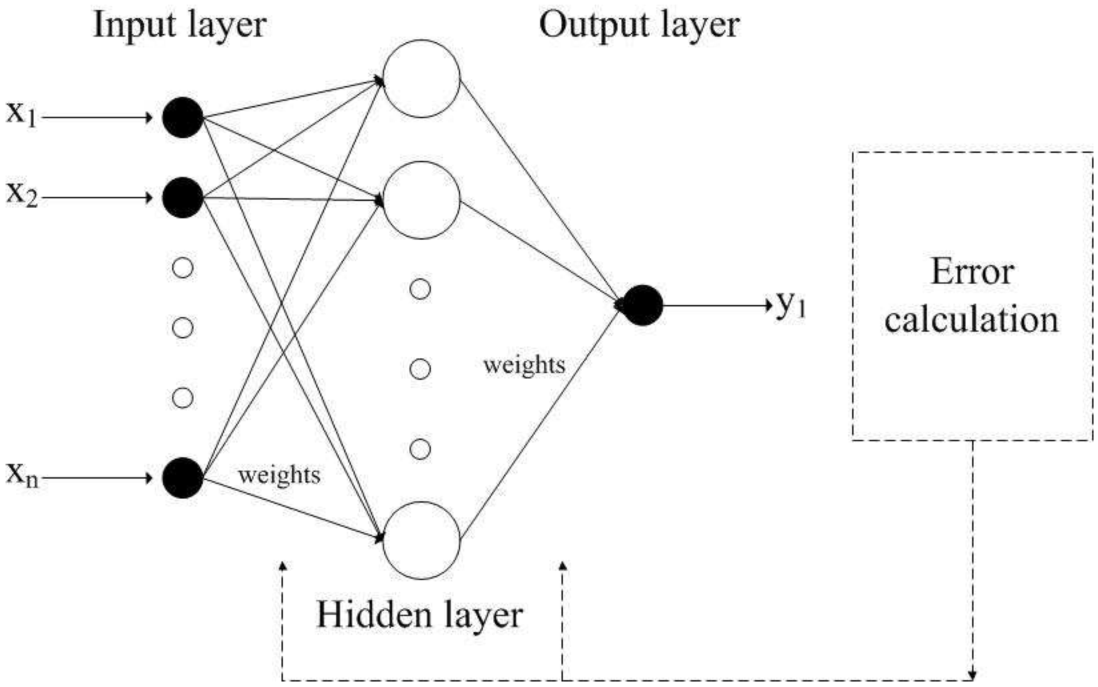


Figure 2.4. Structure of the MLP neural network

In the MLP in Figure 2.4, the output y_1 is calculated as shown in Eq. (2.4)

$$y_1 = f_o(\sum_j f_h(\sum_i x_i w_{ij} + b_j) w_{j1} + b_1) \tag{2.4}$$

where i denotes the ith neuron in the input layer, j is the jth neuron in the hidden layer, f_o and f_h are the activation function for output layer and hidden layer, respectively. w_{ij} is the weight connecting the ith neuron to the jth neuron, and w_{j1} is the weight between the jth neuron in the hidden layer to the neuron in the output layer. b_j and b_1 are the bias for neuron j and output neuron.

The weight is calculated from Eq. (2.5) during the training process so as to minimize the target output

$$\varepsilon(n) = \frac{1}{2}\sum_k (T(n) - y_1(n))^2 \tag{2.5}$$

where ε is the mean of square error, n denotes the nth data point, k is the kth output neuron (k equals to one in this work), T represents the targeted output value.

In total 200 MLP neural networks were trained to get a generalized net structure. The number of neurons in hidden layer varied from 3 to 30. To improve the convergence speed of the training process, BFGS (Broyden-Fletcher-Goldfarb-Shanno) algorithm [29] was used. The weights were randomly initialized between -1 and 1 and iteratively improved by minimizing the mean of square error with iterations. The algorithm would stop when the error was smaller than the set threshold or the number of maximum number of iterations was reached.

The influent flow prediction model at current time t was firstly built. The dataset described in Section 2.2 was used to train and test the MLP neural networks. The best MLP had 25 neurons in the hidden layer with the logistic hidden activation function and the exponential output activation function. The calculated MAE, MSE and correlation coefficient were 1.09 MGD, 4.21 MGD^2, and 0.988, respectively. These metrics indicate that the prediction model is accurate. The first 300 observed and predicted influent flow values from the test dataset are shown in Figure 2.5. Most predicted values are very close to the observed ones, and the predicted influent flow follows the trend of the observed flow rate.

MLP neural networks models were also built at t + 15 min, t + 30 min, t + 60 min, t + 90 min, t + 120 min, t + 150 min, and t + 180 min. As shown in Figure 2.6, the predicted influent flow is close to the observed value, and the predicted trend is the same as the observed one. However, a small time lag between the predicted and observed influent flow appears. The lag increases fast and can be clearly observed in Figure 2.7 which predicts the influent flow time t + 180 min ahead.

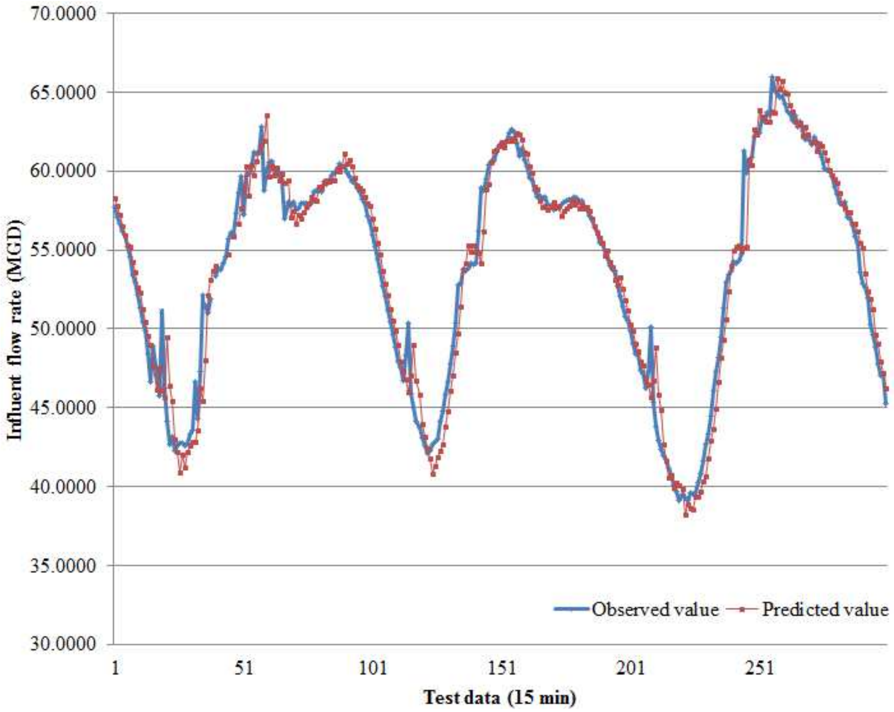


Figure 2.5. Predicted and actual influent flow at current time t

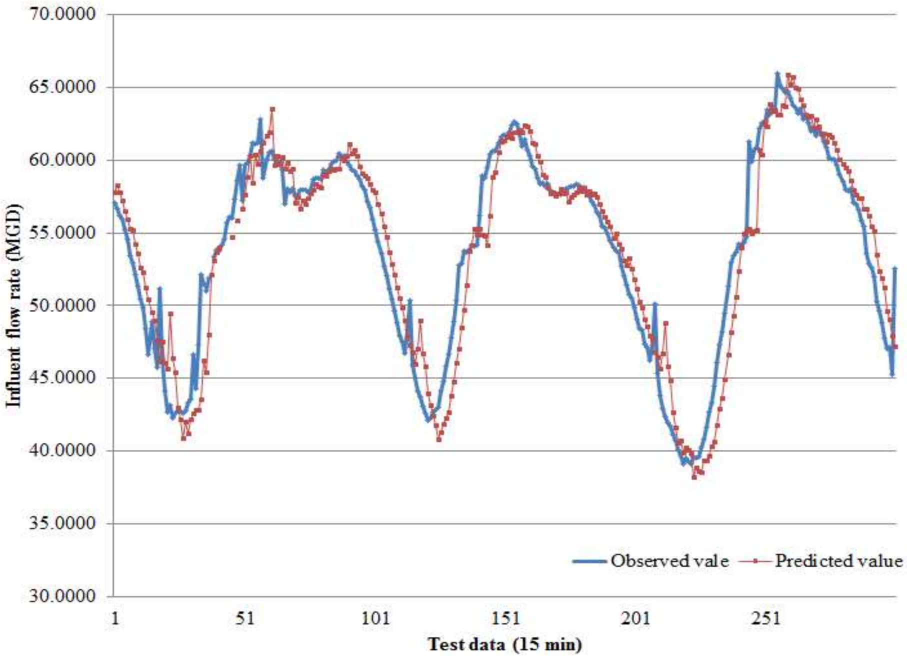


Figure 2.6. Predicted and actual influent flow at time t + 30 min.

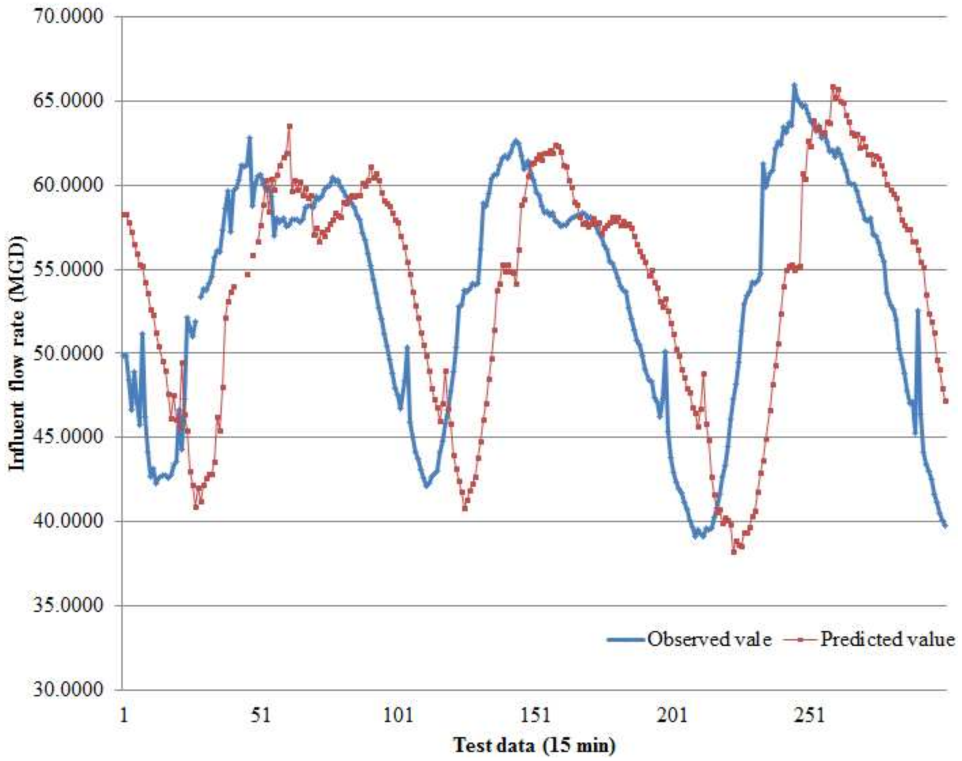


Figure 2.7. Predicted and actual influent flow at time t + 180 min

Table 2.2. Prediction accuracy

Prediction horizon	MAE (%)	MSE(%)	Correlation coefficient
t	1.09	4.21	0.988
t + 15	1.48	5.83	0.983
t + 30	1.89	8.20	0.976
t + 60	2.75	14.59	0.958
t + 90	3.61	22.95	0.934
t + 120	4.46	33.21	0.905
t + 150	5.26	44.88	0.872
t + 180	6.02	57.39	0.836

Table 2.2 summarizes the accuracy results for predictions at the current time t through t + 180 min. The prediction accuracy decreases with increase of the time horizon. The MAE and MSE increase fast after t + 30 min, with the correlation decreasing as well.

The prediction models for horizons smaller than t + 150 min have acceptable accuracy if the threshold of correlation coefficient is set as 85%. Even the trend can be well predicted, the time lag is too large to provide useful real-time influent flow information.

2.4 Modeling by improved dynamic neural network

The computational results in Section 3 indicate that the static MLP neural network is not able to capture the dynamics in the dataset at long time horizons. To deal with this issue and improve prediction accuracy, a dynamic neural network with online corrector was proposed and tested. Successful applications of the dynamic neural network have been reported in literature [30-32]. A dynamic neural network involves a memory and a predictor. As the memory captures the past time series information, it can be used to learn the temporal patterns of the time series by the predictor. This research used focused time-delay neural network (FTDNN) as the predictor [33]. The basic network is MLP network as it can well handle the spatial data. The dynamics appears at the input layer of the network to process the temporal information.

To address the time lag issue caused by the static MLP neural network, an online corrector is proposed. The structure of the final dynamic neural network is shown in Figure 2.8. The details of FTDNN are covered in the literature, e.g., [34]. The inputs of the prediction model included four past values of influent flow (as memory values), radar reflectivity, rainfall, and the online corrector $e(t)$ (Eq. 2.6) at current time t.

$$e(t) = | y_p(t) - y_o(t) | \tag{2.6}$$

where $y_p(t)$ and $y_o(t)$ are the predicted and actual influent flow at current time t. In fact, the online corrector provides the time lag information back to the input layer to calibrate the prediction results during the training iterations.

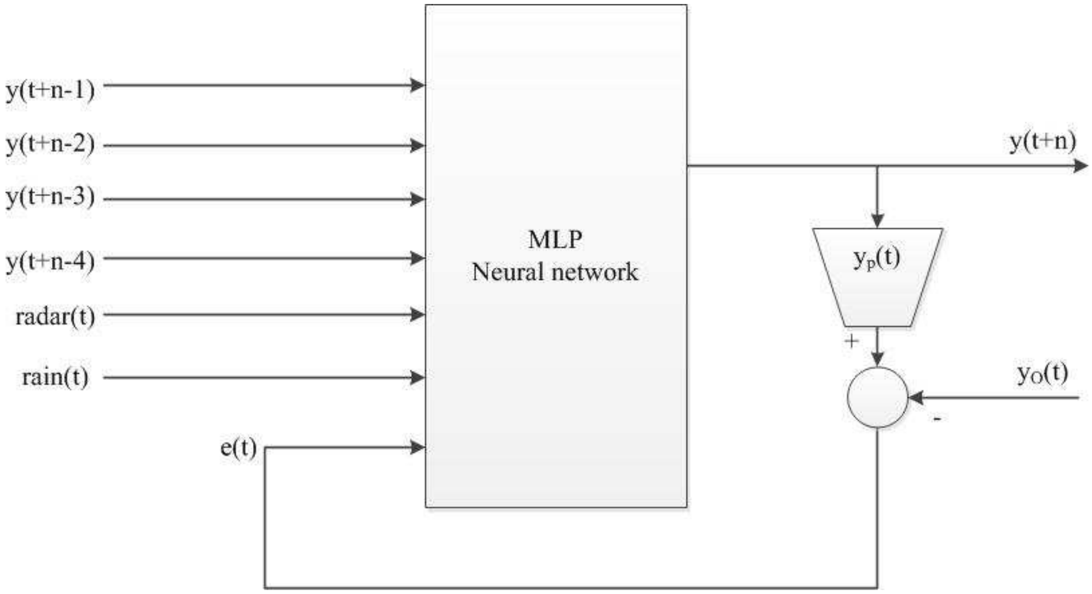


Figure 2.8. Structure of the dynamic neural network

The approach presented in in Section 2.3 was applied to train the dynamic neural network. As shown in Figure 2.9, the influent flow is well predicted at time t +30. There is a slight time lag. Figure 2.10 shows the predicted influent flow and the observed values at time t + 180 min for the dynamic and the static networks. It clearly shows that the time lag of the predictions by dynamic neural network is much smaller than the time lag of the prediction by static MLP neural network. MAE, MSE and correlation coefficient of two neural networks are illustrated in Figure 2.11, 2.12 and 2.13. The built prediction model by dynamic neural network outperforms the model by static MLP neural network. Its MAE and MSE increase slowly with longer time horizons. The correlation coefficient decreases slowly and is still acceptable at time t + 300 min ($R^2 > 0.85$).

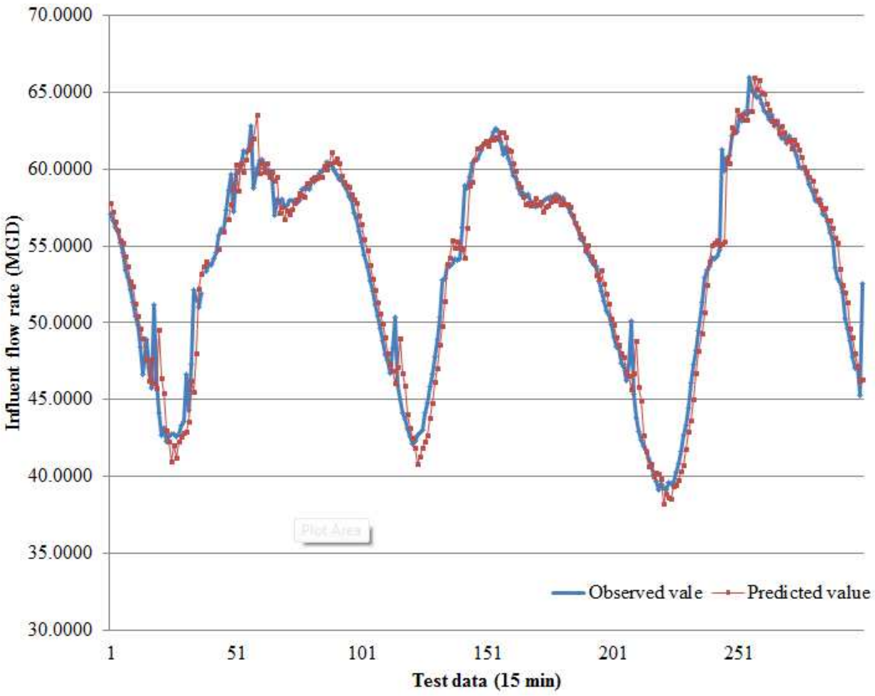


Figure 2.9. Predicted and actual influent flow at time t + 30 min

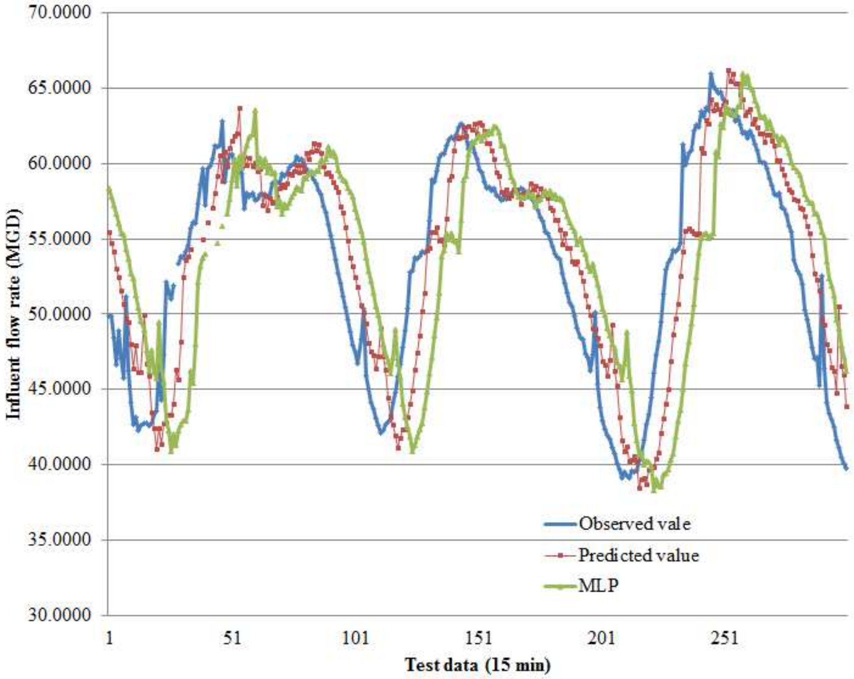


Figure 2.10. Predicted and actual influent flow at time t + 180 min for two models

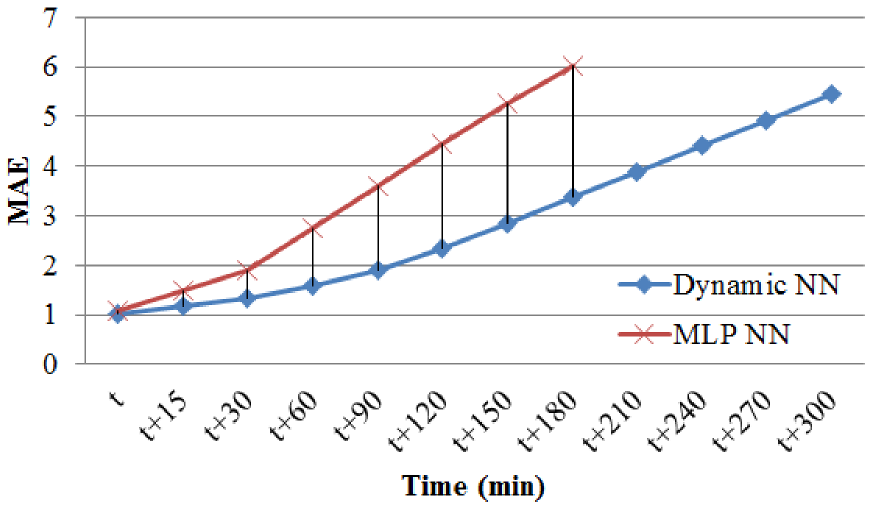


Figure 2.11. MAE of the prediction models by two neural networks

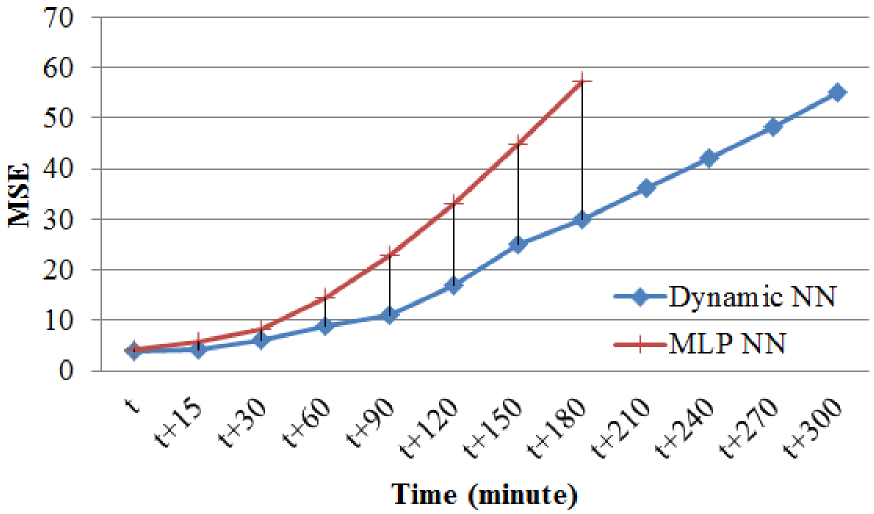


Figure 2.12. MSE of the prediction models by two neural networks

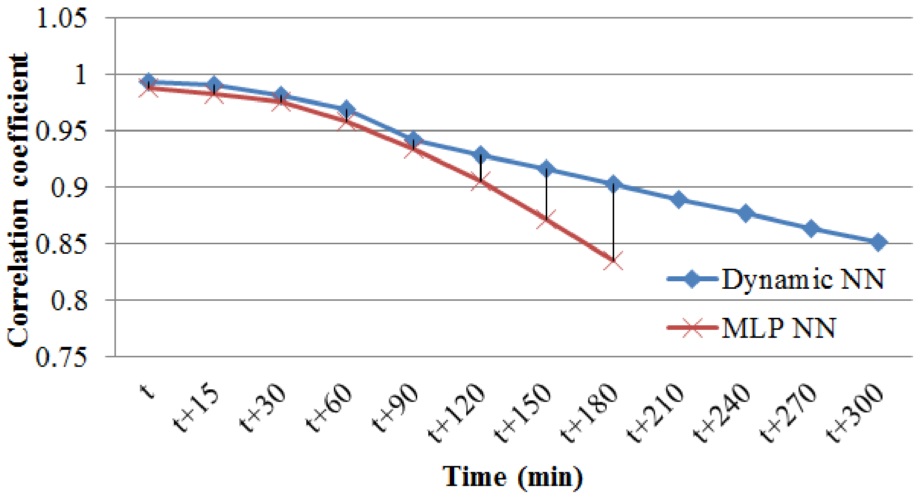


Figure 2.13. Correlation coefficient of the prediction models by two neural networks

The results indicate that dynamic neural network is capable of modeling the influent flow. Static MLP neural network is effective in handling complex non-linear relationships rather than temporal time series. On the other hand, dynamic neural network is suitable for temporal data processing. The online corrector provides additional time series information as an input to correct the time lag generated in the model. The accuracy gain comes at a cost of additional computation time needed to construct the dynamic neural network.

As knowing the future values of influent flow is important for management of WWTPs, the 300 min ahead predictions provided by a dynamic neural network offer ample time to schedule the pumping system and adjust the treatment process parameters. However, the 150 min ahead predictions offered by the static MLP neural network are acceptable in lower precipitation seasons (for example, spring and winter) by saving computation time.

CHAPTER 3

PREDICTING OF THE TOTOAL SUSPENDED SOLIDS IN WASTEWATER

3.1 Introduction

Total suspended solids (TSS) are considered to be one of the major pollutants that contributes to the deterioration of water quality, contributing to higher costs for water treatment, decreases in fish resources, and the general aesthetics of the water [35]. The activities associated with wastewater treatment include control of water quality, protection of the shoreline, and identification of economic life of protective structures. Predicting suspended sediments is important in controlling the quality of waste water. TSS is an important parameter, because excess TSS depletes the dissolved oxygen (DO) in the effluent water. Thus, it is imperative to know the values of influent TSS at future time horizons in order to maintain the desired characteristics of the effluent.

Industrial facilities usually measure the water quality parameters of their influents two or three times a week, and the measurements include CBOD, pH, and TSS [36, 37]. Thus, the infrequently recorded data must be modified to make it suitable for time-series analysis. Sufficient associated parameters must be available to develop accurate TSS prediction models. Wastewater treatment involves complex physical, chemical, and biological processes that cannot be accurately represented in paramedic models. Understanding the relationships among the parameters of the wastewater treatment process can be accomplished by mining the historical data. A detailed description of various waste water treatment plant (WWTP) modeling approaches is described in [38]. Their review work is mainly focused on application of white-box modeling, and artificial intelligence to capture the behavior of numerous WWTP processes. Poch et al. [39] developed an environmental decision support system (EDSS) to build real world waste water treatment processes. In another research, Rivas et al. [40] utilized mathematical programming approach to identify the WWTP design parameters.

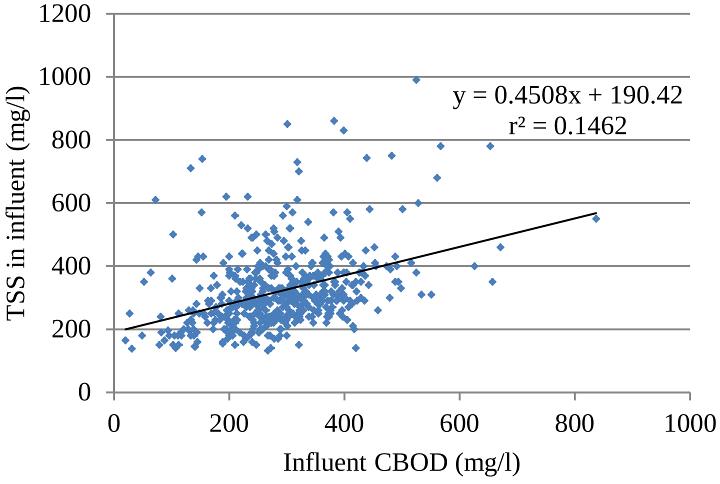
Data-mining algorithms are useful in wastewater research. Examples of data-mining applications reported in the literature include the following: (1) prediction of the inlet and outlet biochemical oxygen demand (BOD) using multi-layered perceptrons (MLPs), and function-linked, neural networks (FNNs); (2) modeling the impact of the biological treatment process with time-delay neural networks (TDNN) [41]; (3) predicting future values of influent flow rate using a k-step predictor [42]; (4) estimation of flow patterns using auto-regressive with exogenous input (ARX) filters; (5) clustering based step-wise process estimation; and (5) rapid performance evaluation of WWTP using artificial neural network.

In the research reported in this chapter, the influent flow rate and the influent CBOD were used as inputs to estimate TSS. Due to the limitations of the industrial data-acquisition system, the TSS values are recorded only two or three times per week. The data must be consistent in order to develop time-series prediction models. Thus, we established two goals for our research goals: (1) to construct TSS time series using influent flow rate and influent CBOD as inputs and (2) to develop models that can predict TSS using the TSS values recorded in the past.

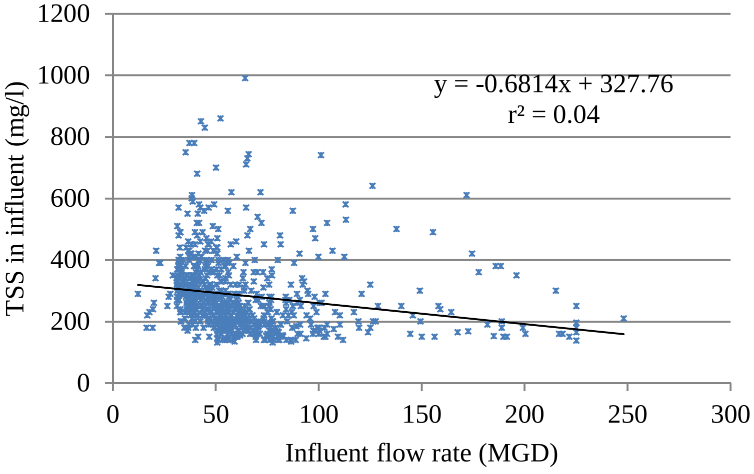
The chapter is organized as follows. Section 3.2 provides details of the dataset used in the research. In Section 3.3, the TSS time-series models are discussed. In Section 3.4, data-mining models are constructed for predicting TSS. The computational results are discussed in Section 3.5.

3.2 Data preparation

The influent flow rate is calculated at 15-min intervals, whereas influent CBOD and TSS are measured only two or three times per week based on the daily concentration values. A five-year data record, collected from 1/1/2005 to 12/31/2010, was available for the research reported in this research. To visualize the relationship between the TSS (output) and the influent flow rate and the influent CBOD as inputs, scatter-point diagrams are presented in Figs. 3.1(a)-(b). The low values of the coefficient of determination (r2) shown in the figures indicate a weak linear correlation between the input and output variables (parameters).



(a)



(b)

Figure 3.1. Relationship between TSS and input parameters: (a) influent CBOD, (b) influent flow rate (daily average values)

Thus, linear regression models are not suitable for predicting TSS using either the influent flow rate or the CBOD as inputs. A non-linear correlation measure, namely, the Spearman correlation coefficient, was computed (Table 3.1). The results provided in Table 3.1 suggest a significant non-linear correlation between the input and output parameters. Based on the non-linear relationship between the influent flow rate and CBOD and TSS, non-parametric approaches were explored.

Table 3.1. Spearman correlation coefficients

	TSS (mg/l)
Influent CBOD (mg/l)	0.5019
Influent flow rate (MGD)	-0.4087

To develop accurate prediction models, data outliers must be removed. Figure 3.2 presents the box plot of TSS values with the outliers identified. In general, the TSS values remain between 32 mg/l and 530 mg/l). However, the outlier data points occur due to errors in the measurements.

A normal, two-sided, outlier-detection approach was used. In two-sided outlier detection, values that exceed +3 σ and values that are smaller than -3 σ are considered to be outliers. Almost 4% of the data points have been determined to be outliers and removed from the analysis. Figure 3.3 provides the box plot of TSS after the outliers are removed.

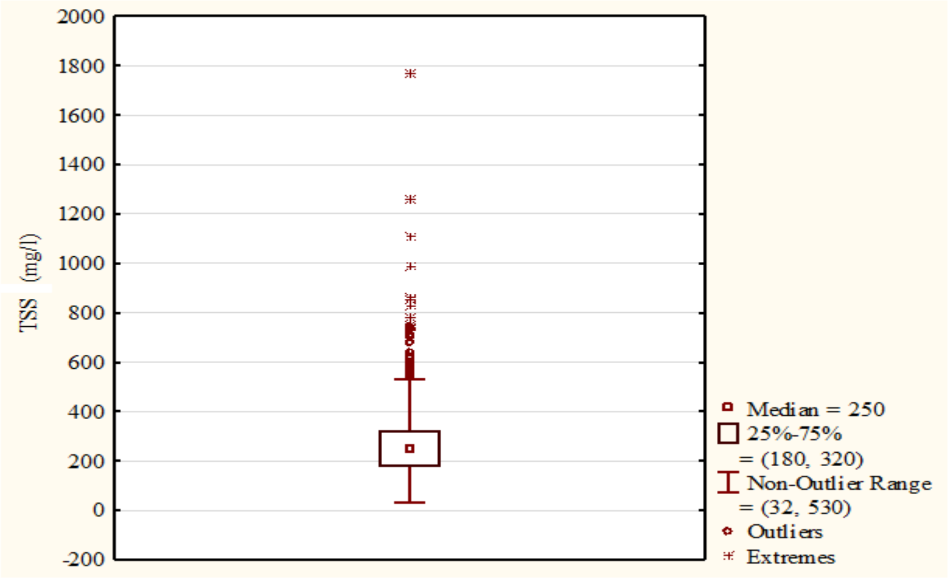


Figure 3.2. Box plot of TSS values

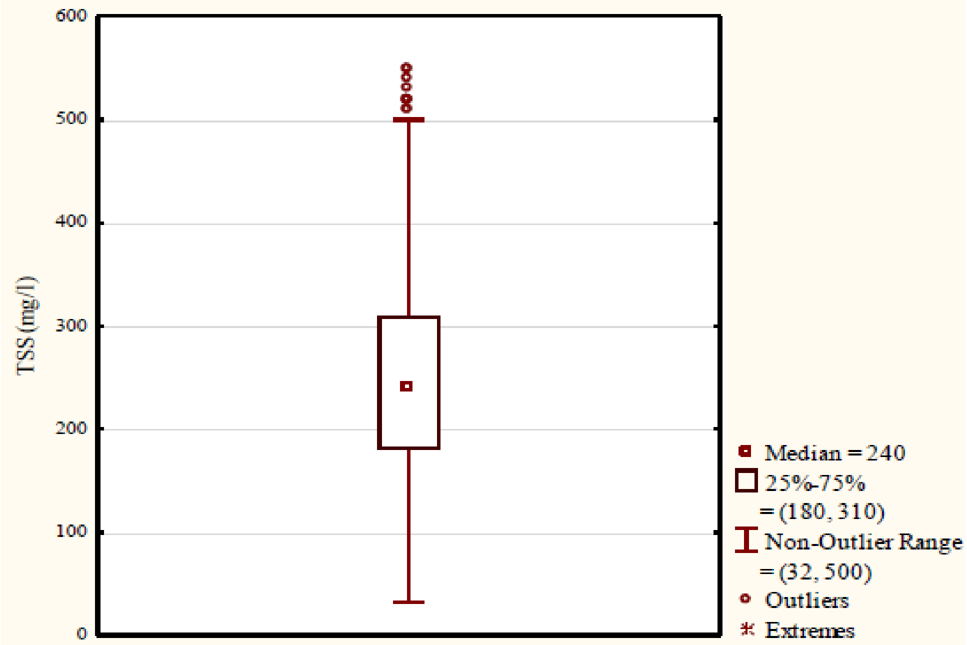


Figure 3.3. Distribution of TSS values after removing outliers

In the next section, methods are discussed for constructing time-series data for TSS.

3.3 Construction of time-series data for TSS

Models that can approximately determine TSS values have been developed using influent flow rate and influent CBOD as input parameters. First, the most relevant parameters are selected to obtain robust models. It is also essential for the reduction of the dimensionality of the data. Approaches for selecting parameters, such as the boosting-tree algorithm, correlation coefficient, and principal component analysis, are often used for this purpose.

