UNIVERSITETET I AGDER

MASTER THESIS

Optimization of multi-reservoir pumped-storage hydropower system using differential evolution

Author:
Rohit SHARMA

Supervisor:
Joao LEAL and Torstein BJØRGUM

UNIVERSITY OF AGDER

Faculty of Engineering & Science

May 23, 2019

UNIVERSITY OF AGDER

Abstract

Faculty of Engineering & Science

MSc in Renewable Energy

Optimization of multi-reservoir pumped-storage hydropower system using differential evolution

by Rohit SHARMA

The study has proposed a long-term optimization model of a multi-reservoir pumped-storage hydropower system which is located in the Aust-Agder region. Based on the requirement a pump has been installed at a particular site on the same channel as of the turbine unit. The purpose of the pump installation is to elevate water from lower to the upper altitude reservoir during low power demand to avoid the overflow or flooding. The flow rate and power data provided by the company is used to do primary computations to calculate the necessary parameters. The data is discrete, therefore to get the variable expressions second-order polynomial technique is used for curve fitting. To compute the revenue and profit of the hydropower system, the price data for the site region, i.e., NO2-region is taken from NORDPOOL for Oct 2016 - Oct 2017.

Due to the large size of data, the optimization is executed by using daily average values, and then the profit achieved by the algorithm is presented. Further to check the performance of the algorithm a comparison is made between daily average and hourly optimization.

Based on the study, the differential evolution algorithm can accurately describe the long-term operation modes of pumped-storage hydropower system, and its calculation methods are appropriate for this kind of large-scale optimized decision problem.

Further, it is intended to give maximum revenue from the hydropower system within the flow and reservoir level limits. Due to its capability to identify different possible events occurring in the system, the DE algorithm provides encouraging solutions for discharge and level curves.

Acknowledgements

First and foremost I am astonishingly thankful to my supervisor, professor Joao Leal. He has been supportive since the days I began working on this project; Ever since Joao has supported me not only by giving research supervision but also academically and several insightful conversations and recommendations to complete this thesis.

I want to express my special acknowledgments to the Department of Engineering, the University of Agder for allowing doing this astonishing research.

I also want to thank Torstein Bjørgum, who is working as a chief engineer (electro) in Otra Kraft for his supervision and accommodating us with real power plant data. Because of Joao and Torstein, I also got an opportunity to visit study site to investigate the actual issue and got chance to present this research work at Otra hydropower station in front of Otra representatives which makes this research more unique toward learning point of view.

Declaration of Authorship

I, Rohit SHARMA, hereby declare that this research project titled, "Optimization of multi-reservoir pumped-storage hydropower system using differential evolution" and the work presented in it are my own. I confirm that this report:

- has not been used for another exam at another department/university/university college in Norway or abroad;
- Does not refer to the work of others without it being stated;
- Does not refer to own previous work without it being stated;
- Have all the references given in the literature list;
- Is not a copy, duplicate or copy of another's work or manuscript
- I am aware that violation of the above is regarded as cheating and may result in cancellation of exams and exclusion from universities and colleges in Norway, see Universitets- og høgskoleloven §§4-7 og 4-8 og Forskrift om eksamen §§ 31.
- I am aware that all submitted reports may be checked for plagiarism.
- I am aware that the University of Agder will deal with all cases where there is suspicion of cheating according to the university's guidelines for dealing with cases of cheating.
- I have incorporated the rules and guidelines in the use of sources and references on the library's web pages.

Signed: Rohit Sharma

Date: 23/05/2019

Publishing Agreement

Authorization for electronic publishing of the report. Author(s) have copyrights of the report. This means, among other things, the exclusive right to make the work available to the general public (Åndsverkloven. §2). All theses that fulfill the criteria will be registered and published in Brage Aura and on UiA's web pages with author's approval. Reports that are not public or are confidential will not be published.

1	I hereby give the University of Agder a free right to make the task available for electronic publishing:	Yes ☑	No ☐
2	Is the report confidential? (confidential agreement must be completed and signed by the Head of the Department) - If yes:	Yes ☐	No ☑
3	Can the report be published when the confidentiality period is over?	Yes ☐	No ☐
4	Is the task except for public disclosure? (contains confidential information. see Offl. §13/Fvl. §13)	Yes ☐	No ☑

Contents

List of Figures

List of Tables

List of Abbreviations

PSHPS	**P**umped-Storage Hydro Power System
TDH	**T**otal Dynamic Head
FRE	**F**luctuating Renewable Energy
TPT	**T**otal Pump Thrust
ITF	**I**mpeller Thrust Factor
GPM	**G**allon Per Minute
TBL	**T**hrust Bearing Loss
G1	**G**eneration Unit 1
G2	**G**eneration Unit 2
G3	**G**eneration Unit 3
HRV	**H**øyeste Regulerte Vannstand
LRV	**L**aveste Regulerte Vannstand
FCR	**F**requency Containment Reserves
TSO	**T**ransmission System Operators
FRR	**F**requency Restoration Reserves
max.	**M**aximum
min.	**M**inimum
lb	**L**ower Bound
ub	**U**pper Bound
DE	**D**ifferential Evolution
m.o.h	**M**eter over havet (sea level)

Physical Constants

Speed of Sound in penstock	$a = 2.997\,924\,58 \times 10^{8}\,\mathrm{m\,s^{-1}}$
Density of Water	$\rho = 1000\,\mathrm{kg/m^3}$
gravitational acceleration	$g = 9.81\,\mathrm{m/s^2}$
Frequency	$f = 50\,\mathrm{Hz}$
Dynamic Viscosity of Water	$\mu = 1.5 \times 10^{-3}\,\mathrm{kg\,s^{-1}\,m^{-1}}$

List of Symbols

P	power	W (J s^{-1})
H_L	Head loss	m
Q	Flow Rate	m^3/sec
m	Mass	kg
h_f	Pressure drop	$-$
f	Friction Factor	$-$
P_{shaft}	Shaft power	W
P_{pump}	Pump power	W
P_{HP}	Hydraulic power	W
P_{shaft}	Shaft power	W
A	Area	m^2
L	Length	m
SG	Specific Gravity	$-$
D	Diameter	m
Re	Reynolds number	$-$
A	Area	m^2
T_c	Time to close the main valve	sec
T	Temperature	°C
p	Pressure	pa
$V_{Reservoir}$	Reservoir volume	$Mill.m^3$
z_w	Reservoir level	m
$Pump_{rpm}$	Pump speed	rpm
$W_{rotor/stage}$	Rotor weight per stage	kg
ω	angular frequency	rad
η	Efficiency	%
ρ	Density	kg/m^3
μ	Dynamic viscosity	$mPa-s$
α	Coefficient of thermal expansion	$m/°Cm$
σ_t	Stresses in the pipe material	pa

Dedicated to my son…

Chapter 1

Introduction

In this chapter, an overview of the thesis structure together with the previous work and objective are described.

1.1 Motivation

The operational method of a standard hydropower system contributes a mechanism of controlling and adjusting the downstream water stream through the reservoirs. Despite, with increase in water inflow through the catchment or the location where reservoir capacity is low, and water inflow is high due to rain or snow melting, pumped storage is the most convenient approach to resolve this quandary. This approach is likewise useful in countries where water resources are limited, so that they can reuse water from reservoirs.

Various programs can do optimization of the Hydropower system. Previously the researchers used methods like **Dynamic programming (DP)**, **mixed Integer Linear programming (MILP)**, **Quadratic programming**, **FMINCON (Matlab)**, **Approximate Dynamic Programming (ADP)**, **Model predictive control (MPC)/Non-linear model predictive control (NMPC)**, and **Artificial Bee Colony (ABC) Algorithm**, to optimize the Hydropower system for one or more generation units .

The Computational precision of the DP model depends on the quantization degree of energy storage potential, and the curse of dimensionality remained unclear in the previous papers. The influential weakness in using dynamic programming (DP) is the number of partial solutions and we must retain a record. The partial solutions can completely define by particularizing the stopping points in the input data because the combinatorial objects determined on all have an original order specified upon their constituents. This order cannot combine without totally altering the problem (online, 1977). Once the order is set, there are comparatively less feasible stopping positions, so we arrange practical algorithms. On the other hand, the ADP can achieve better optimization results as compared with DP method because it can optimize both pumping duration and electricity generation. However, it is just an approximate method (Warren B. Powell, 2012).

The ABC algorithm is also utilized by the researchers to optimize the Hydropower operating system, and it also provides optimum results concerning water discharge curves. Still, this method is slow when it operates with subsequent processing. Furthermore, it requires a more significant number of objective function evaluations and demands a new fitness test on new Algorithm parameters (Choong and El-Shafie, 2015).

The results achieved by using MILP and Quadratic programming for long periods are not satisfactory too because it can solve only linear function. Furthermore, the discrete part of the system must transform into multi integer linear inequalities, and the efficiency of the programming depends on the tightness of the continuous linear programming recreations (Antonio Frangioni, 2009). To practice the large-scale optimization by using FMINCON has a limitation, the user must provide the gradient in function (fun). Additionally, the upper and lower bounds constraints can be specified, or only linear equality constraints must exist, and Aeq (rare variable for equality constraints) cannot hold more rows than columns (Han, 1977, Powell, 1978). Lastly, the FMINCON method for optimization gives unsatisfactory results as shown in Appendix E we can observe that FMINCON gives results that reach to the lowest value of flow rate for the same problem that is executed in this thesis.

The adoption of Differential Evolution method for optimization is made by taking into account its capability to handle non-differentiable and nonlinear cost functions, parallelizability to manage calculation for simple cost functions and Genuine convergence characteristics, i.e., uniform convergence to the global minimum in following independent cases.

1.2 Previous research

The primary purpose of pump installation is to reutilize the water at Botsvatn. During the low power demand, the water from Botsvatn reservoir gets discharged to avoid the risk of exceeding HRV (Høyeste regulerte vannstand) due to which the Brokke power station forced to operate. To troubleshoot this problem, a pump system has to be established between the reservoir Botsvatn and Urevatn to reutilize the water in an optimized way.

Previously, in the energy research project, a pump was chosen for the Otra kraft hydropower site. The company requires to install pump in Botsvatn reservoir on the same channel as of Holen-3 turbine unit. The selected pump is shown in figure 1.1 is a multistage submersible pump of flow rate capacity 4.43 m^3/s which can pump water up to 1000 m head but in actual the head needed is 650 m as shown in Figure 3.6. However, the flow requirement is 20 m^3/s. Therefore, four such pumps can be connected in parallel to reach this flow rate. The maximum speed of the pump is 740 rpm and has an efficiency of 83% per stage. The corresponding power is 4146 kW or 5637 hp.

FIGURE 1.1: Selected pump for the site (Pumps, Goulds online)

In addition to the pump selection, the estimation of various important factors during the selection of pump for example efficiency, total dynamic head and required power is executed as shown in Appendix F.

1.3 Objective

The planning is to optimize the system by using Differential evolution algorithm in MATLAB based on the actual historical data provided by Otra Kraft DA and further to examine different scenarios by utilizing hourly, daily and weekly data consequently to establish an optimized operation concerning flow rates and reservoir levels for a year (Oct 2016 - Oct 2017). Subsequently, the mass balance procedure for getting the volume and flow rate from the catchments developed previously in the energy research project (Sharma, 2018) for hourly data and was applied in the thesis to get daily and weekly average curves.

In chapter 2, the introduction about the Norwegian hydropower system and the electricity market along with the market strategies is briefed.

In chapter 3, a detailed description of study site and problem description is mentioned together with the company operational and background information.

In chapter 4, The data that is provided by the company is used to estimate various power, flow, and volume relations to use it for the optimization. To handle the discrete data second order polynomial fit technique is used to establish fitted curves which estimate relations between the decision variables in the algorithm. Further, due to the high volume of available data, the outlining is to conduct a categorized optimization.

In chapter 5, the optimization is carried for daily average value throughout the year to determine the optimize reservoir level, flows, volume to generate maximum revenue from proposed system. Furthermore, a hourly optimization is executed for the randomly chosen weeks throughout the year and compared with the daily average optimization results to check the accuracy of proposed algorithm. Lastly, the results for hourly optimization is presented and compared with the actual operational values of the system to show the benefit of proposed algorithm.

1.4 Research Questions

- Is it feasible to establish a pump at given site on the same channel as of turbine?
- Will this strategy competent in providing optimize results or deliver a profit?
- Is it conceivable to maintain the mass balance between the reservoir with the pumping system?
- Is differential evolution is an attractive option to achieve optimization goal?

Chapter 2

Hydropower System

2.1 Introduction

Hydropower engineering deals with the different techniques for transforming flow water energy into the electricity. The main elements in hydropower stations are Reservoirs (dams), pipes, Generator, and turbines. To generate an adequate amount of energy from the system, these determinants should be constructed or organized accurately (Kaltenekker, 2018). The process of hydropower generation demands to create a barrier in the route of streaming water. The water is routed and compelled to drop where its potential energy transformed into kinetic energy, and this energy extracted through turbines. The range of hydropower is remarkably vast, and energy can be generated from a few watts to several gigawatts. The biggest power plant is a hydro energy project called the "Three Gorges Dam" in China it has a nameplate capacity of a whopping 22 gigawatts.

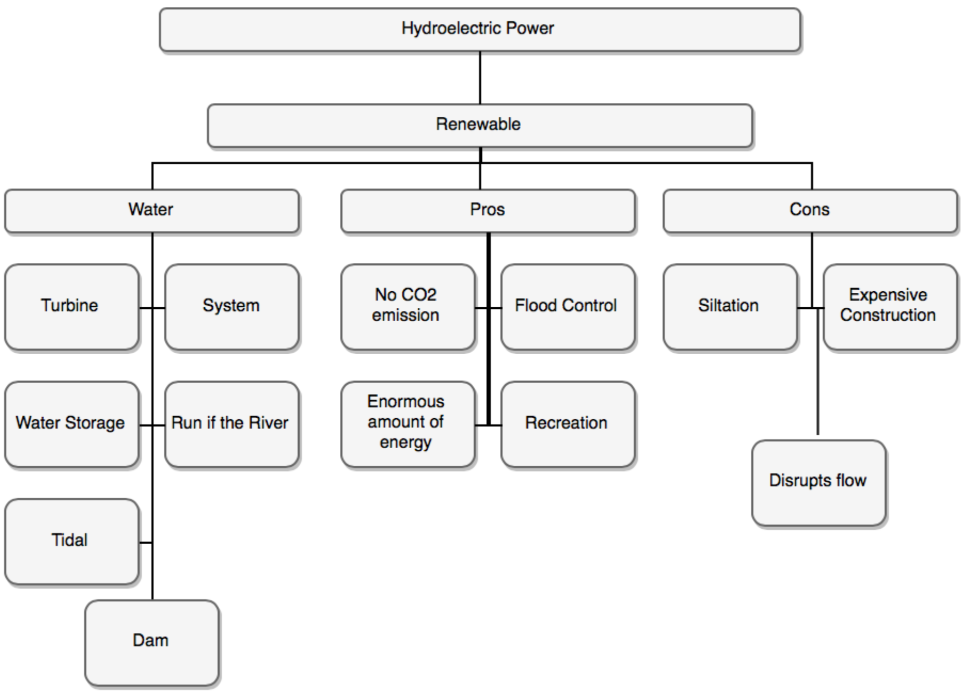

FIGURE 2.1: Hydropower Introduction

The energy derived from hydropower is renewable energy, as water is continually replenished by nature because of the water cycle and there is no CO_2 emission when converting water kinetic energy into electrical energy. Streams and rivers all have varying levels of water flow. The flow rate through catchments also varies along the year so as the head of water; therefore to extract energy from water stream there are unique types of turbines available. The head of water is the measure of hydrostatic energy of water. It is merely the height of water over a certain point. The flow rate is the volume of water passing a specific point in a second. Based on the head of water and flow rate at a specified location, the turbine type is selected for example Kaplan, Francis, Pelton wheel and cross-flow turbine .

2.2 Norwegian Hydropower

Norway is a beautiful country gifted with natural resources and geography which facilitates to develop a large number of hydropower stations. Norway represents about 50 percent of the reservoir capacity in Europe. Furthermore, the reservoir is an essential medium to decrease floods and droughts while producing clean, renewable and affordable energy (Statkraft, 2018).

Modern Norway industrialized after utilization of waterfalls and rivers to produce electricity. Hydropower is one of the primary factors for economic growth in Norway and will remain so in the foreseeable prospect. From the prior 100 years after the first hydroelectric power plant built in Norway, approximately 99% of the electricity production based on hydropower. In the late 19^{th} century, the rights to build power plants in Telemark county achieved by an industrialist named "Sam Eyde," the determination was to convert the natural water to low-cost electricity for the industrial expansion. Afterward, many precious companies are established. Today, Norwegian companies contribute to the advancement of proficiency in nations with hydropower resources (Kjersti, 2015). Figure 2.2 represents the contribution of hydropower in the Nordic countries. It can be observed that Norway principally generate electricity from the hydropower system. The contribution of traditional thermal and Nuclear power is nearly negligible.

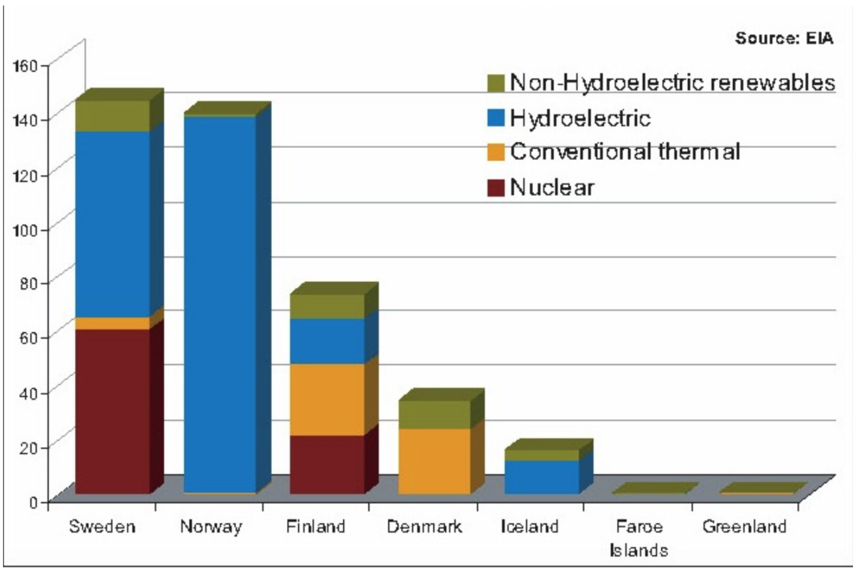

FIGURE 2.2: Hydropower production in Nordic countries (Nordegio, 2008)

2.2.1 Importance of Norway in contributing large-scale balancing power:

According to the German Advisory Council on Environment (2011) the Norwegian hydropower system remains a significant option for balancing variable renewable energy production in Northern Europe, (Julie Charmasson, 2017).
The Norwegian hydroelectric power stations can increase the balance power capacity by raising the consumption potential and by establishing Pump storage hydropower system. The balance power potential depends on the evaluation of power supplied during the periods of high demand and power absorption during low demand. The power absorption can be done by using excess generated power for pumping water to higher reservoirs as to use the stream again for power production when the electricity demand is high. This can also be used to avoid unnecessary production of electricity when the electricity demand is low. Enhanced exploitation of reservoirs for generating balance power is organized to ensure a place in agreement with existing laws governing the highest and lowest regulated water levels (HRV and LRV) .

