Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Electric Power Engineering

Øystein Steinsvik Evjen

Implementing LSTM machine learning in long-term hydropower scheduling

Master's thesis in Energi og miljø
Supervisor: Jayaprakash Rajasekharan
Co-supervisor: Jinghao Wang
July 2023

Norwegian University of Science and Technology

Øystein Steinsvik Evjen

Implementing LSTM machine learning in long-term hydropower scheduling

Master's thesis in Energi og miljø
Supervisor: Jayaprakash Rajasekharan
Co-supervisor: Jinghao Wang
July 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of Electric Power Engineering

NTNU
Norwegian University of
Science and Technology

Abstract

The energy demand is rapidly increasing worldwide, and with the environmental crisis is more people substituting fossil fuels with renewable energy sources. The new renewable energy sources are not as flexible and reliable as the old fossil power sources which can unstabilize the power grid. Hydropower plants can be scheduled, and therefore stabilize the power grid, but the higher complexity in scheduling makes the computational time too high to conduct a hydropower schedule often enough. Therefore might machine learning help with lowering the time complexity, and thereby let the scheduling be done at a higher frequency. From a literature review have the LSTM neural network model been a possible fit for this research. After testing the model the LSTM model had a 99.7% lower computational time, but the accuracy with MAPE evaluation as low as 67.5% did not hold a standard where the model could be recommended as a substitute, but it can be used as a supplement.

Sammendrag

Energiforbruket øker raskt globalt, og med miljøkrisen erstatter stadig flere mennesker fossile brensler med fornybare energikilder. De nye fornybare energikildene er ikke like fleksible og pålitelige som de gamle fossile energikildene, noe som kan føre til ustabilitet i strømnettet. Vannkraftverk kan planlegges, og dermed stabilisere strømnettet, men den økte kompleksiteten i planleggingen gjør at beregningstiden blir for lang til å gjennomføre en vanlig vannkraftplan ofte nok. Derfor kan maskinlæring bidra til å redusere tidskompleksiteten og dermed tillate hyppigere planlegging. Etter en litteraturgjennomgang har LSTM nevrale nettverkmodellen vist seg å være en mulig løsning for denne forskningen. Etter å ha testet modellen viste det seg at LSTM-modellen hadde 99,7% lavere beregningstid, men nøyaktigheten med MAPE-evalueringen så lav som 67,5% oppfylte ikke en standard der modellen kunne anbefales som en erstatning, men den kan brukes som et supplement.

Preface

This master thesis is written for the course TET4900 - Elektrisk energi og energisystemer, masteroppgave at NTNU the spring semester of 2023.

I extend my gratitude to my supervisor, Associate Professor Jayaprakash Rajasekharan, for his guidance and constructive feedback throughout my journey. I would also especially like to express my appreciation to my co-supervisor, Ph.D. Candidate Jinghao Wang for his opinion and contributions in providing data, and running data for me. His support has been very helpful on to write this master thesis.

Table of Contents

List of Figures iv

List of Tables v

1 Introduction 1

2 Theory 2

2.1 Hydropower Plant 2

2.1.1 Hydro power scheduling 4

2.1.2 Load Shedding 4

2.2 Benders Decomposition 5

2.3 Machine Learning and Neural Networks 6

2.3.1 Linear Regression 7

2.3.2 Neural Network 7

2.3.3 Long Short-Term Memory (LSTM) 9

2.3.4 Bayesian Optimization 9

2.3.5 Ensemble Models 10

2.3.6 Performance Measuring 10

2.4 Syntetic data 12

2.4.1 Data types 12

2.5 Theory conclusion 12

3 Literature review 14

3.1 Hydro power scheduling 14

3.2 Machine learning in Hydro power scheduling 15

3.2.1 Reinforcement learning 15

3.2.2 Neural Network 15

3.2.3 Other approaches 16

3.3 Conclution of literature review 16

4 Mathematical model 18

4.1 Optimization model 18

4.1.1 Notations . . . 18
4.1.2 Mathematical model . . . 20
4.2 System . . . 21
4.2.1 Reservoir . . . 21
4.2.2 Water flow . . . 22
4.2.3 Production . . . 22
4.3 Scenario fan simulator . . . 23
4.4 Conclusion Method . . . 25

5 Case study **26**
5.1 Description . . . 26
5.2 Synthetic data . . . 28
5.3 Machine learning models . . . 28
5.4 Conclusion of Case study . . . 29

6 Result and discusion **30**
6.1 Hydro power production . . . 30
6.2 Water value . . . 35
6.3 Reservoir level . . . 39
6.4 Computational time . . . 43
6.5 General source of the error . . . 43

7 Conclution **45**

8 Future work **46**

References **47**

Appendix **51**
A Appendices . . . 51

List of Figures

1 Cross-sectional view of a typical hydropower plant 3

2 Information flow in Benders Decomposition. 6

3 This is an example of how a Neural Network can look 8

4 system with system borders 22

5 crossection of an dam 23

6 This figure shows the logic of the SFS model for a model with 52 weeks total, n scenarios total for the first two weeks. 24

7 Original hydropower case 26

8 diagram of the hydro power plant 27

9 This figure shows a grid of the weekly production prediction for reservoir 0 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left showed the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method. 31

10 This figure shows a grid of the weekly production prediction for reservoir 1 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left showed the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method. 32

11 This figure shows a grid of the weekly production prediction for reservoir 2 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left showed the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method. 33

12 This figure shows a grid of the weekly water value prediction for reservoir 0 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left showed the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method. 36

13 This figure shows a grid of the weekly water value prediction for reservoir 1 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left showed the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method. 37

14 This figure shows a grid of the weekly water value prediction for reservoir 2 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left showed the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method. 38

15 This figure shows a grid of the weekly reservoir level prediction for reservoir 0 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left shows the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method. 40

16 This figure shows a grid of the weekly reservoir level prediction for reservoir 1 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left shows the prediction conducted with LSTM, and the lower right figure shows the prediction conducted with the ensemble method. 41

17 This figure shows a grid of the weekly reservoir level prediction for reservoir 2 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left shows the prediction conducted with LSTM, and the lower right figure shows the prediction conducted with the ensemble method. 42

List of Tables

1 Table containing the cost penalties used in the model 27

2 The table contains data for the reservoir size and maximum value for production and discharge 27

3 Tabel containing the bounds of the hyperparameters for the long short-term memory natural network 28

4 The evaluation results for production prediction 30

5 The evaluation results for water value prediction 35

6 The evaluation results for reservoir level prediction 39

Nomenclature

LTHPS- long term hydro power scheduling DSO kV km FoS

1 Introduction

The world is currently experiencing a growing energy demand. This demand for energy, along with the environmental crisis, has made most of the newly created power plants renewable. The environmental crisis has caused some of the energy consumers to shift from fossil fuel to electric fuel thereby reinforcing the growing energy demand on a worldwide scale. The new renewable energy sources are less stable and have less flexibility than fossil energy sources like coal power plants.

Hydropower, on the other hand, has the advantage that it is not only flexible but also reliable as it can be scheduled when it is needed.. This can help to counteract the variation of renewable power sources and stabilize the power grid. Hydropower plants can have multiple cascading reservoirs which makes planning and scheduling difficult/harder to enact. This makes hydropower scheduling a more computationally time-heavy task. The downside of hydropower scheduling's long completion time can be a) the scheduling can either not be done rapidly enough to take unpredicted changes into consideration, or b)might somewhat lack enough details on the scheduling.

Using machine learning is a great way to lower the computational time, making this a more feasible approach. Machine learning requires a very lengthy and time-heavy computational time when training said system. However after the training is completed , the calculations can be conducted inn only a fraction of the time the training itself required, although it is important to note that it is not necessarily as accurate as the original model.

The aim of this master's thesis is, therefore, to look into if, with the use of machine learning, one could lower the computational time while still keeping a good accuracy. This will be explored through a case study containing a hydropower plant with three cascading reservoirs.

The main research question for this master thesis is: Can the optimization model in long-term hydropower scheduling be substituted by a machine learning model in order to reduce the computational time?

To achieve this will this masters thesis aim to:

- Present a relevant theory for the master's thesis.
- Assess literature surrounding long-term hydropower schedules in order to find a state-of-the-art scheduling model.
- Examine literature concerned with machine learning in hydropower schedules to find a fitting machine learning model to test against the state of art scheduling model.
- Presents the targeted optimization model.
- Present and discuss the results of the machine learning model.
- Presents further work that can be conducted as an explanation of this master thesis and for the research area.

This master's thesis is building on the specialization project that was conducted last semester. Some of the theory and the long-term hydropower scheduling optimization model have been borrowed and extended to fit this master thesis. Some of the figures created for the specialization project have also been transferred.

2 Theory

In this chapter, the theoretical foundations underpinning this thesis are presented. Section 2.1 introduces the fundamental concepts of hydropower plants, their key classifications, construction, and principles of operation. It also provides an overview of hydropower scheduling and elucidates some of the main concepts involved in the process. Section **??** offers an explanation of the Benders Decomposition technique, a powerful tool used in optimization problems. Machine learning, with its different techniques, models, and evaluation methods, is presented in section 2.3. The chapter concludes with a summary and introduction to the following chapter in section 2.5.

2.1 Hydropower Plant

To fully comprehend the complexities of hydropower scheduling, it is crucial to understand the structure and operation of a hydropower plant. Hydropower plants can be categorized based on various criteria, with their capacity for water storage being one of the most distinguishing factors. Broadly, they are classified as either reservoir-based or run-of-river hydropower plants [1].

Reservoir-based hydropower plants have large storage facilities, allowing them to hold substantial amounts of water. The storage capacity enables the plant operators to schedule electricity generation to the most optimal times. Reservoir-based hydropower plant generates electricity by harnessing the potential energy of the water stored in the reservoir from the height difference between the reservoir and the generator. This can be expressed through the following equation [2]:

$$P = \eta * \rho * g * h * Q \tag{1}$$

Where P is the power output, η is the efficiency of the plant, ρ is the density of water, g is the acceleration due to gravity, h represents the height difference utilized to generate electricity, and Q is the flow rate of water through the turbines.

On the other hand, run-of-river hydropower plants have little to no storage capability. As the name suggests, they rely primarily on the river's natural flow to generate electricity. The power generated in these plants results from the kinetic energy in the water stream as it flows through the turbines. This kinetic energy conversion can be represented using the equation [2]:

$$P = \frac{1}{2} * \eta * \rho * A * v^2 \tag{2}$$

In this equation, P is the power output, η is the plant efficiency, ρ is the water density, A is the cross-sectional area of the stream, and v is the velocity of the water flow.

A typical reservoir-based hydropower plant can be classified into five key components: the reservoir, the tunnel, the turbine and generator, the discharge tunnel, and the discharge pool.

The reservoir stores and holds the water naturally flowing into an area [1]. The reservoir can take the form of a natural pond or have an artificial body created using a dam. The geographical characteristics of the location determine the choice of dam type and size. A regulation hatch is incorporated to release the water for maintenance or emergencies. To prevent structural damage to the dam, a spillway is commonly incorporated to discharge the water safely in the event of overflow or spillage. Any overflow or spillage from the reservoir results in the loss of potential energy and revenue for the plant.

The water from the reservoir to the turbine is transported through the water tunnel, referred to as the tunnel. The design and roughness of the tunnel play a crucial role in the plant's efficiency since bends, turns, and rough surfaces can lead to efficiency losses. In figure 8 can, a vertical tunnel connected to the main water tunnel can be seen, this is called the surge shaft . The surge shaft relieves pressure and allows the water to oscillate and lose momentum when the plant is shut down, reducing potential damage to the plant [3]. Additionally, can this tunnel be placed strategically to allow unregulated water to flow into the system.

The turbine and generator constitute the core of the hydropower plant. Here, the potential energy of the water is converted into electrical energy [4]. The water pressure propels the turbine, which rotates the generator to produce electricity. Different types of turbines are used based on the height difference utilized in the plant and the volume of water flow. Different types of turbines are used for different height differences. Each turbine will have different efficiency depending on the water realized [5]. The difference in efficiency can be minimized by having multiple pairs of turbines and generators to minimize the efficiency loss by utilizing the synergy between them.

After passing through the turbine, the water is guided to the discharge pool via the discharge tunnel. The tunnel is positioned lower than the reservoir to prevent the creation of a vacuum in the water tunnel, which would negatively affect the plant's efficiency. The discharge pool signifies the end of the hydropower plant, and it can take the form of another reservoir, an unregulated river, or a direct outlet to the sea.

The overall efficiency of a hydropower plant is a combination of losses throughout the system, including the tunnel, turbine, and any vacuum created. This can be represented by the following equation:

$$\eta = 1 - \eta_{tunnel} - \eta_{turbine} - \eta_{vacuum} - ... \tag{3}$$

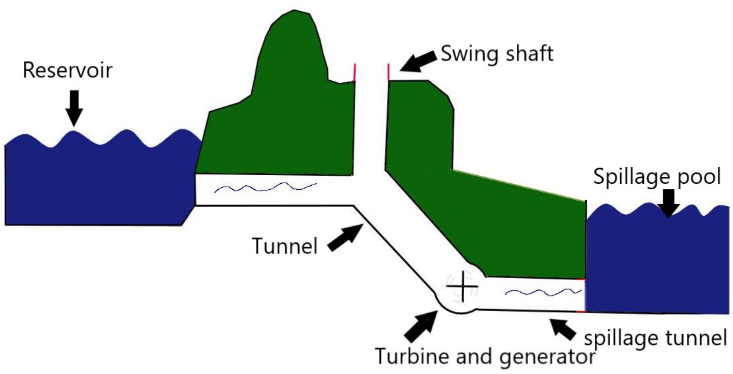

Figure 1: Cross-sectional view of a typical hydropower plant

This section provides a basic understanding of how a typical reservoir-based hydropower plant functions, laying the groundwork for subsequent discussions on hydropower scheduling.

2.1.1 Hydro power scheduling

The scheduling of reservoir-based hydropower plants poses a significant optimization challenge. As these plants can store water for future use, determining the optimal use of the stored water becomes a scheduling problem [6].

The complexity of this problem is further exacerbated in cases where there are a network of cascading hydropower plants, where the output of one plant serves as the input for another. The timing and volume of water release must be carefully coordinated to ensure optimal operations across the entire system. To manage these complexities, hydropower scheduling is often divided into several components, each serving different time horizons: long-term (1-5 years), mid-term (1-52 weeks), short-term (1-7 days), and real-time.

Long-term hydropower scheduling focuses on strategic decisions and provides a broad schedule of the reservoir level and water value on a weekly basis. This is crucial for overall system planning and ensuring water resources are used optimally. Water value is the value the system would obtain if it obtained one additional unit of water. This serves as an essential determinant of when and how to produce power. The output of the long-term hydropower scheduling serves as input to the mid-term scheduling. The midterm hydropower scheduling provides greater details, often providing daily time steps. The mid-term scheduling is often considered the binding between long-term and short-term scheduling. Both the mid-term and long-term scheduling processes must incorporate uncertainty modeling into their calculations. This is due to unpredictable variables like rainfall and further demand.

The output of the mid-term scheduling is used as input to the short-term scheduling. The short-term scheduling operates on more certainty regarding demand and weather forecast, allowing a detailed hourly plan to be generated. This plan is then taken to the power market, where the final decision on how much and when to sell is made. If the condition changes, the day of production will the final production plan be changed with real-time scheduling.