The frequency of the measurement of output TSS is once per day, whereas the flow rate of the influent is recorded every 15 minutes. Considering the influent flow rate recorded in a day, the input data-dimension becomes 96. In the first approach for reducing the dimensionality of the data, the boosting-tree parameter selection approach and the correlation coefficient approach were used to identify the best time of day for estimating the values of TSS. The approach uses the total squared error computed at each split of the input parameters. The parameter with the best split is assigned a value of 1, and the less-preferred parameters are assigned values smaller than 1. The boosting-tree algorithm computes the relative influence of the parameters using Eq. (3.1).

$$\tilde{J}_j^2(T) = \sum_{t=1}^{L-1} \tilde{I}_t^2 1(v_t = j) \tag{3.1}$$

where $\tilde{J}_j^2(T)$ is the relative significance of parameter j, i is the index of the tree, vt is the splitting feature associated with node t, L is the number of terminal nodes in the tree, and $\tilde{I}_t^2$ is the improvement of the squared error.

The Spearman correlation coefficient (Eq. (3.2)) reflects the non-linear correlation between the input and output variables [43]. It is a form of the Pearson coefficient with the data converted to rankings.

$$\rho = 1 - \frac{6\sum_{i=1}^{n}(x_i - y_i)^2}{n(n^2-1)} \tag{3.2}$$

where y is the predictor, x is the input variable, and n is the total number of observations. The boosting-tree algorithm ranks the parameter in the range 0 to 1, whereas the correlation coefficients of the parameters can be in the range of -1 to +1. Figure 3.4 provides the ranking of the parameters generated by the boosting-tree algorithm and the Spearman correlation coefficient (absolute value). Both metrics point to the significance of the flow rate of the influent in the time window from 12:00 A.M. to 5:15 A.M.

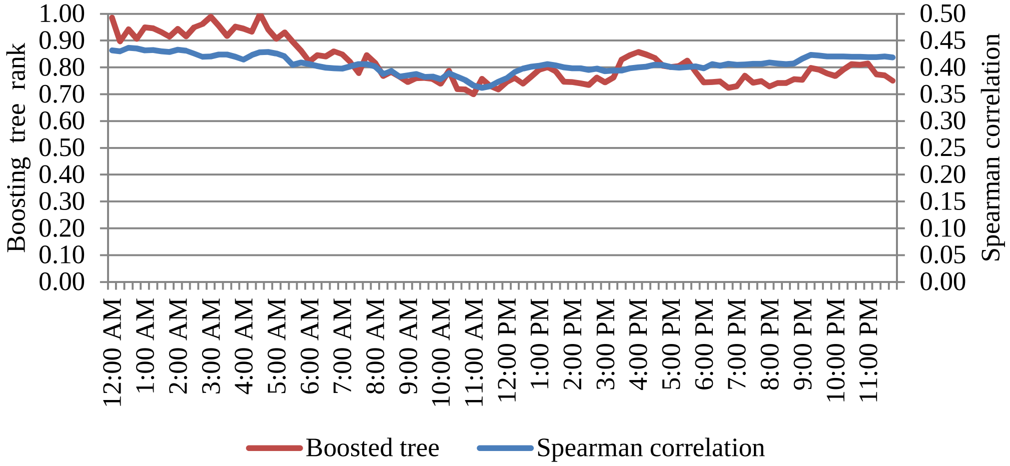


Figure 3.4. Temporal significance of influent flow rate on the TSS in the influent

In the second approach, a principal component analysis (PCA) was used to reduce the dimensionality of the dataset. In PCA, the data undergo an orthogonal, linear transformation to a new coordinate system so that the greatest variance by any projection of the data is realized on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on [44].

Table 3.2 presents the five principal components when applied to the 96 dimensional dataset. With an aim to retain 95% variability of the original dataset, two principal components (i.e., PC1 and PC2 were selected). Influent recorded at 2:00 P.M. - 2:30 P.M., 3:15 P.M., and 5:45 P.M. contributed the most to the first principal component (i.e., PC1).

Table 3.2. Parameters of the principal component analysis (PCA)

Principal Component	Eigenvalue	Variance	Cumulative Variance	Coefficient (Parameter)
PC1*	88.68661	0.92382	0.92382	0.104 (2:15 PM) + 0.104 (2:00 PM) + 0.104 (2:30 PM) + 0.104 (3:15 PM) + 0.104 (5:45 PM)
PC2*	3.23016	0.03365	0.95747	-0.144 (1:30 AM) - 0.144 (1:45 AM) - 0.143 (1:15 AM) + 0.143 (11:00 PM) + 0.143 (10:15 PM)
PC3	1.5456	0.0161	0.97357	-0.188 (11:15 AM) - 0.184 (11:30 AM)-0.183 (11:00 AM)- 0.18 (10:45 AM) - 0.179 (10:30 AM)
PC4	0.655	0.00683	0.9804	0.19 (11:30 PM) + 0.185 (11:45 PM) + 0.183 (11:15 PM) + 0.176 (11:00 PM) + 0.174 (10:45 PM)
PC5	0.374	0.0039	0.9843	-0.426 (7:45 AM) - 0.423 (7:30 AM) -0.327 (8:00 AM) - 0.212 (7:15 AM) - 0.162 (8:15 AM)
* Selected PCs				

Based on the number of input parameters, data frequency, and parameter selection, five different scenarios were investigated and reported in this research (Table 3.3).

Table 3.3. Models for estimating influent TSS

Scenario Number	Input Parameter (Frequency)	No. of Input Parameters
1	CBOD (daily average)	1
2	Influent flow rate, influent CBOD (daily average)	2
3	Influent flow rate (15 min)	96
4	Influent flow rate (15 min, boosting tree ranking > = 0.9, and absolute correlation > = 0.4)	22
5	Influent flow rate (15 min, PC1, PC2)	2

In this chapter, neural networks (NNs) are employed to model the data scenarios listed in Table 3.4. Due to the complex, non-linear behavior of the data used in modeling, 500 neural networks were trained by varying the number of hidden units and activation functions. The number of hidden layers was 1, whereas the number of neurons in a hidden layer varied from 5 to 25. Five different activation functions, i.e., ‘logistic,’ ‘tanh,’ ‘sigmoid,’ ‘exponential,’ and ‘identity,’ were used. For each of the five scenarios mentioned in Table 4, two-thirds of the data were used to derive the model, whereas the remaining one-third of the data was used for testing. Table 3.4 summarizes the testing results obtained for the five scenarios.

While most of the data models discussed in this research have rather high error rates, the results obtained in Scenario 4 are promising. The reported results indicate the significance of high-frequency data and the appropriate selection of parameters in improving the accuracy of the predictions. Based on the results presented in Table 3.4, Scenario 4 was used to construct the time series for TSS. Figure 3.5 compares the actual and predicted values of TSS using the MLP model of Scenario 4. The results in Figure 3.5 indicate a high coefficient of determination (r2 = 0.803).

Table 3.4. Models for estimating the TSS in the influent

Function Approximator	MLP Structure	Hidden Activation	Output Activation	MAE	MRE (%)
Scenario 1	MLP 1-5-1	Tanh	Identity	69.29	24.08%
Scenario 2	MLP 2-25-1	Tanh	Identity	64.47	21.49%
Scenario 3	MLP 96-15-1	Identity	Exponential	64.69	33.10%
Scenario **4**	**MLP 22-16-1**	**Tanh**	**Identity**	**28.11**	**13.34%**
Scenario 5	MLP 2-24-1	Tanh	Tanh	60.88	31.38%

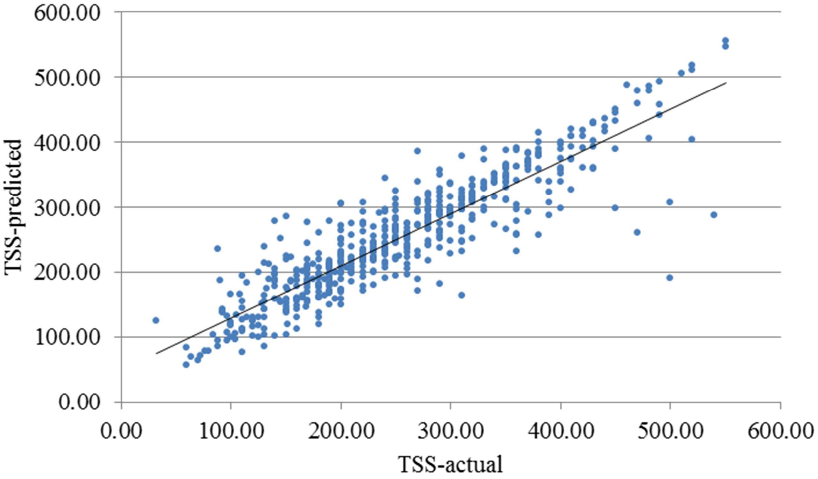


Figure 3.5. Comparison of the actual and the predicted values of TSS (Scenario 4)

The model in Scenario 4 predicted the values of TSS with 86.66% accuracy. These values are used to fill almost 60% of the data needed to construct a five-year TSS time series for the period from January 2005 through December 2010. Figure 3.6 presents the run chart of the actual and predicted values of TSS values over a period of five years. The TSS data displayed in Figure 6 were used to build the time-series prediction model discussed in the next section.

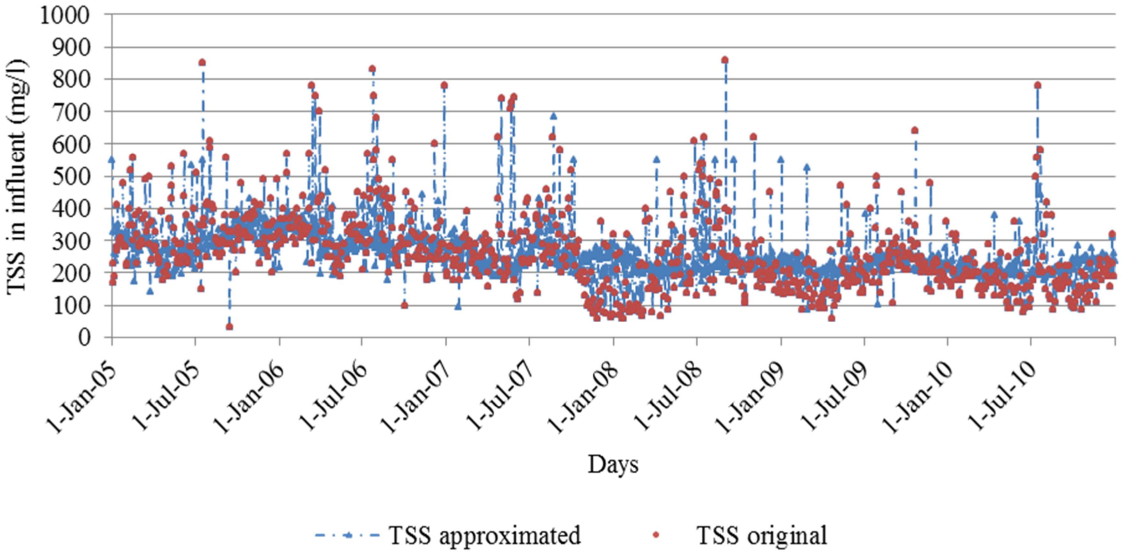


Figure 3.6. Predicted five-year time series for TSS in influent (data from January 1, 2005 through December 31, 2010)

3.4 Predicting the TSS

Considering the univariate nature of the data, the past recorded values of TSS were used as the input to predict the current and future values of TSS. Such past values of the parameters are known as the memory values of the parameters. Memory values have been used extensively to improve the accuracy of the predictions of various models developed for different applications [45, 46]. The values of TSS over the past 10 days were used as input parameters in the expression shown in (3.5):

$$\hat{TSS}(t) = f\begin{pmatrix} TSS(t-T),\ TSS(t-2T),\ TSS(t-3T),\ TSS(t-4T),\ TSS(t-5T) \\ TSS(t-6T),\ TSS(t-7T),\ TSS(t-8T),\ TSS(t-9T),\ TSS(t-10T) \end{pmatrix} \qquad (3.5)$$

The autocorrelation and the boosting-tree algorithm were used to rank the 10 memory parameters. The coefficients produced by the two approaches reflect a similar ranking of the input parameters (Figure 3.7). As anticipated, the immediate past value is the best predictor, but the values recorded a week in the past are more significant than the values recorded two or three days in the past. The ranking of parameters is expressed in Eq. (3.6).

$$\begin{gathered} \zeta[TSS(t-T)] > \zeta[TSS(t-7T)] > \zeta[TSS(t-6T)] > \zeta[TSS(t-8T)] > \zeta[TSS(t-2T)] > \\ \zeta[TSS(t-9T)] > \zeta[TSS(t-5T)] > \zeta[TSS(t-3T)] > \zeta[TSS(t-10T)] > \zeta[TSS(t-4T)] \end{gathered} \qquad (3.6)$$

where $\varsigma[.]$ is the significance of the parameter.

The five best predictors from Eq. (3.6) were selected to develop the model for predicting day-ahead values of TSS. Descriptions of the selected data-mining algorithms for model construction are provided in the next section.

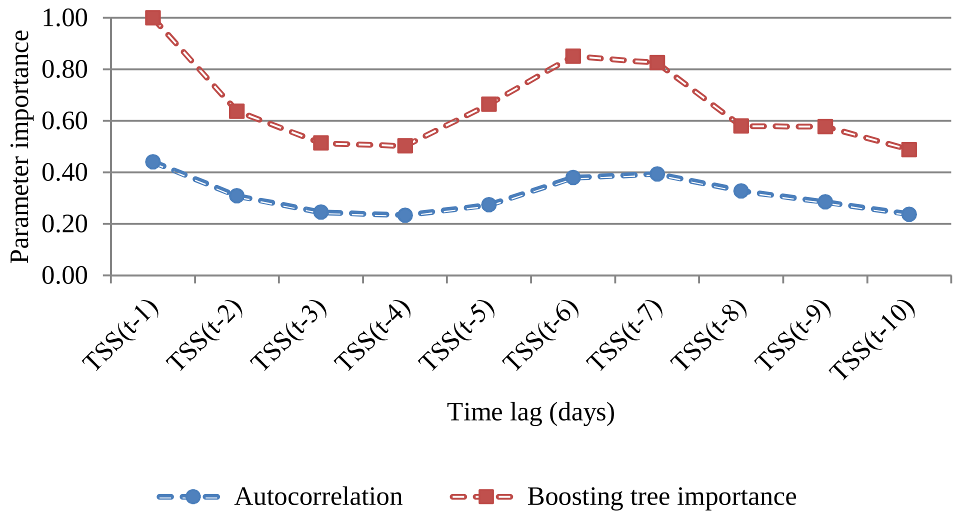


Figure 3.7. Ranking of memory parameters used to predict future values of TSS

3.4.1. Algorithm selection

Five data-mining algorithms, i.e., the k-nearest neighbors (k-NN) ; multi-variate adaptive regression spline (MARS); neural network (NN); support vector machine (SVM) ; and random forest (RF) algorithms, were considered to predict future values of TSS. A back-propagation algorithm determines the best fit NN. SVM constructs a set of hyper planes in high-dimensional space, which can be used for classification and regression. RF is an ensemble learning method in which multiple trees are generated. It selects n input parameters randomly to split the tree nodes. MARS is a non-parametric procedure for regression analysis. It constructs the functional relationship between input and output variables from a set of coefficients and basis functions, all driven by regression data. The k-NN approach is an instance-based learning method in which the function is approximated locally. For regression models, k-NN output is the average of the k-nearest neighbors' outcomes.

An algorithm predicting day-ahead values of TSS with minimum error was selected to construct models for seven-day-ahead predictions. NN was trained with 100 multi-layered perceptron (MLPs) by varying hidden, output activation functions and the number of neurons in the hidden layers. Activation functions, e.g., 'logistic,' 'tanh,' 'sigmoid,' 'exponential,' and 'identity' were considered for both hidden and output nodes. A single hidden-layer was used in this network, while the number of neurons varied from 5 to 25. SVM was trained using four different kernels, i.e., RBF, polynomial, linear, and sigmoid kernels. The number of nearest neighbors in the k-NN algorithm was varied from 2 to 10 in training, while the Euclidean distance was used as a distance metric. MARS was trained on a number of basis functions, with the maximum equal to 500. RF was trained by setting the number of random predictors to three, while the maximum number of trees was 500. Table 3.5 presents the 10-fold, cross-validation result obtained using five data-mining algorithms.

Table 3.5. Day-ahead prediction of TSS in influent with data-mining algorithms

Algorithm	MAE	MRE (%)
k-NN (k = 10)	62.15	26.46%
RF	52.19	21.66%
NN	**38.88**	**16.15%**
MARS	44.59	18.29%
SVM	61.36	26.10%

Based on the results in Table 3.5, the NN algorithm (MLP 5-24-1, hidden activation: tanh, output activation: exponential) outperforms the other algorithms by providing the lowest MAE and MRE errors. Figure 3.8 illustrates the run chart of the actual and MLP-predicted TSS values. The results in Figure 3.8 show that the MLP algorithm is the most accurate predictor of future values of TSS.

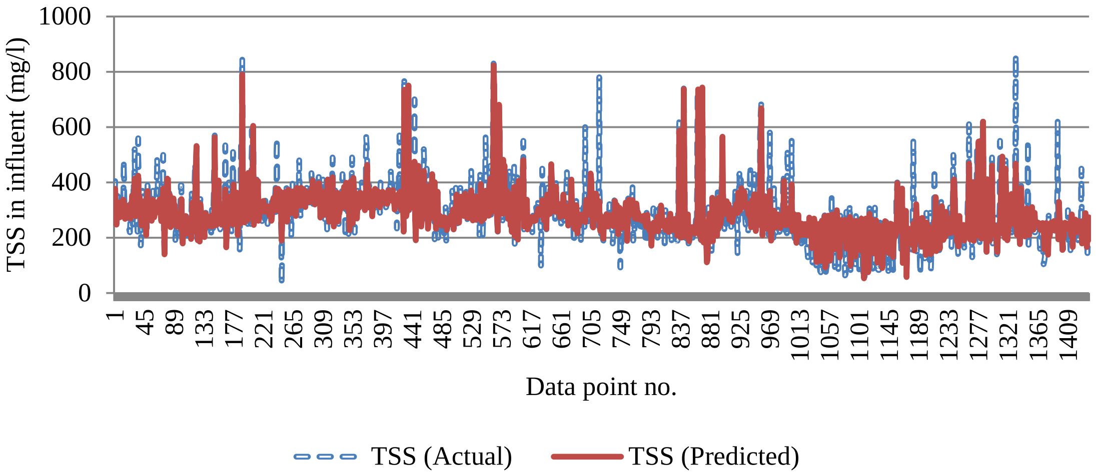


Figure 3.8. Comparison of the actual and MLP model-predicted values of TSS

3.4.2 Iterative learning

Even though the results produced by MLPs (Table 3.6) were promising, the prediction error can be reduced further by updating the prediction model iteratively for the next time-step prediction. A sliding window was utilized with NN models to predict future values of TSS iteratively. The value of TSS predicted by NN model at the current time (TSS(t)) was used as the input to predict the values of TSS at some future time (TSS(t + 1)). The least significant parameter was replaced with the predicted output to keep the dimensions of the input data constant. Figure 3.9 illustrates the concept of iterative learning. After each iteration, the least-significant memory parameter was replaced with the parameter predicted in the previous iteration. Thus, for predicting the values of TSS two days ahead, the one-day ahead predicted value of TSS was used as an input, and this process was repeated until it was ended.

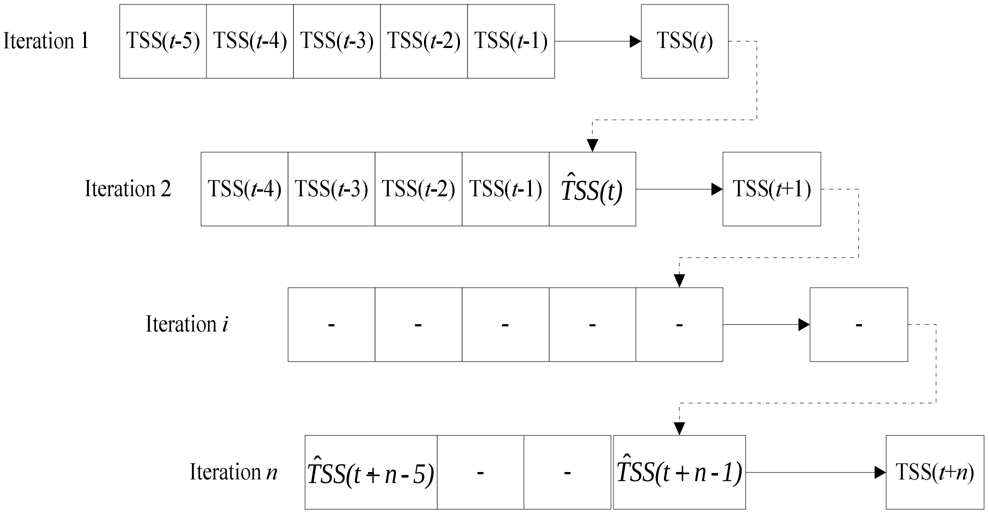


Figure 3.9. Iterative learning procedure

In this research, seven consecutive memory parameters, i.e., {TSS (t-7), TSS (t-6), TSS (t-5), TSS (t-4), TSS (t-3), TSS (t-2), and TSS (t-1)}, were used as inputs to predict the current value {TSS (t)}. Seven MLP models were constructed iteratively from the training data using 10-fold cross validations. Table 3.6 presents the results obtained by the MLP at each learning step.

Table 3.6. MLP learning results

Learning Steps [days]	MAE	MRE (%)	MLP Structure	Hidden Activation	Output Activation
1	44.04	18.54	MLP 5-12-1	Tanh	Exponential
2	46.15	19.19	MLP 5-21-1	Identity	Exponential
3	46.80	19.60	MLP 5-3-1	Logistic	Identity
4	47.05	23.14	MLP 5-25-1	Exponential	Identity
5	49.99	23.82	MLP 5-25-1	Exponential	Exponential
6	51.22	25.74	MLP 5-13-1	Tanh	Tanh
7	50.76	26.58	MLP 5-2-51	Identity	Exponential

In the next section, the best data-mining models are used to predict the future values of TSS. The prediction results obtained using basic and iterative learning are compared.

3.5 Computational results

The values of TSS were predicted up to seven days ahead with the MLP models developed in Section 4 (Table 3.6). Table 8 presents the results obtained using MLP at seven time steps, spaced at one-day intervals. MAE was found to be in the range of 41-55 mg/l, whereas the MRE ranges from 22% - 32% for seven-day prediction. The results in Table 3.7 indicate that the week-ahead values of TSS can be predicted with almost 68% accuracy.

Table 3.7. TSS prediction results with NN (MLP 5-24-1, hidden activation function: Tanh, output activation: exponential algorithm)

Time Steps [days]	MAE	MRE (%)
t +1	41.05	22.02
t +2	44.76	24.18
t +3	48.55	26.32
t +4	50.30	27.01
t +5	49.66	27.20
t +6	53.85	28.49
t +7	55.24	31.34

In this section, the models constructed by seven MLP algorithms (Table 3.6) are applied iteratively to the test data. Table 3.8 provides the MAE and MRE statistics for the test dataset used for prediction. The computational results in Table 3.8 indicate that TSS can be predicted a week ahead with accuracy up to 73%, with the MAE in the range of 40.95 - 52.30 mg/l and the MRE in the range of 21.85% - 27.55%.

Figure 3.10 illustrates the error improvement over time for the dynamic learning scheme. By applying the iterative NN learning scheme, a 5% improvement in the MRE and a 4% improvement in the MAE were obtained. The results shown in Figure 3.10 indicate that the iterative learning scheme can be useful in making long-term predictions.

Table 3.8. Results of the prediction of TSS using MLP algorithms
(dynamic learning scheme)

Time Steps[days]	MAE	MRE (%)
t +1	40.95	21.85
t +2	44.32	24.04
t +3	45.95	24.32
t +4	47.70	25.88
t +5	49.38	27.01
t +6	49.66	27.20
t +7	52.30	27.55

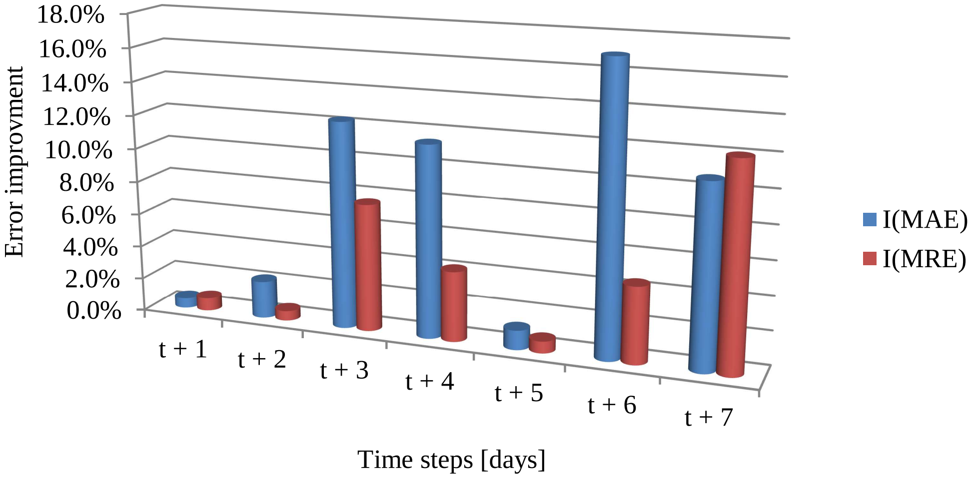


Figure 3.10. Error improvement over different time steps

CHAPTER 4

PREDICTING OF CBOD IN WASTEWATER

4.1 Introduction

Wastewater treatment plants involve several processes for converting raw influent into an efficient effluent [47]. The unsteady flow rate of influent wastewater calls for efficient control solutions. From measurement of the concentration of influent waste, useful information for the control can be obtained. In the literature, biochemical oxygen demand (BOD), chemical oxygen demand (COD), potential of hydrogen (pH), and total suspended solids (TSS) are widely used indicators of wastewater quality [48-51].

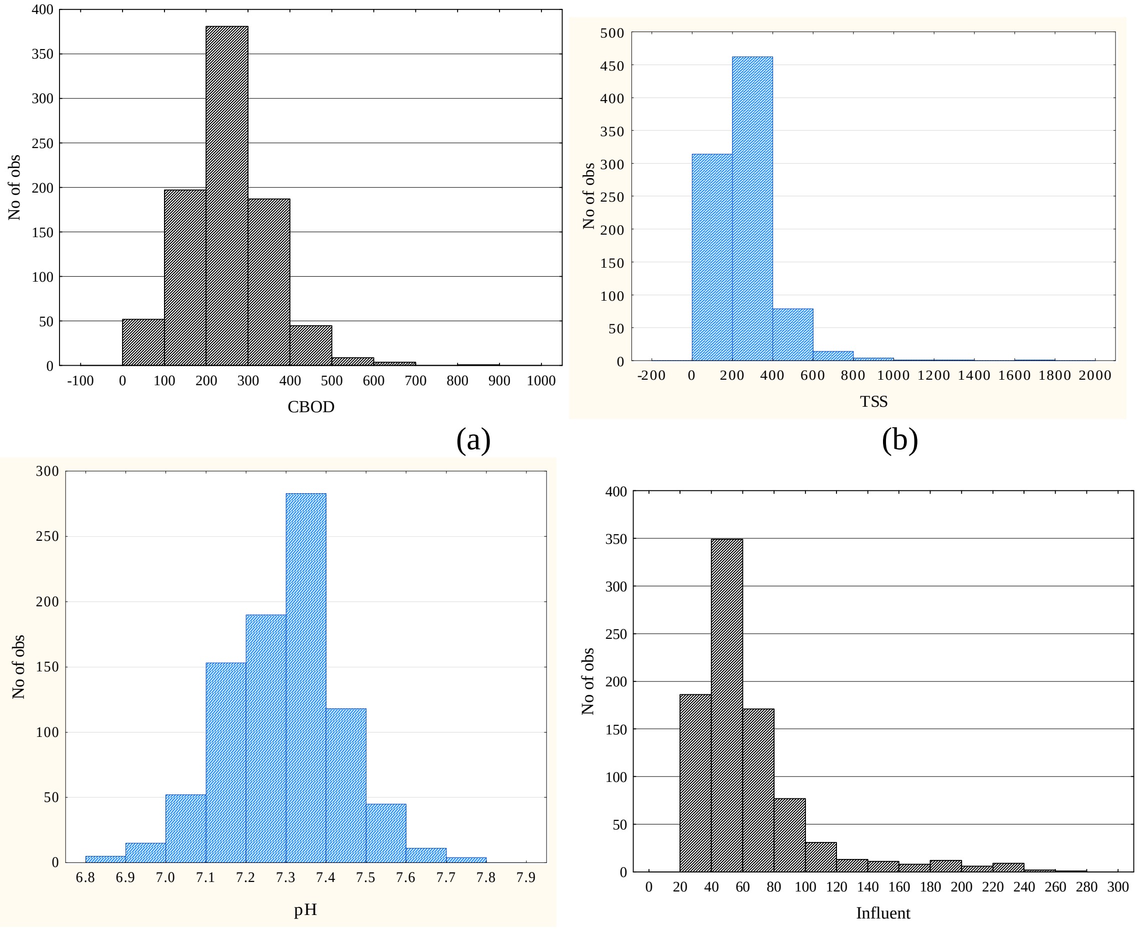
In practice, the influent water quality is not measured with online sensors [52, 53]. CBOD, pH, and TSS are usually measured 2 or 3 times a week. This time span is too long for real-time control purposes [54, 55]. Monitoring the waste concentration has been considered in the literature as a way to address the influent quality issue. Various deterministic models are presented in [56, 57]. Holmberg [58] presented a method to estimate the influent BOD concentration based on a simplified dynamic model. Onnerth et al. [59] proposed a model-based software sensor to identify process relations, and implemented on-line control strategies. Their experimental results have shown a 30% reduction of energy use.

An alternative way to estimate influent quality is by using a data-driven approach. Wastewater treatment plants record the water quality parameters on a regular basis. Using the existing data, the relationship between the waste concentration and the parameters, such as influent flow, which is usually measured continuously, could be identified by data-mining algorithms. Over the past few years, data mining has been successfully deployed in business and industrial [60], engineering [61], and science applications, and has been proven to provide useful results. Related applications of data mining include analysis of the pollution level in a wastewater treatment plant emissary [62], monitoring an acidic chromic wastewater treatment plant using self-organizing maps, and discovering hidden patterns in wastewater treatment data with induction rule techniques [63].

In this chapter, CBOD is used as a metric to represent the quality of the wastewater. Data-mining- and statistics-based approaches are employed to identify the relationship between the influent flow rate and CBOD. Four data-mining algorithms are used to predict the CBOD on daily data.

4.2 Data description and statistical analysis

The influent rate is calculated at 15 min intervals, whereas CBOD, pH, and TSS are calculated 2-3 times a week based on daily concentration values. A five-year record of long data from 1/4/2005 to 12/29/2010 was available for the research reported in this research. Fig. 4.1(a)-(d) presents the histograms of four parameters. The data suggests that the influent rate is concentrated in the range of (20-100), pH in the range of (7.1-7.5), CBOD in the range of (100-400 mg/l), and TSS in the range of (100-400 mg/l). To visualize the relationship between the input influent rate and various outputs, scatter point diagrams are presented in Fig. 4.2(a)-(c). It can be observed that CBOD and TSS decrease exponentially as the influent rate increases, whereas pH does not suggest any direct relationship to the influent rate. The correlation coefficients are provided in Table 4.1. Based on the correlation coefficients, models predicting CBOD are described in the next section.



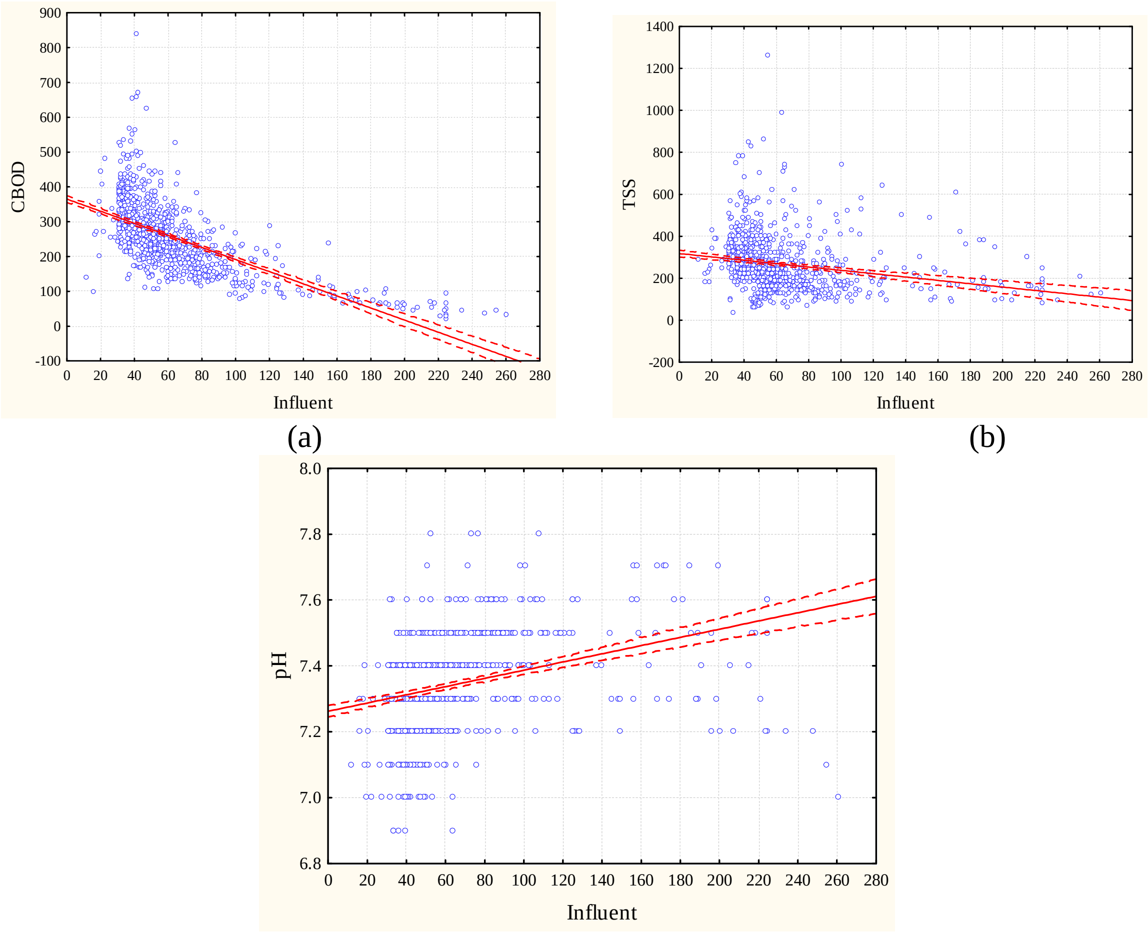
(a) (b)

(c) (d)

Figure 4.1. Histogram of input data (a) CBOD, (b) TSS, (c) pH, and (d) influent flow rate

Table 4.1. Correlation coefficients

	Influent flow rate
CBOD	-0.653
TSS	-0.235
pH	0.32



(a) (b)

(c)

Figure 4.2. Relationship between influent flow rate (input) and output, (a) CBOD, (b) TSS, and (c) pH

4.3 Modeling and solution methodology

This section presents a three-step methodology to predict CBOD values of wastewater (see Fig. 4.3). In Step 1, the time-stamped data is integrated to ensure that both input and output data are of the same frequency. In Step 2, the missing data is addressed by approximating the relationship between influent flow rate and output. In Step 3, data-mining algorithms are employed to construct the models predicting CBOD. Owing to the seasonal effects of the CBOD, models for separate seasons are also analyzed.