FIGURE 2.3: Cable links to the United Kingdom, the Netherlands, Germany and Denmark (Eivind Solvang, 2012)

Figure 2.3 represents the cable links within the proposed hydropower plants in south-western Norway to the United Kingdom, the Netherlands, Germany, and Denmark (**Note:** these connection does not designate the expected cable connections between the countries which means the connection lines can be interchanged).

In order to use Norwegian hydropower reservoirs as rechargeable batteries for the European power supply more transmission capacity must be installed between Norway and the major European consumption regions. Pumped storage power solutions must improve in such a way that the environmental impacts reduce to the minimum. The climate threat is a global challenge. Moreover, the further development of the Norwegian hydropower can contribute to the more reliable and clean European energy supply.

Today, Norway has a total of 17 power connections abroad. Norway is an integrated part of the Nordic power grid, and have power cables for Sweden, Denmark, the Netherlands, and Finland, as well as one for Russia. A cable is being built for Germany and one for England. Besides, a power cable to Scotland during application processing (Abrenna, 2019).

Total Electricity imported/exported

The data obtained from Nordpool power exchange concerning the amount of electricity imported and exported across the international cables every single hour throughout 2018. From the figures, we can observe the export surplus of 10.2 TWh In total, Norway had a net export surplus of 10.2 TWh in 2018. If we begin with standard Norwegian consumption of 20,000 kWh a year, this means that Norway had a total net export surplus corresponding to enough electricity to satisfy 510,000 Norwegian homes. If we split it down on each country, then we recognize that Norway has a net export surplus to all countries except one ,i.e. Russia. Norway exported 2.5 TWh to Denmark, 3.7 TWh to the Netherlands and 4.0 TWh to Sweden. The three countries collectively accounted for 10.2 TWh of Norway's net electricity exports. Norway also had a net surplus of electricity exports to Finland, but it amounted only 0.1 TWh. The only country Norway did not have net electricity exports is Russia. It is unfamiliar, as Norway import electricity from there. The capacity is also deficient as few people know that Norway has a separate international cable there. In 2018, Norway imported a total of 19.0 GWh, i.e. 0.019 TWh, from Russia in the east (NORDPOOL, 2018).

Import and export throughout the year

If we arrange data for imports and exports of electricity down every month, we notice that Norway has net exports of electricity in all months excluding March and April. Figure 2.4 shows that exports and imports of electricity fluctuate widely from day to day. In 94 of 365 days, Norway imported more electricity than exported. It varies significantly from day to day whether Norway has net exports or imports of electricity. However, we can witness there is a more substantial number of days of export.

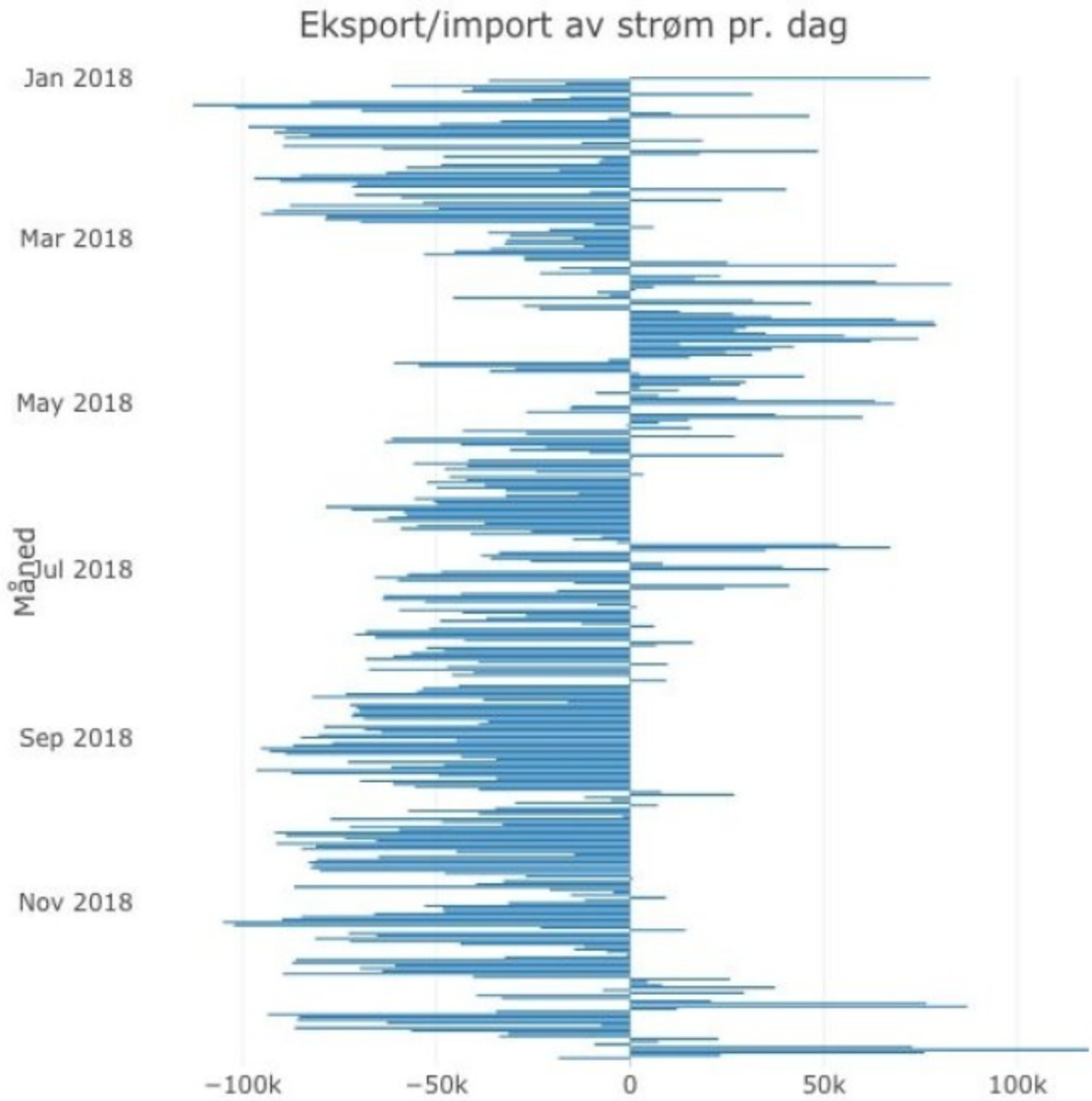

FIGURE 2.4: Figure showing net export/import on daily basis (Abrenna, 2019)

Import of Electricity throughout a day

It is not just day to day that fluctuates between net exports and imports of electricity. It also fluctuates throughout the day. In a typical day, Norway imports electricity at night and exports during the day. Specifically, this means that from 00:00 to 06:00 Norway ordinarily import electricity from abroad, while Norway has net exports for the rest of the day. If we break the numbers down in each country, we see that there is a little different from country to country. Norway import electricity from both Sweden and Denmark at night, but from Sweden, Norway has net imports already from 11:00 p.m. Figure 2.5 at the top left illustrates Norway imports and exports electricity in total to Denmark, Finland, the Netherlands, Russia, and Sweden. When the bars are above the line, they confer net imports, while they show net exports when they are below the line.

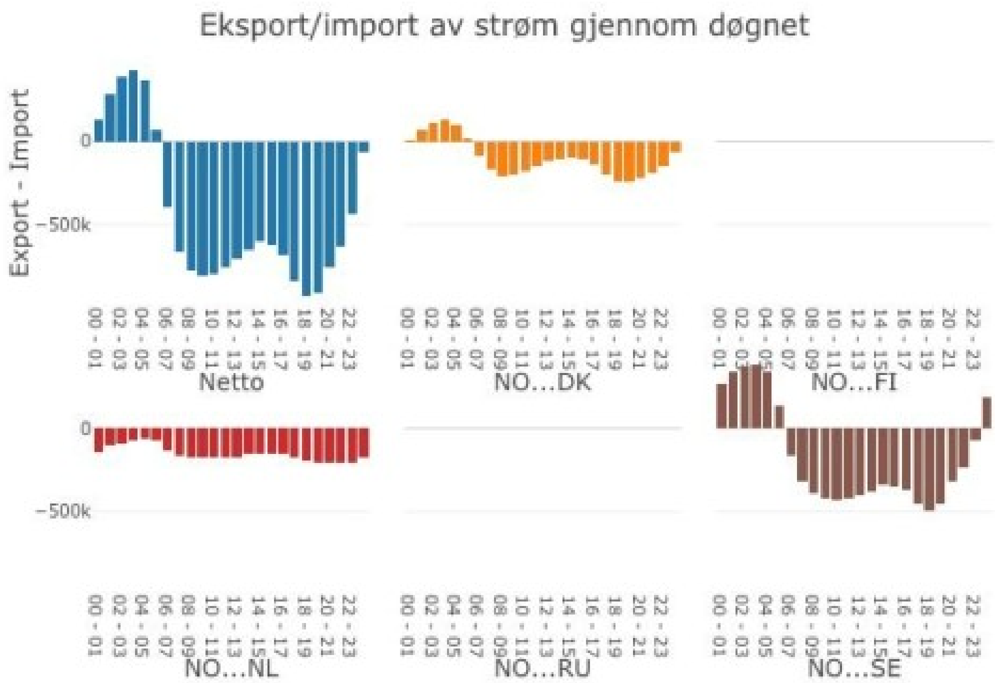

FIGURE 2.5: Figure showing net export/import during 24 hours a day (Abrenna, 2019)

2.3 Typical Hydropower System

Figure 2.6 pictures the typical hydropower plant. It consists of two reservoirs one remains at the bottom level (optional), and the other one is on the upper level. While the energy demand is high or if there is a requirement of electricity production the gates of upper reservoir opened, and the water from the upper reservoir flows to the lower reservoir through the pipe network. The high-pressure water streams through the turbine which initiates it to rotate and the mechanical energy of the turbine get transmitted to the electric energy (Mosonyi, 2016).

Production of electricity depends on the head of the water from ground level (H_{net}), the volume of water flowing per unit time or flow rate (Q) and efficiency of a turbine (η). Water stored in the reservoirs has potential energy the water under pressure is carried by penstock and supply to the turbine through the inlet valve where penstock is a pipe or tunnel made of steel or concrete. Due to the force of water, the turbine starts rotating due to which the mechanical energy produced since the turbine shaft is attached to the generator. Therefore, the generator produces electricity and the voltage of electricity raised by using a transformer and further transferred by distribution lines (Whiticar, 2016).

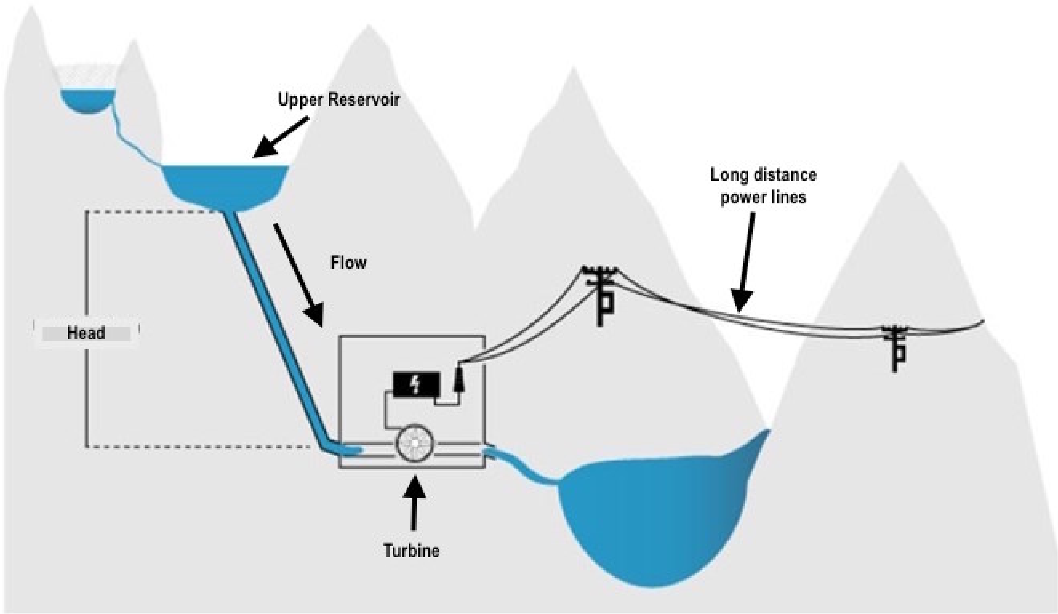

FIGURE 2.6: Standard Hydropower Plant,(Statkraft, online)

2.4 Pumped-Storage Hydropower System

The reservoir-based pumped storage plant is an evolution of the conventional hydropower plant to allow it to manipulate reversibly. As in a conventional hydropower plant with a reservoir in which water collects in the reservoir and then discharged through the plant's turbines concerning the power demand.

However, in conventional hydropower plant the water in the reservoir can be used only once. Whereas in a pumped storage plant shown in figure 2.7 there is another reservoir below the turbine hall. The water discharged from the first reservoir used for the generation of electricity can be collected in the second reservoir and pumped back into the first reservoir by the utilization of a pump. Hence water is cycled within the two reservoirs to accommodate either power or energy storage demand. This type of power-plant is remarkably robust and higher in potential (Ioannis Kougias, 2017).

During the low energy demand surplus power could be utilized to elevate water from low to high altitude reservoirs. This water could, in turn, be released to generate power on the periods when demand is high this loop can recite again and again.

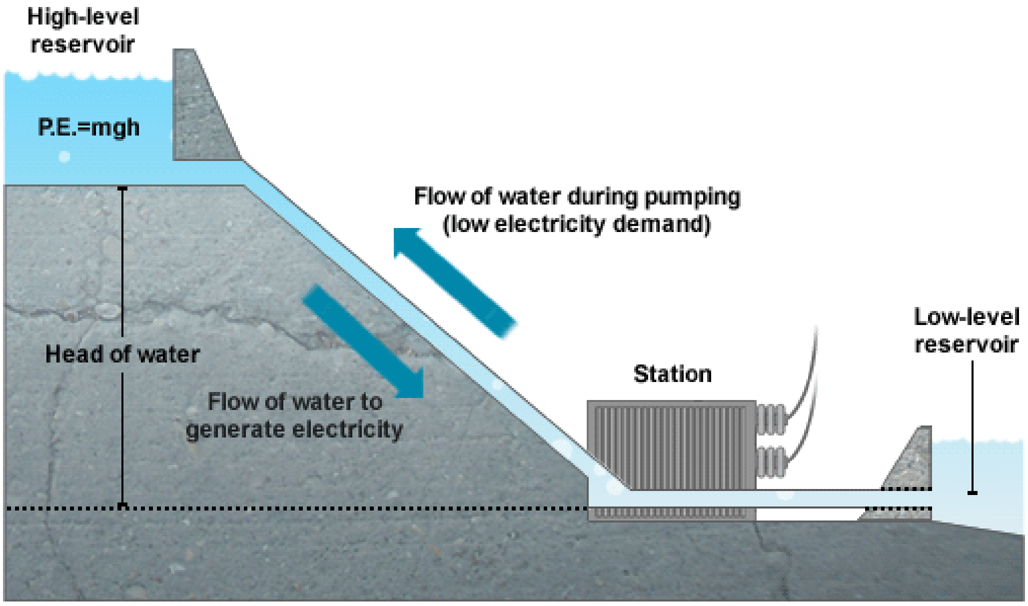

FIGURE 2.7: Pumped Hydropower Station,(Whiticar, 2016)

2.5 Energy and power equations in Hydropower system

2.5.1 Energy potential

The Bernoulli theorem can be applied to estimate the energy of the system. The Bernoulli equation implies a constant discharge rate and declares that the energy head at any point in the system is equivalent to any point downstream in the system plus any losses, for example, frictional losses in the pipe, head losses, pump/turbine losses, etc.. The Bernoulli equation can be expressed as:

$$z_1 + \frac{v_1^2}{2g} + \frac{p_1}{\gamma} = z_2 + \frac{v_2^2}{2g} + \frac{p_2}{\gamma} + h_L \tag{2.1}$$

Where, z is elevation head (m), p is pressure (N/m^2), h_L is total head loss between upper and lower reservoirs (m), v is velocity (m/s), and γ is defined as specific weight (N/m^3).

2.5.2 Head loss equation

To calculate the head loss in pipe following equation can be used:

$$H_l = f * \frac{L * v^2}{D * 2 * g} \tag{2.2}$$

Where, f is friction factor, D is diameter of pipe (m), L is length of pipe and v is velocity of water in pipe (m/s).

Further, the friction factor for turbulent flow (Re>2300) can be computed using the Colebrook-White equation (Munson and Okiishi, 1998b):

$$\frac{1}{\sqrt{f}} = -2log_{10}(\frac{\varepsilon}{3.7D} + \frac{2.51}{Re\sqrt{f}}) \tag{2.3}$$

Where ε/D is pipe relative roughness, and Re is defined as Reynold's number which can be computed using:

$$Re = \frac{4 * \rho * Q}{\pi * \mu * D} \tag{2.4}$$

Where, ρ is density of fluid (kg/m^3), μ is the dynamic viscosity of fluid (kg/sm), and D is the diameter of pipe (m).