2.1.2 Load Shedding

Load shedding is a tool used when there is insufficient power to satisfy the demand [3]. Load shedding works by reducing or cutting off the power to consumers or areas. This can be planned in cases like maintenance or if there is extreme weather or lack of power in the market. The goal of load shedding is to reduce the stress on the power grid to prevent it from overloading. The downtime can vary from minutes to hours [7]. Load shedding is usually tried to minimize the inconvenience of the society, therefore, planned load shedding is often done at night, and unplanned load shedding often follows a prioritized list where critical buildings like hospitals are prioritized [8].

2.2 Benders Decomposition

When solving large-scale optimization problems, the overall complexity might be very high. To lower the complexity, a decomposition technique might be used. There exist Multiple decomposition techniques like Bender's Decomposition, Dantzig–Wolfe Decomposition, and Lagrangian Relaxation [9]. Benders Decomposition is particularly notable for its efficacy in handling high-dimensional, nonlinear problems.

Benders decomposition was first presented in 1962 in a paper by J. F. Benders [10].

Benders decomposition divides an optimization problem into one master problem and several subproblems. The master problem is then solved, and the complicated value is sent down to the subproblem as a constant instead of a variable. The subproblems are then solved, providing information to the master problem and generating new cuts to the master problem, this can be seen in figure 2. These prosess continue iteratively until a solution is found. Benders decomposition can be shown mathematically as follows [11].

Original problem

$$min \sum_{i=1}^{N} c_i x_i + \sum_{j=1}^{M} d_j y_j \tag{4}$$

Subject to

$$\sum_{i=1}^{N} a_{l,i} x_i + \sum_{j=1}^{M} e_{l,j} y_j ; l = 1, .., q \tag{5}$$

$$0 \leq x_i \leq x_i^{up} ; i = 1, ..., N \tag{6}$$

$$0 \leq y_j \leq y_j^{up} ; j = 1, ..., M \tag{7}$$

Is transformed into:

Master problem

$$min \sum_{i=1}^{N} c_i x_i + \alpha \tag{8}$$

subject to

$$\sum_{j=1}^{M} d_j y_j^k + \sum_{i=1}^{N} \lambda_i^k (x_i - x_i^k) \leq \alpha \tag{9}$$

$$0 \leq x_i \leq x_i^{up} ; i = 1, ..., N \tag{10}$$

$$\alpha \geq \alpha^{low} \tag{11}$$

subproblem

$$min \sum_{j=1}^{m} d_j y_j \tag{12}$$

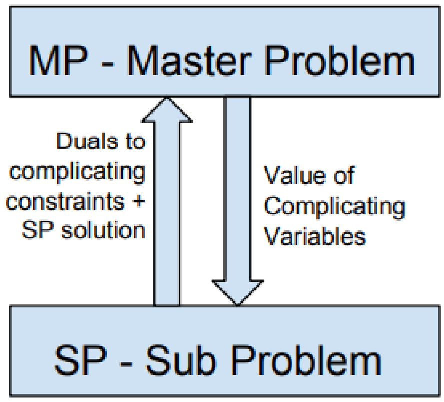

Figure 2: Information flow in Benders Decomposition.

Subjected to

$$\sum_{i=1}^{N} c_{l,i}x_i + \sum_{j=1}^{M} e_{l,j}y_j \leq b^l; l = 1, ..., q \tag{13}$$

$$0 \leq y_j \leq y_j^{up}; j = 1, ..., M \tag{14}$$

$$x_i = x_i^k : \lambda_i; i = 1, ..., N \tag{15}$$

Where c, d, a, and e are constants, x and y are variables, λ is the dual value of the complicating constraint, l, I, and j are indexes going respectively from 1 to q, N and M, y^{up} and x^{up} are the upper limit of x and y while α^{low} is the lower limit, k is the iteration, and α is the connection variable taking the cuts into the optimal value.

2.3 Machine Learning and Neural Networks

Machine learning(ML) is a subcategory of artificial intelligence [12]. ML recognizes patterns, makes decisions, and improves with experience. ML is divided into four major parts; Supervised, unsupervised, semi-supervised, and reinforced learning [13]

Supervised learning is when the model is trained on labeled datasets, where the input data and corresponding target variables have a known relationship [14]. The primary objective is to learn a mapping function capable of predicting the target variables based on the input data. In contrast, unsupervised learning concerns the training of models on unlabeled data, with the principal goal being to recognize trends or patterns in the data without prior knowledge. Semi-supervised learning combines both approaches, where the model is trained on labeled and unlabeled data. Reinforcement Learning differs somewhat from the others. It operates around the concept of a system of rewards and punishments. An agent interacts with an environment, receiving feedback in the form of punishments or rewards for these interactions. Through this feedback mechanism, the agent learns the optimal strategy to maximize long-term rewards based on actions.

2.3.1 Linear Regression

Linear regression is an ML technique defined as an equation that determines the linear relationship between a single dependent variable Y, and one or more independent variables [15]. The simplest form of the linear regression equation is often called classic linear regression, it has one dependent and one independent variable and is defined by the formula:

$$Y = \beta_0 + \beta_1 * X + \epsilon \tag{16}$$

where:

Y is the dependent variable. X is the independent variable. β_0 is the intercept of the line at Y when $X = 0$. β_1 is the slope of the line. ϵ is the error term for anything that may affect the dependent variable other than the independent variable, including measurement errors. Determining the line is a method to minimize the differences of the observed known value and the predicted value. The common method to achieve this is known as the least squares method, which can mathematically be represented as:

$$min_{\beta_0,\beta_1} \sum_{i=1}^{n} (Y_i - \beta_0 - \beta_1 * X_i)^2 \tag{17}$$

There are two main types of linear regression, classical linear regression and multiple linear regression. The main difference between them is how many independent variables they use to make a prediction. Multiple linear regression can be represented by this equation:

$$Y = \beta_0 + \beta_1 * X_1 + \beta_2 * X_2 + ... + \beta_p * X_p + \epsilon \tag{18}$$

Each coefficient β_i now represents the change in the average value of Y due to a one-unit change in X_i, holding all other variabels fixed.

2.3.2 Neural Network

The Neural network is under the ML category of supervised learning. It is inspired by how the nervous system of the human brain works [16]. Neural networks are constructed as a graph consisting of an input layer, one or several hidden layers, and an output layer as shown in figure 3. Each layer contains a series of neurons, or nodes, and each node in one layer is connected to each node in the next layer. To know how often the information should be passed from a node to another is a number called weights, this number often comes as fraction. The weights of these connections, are denoted as $w^l_{j,k}$ where j and k are the neurons the weight is connected from and to and l is the layer number, are learned during the training process. The number of neurons in the input layer corresponds to the number of input features, and the number of neurons in the output layer corresponds to the number of output variables. The number of neurons in the hidden layers is often chosen based on trial and error or an optimization algorithm.

There are several types of neural networks, each with their unique architecture and use cases. The most common types are Feedforward Neural Networks (FNNs), Convolutional

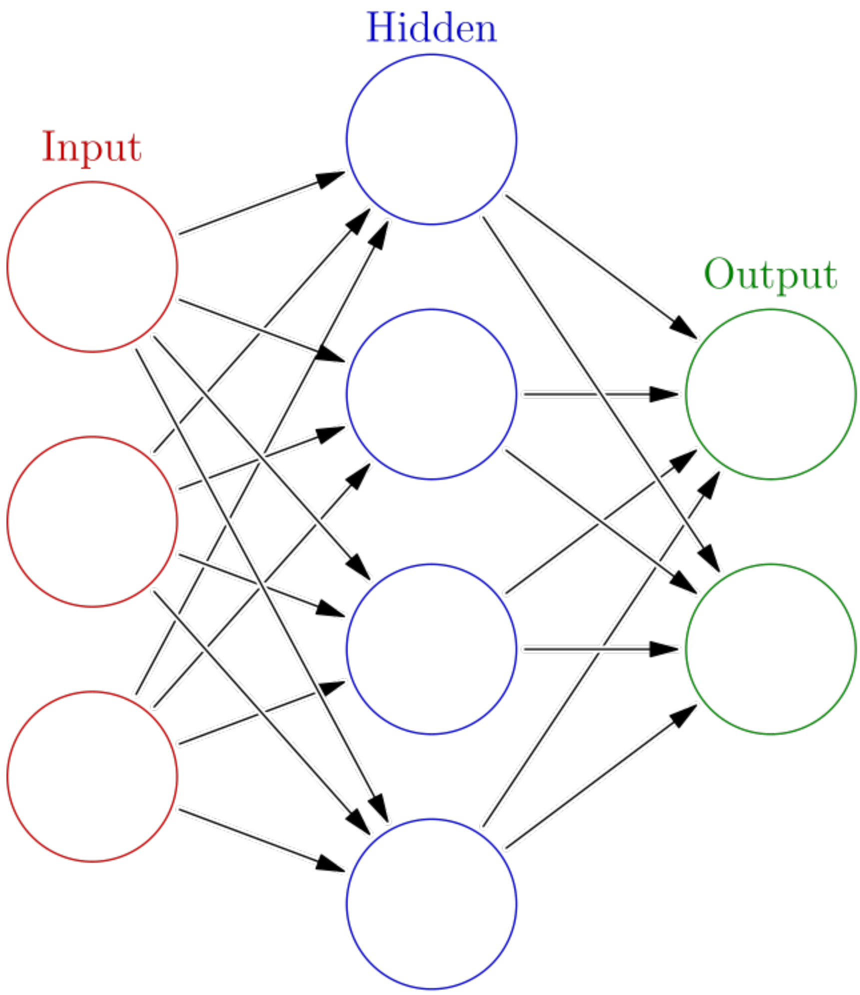

Figure 3: This is an example of how a Neural Network can look

Neural Networks (CNNs), and Recurrent Neural Networks (RNNs). Each type has its strengths and weaknesses, so the choice of type will dependend on the problem, and the available data, one overall weakness of the neural network is the amount of data neded to make good predictions. Feedforward Neural Networks (FNNs) is the simplest type of neural network, FNNs consist of an input layer, one or more hidden layers, and an output layer, each neuron in a layer is connected to every neuron in the next layer [17]. The data flows from the input layer to the output layer without looping back. Convolutional Neural Networks (CNNs) are designed to automatically and adaptively learn spatial hierarchies of features from the input data [18]. The main difference between FNNs and CNNs is that CNNs take advantage of the hierarchical pattern in data and assemble more complex patterns using smaller and simpler patterns. Recurrent Neural Networks (RNNs) are used when there is a sequential relationship in the input data, such as a time series or a sentence [17]. Unlike FNNs and CNNs, which have no memory and process each input independently, RNNs can use their memory to process sequences of inputs, meaning output for a given input can depend on previous inputs.

2.3.3 Long Short-Term Memory (LSTM)

Recurrent Neural Networks are suffering from challenges in learning long-term dependencies due to the vanishing or exploding gradients problem [19]. The vanishing gradient problem is when the weights in the network don't get updated due to the gradient value being too small. Normally the weights are updated proportionally to the gradient value. If this becomes too small the problem may stop the training even if it doesn't have the optimal solution to the problem. This makes it hard for the RNN to learn and adjust its parameters to capture dependencies in sequences that span more than a few steps. Long Short-Term Memory (LSTM) units are a type of RNN architecture designed to combat this issue and effectively capture long-term dependencies in sequence data.

The key idea behind LSTM units is the cell state. The cell state can provides information from earlier steps in the sequence to later steps. Gates control what information enters, remains in, and leaves the cell state. Each LSTM cell consists of an input gate, a forget gate, an output gate, and a cell state. The gates are used to regulate the information flow in the LSTM cell—deciding what information to keep or discard. The cell state acts as a "conveyor belt" carrying memory from earlier in the sequence. Through these mechanisms, the LSTM cell can learn to recognize important patterns over time, forget irrelevant data, and make predictions based on the relevant information it has retained.

2.3.4 Bayesian Optimization

Bayesian optimization is a model-based method for global optimization of black-box functions, particularly useful for expensive to evaluate functions or those with no closed-form expression or gradient information. This technique becomes very efficient when the number of function evaluations is limited due to constraints such as time or cost.

The main components of Bayesian optimization are:

- **Surrogate Model:** This is a probability model that is easy to optimize and is used to approximate the unknown function. Gaussian Processes (GPs) are commonly used due to their capacity to provide a measure of uncertainty.
- **Acquisition Function:** This is a utility function built from the surrogate model that provides a measure of which point in the input space should be evaluated next. The acquisition function trades off exploitation, i.e., sampling where the surrogate model predicts high performance, and exploration, i.e., sampling where the uncertainty is high. Examples of acquisition functions include Expected Improvement (EI), Probability of Improvement (PI), and Upper Confidence Bound (UCB).

The Bayesian Optimization process follows these steps:

1. Initialize with a set of points and corresponding function evaluations.
2. Fit the surrogate model to the current set of evaluations.
3. Find the point that maximizes the acquisition function.
4. Evaluate the objective function at this new point and augment the existing set of points.

5. Repeat steps 2-4 until a stopping criterion is met, e.g., a maximum number of function evaluations, a maximum time, or the acquisition function is below a threshold.

The performance of Bayesian Optimization heavily depends on the choice of the surrogate model and the acquisition function, and the way in which these components are combined.

2.3.5 Ensemble Models

Ensemble models is a machine learning technique that combines the prediction of different ML models to improve the overall performance [20]. This can lead to improved predictive performance compared to any single model. The prediction of the different models is aggregated into one ensembled model. This technique minimizes the error one individual model may have by utilizing the diversity of the different models. Each ML model will have some variation on the prediction, which can be reduced when combined as long as no model is correlating. With lower variation, the prediction can be more stable and reliable. This can be shown mathematically with these equations:

The variance of the ensemble's average prediction is given by:

$$Var(\bar{y}) = Var\left(\frac{1}{N}\sum_{i=1}^{N} y_i\right) \tag{19}$$

where N is the number of models, ρ is the correlation between any two models and σ^2 is the variance. this equation is equal to:

$$Var(\bar{y}) = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} Cov(y_i, y_j) \tag{20}$$

When $i = j$ will $Cov(y_i, y_j)$ be the variance of the model σ^2, this occurs N times. When $i \neq j$ will $Cov(y_i, y_j)$ be $\rho\sigma^2$, this will occur $N^2 - N$ times.

The equation will then be:

$$Var(\bar{y}) = \frac{1}{N^2}[N\sigma^2 + (N^2 - N)\rho\sigma^2] \tag{21}$$

Simplifying this:

$$Var(\bar{y}) = \frac{\sigma^2}{N} + (1 - \frac{1}{N})\rho\sigma^2 \tag{22}$$

From equation 22, the variation of the ensembled model can be calculated. The best case for the ensembling model is when the models are completely unrelated $\rho = 0$. This gives a variance of $\frac{\sigma^2}{N}$. The worst case is when the model is 100% correlated $\rho = 1$. This in turn gives an variance of σ^2, the same as that of a single model. The ensemble model's variance is always as good or better than the variance of a single model.

2.3.6 Performance Measuring

The overarching goal of this study is to assess the potential of machine learning algorithms in long-term hydropower scheduling, thus reducing the overall time required. To effectively evaluate the efficiency of machine learning techniques relative to traditional optimization methodologies, two principal factors must be considered: the computational time and the accuracy of the program.