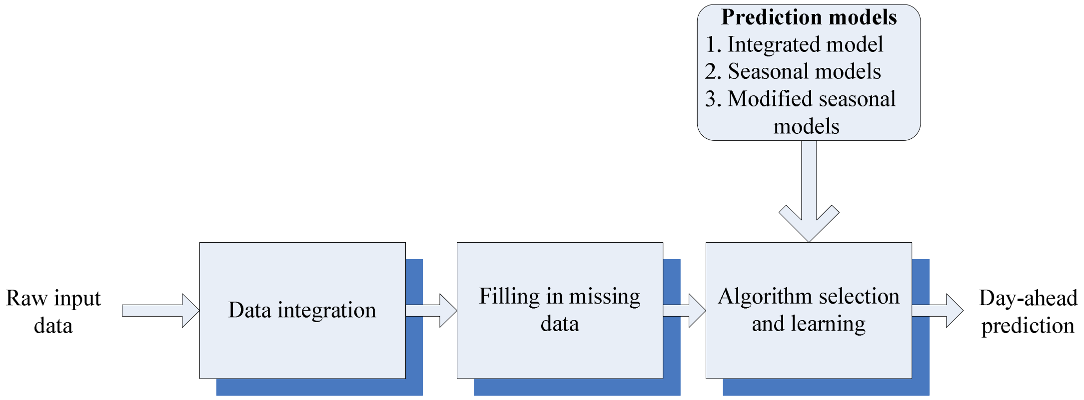


Figure 4.3. The tree-step modeling methodology

Due to the limitations of the measurements, the frequency of the data varied. Here, the influent rate data is more frequent than the CBOD data. Models based on daily average and 15-min influent rate are analyzed. The time-stamp alignment of data ensures the consistency of the frequency of the data.

4.3.1Filling in missing data

Though the frequency of the output data is 24 h, some data has not been recorded on a daily basis. This has created gaps in the dataset, which needed to be addressed before a time-based prediction could be performed. In this research, both univariate daily average influent rate data and multivariate (15 min influent) data is analyzed to fill in the missing CBOD values. The aim is to derive a model that generates CBOD values using influent flow rate as input. For the univariate data, curve fitting [64] and genetic programming is used, whereas, for the multivariate data, genetic programming and neural networks are used.

Curve fitting identifies the best fitting curve or the corresponding equation given a series of data points. The least error is often used to measure this fit. In the research reported in this research, the data plot of the influent rate and CBOD indicates an exponential function (see Fig. 4.1). Therefore, the original data is fitted with an exponential decay function. The ability of genetic programming to obtain the exact mathematical equations from the input data has attracted the attention of many researchers. In this research, a GP is utilized to identify the mathematical relationship between the influent rate and CBOD. The operators, namely ‘multiplication’, ‘subtraction’, ‘divide’, ‘addition’, ‘constant’, ‘exponential’, ‘sine’, ‘cosine’, ‘square root’, ‘logistic’, and ‘gaussian’, constitutes the building blocks of function operators. Neural networks (NNs) are the complex structures of neurons that work together to solve a specific problem. The network structure consists of input layers, hidden layers, an output layer and hidden layers to generate the output. NNs are well suited to model non-linear data. As shown in Fig. 4.2(a), the data is highly non-linear; therefore, 100 NNs are trained to obtain the best NN structure. The number of hidden layers is kept at 1, whereas the number of neurons in a hidden layer varies from 5 to 25. Five different activation functions – ‘logistic’, ‘tanh’, ‘sigmoid’, ‘exponential’, and ‘identity’ – are employed. NNs are applied on three different datasets with different numbers of input parameters.

For both univariate and multivariate data sets, two-thirds of the input preprocessed data is used to derive the model, whereas the remaining one-third of the data is used for testing. The distributions of the training and testing instances are identical. More specifically, the initial data is stratified into 3 folds, whereas the first 2-folds of data are used for training, and the remaining 1-fold is used for testing.

In the research reported in the research influent rates collected at 15-min intervals is used. Therefore, the multivariate dataset has 96 (4×24) inputs. The 15-min data is combined with the CBOD data measured daily. The higher frequency (15-min) data improves prediction accuracy of CBOD. The number of inputs needs to be reduced to improve prediction accuracy. Two different approaches of data dimensionality reduction are used. In the first approach, a correlation coefficient between influent flow rate and CBOD is used to identify the best time of the day for estimating CBOD (Fig.4.4) (Hall

1998). The correlation coefficients follow different patterns through the day. A higher correlation value is obtained in the time period from 4:00 to 9:00 pm. Thus, using the correlation coefficient value, the initial 96 dimensions of data are reduced to 20.

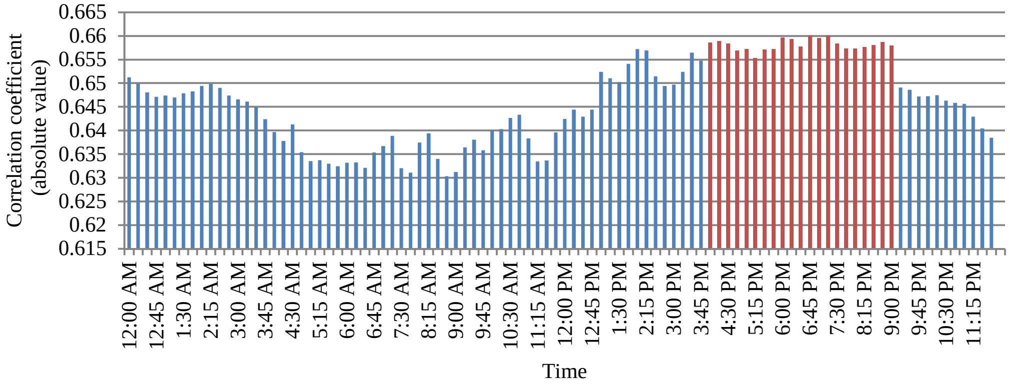


Figure 4.4. Correlation coefficient between influent flow rate and CBOD

In the second approach, two data-mining algorithms – the boosting tree and the Wrapper with genetic search – are used. Boosting tree ranks the parameters based on the sum of the squared error computed at each split of the input parameter [65]. The average statistic is calculated for all splits. The parameter with the best split is assigned a value of 1, and so on. Wrapper is a supervised learning approach that uses genetic search to select the relevant parameters by performing 10-fold cross validation [66]. Table 4.2 provides the 10 best input parameters obtained using the boosting tree and the Wrapper genetic search algorithms. Overall, 18 distinct input parameters are obtained.

Table 4.2. Elected parameters using data-mining algorithms

Boosting tree		Wrapper genetic search	
Parameter	Importance	Parameter	Importance
Influent$_{6:15\ \mathrm{PM}}$	100	Influent$_{12:00\ \mathrm{AM}}$	100
Influent$_{7:45\ \mathrm{PM}}$	99	Influent$_{10:15\ \mathrm{AM}}$	60
Influent$_{6:45\ \mathrm{PM}}$	99	Influent$_{04:00\ \mathrm{PM}}$	60
Influent$_{6:30\ \mathrm{PM}}$	98	**Influent$_{04:15\ \mathrm{PM}}$**	**60**
Influent$_{7:15\ \mathrm{PM}}$	98	Influent$_{10:00\ \mathrm{AM}}$	40
Influent$_{7:00\ \mathrm{PM}}$	97	Influent$_{12:15\ \mathrm{PM}}$	40
Influent$_{7:30\ \mathrm{PM}}$	97	Influent$_{01:30\ \mathrm{PM}}$	40
Influent$_{4:30\ \mathrm{PM}}$	**97**	**Influent$_{04:30\ \mathrm{PM}}$**	**40**
Influent$_{8:00\ \mathrm{PM}}$	97	Influent$_{08:45\ \mathrm{PM}}$	40
Influent$_{4:15\ \mathrm{PM}}$	**96**	Influent$_{12:15\ \mathrm{AM}}$	20

The equations (4.1)-(4.4) represent the approximating functions obtained with curve fitting and genetic programming (GP) algorithms based on the univariate and multivariate influent rate data. The population size of GP is set to 64, whereas the crossover and mutation rate are 0.8 and 0.01, respectively.

$$CBOD_1 = a \times \exp(b \cdot \mathrm{Influent}_{\mathrm{avg}}) \tag{4.1}$$

$$CBOD_2 = 79.1 + \frac{1.04 \times 10^4}{\mathrm{influent}_{\mathrm{avg}} + \cos\left(\left(\mathrm{influent}_{\mathrm{avg}}\right)^2\right) + 1.66\cos\left(\left(\mathrm{influent}_{\mathrm{avg}}\right)^2\right)^2} \tag{4.2}$$

$$CBOD_3 = 2.77 \times \mathrm{influent}_{8:00\ \mathrm{PM}} + \frac{1.45\sqrt{\mathrm{influent}_{5:00\ \mathrm{PM}} - 36.5}}{0.0007 \times \mathrm{influent}_{8:30\ \mathrm{PM}} - 0.0224} + \frac{\cos\left(0.185 \times \mathrm{influent}_{7:15\ \mathrm{PM}}\right)}{0.0028 \times \mathrm{influent}_{7:15\ \mathrm{PM}}} - 2.83 \times \mathrm{influent}_{5:00\ \mathrm{PM}} - 124.42 \tag{4.3}$$

$$CBOD_4 = 9.44 \times \cos\left(-0.177 \times \mathrm{influent}_{8:30\ \mathrm{PM}}\right) - 94.3 + \frac{7.05 \times \mathrm{influent}_{4:30\ \mathrm{PM}}}{0.0074 \times \mathrm{influent}_{8:30\ \mathrm{PM}} - 0.0786} \tag{4.4}$$

Subscripts 1-4 in the CBOD equations (4.1)-(4.4) represent a modeling approach; namely, curve fitting, GP with univariate data, multivariate GP with correlation coefficient, and multivariate GP with a data-mining algorithm, respectively. In equation (4.1), the optimal values of a and b are 492.51 and 0.0113, respectively. The influent_{avg} in equation (4.2) is the daily average influent rate computed for each 15-min interval; whereas, in equations (4.3)-(4.4), the influent flow rate recorded at the specific time of day is indicated by the subscript.

Table 4.3 presents the results produced by various function approximators. The NN-model built according to the input parameters determined by the correlation approach outperformed all other approaches (it has the smallest MRE). The results in Table 4.3 indicate that the models built from the multivariate data yielded smaller error than those from the univariate data.

Table 4.3. Test results produced by different function approximators

Function approximator	Description	No. of input parameters	MAE	MRE (%)	Accuracy (%)
$CBOD_1$	Curve fitting (exponential decay)	01	51.86	29.21	70.78
$CBOD_2$	GP-univariate	01	58.65	30.98	69.00
$CBOD_3$	GP-multivariate (correlation-based)	21	80.32	32.14	67.85
$CBOD_4$	GP-multivariate (data-mining-based)	18	80.24	31.93	68.06
*$CBOD_5$	NN-multivariate (correlation-based)	21	47.32	26.15	73.85
$CBOD_6$	NN-multivariate (data-mining-based)	18	49.44	27.32	72.67
$CBOD_7$	NN-multivariate (all parameters)	96	60.32	30.15	69.85
*Best function approximator					

The analysis indicates that the NN (data-mining-based) and NN (correlation-based) approaches achieved an accuracy of over 70%. The best function approximator, CBOD5, is used to fill in the missed data in the CBOD time series. Fig. 4.5 shows the run chart of the observed and predicted values from the CBOD5 model.

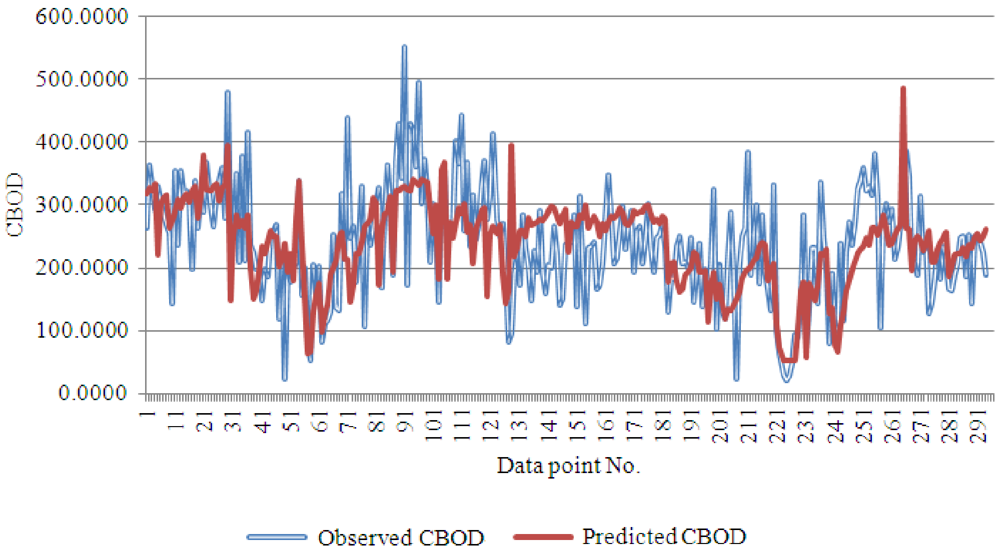
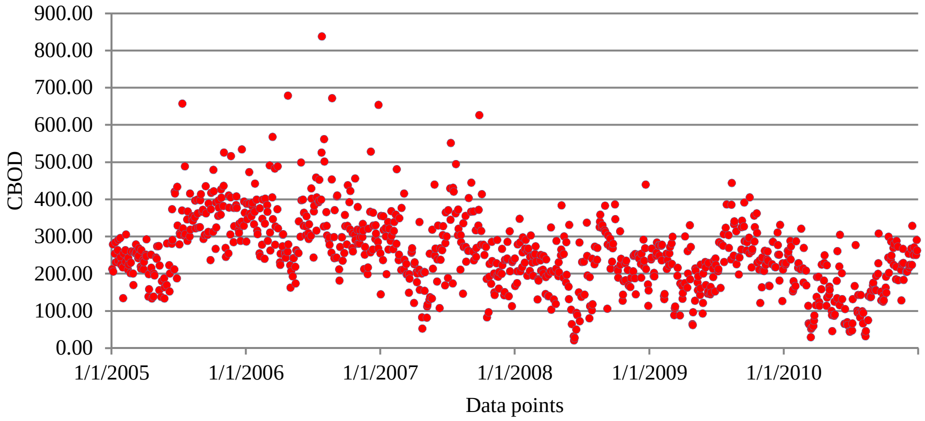


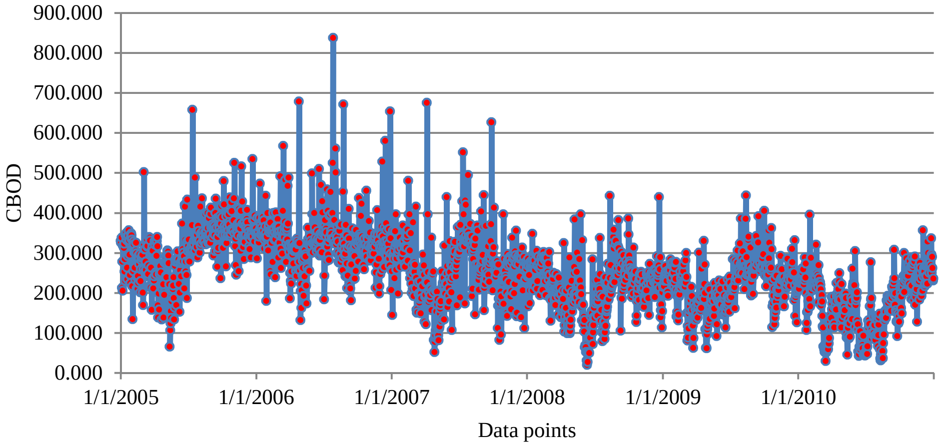
Figure 4.5. Run chart of the actual and predicted CBOD values

Fig. 4.6 (a)-(b) presents the original five-year CBOD data series (data with gaps), and CBOD data series with the gaps filled. The run chart shown in Fig. 4.6(b) is consistent with that of Fig. 4.6(a).

In the next section, the prediction of CBOD values at future time horizons with data-mining models is discussed.



(a)



(b)

Figure 4.6. Time-series plot of CBOD: (a) Original data with gaps, (b) Data with filled gaps

4.3.2. Algorithm selection and learning

The resulting processed data (Fig. 4.6(b)) is used for predicting CBOD values. The input parameters predicting CBOD consists of the CBOD values recorded in the past, also referred to as ‘memory parameters’.

Owing to seasonal variations in the CBOD values, models incorporating seasonality are also considered. The initial data is divided into four seasons: Season 1

(Jan-Mar), Season 2 (Apr-June), Season 3 (July-Sep), and Season 4 (Oct-Dec). The run chart of the CBOD values for the four data seasons is plotted. It can be seen in Fig. 4.7 that CBOD in Season 1 behaves similarly to Season 2, whereas Season 3 is similar to Season 4. Based on this observation, data corresponding to Seasons 1 and 2 is combined. The resulting combination constitutes the modified seasonal model. The distribution of the data split is provided in Table 4.4.

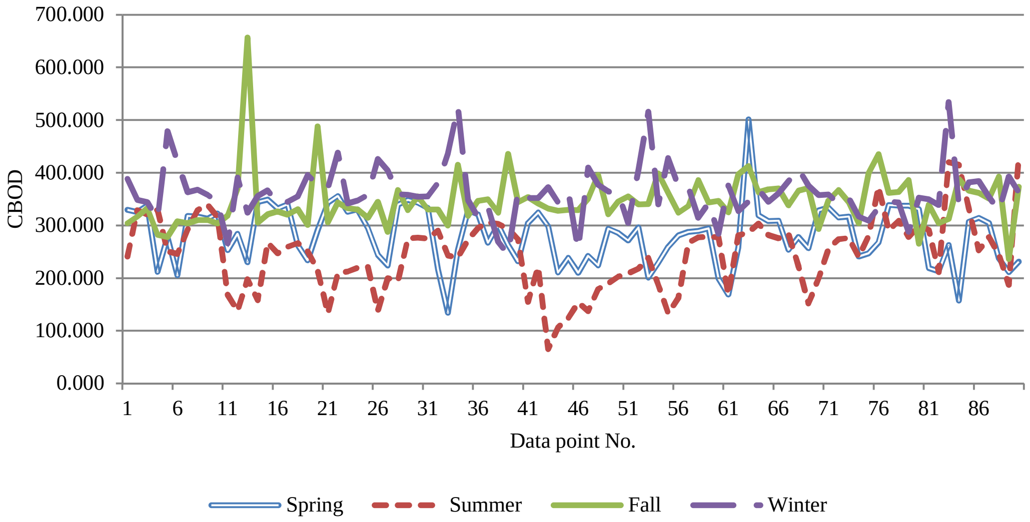


Figure 4.7. Run chart of CBOD in different seasons

Table 4.4. Data split description

No.	Model Type	Dataset No.	Remarks
1	Integrated model	01	Entire year data
2	Seasonal model	04 (Spring, Summer, Fall, Winter)	Seasonal data (Data from Jan-Mar, Apr-Jun, Jul-Sep, and Oct-Dec)
3	Modified seasonal model	02 (Low CBOD Season, High CBOD Season)	Seasonal data (Data from Jan-Jun, and Jul-Dec)

In the research reported in this research, four data-mining algorithms are applied to the three datasets. The CBOD prediction models are expressed in (4.5)-(4.6).

$$\hat{y}_{\text{integrated}}(t) = f\begin{cases} \text{CBOD}_d(t-1), \text{CBOD}_d(t-2), \text{CBOD}_d(t-3), \ldots \text{CBOD}_d(t\text{ -}10), \\ MAX\{\text{CBOD}_d(t-1), \ldots \text{CBOD}_d(t-10)\}, MEAN\{\text{CBOD}_d(t-1), \ldots \text{CBOD}_d(t-10)\}, \\ MIN\{\text{CBOD}_d(t-1), \ldots \text{CBOD}_d(t-10)\}, StdDev\{\text{CBOD}_d(t-1), \ldots \text{CBOD}_d(t-10)\} \end{cases} \tag{4.5}$$

$$\hat{y}_{\text{seasonal}}(t) = f\begin{cases} \text{CBOD}_y(t-1), \text{CBOD}_y(t-2), \text{CBOD}_y(t-3), \ldots \text{CBOD}_y(t\text{ -}5), \\ MAX\{\text{CBOD}_y(t-1), \ldots \text{CBOD}_y(t-5)\}, MEAN\{\text{CBOD}_y(t-1), \ldots \text{CBOD}_y(t-5)\}, \\ MIN\{\text{CBOD}_y(t-1), \ldots \text{CBOD}_y(t-5)\}, StdDev\{\text{CBOD}_y(t-1), \ldots \text{CBOD}_y(t-5)\}, \\ \text{CBOD}_d(t-1), \text{CBOD}_d(t-2), \text{CBOD}_d(t-3), \ldots \text{CBOD}_d(t\text{ -}10), \\ MAX\{\text{CBOD}_d(t-1), \ldots \text{CBOD}_d(t-10)\}, MEAN\{\text{CBOD}_d(t-1), \ldots \text{CBOD}_d(t-10)\}, \\ MIN\{\text{CBOD}_d(t-1), \ldots \text{CBOD}_d(t-10)\}, StdDev\{\text{CBOD}_d(t-1), \ldots \text{CBOD}_d(t-10)\} \end{cases} \tag{4.6}$$

(10)

In the integrated model (4.5), the predicted daily CBOD value ($\hat{y}_{\text{integrated}}(t)$) is a function of the memory parameters (i.e., CBOD values of past 10 days) and the statistical measures, such as mean, maximum, minimum, and standard deviation of past 10-day data. Thus, the total number of input parameters is 14. In the seasonal model (10), $\hat{y}_{\text{seasonal}}(t)$ is a function of the input parameter over the past 5-year data (denoted as y in the CBOD subscript), the past 10-day data of the same year as its memory parameter, and the statistical measures of the 5-year and 10-day data. The overall number of input parameters for the seasonal model is 23. The same function is applied to the modified seasonal dataset.

Four data-mining algorithms – the multilayered perceptron (MLP), the classification and regression tree (C&RT), the multivariate adaptive regression spline (MARS), and the random forest (RF) – are employed to construct prediction models. MLP is a feed-forward neural network algorithm; it learns hidden patterns in the data by adaptively adjusting the weights of its neurons The MLP structure includes input layers, hidden layers, and an output layer. MARS is a non-parametric algorithm for solving regression-type problems. It predicts continuous parameters based on a set of predictors.

RF is data-mining approach used for both classification and regression; it selects the best split of a node based on the randomly selected subset of predictors. C&RT constructs binary trees for both classification trees and regression and uses the minimization of prediction square errors as criteria for splitting the nodes.

Table 4.5 describes the results produced by the four data-mining algorithms. For the integrated model, the MLP outperformed the remaining three data-mining algorithms by achieving a prediction accuracy of over 85%. The accuracy of the other three algorithms – MARS, C&RT, and RF – is similar.

Table 4.5. Integrated model training results

Algorithm	MAE	MRE (%)	Accuracy (%)
MLP	35.84	14.91	85.08
C&RT	38.92	16.64	83.35
MARS	39.14	16.39	83.60
RF	37.27	16.63	83.36

Table 4.6 presents the test results produced by the four seasonal models constructed by different data-mining algorithms. Satisfactory accuracy was found for the fall seasons (86.88%) and winter (89.84%); however, spring and summer (30.00%) seasons yielded rather low accuracy. The reason behind this poor accuracy is the low target CBOD values which were not seen in the training data set. Figs (4.8)-(4.9) compares the results for the spring and winter seasons. It can be seen from Fig. 8 that the actual CBOD values are low. The MARS algorithm frequently over-predicted as it learned from higher values. The predicted values shown in Figs (4.8)-(4.9) were produced by the best-performing algorithms.

Talbe 4.6. Test results for seasonal models

Spring			
Algorithm	MAE	MRE (%)	Accuracy (%)
MLP	146.0	218.86	N/A
C&RT	213.6	296.3	N/A
MARS	**87.04**	**135.0**	**N/A**
RF	166.7	240.8	N/A
Summer			
Algorithm	MAE	MRE (%)	Accuracy (%)
MLP	152	230.0	N/A
C&RT	90.66	143.38	N/A
MARS	**45.49**	**69.00**	**30.99**
RF	97.00	166.13	N/A
Fall			
Algorithm	MAE	MRE (%)	Accuracy (%)
MLP	40.17	21.18	78.81
C&RT	23.69	13.80	86.19
MARS	**24.06**	**13.11**	**86.88**
RF	53.99	27.34	72.65
Winter			
Algorithm	MAE	MRE (%)	Accuracy (%)
MLP	42.06	15.56	84.44
C&RT	59.43	21.77	78.22
MARS	39.56	15.17	84.82
RF	**28.53**	**10.15**	**89.84**

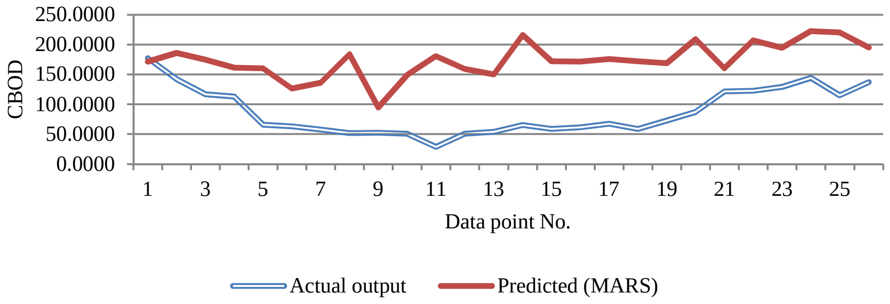


Figure 4.8. Actual and predicted CBOD values in spring season

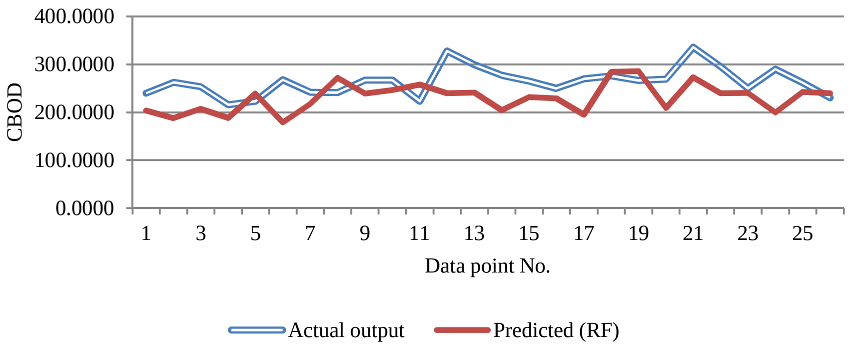


Figure 4.9. Actual and predicted CBOD values in winter season

Overall, the MARS algorithm provided the best accuracy for most seasonal models. The algorithms selected to perform time-ahead predictions are shown in bold (Table 4.6). Table 4.7 summarizes the results obtained from the modified seasonal models; namely, high CBOD and low CBOD seasons. As anticipated, algorithms predicting high CBOD values yielded better results when compared with the algorithms predicting low CBOD. For the high CBOD model, C&RT yielded the best accuracy of 86.51%; whereas, for the low CBOD model, MLP yielded the best accuracy of 69.76%.

Table 4.7. Test results produced from the modified seasonal model

Low CBOD season			
Algorithm	MAE	MRE (%)	Accuracy (%)
MLP	**27.88**	**30.24**	**69.76**
C&RT	51.73	46.88	53.12
MARS	34.26	34.73	65.21
RF	36.89	43.47	56.52
High CBOD season			
Algorithm	MAE	MRE (%)	Accuracy (%)
MLP	77.66	30.99	69.00%
C&RT	**31.90**	**13.48**	**86.51%**
MARS	33.33	14.51	85.48
RF	36.46	14.58	85.41

4.4 Computational results

In this section, the algorithms that produced the best-performing models are employed to perform time-ahead predictions. The maximum prediction length is five days.

4.4.1 Prediction results for integrated model

In this model, the MLP algorithm is used to perform predictions. Table 4.8 presents the results obtained for five-day-ahead predictions. Accuracy in the range of 67.43-77.06% is found. Fig. 4.10 displays the actual and predicted CBOD values at t + 0 days. Fig. 10 demonstrates that, in most cases, the MLP is able to approximate the CBOD values.

Table 4.8. Time-ahead predictions by the integrated model

Time stamp [days]	MAE	MRE (%)	Accuracy
t + 0	30.88	22.93	77.06
t + 1	33.78	25.44	74.55
t + 2	36.08	27.67	72.32
t + 3	38.38	29.74	70.25
t + 4	40.02	31.36	68.63
t + 5	41.17	32.57	67.43

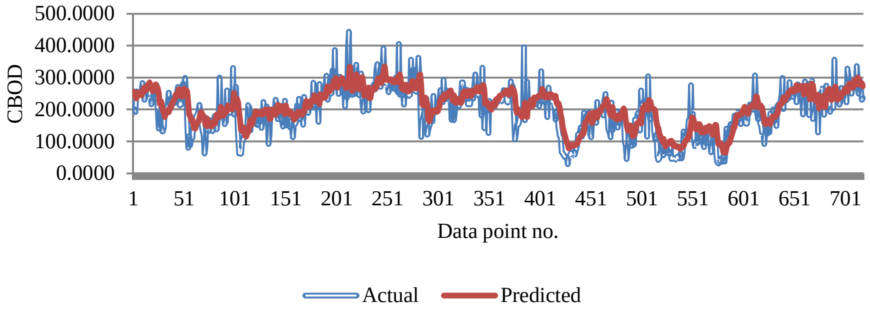


Figure 4.10. Comparison of actual and predicted CBOD values produced with the MLP algorithm

4.4.2 Prediction results for seasonal data

In this section, CBOD in the fall season is predicted using MARS; whereas, in the winter, it is predicted with RF. The fall season accuracy is in the range of 75.61-85.81% for five days, whereas the winter season accuracy is in the range of 88.73-89.00% (see Table 4.9). RF yielded consistent accuracy in five-day-ahead predictions with steady predicted values. The pattern of actual values was not successfully found. Figs 4.11-4.12 compare the actual and predicted values for t + 0 days ahead for the fall and winter seasons, respectively.

Table 4.9. Accuracy of the time-ahead prediction of seasonal models

<table>
<tr><td colspan="4">Fall season (MARS)</td></tr>
<tr><td>Time stamp [days]</td><td>MAE</td><td>MRE (%)</td><td>Accuracy</td></tr>
<tr><td>*t* + 0</td><td>26.06</td><td>14.18</td><td>85.81</td></tr>
<tr><td>*t* + 1</td><td>32.23</td><td>17.03</td><td>82.96</td></tr>
<tr><td>*t* + 2</td><td>33.91</td><td>18.67</td><td>81.32</td></tr>
<tr><td>*t* + 3</td><td>37.31</td><td>20.78</td><td>79.21</td></tr>
<tr><td>*t* + 4</td><td>38.61</td><td>22.20</td><td>77.79</td></tr>
<tr><td>*t* + 5</td><td>42.33</td><td>24.38</td><td>75.61</td></tr>
<tr><td colspan="4">Winter season (RF)</td></tr>
<tr><td>Time stamp [days]</td><td>MAE</td><td>MRE (%)</td><td>Accuracy</td></tr>
<tr><td>*t* + 0</td><td>31.44</td><td>10.95</td><td>89.04</td></tr>
<tr><td>*t* + 1</td><td>31.46</td><td>10.99</td><td>89.00</td></tr>
<tr><td>*t* + 2</td><td>31.52</td><td>11.00</td><td>88.99</td></tr>
<tr><td>*t* + 3</td><td>31.55</td><td>11.00</td><td>89.00</td></tr>
<tr><td>*t* + 4</td><td>31.87</td><td>11.14</td><td>88.85</td></tr>
<tr><td>*t* + 5</td><td>32.17</td><td>11.26</td><td>88.73</td></tr>
</table>

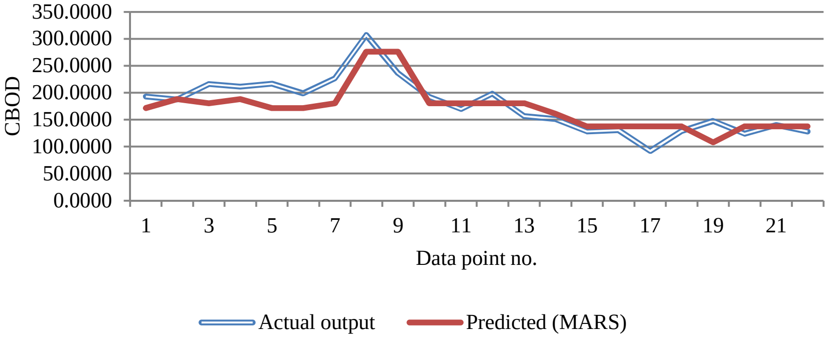
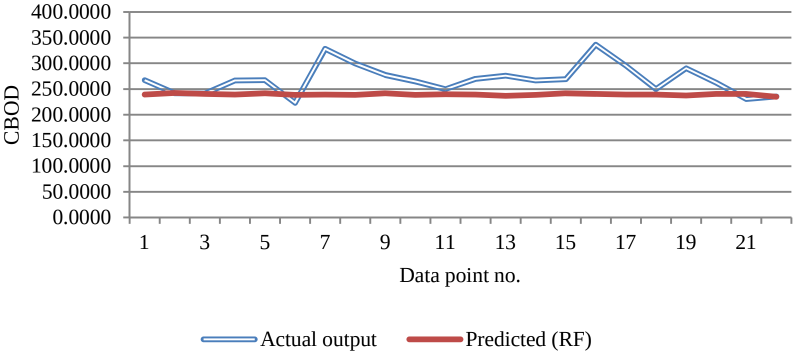


Figure 4.11. Comparison of the actual and predicted CBOD values in the fall season



Fiture 4.12. Comparison of the actual and predicted CBOD values in the winter season

4.4.3 Prediction results for modified seasonal data

In this section, the low CBOD season data is predicted with the MLP algorithm; whereas the high CBOD season data is predicted with the C&RT algorithm. Table 10 describes the results produced for both seasons. The accuracy in the high CBOD season is in the range of 84.82-87.69%. In the low CBOD season, the accuracy is in the range of 46.31-70.51%. Compared with the results obtained in the spring and summer seasons (discussed in Section 3.2.2), an improvement of 16-40% in overall accuracy is obtained; however, compared with the high CBOD season, the accuracy is still quite low. Figs

(4.13)-(4.14) show the run chart comparison of the actual and predicted values in high CBOD and low CBOD seasons, respectively.

Table 4.10. Prediction results for the modified seasonal data

High CBOD season (C&RT)			
Time stamp [days]	MAE	MRE (%)	Accuracy
t + 0	32.61	13.93	86.02
t + 1	35.37	15.17	84.82
t + 2	31.48	13.75	86.24
t + 3	32.28	13.70	86.29
t + 4	34.13	14.04	85.95
t + 5	29.39	12.32	87.69
Low CBOD season (MLP)			
Time stamp [days]	MAE	MRE (%)	Accuracy
t + 0	28.91	30.16	70.51
t + 1	32.76	34.76	65.90
t + 2	36.93	39.97	60.02
t + 3	40.06	45.25	54.74
t + 4	40.66	47.21	52.78
t + 5	46.54	53.55	46.31

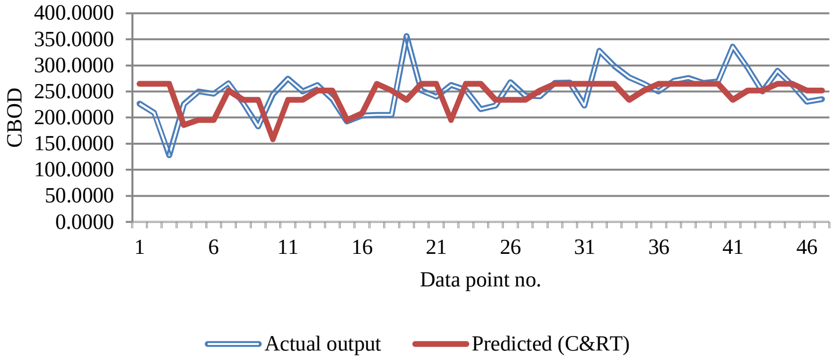


Figure 4.13. Comparison of the actual and predicted values in the high CBOD season

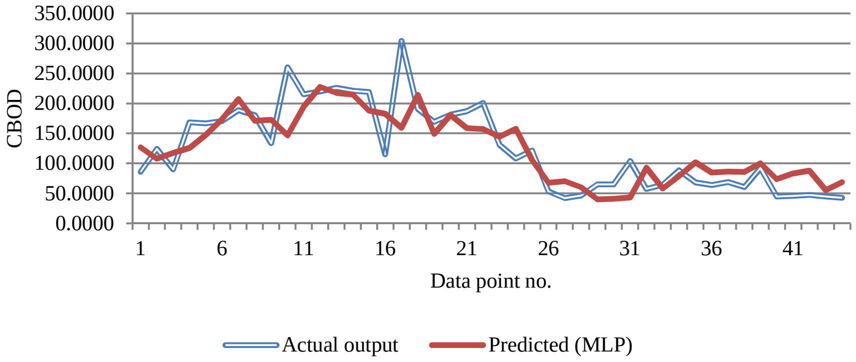


Figure 4.14. Comparison of the actual and predicted values in the low CBOD season

Based on the results shown above, the seasonal models are suitable for prediction in the fall and winter seasons when the CBOB values are high; for the lower CBOD values, however, the modified seasonal models are appropriate. Even though the average accuracy of the integrated model is high compared with the low CBOD season model, the prediction accuracy for low CBOD seasons is rather low.