2.5.3 Power

Power can be defined as the rate of utilized energy by change in time period or can be defined as the product of the specific weight of water, hydraulic head, and the discharge rate which is also known as hydraulic power equation.

$$P_{HP} = \rho.H.g.Q \tag{2.5}$$

The power generated during turbine operation is calculated as:

$$P_{turbine} = \rho.H.g.Q.\eta_t \tag{2.6}$$

Where, $P_{turbine}$ is the power from turbine generation unit, and η_t is the turbine efficiency.
The power required to pump the water to upper reservoir can be calculated as:

$$P_{pump} = \frac{\rho.H.g.Q}{\eta_{pump}} \tag{2.7}$$

Where, P_{pump} is the power from pump unit, and η_{pump} is the pumping efficiency.

2.6 Pumped Storage Influence on Reservoir

In Pumped storage hydropower system behaves like a battery, storing power in the form of water during electricity demands are low and producing maximum power during daily and seasonal peak times. An advantage of pumped storage is that hydroelectric generating units are capable of starting up instantly and perform fast regulations in output. They work efficiently while used for one hour or several hours.

The inconstancy of reservoir levels is due to the inflow and outflow of water. The primary reference of inflow to the reservoir is the catchment. In the pumped storage power station the reservoir receives intake from the pump too. As a consequence, the frequency of draining and filling of the reservoir will increase. Moreover, the temperatures in downstream reservoirs will reduce which affect the ice formation (Whiticar, 2016).

2.7 Environmental Impacts of Reservoir/Dams

The dams comprise impacts to the biological, chemical and physical properties of the river. By regulating the river flow or by creating a dam alters the oxygen level and stops the sediments that would naturally replenish the river and that will put an impact on the aquatic plant and animals present in the particular site. Reservoirs often host non-native and invasive species for example snails, algae, etc. Which means reservoirs become breeding territories for disease vectors. It ensures correct particularly in tropical areas where mosquitoes (which causes malaria) and snails (which causes Schistosomiasis) can take benefit of this slow streaming water. Furthermore, large dams or high capacity dams have led to the extinction of various fish and other aquatic species (Wikipedia, 2019a).

2.8 Electricity Market

In the Nordic countries, the generated electricity traded to the day-ahead market of Nordpool Spot. The spot price for every hour estimated from generation and consumption proposals presented by associates/partners (NORDPOOL, 2018).

The Norwegian Energy Act established on the principle that electricity generation and trading should be market-based, while grid operations are rigorously regulated. The power market guarantees the adequate utilization of resources and reasonable prices on electricity. Electricity transmission and distribution is a fundamental monopoly, and not subject to competition. However, the day-ahead and intraday markets make an offset within generation and consumption. There are bound to be an occurrence that disturbs the offset within a particular hour of operation. In the Nordic region, the balancing markets classified into primary reserves (FCR), secondary reserves (FRR-A) and tertiary reserves (FRR-M). Primary and secondary reserves are initiated automatically in reply to variations in frequency, while tertiary reserves are initiated manually by the Nordic Transmission System Operators (TSOs) (Energy facts, 2019).

In this project, only the Elspot prices are considered, as they are the only appropriate references of revenue for the plant in interrogation.

2.8.1 Nord Pool

Nord Pool is Europe's largest electricity exchange market measured in volume (512 TWh) in 2017. Nord Pool trades more than 80% of the total consumption of electrical energy in the Nordic market where companies can buy electricity at the market price. However, electricity prices are frequently regulated within each country (Wikipedia, 2019b).

Chapter 3

Study location

3.1 Introduction

The Hydropower system is located in Valle municipality (Setesdal), shown in figure 3.1 and the boundaries in the valley pass to the south towards Bygland and the north towards Bykle municipality. The prime source of income for the Valle and Bykle municipalities originate from Hydropower. In the 1960s the improvement of the power plants in these municipalities began. Otra Kraft, which is located at Brokke power plant at Nomeland, operates several power stations in Setesdal.

Presently, Otra Kraft owns three power station (Brokke, Holen, and Skarg). Water from Botsvatn reservoir is utilized by Brokke power station which is joined by 31 km long tunnel to Botsvatn reservoir. Holen consists of two Hydropower stations Holen1-2 and Holen3, Holen1-2 utilizes water from Vatnedalsvatn reservoir, and from six branches and streams that are redirected into the intake of tunnel. Furthermore, lake Ormsavatnet also supplies this power station. The other power station (Holen3) uses water from Urevatn, Reinevatn, and Skarjesvatn reservoirs. Holen1-2 generating units was approved in 1981. Whereas, Holen3 generating unit was approved in 1986 and Brokke generation unit was approved in 1965. During the last ten years, these power stations collectively generated 2,710 million kWh, which represents 2% of Norway's total power production. This Hydropower system owned by Skagerak Energi AS (31.4%) and Agder Energi AS (68.6%). Otra Kraft DA is responsible for everyday operations, and administers the licenses and obligations in terms of river/flow management and controlling of the watercourse (Energi, 2019, Kraft, 2019).

FIGURE 3.1: Hydropower site - Aust-Agder (maps, 2019)

Figure 3.2 shows the connectivity of These three power station and the reservoirs (Botsvatn, Urevatn, and Vatnedalsvatn). All three units attired with Francis turbines. The head for generating units 1 and 2 (Holen1-2) is 316 meters, while the head for the third unit (Holen3) is 650 meters. Table 3.1 shows the parameters related to the reservoirs.

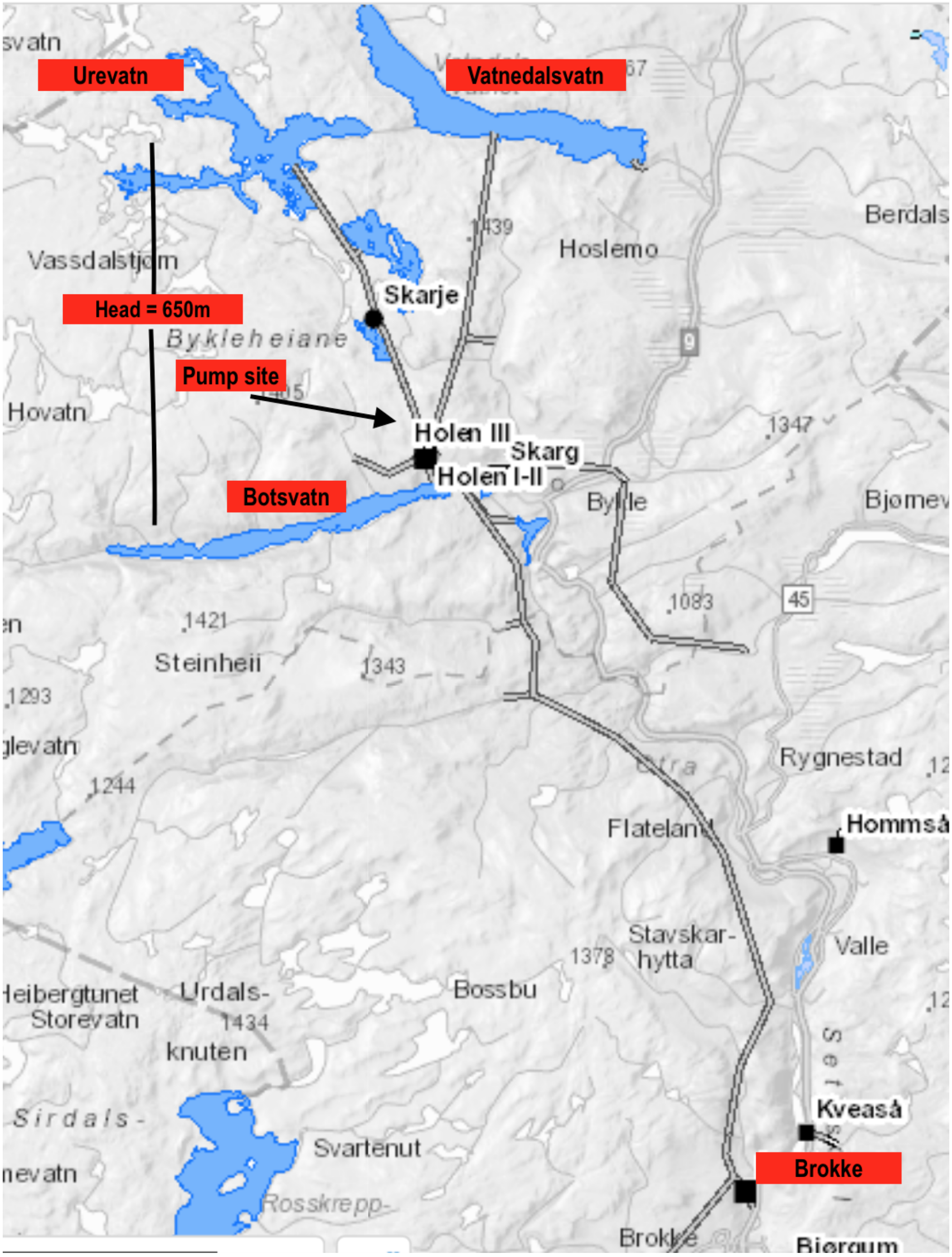

FIGURE 3.2: Pumped Hydropower Station,(Atlas, 2018)

Reservoir	Volume[Millm^3]	HRV[m.o.h]	LRV[m.o.h]	Station
Urevatn	253.4	1175	1141	Holen 3
Vatnedalsvatn	1150	840	700	Holen 1-2
Botsvatn	296	551	495	Brokke

TABLE 3.1: Reservoirs data (NVE Atlas, 2018)

3.2 Botsvatn Reservoir

Figure 3.3 shows the catchment area of Botsvatn reservoir. This figure is generated from NEVINA, 2019. The estimation of the catchment is vital to perform the mass balance in the reservoirs.

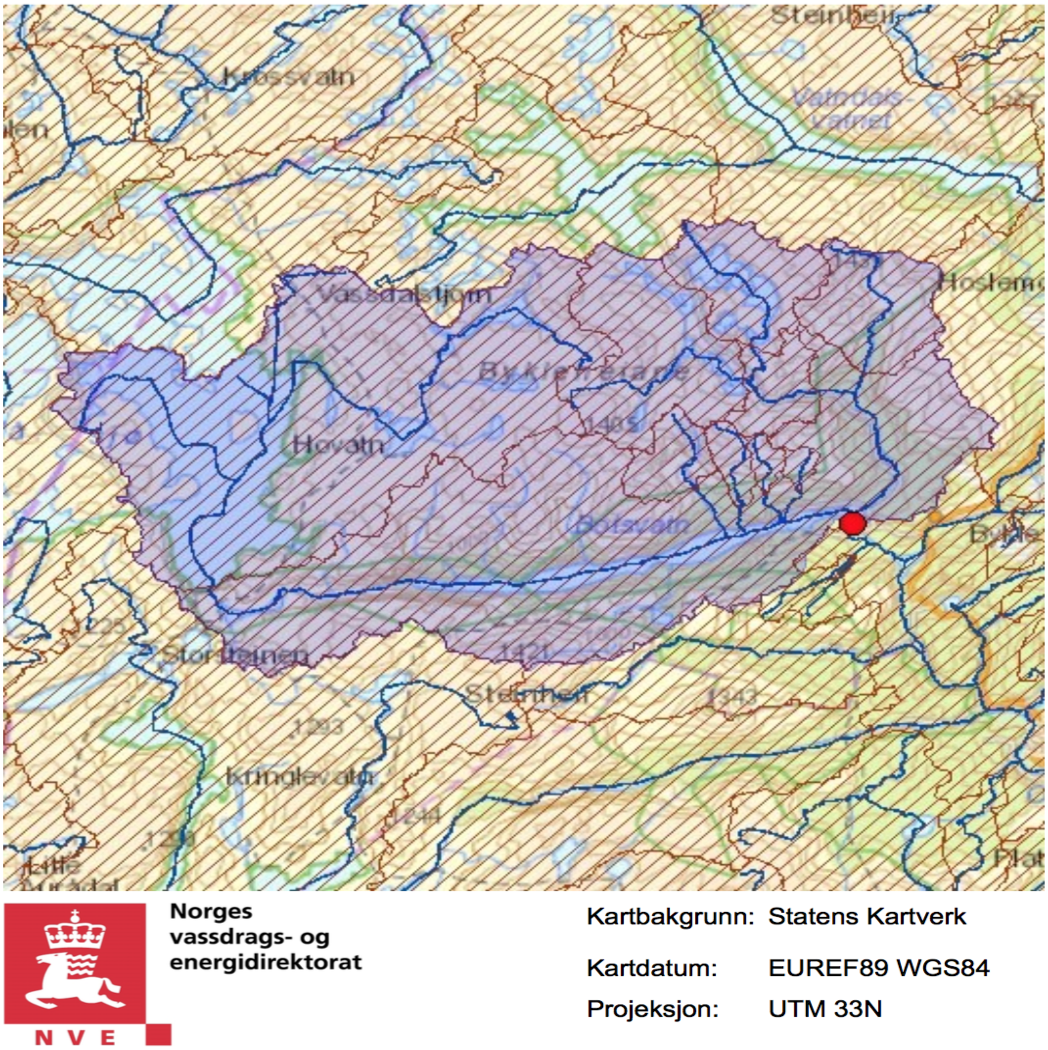

FIGURE 3.3: Figure showing catchment area at Botsvatn reservoir

Following tables 3.2 and 3.4 gives the Precipitation boundaries, field parameters, and water flow indices which are generated from NEVINA, 2019.
The discharge records or stream-flows which are mentioned in table 3.3 is eminent for the flood-risk management since they present essential knowledge for planning the operations before, through or after floods. The discharge records help enhance the precision of hydrology models utilized for forecasting stream-flows. However, the flow data is not only beneficial for flood forecasting or monitoring but also for numerous different practical uses for example for securing healthy ecological flows (Muste and Hoitink, 2017). The flood values give the extent of the culmination floods for varying repeat intervals. These values are generated from NVE –Report 7/2015 *"Guide to flood calculations in small, unregulated fields."*

However, It still demands an investigation to determine climate consequences for instantaneous floods in small precipitation fields. Until the results from these projects are available, a climate impact of 1.2 is recommended for the 24-hour flood and 1.4 for the culmination flood in small precipitation fields (NEVINA, 2019).

TABLE 3.2: Low water indexes and climate figures.

Waterway No .:	021.GB2
Municipality:	Bykle
County:	Aust-Agder
Waterways:	Bossvassåi
Water flow index	
Medium water flow (61-90)	72.3 $\mathrm{l}/(s*km^2)$
Ordinary low water flow	3.9 $\mathrm{l}/(s*km^2)$
Base flow	31.1 $l/(s*km2)$
BFI	0.4
Climate	
Climate region	South Norway
Summer precipitation	519 mm
Winter precipitation	906 mm
Annual temperature	-0.3 °C
Summer temperature	5.2 °C
Winter temperature	-4.2 °C
July temperature	7.4 °C
August temperature	8.4 °C

TABLE 3.3: Physical Characteristics (Botsvatn)

	Q^M $[m^3/s]$	Q^M $[l/(s*km^2)]$	**Q5**	**Q10**	**Q20**	**Q50**	**Q100**	**Q200**
Flood Frequency Factors	-	-	1.21	1.42	1.64	1.98	2.28	2.63
95 % interval upper limit (m^3/s)	330.5	1000.6	410.0	490.0	579.5	722.0	853.3	983.5
Flood values (m^3/s)	186.7	565	226.5	264.9	306.6	370.2	426.6	491.7
95 % interval lower limit (m^3/s)	105.5	319	125.2	143.2	162.2	189.9	213.3	245.9
Floods with climate impact(m^3/s)	261.4	791.4	226.5	370.8	429.3	518.3	597.3	688.4

TABLE 3.4: Botsvatn field parameters.

Field parameters	
Area (A)	330.3 km^2
Effective Sea (S_eff)	3.1 %
River Length (E_L)	50.3 km
River gradient (E_C)	-10.5 m/km
River gradient_1085 (G_{1085})	16.7 m/km
Field length (F_L)	25.7 km
H_min	550 moh
H_10	799 moh
H_20	968 moh
H_30	1055 moh
H_40	1069 moh
H_50	1109 moh
H_60	1148 moh
H_70	1186 moh
H_80	1222 moh
H_90	1271 moh
H_max	1435 moh
Bre	0.0 %
Cropland	0.0 %
Swamp	0.6 %
sea	18.7 %
Forest	12.0 %
Bare mountain	67.7 %
Urban	0.1 %

3.3 Vatnedalsvatn Reservoir

We can observe in figure 3.4 the catchment of Vatnedalsvatn included Urevatn reservoir which means the water from Urevatn (spillway) will stream towards Vatnedalvatn reservoir hence, during the mass balance of Urevatn and Vatnedalsvatn this practical approach has been taken into account.