Computational time, denoting the duration a computer algorithm requires to complete its task, provides an efficient and practical way of comparing distinct algorithms [21]. The hardware setup will impact the performance of computational time, it is, therefore, essential to run the algorithms on the same or equal computers when comparing.

To objectively measure the efficacy of the machine learning model, various performance metrics are employed, including the Mean Squared Error (MSE), Root Mean Squared Error (RMSE), mean absolute error(MAE), mean absolute percentage error (MAPE), and Root Mean Squared Logarithmic Error (RMSLE). This research will primarily utilize the first three metrics: MSE, RMSE, and MAE. Using diverse evaluation techniques allows for a more comprehensive analysis, considering each metric has unique strengths and limitations.

The Mean Squared Error (MSE) is computed as follows [22]:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(\hat{r_i} - r_i)^2 \tag{23}$$

where N denotes the total number of predicted values, $\hat{r_i}$ represents the predicted values, and r_i is the actual values. The MSE computes the average of the squares of the errors, emphasizing larger errors due to the squaring operation.

The Root Mean Squared Error (RMSE), a derivative of MSE, is expressed as [23]:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{r_i} - r_i)^2} \tag{24}$$

where N is the total number of predicted values, $\hat{r_i}$ signifies the predicted values, and r_i represents the actual values. The RMSE is the square root of the MSE, mitigating the heavy penalization of larger errors exhibited by MSE, thus providing a more balanced measure of model performance.

This equation can calculate the mean absolute error [24]:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|r_i - \hat{r_i}| \tag{25}$$

where N is the total number of predicted values, $\hat{r_i}$ is the predicted values, r_i signifies the actual values. MAE gives the mean absolute value between the predicted values and the measured data without considering the direction of the error.

mean absolute percentage error (MAPE) can be calculated through this equation:

$$MAPE = 1 - \frac{1}{N}\sum_{i=1}^{N}|\frac{r_i}{r_{max}}r_i - \frac{\hat{r_i}}{\hat{r_{max}}}| \tag{26}$$

where N is the total number of predicted values, $\hat{r_i}$ is the predicted values, $\hat{r_{max}}$ is the highest predicted value, r_i signifies the actual values, r_{max} is the highest nesting value. MAPE gives the mean absolute percentage value between the predicted values and the measured data without considering the direction of the error.

2.4 Syntetic data

To have a well-trained ML, an extensive amount of data is needed. This is not always possible to obtain from industrial partners. Therefore, algorithms to obtain representative synthetic data are made. European data protection supervisor defines synthetic data as "synthetic data is artificial data that is generated from original data and a model that is trained to reproduce the characteristics and structure of the original data" [25]. The synthetic and original data should have a similar structure and provide good data for ML training and testing. To generate the synthetic data, different models and technics can be used. This thesis focuses on Time-series Generative Adversarial Networks (TimeGAN) [26]. TimeGan is a good model for generating time-series data. This model strives to give flexibility from an unsupervised paradigm while having the control given by supervised training. From Yoon, Jarrett, and Schaar's research, this model outperforms other state-of-the-art models. The model has four network components, a recovery function, an embedding function, a sequence generator, and a sequence discriminator. The autoencoding components (first two) and the adversarial components (last two) are trained jointly so the TimeGan simultaneously learns to encode features, generate representations and iterate across time. The embedding network provides the latent space, a lower dimensional space where the input data is mapped out. The adversarial network operates within this space, distinguishing between real and synthetic data. Through a supervised loss, synchronizing both real and synthetic data, this helps to make the synthetical data more realistic.

2.4.1 Data types

Machine Learning (ML) leverages various data categories, differentiated by their characteristics and format. These include Numeric Data, Categorical Data, Text Data, Image Data, and Time Series Data. The data used in this thesis are primarily Time Series Data.

Time series data consists of a series of data points collected at consecutive time intervals [27]. These data points represent the same variable and are chronologically recorded, enabling the study of changes to the variable over time. Time series data can either be regular or irregular, depending on the uniformity of the time intervals at which the data points are collected. In regular time series data, data points are collected at evenly spaced time intervals. The regularity of the time intervals allows for more straightforward analysis and forecasting as patterns such as trends and seasonality can often be more easily identified. Irregular time series data are the data points collected at irregular time intervals. Analysis and forecasting for irregular time series data can be more challenging due to the inconsistent intervals between data points.

2.5 Theory conclusion

This chapter is the main theory used in this master's thesis. The next chapter presents a literature review to find the most suitable machine learning model and hydropower scheduling technique to answer the research question.

3 Literature review

This master's thesis aims to investigate the performance of machine learning as a substitute for the optimization techniques used in long-term hydroelectric power scheduling (LTHPS). To achieve this goal, it is vital to conduct a thorough literature review. The literature review for this thesis has three main goals: to find the state-of-the-art optimization techniques for LTHPS, to identify which machine learning techniques are being used in the field, and to determine how different machine learning techniques have performed. To ensure the research is most relevant to the field, it is important to compare the results of machine learning to the state-of-the-art LTHPS optimization techniques while considering the main breakthroughs in LTHPS optimization techniques. Different machine learning techniques have been used in the field; therefore, it is important to ensure that the research in this thesis uses a new technique or approach to further advance the research in the field. In section 3.1 will look at some of the most important hydro power scheduling techniques. The different types of machine learning techniques used in hydropower scheduling are looked at in section 3.2. This chapter ends with a conclusion in section 3.3

3.1 Hydro power scheduling

In paper [28] is one of the earliest publications on hydropower scheduling using water value as far as the author of this thesis knows. Here the motivation for having a water value is to better schedule the water resources against the thermal plants. The water value takes in the losses from different discharge flows and different reservoir levels. This method follows the principles of stochastic dynamic programming (SDP) and is advised to have 30 years of data to do the best calculation. The method is primarily applicable to a single reservoir system, but if the system has multiple reservoirs, and it can be approximated that there is no unnecessary spillage, the water value can be calculated for the aggregated system. In [29] there are two new methods for including multi reservoirs hydropower scheduling presented that are also based on SDP. The first is called the one-at-a-time method, and this method breaks the multi-variable problem into a series of one-state subproblems which is solved using dynamic programming. The second method is called the aggregation decomposition method, this method breaks the problem into a series of two-stage sub-problems and solves it with dynamic programming. Turgeon concluded that the aggregated method is a better operating policy than the one-at-a-time method, and emphasizes that the method will only increase linearly with added reservoirs. To better solve problems with multiple reservoirs, Pereira and Pinto presented a new method in this paper [30]. The new approach is called stochastic dual dynamic programming (SDDP), it is based on the approximation of the expected-cost-to-go functions of SDP by a piecewise linear function. The new approach obtains the dual solution from the optimization problem at each stage, which corresponds to the cuts from benders decomposition. In this paper [31] a new model to solve long-term multi reservoirs hydropower scheduling is proposed. The new model is referred to as a scenario fan simulator (SFS). This approach combines optimization techniques with simulation. The SFS method represents the stochastic reservoir inflow for further weeks as a fan of scenarios in a sequence of two-stage stochastic linear programming (SLP) problems. The Benders Decomposition is used to solve each SLP problem efficiently, the solution is then saved. The simulation approach consists of solving the SLP problem for all first-week inflow scenarios for all time steps, the solution is stored. For week t+1 the optimal end value of t will be the starting point of t+1. Helseth, Mo and Warland conclude that the new method is competitive with an established model based on SDDP for market simulation purposes, provides better results in terms of socioeconomic surplus, and handles extreme weather slightly better. A drawback of this model is the computation time, but this can be eliminated by running the different scenarios on parallel processors. This method is used in the new market model made by SINTEF energy [32]. This market model is aiming to give a detailed representation of the hydropower system while not relying on the aggregation of the hydropower system. SINTEF is one of Europe's largest independent research organizations [33]. SINTEF was established in 1950 in Trondheim and has today become a world-leading research institute. SINTEF energy created one of the existing models used in the market called EMPS model [34].

3.2 Machine learning in Hydro power scheduling

There are several different machine learning models and approaches that can be used to solve the LTHPS problem. Bordin, Skjelbred, Kong, and Yang conducted state-of-the-art research in 2020 with a main focus on short-term hydroelectric power scheduling. The article looked at both river-based and reservoir-based hydroelectric power production. However, it is worth noting that this literature review will focus primarily on reservoir-based hydroelectric power production. The study reviewed 23 different papers, ten of which used machine learning for river-based hydroelectric power scheduling, and three of which used machine learning to predict the inflow.

3.2.1 Reinforcement learning

Riemer-Sørensen and Rosenlund take a look at Deep Reinforcement Learning for Long Term scheduling [35]. The challenge the article is looking at is deciding when to sell or store water given the electricity price on a week-to-week basis. The reinforcement machine learning is trained with real and synthetic data corresponding to 5700 years. The author's results shows that the reinforcement model ends up with a different schedule than the classical optimization models, the author then concludes that this model can be a good supplement to the traditional optimization tools, but not a replacement as of now. In this article [36] reinforcement learning is used on a long-term hydropower schedule, the time horizon is one year with the predicted value being water released on a daily timescale. The released water goes into a river that contains multiple runs of river power plants and provides water to irrigated agricultural areas. The author of this article concludes that RL can be an alternative to mitigate the use of SDP.

3.2.2 Neural Network

Reference [37] looks into an artificial neural network(ANN) as a possible model to substitute for optimizing hydropower reservoir management. To train and test the ANN, 43 years of inflow data was used. The model predicted hatch release, and where compared to a simulation model using stochastic dynamic programming (SDP). To evaluate the results R^2, RMS and RE are used. Haddad and Alimohanmmadi conclude that ANN is a good substitution for the simulation model for this case. The ANN predicted very close results to the simulation model. Reference [38] trained and tested an ANN with 41 years of monthly inflow data for one reservoir. The study aimed to estimate three variables in reservoir management: monthly inflow, monthly evaporation, and monthly storage. Two ANN methods were used to test this, the feed-forward backpropagation and the radial basis function. KILINÇ and CIĞIZOĞLU conclude that both methods provide satisfactory estimations considering the plots and performance evaluation.

Reference [39] artificial neural network for predicting the reservoir level of cascaded reservoirs was looked at. The reservoir is supplying water to society, works as a flood stop, and provides water to hydropower plants. The paper looks at three plants where the ANN has been trained and tested with 29, 43, and 23 years of monthly historical data for the different plants. The performance on training was 95%, 69%, and 98% while the performance on the testing was 97%, 75%, and 97% for the respective plants. Abdulkadir, Salami Sule, and Adeyemo conclude that the neural network has reliable results, and is a very versatile forecasting tool in reservoir management modeling.

This study [40] looks at two multi-purpose reservoirs. The purposes looked at in this article include irrigation, water supply, industries, and hydropower production with more. Two ANN were trained and tested on 26 and 40 years of data respectively, they performed 95% and 69% prediction rate on the training part, while the performance on the testing data was 97% and 75% for the two ANN respectively. Abdulkadir, Sule, and Salami conclude that the ANN is a very versatile toll in reservoir management modeling.

The paper [41] demonstrates how implicit stochastic optimization combined with deep neural networks can be used to find optimal reservoir operating. The case study contained two cascading reservoirs. The DNN was trained with a daily timescale with a total of 8299 days of inflow and price data. Concludes that the model is from an operational standpoint computationally inexpensive, and may be utilized in combination with other technics.

3.2.3 Other approaches

The research goal of this study [42] is to propose a new approach to trend assessment. To test the new approach the inflow data of 30 hydropower plants (HPPs) is used. The new approach linear mowing window (LMW) took a fixed time of 30 years of the data, used linear regression on this, for then to move the window one year. Concludes that LMW is a more reliable technique than linear regression.

3.3 Conclution of literature review

In section 3.1 different hydropower scheduling technique using SDP, SDDP, and SFS was described. From the section, it can be seen that the scheduling techniques evolved over time to incorporate more details for multi-reservoir hydropower scheduling problems. As far as the author of this thesis knows, the newest technique is Scenario fanning simulation. This is further reinforced by the fact that SINTEF Energy included this technic in their new market model. SINTEF energy created the EMPS model which is one of the models used today. This implies that the SFS technique will be used for long-term hydropower scheduling in the next years to come. The downside with SFS was that it can be a bit time-consuming which might be solved by implementing machine learning. The state-of-the-art hydropower scheduling technique and the one that will be looked at in this thesis is the SFS technique.

Section 3.2 went through some of the machine learning models used in hydropower scheduling. Here can it be seen that Naural Network and reinforced machine learning were two of the most used machine learning technique for hydropower scheduling. Reinforcement learning did not perform well in this field, but different types of Neural Networks did produce promising results. Therefore a neural network model will be the best fit for this master's thesis. As far as the author of this master thesis knows has Long Short-term memory Neural Networks not been used in hydropower scheduling, so this is, therefore, the choice of the neural network model used in this thesis.

In the next chapter, the mathematical model based on the scenario fanning simulation model will be presented, it will go into further detail for the different parameters and variables of real-world equivalent.

4 Mathematical model

In this chapter, the mathematical model is presented in section 4.1. Section 4.2 explains how the parameters and variables are connected to a hydropower system. This section also explains some of the simplifications used in this model, their reason, and how they are predicted to impact the results. The model presented in this chapter is an SDDP model. Section 4.3 explains how this model is further evolved into an SFS model by combining optimization with simulation. In section 4.4 a summary of this chapter is presented.

4.1 Optimization model

To test how the machine learning model works well, a model for long-term hydropower scheduling will be made as a base case for the performance of the machine learning modes. The optimization model for long-term hydropower scheduling uses a combination of Benders Decomposition and scenario simulation. This is the state-of-the-art hydropower scheduling technique shown in section 3.1, and will be shown in the following sections. Benders decomposition splits the scenario into a first and second stage as explained in 2.2.