CHAPTER 5

OPTIMIZATION OF WASTEWATER PUMPING PROCESS

5.1 Introduction

To lift raw wastewater collected from sewer lines to following treatment process, particularly where the elevation of the wastewater inlet is not sufficient for gravity flow, boosting pumps are usually used [67, 68]. This wastewater pumping process consumes a significant electricity to deliver wastewater. It typically consumes 10% to 20% of the total energy used by the whole plant.

Conventional wastewater treatment plants use a wet well to temporally store the raw wastewater. The purpose of it is to provide a method allowing automatic operation of the boosting pumps with a simple control [69]. Multiple pumps are controlled and supervised by the programmable logic controller (PLC). The drawback of such simple PLC control is not efficient in pumping operation. The pre-defined control scheme may not be able to select the best pump configuration to run and properly adjust the pump rotating speed. In addition, the inflow rate of the raw wastewater changes significantly over time and weather condition such as rainstorm [70]. The pump performance also declines with time, and the efficiency of the pump usually gets worse. Therefore the operation of the pumping process is not highly efficient under such control scheme, resulting in energy waste.

To save energy consumption in wastewater pumping process, different approaches have been investigated. More efficient pumps have been designed and many control schemes have been developing [71-74]. Besides the improvement on the single pump, Ma and Wang [75] has presented an optimal control strategies for variable speed pumps with different configurations in complex building air conditioning systems to enhance the energy efficiency. The results showed that about 12-32% of pump energy could be reduced by using the optimal control settings. A pilot study by Zhang et.al [76] obtained a 24.25% energy saving by using the optimal pump system schedule in a wastewater treatment plant.

In this work, pumping operation is optimized with a data-driven model derived by multi-layer perceptron (MLP) neural networks. Data-driven approach has been successfully used in scientific and engineering applications, such as wind energy, HVAC, and wastewater treatment [77-79] to improve process or energy efficiency. The goal of optimization in this research is to reduce the energy consumption by wastewater pumping process. The configuration of pumps, i.e., the number of running pumps and the rotating speed of the pump are two control variables in the problem formulation. To find the optimal operation configuration, a two-level integration algorithm is proposed and employed to find solutions for running number of pumps and rotating speed. The model and the optimization results are then discussed in detail.

5.2 Data description

The data used in this work was also collected at the Wastewater Reclamation Facility (WRF), located in Des Moines, Iowa. WRF is operated to treat wastewater from 17 metro area municipalities, counties, and sewer districts. It serves a population of 500,000 and processes over 50 MGD of raw wastewater per day.

WRF operates 6 pumps to lift the raw wastewater. As shown in Figure 5.1, the wastewater collected from areas is flown into the plant and pass the bar screens. The big items, such as rocks and trunks, are screened out for later disposal. The grit chamber is used to settle down stones and sands. Then the wastewater enters into the wet well and is lifted by raw wastewater boosting pumps to the primary clarifier for the initial treatment. Each pump is rated as 55 MGD capacity and 700 kW per 15 minutes with variable speed drive. Based on the inflow rate and wet well level, the number of operating pumps varies to keep the wet well level. For example, more pumps are operated when wet level and inflow rate are high. The control scheme is determined by a pre-programmed PLC control.

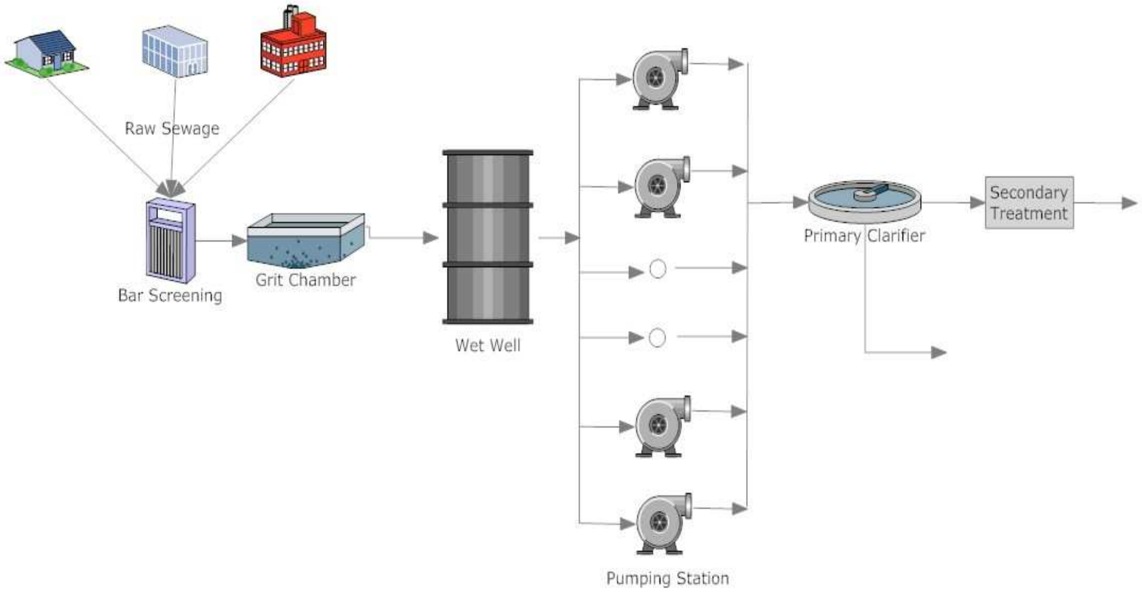


Figure 5.1. Flow chart of wastewater pumping process

The wastewater pumping process data used in this research is taken over the period from 4/1/2011 to 8/4/2012. The data is processed to exclude errors and outliers. There are 62 possible configurations theoretically for 6 pumps operation. Through mining the dataset, 20 different configurations are found. Other configurations are not available in current dataset because of the operation practice in the plant. The dataset used to build models is summarized in Table 5.1. The operated pump in each configuration is shown in the second column. For example, pumps {2, 3} are running simultaneously in configuration C10, and pumps {1, 3, 4, 5} are operated at the same in configuration C20. The available cases for each configuration is divided into training set to build the models and the test set to validate the accuracy of the models.

Table 5.1. Dataset description

Configuration	Pumps	Available cases	Training cases	test cases
C_1	{1}	5374	4030	1344
C_2	{2}	4421	3315	1106
C_3	{3}	5550	4162	1388
C_4	{4}	7904	5928	1976
C_5	{5}	4397	3297	1100
C_6	{6}	2002	1501	501
C_7	{1,2}	491	368	123
C_8	{1,4}	607	455	152
C_9	{1,5}	377	282	95
C_{10}	{2,3}	2334	1750	584
C_{11}	{2,4}	897	672	225
C_{12}	{3,4}	364	273	91
C_{13}	{4,5}	277	207	70
C_{14}	{4,6}	1052	789	263
C_{15}	{5,6}	714	535	179
C_{16}	{1,4,5}	405	303	102
C_{17}	{2,4,5}	353	264	89
C_{18}	{2,4,6}	468	351	117
C_{19}	{3,4,5}	482	361	121
C_{20}	{1,3,4,5}	330	247	83

5.3 Building and validating models

To optimize the wastewater pumping process, the model of total pumping energy consumption should be built. Different from a mathematical equation from the engineering principle, the model is built based on the operation data by data-mining algorithm. Equation (5.1) and (5.2) express the total pumping energy consumption P and energy usage $p_{i,t}$ of a single configuration:

$$P = \sum_{i=C_1}^{C_{20}} p_{i,t} x_{i,t} \tag{5.1}$$

$$p_{i,t} = f(v_{i,t} L_t) \tag{5.2}$$

where i indicates the ith number of configuration listed in Table 5.1 . $x_{i,t}$ is a binary variable which either equals one or zero. The summation of all $x_{i,t}$ should be equal to 1 as only one configuration should be on at time t . $v_{i,t}$ is the pump speed for that configuration and L_t is the wet well level at time t .

The energy usage $p_{i,t}$ of a single configuration is modeled by multi-layer perceptron (MLP) neural network. MLP neural networks have been successfully used to find the complex and non-linear pattern in science and engineering problems [80, 81]. In this work the number of hidden layer is 1 and the number of neurons in the hidden layer varies in order to obtain higher accuracy. Different activation functions, i.e., logistic, tanh, sigmoid, identity and sigmoid, are selected during training process.

As shown in Figure 5.2, the first 40 cases in the test set in configuration C1 have been successfully modeled by MLP neural network. The observed and predicted values are very close to each other as well as the well modeled trend. Figure 5.3 shows the similar modeling results. Table 5.2 summarizes the prediction accuracy metrics for all configurations. It also illustrates the excellent accuracy of the models.

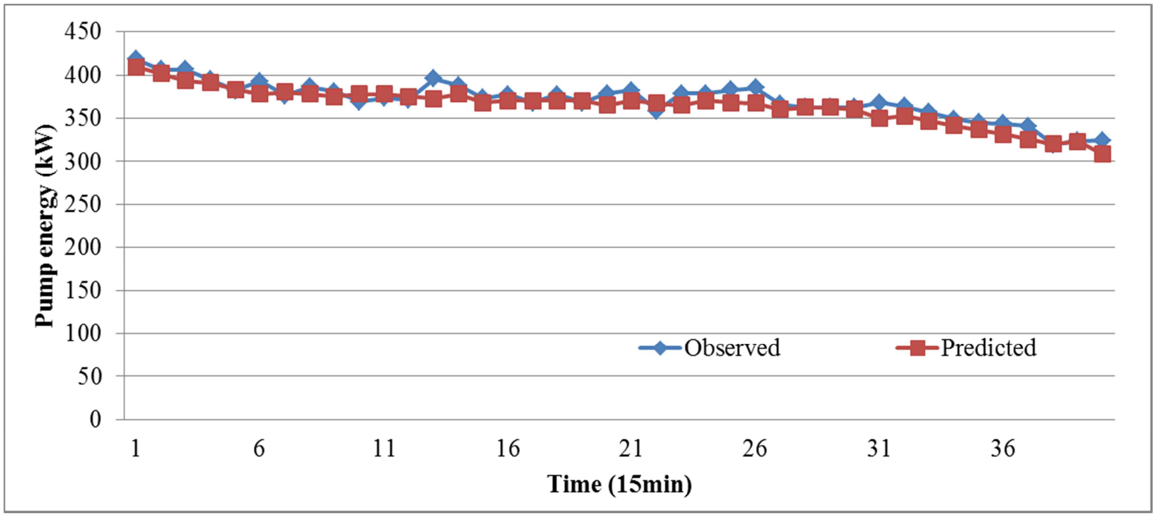


Figure 5.2. Observed and MLP neural network model predicted energy consumption for C1

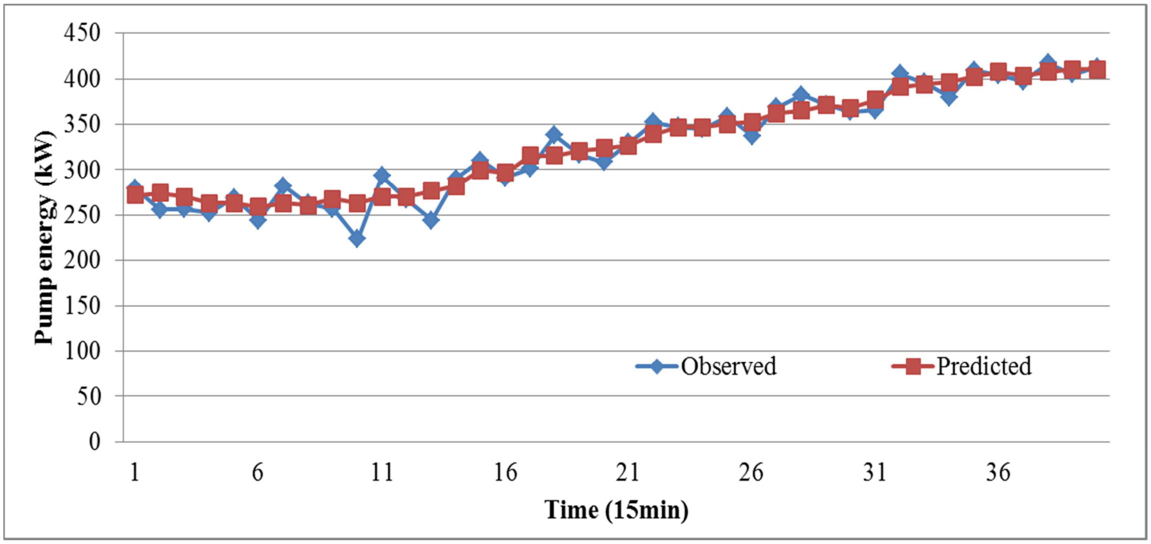


Figure 5.3. Observed and MLP neural network model predicted energy consumption for C20

Table 5.2. Performance metrics of energy consumption models

Configuration	MAE	MAPE	SD of APE
C_1	5.860669733	0.016959492	0.012649011
C_2	24.59039875	0.064456634	0.040341825
C_3	8.965691004	0.021630219	0.017612145
C_4	8.044528664	0.020567482	0.015890538
C_5	5.269917086	0.016643775	0.022236368
C_6	11.05984522	0.034307164	0.03591208
C_7	18.40614128	0.032814162	0.03080949
C_8	5.34561635	0.008269741	0.006961746
C_9	11.42316688	0.014271535	0.012547263
C_{10}	12.09757271	0.018873324	0.012661691
C_{11}	14.71821326	0.025777816	0.02445185
C_{12}	5.371886446	0.00988624	0.006759664
C_{13}	10.00627462	0.016004065	0.023185261
C_{14}	15.35672135	0.027175781	0.020558307
C_{15}	19.12878501	0.038058685	0.030396031
C_{16}	11.83564606	0.010016076	0.008311363
C_{17}	8.074325455	0.00734845	0.00554528
C_{18}	13.84064385	0.014572221	0.014619868
C_{19}	15.03526129	0.015937855	0.015868939
C_{20}	20.10760802	0.012819738	0.008056547

Besides energy consumption model, the wastewater outflow rate model is also built by MLP neural network. The purpose to develop this model is to compute the wet well level change at two continuous time step. The outflow rate $O_{i,t}$ for the ith configuration can be expressed in Equation (5.3), and the wet well level change can be calculated as shown in Equation (5.4):

$$O_{i,t} = h(v_{i,t} L_t) \tag{5.3}$$

$$L_t - L_{t-T} = \frac{I_t - O_{i,t}}{A} \tag{5.4}$$

where h is the function which is obtained by MLP neural network. I_t is the inflow rate flowing into the wet well at time t, A is the area of the wet well.

Figure 5.4 and 5.5 show the first 40 cases of modeling results of the wastewater outflow rate for configuration 1 and configuration 20. The prediction is well performed by MLP neural network. The computed prediction metrics shown in Table 5.3 also indicate the outflow rate is well modeled.

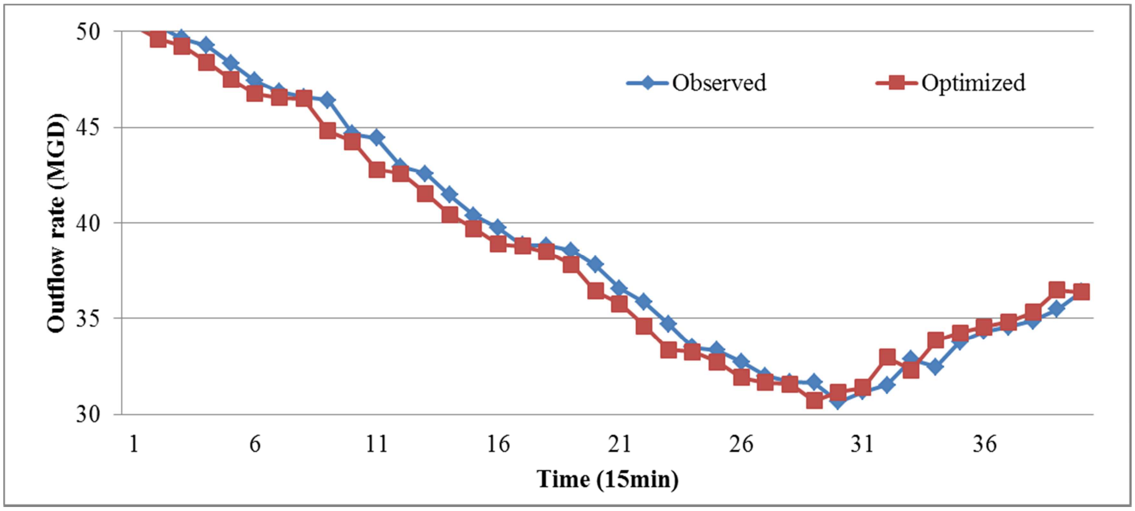


Figure 5.4. Observed and MLP neural network model predicted outflow rate for C1

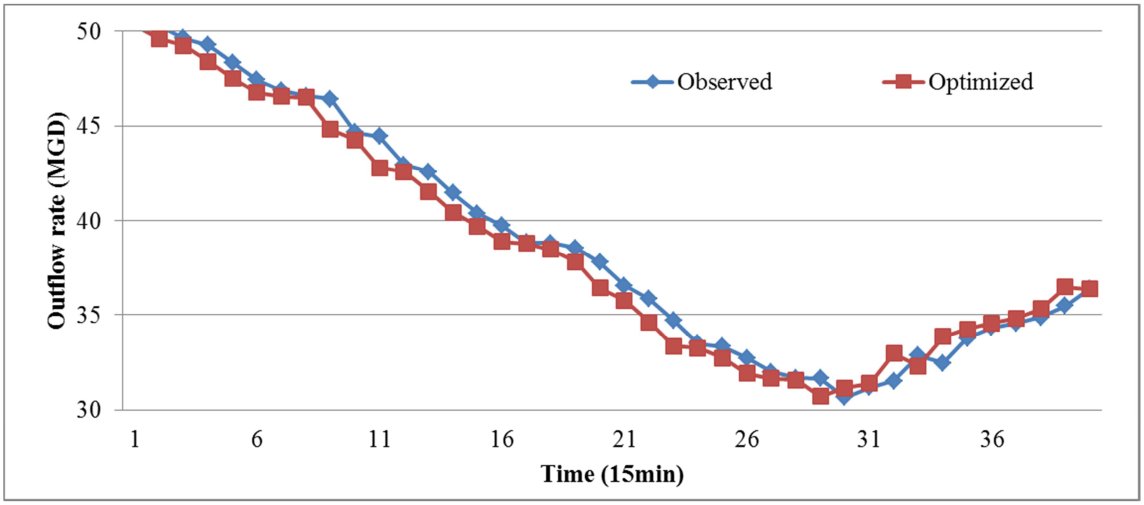


Figure 5.5. Observed and MLP neural network model predicted outflow rate for C20

Table 5.3. Performance metrics of energy outflow rate models

Configuration	MAE	MAPE	SD of APE
C_1	2.762826	0.07105776	0.069098709
C_2	3.617816	0.08703103	0.076412469
C_3	2.901154	0.06592743	0.080804995
C_4	3.651067	0.08249157	0.08752017
C_5	2.98317	0.06720429	0.061726867
C_6	6.71276	0.16561686	0.130624141
C_7	3.886455	0.06027305	0.049863389
C_8	1.260544	0.01558512	0.010764965
C_9	2.481695	0.03171347	0.11446598
C_{10}	2.026298	0.02906259	0.02413035
C_{11}	5.13326	0.07972925	0.075420871
C_{12}	1.951649	0.03241246	0.026701404
C_{13}	5.569913	0.07035922	0.085815405
C_{14}	6.493714	0.0887062	0.163131003
C_{15}	4.688093	0.0700909	0.166293597
C_{16}	3.210542	0.02026464	0.025675297
C_{17}	1.194051	0.00877542	0.007754813
C_{18}	2.334557	0.02118571	0.017524321
C_{19}	5.110837	0.04467025	0.052162884
C_{20}	2.732426	0.01506617	0.011233222

5.4 Optimizing pumping process

5.4.1 Problem formulation

The model trained by the MLP neural network is used to construct the optimization model. To optimize pumping process, the single objective minimizing the energy consumption of pumping process can be expressed as a function of control variables subjecting various constraints.

According to the operation conditions at WRF, the number of operating pumps at the same can be adjusted as well as the rotating speed of the pump. They are the two decision variables to be optimized in this research.

The constraints are based on physical limitation and operation practice.

1. Only one case described in Section 5.3 can be effective at one time step. It means that all other cases are off when one case is on.

2. The wet well level change cannot exceed a defined value δ

3. The high wet well level must be no more than the maximum allowable value $L_{\max}$.

4. The rotating speed of pump should be from 80% to 100% full speed to keep the pump efficiency.

Therefore, the single optimization problem can then be formulated in (5.5):

$$\min_{x,v} P$$

subject to:

$$P = \sum_{i=C_1}^{C_{20}} p_{i,t} x_{i,t}$$

$$p_{i,t} = f(v_{i,t} L_t)$$

$$\sum_{C_1}^{C_{20}} x_{i,t} = 1$$

$$| L_t - L_{t-T} | < \delta$$

$$L_t < L_{\max}$$

$$0.8 \le v_{i,t} \le 1$$

$$x_{i,t} \in \langle 0\ or\ 1 \rangle \tag{5.5}$$

5.4.2 Two level integration algorithm

Finding the optimal solution for an optimization problem is challenging, especially for the problem presented in this chapter, which is a mixed-integer nonlinear programming problem. It is not suitable to solve it with a traditional optimization algorithm, such as Genetic Algorithm (GA) or Particle Swarm Optimization Algorithm (PSO).

In this research, a two-level intelligent algorithm is proposed to solve the mixed-integer nonlinear optimization problem. Genetic algorithm [82] is a powerful, general purpose optimization method mimicking biological evolution and it has been successfully applied to solve complex optimization problems. The solutions of a GA are encoded as chromosomes and are evaluated by a fitness function. Pairs of chromosomes selected are based on the values of the fitness function to produce new solutions using crossover operator. Mutation of solutions is also used to promote genetic diversity. GA algorithm preserves solutions that yield high fitness values, and disregards poor quality solutions. As it performs well in discrete optimization problem, it is employed at the first level to find the optimal solution of the discrete decision variable in (5.5).

At the second level, the decision variable of the pump rotating speed should be solved. As it is a continuous variable, therefore, an improved particle swarm optimization is employed at the second level to optimize the pump rotating speed. Particle swarm optimization (PSO) is a stochastic optimization algorithm developed by Eberhart and Kennedy in 1995 [83] which was inspired by the social behavior of birds flocking or fish schooling. PSO has been successfully applied in many research and application areas. It has been demonstrated that PSO produces better results in lower computational time compared with other algorithms. Moreover, PSO has relatively few parameters that need to be adjusted which makes it easier to use.

The standard PSO has some drawbacks. Firstly, the computation results have direct relationship with the parameters. A good selection of parameters can result in better solutions. Secondly, the parameters of PSO are constants. It means they are not changing during the optimization process. And lastly, PSO is easy to be trapped in local optimal.

The larger inertia weight when updating the velocities of the particles in PSO ensures a more effective global search, while smaller inertia weight enables a more efficient local search [84]. Similarly, the learning factors control the ability of local extremes and global extremes search. Therefore, the improved PSO introduces dynamic inertia weight and learning factors, so the particles can search the entire solution space without falling into local optimum in the early iterations. The exceeding boundary control is also introduced the improved PSO algorithm to enhance the overall ability of the algorithm. When the position of the particle is out of the boundary, the position is redefined to make it is in the range of feasible solutions. This approach also ensures the diversity of new particles for undetermined boundary constraints.

The steps of the proposed two-level intelligent algorithm can be expressed as follows:

First level: Genetic Algorithm

Step 1: Generate initial population of (n) chromosomes.

Step 2: Assign fitness of each chromosome in that population using the improved PSO at level 2.

Step 3: Select two best fit chromosomes for reproduction.

Step 4: With a crossover probability cross over the parents to breed a new offspring.

Step 5: With a mutation probability mutate new offspring at each locus. Place new offspring in a new population.

Step 6: If stopping condition is satisfied, then stop and return the best solution in current population. Otherwise, return to Step 2 to start next iteration.

Second level: Improved Particle Swarm Optimization Algorithm

Step 1: Randomly initialize n particle positions $d_i \in R^n$ and velocities $v_i \in R^n$.

Step 2: Evaluate fitness value f_i using current particle positions.

If $f_i \le f_i^b$, then $f_i^b = f_i$, $p_i^b = d_i$

If $f_i \le f^g$, then $f^g = f_i$, $p^g = d_i$

Step 3: Update all particle velocities v_i

$$v_i = \omega v_i + c_1 r_1 (p_i - d_i) + c_2 r_2 (p^g - d_i)$$

$$\omega = \omega_{\max} - k(\omega_{\max} - \omega_{\min}) / It$$

$$c_1 = c_{1start} - k(c_{1start} - c_{1end}) / It$$

$$c_2 = c_{2start} - k(c_{2start} - c_{2end}) / It$$

Step 4: Update all particle positions d_i

$$d_i = d_i + v_i$$

If the particle position is moved out of the boundary, the current velocity of the particle is recalculated based on its distance to the boundary and the variation range.

$$v_i = -dis * v_i / Vari$$

Step 5: Update fitness value f_i^b and f^g

Step 6: If the stopping condition is satisfied, then f^g is the final optimal solution with the particle position p^g. Otherwise, return to step 3 to start next iteration.

Here, n is the number of chromosomes of the first level GA algorithm, particle's position and velocity are d_i and v_i respectively. . Parameter p_i^b is the best individual particle position, and p^g is the best global position. ω is the inertia weight, $\omega_{\max}$ is the maximum and $\omega_{\min}$ is the minimum weight c_1 and c_2 are cognitive and social parameters, c_{1start}, c_{1end}, c_{2start}, and c_{2end} are the start and end values of these learning factors. k and It are current iteration number and total iteration numbers, respectively. r_1 and r_2 are random numbers between 0 and 1. *dis* is the distance of the particle which moves outside the boundary to the boundary, and *Vari* is the variation range. Figure 5.6 shows the flow chart diagram of the proposed two-level intelligent algorithm.

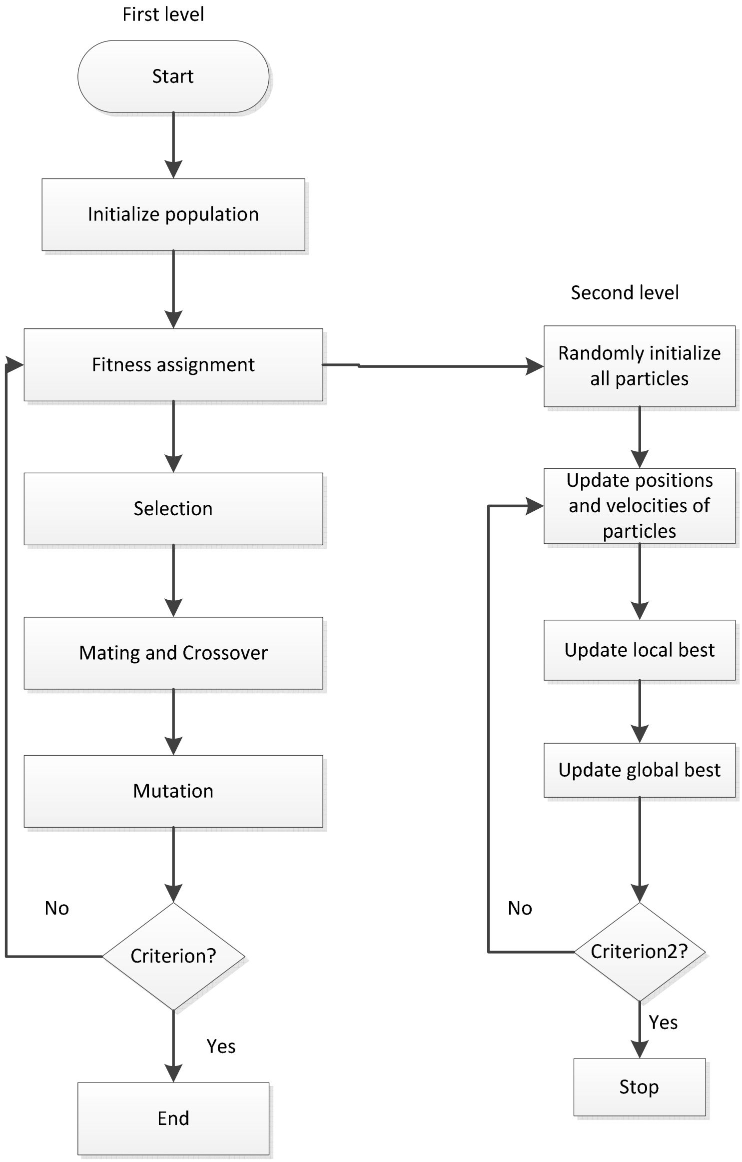


Figure 5.6. The two-level intelligent algorithm

5.4.3 Results and discussion

120 cases from the dataset are used to solve model (5.5) with the proposed two level integration algorithm. As the quantity of wastewater inflow rate has impact on pumping energy consumption, 3 scenarios are investigated in this research and each scenario uses 40 cases. High wastewater inflow rate from 130 to 170 MGD is classified as scenario 1. Scenario 2 with medium inflow rate is from 70 to 80 MGD, and scenario 3 has low inflow rate ranging between 30 to 50 MGD.

Figure 5.7 shows the optimization results for scenario 1. Under the optimized operation, i.e., the optimal combination of running pumps and the optimal rotating speed, the pumping energy consumption can be reduced by 26.9%. The reduced energy usage is mainly due to the finding of a better operating pump combination than the automatic selection by PLC. For example, pump {1, 2, 3, 4} was the operated pumps in the first case, however, pump {3, 4, 5} is the better configuration for the same inflow rate based on the optimization result. This different configuration can save significant energy consumption while keeping almost same wet well level and deliver the same amount of outflow (shown in Figure 5.8 and 5.9). This enables the stable operation of following wastewater treatment process. It is worth to point that the fluctuation in the optimized pump energy consumption is caused by the optimal pump rotating speed. For scenario 1, the original cases have used different pump configurations. The optimized cases have only one configuration but they consume different energy with different pump speeds. To limit the impact to the electricity grid, a constraint to restrain the fluctuation of the energy consumption or smaller the pump speed range may be considered in future research.

Figure 5.10 and 5.11 illustrate the optimization results for scenario 2 and 3. Several cases in scenario 2 consume little higher energy when comparing between observed and optimized results. That is due to the problem formulated in Section 5.4.1 is global optimized. For scenario 2 and 3, the pump energy consumption are reduced 31.6%

and 18.2%, respectively. The main reduction also comes from the optimal pump configuration and rotating speed of the pumps.

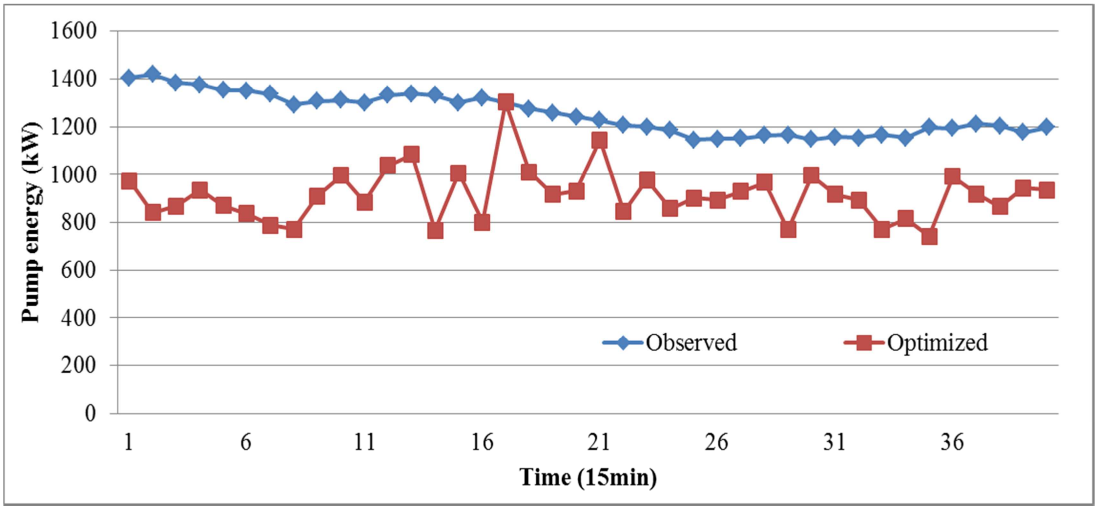


Figure 5.7. Observed and optimized pump energy consumption for scenario 1

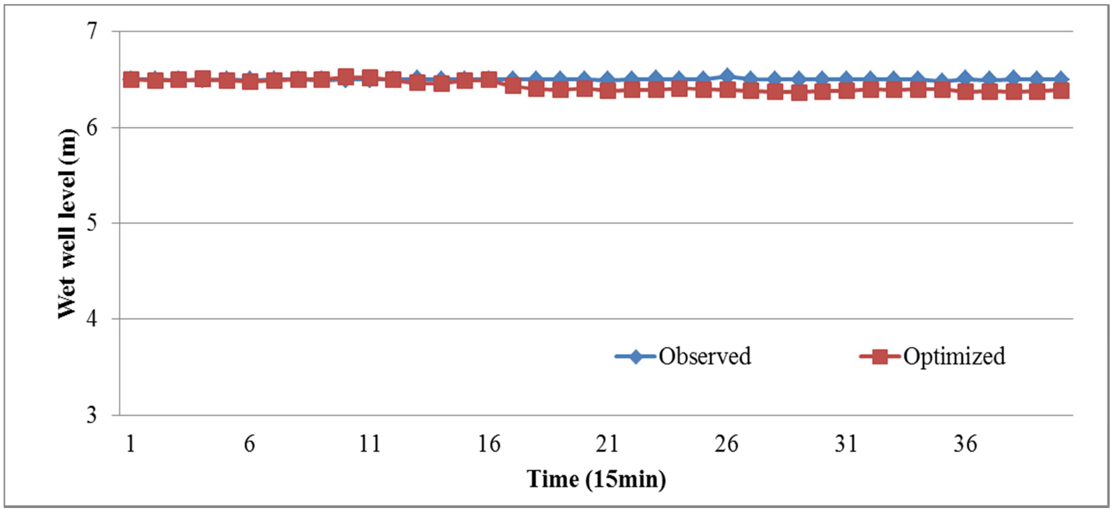


Figure 5.8. Observed and optimized wet well level for scenario 1

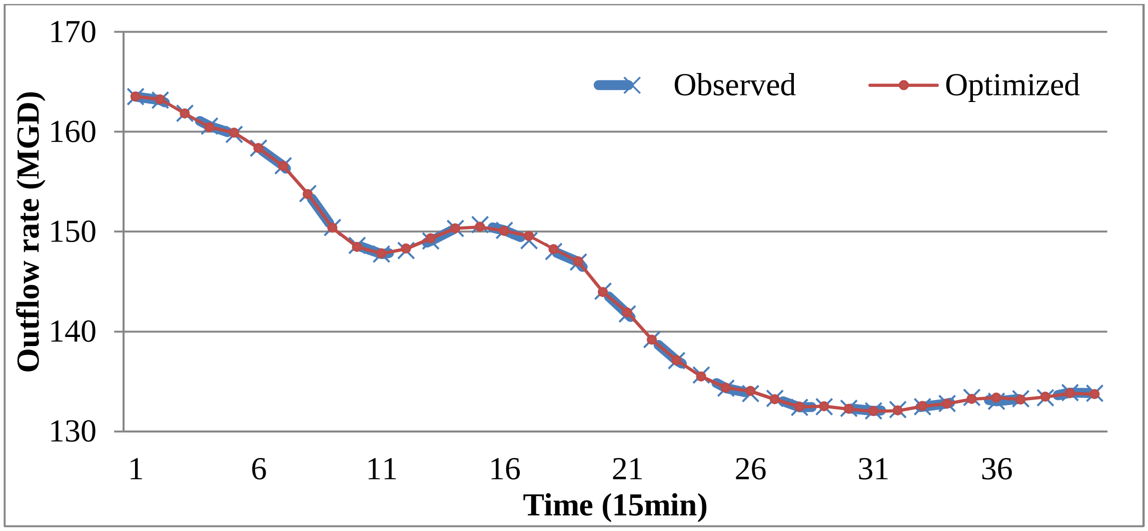


Figure 5.9. Observed and optimized outflow rate for scenario 1

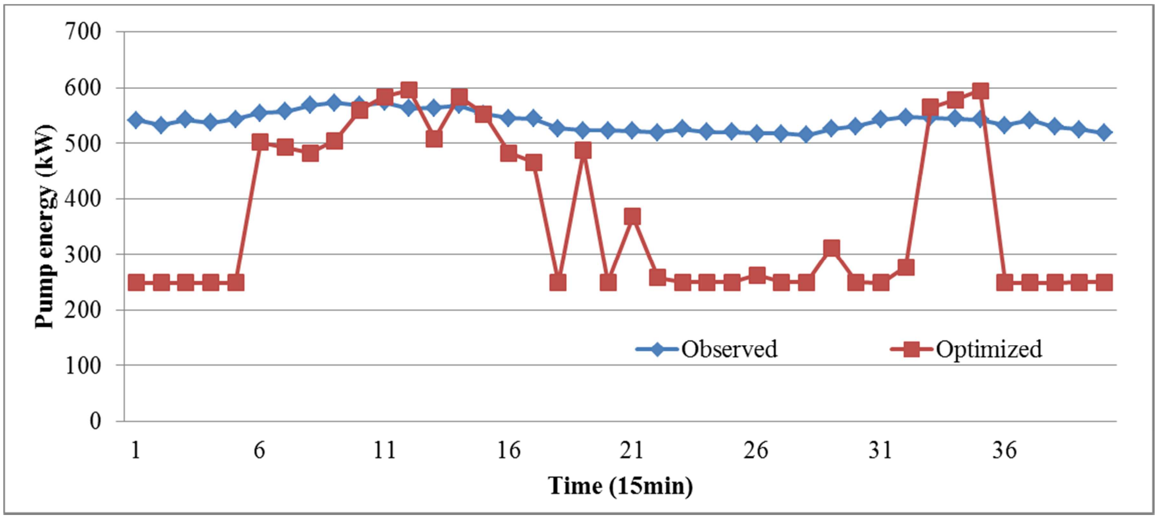


Figure 5.10. Observed and optimized pump energy consumption for scenario 2

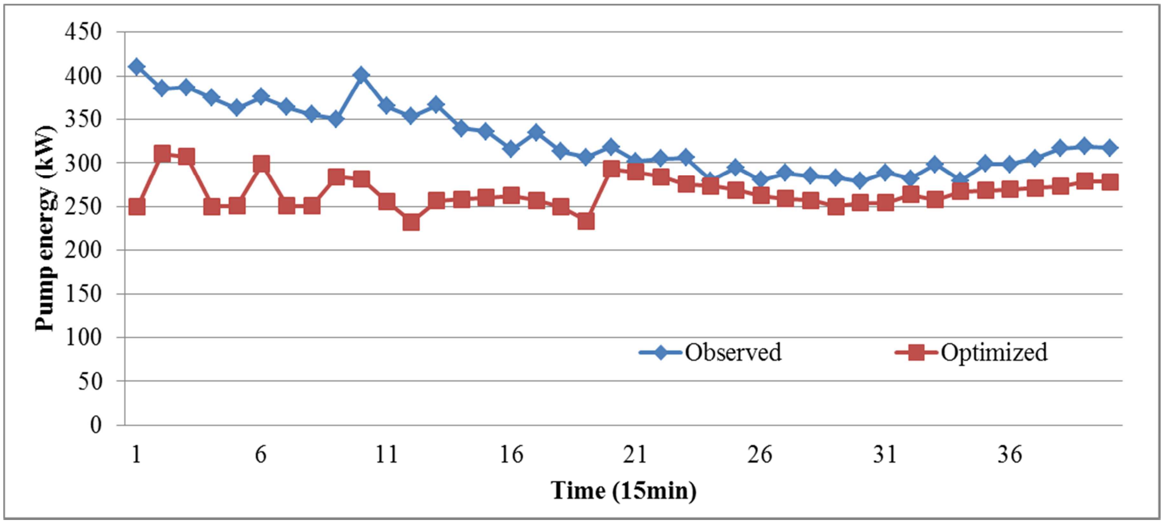


Figure 5.11. Observed and optimized pump energy consumption for scenario 3

Figure 5.12 and 5.13 show the optimization results of wet well level and outflow rate for scenario 2, as well Figure 5.14 and Figure 5.15 for scenario 3. The optimized wet well level and outflow rate are very close to the observed values for scenario 2. For scenario 3, the optimized outflow rate has some variations from observed values. That is because the flow rate is low for this scenario. Small changes in pump speed may cause large fluctuation in outflow rate in this scale.