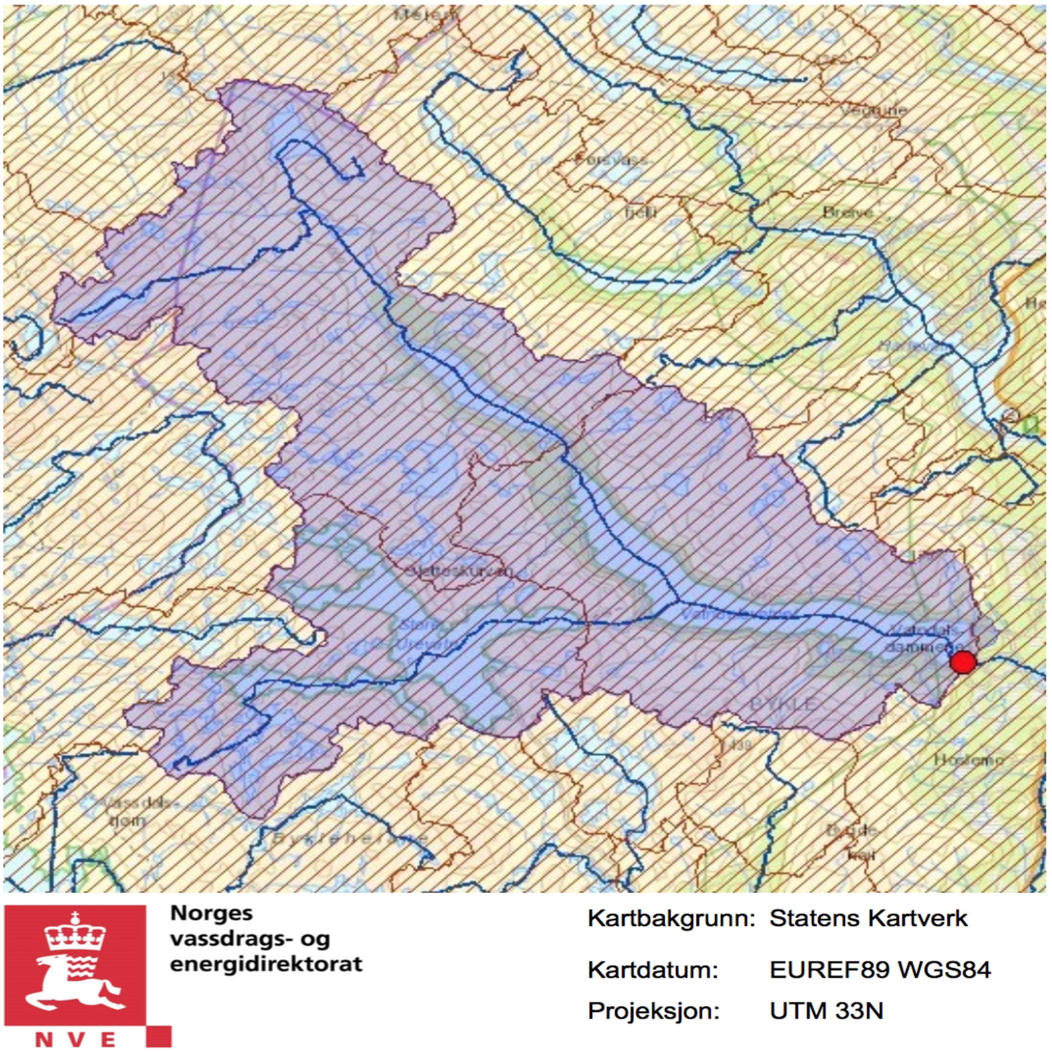

FIGURE 3.4: Figure showing catchment area at Vatnedalsvatn reservoir

Following tables 3.5 and 3.7 gives the Precipitation boundaries, field parameters, and water flow indices for Vatnedalsvatn reservoir which are generated from NEVINA, 2019.

TABLE 3.5: Low water indexes and climate figures (Vatnedalsvatn).

Waterway No .:	021.HB2
Municipality:	Bykle
County:	Aust-Agder
Waterways:	Løyningsåni
Water flow index	
Medium water flow (61-90)	61.3 l/$(s * km^2)$
Ordinary low water flow	4.4 l/$(s * km^2)$
Base flow	26.4 $l/(s * km2)$
BFI	0.4
Climate	
Climate region	South Norway
Summer precipitation	505 mm
Winter precipitation	855 mm
Annual temperature	-0.5 °C
Summer temperature	4.8 °C
Winter temperature	-4.2 °C
July temperature	6.9 °C
August temperature	8.1 °C

TABLE 3.6: Physical Characteristics (Vatnedalsvatn)

	Q^M $[m^3/s]$	**Q^M** $[l/(s * km^2)]$	**Q5**	**Q10**	**Q20**	**Q50**	**Q100**	**Q200**
Flood Frequency Factors	-	-	1.21	1.44	1.69	2.08	2.44	2.87
95 % interval upper limit (m^3/s)	166.6	692.5	206.6	250.1	300.2	382.0	459.4	539.5
Flood values (m^3/s)	94.1	391	114.2	135.2	158.8	195.9	229.7	269.8
95 % interval lower limit (m^3/s)	53.2	221	63.1	73.1	84.0	100.5	114.9	134.9
Floods with climate impact(m^3/s)	131.7	547.8	114.2	189.3	222.3	274.2	321.6	377.7

TABLE 3.7: Vatnedalsvatn field parameters.

Field parameters	
Area (A)	240.5 km^2
Effective Sea (S_eff)	8.0 %
River Length (E_L)	34.6 km
River gradient (E_C)	-23.7 m/km
River gradient_1085 (G_{1085})	10.3 m/km
Field length (F_L)	26.5 km
H_min	795 moh
H_10	892 moh
H_20	1019 moh
H_30	1092 moh
H_40	1143 moh
H_50	1175 moh
H_60	1189 moh
H_70	1209 moh
H_80	1233 moh
H_90	1269 moh
H_max	1477 moh
Bre	0.0 %
Cropland	0.0 %
Swamp	0.3 %
sea	20.0 %
Forest	9.1 %
Bare mountain	70.6 %
Urban	0.0 %

3.4 Urevatn Reservoir

Figure 3.5 represents the catchment area of Urevatn reservoir does not include any other reservoir in the system, so water from spillway does not have any connection with other reservoirs.

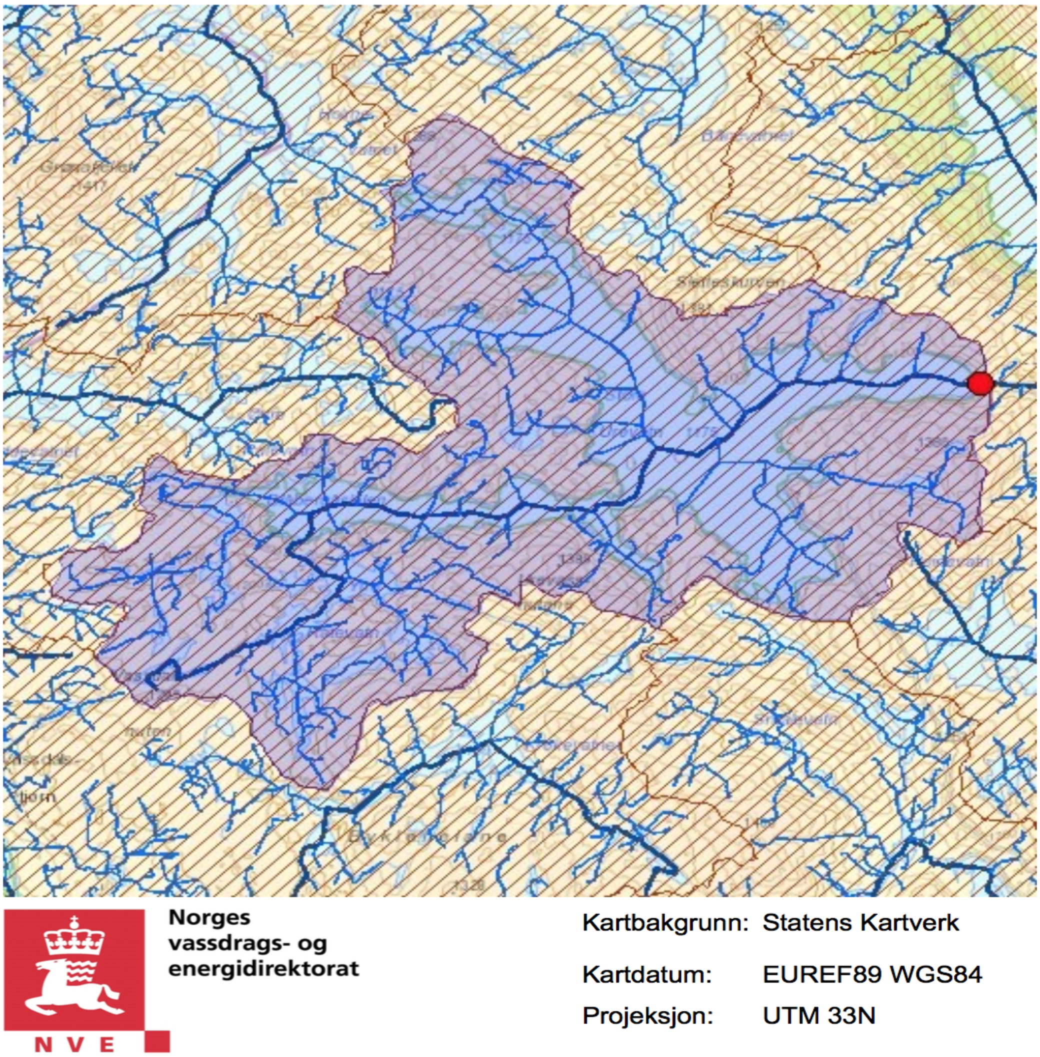

FIGURE 3.5: Figure showing catchment area at Urevatn reservoir

Following tables 3.8 and 3.10 gives the Precipitation boundaries, field parameters, and water flow indices for Urevatn reservoir which are generated from NEVINA, 2019.

TABLE 3.8: Low water indexes and climate figures (Urevatn).

Waterway No .:	021.HBB
Municipality:	Bykle
County:	Aust-Agder
Waterways:	Uraåni
Water flow index	
Medium water flow (61-90)	77.2 l/$(s * km^2)$
Ordinary low water flow	6.9 l/$(s * km^2)$
Base flow	34.0 $l/(s * km2)$
BFI	0.4
Climate	
Climate region	South Norway
Summer precipitation	540 mm
Winter precipitation	929 mm
Annual temperature	-1.3 °C
Summer temperature	4.3 °C
Winter temperature	-5.4 °C
July temperature	6.5 °C
August temperature	7.6 °C

TABLE 3.9: Physical Characteristics (Urevatn)

	Q^M $[m^3/s]$	**Q^M** $[l/(s * km^2)]$	**Q5**	**Q10**	**Q20**	**Q50**	**Q100**	**Q200**
Flood Frequency Factors	-	-	1.19	1.45	1.79	2.36	2.94	3.68
95 % interval upper limit (m^3/s)	32.4	598.2	39.2	49.2	61.7	84.1	107.4	134.5
Flood values (m^3/s)	18.3	338	21.7	26.6	32.6	43.1	53.7	67.3
95 % interval lower limit (m^3/s)	10.3	191	12.0	14.4	17.3	22.1	26.8	33.6
Floods with climate impact(m^3/s)	25.6	473.1	21.7	37.2	45.7	60.4	75.2	94.2

TABLE 3.10: Urevatn field parameters.

Field parameters	
Area (A)	54.1 km^2
Effective Sea (S_{eff})	25.1 %
River Length (E_L)	15.4 km
River gradient (E_C)	-75.1 m/km
River gradient_1085 (G_{1085})	0.6 m/km
Field length (F_L)	12.3 km
H_min	1175 moh
H_10	1175 moh
H_20	moh
H_30	1182 moh
H_40	1190 moh
H_50	1201 moh
H_60	1213 moh
H_70	1228 moh
H_80	1249 moh
H_90	1280 moh
H_max	1385 moh
Bre	0.0 %
Cropland	0.0 %
Swamp	0.0 %
sea	32.3 %
Forest	0.0 %
Bare mountain	67.7 %
Urban	0.0 %

3.5 Existing System

Otra Kraft Hydropower System is a conventional Hydropower System that only enables the generation of electricity. The system consists of three power generation units as shown in Figure 3.6 i.e., Brokke, Holen 1-2, and Holen 3.

The average annual production of Brokke power station is 1416 GWh and has the nameplate capacity of 330 MW. Whereas, Holen power unit consists of two power generation unit Holen 1-2 and Holen 3. The average annual production of Holen power station is 831 GWh and has a maximum capacity of 390 MW (NVE, 2013).

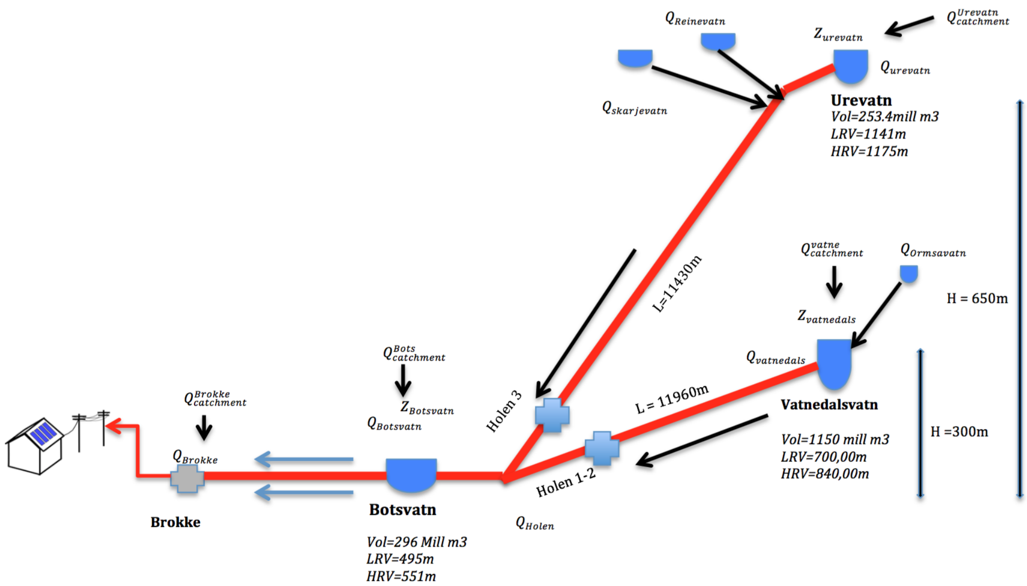

FIGURE 3.6: Figure representing the actual Otra Hydropower system

3.5.1 Disadvantages of Existing System

Since Hydropower is a renewable source of energy but it has some drawbacks too. The foremost drawback of Otra Kraft Hydropower system is the excess inflow from catchment during the summertime. Due to the limited capacity of Botsvatn reservoir, a large volume of catchment forces Botsvatn reservoir to release water. As a result, the Brokke generation unit force to operate regardless of power demand.

3.6 Need for new system

Since, with high spot price and inadequate reservoir levels, it is advantageous to reverse the process and transform electrical energy to potential energy through pumping system so that the water can be later used when power demand is high to earn more profit and to prevent flooding.

3.6.1 Critical period

Figure 3.7 exhibits the inflows to the reservoirs. We can examine during the midst of May to June 2017 (defined as a critical period in this thesis) the volume of water from catchment to the Botsvatn reservoir suddenly rises which develops the risk of flooding or overflow. This situation makes Brokke turbine unit to operate to the maximum capacity regardless of power demand. Moreover, during this situation, Holen 1-2 and Holen 3 generation unit gets stopped to prevent the water to stream into Botsvatn reservoir. Therefore, a pumped storage system is a necessity for this kind of hydropower systems.

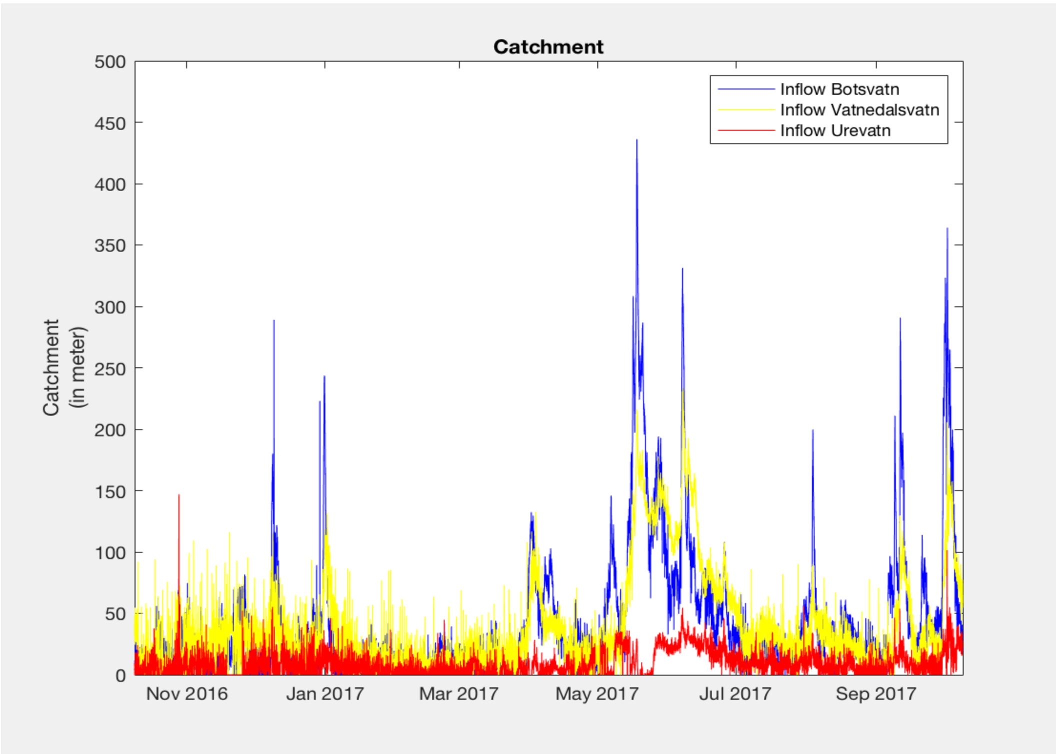

FIGURE 3.7: Price and reservoir level overview during critical period

3.6.2 Reservoirs level during critical period

Figure 3.8 shows the level of Botsvatn and Urevatn reservoir during the critical period. We can observe in the figure, throughout the critical period the level at Urevatn reservoir is well below the HRV (1175m). Hence, it is feasible to use those instances for pumping operation.

Additionally, gray patches in figure 3.8 represent the period when the electricity price is below its average value and this period can be used for pumping operation for economic benefit.