4.1.1 Notations

Sets first stage:

$$G = \text{number of generators} \tag{27}$$

$$C = \text{number of cuts} \tag{28}$$

$$F = \text{number of resevoirs} \tag{29}$$

Index first stage:

$$g \in G \tag{30}$$

$$c \in C \tag{31}$$

$$f \in F \tag{32}$$

First stage variables:

$$P_g = \text{Production from generator g} \tag{33}$$

$$S_g = \text{Spillage from the reservoir connected to generator g} \tag{34}$$

$$B_g = \text{Bypass water connected to generator g} \tag{35}$$

$$L = \text{Load shed} \tag{36}$$

$$R_g = \text{Reservoir level in the reservoir connected to generator g} \tag{37}$$

$$D_g = \text{Discharge water out of generator g} \tag{38}$$

$$Re_g = \text{Released water from the reservoir to the water tunnel going to generator g} \tag{39}$$

$$\alpha = \text{Penalty factor used in benders to consider the second stage} \tag{40}$$

Parameters first stage:

$$r_g^{Max} = \text{The upper limit for the reservoir level connected to generator g} \tag{41}$$

$$p_g^{Max} = \text{The maximum production for generator g} \tag{42}$$

$$d_g^{Max} = \text{The maximum discharge from generator g} \tag{43}$$

$$i_g^{R} = \text{The regulated inflow to the reservoir connected to generator g} \tag{44}$$

$$i_g^{U} = \text{The unregulated inflow to the Watergate way connected to generator g} \tag{45}$$

$$r_g^{Int} = \text{The initial level for the reservoir connected to generator g} \tag{46}$$

$$l = \text{The load needed to be met} \tag{47}$$

$$conv = \text{The conversion factor to convert } m^3 \text{to Mwh} \tag{48}$$

$$cut = \text{Cuts generated from second stage} \tag{49}$$

$$c^{shed} = \text{The cost of load shedding} \tag{50}$$

$$c^{spil} = \text{The cost of spillage} \tag{51}$$

$$c^{by} = \text{The cost of bypassing water} \tag{52}$$

$$f_{g,f} = \text{Binary variable telling which generator obtains the discharge water from which reservoir} \tag{53}$$

Sets second stage:

$$G = \text{number of generators} \tag{54}$$

$$W = \text{number of Weeks} \tag{55}$$

$$S = \text{number of Scenarios} \tag{56}$$

$$F = \text{number of reservoirs} \tag{57}$$

Index second stage:

$$g \in G \tag{58}$$

$$w \in W \tag{59}$$

$$s \in S \tag{60}$$

$$f \in F \tag{61}$$

Second stage variables:

$$P_{w,g} = \text{Production in week w from generator g} \tag{62}$$

$$S_{w,g} = \text{Spillage in week w from the reservoir connected to generator g} \tag{63}$$

$$B_{w,g} = \text{Bypass water in week w connected to generator g} \tag{64}$$

$$L_w = \text{Load shedding in week w} \tag{65}$$

$$R_{w,g} = \text{Reservoir level in week w for the reservoir connected to generator g} \tag{66}$$

$$D_{w,g} = \text{Discharge water in week w out of generator g} \tag{67}$$

$$Re_{w,g} = \text{Released water in week w from the reservoir to the water tunnel going to generator g} \tag{68}$$

Parameters second stage:

$$r_g^{Max} = \text{The upper limit for the reservoir level connected to generator g} \tag{69}$$

$$r^{Min} = \text{The lower limit for the aggregated system} \tag{70}$$

$$p_g^{Max} = \text{The maximum production for generator g} \tag{71}$$

$$p_g^{min} = \text{The minimum production for generator g} \tag{72}$$

$$d_g^{Max} = \text{The maximum discharge from generator g} \tag{73}$$

$$i_{w,g}^{R} = \text{The regulated inflow in week w to the reservoir connected to generator g} \tag{74}$$

$$i_{w,g}^{U} = \text{The unregulated inflow in week w to the Watergate way connected to generator g} \tag{75}$$

$$r_g^{Int} = \text{The initial level for the reservoir connected to generator g} \tag{76}$$

$$l_h = \text{The load in week w} \tag{77}$$

$$conv = \text{The conversion factor to convert } m^3 \text{to Mwh} \tag{78}$$

$$cut = \text{Cuts generated from second stage} \tag{79}$$

$$c^{shed} = \text{The cost of load shedding} \tag{80}$$

$$c^{spil} = \text{The cost of spillage} \tag{81}$$

$$c^{by} = \text{The cost of bypassing water} \tag{82}$$

$$f_{g,f,w} = \text{Binary variable telling which generator obtains the discharge water from which reservoir} \tag{83}$$

4.1.2 Mathematical model

$$min \sum_{g \in G} c^{spil} * S_g + c^{by} * B_g + c^{shed} * L + \alpha \tag{84}$$

s.t

$$R_g \leq r_g^{Max}, g \in G \tag{85}$$

$$\sum_{g \in G} R_g \geq r^{Min} \tag{86}$$

$$P_g \leq p_g^{Max}, g \in G \tag{87}$$

$$P_g \leq p_g^{Min}, g \in G \tag{88}$$

$$D_g \leq d_g^{Max}, g \in G \tag{89}$$

$$\sum_{g \in G} P_g + L = l \tag{90}$$

$$r_g^{Int} - S_g - Re_g + i_g^R + f_{g,f} D_g = R_g, g \in G, f \in F \tag{91}$$

$$Re_g - B_g + i_g^U = D_g, g \in G \tag{92}$$

$$D_g * conv = P_g, g \in G \tag{93}$$

$$\alpha \geq cut \tag{94}$$

$$min \sum_{w \in W} \sum_{g \in G} c^{spil} * S_{w,g} + c^{by} * B_{w,g} + \sum_{w \in W} c^{shed} * L_w \tag{95}$$

s.t

$$R_{w,g} \leq r_g^{Max}, w \in W, g \in G \tag{96}$$

$$\sum_{g \in G} R_g \geq r^{Min} \tag{97}$$

$$P_{w,g} \leq p_g^{Max}, w \in W, g \in G \tag{98}$$

$$P_{w,g} \leq p_g^{min}, w \in W, g \in G \tag{99}$$

$$D_{w,g} \leq d_g^{Max}, w \in W, g \in G \tag{100}$$

$$\sum_{g \in G} P_{w,g} + L_w = l, w \in W \tag{101}$$

$$R_{w,g} - S_{w,g} - Re_{w,g} + i_{w,g}^R + f_{w,g,f} D_{w,g} = R_{w,g}, w \in W, g \in G \tag{102}$$

$$Re_{w,g} - B_{w,g} + i_{w,g}^U = D_{w,g}, w \in W g \in G \tag{103}$$

$$D_{w,g} * conv = P_{w,g}, w \in W g \in G \tag{104}$$

4.2 System

This system only looks at the scheduling of a cascading hydropower plant. The scheduling only looks at meeting the planned demand for the system. There is therefore no need to look at other energy producers or transfer lines in the model. The system borders can be seen in 4. The load inside the system borders is the load that the hydropower plant has to meet in the scheduling. This demand can be demand needed inside the node, it can also be the energy being transferred out of the system border, or it can be a combination of this. Where the demand in the scheduling ends up does not impact how the scheduling is being made in this case. For this system, the import of energy is not possible, this is a conscious choice to lower the overall complexity of the system and will have minimal impact on the results in this master's thesis. If the load is not met, the model has to compensate with loadsheading as seen in equation (90) and (101), which is penalized in the objective value equation (84) and (95).

4.2.1 Reservoir

The system has physical limitations that have to be taken into consideration when making the mathematical model, one of these limitations is the reservoir. The reservoir will usually have an upper and lower water level limit. The upper limit will be set by the size of the dam, it is usually set a bit under the height of the dam to avoid spillage. From equation (85) and (96) combined with equation (91) and (102) it can be seen that the maximum reservoir level is set to the absolute limit of the dam since the model obtains spillage when r_g^{Max} is reached. This is a simplification of a real-life system, but this simplification significantly reduces the complexity of the problem, while not impacting the results of this thesis. There are multiple factors that can set the lower limit of the system. How much water the wildlife around and in the reservoir need is one of them, this is usually set by governments when the hydropower plant gets permission to be built. Another is what the geology looks like, if the reservoir has a lot of finer sediment on the floor of the

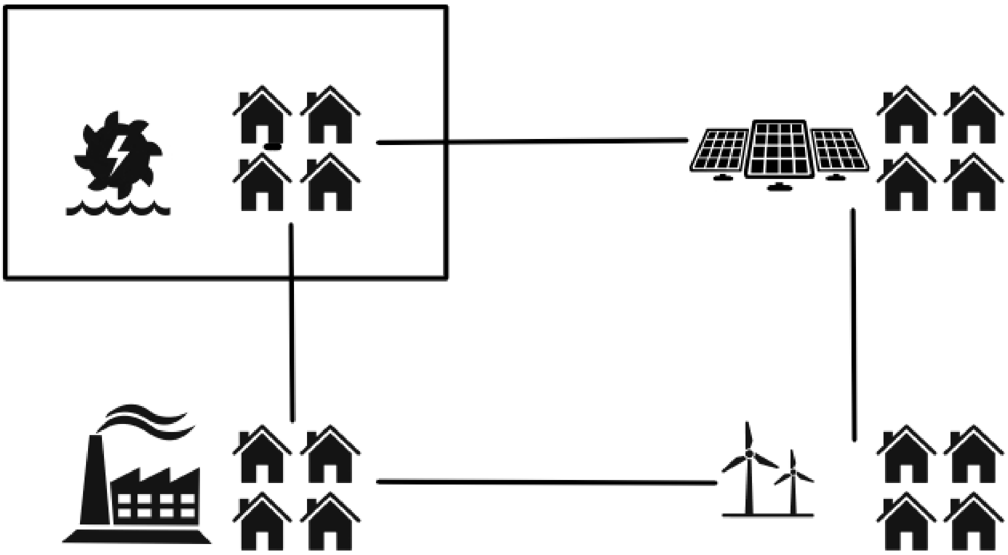

Figure 4: system with system borders

reservoir, the intake gate will be higher to avoid getting too much of the sediment into the plant. From equation (86) and (97) it can be seen that the lower limit of the reservoir is aggregated in this case, this as recommended by the industry partner to get a more realistic scheduling for the user case. The belief is that whether to aggregate this or not will not impact the results of this research. The lower and upper limit of the reservoir can be seen illustrated in figure 5

4.2.2 Water flow

The amount of water going into the hydropower plant is limited by the cross-section of the water and discharge tunnel, which is usually built to transfer the same amount of water, this can be seen in equation (89) and (100). In equation (92) and (103), it can be seen that unregulated inflow is taken in, this is the water coming in through the swing shaft. This water cannot be regulated, so the model has to schedule for bypassing water to stabilize the amount of water needed for production. The reservoir is filled up with the regulated inflow, and the discharged water from the generators is higher up in the system. The reservoir is lowered by spillage and releases water into the water tunnels, and this can be seen in equation (91) and(102). The objective function is a minimizing function, and this tells the model to minimize the spillage, bypassed water, and load shed. The cost of this differs from what the plant's owner feels is most disturbing for the system, but the load shed is usually the highest. In the first stage, there is also an α which represents the cost found in the second stage. This allows the first stage to take the second stage into consideration and plan the current week accordingly.

4.2.3 Production

From equation (87) and (98) it can be seen that the model has an upper limit for production, this represents the physical limitations of the turbine and generator. This takes into account how much power can be transferred out of the generator, and how much water the

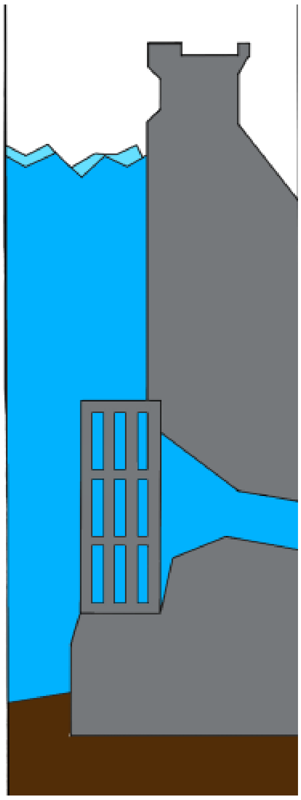

Figure 5: crossection of an dam

rotator can take for a given moment. This upper limit also has a safety factor to minimize the damage to the system. The system has a minimum production to let the turbine spin and generate power. However, this is usually not the lower limit of the production. The lower limit of the production is often set from an economical point of view. The lower limit for production is often set to where the efficiency of the system is tolerable. The lower the efficiency of the system is the more water has to be spent producing the same amount of energy, therefore the lower limit is often set where the owners of the system feel comfortable with it from an economical point of view. The lower limit of the model can be seen in equation (88) and (99).

4.3 Scenario fan simulator

Scenario fan simulator(SFS) was introduced in section 3.1, it is a combination of optimization and simulation. The sections above have explained the optimization part of this model, and the simulation part takes care of how the model stores results and when the optimization is conducted. The inflow for the current week w is a known constant and the optimization model is then used to find the optimal solution, and the optimal solution is then stored and used to calculate the constant known inflow for next week w+1. This is repeated until the SFS model has gone through all weeks in the system. This is illustrated in 6 for a model with 52 weeks and n scenarios for the two first weeks

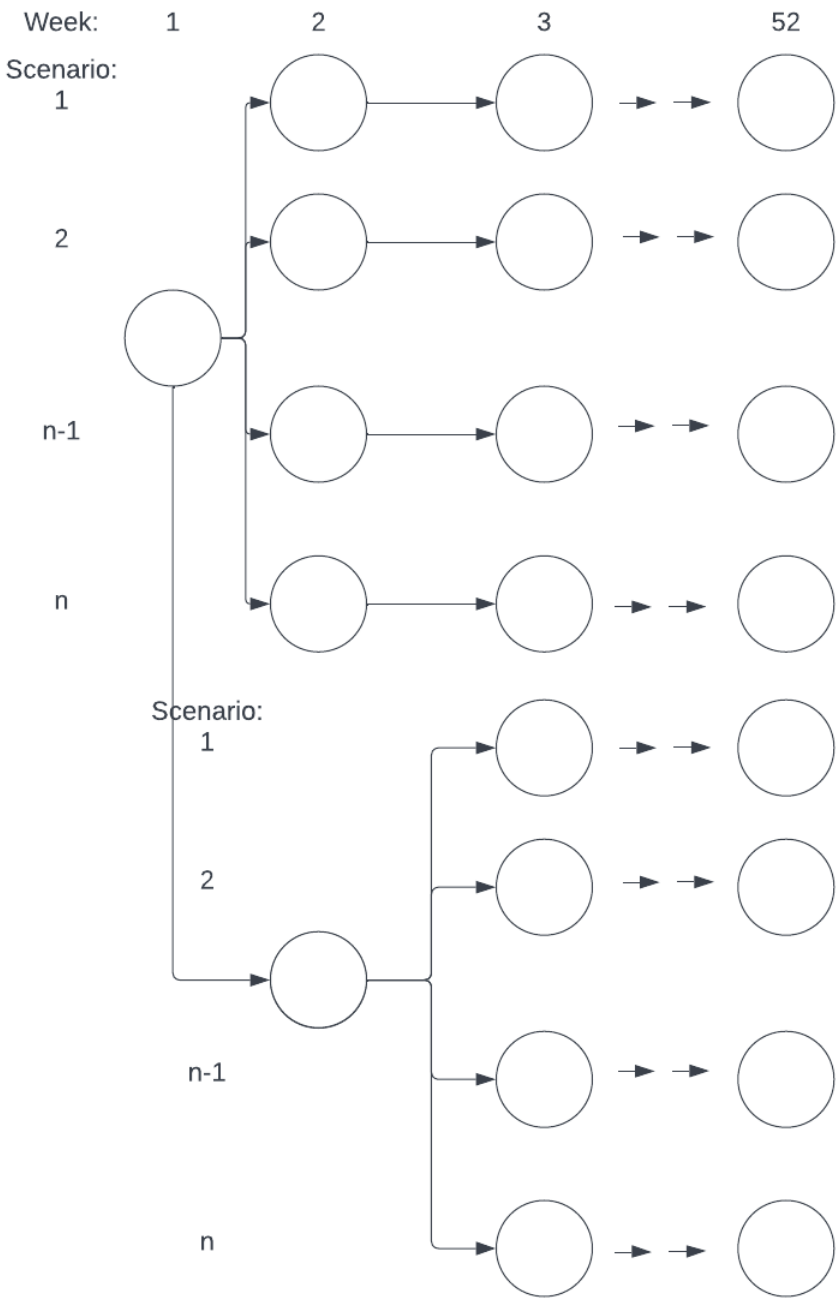

Figure 6: This figure shows the logic of the SFS model for a model with 52 weeks total, n scenarios total for the first two weeks.