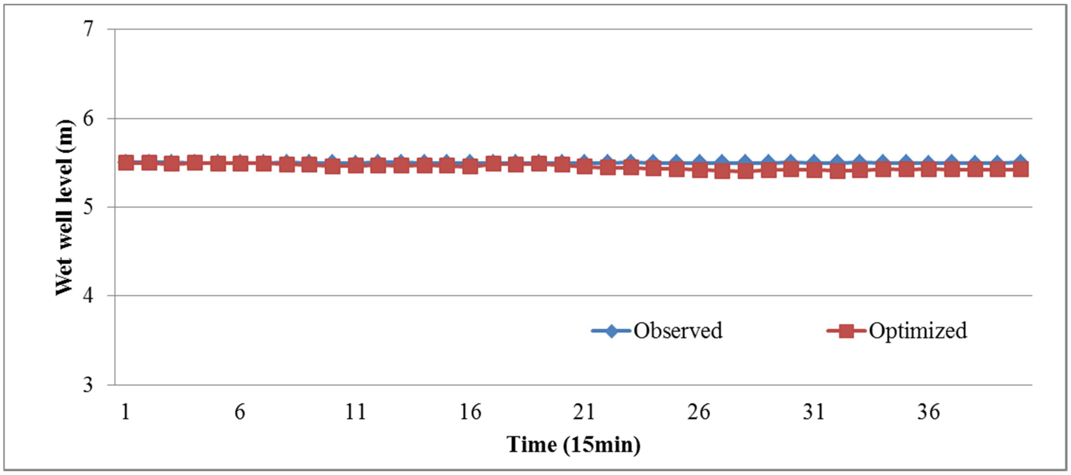


Figure 5.12. Observed and optimized wet well level for scenario 2

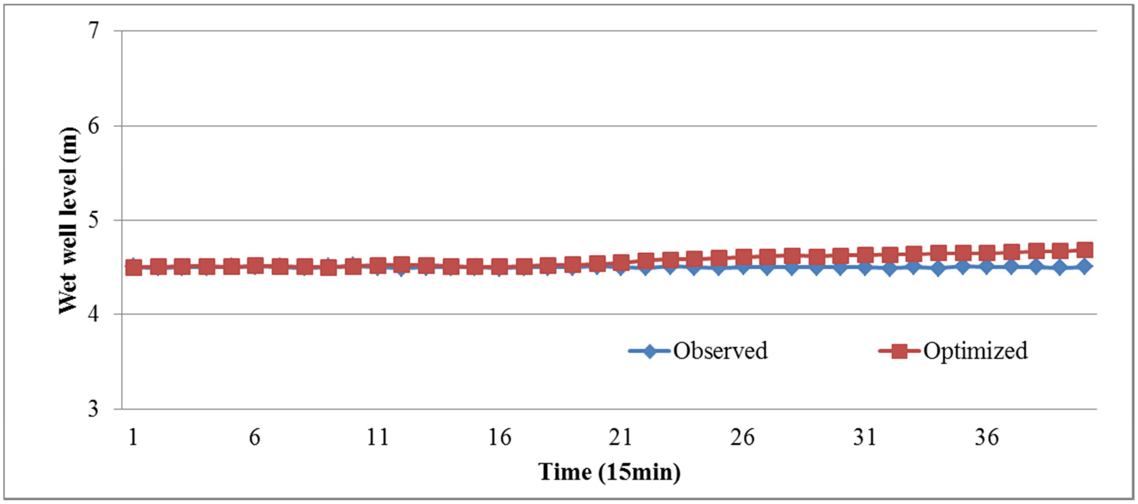


Figure 5.13. Observed and optimized outflow rate for scenario 2

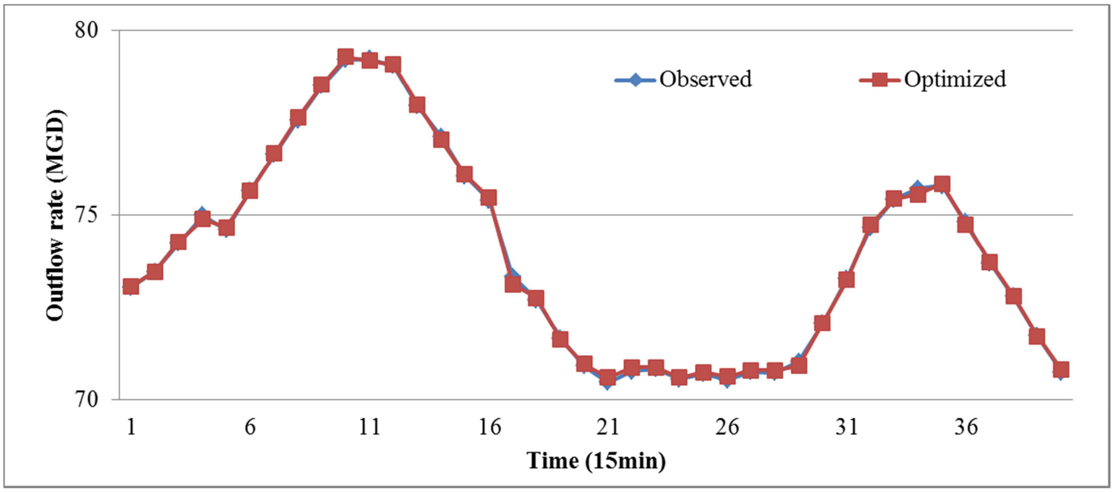


Figure 5.14. Observed and optimized wet well level for scenario 3

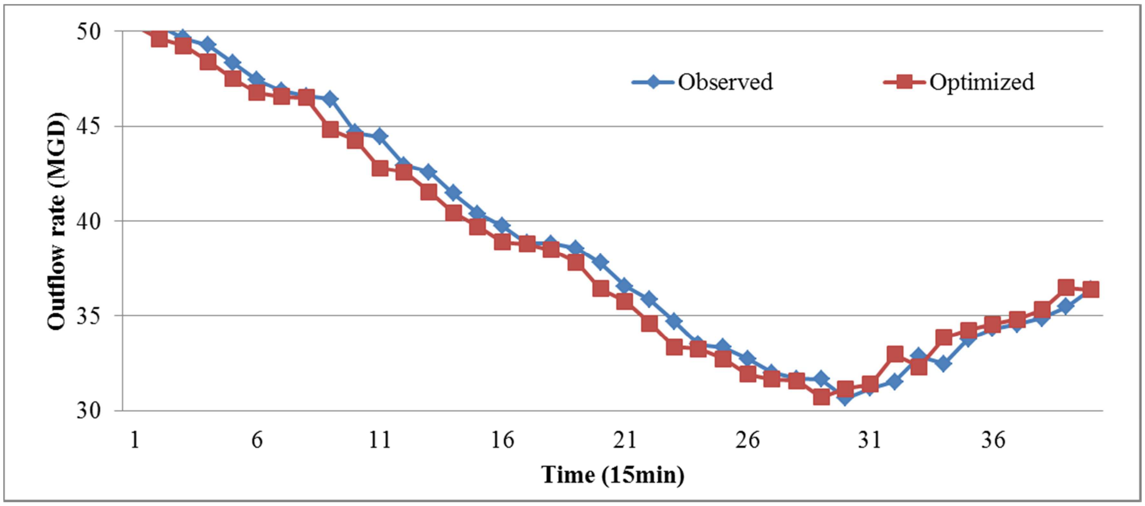


Figure 5.15. Observed and optimized outflow rate for scenario 3

The optimized results for three scenarios are summarized in Table 4. It can be seen that the wet level and outflow rate have little changes before and after optimization. But huge energy reduction can be achieved for all scenarios, where scenario 2 with medium inflow rate has the biggest energy decrease. The results reveal that the data-driven approach is capable to optimize the wastewater pumping process and reduce the energy cost while keeping stably operation of the following process.

Table 5.4. Pumping process optimization results

	Energy consumption (%)	Wet well level (%)	Outflow rate (%)
Scenario 1	-26.9	-1.1	0
Scenario 2	-31.6	-0.8	0
Scenario 3	-18.2	1.5	-1.0

CHAPTER 6

ENERGY EFFICIENCY OPTIMIZAITON OF THE ACTIVATED SLUDGE PROCESS

6.1 Introduction

The activated sludge process (ASP) is one of the key processes in wastewater treatment plants (WWTPs) [85]. Microorganisms in the recycled sludge feed on organic matter in the wastewater, and they grow to form flocs that clump together and settle to the bottom of the final clarifier, leaving a relatively clear, liquid-free organic material and suspended solids [86].

In the activated sludge process, a large amount of air is injected into an aeration basin to provide oxygen for growing microorganisms and to keep solids in suspension [87]. The air is usually provided by high volume blowers or low pressure compressors through a diffused aeration system. These various components of the activated sludge process account for 45% to 60% of the WWTP's total energy consumption.

Reduction of the energy consumption of the activated sludge process is of interest to researchers and engineers [88-94]. A traditional approach for reducing energy consumption was based on improving the design of the aeration system. Using fine-pore diffusers instead of coarse bubble or surface aerators, the oxygen transfer rate can be improved significantly [95]. Appropriate sizing and selection of aeration equipment may impact energy use. Since blower power is a cubic function of airflow, it is important to select properly-sized blowers supplying sufficient, but not excess, air to the aeration basin.

An alternative way to improve the efficiency of the activated sludge process is to implement process control systems [96-100]. Either excessive or inadequate air in the activated sludge process may lead to energy waste and operational problems. Excessive air can result in energy waste and an increased sludge volume index, which may be harmful to the sludge. Inadequate air can cause sludge settling problems as microorganisms die due to insufficient oxygen. Through proper process control, the activated sludge process can meet the effluent requirements while minimizing energy consumption.

Dissolved oxygen (DO) is a crucial control variable that indicates the amount of oxygen in the wastewater. It implies how much airflow is required by the process. In practice, the DO concentration can be controlled between 1.0 and 6.5 mg/L. Various controllers have been developed to control DO concentration in the wastewater [101-103]. Piotrowski et al. [104] proposed a hierarchical controller for tracking the DO reference trajectory in activated sludge processes. A non-linear model predictive control algorithm was used to design the controller. The simulation showed promising results in energy reduction and robust DO tracking. DO is useful for controlling the activated sludge process, but the proper determination of DO concentration over time is complicated by the existence of pollutants in the wastewater and by the fact that the influent flow rate and other uncontrollable variables, such as temperature, are interrelated and change over time.

In this research, a data-driven approach is presented for the optimization of energy efficiency in the activated sludge process. The air required by the process is optimized by an evolutionary computation algorithm by controlling the DO concentration as identified by a data-mining algorithm. Data-mining algorithms determine relationships between input and output variables based on the process data. The data-derived model is solved with an evolutionary computational algorithm to produce optimized values of the control variables. Data-driven approaches have been used successfully in engineering and industry [105-108] when physics-based models could not be used.

This work is organized as follows. Section 6.2 describes the data used in this research and details of our data processing techniques. In Section 6.3, neural network algorithm is used to design and validate the models of airflow rate and effluent pollutant concentrations. In Section 6.4.1, Strength Pareto Evolutionary Algorithm 2 (SPEA2) is used to optimize the activated sludge process to determine the optimal concentration of DO. Section 6.4.2 formulates the optimization problem of the activated sludge process. Section 6.4.3 presents the optimization results of three scenarios representing a tradeoff between energy consumption and quality of the effluent. Two control strategies, i.e., 1) varying the DO concentration hourly and 2) keeping the DO concentration constant daily, were implemented and compared.

6.2 Data description

The Des Moines Wastewater Reclamation Authority (WRF) operates a 97-million-gallon-per-day (MGD) regional wastewater treatment plant in southeast Des Moines, Iowa. The peak influent flow rate can be as high as 200 MGD. The plant was constructed in the mid-1980s mainly to treat municipal wastewater and storm water from the greater Des Moines metropolitan area. The activated sludge process is used to remove organic compounds from the wastewater via biological processes.

A flow diagram of the wastewater treatment process that the WRF uses at the regional plant is shown in Figure 6.1. The collected wastewater enters the plant and passes through five bar screens. Large items, such as rags and sticks, are screened out for later disposal. After screening, the influent wastewater enters a wet well and then is pumped to primary clarifiers. After a retention time of 1 to 2 h, scum floats to the surface where it is removed by a skimmer. Then, the wastewater is delivered by intermediate pumps to six adjacent aeration tanks, each of which is divided into four basins. Each basin is 300 feet long, 35 feet wide, and 20.9 feet deep. Process air is provided by single-stage centrifugal blowers with maximum capacity of 35,000 scfm. During normal operations, a required quantity of the sludge from the secondary clarifiers, called returned activated sludge (RSL), enters the aeration tanks through sludge pumps. When the RSL and the wastewater are mixed, microorganisms in the activated sludge use oxygen provided by the fine bubble diffusers located on the bottom of the aeration basins to break down the organic matter. The remaining sludge from the secondary clarifiers and the sludge from the primary clarifiers are pumped to the anaerobic digesters to produce biogas. The liquid from the secondary clarifiers flows to the chlorine contact tanks where chlorine is injected to kill most bacteria. The final effluent is discharged to the river.

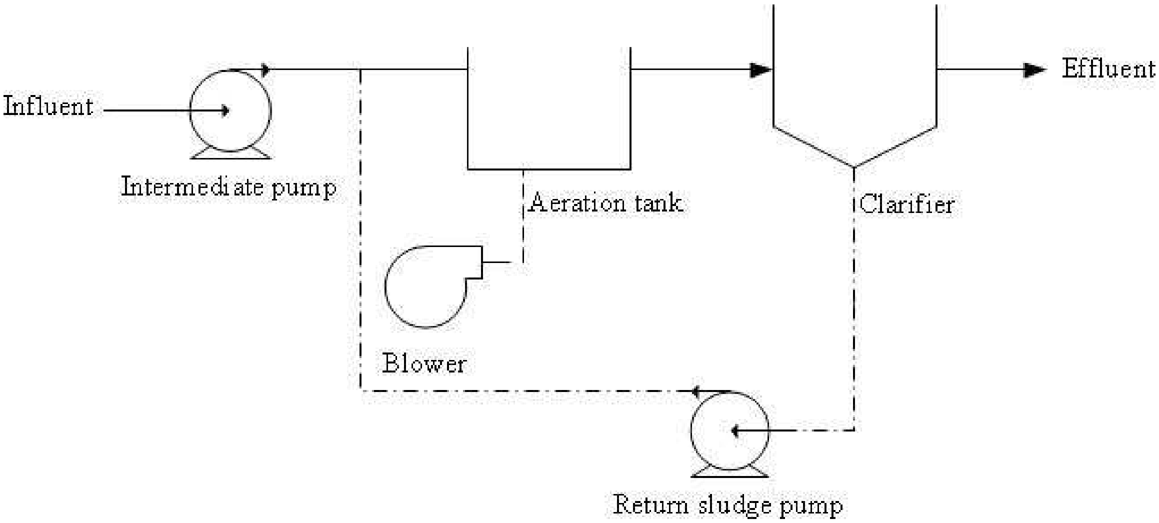


Figure 6.1. Flow diagram of the activated sludge process

The data used in this research were collected from 1/1/2010 through 12/31/2010. Since many variables are involved in the activated sludge process, a boosting tree algorithm [109] was used to evaluate the relative importance of the process variables. Hourly and daily data were collected. The hourly data included the influent flow rate, returned sludge flow rate, DO concentration, and airflow rate. The daily data (measured three times a week) included the carbonaceous biochemical oxygen demand (CBOD) in the effluent, the total suspended solids (TSS) in the effluent, and the temperature and pH in the aeration tank. These data were used to develop the models of the airflow rate and effluent pollutant concentrations by a data-mining algorithm. Errors and outliers in the dataset were removed to improve the accuracy of the models. The final dataset was divided into a training dataset and a test dataset. The two datasets are described in Table 6.1.

Table 6.1. Description of the datasets

Dataset	Description	Observations	Time Period
1	Model training dataset: Building prediction models	Hourly 4368 data points Daily 90 data points	1/1/2010 through 8/25/2010
2	Model test dataset: Testing prediction model and optimization	Hourly 2544 data points Daily 50 data points	8/26/2010 through 12/31/2010

The airflow rate of the ASP provides a measure of the energy consumed, which was one of the objectives of this study. In the model that predicted the airflow rate, DO concentration was a controllable variable. The flow rates of the influent and the returned sludge were uncontrollable variables. With less air flowing into aeration tanks, the quality of the effluent was degraded, which was a matter of concern because it is desirable to maximize the quality of the effluent to meet federal and state requirements. Since CBOD and TSS in the effluent reflect the quality of the effluent, the objective can be transformed to minimize the concentrations of CBOD and TSS in the effluent, thus CBOD and TSS models were built by data-mining algorithm. Temperature and pH values are uncontrollable variables that affect the quality of the effluent. All variables used in this research are presented in Table 6.2.

Table 6. 2. Variables and their units

Variable	Description	Unit
u_1	Influent flow rate	MGD
u_2	Returned sludge flow	MGD
u_3	Temperature	°C
u_4	pH	
x	Dissolved oxygen	mg/L
y_1	Airflow rate	scfm
y_2	Effluent CBOD	mg/L
y_3	Effluent TSS	mg/L

A multi-objective model that minimized the airflow rate, y_1, the effluent COBD, y_2, and the effluent TSS, y_3, was formulated in (1). Since the CBOD and TSS of the effluent were daily data, the hourly data of influent flow rate, returned sludge flow rate, and the concentration of DO was averaged to daily values. The constraints used in the model are discussed in Section 6.4.

$$\begin{aligned} &\min(y_1, y_2, y_3) \\ \text{where:}\quad & y_1 = f_1(u_1, u_2, x) \\ & y_2 = f_2(u_{1\_ave}, u_{2\_ave}, u_3, u_4, x_{ave}) \\ & y_3 = f_3(u_{1\_ave}, u_{2\_ave}, u_3, u_4, x_{ave}) \end{aligned} \tag{6.1}$$

6.3 Model building and validating

Neural network (NN) algorithm was used to build the models of the functions in (6.1). The NN algorithm captures and represents the non-linear and complex relationships between the input and output variables [110]. In this research, a supervised, back-propagation neural network was used. The input data was presented to the network repeatedly. With each presentation, the final output was improved by calculating the error between the output and the desired output, and the result was used as feedback to the network to adjust the weights. The complex and non-linear relationship between input and output can be modeled with good accuracy [111]. A graphical representation of a neural network is shown in Figure 6.2.

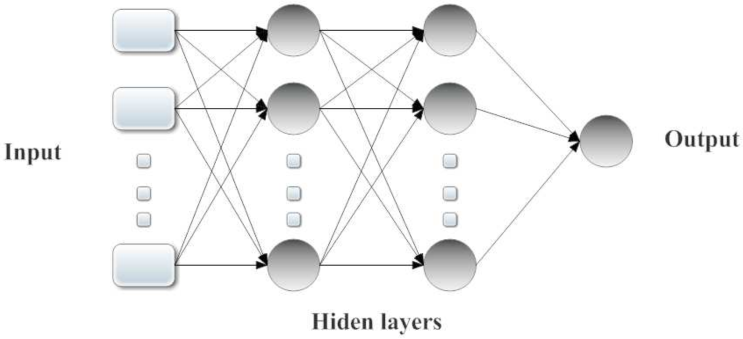


Figure 6. 2. Block diagram of a neural network

Three models, airflow rate, effluent CBOD, and effluent TSS, were built with neural networks. For each model, 200 neural networks were trained with one hidden layer and the number of neurons between 3 and 30. The Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm [28] was used to minimize the sum of the squares (SOS) cost function in building each network. The iden112y function, logistic sigmoid function, hyperbolic tangent function, negative exponential function, and the standard sine function were selected as hidden and output activation functions. Details of the best performing neural networks are presented in Table 6.3.

Table 6.3. Multiple layer perceptron neural networks

Model	NN Structure	Training Algorithm	Error Function	Hidden Activation	Output Activation
Airflow rate	3-9-1	BFGS 48	SOS	Tanh	Exponential
Effluent COBD	5-21-1	BFGS 38	SOS	Logistic	Exponential
Effluent TSS	5-8-1	BFGS 15	SOS	Tanh	Tanh

Figures 6.3 through 6. 5 compare the first 50 observed values (from dataset 2) and the predicted values of the airflow rate, the effluent CBOD, and the effluent TSS concentration. The trend of airflow rate was predicted well, and the predicted values and the observed values are close to each other. The models derived from the hourly data were accurate. Most predicted values of the effluent CBOD were greater than the observed values. Since most of the collected data were around 4 mg/L, sharp peaks can cause significant errors in the NN model. The pattern of the effluent TSS concentration was predicted well, and gaps existed between the observed and predicted values for the low values of the COBD. However, the prediction error for the peak values was small, as shown in Figure 6.5. Due to the fact that the maximum pollutant concentration in the effluent is regulated, this model was useful because of its accurate predictions of peak values. The accuracies of the predicted results are presented in Table 6.4. The MAE, MSE, and SD of the airflow model were large because the observed values were large. The metrics indicated that the accuracy of the prediction of effluent CBOD was better than the accuracy of the prediction of effluent TSS.

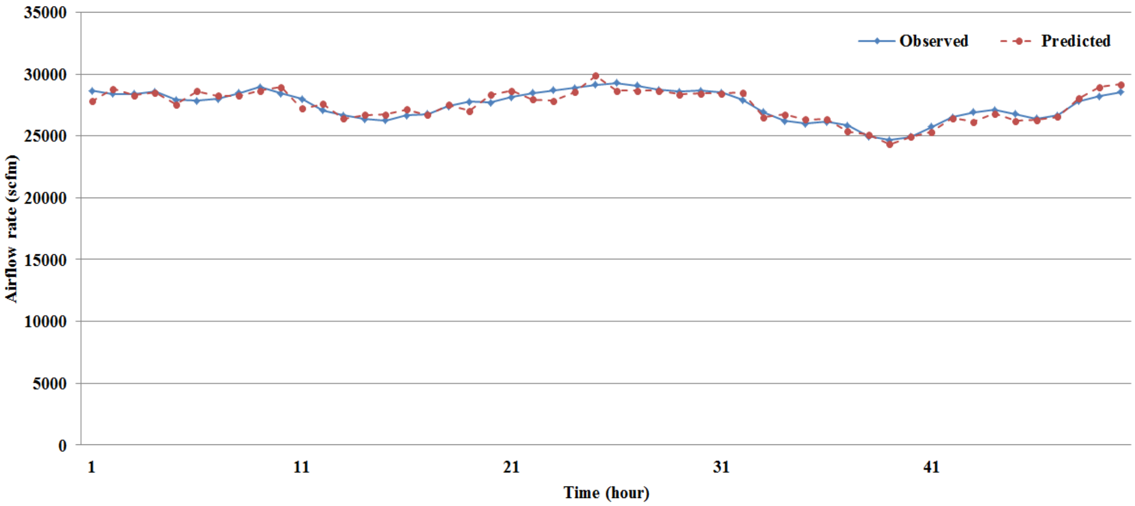


Figure 6.3. Airflow rates observed and predicted by the neural network model

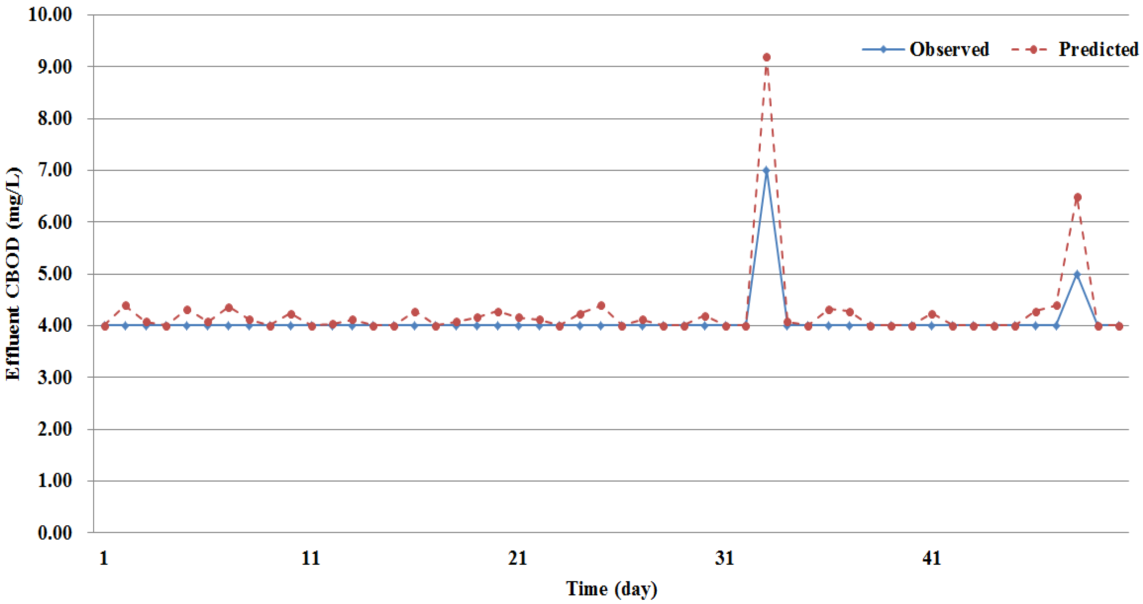


Figure 6.4. Effluent CBOD concentrations observed and predicted by the neural network model

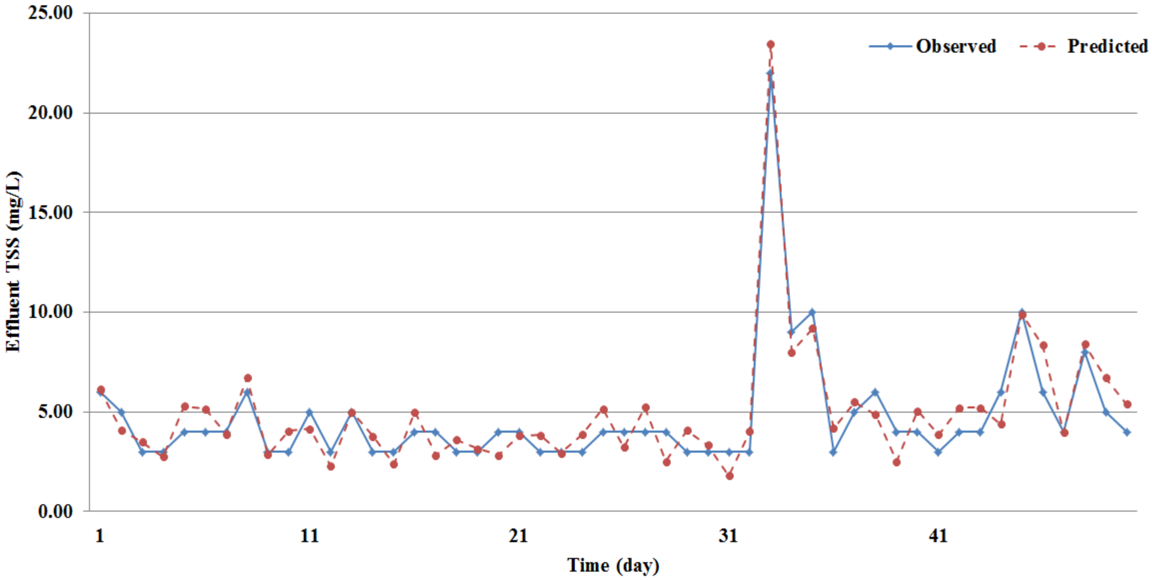


Figure 6.5. Effluent TSS concentrations observed and predicted by the neural network model

Table 6.4. Accuracies of the predictions of the three models

	MAE	MSE	SD
Airflow rate	398.12 scfm	215,087 $scfm^2$	240.30 scfm
Effluent CBOD	0.18 mg/L	0.17 mg^2/L^2	0.38 mg/L
Effluent TSS	0.86 mg/L	0.99 mg^2/L^2	0.51 mg/L

6.4 Multi-objective optimization

6.4.1 SPEA 2 optimization algorithm

In this work, Strength Pareto Evolutionary Algorithm 2 (SPEA2) was used to solve the multi-objective optimization problem formulated in (1). The SPEA2 proposed by Zitzler and Thiele [113] has been used extensively in optimization due to its improved performance. The pseudo code of the algorithm SPEA2 is shown next.

```
SPEA 2 Algorithm
1: Initialize population P
2: Create empty external archive E = Ø; Set t=0
3: for t =0 to max iteration do
4: Calculate fitness values of individuals in $P_t$ and $E_t$
5: Copy all non-dominated individuals in in $P_t$ and Et to $E_{t+1}$
6: Use the truncation operator to remove elements from $E_{t+1}$ when the capacity of the file has been exceeded
7: If the capacity of E' has not been exceeded, use dominated individuals in $P_t$ and $E_t$ to fill $E_{t+1}$
8: If t ≥ max iteration or some other stopping criterion is satisfied, set output
```

```
to the set of decision vectors represented by the non-dominated individuals in $E_{t+1}$. Stop.
9: Perform binary tournament selection with replacement on $E_{t+1}$ to fill the mating pool
10: Apply crossover and mutation to the mating pool. Increase $t = t + 1$
11: End for
12: End procedure
```

The control parameters of the SPEA2 algorithm must be tuned for it to achieve its best performance. In this research, the population size P was set to 100. The ratio between the archive size and the population size was set to 0.5 to provide an adequate selection pressure for the elite solutions. The mutation operator was selected as 0.5 for global search and 0.1 for local search. The maximum iteration number was set to 200.

6.4.2 Problem formulation

The variables of the model (6.1) were constrained according to the operational practice at WRF. The lower and upper limits of the DO concentration were 1.0 and 6.5 mg/L, respectively. To safely operate the activated sludge process and avoid the death of microorganisms due to insufficient oxygen, a DO concentration of 2.0 mg/L, a value that is maintained in most activated sludge processes, was used as the lower limit. The airflow rate was constrained to the range of 18,000 to 50,000 scfm, and the maximum limits of the effluent CBOD and TSS were 25 mg/L and 30 mg/L, in compliance with environmental regulations.

The number of possible solutions to a multi-objective optimization problem could be infinite. A set of these non-dominated solutions forms the Pareto front. Each solution on the Pareto front represents an optimal solution with different trade-offs in satisfying different objectives. In this research, the weighted sum approach was used to consider three optimization scenarios involving different scenarios of energy savings and effluent quality. The final multi-objective optimization model is formulated in (6.2).

$$\min(w_1 y_1 + w_2 y_2 + w_3 y_3)$$

subject to:

$$2.0 \le x \le 6.5$$

$$18000 \le y_1 \le 50000$$

$$y_2 \le 25$$

$$y_3 \le 30 \qquad (6.2)$$

where w_1, w_2, and w_3 are the weights for objectives y_1, y_2 and y_3, respectively. The values assigned to the weights and the meanings of all of the scenarios are presented in Table 6.5.

Table 6.5. Description of three optimization scenarios

Scenario	Energy Weight	Effluent CBOD Weight	Effluent TSS Weight	Description
1	$w_1 = 1$	$w_2 = 0$	$w_3 = 0$	Energy savings preferred
2	$w_1 = 0.5$	$w_2 = 0.25$	$w_3 = 0.25$	Equal importance of energy savings and effluent quality
3	$w_1 = 0$	$w_2 = 0.5$	$w_3 = 0.5$	Effluent quality preferred

6.4.3 Results and discussion

To solve the multi-objective optimization model in (6.2), two control strategies were tested in this research. Strategy A maintained the hourly DO concentration constant for a day. This was beneficial for older WWTPs equipped with constant speed blowers. By adjusting the arrangement and the number of blowers, the total airflow rate to the aeration tanks changes. In newer WWTPs, variable frequency drives are used to control the blowers. Strategy B was suitable for applications in which hourly DO values can be controlled. In strategy B, the blowers were adjusted hourly based on the hourly computed DO concentrations.