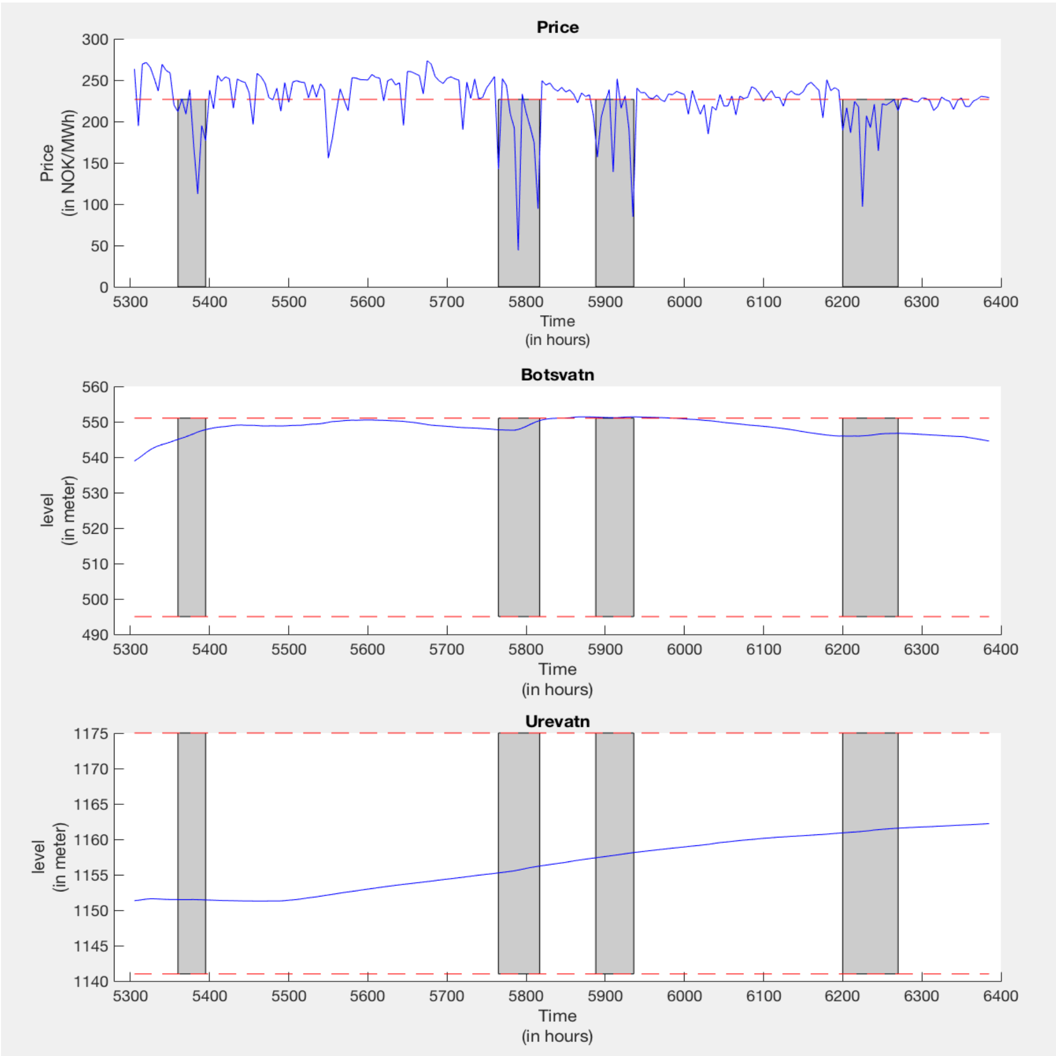

FIGURE 3.8: Price and reservoir level overview during critical period

3.6.3 Advantages of Proposed System

Pumped-storage hydropower system potential account for 2.5 % of worldwide installed capacity. It is the most reasonable and genuine method of storing electricity, allowing both the effective use of surplus energy and the delivering of a requisite amount of energy back on the grid.

As the pumped-storage system uses surplus electricity generated while there is a low demand for electricity performance averages out at approximately 70 % or somewhat higher, signifying that for every 10 kWh applied for storing, 7 kWh produced during generation. This medium causes these plants famous for grid management because they can be set into service in a pretty short period and undertake the grid's load fluctuations, are considered reliable and are not influenced from outside, as they often operate in closed cycles (enel, 2014).

3.7 Problem description

Figure 3.9 shows the proposed Otra kraft pumped storage hydropower system. The system consists of three separate turbine generation unit, i.e., Brokke, Holen 1-2 and Holen 3. According to the requirement, the pump is to be installed in Botsvatn reservoir, together with Holen-3 turbine unit in the same tunnel, so that Holen-3 turbine and pumping unit cannot operate at the same time, therefore, need to establish ON/OFF condition in the optimization problem.

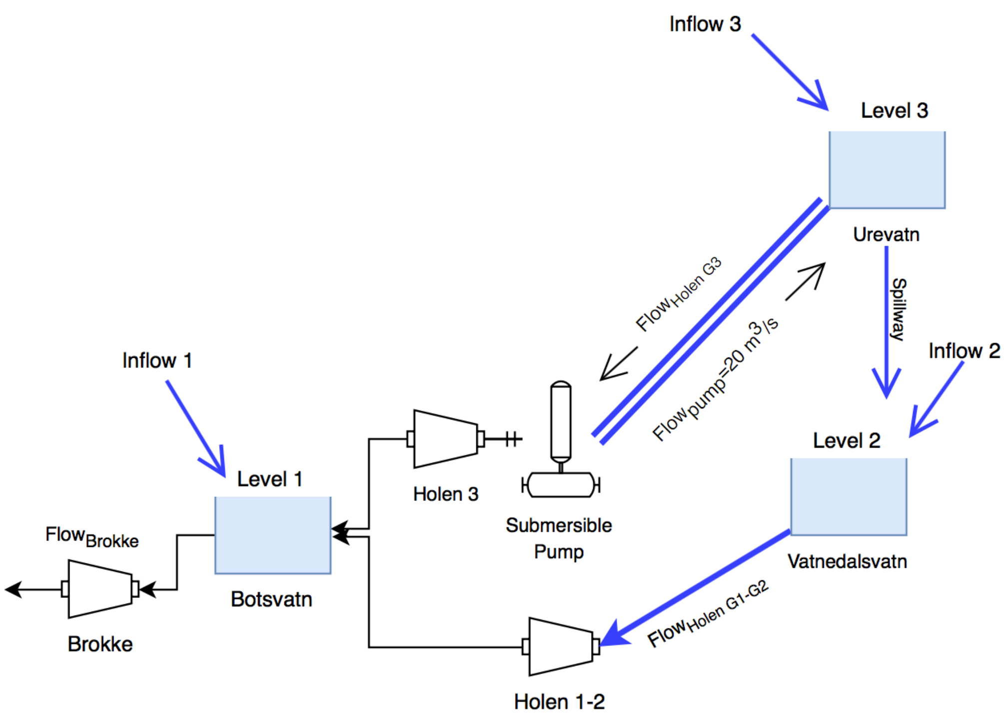

FIGURE 3.9: Figure representing pumped-storage hydropower system (proposed)

Table 3.11 represents the variable description which is used for optimization problem execution for the simplicity.

Assigned variable	**Description**
Level 1	Bostsvatn reservoir level
Level 2	Vatnedalsvatn reservoir level
Level 3	Urevatn reservoir level
Inflow 1	Inflow into Botsvatn reservoir from catchment and other connected reservoir outside the system
Inflow 2	Inflow into Vatnedalsvatn reservoir from catchment and other connected reservoir outside the system, i.e. $Q_{skarjevatn}$ + $Q_{Reinevatn}$ + $Q_{Catchment}$
Inflow 3	Inflow into Urevatn reservoir from catchment and other connected reservoir outside the system, i.e. $Q_{Ormsavatn}$ + $Q_{Catchment}$
volume 1	Botsvatn reservoir volume
volume 2	Vatnedalsvatn reservoir volume
volume 3	Urevatn reservoir volume
$volume1_{spill}$	Botsvatn spillway volume
$volume2_{spill}$	Vatnedalsvatn spillway volume
$volume3_{spill}$	Urevatn spillway volume

TABLE 3.11: Description of assigned variable from figure 3.9

Chapter 4

Data Composition

4.1 Introduction

Based on the data provided by the Otra kraft DA concerning reservoir level, volume percentage, and power, necessary computations have been made to perform the optimization. Reservoir level causing flow through each turbine during a year (Oct 2016 - Oct 2017), the flow rate through reservoir and power production from each generation unit are received. From these data, catchments (inflow) and volume corresponding to each reservoir are calculated. According to the data available for flow and level the minimum value is recorded as a lower bound constraint. Similarly, the maximum value recorded as an upper bound of the variable.

4.2 Reservoir Level

As stated earlier the optimization has been done by using 3-distinct reservoirs and the hourly reservoir levels data which is provided by the company. Furthermore, by adopting the following data the minimum and maximum level of each reservoir are noticed to use it as bound constraints during optimization. Further, using the complete data for evaluating the connection between the objective function, level and flow of reservoirs.

4.2.1 Botsvatn

Figure 4.1 displays the level variation at Botsvatn. Through the hourly data, we can recognize the minimum and maximum level of the reservoir, i.e. 529.62 m and 551.38 m. According to the information accessible at NVE Atlas, 2018, the highest regulated water level (HRV) at Botsvatn reservoir is 551 m.o.h and lowest regulated water level (LRV) is 495 m.o.h. Hence by adopting this, we can bound the limit of reservoir levels.

It can be seen in figure 4.1 that the actual level in summer exceeds the upper limit which is defined as **"Critical period"** in this thesis. Due to this condition, the Botsvatn reservoir forced to release water regardless of the power demand.

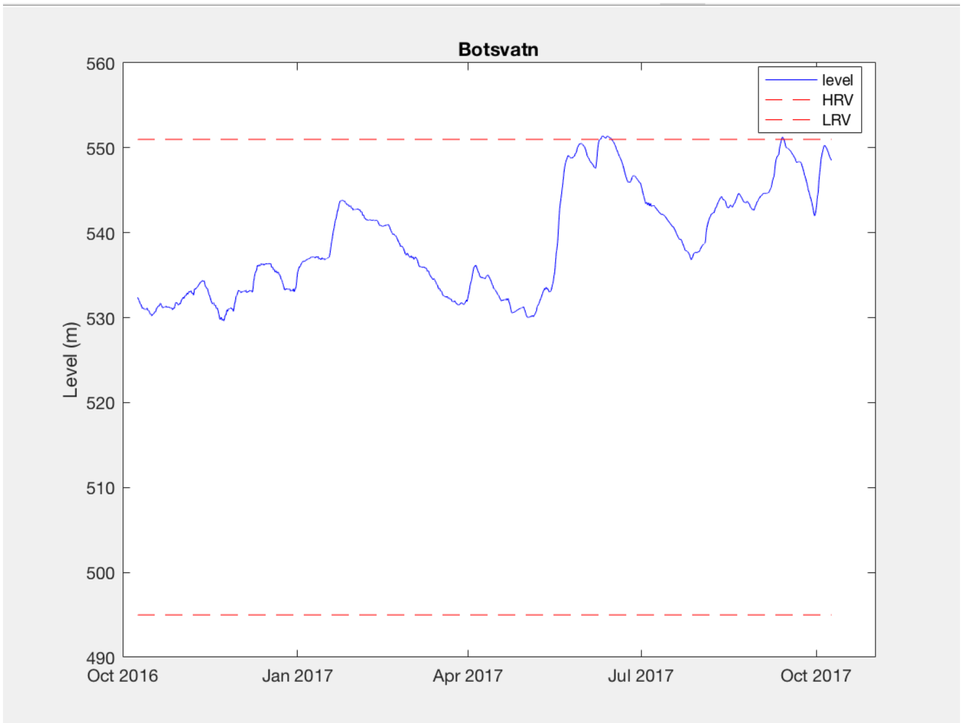

FIGURE 4.1: Botsvatn level variation

4.2.2 Vatnedalsvatn

Figure 4.2 gives the variation in Vatnedalsvatn reservoir. Moreover, we can testify the minimum and maximum level of the reservoir, i.e. 785.38 m and 836.12 m which is between LRV and HRV, i.e. 700 m.o.h and 840 m.o.h.

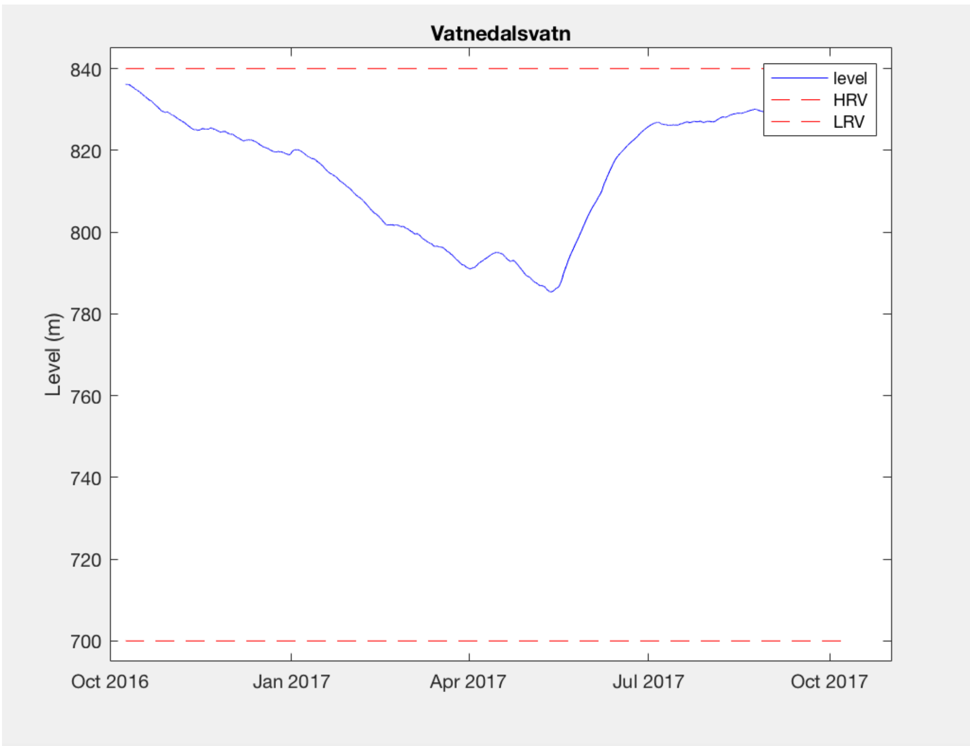

FIGURE 4.2: Vatnedalsvatn level variation

4.2.3 Urevatn

Similarly, Figure 4.3 shows the variation in Urevatn reservoir. The minimum and maximum value for level are 1151 m and 1173.6 m which is between the LRV and HRV limit of the reservoir i.e. 1141 m.o.h and 1175 m.o.h.

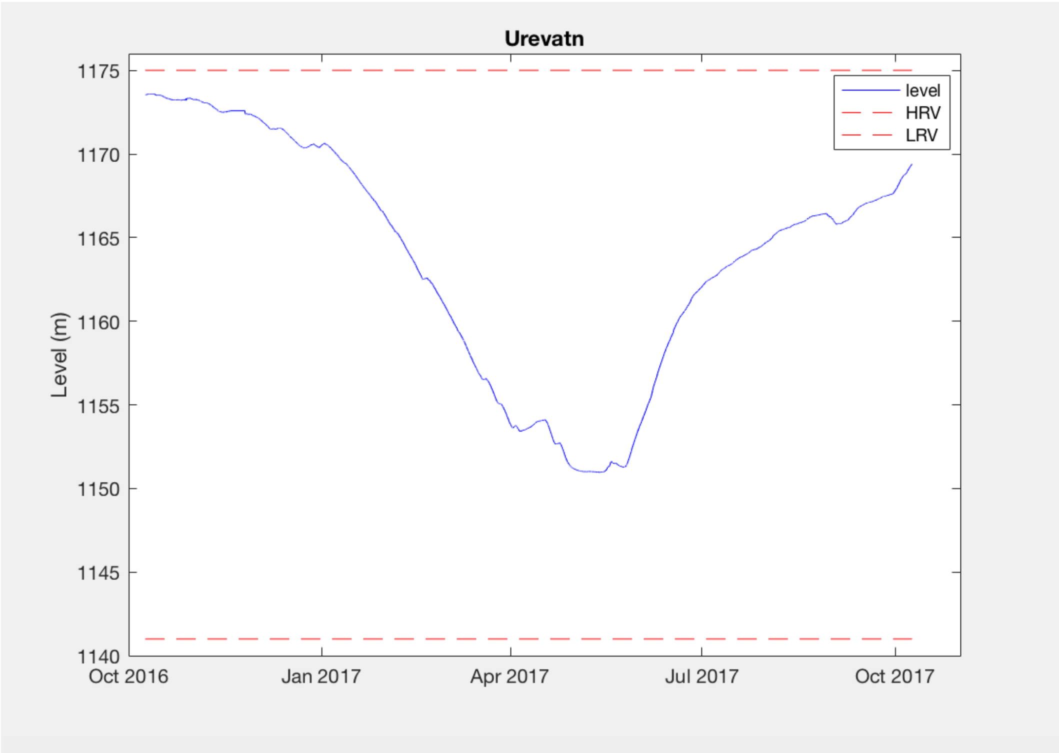

FIGURE 4.3: Urevatn level variation

The level of Urevatn reservoir is considerably beneath the upper limit throughout the critical period.

4.3 Computations

According to data available concerning power from generation units, volume percentage and the level of the reservoirs we can estimate the reservoir curve by using polynomial fit. In MATLAB, we can use 'fit' command on the level, power and volume percentage data specifying that we want a second order polynomial to display the line (bestfit) approximating the data. Further, we can evaluate some point on a line from the polynomial coefficients from the previous command and calculate values at the same values of level, volume, and power as original data.

4.3.1 Reservoirs curve

The governor of the water flow depends on the amount of electrical power the generator is compelled to generate. The correlation within flow rate, level, and power generated is essential for power generation management to follow.

The curve and computations presented underneath are done for hourly data in Sharma, 2018 and revised for daily average and weekly average data. The curves and computations represent the daily average data. The computations and plots for weekly average and hourly data are inserted in the Appendix B.

Botsvatn Level

Figure 4.4 represents the best line that goes through the data points (i.e. volume percentage and level) also defined as 'bestfit' and signifies the relation between Botsvatn reservoir level and volume percentage as stated in the following second-order polynomial Equation 4.1.