4.4 Conclusion Method

The mathematical model has been presented in this chapter, and the variables and parameters connections have been found in the system. Further, the SFS model has been presented and explained how it was implemented. In the next chapter, a case study using this model will be presented, data-gathering will be discussed, and the technique used to create the machine-learning model will be discussed.

5 Case study

To train and test a machine learning model is it essential to obtain data. This data gathering requires a case study to have a basis for the data. The case study for this master's thesis is described in section 5.1. Further, the process to obtain a bigger dataset than what industry partners can provide is given in section 5.2. The machine learning models are used in this thesis that are discussed in section 5.3. Here, the evaluation method used in this thesis is also described. This chapter ends with a summary in section 5.4

5.1 Description

To better test the performance of the machine learnings model against the optimization model, a case study was created. The case study is a smaller version of another case study from a PHD paper. In the original case study, 12 hydro power reservoirs, transformer lines, wind production, and thermal power production were included. To test the performance of the machine learning model, a smaller case focusing on three cascading hydropower reservors was enough. From 7 it can be seen that reservoirs eight, nine, and ten, were renamed zero, one, and two for this thesis.

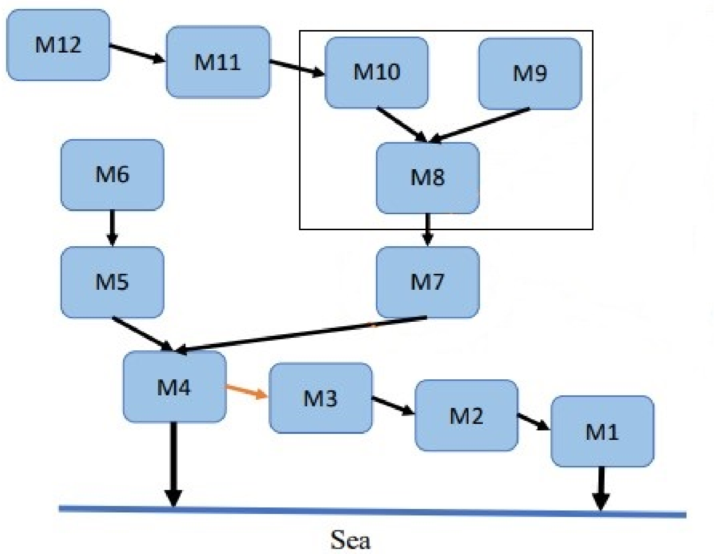

Figure 7: Original hydropower case

The system has three cascading reservoirs with one generator connected to each reservoir. The discharged water from the upper generators, generators one and two, flows into the lower reservoir zero. From figure 8 the details for the flow of the system can be seen. The reservoir is filled up with discharged water and regulated inflow water. If the reservoir is full, any inflowing water will be spillage. The reservoir is then lowered by releasing water into the water tunnel, unregulated inflow enters the water tunnel through the swinging shaft and the excess water is tunneled away as bypassed water. The released water then enters the generator and is discharged to the lower reservoir or the river.

The mathematical model presented in chapter 4 is a general model, in this chapter can

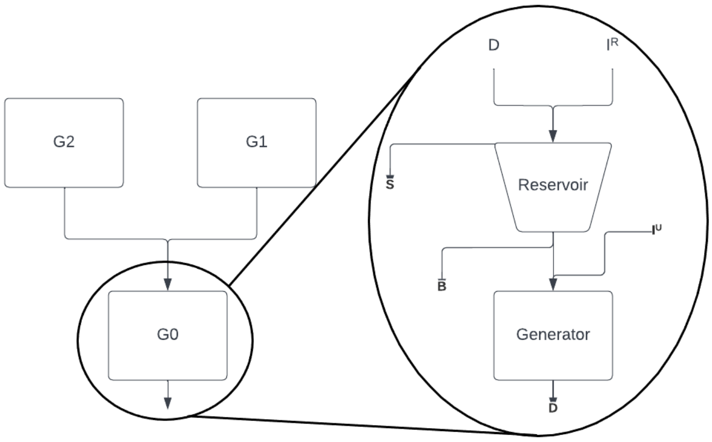

Figure 8: diagram of the hydro power plant

the formulation be specified do to the information from the case study. From table 1 the cost parameters used in the objective function can be found. Here, it can be seen that the cost of load shedding is higher than spillage and bypass, this is to encourage the model to always meet the load demand. The time horizon for the case study is one year with weekly time steps. The parameter cost for maximum discharge, reservoir level, and production is given in table 2. The case does not have any minimum discharge or production, but it has a minimum reservoir level of 10% of maximum capacity. The initial reservoir level at week zero is set to 65 % of the maximum reservoir level.

The data set for this case study was provided by an industrial partner, for a bigger case study as mentioned above. Therefore it was necessary to scale the data somewhat to be used in this data set. The load data was scaled to have a maximum of 140 MWh, the scaling was conducted by dividing all the demand with maximum demand and multiplying it by the new maximum load. With this approach, the trend of the load will be kept, while the new dataset would better fit this case study. The inflow data was also modified, this was increased by a factor of 15. All the data points were multiplied by 15, this was conducted with both the unregulated and regulated inflow.

	Cost
Load shedding	4500
Spillage	80
bypass	20

Table 1: Table containing the cost penalties used in the model

	The capacity of the reservoir (V_{Max})-Mm^3	Maximum production of hydro-plant (Q_{Max}) -VM	Maximum discharge (P_{Max}) $- Mm^3$
Generator 0	15	50.0	125.3
Generator 1	30	12	24
Generator 2	47.3	30.3	28.5

Table 2: The table contains data for the reservoir size and maximum value for production and discharge

5.2 Synthetic data

Neural network is a good machine learning thickening, but it has one big downside, the amount of data needed to train the models. This is often higher than the amount of data that is possible to obtain, it is therefore not unusual to generate more data through data synthesis. This data is often called synthetic or artificial data. The original dataset was received from an industrial partner. This data set included typology, inflow, and demand data for 50 years. The typology data contained which reservoirs were connected, the efficiency rates, and the different limitations. The inflow data set included regulated and unregulated inflow for every reservoir each week for 50 years. The demand data set included load data for the area. This data is too small to thoroughly train and test the ML, so synthetic data is needed to obtain valid results. The creation of the synthetic dataset was based on the original dataset. To generate the synthetic data for this master's thesis, TimeGan is used to generate inflow data. Using TimeGan, the data set increased from 50 years of data to 2557 years of data both for regulated and unregulated inflow. The original setting for TimeGan was used to create the data.

5.3 Machine learning models

After going through the literature review, a Long short-term memory (LSTM) neural network was chosen as the best fit to be tested for implementation into the LTHPS. This ML technique was, as far as the author of this thesis knows, not tested in hydropower scheduling. There have been multiple other neural network models used like artificial neural networks and deep neural networks which have got promising results. This can indicate that LSTM NN might give good results for this case study.

Six networks were made to ensure as good results as possible, one for each reservoir predicting water value and one for each reservoir for predicting reservoir level.

The trends for the different reservoir levels were so different for each reservoir that multiple networks were expected to get the best results. The data file was split into training and testing data, with 1790 years corresponding to 70% of the data going to training and 767 corresponding to 30 % years going to testing. The input data for both the NN predicting water value and reservoir level was inflow data. The key element of LSTM NN is that it has memory. For the networks in this thesis, the last five predictions were stored in memory.

To find the optimal hyperparameters for the LSTM NN, Bayesian optimization is used. Baysion optimization had 10 iterations to work through. The bounds of the hyperparameters can be seen in table 3. The bounds of the hyperparameters are found using trial and error, the optimal solution was within these bounds throughout all the testing.

	Number of neurons	Number of layers	Learning rate
Lower bound	8	1	1e-4
Upper bound	64	3	1e-1

Table 3: Tabel containing the bounds of the hyperparameters for the long short-term memory natural network

The performance of the machine learning model is evaluated using Root mean square error calculation. This evaluation does not always give a clear picture of the performance, so two other machine learning models using linear regression were created to work as a baseline to compare the performance of the LSTM NN model. The two linear regression machine learning model is ridge regression from the sklearn library and XGBRegression from the xgboost library. The linear regression and neural network predict the model using different technique, so embedding was also included to see if it could be utilized. The embedding was not optimized in any way, so the models were included with equal fractions.

5.4 Conclusion of Case study

In this chapter, the case study has been presented. Further, the method to generate synthetic data to increase the data set has been discussed. The chapter ended with a discussion on which machine learning models going to be used in this thesis and how it is are going to be evaluated. In the next chapter, the results will be presented and discussed.

6 Result and discusion

6.1 Hydro power production

In tabel 4 can the evaluation results for predicting the production of the different reservoirs be seen. This shows the mean squar error (MSE), the mean average error (MAE), the root mean squear error (RMSE) and the mean average prosentage error (MAPE) for the four different machine learning technique, ridge, XBGRegressor, LSTM and Ensemble for each reservoir. MAPE is presented as an prosentage, while the rest is decimal numbers cut at six decimals.

		MSE	MAE	RMSE	%
Production Reservoir 8	Ridge	30.953785	3.835969	5.563613	95.0501 %
	XBGRegressor	36.756865	4.125552	6.062744	94.7886 %
	LSTM	30.298562	3.809250	5.504413	94.7394 %
	Ensemble	31.093137	3.857451	5.576122	94.9646 %
Production Reservoir 9	Ridge	8.177210	1.194694	2.859582	68.3593 %
	XBGRegressor	9.546925	1.281636	3.089810	74.9374 %
	LSTM	7.926746	1.171228	2.815448	67.4889 %
	Ensemble	8.165172	1.192929	2.857477	69.1512 %
Production Reservoir 10	Ridge	17.611084	3.065050	4.196556	74.9060 %
	XBGRegressor	20.912075	3.275638	4.572972	74.6512 %
	LSTM	16.979026	3.021703	4.120561	75.0366 %
	Ensemble	17.625071	3.068865	4.198222	74.8468 %

Table 4: The evaluation results for production prediction

Figure 9 - 11 is presented as grids of figures corresponding to the reservoir zero, one and two. Each grid containing a figure showing one ML models predicted values against the testing values. The same testing values is showed for each of the ML models. Figure 9 has the lowest value at around 40, and the highest value with around 90,figure 10 has the highest value with around 25 and the lowest around 5 and figure 11 has the highest value at around 14 and the lowest around 0.

Table 4 shows that reservoir zero outperformed both reservoirs one and two when using the MAPE evaluation. However, reservoir zero is still performing inefficiently when juxtaposed with the other reservoirs when utilizing the other evaluation techniques. This is especially evident when using the MSE evaluation. A possible reason for this discrepancy comes from the large variation of data points. Looking at Figure 9, the maximum value reaches 90, which is exponentially higher when compared to Figures 10 and 11, where the maximum value is shown to be around 25 and 14. The RMSE and MAE do to some degree take this variation into consideration, but will not be as accurate when opposed to the MAPE evaluation, which normalizes the values before evaluating them. This variation will therefore not affect the MAPE evaluation, making the MAPE evaluation a preferable evaluation technique used in this thesis.

When taking the normalized MAPE values into account, it can be seen that reservoir one and two both produced significantly inferior results than reservoir zero did. Causation can be a result of the shape of the data itself. This statement is further supported when looking through Figures 10 and 11, where the data is shown to have a more discrete form as opposed to the data shown in Figure 9. The fast-changing discrete data could be harder

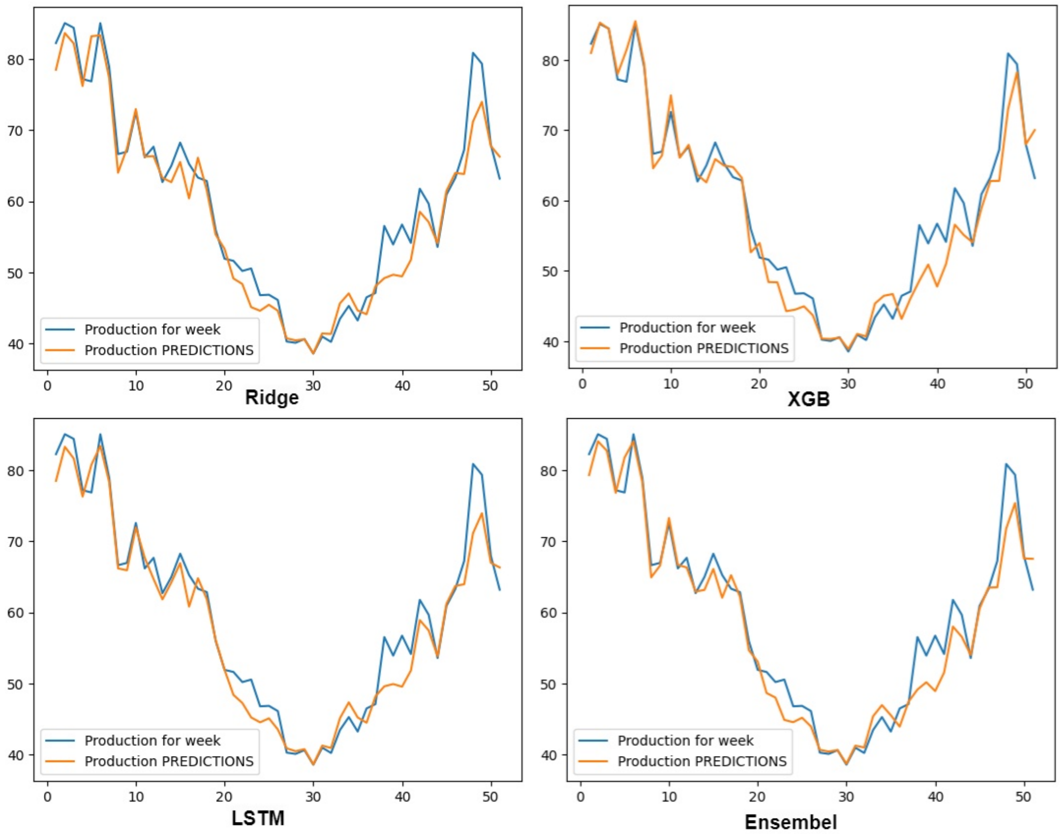

Figure 9: This figure shows a grid of the weekly production prediction for reservoir 0 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left showed the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method.