To reduce the computation time when solving the multi-objective optimization model, data points collected over a seven-day period were considered in this work, i.e., 168 data points for the airflow rate and DO concentration, and seven data points for the effluent CBOD and TSS concentration. Figures 6.6 through 6.9 and Figures 6.10 through 6.13 illustrate the optimization results for Scenario 1 with Strategies A and B, respectively. It can be seen that the airflow rate was reduced significantly by controlling the optimal setting of the DO concentration at a constant value or at hourly-adjustable values. In this scenario, the DO concentration was optimized as low as possible with preference given to saving energy. Some DO concentrations with Strategy B were optimized to the lower bound, i.e., 2.0 mg/L. On the other hand, the concentrations of CBOD and TSS in the effluent pollutant reach high values, approaching their upper limits (as shown in Figures 6.9 and 6.13).

The optimization results of Scenario 2 with Strategy B are shown in Figures 6.14 through 6.17. The optimized airflow rate also was reduced significantly compared to the original observations. The lower optimal DO concentration was close, but it did not reach the defined lower limit (shown in Figure 6.15), which leaves extra room for operational safety. The resulting concentrations of CBOD and TSS increased in the effluent, but the peaks are not as high as in Scenario 1.

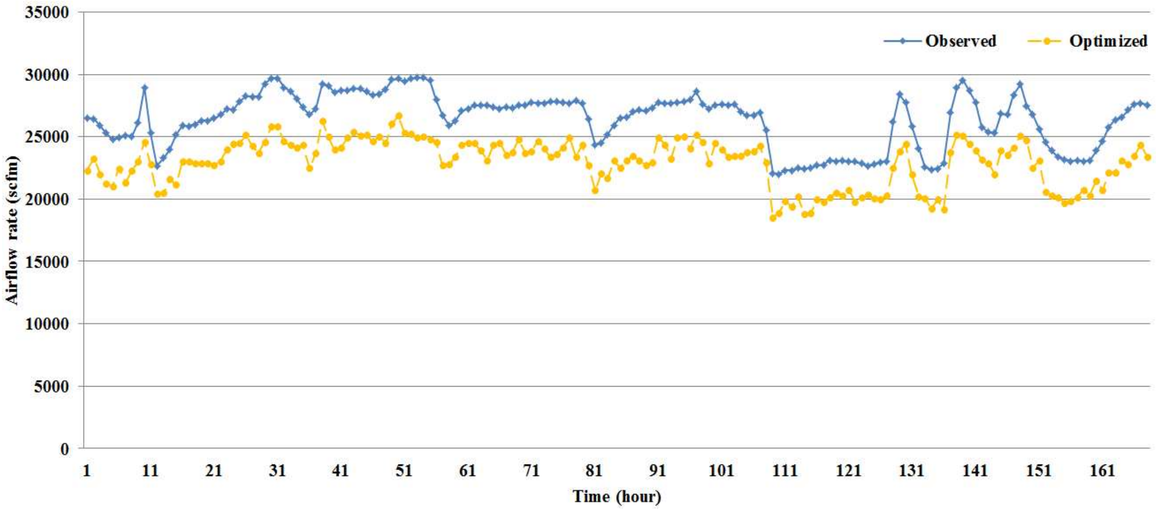


Figure 6.6. Observed and optimized airflow rates for Scenario 1 of Strategy A

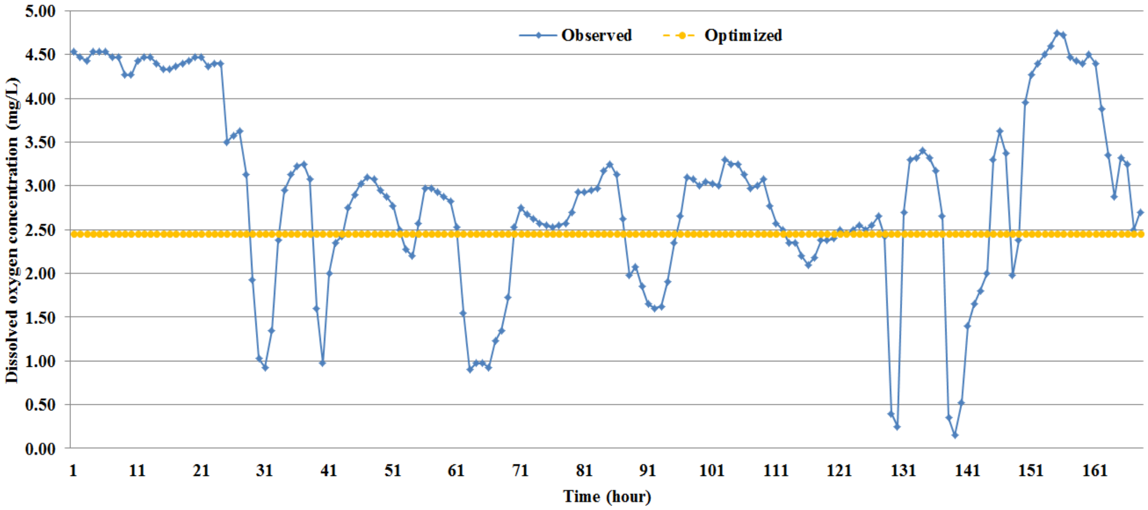


Figure 6.7. Observed and optimized DO concentrations for Scenario1 of Strategy A

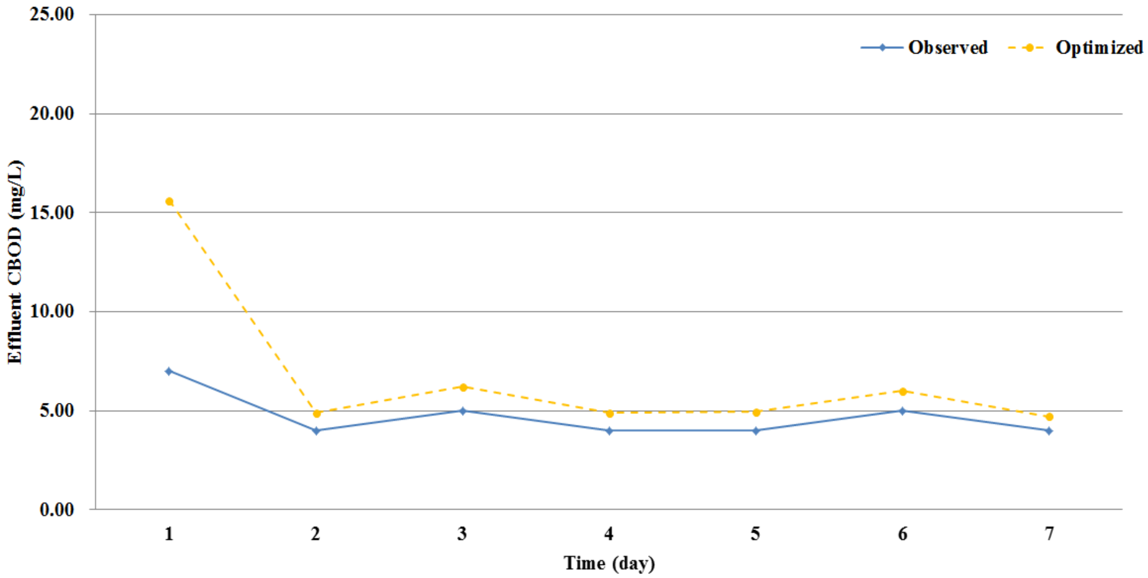


Figure 6.8. Observed and optimized effluent CBOD concentrations for Scenario 1 of Strategy A

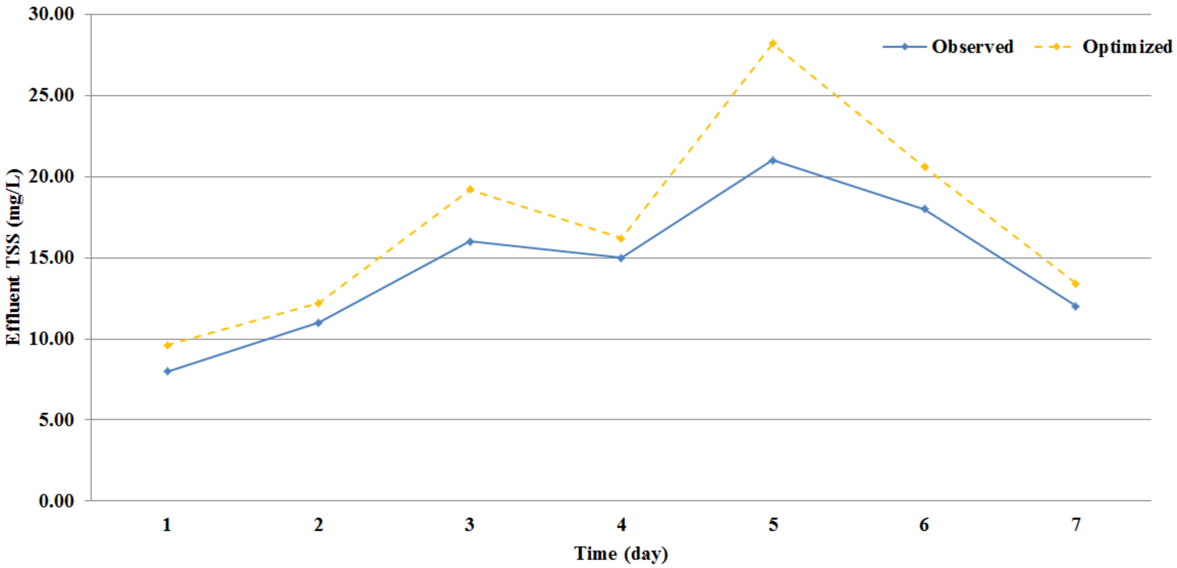


Figure 6.9. Observed and optimized effluent TSS concentrations for Scenario 1 of Strategy A

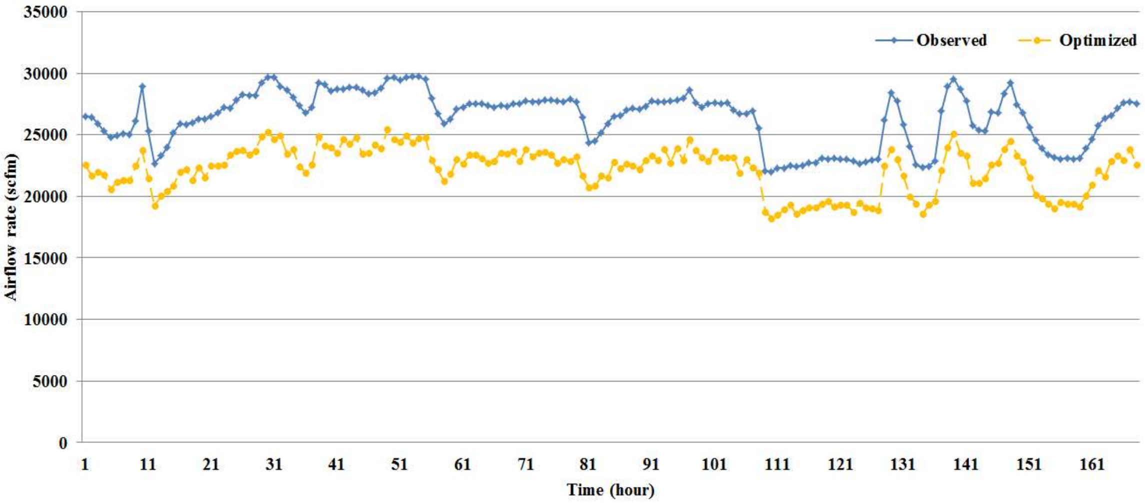


Figure 6.10. Observed and optimized airflow rates for Scenario 1 of Strategy B

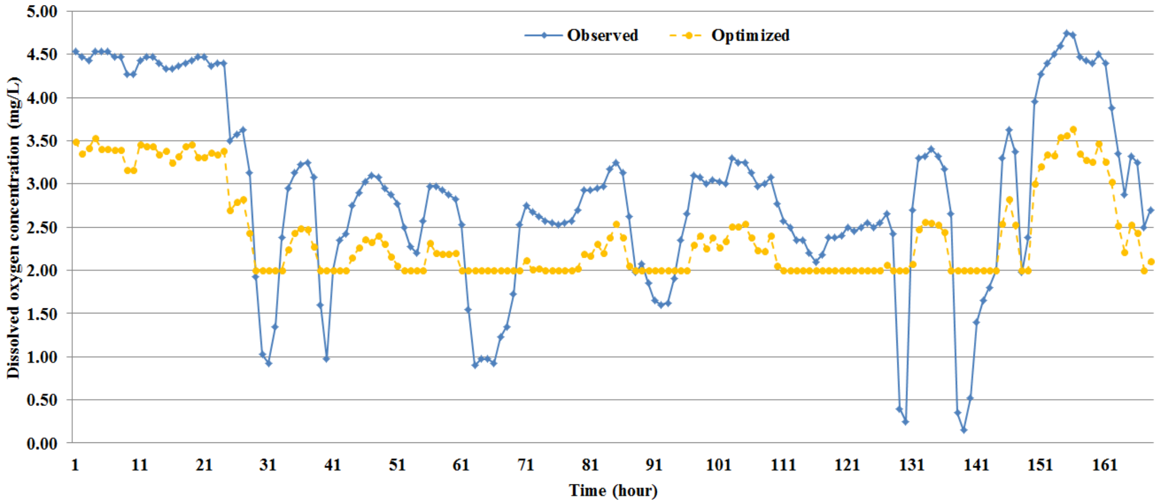


Figure 6.11. Observed and optimized DO concentrations for Scenario 1 of Strategy B

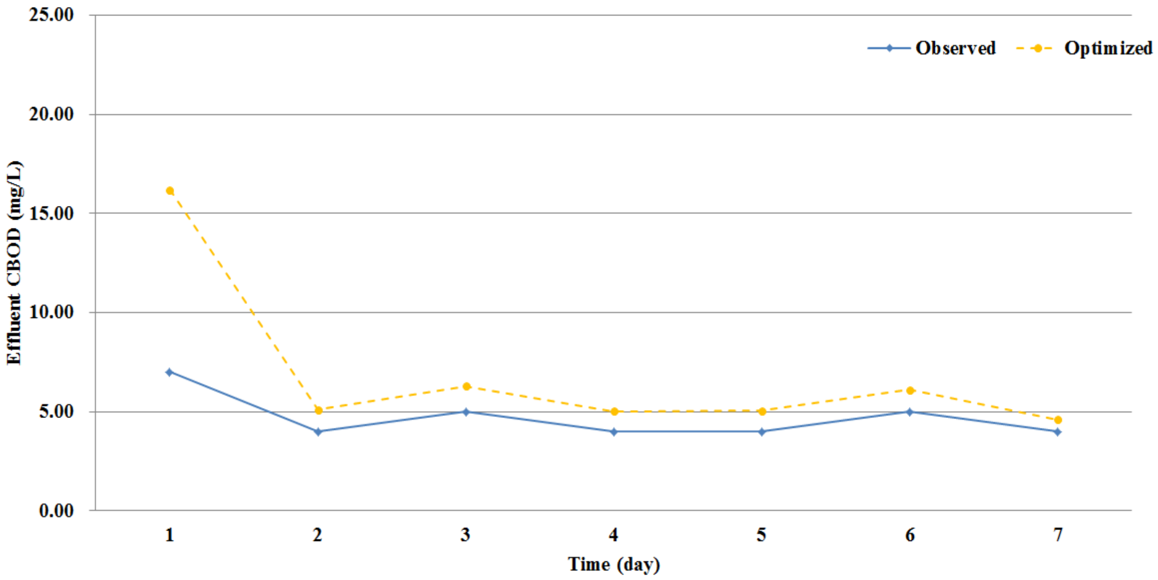


Figure 6.12. Observed and optimized effluent CBOD concentrations for Scenario 1 of Strategy B

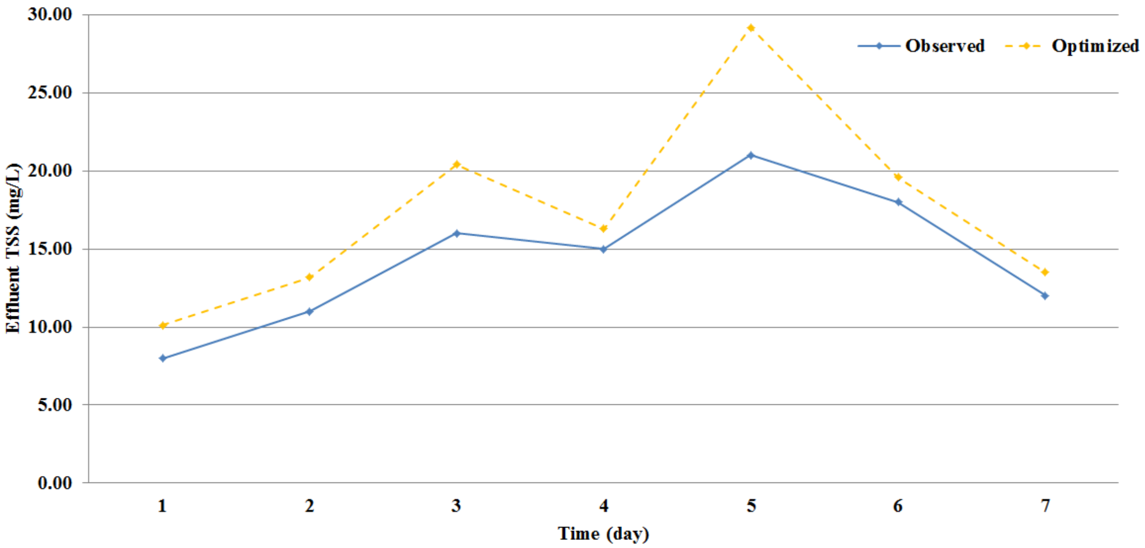


Figure 6.13. Observed and optimized effluent TSS concentrations for Scenario 1 of Strategy B

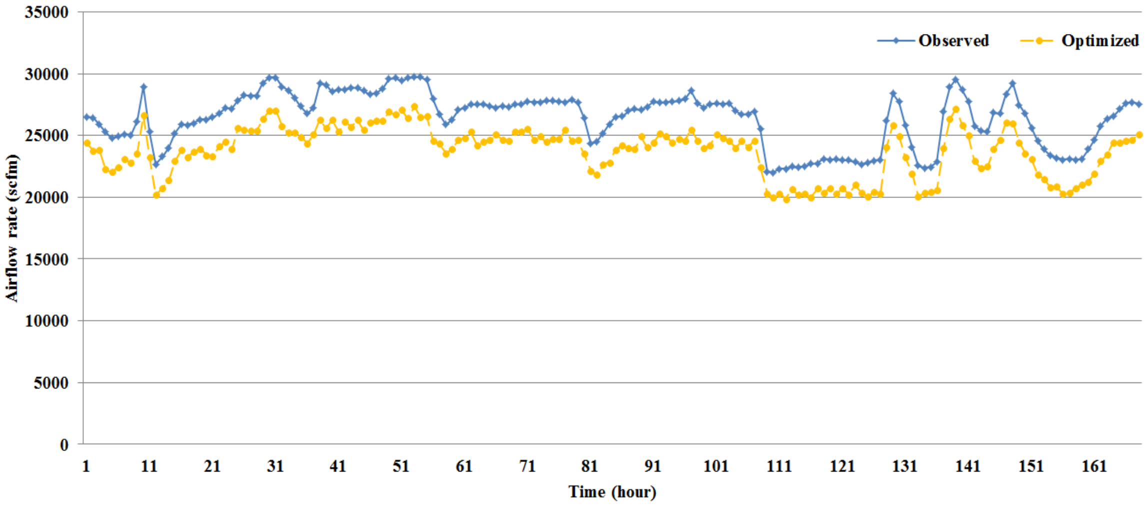


Figure 6.14. Observed and optimized airflow rates for Scenario 2 of Strategy B

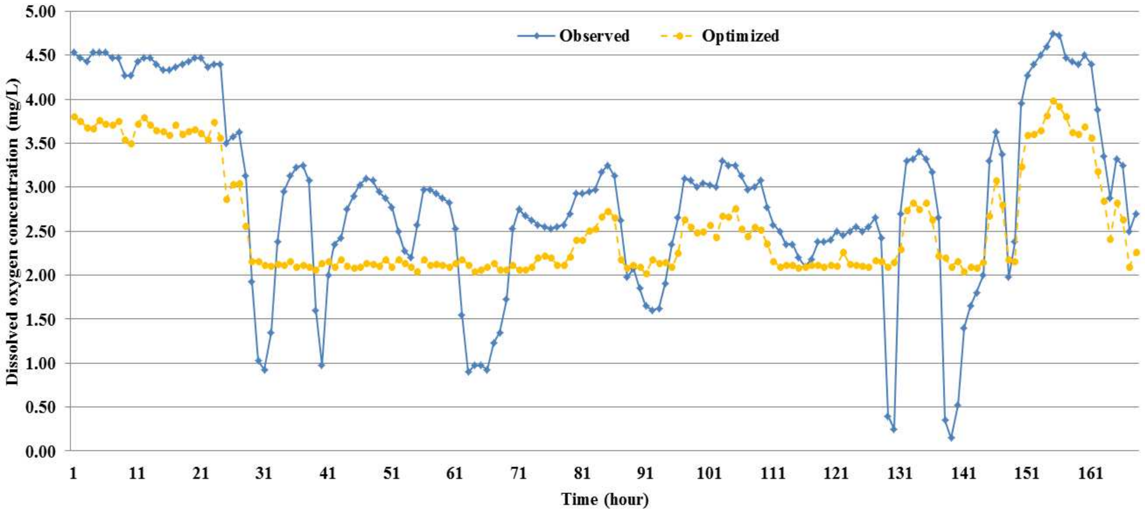


Figure 6.15. Observed and optimized DO concentrations for Scenario 2 of Strategy B

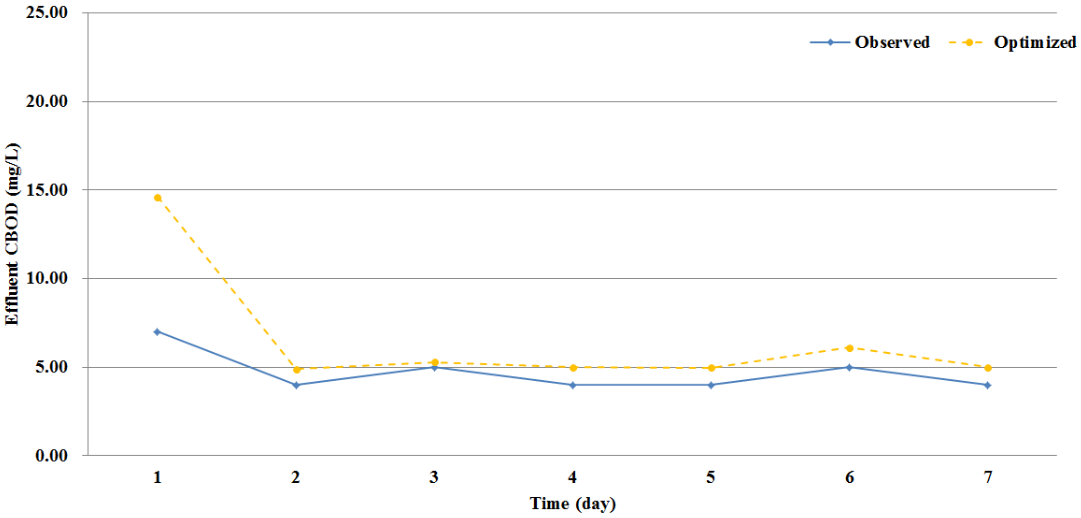


Figure 6.16. Observed and optimized effluent CBOD concentrations for Scenario 2 of Strategy B

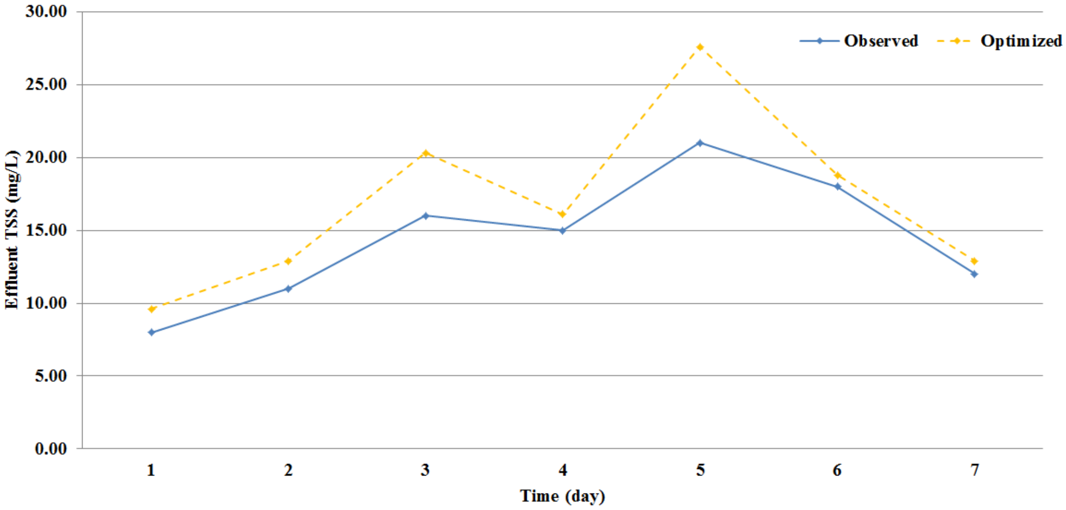


Figure 6.17. Observed and optimized effluent TSS concentrations for Scenario 2 of Strategy B

The airflow rate changes for the three scenarios are presented in Table 6.6. In general, Scenario 1 saved more energy than the other two scenarios when energy savings were preferred. For Scenario 1, the energy savings were 13% and 16% for Strategies A and B, respectively. If the preference were assigned to the effluent quality, it would consume more energy, as shown in Scenario 3. Energy savings of 9% or 10% were achieved when equal importance was assigned to energy saving and effluent quality. In addition, Strategy B had a lower airflow need than Strategy A for all of the different scenarios. It is reasonable that the DO concentration varied over a larger range if it were adjustable hourly as opposed to being a constant throughout the day.

Table 6.6. Reductions in airflow rate requirements for Scenarios 1, 2, and 3

Airflow Rate Reduction	Scenario 1	Scenario 2	Scenario 3
Strategy A	13%	9%	-5%
Strategy B	16%	10%	-7%

Based on the presented results, Scenario 2 with Strategy B is recommended because it saved energy while it maintained effluent quality at an acceptable level. Compared with Scenario 1 in which the use of energy by the activated sludge process was minimized without considering the effluent quality, Scenario 2 took less risk of microorganisms dying because of insufficient airflow or of effluent pollutants exceeding the established upper limits. If the effluent quality was the main concern, Scenario 3, which minimized the pollutant concentration in the effluent, may be considered.

In this work, energy efficiency of the activated sludge process in a wastewater treatment plant was optimized with a data-driven approach. Two objectives were considered, i.e., minimizing the energy use and maximizing the effluent quality. Industrial data were collected at a wastewater treatment plant and used to build the models by neural networks. The model of airflow rate, representing energy use, was built with controllable variable DO concentration and uncontrollable variables, including influent flow rate and return sludge rate. The models of effluent CBOD and TSS concentration, indicating the effluent quality, included the average DO concentration as a controllable variable and influent flow rate, return sludge rate, temperature, and pH values as uncontrollable variables. The model of airflow rate was highly accurate, and the models of the effluent CBOD and TSS had acceptable accuracy. The evolutionary algorithm was used to find the optimal DO concentration that minimized energy use and effluent pollutant concentration.

Two control strategies, constant and hourly variable DO concentrations, were investigated to find the optimal DO concentrations for three different scenarios representing the preference over energy saving or effluent quality. The computational results indicated as much as 16% of the energy used in the process could be saved when preference was given to energy saving. A scenario that gave equal importance to energy saving and effluent quality was recommended to safely operate the activated sludge process. It could save 10% of the energy consumed with hourly-variable, optimal DO concentrations.

The results presented in this work indicated that optimization of DO concentration was a useful approach for optimizing the energy efficiency of the activated sludge process in a wastewater treatment plant. Based on the preference of individual WWTPs, different scenario with different control strategies could be implemented to achieve the desired optimization objective.

CHAPTER 7

OPTIMIZATION OF BIOGAS PRODUCTION PROCESS

7.1 Introduction

Sludge is a byproduct in wastewater treatment process and is suitable for biogas production. Anaerobic digestion of sludge involves a process in which microorganisms break down biodegradable waste in the absence of oxygen [114, 115]. As a result of anaerobic digestion, biogas and carbon dioxide rich biogas are produced. The biogas can then be used to generate electricity or heat.

Biological, chemical, and physical reactions are involved in biogas production process [116]. Due to the complexity of the process building formal models is a challenge. The models presented in literature are usually nonlinear and non-stationary with restrictive assumptions that may not hold in practice [117].

Numerical studies on comprehensive biogas production models for prediction and optimization [118-120] have been reported in the literature. Data-mining algorithms such as neural networks have shown success in building models of the biogas production process. An adaptive neuro-fuzzy inference system was applied for modeling anaerobic digestion of primary sludge in a wastewater treatment plant [121]. The model satisfactorily predicted effluent volatile solid and biogas yield. Holubar et al. [122] used several feed-forward back propagation neural networks to model and subsequently control biogas production in anaerobic digesters. Gas composition, biogas production rate, pH, volatile suspended solids and other parameters were measured and simulated to determine the best feeding profile.

Data mining is a powerful tool to analyze data in scientific and engineering applications, such as bioinformatics, manufacturing, and wind energy [123-126]. Evolutionary computation algorithms are widely used to solve complex, linear and nonlinear optimization problems. Successful applications of evolutionary computation algorithms have been reported in engineering, marketing and science [127-129].

In this work, biogas production is optimized with a data-driven model derived by a multi-layer perceptron (MLP) neural network. Process temperature, total solids, volatile solids, and pH value are selected as controllable variables for the model. The uncontrollable variables include sludge flow rate, organic load, and detention time. To maximize biogas production, particle swarm optimization (PSO) is employed to find optimal solutions for control variables and other uncontrollable variables. The model and the optimization results are then discussed in detail.

7.2 Data description

The data used in this work was collected at the Des Moines Wastewater Reclamation Facility (WRF), located in Des Moines, Iowa. WRF is designed to recycle wastewater from 16 municipalities, counties, and sewer districts.

The WRF includes three complexes to process sludge and produce biogas. Each of the complexes has 2 anaerobic digesters that are 115 ft (35.05 m) in diameter, 29.5 ft (8.99 m) for side water depth and 15 ft (4.57 m) for cone depth. The sludge is delivered to the digesters after being mixed in sludge-blending tanks. The temperature in the digester is kept in a range from 90°F (32.22°C) to 105°F (40.56°C) and is controlled by a sludge heat exchanger. The temperature is generally maintained at 100°F (37.78°C). In the absence of oxygen, microorganisms break down sludge and produce biogas and carbon dioxide, which is first stored in a gas sphere with 141,260 cubic feet (4000 m3) of capacity to meet peak usage demand, and then are delivered to gas generators to produce electricity. The heat generated in the gas combustion is used to maintain the temperature of the sludge heat exchangers and also heat plant buildings in the winter season. A flow chart diagram of the anaerobic digestion at WRF is shown in Figure 7.1.

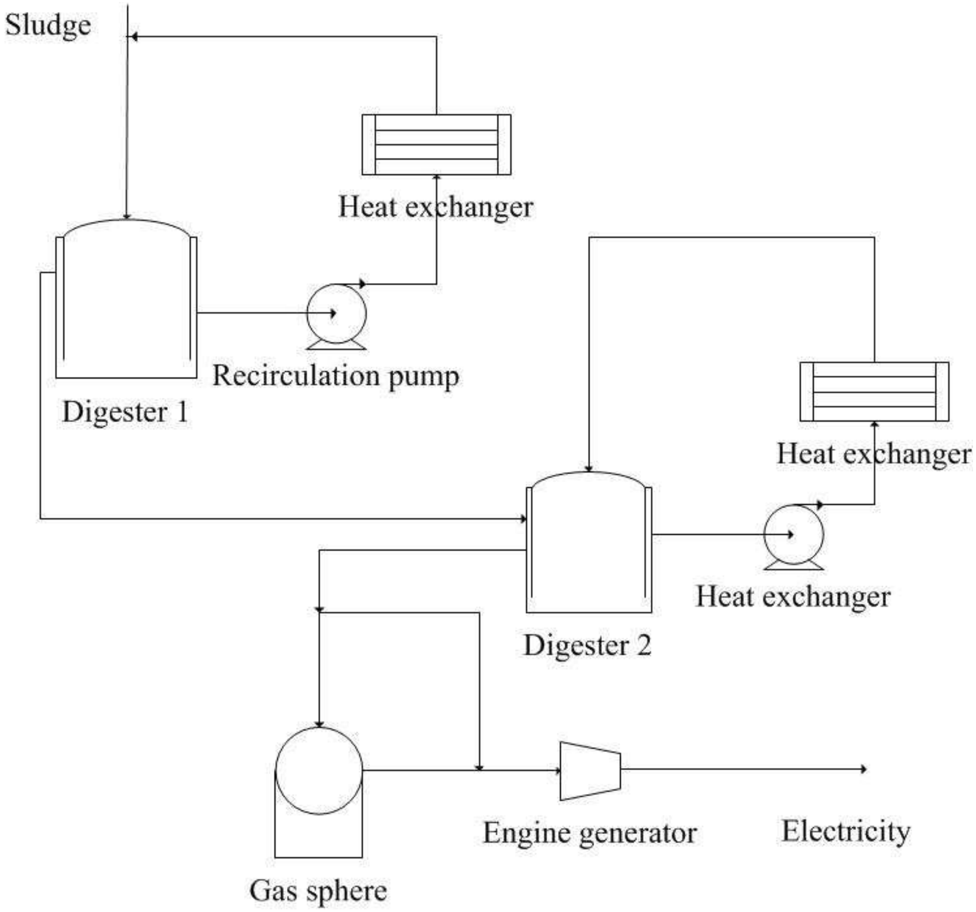


Figure 7.1. Flow chart of anaerobic digestion

The biogas production data used in this work was from daily samples taken over the period from 1/2/2008 to 12/31/2010. As some data points involved errors, e.g., out of range values, the dataset was preprocessed. The processed dataset included 724 data points and it was divided into training and test sets. The training set was from 1/2/2008 to 3/31/2010, and it included 576 data points to build the prediction model. The test set from 4/1/2010 to 12/31/2010 contained 148 data points and it was used to test the developed model. The dataset descriptions are provided in Table 7.1.

Table 7.1. Dataset description

Dataset	Description	No. of instances
1	Training dataset: building prediction model	576 observations
2	Test dataset: testing prediction model	148 observations
	Total dataset: 724 data points	724 observations

Although the original dataset included 11 parameters, some of them were removed because as they did not have obvious influence on the biogas production. In this research, digester temperature, volatile solids, total solids, detention time of sludge, pH value and biogas production were selected to build a prediction model. At WRF, digester temperature and volatile solids fraction are controllable, and other parameters are uncontrollable variables. The list of parameters with their ranges is shown in Table 7.2.

Table 7.2. List of parameters

Variable	Variable Name	Unit
x_1	Temperature	°C
x_2	Total solids	%
x_3	Volatile solids	%
x_4	pH	number
u_1	Sludge flow rate	m^3/day
u_2	Organic load	kg/m^3
u_3	Detention time	day
y	Biogas production	m^3/day

The biogas production model involving the selected parameters is expressed in equation (1).

$$y = f(x_1, x_2, x_3, x_4, u_1, u_2, u_3) \tag{7.1}$$

Temperature is the most important variable affecting the rate of digestion and biogas production. Even though anaerobic microorganism communities can endure temperature ranging from below freezing to above 135°F(57.22°C), they thrive best at temperatures from 68°F (20°C) to 105°F (40.56°C) for mesophilic and 113°F (45°C) to 135°F (57.22°C) for thermophilic. To increase biogas production, the digester must be kept at a consistent temperature, as rapid changes will disturb bacterial activities. This is also the reason why most anaerobic digesters require some level of insulation or heating which will generally increase biogas production in cold seasons.