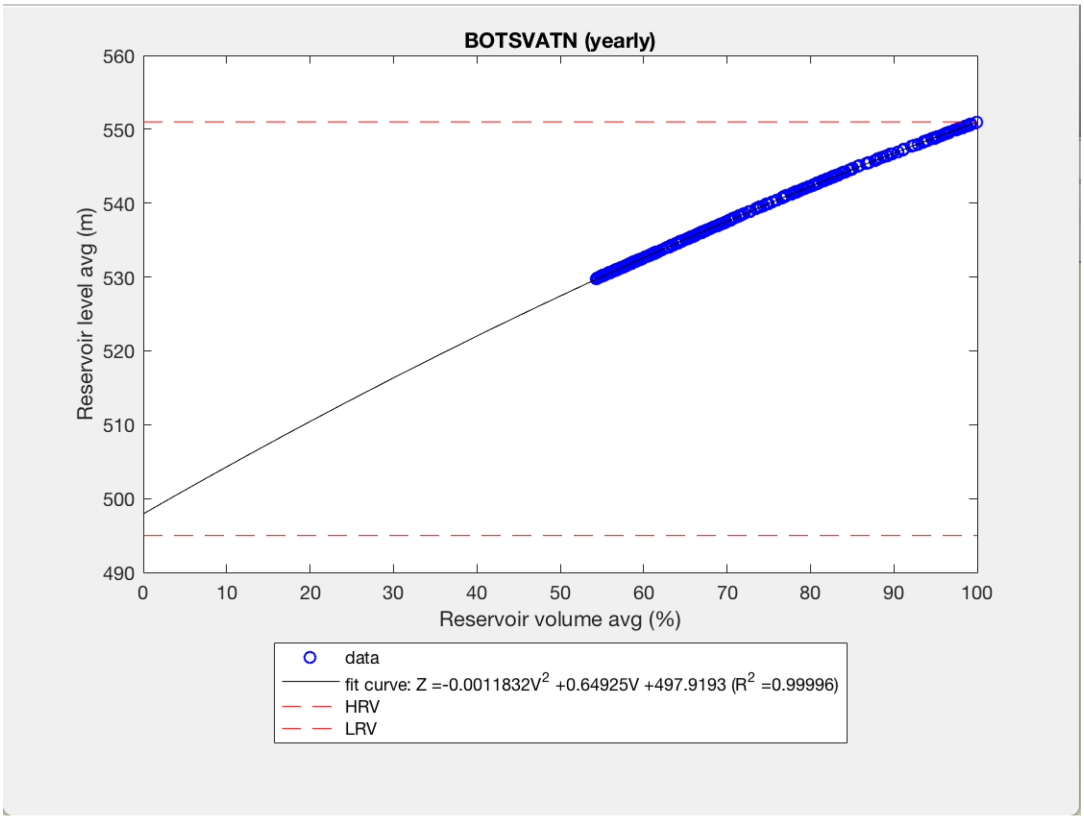

FIGURE 4.4: Botsvatn: Level vs Volume %

$$z_{R1_{i+1}} = -0.0011832(\frac{Volume_{R1_{i+1}}}{296 * 10^6})^2 + 0.64925(\frac{Volume_{R1_{i+1}}}{296 * 10^6}) + 497.9193 \quad (R^2 = 1.0000) \tag{4.1}$$

where, z_{R1} is the level (m) and $Volume_{R1}$ is the volume of Botsvatn reservoir (m^3).

Further, R^2 is a statistical measure which expresses how successful the fit is in demonstrating the variation of the data or R^2 is the square of the relationship within the response values and the divined response values. It is also defined as the square of the various correlation coefficient and the coefficient of multiple measurements.

R^2 varies between 0-1 as the range demonstrates the proportion of variance demonstrated by fit for example, with a value closer to 1 designates a better fit. Therefore, in our case R^2 value is 1 which indicates that the fit describes 100% of the total variation in the data.

From equation 4.1 we can calculate the level of the reservoir at each instance. However, the volume of the reservoir can be calculated from volume percentage and total volume of the reservoir as expressed in equation 4.2.

$$Volume_{Botsvatn,i} = \frac{Volume\%_{Botsvatn,i} * Volume^{total}_{Botsvatn}}{100} \tag{4.2}$$

Vatnedalsvatn Level

Figure 4.5 represents the 'bestfit' curve and illustrates the relation between Vatnedalsvatn reservoir level and volume percentage as stated in the following second-order polynomial Equation 4.3.

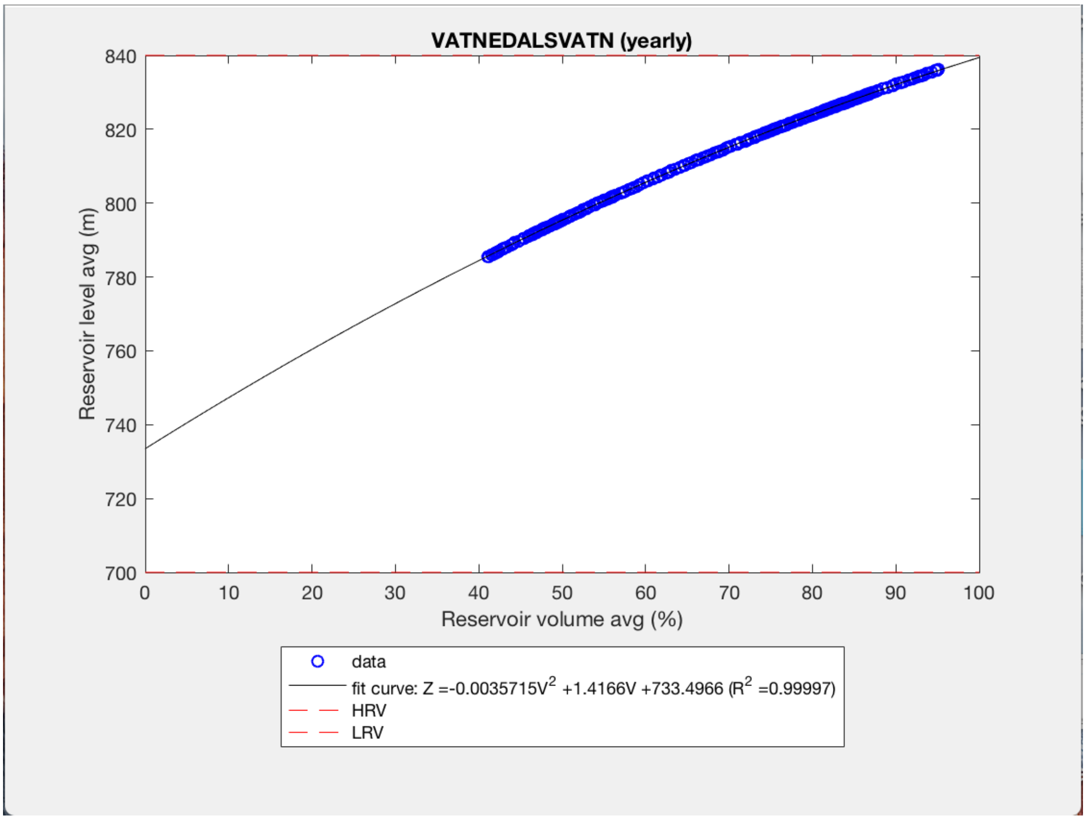

FIGURE 4.5: Vatnedalsvatn: Level vs Volume %

$$z_{R2_{i+1}} = -0.0035715(\frac{Volume_{R2_{i+1}}}{1150 * 10^6})^2 + 1.4166(\frac{Volume_{R2_{i+1}}}{1150 * 10^6}) + 733.4966 \quad (R^2 = 1.0000) \tag{4.3}$$

where, z_{R2} is the level (m) and $Volume_{R2}$ is the volume of Vatnedalsvatn reservoir (m^3).

From equation 4.3 we can calculate the level of the reservoir at each instance. However, the volume of the reservoir can be calculated from volume percentage and total volume of the reservoir as expressed in equation 4.4.

$$Volume_{Vatnedalsvatn,i} = \frac{Volume\%_{Vatnedalsvatn,i} * Volume^{total}_{Vatnedalsvatn}}{100} \tag{4.4}$$

Urevatn Level

Figure 4.6 represents how Urevatn reservoir volume changes with change in level within LRV and HRV. The 'bestfit' curve illustrates the relation between Urevatn reservoir level and volume percentage as stated in the following second-order polynomial Equation 4.5.

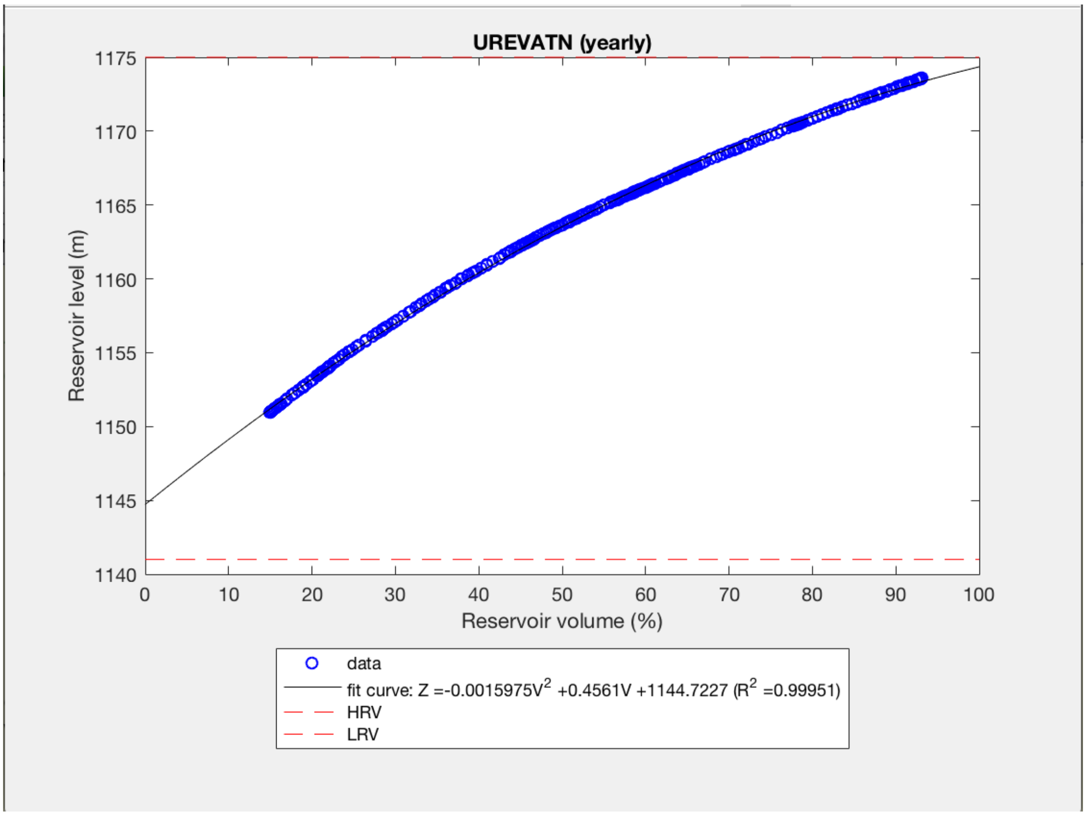

FIGURE 4.6: Urevatn: Level vs Volume %

$$z_{R3_{i+1}} = -0.0015975(\frac{Volume_{R3_{i+1}}}{253.4 * 10^6})^2 + 0.4561(\frac{Volume_{R3_{i+1}}}{253.4 * 10^6}) + 1144.7227 \quad (R^2 = 0.9995) \tag{4.5}$$

where, z_{R3} is the level (m) and $Volume_{R3}$ is the volume of Urevatn reservoir (m^3).

From equation 4.5 we can calculate the level of the reservoir at each instance. However, the volume of the reservoir can be calculated from volume percentage and total volume of the reservoir as expressed in equation 4.6.

$$Volume_{Urevatn,i} = \frac{Volume\%_{Urevatn,i} * Volume^{total}_{Urevatn}}{100} \tag{4.6}$$

4.3.2 Power Curves

To determine the power in hydropower setup equation 4.7 can be used, and it gives the dependence of power on flow rate and head loss where the head loss is equivalent to the difference between level of the upper and lower reservoir.

$$P_{HP} = \rho.H_{net}.g.Q \tag{4.7}$$

Where, H_{net} is net-head of reservoir (m), and Q is flow rate (m^3/s).

According to the availability of Power data from the actual hydropower station, the approach is to fit the level curve and flow data by using 2^{nd} order polynomial fit to get the expression for distinct power units which confers the dependence of power on flow rate and level in corresponding to the real data.

Brokke

Figure 4.7 illustrates the 'bestfit' curve by using real daily average flow and power data.

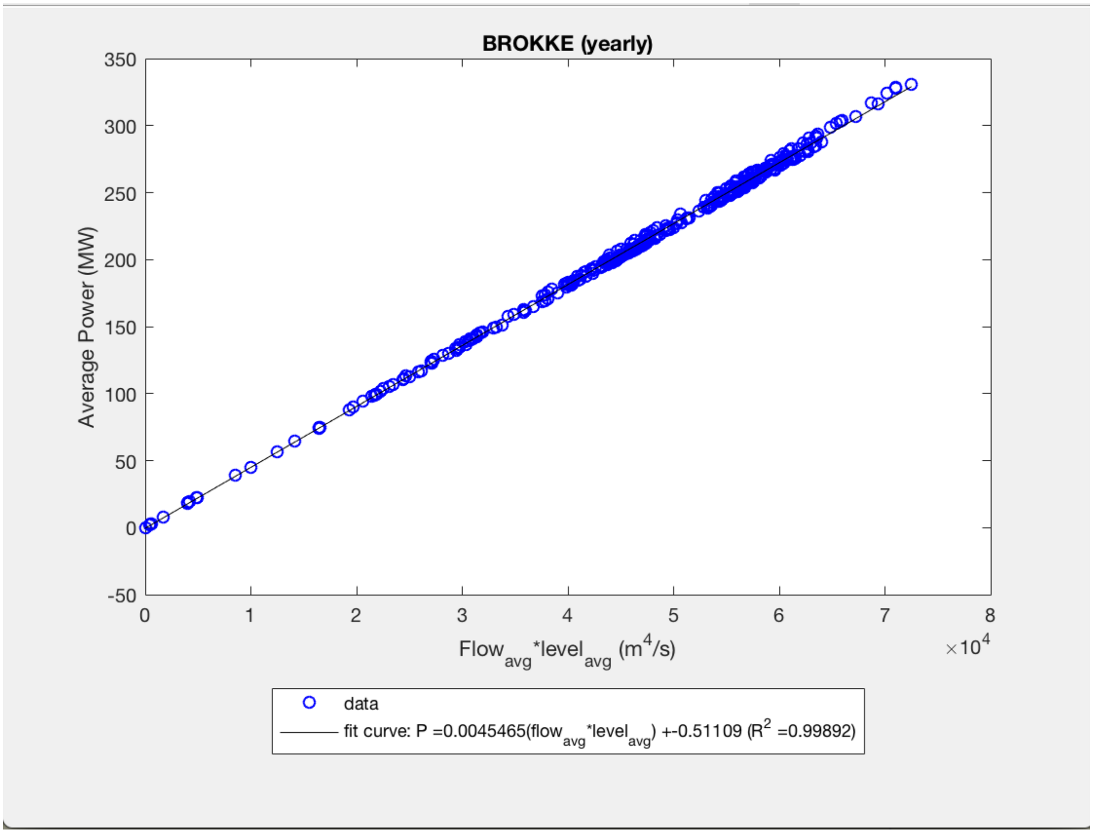

FIGURE 4.7: Power Brokke (Average)

Equation 4.8 shows the dependence of power generated from Brokke power unit on the flow-rate and level at Botsvatn (i.e. Q_{R1} and z_{R1}).

$$Power_{1,i} = 0.0045465(Q_{R1,i} * z_{R1,i}) - 0.51109 \quad (R^2 = 0.9989) \tag{4.8}$$

where, $Power_1$ is power generated from Brokke unit (MW), Q_{R1} is the flow-rate (m^3/s) ,and z_{R2} is the level at Vatnedalsvatn reservoir (m).

Holen G1-G2

Figure 4.8 shows the dependence of Power from Holen G1-G2 unit on the flow-rate and level at Vatnedalsvatn reservoir as expressed in the following polynomial equation 4.9.

$$Power_{2,i} = 0.0029478(Q_{R2,i} * z_{R2,i}) - 0.077604 \quad (R^2 = 0.9965) \tag{4.9}$$

where, $Power_2$ is power genrated from Holen 1-2 unit (MW), Q_{R2} is the flow-rate (m^3/s), and z_{R2} is the level at Vatnedalsvatn reservoir (m).

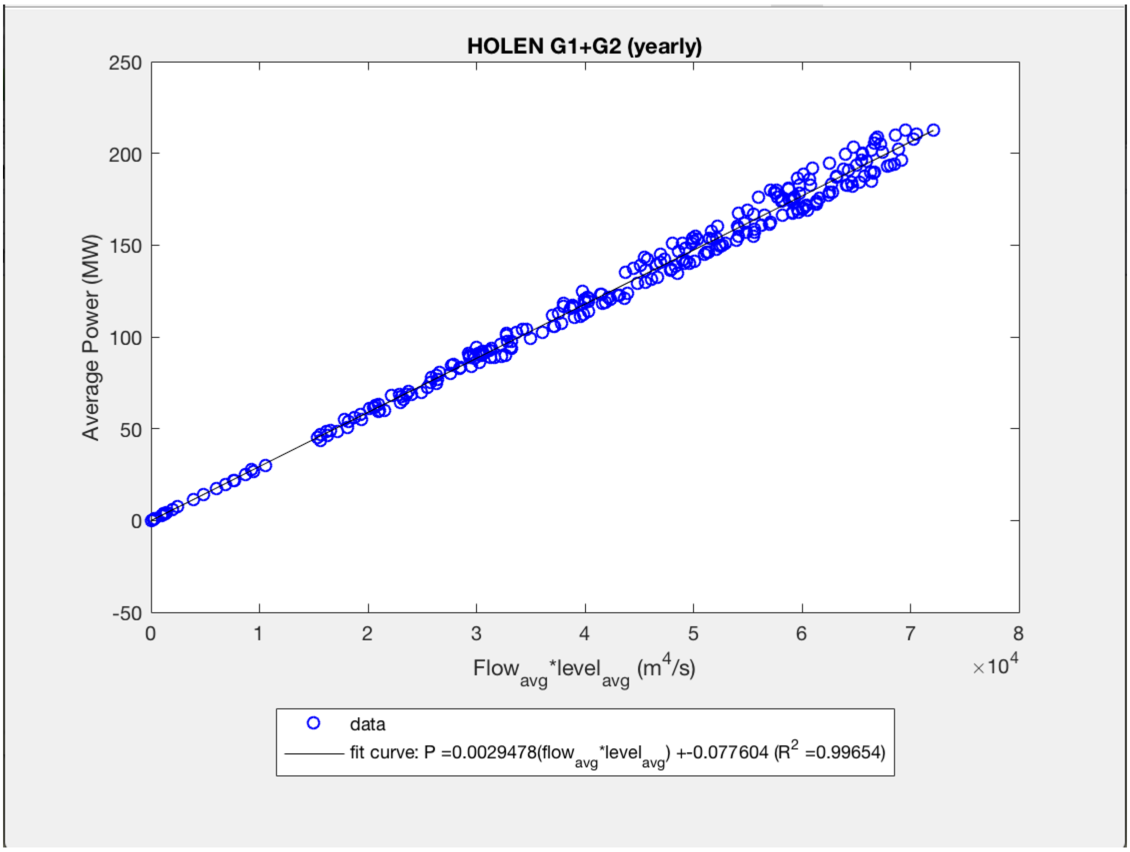

FIGURE 4.8: Power Holen G1-G2 (Average)

Holen G3

Figure 4.9 shows the dependence of Power from Holen G3 unit on the flow-rate and level at Urevatn reservoir as expressed in the following polynomial equation 4.10.