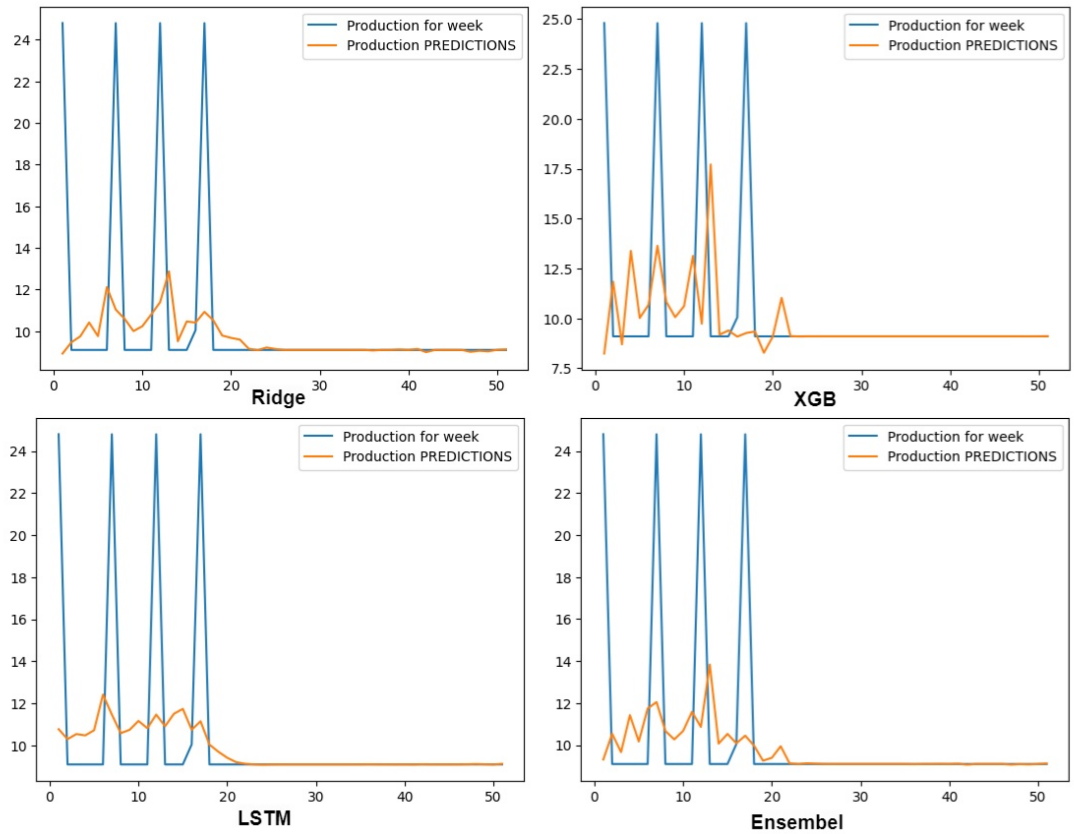

Figure 10: This figure shows a grid of the weekly production prediction for reservoir 1 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left showed the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method.

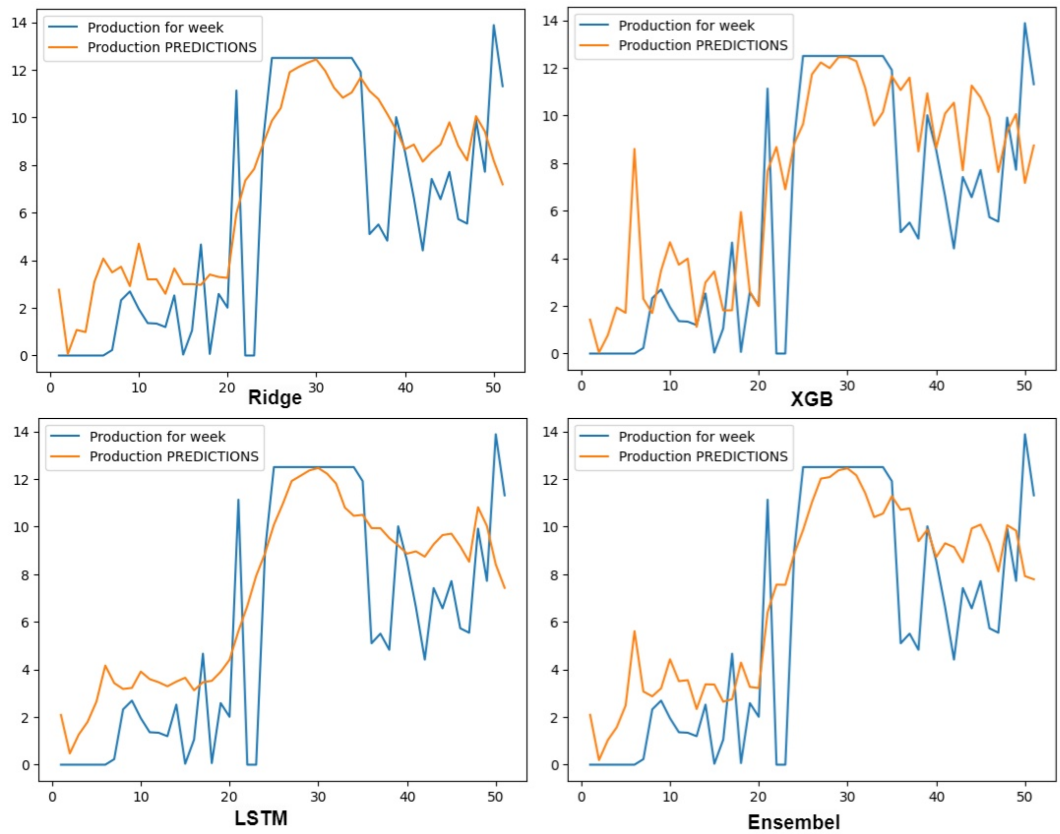

Figure 11: This figure shows a grid of the weekly production prediction for reservoir 2 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left showed the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method.

to gather from the inflow data. The potential source of this data shape originates from the downscale from the original dataset, illustrated in Figure 8. This downscale can remove some of the essential input data needed for the machine learning models to perform well, potentially producing less precise results.

Looking back at Table 4, the data shows that the LSTM model slightly outperformed the other machine learning models on reservoir two, but was again slightly outperformed on reservoirs zero and one. An explanation for this can be that to successfully trains a Neural network, a huge amount of data is needed. The ridge and XBG both have an easier task of finding patterns with less data than the LSTM does. However, the LSTM model might do better to find lees occurring trends in the dataset. However, the LSTM model has a higher chance to find less occurring trends in the dataset which is good because this will stop the model from predicting infeasible solutions, and therefore be closer the testing value.

6.2 Water value

In tabel 5 can the evaluation results for predicting the production of the different reservoirs be seen. This shows the mean squar error (MSE), the mean average error (MAE), the root mean squear error (RMSE) and the mean average prosentage error (MAPE) for the four different machine learning technique, ridge, XBGRegressor, LSTM and Ensemble for each reservoir. MAPE is presented as an prosentage, while the rest is decimal numbers cut at six decimals. Figure 12 - 14 is presented as grids of figures corresponding to the reservoir

		MSE	MAE	RMSE	MAPE
Water value Reservoir 8	Ridge	15.675099	1.610445	3.959179	91.2398 %
	XBGRegressor	18.746445	1.738796	4.329716	88.3843 %
	LSTM	15.344949	1.603208	3.917263	91.9840 %
	Ensemble	15.756544	1.623405	3.969451	91.7508 %
Water value Reservoir 9	Ridge	55.325032	2.926293	7.438080	91.8299 %
	XBGRegressor	66.112236	3.200485	8.130943	88.6395 %
	LSTM	175.800083	12.972585	13.258962	92.4058 %
	Ensemble	69.176923	6.245184	8.317267	92.4326 %
Water value Reservoir 10	Ridge	125.011053	6.506435	11.180834	83.6771 %
	XBGRegressor	144.243526	6.575063	12.010143	82.6121 %
	LSTM	122.341946	6.557204	11.060829	83.6163 %
	Ensemble	124.463075	6.495852	11.156302	83.7656 %

Table 5: The evaluation results for water value prediction

zero, one and two. Each grid containing a figure showing one ML models predicted values against the testing values. The same testing values are shown for each of the ML models. Figure 12 has the lowest value at around 0, and the highest value with around 20,figure 13 has the highest value with around 30 and the lowest around 0 and figure 14 has the highest value at around 40 and the lowest around 0.

The contents of Table 5 show overall good results from the prediction of the water value in all of the reservoirs. When looking at the normalized MAPE, the best result was over 92%, with the lowest result reaching over 82%. In contrast, the lowest result from Table 4 was 67.4889%. The good results are likely affected by the shape of the data. The data itself had only a small number of different values to predict, therefore making it easier to recognize the pattern. We see this corroborated in Figure 12, where the water value is around 20 for most of the year, this is an abnormal outcome where there machine learning models find the pattern quite easy. Continuing with Table 5, one can see that the LSTM model do outperform the other models when using the MAPE evaluation in reservoir zero, however it is still being outperformed by the other two reservoirs when using the ensemble method. This method might be a better fit when the data is shaped with the lower variance, as seen in figure 12. Table 5 also shows that the MSE values are significantly higher for the LSTM model than they were for the other reservoir one models, this might come from the difference in how the LSTM model do prediction and the difference in how MSE and MAPE evaluate the data.

Figure 13, shows the reader that the LSTM model uses 3-4 weeks longer than the other models, to reach testing values after week 20, but follows the swinging pattern closer than the other models do until then, this might be the reason that MSE and MAPE give vastly different evaluation for the LSTM on one. None of the predicted values for all models can follow the fast-changing values of the water value in Figures 12 to 14. Here, the Water

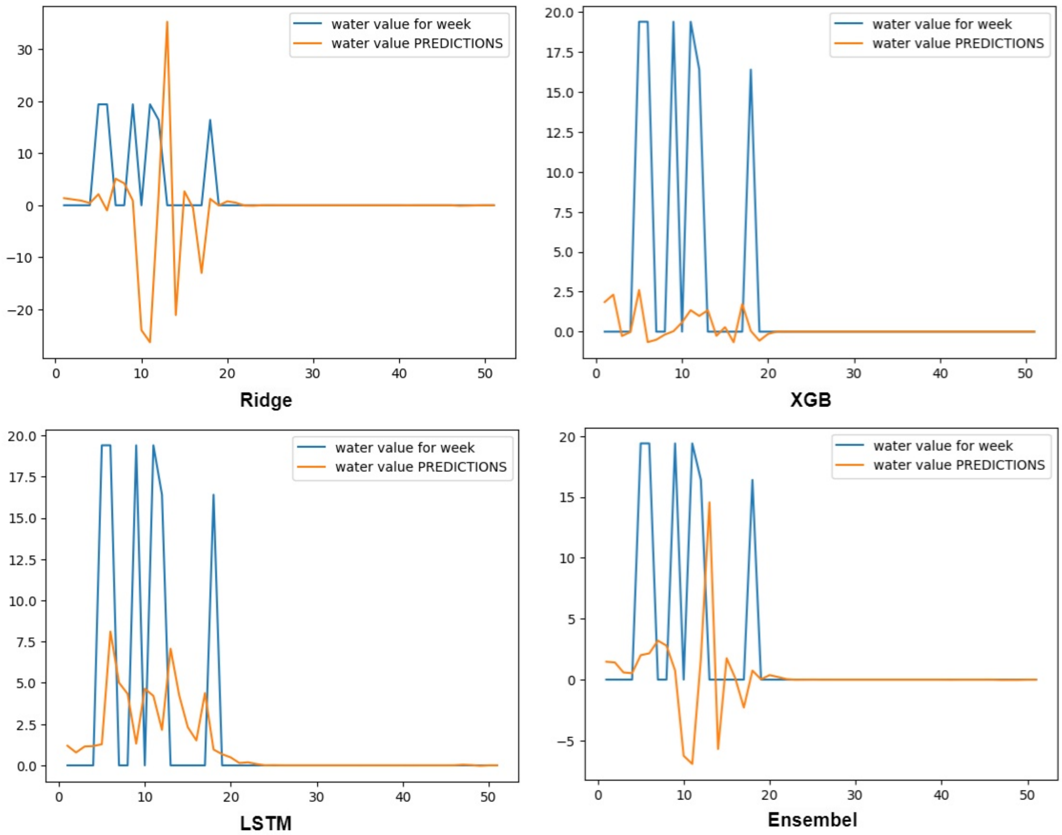

Figure 12: This figure shows a grid of the weekly water value prediction for reservoir 0 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left showed the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method.

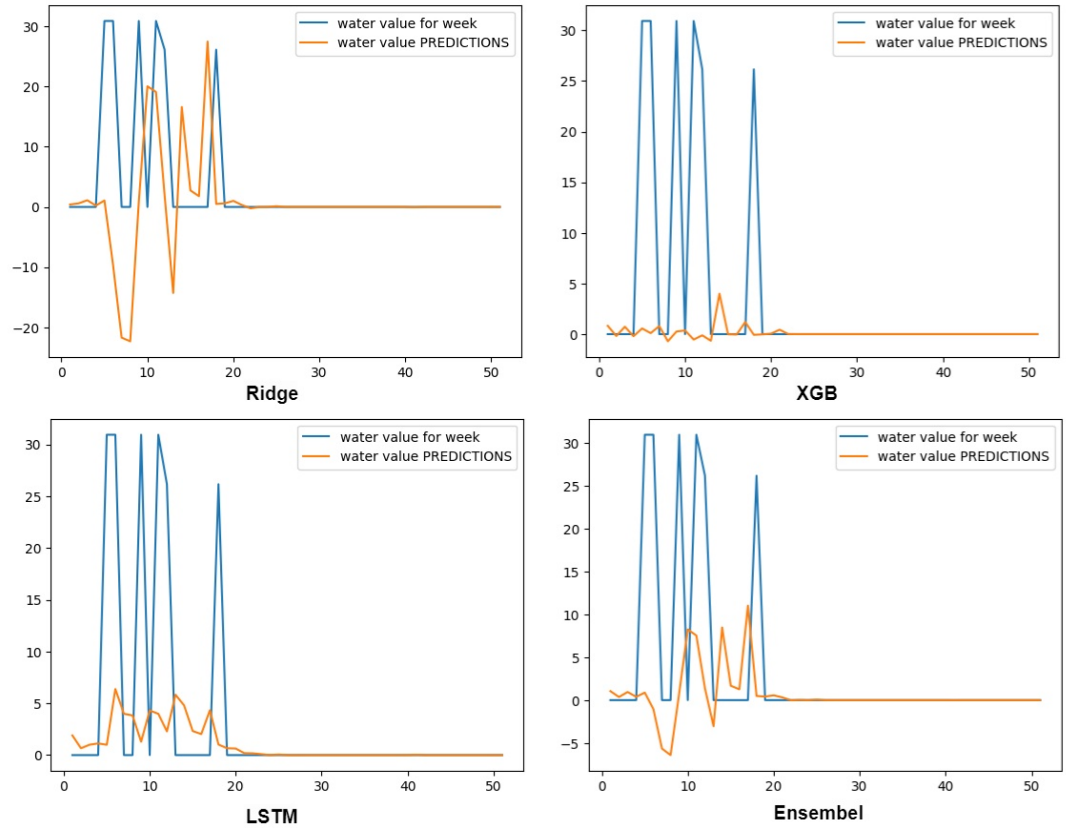

Figure 13: This figure shows a grid of the weekly water value prediction for reservoir 1 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left showed the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method.