The total solids in wastewater is another variable influencing biogas production. The concentration of total solids has an impact on the effectiveness of the microorganisms in the decomposition process during anaerobic digestion. Igoni et al. [130] found that biogas production increased when the percentage total solids of waste increased. A statistical analysis showed that the former was a power function of the latter, and there was a point where no further increase in the biogas production would be obtained when the percentage of total solids kept increasing.

The volatile solids in wastewater is measured as the total solids, excluding the ash content, as obtained by complete combustion of the feeding waste. It contains the biodegradable volatile solids and refractory volatile solids. The former is useful in estimation of the biodegradability of the waste and the latter is not easily degraded by microorganisms, so volatile solids concentration affects both biogas production and the composition quality.

Detention time is a critical element in control of anaerobic process. As the biogas formers are slower to grow and are sensitive to changes in the operation conditions, a short detention time results in sludge being washed out of the digester. A longer detention time allows the creation of a buffering alkalinity to form and stabilization of the microorganism environment.

Anaerobic microorganisms, especially methanogens, are sensitive to pH in the digester and their growth can be inhibited by acidic conditions. The pH value for anaerobic digestion usually varies between 5.5 and 8.5. At an early stage of digestion, acetogenesis can lead to accumulation of large amounts of organic acids resulting in an acidic environment with a low pH value. When digestion reaches the late methanogenesis stage, the concentration of ammonia rises and the pH value may exceed 8.

7.3 Model building and validating

The multi-layer perceptron (MLP) neural network involves multiple fully connected layers. Except of the input nodes, each node is a neuron with a nonlinear activation function. MLP utilizes a supervised learning mechanism, called back propagation, for training. MLP is a modification of the standard linear perceptron able to distinguish data that is not linearly separable [131].

Dataset 1 and dataset 2 of Table 7.1 were used to train and test the MLP neural network model. In all, 2000 single hidden-layer neural networks were trained. The number of neurons in the hidden layer varied from 3 to 22. Table 7.3 summarizes the best performing neural networks. To measure the model accuracy, the sum of squared error (SSE) is used in this research. The SSE is the sum of the squared difference between the target and actual output values on each output unit [108]. The Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm was used to minimize SSE while building each neural network model. The iteration number of the BFGS algorithm ranged between 18 and 83 as shown in the 3rd column in Table 7.3. Multiple nonlinear hidden and output activation functions, including identity, logistic, tanh, and exponential, were used for the neurons. The activation functions used in the models are listed in Table 3. The MLP neural networks successfully identify the nonlinear relationship between the process variables as demonstrated with the low validation errors shown in Table 7.3.

Table 7.3. MLP neural networks

MLP Structure	Validation Error	Training Algorithm	Error Function	Hidden Activation	Output Activation
MLP 5-4-1	0.0055	BFGS 44	SSE	Tanh	Logistic
MLP 5-3-1	0.0057	BFGS 32	SSE	Tanh	Logistic
MLP 5-10-1	0.0058	BFGS 83	SSE	Tanh	Identify
MLP 5-11-1	0.0059	BFGS 18	SSE	Logistic	Logistic
MLP 5-22-1	0.0058	BFGS 44	SSE	Tanh	Sine

As illustrated in Figure 7.2, the MLP neural network successfully predicts biogas production based on input variables. It implies MLP neural network learns the relationship between the input variables to the digester and the output biogas production. Most biogas production patterns and peaks are clearly recognized by the model built, except of the data points 70 and 110 with the smallest and the largest values. Prediction accuracy of these two points and that of their neighbors is low, likely due to noise in the data.

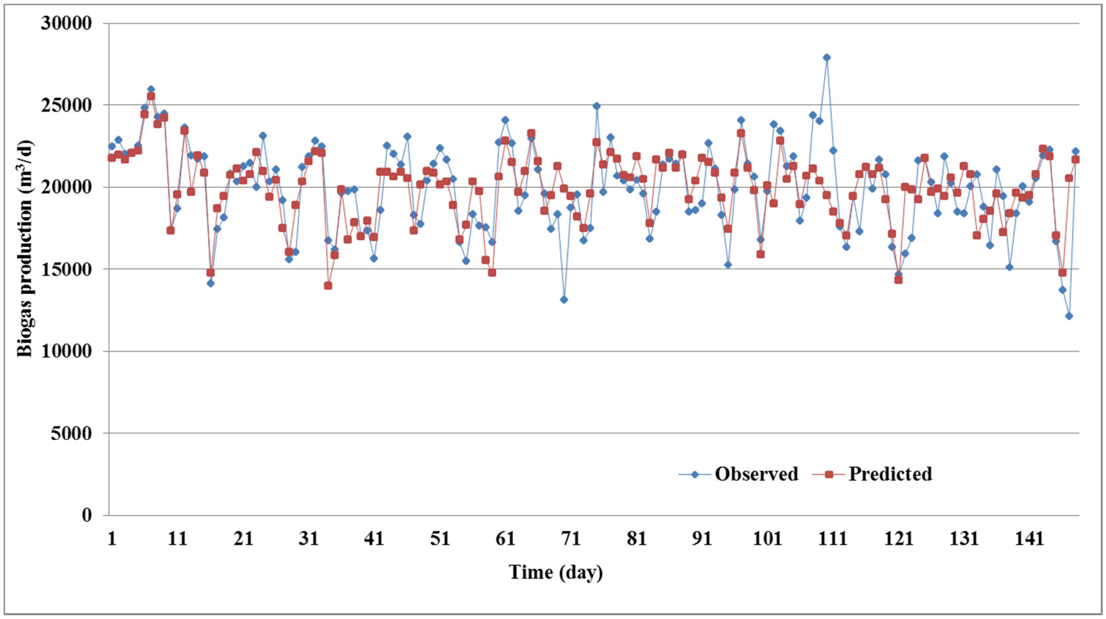


Figure 7.2. Observed and neural network model predicted biogas production

Performance of the MLP neural network derived model has been compared with other four data-mining algorithms, classification and regression tree (C&RT), random forest, k-nearest neighbor (KNN) and support vector machine (SVM). These algorithms can be used for classification and regression analysis. By assigning the proper value for the object to be the average of the values of its k nearest neighbors, KNN can be applied for regression. Training the SVM model on a dataset, the function dependence of the dependent variable on a set of independent variables can be estimated for regression problems. Figure 7.3 to 7.6 show the observed and predicted biogas production by C&RT, random forest, KNN and SVM.

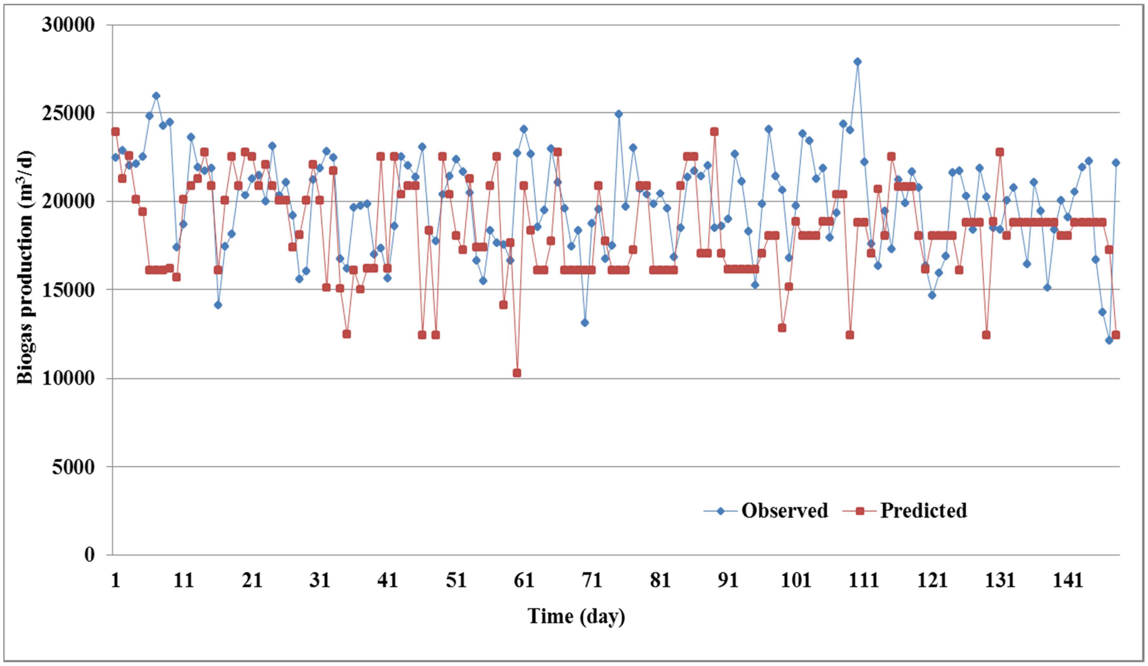


Figure 7.3. Observed and C&RT model predicted biogas production

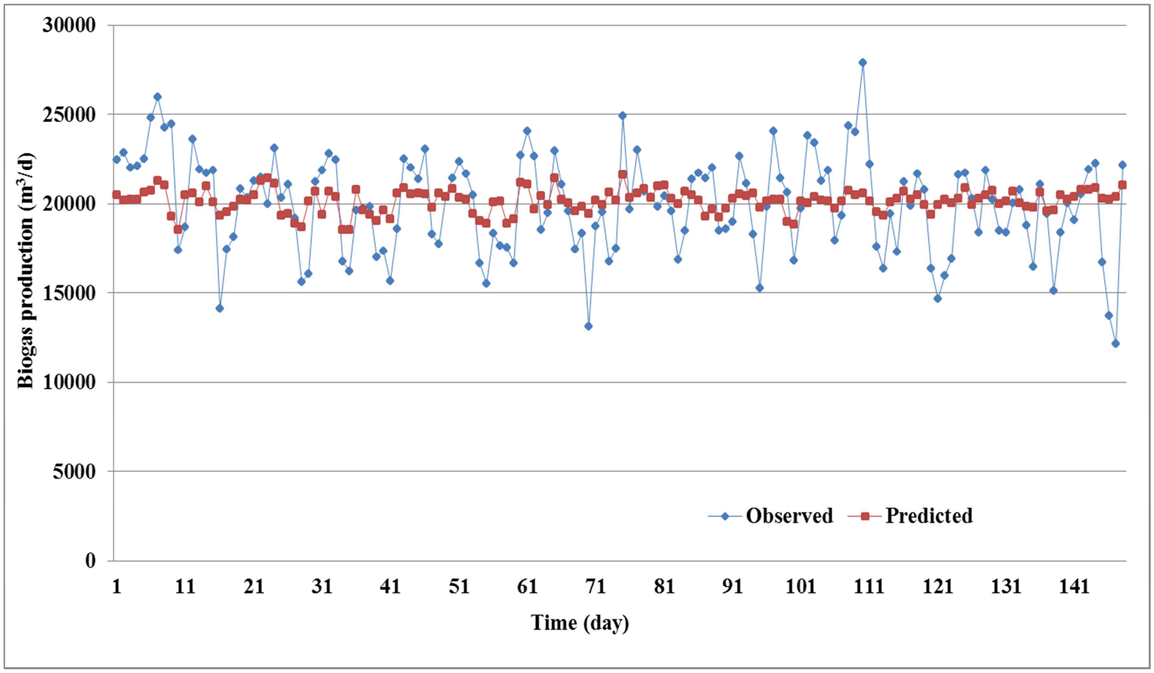


Figure 7.4. Observed and random forest model predicted biogas production

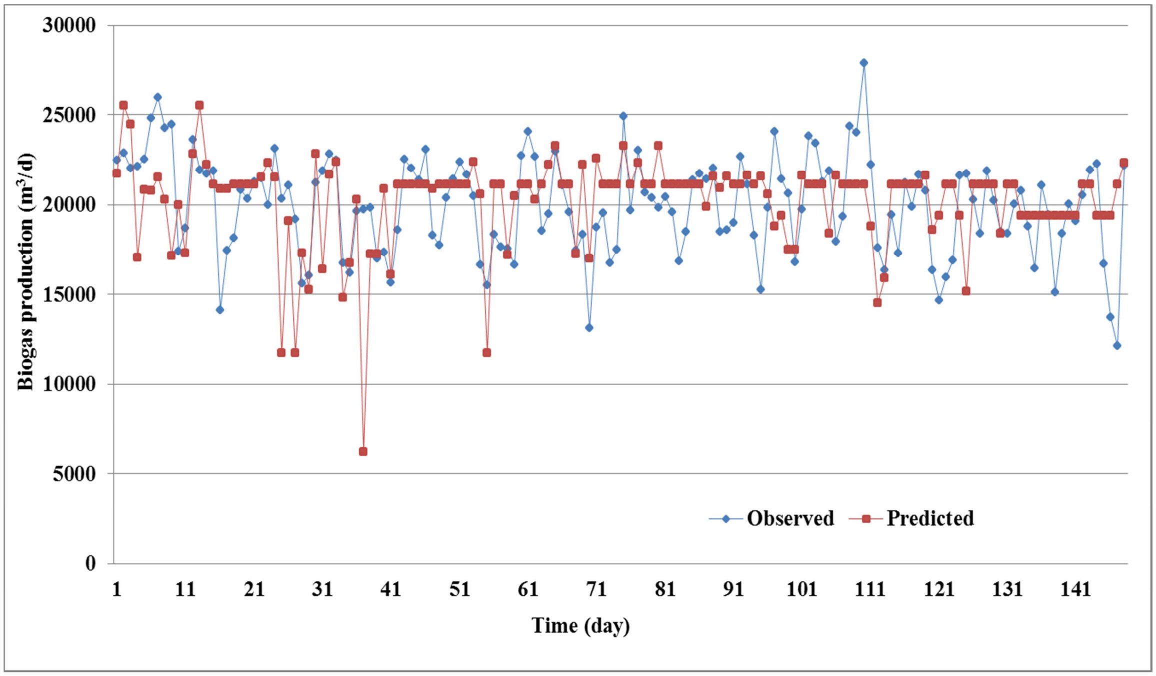


Figure 7.5. Observed and KNN model predicted biogas production

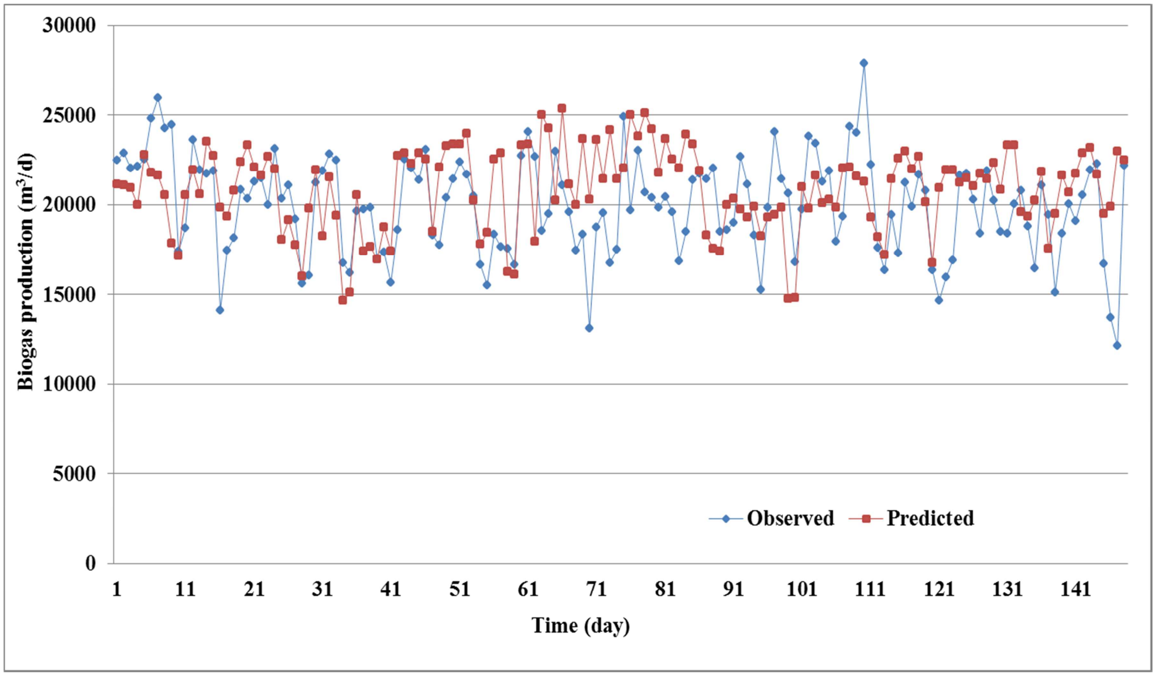


Figure 7.6. Observed and SVM model predicted biogas production

The comparative results in Table 7.4 demonstrate that the prediction model built by the MLP neural network offers better prediction accuracy than the other models. Specifically, the mean absolute percentage error of the constructed model is 0.07. This error expresses the relative accuracy of the model. Fractional bias at 0.00 indicates a satisfactory agreement between the predicted and the observed value. The root mean square error of the model is 68,302, which is a large number. However, the value of the biogas production could be larger than 28,317 m3/d. A relatively small difference between predicted and observed value will cause a large root mean square error. Normalized mean square error expressing the normalized average of the square error is 0.01 for the built model. The index of agreement of 0.99 indicates a high correlation between the predicted and observed values. The direct comparison of PE and FB among all five data-mining algorithms can be seen in Figure 7.7.

Table 7.4. Performance metrics

	PE	FB	RMSE	NMSE	IA
NN	0.07	0.00	68,302	0.01	0.99
C&RT	0.15	0.08	139,378	0.04	0.99
Random forest	0.10	0.01	86,836	0.01	0.99
K-nearest neighbor	0.12	0.01	110,666	0.02	0.99
SVM	0.13	0.04	110,898	0.02	0.99

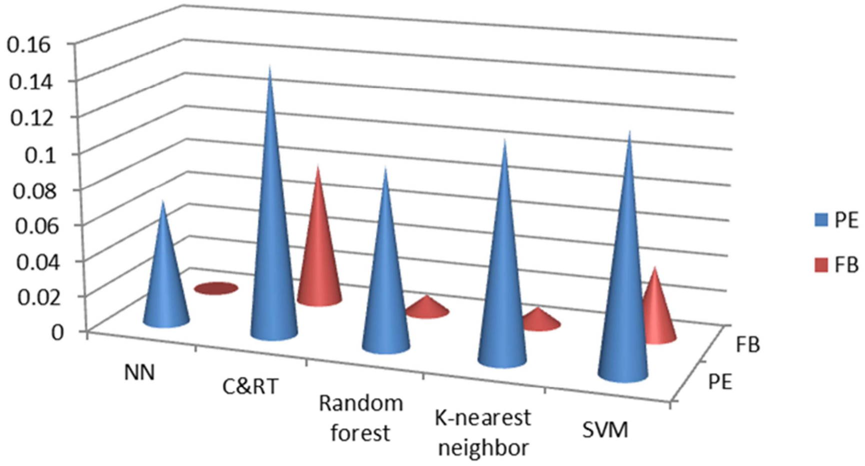


Figure 7.7. Comparison among five algorithms

According to above results and analysis, the MLP neural network model performs better than models built by the remaining four data-mining algorithms. Therefore, MLP neural network has been selected to optimize the biogas production process.

7.4 Optimization of the biogas production

7.4.1 Problem formulation

The model trained by the MLP neural network was used to construct the optimization model. To optimize the biogas production process, the single objective can be expressed as a function of control variables. According to the operation conditions at

WRF, the digester temperature was constrained from 32.22°C to 40.56°C, the total solids was constrained between 2% and 12%, the volatile solid fraction was constrained from 65% to 85%, and the pH value was constrained between 6.8 and 8.0. The single optimization problem can then be presented in (7.2):

$$\max_{x_1,x_2,x_3,x_4} f(x_1,x_2,x_3,x_4,u_1,u_2,u_3)$$

subject to:

$$\begin{aligned} 32.22 \le x_1 &\le 40.56 \\ 2\% \le x_2 &\le 12\% \\ 65\% \le x_3 &\le 85\% \\ 6.8 \le x_4 &\le 8.0 \end{aligned} \tag{7.2}$$

where f is the function in Eq. (7.1) and refers to the model built in Section 3. The descriptions of the seven input variables of model (7.2) are shown in Table 7.1.

Solving the complex biogas production model with mathematical programming algorithms is a challenge. Heuristic search algorithms like greedy search [132], and evolutional algorithms like genetic algorithm [133], are good choices for solving complex models. In this chapter, the standard PSO algorithm was applied in this research to solve model (7.2). The standard PSO algorithm is presented next.

Step 1: Randomly initialize n particle positions $d_i \in R^n$ and velocities $v_i \in R^n$.

Step 2: Evaluate fitness value f_i using current particle positions.

If $f_i \le f_i^b$, then $f_i^b = f_i$, $p_i^b = d_i$

If $f_i \le f^g$, then $f^g = f_i$, $p^g = d_i$

Step 3: Update all particle velocities v_i

$$v_i = v_i + c_1 r_1 (p_i - d_i) + c_2 r_2 (p^g - d_i)$$

Step 4: Update all particle positions d_i

$$d_i = d_i + v_i$$

Step 5: Update fitness value f_i^b and f^g

Step 6: If the stopping condition is satisfied, then f^g is the final optimal solution with the particle position p^g. Otherwise, return to step 3 to start next iteration.

Here, the dimension for each particle's position d_i and velocity v_i is 4. Parameter p_i^b is the best individual particle position, and p^g is the best global position. c_1 and c_2 are cognitive and social parameters, they are set as 2 in this research. r_1 and r_2 are random numbers between 0 and 1. Figure 7.8 shows the flow chart diagram of the PSO algorithm.

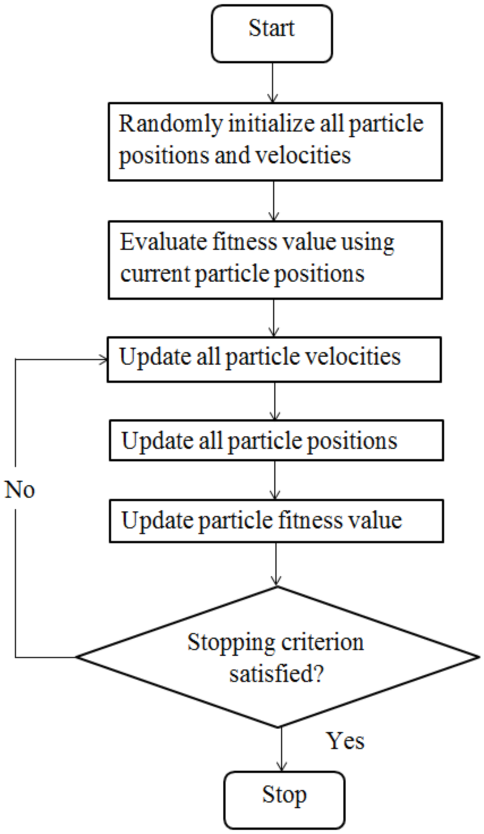


Figure 7.8. Flow chart diagram of the PSO algorithm

7.4.2 Results and discussion

The test set was used to solve model (7.2) with the PSO algorithm. In each iteration, the trained MLP neural network is used to predict the biogas production based on controllable and uncontrollable variables. Then the PSO algorithm determines the best fitness value (here biogas production) by determining the settings of controllable variables. The initial parameters of the PSO algorithm are as follows: the population size is 50 and the maximum number of iterations is set at 20.

To obtain a stable production of biogas, unchanged operational conditions for a period of time is preferable, e.g., a full season. The optimal setting is used for all the time in this period. The optimal value of each controllable variable is first investigated separately, i.e., only one variable is optimized each time. The optimal value of process temperature is found as 39.0 °C (see Figure 7.9 for the optimization results). Under the operational condition in which process temperate is set to 39.0 °C, the biogas production can be improved by 5.3%. The increased biogas production is due to the optimization of controllable setting based on the prediction model in Eq. (7.1). It can be seen that the computed biogas production is usually larger than the observed values. Moreover, the production for the test period shows less variability than the actual values, which have very large differences on a daily basis. The stable output is beneficial for the biogas production process and plant operations.

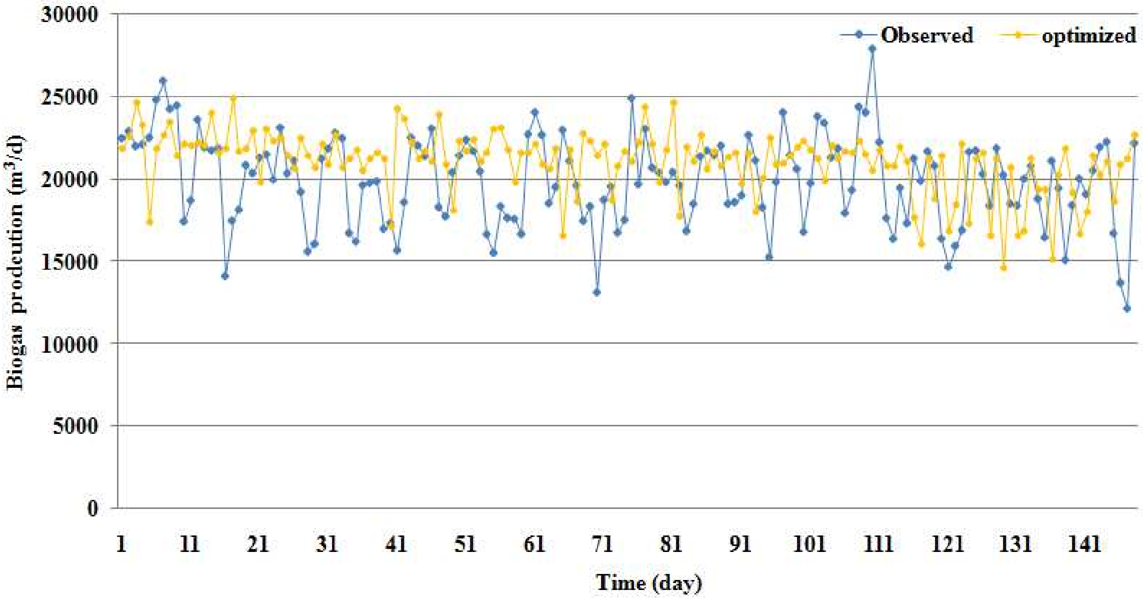


Figure 7.9. Observed and optimized biogas production under optimal temperature setting

Biogas production at different total solids concentration is also investigated. As shown with the dotted line in Figure 7.10, biogas production is increasing with the raise of the total solids concentration from 2% to 12%. However, Table 7.5 illustrates that the biogas production decreases when total solids concentration is less than 5%. This is due to the average value of total solids concentration in the test dataset being around 5%. When total solids concentration is larger than 6%, biogas production is rising until it reaches the maximum value for the total solids concentration reaching its upper constraining limit.

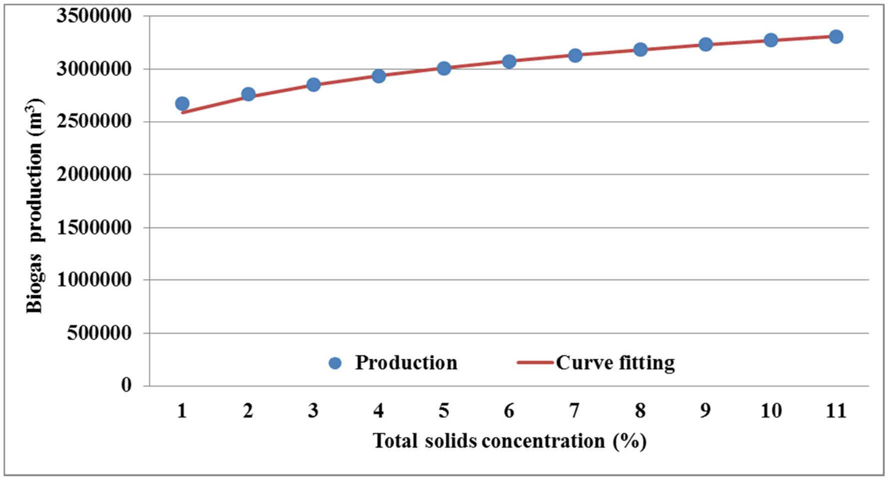


Figure 7.10. Biogas production with total solids concentration

It is observed that the increase slows down as total solids concentration becomes larger. The relationship between biogas production and total solids concentration can be fitted with the power function in (7.3) (the solid line in Figure 7.5).

$$y = a * (TS)^b \tag{7.3}$$

where y is the biogas production, TS is total solids concentration, a and b are constants of the power function. Here, a and b are computed as 8.302×107 and 0.1378, respectively.

The value of power coefficient b is much smaller than the value obtained by Igoni et al. (2007) which was 2.77. The authors believe that the value of the power coefficient calculated in this research is more reasonable. Figure 7.5 illustrates the relationship between the rate of biogas production and total solids concentration. Equation (7.3) indicates that if the power coefficient is smaller than 1, total solids concentration does not significantly increase biogas production. The sludge will become more acidic with higher total solids concentration (Itodo and Awulu, 1999).

Table 7.5. Biogas production change rate in the total solids concentration

Total solids concentration (%)	Change rate (%)
2	-9.4
3	-6.3
4	-3.4
5	-0.7
6	1.8
7	4.0
8	6.0
9	7.8
10	9.4
11	10.8
12	12.1

It has been determined from this model that the biogas production reaches its maximum for the total solids concentration of 12%. Figure 7.11 shows the results of biogas production for the optimized total solid concentration. The total biogas production can be increased up to 12.1%. The biogas production varies in response to the sludge flow rate and other input variables. However, this variability is small which implies stabile biogas output. The biogas production with the total solids concentration higher than 12% is also studied, even though this concentration is outside the range of the WRF operating conditions. It has been found that the biogas production increases 17% when the total solids concentration is 20%. Due to the relatively small data sample, the accuracy cannot be validated; however, it is expected that there is a certain point at which the biogas production will not increase even as the total solids concentration gets higher. A possible reason is the decrease of the water content in the sludge, with the higher total solids concentration resulting in a reduced level of active microorganism-digesting activities.

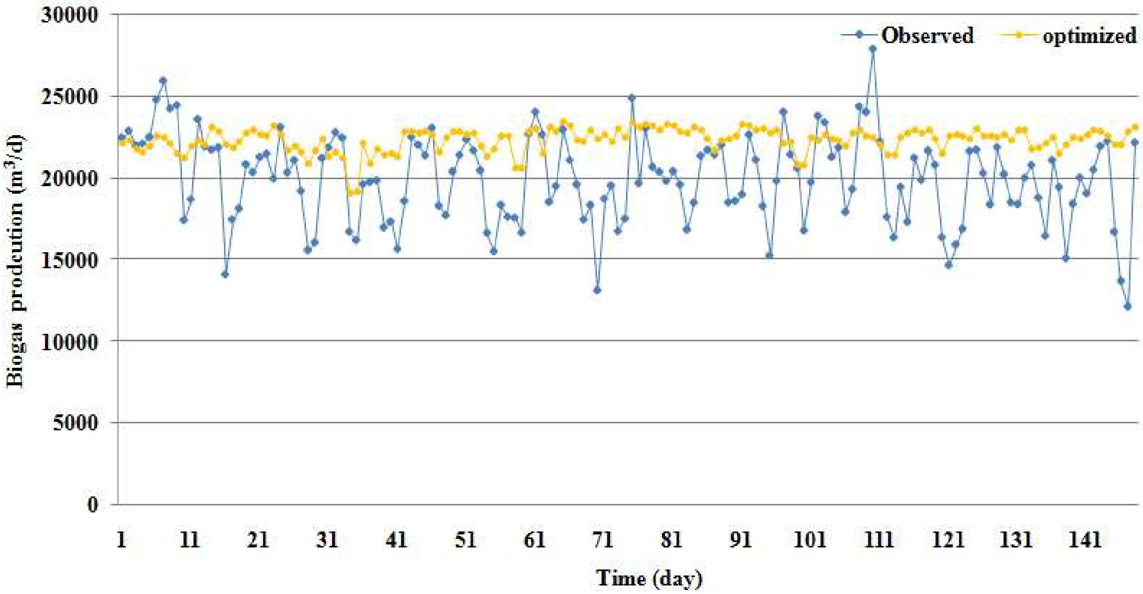


Figure 7.11. Observed and optimized biogas production under optimal total solids setting

It has determined the optimal pH value of 6.8. Given this operations condition, biogas production increased 1.9%. As shown in Figure 7.12, biogas production has increased compared to the biogas production under original sampled pH values. This proves that pH in the range 6.8 to 8.0 has a slight impact on biogas production. Figure 7.13illustrates the impact of pH on biogas production. A slight decrease of biogas production at higher pH values is observed.

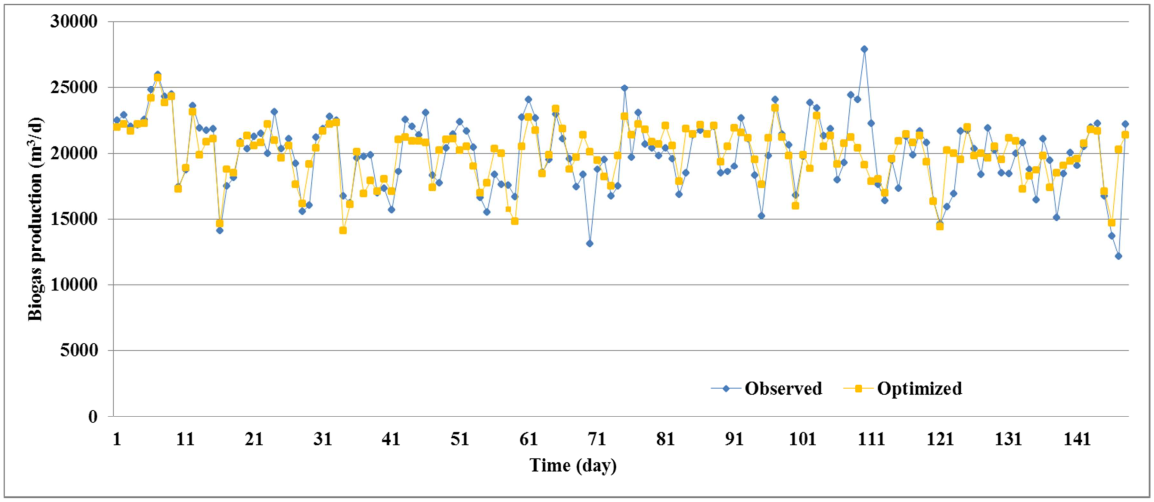


Figure 7.12. Observed and optimized biogas production for pH value of 6.8

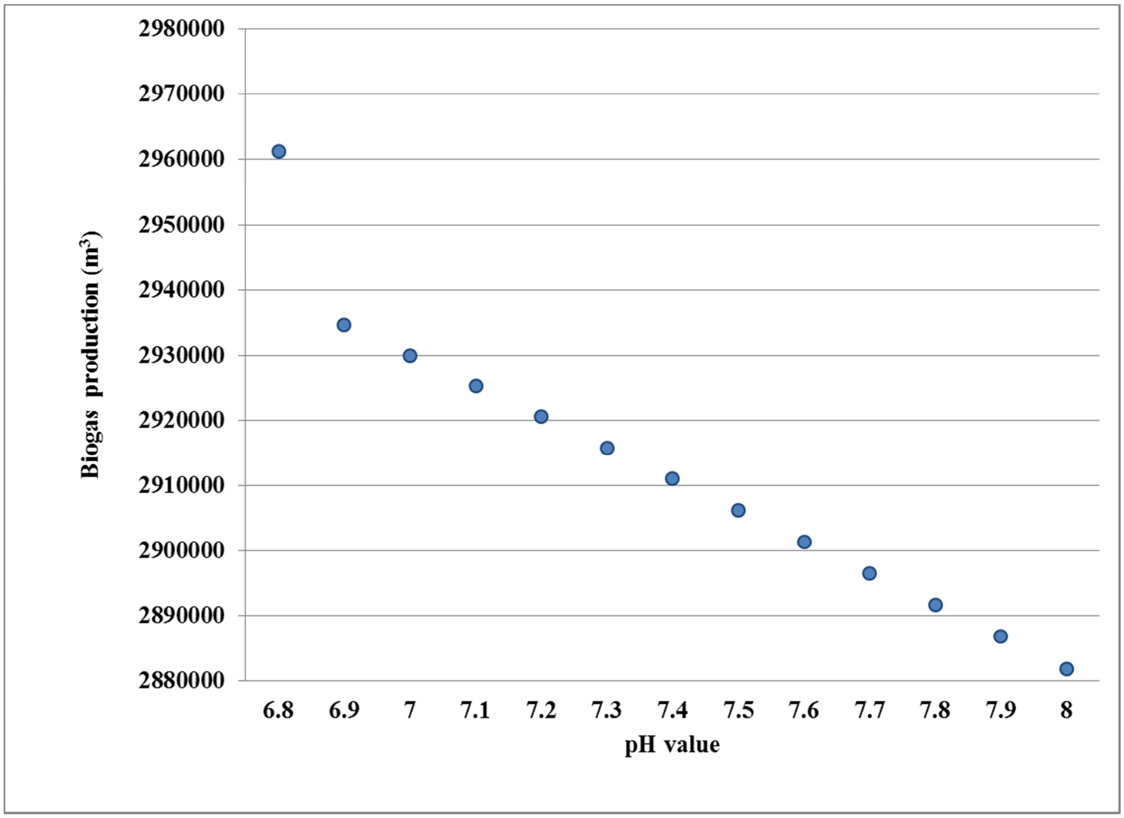


Figure 7.13. Biogas production with pH values

Table 7.6. Biogas production change rate in pH values

pH value	Rate of change (%)
6.8	1.9
6.9	1.0
7.0	0.86
7.1	0.70
7.2	0.54
7.3	0.38
7.4	0.21
7.5	0.05
7.6	-0.11
7.7	-0.29
7.8	-0.45
7.9	-0.63
8.0	-0.79

Table 7.6 indicates that biogas production decreases when pH value is larger than 7.6. This could be due to the fact that the average pH value of the test dataset is 7.53. It is also illustrated that pH has a small impact on biogas production across the range [6.8 - 8.0]. The maximum biogas production is obtained for pH value of 6.8, which is in the recommended range for anaerobic digestion operations. It is worth to clarify that pH might contribute to failures affecting the digestion process when its values are below 6.0 or above 8.0. For values smaller than 6.0, more acidic or basic mixtures ferment at lower speeds. The introduction of new sludge reduces the pH level. Digestion will stop or slow down until the microorganisms have neutralized the acids. High pH values encourage production of acidic carbon dioxide to neutralize the mix.