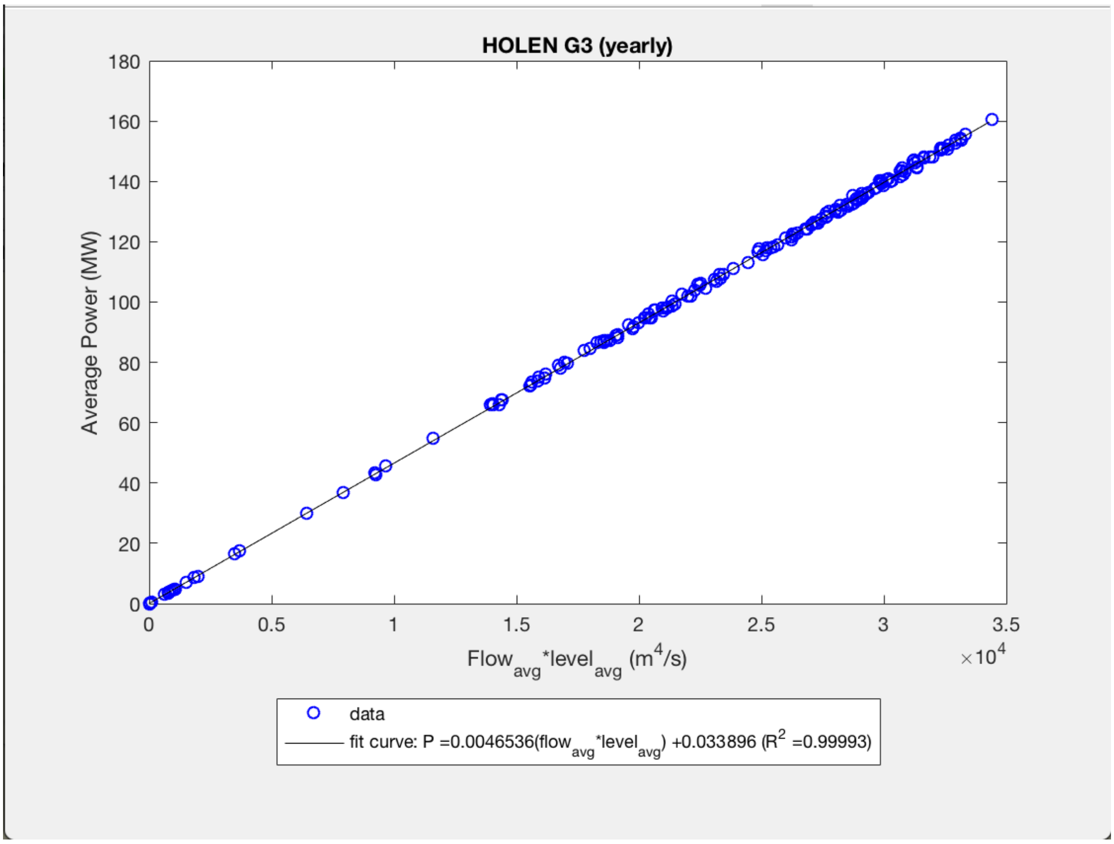

FIGURE 4.9: Holen G3 Brokke (Average)

$$Power_{3,i} = 0.0046536(Q_{R3,i} * z_{R3,i}) + 0.033896 \quad (R^2 = 0.9999) \tag{4.10}$$

where, $Power_3$ is power genrated from Holen 3 unit (MW), Q_{R3} is the flow-rate (m^3/s) and z_{R3} is the level at Urevatn reservoir (m).

4.4 Catchment

A catchment is an area where water is accumulated by natural landscapes and is environed by high features such as hills or mountains. The soil in catchment acts like a sponge, soaking up moistures, storing it in underground aquifers and gradually discharging it in rivers and streams which later flow into dams. Consequently, the level at the reservoir depends on the inflow from the catchment and the estimation of inflow from the catchment into the reservoir can be done by mass balance. By using the volume percentage data and the total volume for the different reservoirs the actual volume of the reservoir is calculated so as to determine the volumetric flow rate for each reservoir.

4.4.1 Inflow: Botsvatn

Figure 4.10 shows the flow through catchment into the Botsvatn reservoir. It can be observed around 5100 - 5700 hour or during the month of June - August the inflow from catchment is at its peak.

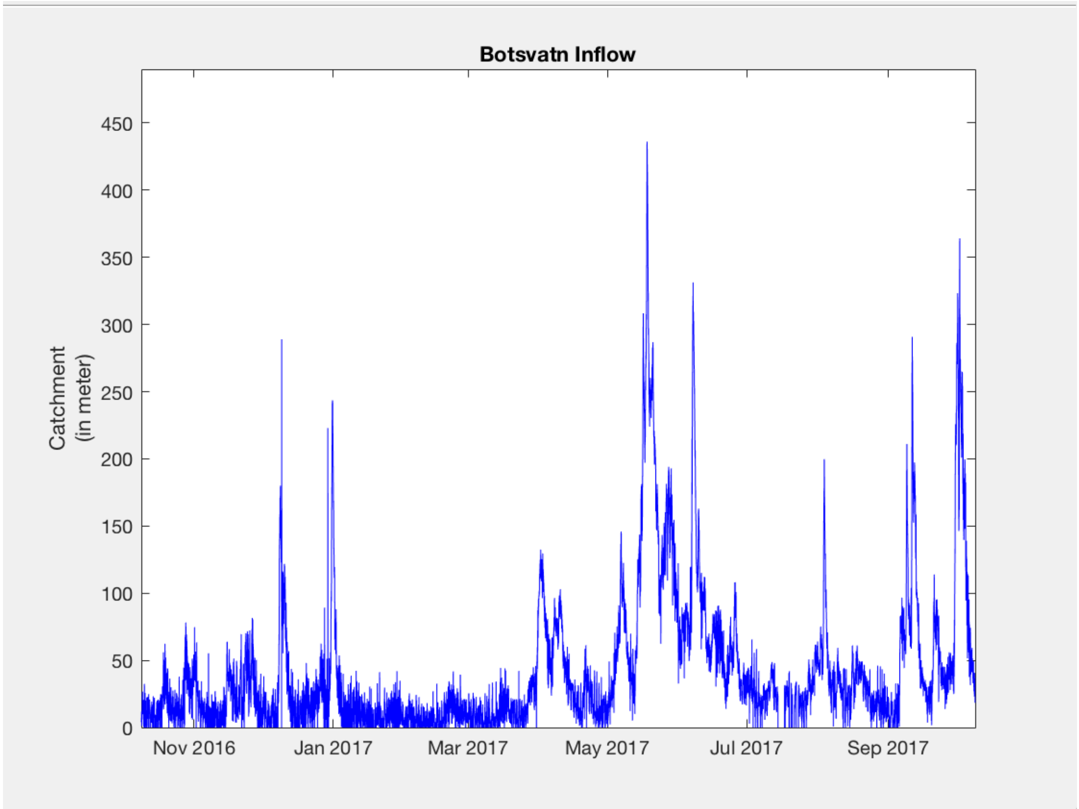

FIGURE 4.10: Inflow Botsvatn Reservoir

Taking into account the flow rate from Holen G1-G2, Holen G3 and Brokke turbine unit the mass balance has been performed as expressed in the equation 4.11.

$$Flow_{catchment,R1} = \frac{volume_{R1}(i+1) - volume_{R1}(i)}{t(i+1) - t(i)} - flow_{G1-G2}(i) - flow_{G3}(i) + flow_{Brokke}(i) \tag{4.11}$$

where, $volume_{R1}$ is the volume of Botsvatn reservoir at particular instant (m^3), t is time (sec), $flow_{G1-G2}$ is flow rate at Holen G1-G2 (m^3/s), $flow_{G3}$ is flow rate at Holen G3 (m^3/s), and $flow_{Brokke}$ is flow rate at Brokke turbine unit (m^3/s).

4.4.2 Inflow: Vatnedalsvatn

Figure 4.11 shows the inflow from the catchment to Vatnedalsvatn reservoir throughout the time interval.

We can observe the rise in inflow from May to June 2017 (critical period) as it behaves similar to the inflow into Botsvatn reservoir. However, the capacity of Vatnedalsvatn reservoir is comparatively larger than Botsvatn reservoir. Hence the risk of overflow is less.

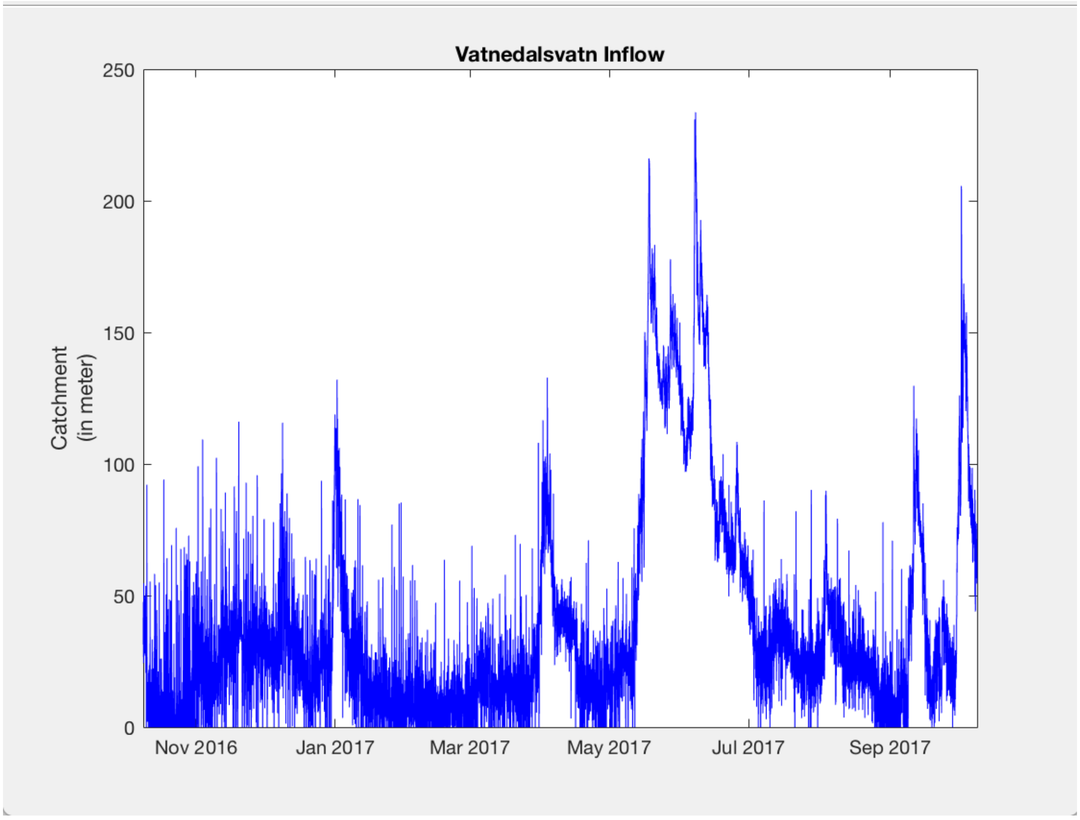

FIGURE 4.11: Inflow Vatnedalsvatn Reservoir

Taking into account the flow rate from Holen G1-G2 turbine unit the mass balance has been performed as expressed in the equation 4.12.

$$Flow_{catchment,R2} = \frac{volume_{R2}(i+1) - volume_{R2}(i)}{t(i+1) - t(i)} + flow_{G1-G2}(i) \tag{4.12}$$

where, $volume_{R2}$ is the volume of Vatnedalsvatn reservoir at particular instant (m^3), t is time (sec), and $flow_{G1-G2}$ is flow rate at Holen G1-G2 (m^3/s).

4.4.3 Inflow: Urevatn

Figure 4.12 shows the inflow from the catchment to Urevatn reservoir through out the time interval.

Unlike other two reservoirs, the inflow into the Urevatn reservoir during May and June is less. Hence, the condition of overflow cannot arise during critical period. However, the pump system will put a substantial impact on the inflow into the Urevatn reservoir throughout the pumping duration.

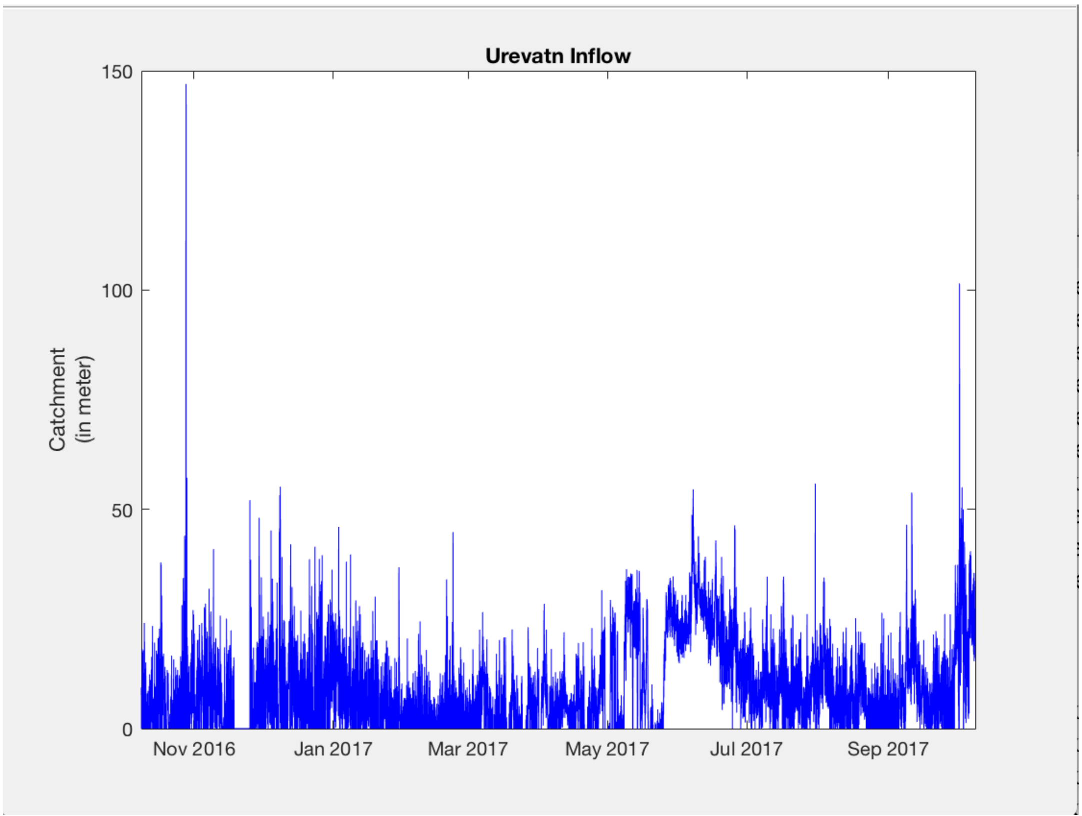

FIGURE 4.12: Inflow Urevatn Reservoir

Taking into account the flow rate from Holen G3 turbine unit the mass balance has been performed as expressed in the equation 4.13.

$$Flow_{catchment,R3} = \frac{volume_{R3}(i+1) - volume_{R3}(i)}{t(i+1) - t(i)} + flow_{G3}(i) \tag{4.13}$$

where, $volume_{R3}$ is the volume of Urevatn reservoir at particular instant (m^3), t is time (sec), and $flow_{G3}$ is flow rate at Holen G3 (m^3/s).

4.5 Price Data

The price data for Kristiansand region (NO2) imported from NORDPOOL, 2018. Price data for Kristiansand region is uploaded corresponding to the data available for flow and power i.e. Oct 2016 - Oct 2017. The following figure 4.13 depicts the variation in yearly price.

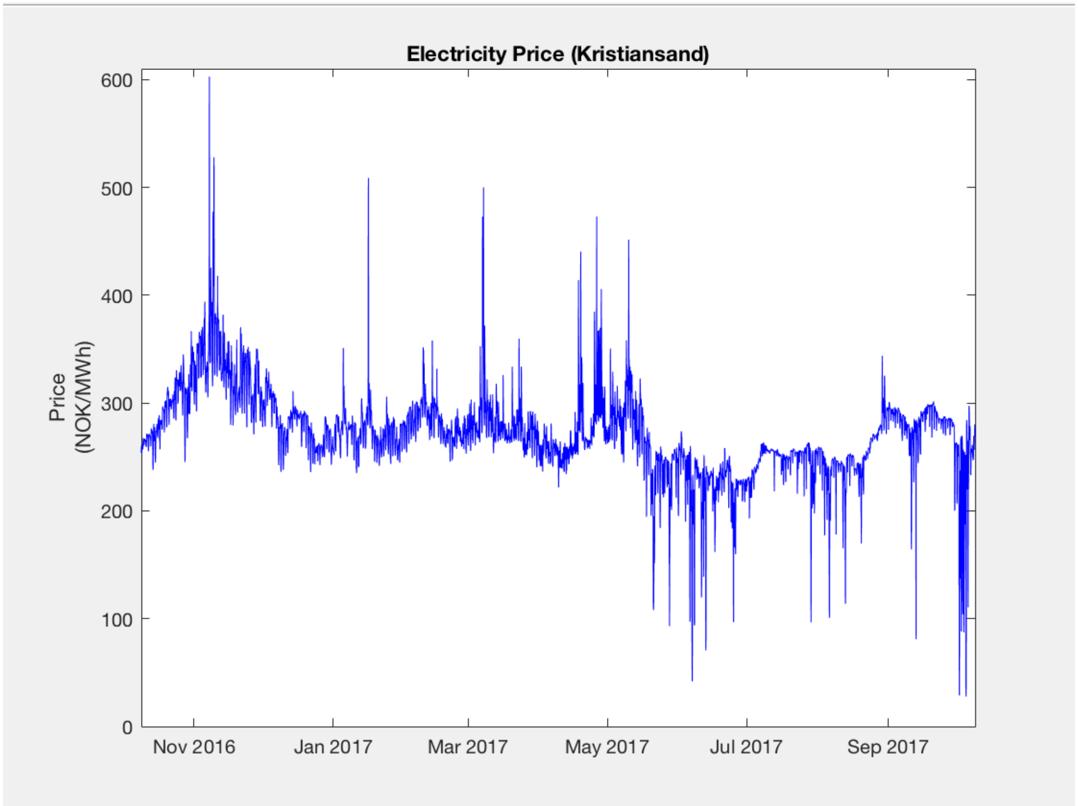

FIGURE 4.13: Price variation - Kristiansand region

It can be observed, during critical period the price is comparatively low. Hence it can be assumed that this duration is feasible for pump operation planning.