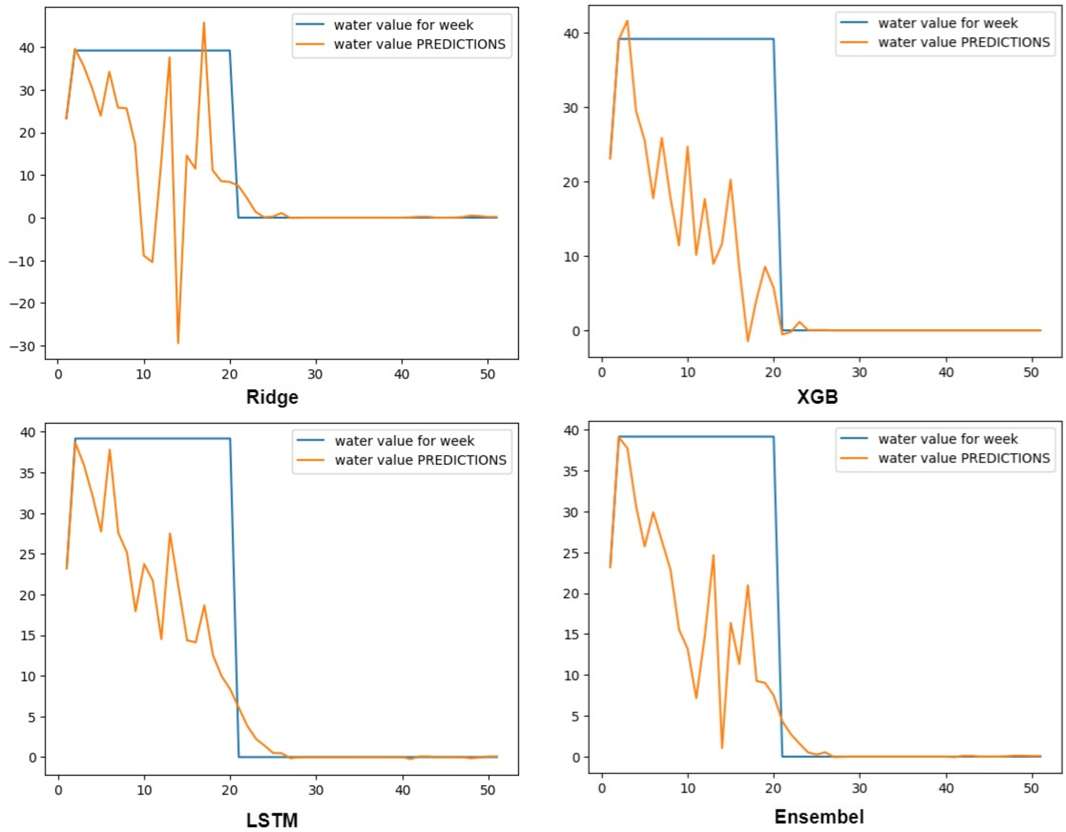

Figure 14: This figure shows a grid of the weekly water value prediction for reservoir 2 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left showed the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method.

value can go from 0 one week to 40 the following week. This might come as a result of the case study not including all the data from every reservoir, and therefore the inflow might not represent the complexity of upstream discharge filling the reservoirs. The LSTM model is the only model that manages to not predict an infeasible solutions ero is sat as the lowest the water value could reach. With this reasoning in mind, it could explain why the LSTM model performed this well in the evaluation.

		MSE	MAE	RMSE	MAPE
Reservoir level Reservoir 0	Ridge	0.057436	0.033883	0.239658	99.1032 %
	XBGRegressor	0.129995	0.022040	0.360549	99.7977 %
	LSTM	0.054181	0.018079	0.232769	99.8795 %
	Ensemble	0.063442	0.021825	0.251876	99.6926 %
Reservoir level Reservoir 1	Ridge	0.097510	0.049436	0.312266	99.4907 %
	XBGRegressor	0.164851	0.047933	0.406019	99.6911 %
	LSTM	0.092811	0.036149	0.304649	99.8796 %
	Ensemble	0.102188	0.040803	0.319668	99.8008 %
Reservoir level Reservoir 2	Ridge	27.602556	2.308737	5.253813	94.9581 %
	XBGRegressor	32.816938	2.501367	5.728607	93.9402 %
	LSTM	27.321315	2.191803	5.226980	95.3620 %
	Ensemble	27.786565	2.281259	5.271296	95.1436 %

Table 6: The evaluation results for reservoir level prediction

6.3 Reservoir level

In tabel 6 can the evaluation results for predicting the production of the different reservoirs be seen. This shows the mean squar error (MSE), the mean average error (MAE), the root mean squear error (RMSE) and the mean average prosentage error (MAPE) for the four different machine learning technique, ridge, XBGRegressor, LSTM and Ensemble for each reservoir. MAPE is presented as an prosentage, while the rest is decimal numbers cut at six decimals.

Figure 15 - 17 is presented as grids of figures corresponding to the reservoir zero, one and two. Each grid contains a figure showing one ML models predicted values against the testing values. The same testing values is showed for each of the ML models. Figure 15 has values around 15, figure 16 has the highest value with around 30 and the lowest around 23 and figure 17 has the highest value at around 47,5 and the lowest around 30.

The prediction given of the reservoir level was fairly accurate with a normalized MAPE between 93.9402 % to 99.8796 %, as can be seen in table 6. This is most likely a result of the data shape, as the data here has a very low variation. The low variation is illustrated in Figures 15 to 17. From the MAPE evaluation, In the same table, the LSTM model performed slightly better than the other models in all reservoirs.

Reasons for this might be, as mentioned previously, the water value, and the fact that the LSTM model does not predict infeasible values.

It has seen an overall trend that it is a hard stop for values at the reservoir's upper limit.

This trend is consistent with the results of this case study 1 and will help with the LSTM model, giving more precise answers, but the model will have to be retrained if some conditions change, for an instant, the reservoir level being increased or decreased.

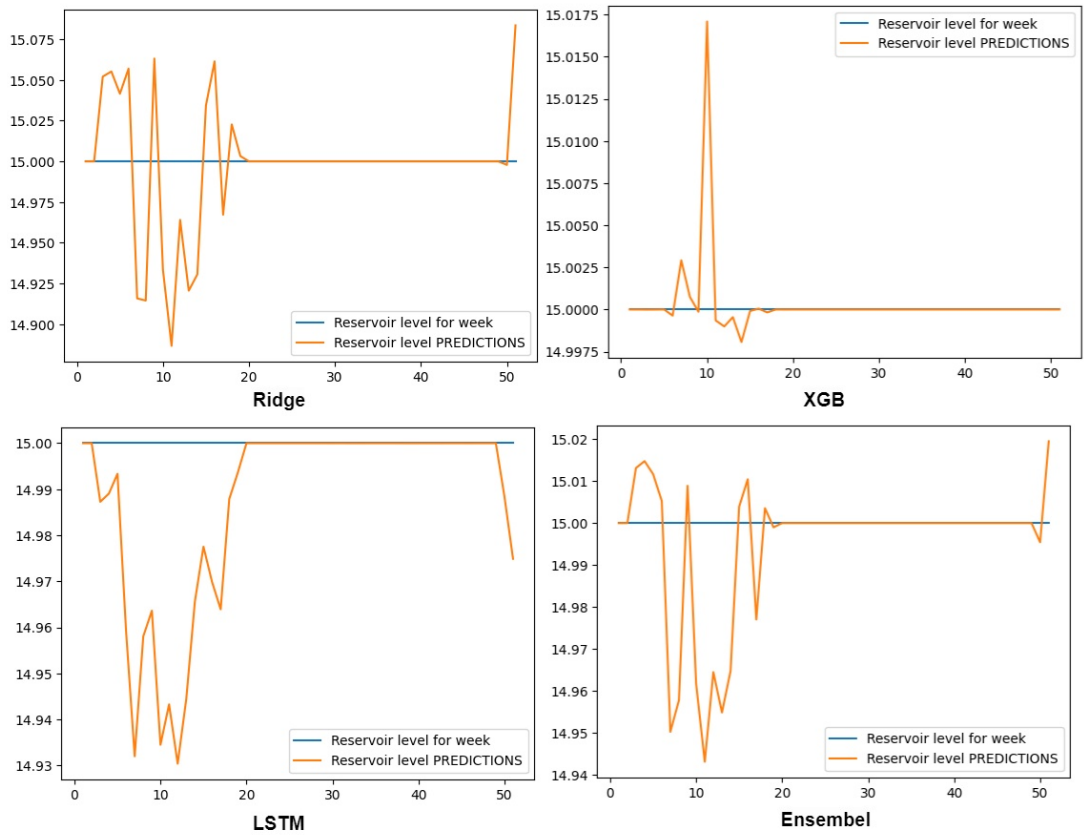

Figure 15: This figure shows a grid of the weekly reservoir level prediction for reservoir 0 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left shows the prediction conducted with LSTM and the lower right figure shows the prediction conducted with the ensemble method.

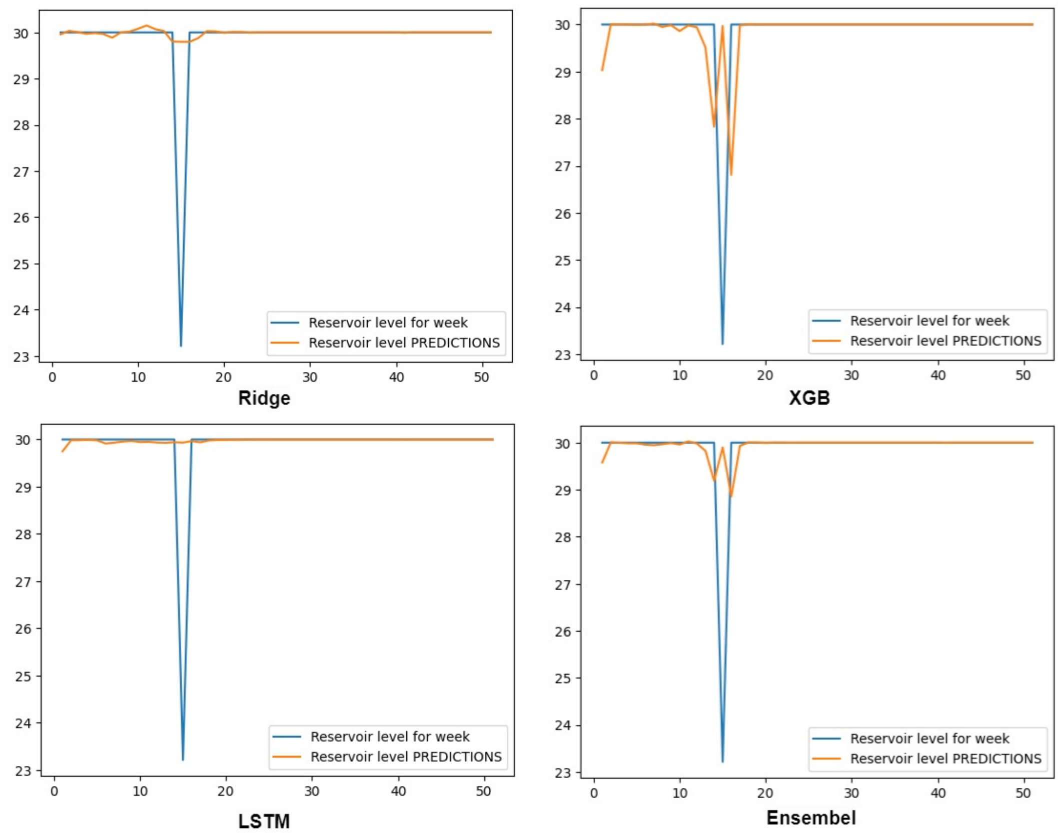

Figure 16: This figure shows a grid of the weekly reservoir level prediction for reservoir 1 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left shows the prediction conducted with LSTM, and the lower right figure shows the prediction conducted with the ensemble method.

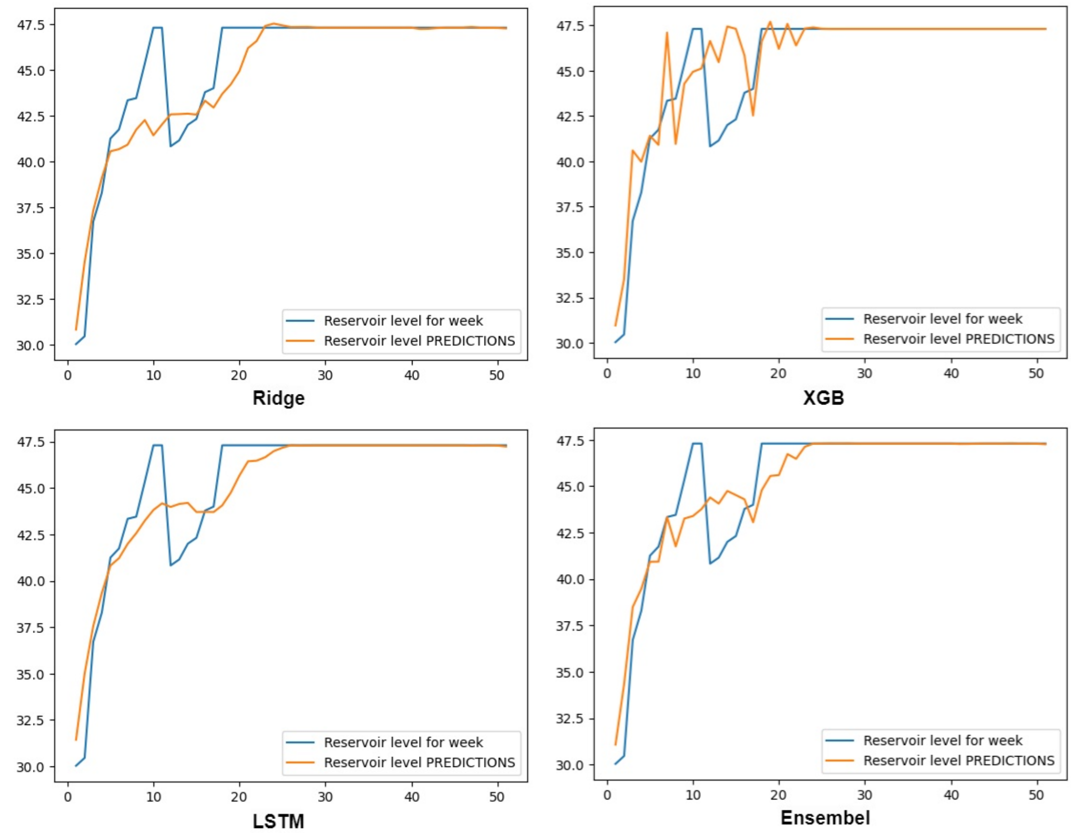

Figure 17: This figure shows a grid of the weekly reservoir level prediction for reservoir 2 for one predicted year. The upper left figure shows the prediction conducted with Ridge regression, the upper right shows the prediction conducted with XGBRegression, the lower left shows the prediction conducted with LSTM, and the lower right figure shows the prediction conducted with the ensemble method.

6.4 Computational time

The computational time is dependent of multiple factors with cuputaisinal power on the hardware, if any other programs are running in the same hardware, heat in the room and on the hardware as a few examples. Therfor is computational time presented here an average of the last ten times the programs have been running.

The SFS model gather all the result in one go and use on average 11 652 sekonds or around 3 hours 14 minutes to gather all the result with the original 50 scenarios. With more scenarios will this computational time increase, but it might give a more detailed results on the data. The machine learning models have been split into 9 separated codes that might run in parallel or it can be run in a series. When the codes are run in a series the average computational time is 34 seconds to gather the water level, reservoir level and production production. When the programs are runn in parallel did it take a total of 15 seconds to gather all the results. The optimization model generate the results in series, therefore will the series computational time be used when comparing. The machine learning model used a total 11618 second less then the optimization model, which is 99.7% time reduction.