The PSO algorithm has determined that the optimal volatile solids is 75%. Given this operations condition, biogas production has increased 0.4%. The results imply that volatile solids has small impact on biogas production when it is in its lower and upper limits.

In the case that all controllable variables are optimized simultaneously, a 20.8% biogas production increase can be obtained as shown in Figure 7.14. This is an ideal situation as not all variables can be adjusted at the same time in wastewater treatment plant operation practice. The optimal values for all variables and the increases of biogas production are summarized in Table 7.7. “NA” means not being optimized in that case.

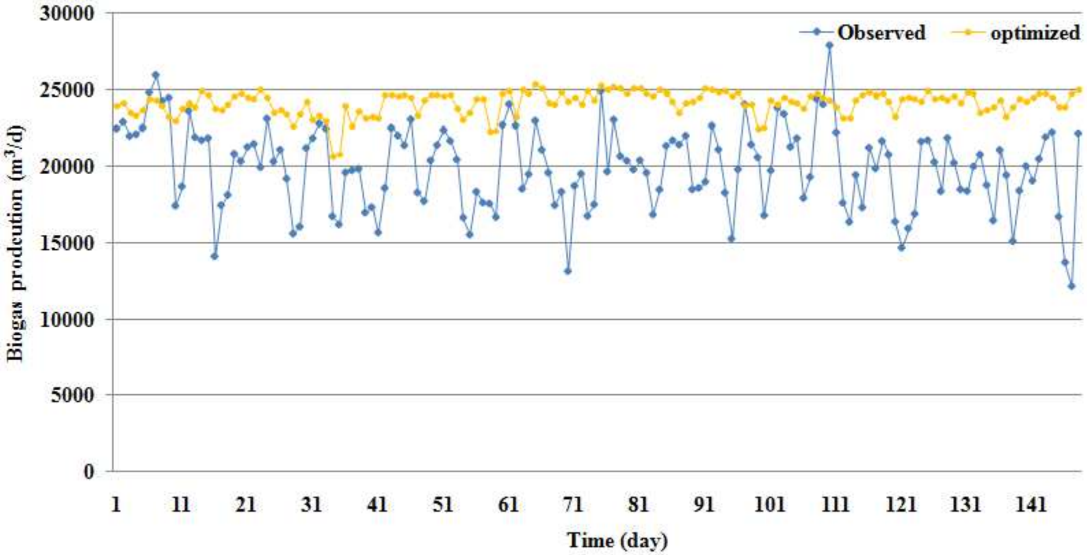


Figure 7.14. Observed and optimized biogas production under optimal settings of all variables

Table 7.7. Biogas production increasing rate with optimal settings

	Temperature (°C)	Total solids (%)	Volatile solids (%)	pH	Increasing rate (%)
1	39.0	NA	NA	N/A	5.3
2	NA	12	NA	N/A	12.1
3	NA	NA	75	N/A	0.4
4	NA	NA	NA	6.8	1.9
5	39.2	12	80	6.8	20.8

CHAPTER 8

CONCLUSION AND FUTURE WORK

8.1 Conclusion

As the wastewater treatment process is complex and dynamic, this dissertation is focused on developing a framework for its modeling and optimization with a data-driven approach. The framework includes two categories. The first category is modeling, where different data-mining algorithms and techniques are used to predict several important parameters in wastewater process, such as the influent flow rate, the total suspended solids, CBOD. The second category is optimization, where process is optimized by evolutional algorithms either to save energy consumption or to maximize the energy generation. The two categories are not separated but coupled together. The predicted values in the first category will be used as one input in the optimization in the second category.

First, the influent flow is forecasted with two data-driven neural networks. To satisfy the spatial and temporal characteristics of the influent flow, rainfall data collected at 6 tipping buckets, radar data measured by a radar station and historical influent data were used as model inputs. The static MLP neural network provided good prediction accuracy up to 150 min ahead. To extend the time horizon of predictions, to 300 min, a dynamic neural network with an online corrector was proposed. The time lag appeared in MLP neural network model was significantly reduced. The extended time horizon is useful for energy efficiency management of WTTPs.

Second, data-mining algorithms are applied to predict TSS in wastewater. Numerous scenarios involving carbonaceous biochemical oxygen demand (CBOD) and influent flow rate were investigated to construct the TSS time-series. The multi-layered perceptron (MLP) model performed best among the five different data-mining models that were derived for predicting TSS. The accuracy of the predictions was improved further by an iterative construction of MLP algorithm models. The values of TSS were predicted seven days in advance with accuracies that ranged from 73% to 79%.

And numerous models predicting carbonaceous biochemical oxygen demand (CBOD) is also investigated in Chapter 4. The performance of individual seasonal models was found to be better for fall and winter seasons, when the CBOD values were high. For low CBOD values, the modified seasonal models were found most accurate. Predictions for up to five days ahead were performed. The reason for the low accuracy of some of the models presented in the research was the low frequency (24 h) of the input data. Once higher frequency data becomes available, the prediction accuracy of CBOD will be improved. Such data will also allow the development of accurate models for predicting the potential of hydrogen (pH) and the total suspended solid (TSS).

Chapter 5 and all subsequent chapters focus on optimization of the process. In Chapter 5, optimization of wastewater pumping process is presented. 20 cases of different operating pump combinations are found through the collected dataset. To minimize energy consumption, a single-objective optimization model is formulated and solved with the proposed two-level intelligent algorithm. Based on the operation practice, decision variables are the number of operating pumps at the same and the rotating speed of the pump. The computational results revealed that a significant energy reduction was observed when the pumping station running under optimized optimal settings. The wet well level and outflow rate had not big difference before and after optimization.

Chapter 6 focuses on the energy efficiency of the activated sludge process. Two objectives are considered, i.e., minimizing the energy use and maximizing the effluent quality. Two control strategies, constant and hourly variable DO concentrations, are investigated to find the optimal DO concentrations for three different scenarios representing the preference over energy saving or effluent quality. The computational results indicated as much as 16% of the energy used in the process could be saved when preference was given to energy saving. A scenario that gave equal importance to energy saving and effluent quality was recommended to safely operate the activated sludge process. It could save 10% of the energy consumed with hourly-variable, optimal DO concentrations.

Optimization of biogas production is presented in Chapter 7. Controllable variables, temperature, total solids, volatile solids, pH, and uncontrollable variables, sludge flow rate, organic load, and detention time were selected to build a prediction model for biogas production with a multi-layer perceptron neural network. To optimize biogas production, a single-objective optimization model is formulated and solved with a particle swarm optimization algorithm. The computational results demonstrated that a 20.8% increase could be obtained when all controllable values were set to the optimal values at the same time.

8.2 Future research

The research reported in this thesis indicates that accurate prediction models resulting in significant energy savings can be developed. The predicted influent flow rate, TSS and CBOD concentration in the raw wastewater can provide useful information to manage the plant. The optimized settings, such as pump speed, configurations, temperature of the sludge, dissolved oxygen concentration in the aeration tank, etc., can give useful information to the plant operators to save energy or improve the biogas production.

Future research should focus on implementation on the proposed framework in the wastewater treatment plant. To apply the research to the wastewater industry, the knowledge from the research must be transferred to the treatment plant. A platform needs to be created with the efforts from both research and the plant. To provide real-time influent flow rate information to the plant, a program must be developed to collect and read the upstream flow information, and output both graphical and texted predicting values after processing the data and modeling at the background. The similar work should be executed for predicting TSS, CBOD, as well optimization of pumping process, the activated sludge process and maximizing the biogas production. During the implementation, the problems can be found and solved. These real-time experiments will improve the confidence of the plant to continue employ the data-driven approach to manage the wastewater treatment process rather than control the settings based on the experience.

Another future direction is the integration of various concepts reported in this dissertation into a comprehensive model. The accomplished tasks cover the main processes in the wastewater treatment plant. It would be interesting to create an integration program which could show all predicted information and optimized variable settings with only several inputs such as upstream flow rates and local temperature, etc. A plant operator then could read the information and make decisions to optimally manage the plant.

As the online sensors are expensive and require frequently maintenance, developing virtual sensors with a data-driven approach is worth to be studied in the future. Using virtual sensors can not only save the investment cost of the devices and high maintenance expenses, but also significantly decrease the noise generated by online sensors. They can be also used to provide inputs to fill the missing values in the collected dataset, which lead to higher accuracy of the predicting models.

REFERENCES

1. Li, X.D.; Zeng, G.M.; Huang, G.H.; Li, J.B. and Jiang, R. Short-term prediction of the influent quantity time series of wastewater treatment plant based on a chaos neural network model. *Frontiers of Environmental Science & Engineering in China*, **2007**, 1(3), pp. 334-338.

2. Bechmann, H.; Nielsen, M.K.; Madsen, H. and Poulsen, N.K. Grey-box modeling of pollutant loads from a sewer system. *Urban Water*, **1999**, 1(1), pp. 71-78.

3. Qasim, S.R. *Wastewater Treatment Plants: Planning, Design, and Operation*; CRC Press: Boca Raton, 1998.

4. Vesillind, P.A. Wastewater Treatment Plant Design. IWA Publishing: Alexandria, 2003.

5. Kim, J.R.; Ko, J.H.; Im, J.H.; Lee, S.H.; Kim, S.H.; Kim, C.W. and Park, T.J. Forecasting influent flow rate and composition with occasional data for supervisory management system by time series model. *Water Science & Technology*, **2006,** 53(4), pp. 185-192.

6. Kurz, G.E.; Ward, B. and Ballard, G.A. Simple method for estimating I/I using treatment plant flow monitoring reports – a self help tool for operators. *Proceedings of the Water Environment Federation, Collection systems,* **2009**, pp. 568-576(9).

7. Beraud, B.; Steyer, J.P.; Lemoine, C.; Gernaey, K.V. and Latrille, E. Model-based generation of continuous influent data from daily mean measurements available at industrial scale. *In Proceedings 3rd International IWA Conference on Automation in Water Quality Monitoring*, September 5-7, Gent, Belgium, 2007.

8. Keyser, W. De; Gevaert, V.; Verdonck, F.; Baets, B. D. and Benedetti, L. An emission time series generator for pollutant release modeling in urban areas. *Environmental Modeling & Software*, **2010**, 25(4), pp. 554-561.

9. Djebbar, Y. and Kadota, P.T. Estimating sanitary flows using neural networks. *Water Science and Technology*, **1998**, 38(10), pp. 215-222.

10. Gernaey, K.V.; Rosen, C. and Jeppsson, U. *BSM2: A model for dynamic influent data generation*, Technical Report, no.8, IWA Task Group on Benchmarking of Control Strategies for Wastewater Treatment Plants, 2010.

11. Hernebring, C.; Jonsson, L.E.; Thoren, U.B. and Moller, A. Dynamic online sewer modeling in Helsingborg. *Water Science and Technology*, **2002**, 45(4-5), pp. 429-436.

12. Gernaey, K.V.; Rosen, C.; Benedetti, L. and Jeppsson, U. Phenomenological modeling of wastewater treatment plant influent disturbance scenarios. *In Proceedings 10th International Conference on Urban Drainage (10ICUD),* August 21-26, Copenhagen, Denmark, 2005.

13. Carstensen, J.; Nielsen, M.K. and Strandbæk, H. Prediction of hydraulic load for urban storm control of a municipal WWT plant. *Water Science and Technology,* **1998**, 37(12), pp. 363-370.

14. Pons, M.N.; Lourenco, M.C. and Bradford J. Modeling of wastewater treatment influent for WWTP benchmarks. *In Proceedings 10th IWA Conference on Conference on Instrumentation, Control and Automation*, June 14-17, Cairns, Australia, 1998.

15. Ismail, M.J.; Ibrahim, R. and Ismail, I. Development of neural network prediction model of energy consumption. *World Academy of Science, Engineering and Technology*, **2011**, 58, pp. 862-867.

16. Parlos, A.G.; Chong, K.T. and Atiya, A.F. Application of the recurrent multilayer perceptron in modeling complex process dynamics. *IEEE Computational Intelligence Society*, **1994**, 6(2), pp. 255-266.

17. Hsu, K.; Gupta, H.V. and Sorroshian, S. Artificial neural network modeling of the rainfall-runoff process. *Water Resources Research*, **1995,** 31(10), pp. 2517-2530.

18. Besaw, L.E.; Rizzo, D.M.; Bierman, P.R. and Hackett, W.R. Advances in ungauged streamflow prediction using artificial neural networks. *Journal of Hydrology*, **2010**, 386, pp. 27-37.

19. Spina, M.S.; Schwartz, M.J. and Staelin, D.H. Application of multilayer feedforward neural networks to precipitation cell-top altitude estimation. *IEEE Transactions on Geoscience and Remote Sensing*, **1998**, 36(1), pp. 154-162.

20. Verma, A.; Wei, X.P. and Kusiak, A. Predicting the total suspended solids in wastewater: a data-mining approach. *Engineering Applications of Artificial Intelligence*, **2013**, 26(4), pp. 1366-1372.

21. Kriger, C. and Tzoneva, R. Neural networks for prediction of wastewater treatment plant influent disturbances. *In Proceedings of IEEE AFRICON Conference,* September 26-28, Windhoek, Namibia, 2007.

22. Vyas, M.; Modhera, B.; Vyas, V. and Sharma, A.K. Performance forecasting of common effluent treatment plant parameters by artificial neural network. *ARPN Journal of Engineering and Applied Sciences*, **2011,**6(1), pp. 38-42.

23. Gurney, K. *An Introduction to Neural Networks*; CRC Press: London, 1997.

24. Haykin, S.S. *Neural networks: a comprehensive foundation*; Prentice Hall: Upper Saddle River, 1999.

25. Hassoun, M. H. *Fundamentals of Artificial Neural Networks*; The MIT Press: Cambridge, 1995.

26. Parlos, A.G.; Fernandez, B.; Atiya, A.F.; Muthusami, J. and Tsai, W.K. An accelerated learning algorithm for multilayer perceptron networks. *IEEE Computational Intelligence Society*, **1994**, 5(3), pp. 493-497.

27. Chaudhuri, B.B. and Bhattacharya, U. Efficient training and improved performance of multilayer perceptron in pattern classification. *Neurocomputing*, **2000**, 34(1-4), pp. 11-27.

28. Gardner, M.W. and Dorling, S.R. Neural network modeling and prediction of hourly NOx and NO2 concentrations in urban air in London. *Atmospheric Environment*, **1999**, 33(5), pp. 709-719.

29. Saini, L.M. Artificial neural network based peak load forecasting using Levenberg-Marquardt and quasi-Newton methods. *IEE Proceedings on Generation, Transmission and Distribution*, **2002**, 149(5), pp. 578-584.

30. Chiang, Y.M.; Chang, L.C.; Tsai, M.J.; Wang, Y.F. and Chang F.J. Dynamic neural networks for real-time water level predictions of sewerage systems – covering gauged and ungauged sites. *Hydrology and Earth System Sciences*, **2010**, 7, pp. 2317-2345.

31. Hussain, A.J.; Jumeily, D.A. and Lisboa, P. Time series prediction using dynamic ridge polynomial neural networks. *2009 Second International Conference on Developments in eSystems Engineering*, pp. 354-363, Abu Dhabi, December 14-16, 2009.

32. Shaw, A.M.; Doyle III, F.J. and Schwaber, J.S. A dynamic neural network approach to nonlinear process modeling. *Computers & Chemical Engineering*, **1997**, 21(20), pp. 371-385.

33. Velasquez, J.D.; Rios, S.A.; Howlett, R.J. and Jain, L.C. *Knowledge-based and Intelligent Information and Engineering Systems*; Springer: Germany, 2009.

34. Htike, K.K. Rainfall forecasting models using focused time-delay neural networks. *2010 International conference on Computer and Communication Engineering*, pp. 1-6, Kuala Lumpur, Malaysia, May 11-13, 2010.

35. Bilotta, G.S. and Brazier, R.E. Understanding the influence of suspended solids on water quality and aquatic biota. *Water Research*, **2008**, 42(3), pp. 2849-2861.

36. Choi, D.J. and Park, H. A hybrid artificial neural network as a software sensor for optimal control of a wastewater treatment process. *Water Research*, **2002**, 35(16), pp. 3959-3967.

37. Cartensen, J.; Harremoes, P. and Strube, R. Software sensors based on the grey-box modeling approach. *Water Science and Technology*, **1996**, 33(1), pp. 117-126.

38. Gernaey, K.V.; Loosdrecht, M.C.M.; Henze, M.; Lind, M. and Jorgensen, S. Activated sludge wastewater treatment plant modeling and simulation: state of the art. *Environmental Modeling and Software*, **2004**, 19(9), pp. 763-783.

39. Poch, M.; Comas, J.; Roda, I.R.; Marre, M.S. and Cortes, U. Designing and building real world environmental decision support systems. *Environmental modeling and software*, **2004**, 19(9), pp. 857-873.

40. Rivas, A.; Irizar, I. and Ayesa, E. Model-based optimization of wastewater treatment plants design. *Environmental modeling and software*, **2008**, 23(4), pp. 435-450.

41. Zhu, J.; Zurcher, J.; Rao, M. and Meng, M.Q.H. An on-line wastewater quality prediction system based on a time delay neural network. *Engineering applications of Artificial Intelligence*, **1998**, 11(2), pp. 747-758.

42. [42] Tan, P.C.; Berger, C.S.; Dabke, K.P. and Mein, R.G. Recursive identification and adaptive prediction of wastewater flows. *Automatica*, **1991**,27(5), pp. 761-768.

43. Choi, S.C. Test of equality of dependent correlations. *Biometrika*, **1977**, 64(3), pp. 645–647.

44. Jolliffe I.T. *Principal Component Analysis*. Springer Series in Statistics, 2nd ed.; Springer: New York, 2002.

45. Kusiak, A. and Li, W. Virtual Models for Prediction of Wind Turbine Parameters. *IEEE Transactions on Energy Conversion*, **2010**, 25(1), pp. 245-252.

46. Kusiak, A, and Song, Z., “Combustion efficiency optimization and virtual testing: A data-mining approach,” IEEE Trans. Ind. Inform., Vol. 2(3), pp. 176–184, 2006.

47. Qasim, S.R. *Wastewater Treatment Plants: Planning, Design, and Operation*; CRC Press: Boca Raton, 1998

48. Hertz, J.A.; Krogh, A. and Palmer, R.G. *Introduction to the theory of neural computation*; Westview Press: Boulder, CO, 1999.

49. Mamais, D., Jenkins, D., & Pitt, P. (1993). A rapid physical-chemical method for the determination of readily biodegradable soluble COD in municipal wastewater. Water Research, 27(1), 195-197.

50. Bernard, O.; Hadj-Sadok, Z.; Dochain, D.; Genovesi, A. and Steyer, J.P. Dynamical model development and parameter identification for an anaerobic wastewater treatment process. *Biotechnology and Bioengineering*, **2001**, 75(4), pp. 424-438.

51. Yu, H.Q. and Fang, H.P. Acidogenesis of gelatin-rich wastewater in an up flow anaerobic reactor: influence of pH and temperature. *Water Research*, **2003**, 37(1), pp.55-66.

52. Choi, D.J. and Park, H. A hybrid artificial neural network as a software sensor for optimal control of a wastewater treatment process. *Water Research*, **2001**, 35(16), pp. 3959-3967.

53. Cartensen, J.; Harremoes, P. and Strube, R. Software sensors based on the grey-box modeling approach. *Water Science and Technology*, **1996**, 33(1), pp. 117-126.

54. Ciaccio, L.L. Instrumental determination of the energy oxygen and BOD5. *Water Science and Technology*, **1992**, 26(5-6), pp. 1345-1353.

55. Spanjers, H.; Olsson, G. and Klapwijk, A. Determining influent short-term biochemical oxygen demand by combined respirometry and estimation. *Water Science and Technology*, **1993**, 28(11-12), pp. 401-414.

56. Acha, V.; Meurens, M.; Naveau, H.; Dochain, D.; Bastin, G. and Agathos, S.N. Model-based estimation of an anaerobic reductive dechlorination process via an attenuated total reflection-Fourier transform infrared sensor. *Water Science and Technology*, **1999**, 40(8), pp. 33-40.

57. Cheruy, A. Software sensors in bioprocess engineering. *Journal of Biotechnology*, **1997**, 52(3), pp. 193-199.

58. Holmberg, A. Modeling of the activated sludge process for microprocessor-based state estimation and control. *Water Research*, **1982**, 16(7), pp. 1233-1246.

59. Onnerth, T.B.; Nielsen M.K. and Stamer, C. Advanced computer control based on real and software sensors. *Water Science and Technology*, **1996**, 33(1), pp. 234-245.

60. Giudici, P. and Figini, S. *Applied Data Mining for Business and Industry*; John Wiley & Sons: Chichester, UK, 2009.

61. Grossman, R.L.; Kamath, C.; Kegelmeyer, P.; Kumar, V. and Namburu, R.R. *Data Mining for Scientific and Engineering Applications*; Kluwer Academic Publishers: The Netherlands, 2001.

62. Madalina, C. The pollution level analysis of a wastewater treatment plant emissary using data mining. *Buletinul*, **2010**, LXII(1), pp. 69-78.

63. Sànchez-Marrè, M.; Gibert, K. and Rodríguez-Roda, I. GESCONDA: A Tool for Knowledge Discovery and Data Mining in Environmental Databases. *Research on Computing Science,* **2004**, 11, pp. 348-364.

64. Coope, I.D. Circle fitting by linear and nonlinear least squares. *Journal of Optimization: Theory and Applications*, **1993**, 76(2).

65. Kudo, T. and Matsumoto, Y. A Boosting Algorithm for Classification of Semi-Structured Text. Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2004.

66. Kohavi, R. and John, G.H. WRFppers for feature subset selection. *Artificial Intelligence*, **1997**, 97(1-2), pp. 273-324.

67. Droste, R.L. *Theory and practice of water and wastewater treatment: 1st Edition*; Wiley, John & Sons: New York, 1996.

68. Tillman, G.M. *Primary Treatment at Wastewater Treatment Plants*; Lewis Publishers, Inc: Chelsea, 1991.

69. Vesilind, P.A. *Wastewater Treatment Plant Design*; International Water Association Publishing: Alexandria, VA, 2003.

70. Wei, X.; Kusiak A. and Sadat H. Prediction of Influent Flow Rate: A Data-Mining Approach. *Journal of Energy Engineering*. doi: 10.1061/(ASCE)EY.1943-7897.0000103, 2012.

71. A Guide to LCC Analysis for Pumping Systems. Hydraulic Institute & Europump, 2001.

72. Proceedings EEMODs ’99 Energy Efficiency in Motors and Driven Systems, January 2000. SAVE, for EC DGXVII. Published by Springer.

73. Centrifugal Pumps, Karassik & McGuire, 2nd Edition; Chapman & Hall: New York, 1998.

74. Kaya, D.; Yagmur, E.A.; Yigit, K.S.; Kilic, F.C.; Eren, A.S. and Celik, C. Energy efficiency in pumps. *Energy Conversion and Management*, **2008**, 49(6), pp. 1662-1673.

75. Ma, Z.J. and Wang ,S.W. Energy efficient control of variable speed pumps in complex building central air-conditioning systems. *Energy and Buildings*, **2009**, 41(2), pp. 197-205.

76. Zhang, Z.J.; Zeng, Y.H. and Kusiak A. Minimizing pump energy in a wastewater processing plant, *Energy*, **2012**, 47, pp. 505-514.

77. Kusiak, A.; Zheng H.Y. and Song, Z. Models for monitoring wind farm power. *Renewable Energy*, **2009**, 34(3), pp. 583-590.

78. Wei, X. and Kusiak, A. Optimization of Biogas Production Process in a Wastewater Treatment Plant. Proceedings of the 2012 Industrial and Systems Engineering Research Conference, Orlando, FL, May 2012, 1-9.

79. Hou, Z.J.; Lian, Z.W.; Yao, Y. and Yuan X.J. Data mining based sensor fault diagnosis and validation for building air conditioning system. *Energy Conversion and Management*, **2006**, 47(15-16), pp. 2479-2490.

80. Kusiak, A. and Zhang, Z. Adaptive control of a wind turbine with data mining and swarm intelligence. *IEEE Transactions on Sustainable Energy*, **2011**, 2(1), pp. 28-36.

81. Kusiak, A. and Li, W.Y. The prediction and diagnosis of wind turbine faults. *Renewable Energy*, **2011**, 36(1), pp. 16-23.

82. Wolfl, C.; McLoone, S. and Bongards, M. Biogas plant optimization using Genetic Algorithms and Particle Swarm Optimization. *Irish Signals and Systems Conference*, **2008**, pp. 244-249.

83. Kennedy, J. and Eberhart, R. Particle swarm optimization. Proceedings of IEEE International Conference on Neural Networks, IV, 1942–1948, 1995.

84. Dai, Y.S. and Niu, H. An improved PSO algorithm and its application in seismic wavelet extraction. *I.J. Intelligent Systems and Applications*, **2011**, 5, pp. 34-40.

85. Eckenfelder, W. W. and Grau P. *Activated sludge process design and control: theory and practice*; Technomic Publishing: Pennsylvania, 1998.

86. Andrews, J.F. *Dynamics and control of the activated sludge process*; Technomic Publishing: Pennsylvania, 1992.

87. Sollfrank, U. and Gujer, W. Characterisation of domestic wastewater for mathematical modeling of the activated sludge process. *Water Science and Technology*, **1990**, 23(4-6), pp. 1057-1066.

88. Dold, P.L.; Ekama, G.A. and Marais, G.R. A general model for the activated sludge process. *Progress in Water Technology*, **1980**, 12(6), pp. 47-77.

89. Wei, Y.S.; Houten, R.T.V.; Borger, A.R.; Eikelboom, D.H. and Fan, Y.B. Minimization of excess sludge production for biological wastewater treatment. *Water Research*, **2003**, 37(18), pp. 4453-4467.

90. Liu, Y. and Tay, J.H. Strategy for minimization of excess sludge production from the activated sludge process. *Biotechnology Advances*, **2001**, 19(2), pp. 97-107.

91. Flores, X.; Bonmati, A.; Poch, M. and Roda, I.R. Selection of the activated sludge configuration during the conceptual deisng of activated sludge plants using multicriteria analysis. *Industrial & Engineering Chemistry Research*, **2005**, 44(10), pp. 3556-3566.

92. Yoo, C.K.; Kim, D.S.; Cho, J.H.; Choi, S.W. and Lee, I.B. Process system engineering in wastewater treatment process. *Korean Journal of Chemical Engineering*, **2001**, 18(4), pp. 408-421.

93. Holenda, B.; Domokos, E.; Redey, A. and Fazakas, J. Aeration optimization of a wastewater treatment plant using genetic algorithm. *Optimal Control Applications and Methods*, **2007**, 28(3), pp. 191-208.

94. Li, P.P.; Shen, L.; Li, F.P.; Jiang, Y. and Zhang, H.P. Optimal design and operation of settling tank in alternating activated sludge process based on 3-D model simulation. *5th International Conference on Bioinformatics and Biomedical Engineering*, **2011**, pp. 1-4.

95. Rosso, D.; Stenstrom, M.K. and Larson, L.E. Aeration of large-scale municipal wastewater treatment plants: state of the art. *Water Science and Technology*, **2008**, 57(7), pp. 973-978.

96. Ibrehem, A.S. and Hussain, M.A. Mathematical model and advanced control for activated sludge process in sequencing batch reactor. *Journal of Chemical Engineering*, **2009**, 9(1), pp. 32-46.

97. Wall, A.; Hart, M.; Mack, J. and O'Brien, M. The use of advanced process monitoring and control to optimize energy use on an activated sludge plant. *Proceedings of the Water Environment Federation*, **2009**, Session 51 through Session 60, pp. 3788-3797.

98. M. O'Brien, J. Mack, B. Lennox, D. Lovett, and A. Wall, "Mode predictive control of an activated sludge process," Control Engineering Practice, vol. 19, no. 1, pp. 54-61. Jan, 2011.

99. Olsson, G. and Newell, B., *Wastewater treatment systems: modellig, diagnosis and control*; IWA Publishing: London, 1999.

100. Gernaey, K.V.; Loosdrecht, M.C.M.V.; Henze, M.; Lind, M. and Jorgensen, S.B. Activated sludge wastewater treatment plant modelling and simulation: state of the art. *Environmental Modelling & Software*, **2004**, 19(9), pp. 763-78.

101. Holenda, B.; Domokos, E.; Redey, A. and Fazakas, J. Dissolved oxygen control of the activated sludge wastewater treatment process using model predictive control. *Computers & Chemical Engineering*, **2008**, 32(6), pp. 1270-1278.

102. Olsson, G. and Andrews, J.F. The dissolved oxygen profile - a valuable tool for control of the activated sludge process. *Water Research*, **1978**, 12(11), pp. 985-1004.

103. Linberg, C.F. and Carlsson, B. Nonlinear and set-point control of the dissolved oxygen concentration in an activated sludge process. *Water Science and Technology*, **1996**, 34(3-4), pp. 135-142.

104. Piotrowski,R.; Brdys, M.A.; Konarczak, K.; Duzinkiewicz, K. and Chotkowski, W. Hierarchical dissolved oxygen control for activated sludge processes. *Control Engineering Practice*, **2008**, 16(1), pp. 114-131.

105. Kusiak, A. and Wei X. A data-driven model for maximization of methane production in a wastewater treatment plant. *Water Science and Technology*, **2012**, 65(6), pp. 1116-1122.

106. Ramakrishnan, N. and Grama, A.Y. Data mining: from serendipity to science. *Computer*, **1999**, 32(108), pp. 34-37.

107. Yassine, A.; Joglekar, N.; Braha, D.; Eppinger, S. and Whitney, D. Information hiding in product development: the design churn effect. *Research in Engineering Design*, **2003**, 14(3), pp. 145-161.

108. Braha, D. *Data mining for design and manufacturing*; Kluwer: Boston, 2001.

109. Friedman, J.; Hastie, T. and Tibshirani, R. Additive logistic regression: a statistical view of boosting. *The Annals of Statistics*, **2000**, 28(2), pp. 337-407.

110. Hagan, M.T.; Demuth, H.B. and Beale, M. *Neural network design*; PWS Publishing: Boston, 1996.

111. Haykin, S. O. *Neural networs and learning machines*; Prentice Hall: New Jersey, 2008.

112. Bonnans, J.F.; Gilbert, J.C.; Lemarechal, C. and Sagastizabal, C.A. *Numerical Optimization: Theoretical and Practical Aspects*; Springer: New York, 2006.

113. Zitzler, E.; Laumanns, M. and Thiele, L. SPEA2: improving the strength pareto evolutionary algorithm. Technical Report, Swiss Fedearal Institute of Technology, 2001.

114. Marchaim, U. Biogas process for sustainable development, Food and Agriculture Organization of the United Nations, Rome, 1992.

115. Lettinga, G. Anaerobic digestion and wastewater treatment systems. *Biomedical and Life Sciences*, **1995**, 67(1), pp.3-28.

116. Batstone, D. J.; Keller, J.; Angelidaki, I.; Kalyuzhnyi, S. V.; Pavlostathis, S. G.; Rozzi, A; Sanders, W. T.; Siegrist, H. and Vavilin, V. A. The IWA anaerobic digestion model No1 (ADM1). *Water Science and Technology*, **2002**, 45(10), pp. 65-73.

117. Mailleret, L.; Bernard, O. and Steyer, J.P. Robust regulation of anaerobic digestion processes. *Water Science and Technology,* **2003**, 48(6), pp. 87-94.

118. Schubert, J.; Simutis, R.; Dors, M.; Havlik, I. and Lubbert, A. Bioprocess optimization and control: application of hybrid modeling, *Journal of Biotechnology*, **1994**, 35(1), pp. 51-68.

119. Tay, J.H. and Zhang, X. Neural fuzzy modeling of anaerobic biological wastewater treatment systems. *ASCE Journal of Environmental Engineering*, **1999**, 125(12), pp. 149-1159.

120. Sotemann, S.W.; Ristow, N.E.; Wentzel, M.C. and Ekama, G.A. A steady state model for anaerobic digestion of sewage sludge. *Water SA*, **2005**, 31(4), pp. 511-527.

121. Cakmakci, M. Adaptive neuro-fuzzy modeling of anaerobic digestion of primary sedimentation sludge. *Bioprocess and Biosystems Engineering,* **2007**, 30(5), pp. 349-357.

122. Holubar, P.; Zani, L.; Hagar, M.; Froschl, W.; Radak, Z. and Braun, R. Advanced controlling of anaerobic digestion by means of hierarchical neural networks. *Water Research*, **2002**, 36(10), pp. 2582-2588.

123. Kusiak, A. and Li, M.Y. Cooling output optimization of an air handling unit. *Applied Energy*, **2010**, 87(3), pp. 901-909.

124. Frank, E.; Hall, M., Trigg L.; Holmes, G. and Witten, I.H. Data mining in bioinformatics using Weka. *Bioinformatics*, **2004**, 20(15), pp. 2479- 2481.

125. Kusiak, A.; Zheng, H.Y. and Song Z. Wind farm power prediction: A data-mining approach. *Wind Energy*, **2009**, 12(3), pp. 275-293.

126. Seckin, N. Modeling flood discharge at ungauged sites across Turkey using neuro-fuzzy and neural networks. *Journal of Hydroinformatics*, **2011**, 13(4), pp.842-849.

127. Dasgupta, D. and Michalewicz, Z. *Evolutionary algorithms in engineering applications*; Springer: New York, 2001.

128. Rud, O.C. *Data mining cookbook: modeling data for marketing, risk and customer relationship management*; John Wiley & Sons: New York, 2001.

129. Thiele, L.; Miettine, K.; Korhonen, P.J. and Molina, J. A preference-based evolutionary algorithm for multi-objective optimization. *Evolutionary Computation*, **2009**, 17(3), pp. 411-436.

130. Igoni, A.H.; Abowei, M.F. N.; Ayotamuno, J.M. and Eze, C.L. Effect of total solids concentration of municipal solid waste in anaerobic batch digestion on the biogas produced. *International Journal of Food, Agriculture and Environment*, **2007**, 5(2), pp. 333-337.

131. Cybenko, G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, **1989**, 2(4), pp. 303-314.

132. Thomas, A. F. and Mauricio, G. C. R. Greedy randomized adaptive search procedures. *Journal of global optimization*, **1995**, 6(2), pp. 109–133.

133. Deb, K.; Pratap, A.; Agarwal, S. and Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, **2002**, 6(2), pp. 182–197.