6.5 General source of the error

When conducting research, there is always a chance that error sources can impact the results. Some of the potential errors concerning this research project have already been discussed in the subsections above. The following subchapter will explore more potential errors that could have impacted the research. The case study used for this master's thesis contains a smaller scope than the original case study but it still uses the same data set. To take in this scope difference, we use an inflow factor and a maximum demand that has been established to make the dataset fit better with this case study. This inflow factor and demand is manually picked and might therefore give some extreme cases. In Figure 13, the reservoir level is the maximum reservoir level throughout all the weeks. This trend continues for most of the testing years for all the reservoirs. This indicates that the inflow factor was too big, so instead of planning when to store and release water, the scheduling would instead always be at maximum capacity. This caused it to spill a lot of water. This makes the case less realistic and the results from the reservoir level and water level less reliable as definitive results for the performance of the machine learning models. This is not believed to impact the result of how the LSTM model could predict the result since it was on a theoretical basis, but after seeing the results is it believed to have given a data shape that had a too simple pattern to predict, and therefore might have produced unrealistically good results.

The dataset used for this master's thesis included 50 years of data, to train a machine learning model, especially for a Neural Network model a significant amount of data is needed. In order to provide this, synthetic data was created. The synthetic data was created by an machine learning model and is being treated as an black box in this master thesis. It can therefore have made artificial data that didn't fit this case study properly and possibly impacted the results.

Since programming for this master thesis was done manually, human errors like misspelling can take place and therefore temper with the results. To minimize this error's ability to affect the research, the code has been split up and tested separately, whenever this was possible. The code used in this master thesis contains some premade userpacks imported and used. These userpacks might contain faults and not do as intended by the author of this master thesis and are therefore a source of potential error in the research. To lower the chance for these errors to impact the research, the imported userpacks have been looked into before being used to ensure that they do as intended.

7 Conclution

This master's thesis presented an overview of hydropower scheduling and machine learning through an extensive theory chapter, a literature review, an in-depth model presentation, and a case study containing a hydropower plant with three cascading reservoirs. The literature review gilded the state-of-the-art long-term hydropower scheduling technic and a possible machine learning technic to test the optimization model against.

The case study used real-life data provided for a hydropower reservoir with 13 cascading reservoirs. This dataset was originally provided for a bigger case study executed by a Ph.D. Candidate, so it where modified with an inflow factor and a demand cap to fit the smaller case study. The original data contained 50 years of data, but to have proper training in the machine learning model were 2507 synthetic years created using TimeGan to a total of 2557 years with data. The case study was conducted using four machine learning technics: Ridge, XGB, LSTM, and Ensemble, where the two first was set as base case models, while the last two would be looked at for performance.

The main results from the case study show that the machine learning models had a computational time reduction of 99.7 %. The accuracy showed that base case models were beaten in 7/9 of both the LSTM and ensemble models, while beaten with at least one of them in 8/9 evaluations when looking at the MAPE evaluation technic. Prediction of the production for reservoir zero where the one evaluation where neither the LSTM or the ensemble model could outperform the base case models. The prediction for the power production where also the one place where the machine learning models scored the lowest. The LSTM model scored 94.7 %, 67.5%, and 75% with the MAPE evaluation. The power production prediction is the one evaluation that will impact the conclusion most since this seems to have the most realistic results. The ensemble model had some times it could compete with the LSTM model, however, the LSTM model manage to outperform it most of the time, so from these results might the LSTM model be better given this case study.

The LSTM model had a fantastic computational time reduction and performed best in overall accuracy, however, it was outperformed once by the ridge model, and it had accuracy scores as low as 67.5%. From the case study conducted in this thesis can it be concluded that the LSTM machine-learning model can be a great supplement, but it is not ready to substitute the optimization model used today.

8 Future work

This chapter will discuss areas of this master's thesis that can be expanded for further work. Some of the areas are already mention under the result and discussion chapter of this thesis

The dataset in this master thesis is not originally planed for the case study used. Therefore was an inflow factor and a maximum demand put in to ensure that the data will be provided could be used. In further work, it would be beneficial to research how machine learning can be used in hydropower scheduling to include a case study that takes the whole dataset unchanged to get more realistic results.

Hydropower is a flexible and reliable power source, but in many areas is there an energy mix more unreliable and variable than simulated in this master's thesis. Therefore might further work a bigger case study containing multiple nodes connected through transmission lines. Each node could contain a renewable power source and some load to better simulate the energy mix in many areas. With this new case, can it be interesting to see if hydropower scheduling using machine learning can stabilize the power grid.

As an extension of this case, pumping power can also be included as an extra stabilator to the grid. Pumping power can utilize the power surplus from the power grid to fill the reservoirs, and then provide that water when the renewables don't provide as much power.

Machine learning models use much less time than the regular hydropower scheduling program, as found out in this research, therefore might a test where the long time hydropower schedule has more details be beneficial, this can for an instant, be a daily timestep for one year and might make the seasonal schedule redundant if the research is successful.

Machine learning is timesaving compared to the optimization model, therefore, might other machine learning models and technics be beneficial to have further research on. The literature review conducted in this thesis has Neural Network potential to be used as a replacement for the optimization model used today and might be beneficial to look into in further research. The results from this master's thesis also find an ensemble model to be beneficial to look into, especially with a Neural Network as a part of the model.

References

[1] S. Vaca-Jiménez, P. Gerbens-Leenes, and S. Nonhebel, "The monthly dynamics of blue water footprints and electricity generation of four types of hydropower plants in ecuador," *Science of the Total Environment*, vol. 713, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0048969720300899?via%3Dihub

[2] "Hydroelectric power calculator," (read - 20.05.2023). [Online]. Available: https://www.omnicalculator.com/ecology/hydroelectric-power

[3] Ånund Killingtveit, *Managing Global Warming An Interface of Technology and Human Issues.* Chapter 8-Hydropower: Academic press, 2019.

[4] B. A. Nasir, "Design considerations of micro-hydro-electric power plant," *Energy Procedia*, vol. 50, pp. 19–29, 2014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1876610214007395

[5] K. Kumar, "A review on operation and maintenance of hydropower plants," *Sustainable Energy Technologies and Assessments*, vol. 49, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S2213138821007189

[6] C. Bordin, H. I. Skjelbred, J. Kong, and Z. Yang, "Machine learning for hydropower scheduling: State of the art and future research directions," *Procedia Computer Science*, vol. 176, pp. 1659–1668, 2020.

[7] "Important information about electricity load shedding (controlled outages)," (read - 15.06.2023). [Online]. Available: https://www.gpltexas.org/outage-center/electricity-load-shed

[8] N. Barney and T. W. C. Nick Barney, "load shedding," (read - 10.06.2023). [Online]. Available: https://www.techtarget.com/searchdatacenter/definition/load-shedding

[9] C. Lim, "Relationship among benders, dantzig–wolfe, and lagrangian optimization," 2011. [Online]. Available: https://www.semanticscholar.org/paper/Relationship-among-Benders%2C-Dantzig%E2%80%93Wolfe%2C-and-Lim/6aae635d2e54eafe5b554e9fb17d9636194eaa78

[10] J. F. Benders, "Partitioning procedures for solving mixed-variables programming problems," *Numerische Mathematik*, vol. 4, pp. 238–252, 1962.

[11] A. Conejo, E. Castillo, R.Miguez, and R. Garcia-Bertrand, *Decomposition Techniques in Mathematical Programming Engineering and Science Applications*, 1st ed. Springer, 2006.

[12] "Artificial intelligence (ai) vs. machine learning," (read - 15.05.2023). [Online]. Available: https://mitsloan.mit.edu/ideas-made-to-matter/machine-learning-explained

[13] A. Haldorai, A. Ramu, and S. A. R. Khan, *Business Intelligence for Enterprise Internet of Things*, 1st ed. Springer, 2020.

[14] B. Mahesh, "Machine learning algorithms -a review," *International Journal of Science and Research*, 2018. [Online]. Available: https://www.researchgate.net/profile/Batta-Mahesh/publication/344717762_Machine_Learning_Algorithms_-A_Review/links/5f8b2365299bf1b53e2d243a/Machine-Learning-Algorithms-A-Review.pdf?eid=5082902844932096

[15] D. H. Maulud and A. M. Abdulazeez, "A review on linear regression comprehensive in machine learning," *Journal of Applied Science and Technology Trends*, vol. 01, no. 4, pp. 140–147, 2020.

[16] C. M. Bishop, "Neural networks and their applications," *Review of Scientific Instruments*, vol. 65, no. 6, pp. 1803—-1832, 1994. [Online]. Available: https://pubs.aip.org/aip/rsi/article/65/6/1803/682910/Neural-networks-and-their-applicationsNeural

[17] M. H. Sazli, "A brief review of feed-forward neural networks," *Communications Faculty of Sciences University of Ankara Series A2-A3 Physical Sciences and Engineering*, vol. 50, pp. 11–17, 2006. [Online]. Available: https://www.researchgate.net/publication/228394623_A_brief_review_of_feed-forward_neural_networks

[18] T. Kattenborn, J. Leitloff, F. Schiefer, and S. Hinz, "Review on convolutional neural networks (cnn) in vegetation remote sensing," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 173, pp. 24–49, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0924271620303488

[19] R. Solgi, H. A. Loáiciga, and M. Kram, "Long short-term memory neural network (lstm-nn) for aquifer level time series forecasting using in-situ piezometric observations," *Journal of Hydrology*, vol. 601, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022169421008507

[20] V. Kotu, A. Ramu, and S. A. R. Khan, *Predictive Analytics and Data Mining*, 1st ed. Chapter 2 - Data Mining Process, 2015.

[21] D. Harris-Birtill and R. Harris-Birtill, "Understanding computation time: A critical discussion of time as a computational performance metric," *Triennial ISST Conference on Time's Urgency*, no. 17, 2019. [Online]. Available: https://research-repository.st-andrews.ac.uk/bitstream/handle/10023/24947/Computation_Time_paper.pdf?sequence=1&isAllowed=y

[22] A. H. Murphy, "Skill scores based on the mean square error and their relationships to the correlation coefficient," *Journal of Sketchy Physics*, vol. 116, no. 12, pp. 2417–2424, 1988.

[23] W. Wang and Y. Lu, "Analysis of the mean absolute error (mae) and the root mean square error (rmse) in assessing rounding model," *IOP Conf. Series: Materials Science and Engineering*, vol. 324, no. 012049, 2018. [Online]. Available: https://iopscience.iop.org/article/10.1088/1757-899X/324/1/012049/meta

[24] "Mae and rmse — which metric is better?" (read - 24.06.2023). [Online]. Available: https://medium.com/human-in-a-machine-world/mae-and-rmse-which-metric-is-better-e60ac3bde13d

[25] "Synthetic data," (read - 18.05.2023). [Online]. Available: https://edps.europa.eu/press-publications/publications/techsonar/synthetic-data_en

[26] J. Yoon, D. Jarrett, and M. van der Schaar, "Time-series generative adversarial networks," *Conference on Neural Information Processing Systemss*, no. 33, 2019. [Online]. Available: https://papers.nips.cc/paper_files/paper/2019/hash/c9efe5f26cd17ba6216bbe2a7d26d490-Abstract.html

[27] "What is time series data?" (read - 18.05.2023). [Online]. Available: https://www.influxdata.com/what-is-time-series-data/

[28] S. Stage and Y. Larsson, "Incremental cost of water power," *Transactions of the American Institute of Electrical Engineers. Part III: Power Apparatus and Systems*, vol. 80, pp. 361–364, 1961. [Online]. Available: https://ieeexplore.ieee.org/document/4501045

[29] A. Turgeon, "Optimal operation of multireservoir power systems with stochastic inflows," *Water Resources Research*, vol. 16, no. 2, pp. 275–183, 1980.

[30] M. Pereira and L. Pinto, "Multi-stage stochastic optimization applied to energy planning," *Mathematical Programming*, vol. 52, pp. 359–375, 1991. [Online]. Available: https://link.springer.com/article/10.1007/BF01582895

[31] A. Helseth, B. Mo, and G. Warland, "Long-term scheduling of hydro-thermal power systems using scenario fans," *Energy Systems*, vol. 1, pp. 377–391, 2010. [Online]. Available: https://link.springer.com/article/10.1007/s12667-010-0020-7

[32] A. Helseth, B. Mo, A. L. Henden, and G. Warland, "Sovn model implementation," Trondheim, Tech. Rep., 2017.

[33] "About sintef - applied research, technology and innovation," (read - 15.06.2023). [Online]. Available: https://www.sintef.no/en/sintef-group/this-is-sintef/

[34] "Emps user manual," SINTEF Energy Research, Tech. Rep., 2016.

[35] S. Riemer-Sørensen and G. H. Rosenlund, "Deep reinforcement learning for long term hydropower production scheduling," *International Conference on Smart Energy Systems and Technologies*, vol. 13, pp. 1–6, 2020.

[36] A. Castelletti, S. Galelli, M. Restelli, and R. Soncini-Sessa, "Tree-based reinforcement learning for optimal water reservoir operation," *Water resources research*, vol. 46, no. 9, 2010. [Online]. Available: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2009WR008898

[37] O. B. Haddad and S. Alimohammadi, "Evaluation of artificial neural networks in optimization models of hydropower reservoir operation," *International Water Technology Conference*, pp. 985–998.

[38] I. KILINÇ and K. CIĞIZOĞLU, "Reservoir management using artificial neural networks," *Reg. Directorate of DSI (State Hydraulic Works)*, no. 14. [Online]. Available: https://balwois.com/wp-content/uploads/old_proc/ffp-471.pdf

[39] T. S. ABDULKADIR, A. W. SALAMI, B. F. SULE, and J. A. ADEYEMO, "Neural network based model for forecasting reservoir storage for hydropower dam operation," *International Journal of Engineering Research and General Science*, vol. 3, no. 5, 2015. [Online]. Available: https://www.researchgate.net/profile/Taofeeq-Sholagberu-Abdulkadir/publication/292970439_Neural_Network_Based_Model_for_Forecasting_Reservoir_Storage_for_Hydropower_Dam_Operation/links/57b3de8d08aeac317784a473/Neural-Network-Based-Model-for-Forecasting-Reservoir-Storage-for-Hydropower-Dam-Operation.pdf

[40] T. S. ABDULKADIR, . B. SULE, and A. W. SALAMI, "Application of artificial neural network model to the management of hydropower reservoirs along river niger, nigeria," *Annals of the Faculty of Engineering Hunedoara*, vol. 419, 2012. [Online]. Available: https://annals.fih.upt.ro/pdf-full/2012/ANNALS-2012-3-73.pdf

[41] B. V. Matheussen, O.-C. Granmo, and J. Sharma, "Hydropower optimization using deep learning," *Journal of Sketchy Physics*, vol. 13, no. 2, pp. 46–129, 2003.

[42] M. Stojković, S. Kostić, S. Prohaska, J. Plavšić, and V. Tripković, "A new approach for trend assessment of annual streamflows: a case study of hydropower plants in serbia," *Water Resour Manage*, vol. 31, pp. 1089–1103, 2017. [Online]. Available: https://link.springer.com/article/10.1007/s11269-017-1583-z

Appendix

A Appendices

NTNU

Norwegian University of
Science and Technology