Chapter 5

Optimization using Evolutionary Algorithm

5.1 Algorithm Selection

Due to the applications to related problems, performance reports, and various practical studies, the adoption of algorithms for this research is made. Possibly, plenty of distinct evolutionary algorithms could be implemented to hydropower optimization problems. Adoption of evolutionary algorithms needed some proficiency and knowledge. One determinant which considered heavily in the selection method was the extent and breadth of previous applications. The high performance of an appropriate algorithm and the number of examples where it has been used to an appropriate degree of optimization problem gives some indication of the algorithm's ability and potential for application in other operations. Based on the multiphase research study, previous application related to constrained optimization and narrowing the options to continuous real-valued algorithms Differential Evolution (DE) method is selected. It supports the principle of natural evolution and the remainder of fittest that is defined by the Darwinian Theory (Prewitt, 2001).
The preeminent applications of differential algorithm are in optimization. However, they have also been applied to administer data mining, create learning systems, etc.

Why Differential Evolution?

Evolutionary algorithms vary from conventional optimization methods as they usually evolve a population of solutions in the search location of decision variables, somewhat of rousing from a single point. In every iteration, differential evolution produces new solutions which are defined as offspring further carries a competitive selection to obtain small solutions.
In contrast with conventional optimization methods and optimizing many real-world problems, such as calculus-based nonlinear programming methods, differential evolution is more robust and deliver a reliable offset within the exploration and exploitation in the search space.

Furthermore, differential evolution has an advantage compared to evolution strategy or evolutionary programming. Like in Gaussian mutation, we have to specify sigma value. However, in this, we do not have to tune the step value the difference will do the work. In the initial stages, the population member will be well separated than the difference vector will have a reasonable large magnitude because the population member is well separated. When the evolutionary algorithm involved over a generation, the population member tends to move to a right region or optimal region hence when the member moves towards the vast region they become closer to each other which means the magnitude of the difference vector will become smaller. So the Differential evolution is naturally altering the search behavior from exploration to exploitation by this difference operation here. So specifying 'F' (DE-stepsize ranges from (0, 2)) is a lot easier than specifying sigma value in the Gaussian mutation (Suganthan, 2018).

5.2 Methodology

Due to the high volume of available data, the outlining is to conduct a categorized optimization. Firstly, the plan is to execute weekly Optimization based on hourly data from the Otra Kraft Hydropower system, and the data is classified to its weekly average value to reduce the size of data and to examine the optimization process.

Following the first operation, the Optimization is carried for daily average value throughout the year to determine the optimize value. Finally, the Optimization is conducted for one year, i.e. from 08 Oct 2016 - 07 Oct 2017 in three steps.

- Weekly average Optimization is conducted for 52 weeks, i.e. from 08 Oct 2016 - 07 Oct 2017. To check the algorithm performance and to reduce the volume of data and to check the effect of constraints (presented in Appendix C).
- Daily average Optimization is conducted for 365 days and corresponding results are presented.
- Hourly Optimization is executed for different weeks throughout the year and compared with the daily average optimization results to check the efficiency of the proposed algorithm.

5.2.1 Differential Evolution

Differential evolution is a population-based, derivative-free function optimizer. This is one of the main properties that makes it much more robust than traditional derivative based methods, where irregular objective functions can lead to a locally optimal solution or in worst cases to unstable behavior (not reaching an appropriate solution). It usually encodes decision variables as floating point numbers and handles them with simple arithmetic operations such as addition, subtraction, and multiplication. It is an evolutionary algorithm and follows similar steps like other algorithms. In our problem, we have **'D'** number of decision variables, and each one have lower and upper bounds. The first step is to set the population size **NP**. The steps are taken to solve the problem shown in Figure 5.1.

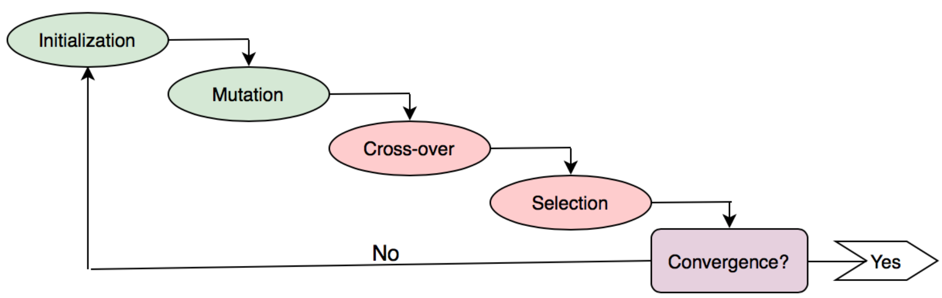

FIGURE 5.1: Steps involves in Differential Evolution Algorithm

5.2.2 Initialization

Initialization can be performed 'randomly' as we have a 2-dimensional matrix and every element in the matrix can be initialized 'randomly' which means every point in a 2-dimensional matrix ('D' number of columns and 'NP' number of rows) is initialized randomly. So we generate random numbers every time between minimum and maximum value for each variable.

$$X_{1,i,0} = \begin{bmatrix} x_{1,i,0}, & x_{2,i,0}, & x_{3,i,0}, & \cdots & x_{D,i,0} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ x_{1,NP,0} & x_{2,NP,0} & x_{3,NP,0} & \cdot & x_{D,NP,0} \end{bmatrix}$$

Where, NP is population size, and D is the dimension of the problem.

$$x_j^{low} \leq x_{j,i,0} \leq x_j^{up}$$

Where, j = 1,2,3...D (Any variable), x_j^{low} = lower limit of j^{th} vector component and x_j^{up} = upper limit of j^{th} vector component.

$$x_{j,i,0} = x_j^{low} + rand_{i,j}[0,1] * (x_j^{up} - x_j^{low}) \tag{5.1}$$

5.2.3 Mutation

To perform mutation for each vector or each member of the population, we usually select three other solutions from the population member randomly. For each generation 'g' the operation creates mutation vector '$v_{i,g}$' based on the current parent population. The most commonly used mutation equation is:

$$v_{i,g} = X_{r0,g} + F_i(X_{r1,g} - X_{r2,g}) \tag{5.2}$$

where, r0,r1,r2 = Different integers uniformly chosen from the set [1,2....,NP], $X_{r1,g}$-$X_{r2,g}$ = Difference vector to mutate the parent, F_i = mutation factor or scale factor ranges from interval [0,1]. The index **'i'** will run from 1 to NP.

5.2.4 Crossover or Recombination

After completion of mutation, we have to perform a crossover operation which performs an information switch between distinct members in the current population and is also known as a 'Binomial Crossover.' This function gives the final trial vector or offspring vector '$u_{i,g}$'. This operation is performed between the i^{th} population member and the mutant vector that is generated earlier in the mutation section. In the syntax, we have to specify the crossover rate (0 to 1). Equation 5.3 shows if the random value is less than the crossover rate than we get the decision variable from the mutant variable. Otherwise, the element is taken from the parent. So we take individual elements either from the parent vector or mutant vector. Once we do the crossover between the mutant vector and parent we will get the offspring vector, i.e. '$u_{i,g}$.'

$$u_{j,i,g} = \begin{cases} v_{j,i,g}, & if\ rand_j(0,1) \leq Cr_i\ or\ j = j_{rand} \\ x_{j,i,g}, & \text{otherwise} \end{cases} \tag{5.3}$$

where, $rand_j(a, b)$= uniform random number on the interval (a,b) and newly generated for each j, j_{rand} = Integer that is randomly chosen from 1....D and CR_i is the crossover probability (0,1).

5.2.5 Selection

Now the computation will be between the offspring vector and the parent vector. If the objective value of offspring is smaller or equal, we will select the offspring. If the objective value of offspring is higher, we will keep the parent (if the objective is to minimize). This operation selects the better one from the parent vector i.e $X_{i,g}$ and the trial vector $u_{i,g}$ according to fitness value (f(.)). For example, if we have a minimization problem, then the selected vector is given by:

$$X_{i,g+1} = \begin{cases} u_{i,g}, & if\ f(u_{i,g}) \leq X_{i,g} \\ X_{i,g}, & \text{otherwise} \end{cases} \tag{5.4}$$

5.3 Model Implementation

The main objective of this study is to minimize the objective function expressed below:

$$min \sum_{t=i} (Power_{4,i} * price_{pump,i} - (Power_{1,i} + Power_{2,i} + Power_{3,i}) * Price_i) \tag{5.5}$$

where, $Power_4$ is pump power, $Power_1$ is power from Brokke generation unit, $Power_2$ is power from Holen 1-2 generation unit, $Power_3$ is power from Holen-3 generation unit, $Price$ represents the electricity price in Kristiansand (NO2) region and $price_{pump}$ denotes the price of electricity for pumping (assumed 20% less than the electricity price for Kristiansand), so that the algorithm operates the pump when the electricity price is well below the average price.
The overflow from the reservoir for each iteration 'i' transpires while the volume of the reservoir exceeds its maximum value or upper bound constraint. The excess volume will be considered as spillway volume, and that can be used in the mass balance continuity equation. The catchment area influenced by reservoir spillway approximated from figures 3.5, 3.3 and 3.4 and implemented in equations 5.12, 5.13 and 5.14. Equations 5.6, 5.7, 5.8, 5.9, 5.10, and 5.11 is used to estimate the value of reservoirs spillway.

If, $Volume1_{spill} \leq$ Total volume of Botsvatn reservoir, then:

$$Volume1_{spill} = 0 \tag{5.6}$$

If, $Volume1_{spill} >$ Total volume of Botsvatn reservoir, then:

$$Volume1_{spill} = Volume1_{i+1} - Total^{Botsvatn}_{volume} \tag{5.7}$$

Where, $Volume1_{spill}$ represents the volume of Botsvatn reservoir spillway.

If, $Volume2_{spill} \leq$ Total volume of Vatnedalsvatn reservoir, then:

$$Volume1_{spill} = 0 \tag{5.8}$$

If, $Volume2_{spill}$ > Total volume of Vatnedalsvatn reservoir, then:

$$Volume2_{spill} = Volume2_{i+1} - Total^{Vatnedalsvatn}_{volume} \tag{5.9}$$

Where, $Volume2_{spill}$ represents the volume of Vatnedalsvatn reservoir spillway.

If, $Volume3_{spill} \leq$ Total volume of Urevatn reservoir, then:

$$Volume3_{spill} = 0 \tag{5.10}$$

If, $Volume3_{spill}$ > Total volume of Urevatn reservoir, then:

$$Volume3_{spill} = Volume3_{i+1} - Total^{Urevatn}_{volume} \tag{5.11}$$

Where, $Volume3_{spill}$ represents the volume of Urevatn reservoir spillway.

The constraints must correlate to the state of transformation equation for the reservoir in order to obtain the full potential of the research. The storage state for every iteration 'i' allied with the earlier release from the reservoir by applying the water balance continuity equation:

$$\begin{split} Volume1_{i+1} = Volume1_i + (inflow1_i + flow_{HolenG1-G2,i} + flow_{HolenG3,i} \\ - flow_{Brokke,i} - flow_{pump,i}) * Interval \end{split} \tag{5.12}$$

where $Volume1_{i+1}$ denotes the final volume at Botsvatn reservoir during the instance 'i', $Volume1_i$ is the initial Botsvatn reservoir volume at instance 'i', $inflow1_i$ denotes the inflow (catchment) into Botsvatn reservoir and $Interval$ represents the time interval in seconds (i.e $60 * 60$ for hourly and $60 * 60 * 24$ for daily optimization).

$$Volume2_{i+1} = Volume2_i + Volume3_{spill} + (inflow2_i - flow_{HolenG1-G2,i}) * Interval \tag{5.13}$$

where $Volume2_{i+1}$ denotes the final volume at Vatnedalsvatn reservoir during the instance 'i', $Volume2_i$ is the initial Vatnedalsvatn reservoir volume at instance 'i', $Volume3_{spill}$ is the volume of Urevatn reservoir spillway and $inflow2_i$ denotes the inflow (catchment) into Vatnedalsvatn reservoir.

$$Volume3_{i+1} = Volume3_i + (inflow3_i - flow_{HolenG3,i} + flow_{pump,i}) * Interval \tag{5.14}$$

where $Volume3_{i+1}$ denotes the final volume at Urevatn reservoir during the instance 'i', $Volume3_i$ is the initial Urevatn reservoir volume at instance 'i' and $inflow3_i$ represents the inflow (catchment) into Urevatn reservoir.

Table 5.1 represents the physical characteristics of the Otra Kraft Hydropower system.

Reservoir	**Min. volume [Millm^3]**	**Max. volume [Millm^3]**	**Min. level [m] (LRV)**	**Max. level [m] (HRV)**	**Station**	**Min. flow [m^3/s]**	**Max. flow [m^3/s]**
Botsvatn	1.599*10^8	2.987*10^8	495	551	Brokke	0	137.57
Vatnedalsvatn	4.72*10^8	1.09*10^9	700	840	Holen G1-G2	0	95.33
Urevatn	3.79*10^7	2.36*10^8	1141	1175	Holen G3	0	31.04

TABLE 5.1: Physical Characteristics

Moreover, by using real data concerning flow rates and level of the reservoirs, the constraints must be implemented in such a way to get the optimization results within the actual limit, i.e. within the bound constraints.
Following equations represent flow rate bound constraints for Brokke, Holen 1-2 and Holen 3 generation units.

$$0 \leq flow_{Brokke} \leq 137.5733 \ m^3/s \tag{5.15}$$

$$0 \leq flow_{HolenG1-G2} \leq 95.3305 \ m^3/s \tag{5.16}$$

$$0 \leq flow_{HolenG3} \leq 31.0410 \ m^3/s \tag{5.17}$$

$$0 \leq flow_{pump} \leq 20 \ m^3/s \tag{5.18}$$

Secondly, the bound constraints implemented in such a way that reservoir water level in each time 't' should not exceed the maximum level and should not be less than the minimum level :

$$495 \leq level1 \leq 551 \ m \tag{5.19}$$

$$700 \leq level2 \leq 840 \ m \tag{5.20}$$

$$1141 \leq level3 \leq 1175 \ m \tag{5.21}$$

Correspond to level constraints the volume constraints are implemented to ensure results boundary will not exceed the maximum value and will not be less than minimum :

$$159.9 * 10^6 \leq Volume1 \leq 298.7 * 10^6 \ m^3 \tag{5.22}$$

$$472 * 10^6 \leq Volume2 \leq 1090 * 10^6 \ m^3 \tag{5.23}$$

$$37.9 * 10^6 \leq Volume3 \leq 236 * 10^6 \ m^3 \tag{5.24}$$

5.4 Algorithm setup

Differential Evolution algorithm is based on a virtual population of NP-independent variables. Throughout each generation, these variables reproduce (offspring) and withstand selection. Merely the best or most suitable variable survive to reproduce in the subsequent generation. Hence after consecutive generations, the population becomes better-thereby recognizing the best optimum of an objective function. The Matlab-script used to perform differential evolution algorithm was adapted from Price, 1997 as reference.

5.4.1 Differential Evolution Parameters

D = 4*Number of hours (hourly optimization) or **D = 4*Number of days** (daily average optimization) and can be defined as the number of parameters of the objective function.

XVmin and XVmax are the lower and upper bounds of the initial population and it covers the region where the global minimum/maximum is expected.

10*D is the number of population members (NP).

itermax is the maximum number of iterations or generations can be used to find the best optimal solution.

F is the stepsize from the interval 0-2.

CR is the crossover probability constant from interval [0, 1] and helps to maintain the diversity of the population and is rather uncritical.

bestmem is the parameter vector with best solution

bestval is the best objective function value.

nfeval is the number of function evaluations

5.4.2 Population (POP)

In the optimization algorithm, we have four decision variables which are flow rate from Brokke, Holen 1-2, Holen 3 and Pump. These flow rates have 169 different values if we run the algorithm on an hourly basis for a week (each hour correspond to a different value) or if we run it for a year with daily average values then there are 365 different values. Accordingly, the dimension of the population matrix will be:

$$pop = \left[NP * D\right]$$

Where, D will be number of decision variables multiply by number of days/hours and NP will be 10*D.

5.4.3 Initialize Randomly

The population (pop) is initialized randomly between the minimum and maximum values of the parameters i.e XVmin and XVmax shown in Equation 5.25.

$$pop(i,:) = XVmin + rand(1, D). * (XVmax - XVmin); \tag{5.25}$$

After initializing the population the next step is to estimate the best member (bestmem) starting from the first population member as shown in syntax 5.26. Sequentially it will find the best objective function value i.e **val**.

$$\begin{aligned} popold &= zeros(size(pop)) \\ val &= zeros(1, NP) \\ bestmem &= zeros(1, D) \\ bestmemit &= zeros(1, D) \\ nfeval &= 0 \end{aligned} \tag{5.26}$$

where, $popold$ is the old population that is to be toggled, val represents the cost array, $bestmem$ represents the best population member, $bestmemit$ represents the best member in iteration and $nfeval$ is the number of function evaluations.
After intialization the evaluation of the best member is executed starting from the first population member.

$$ibest = zeros(size(pop))$$

Estimating the best objective function value so far

$$\begin{aligned} val(1) &= feval(fname, pop(ibest,:), y) \\ bestval &= val(1) \end{aligned}$$

Checking the remaining members in the population

$$\begin{aligned} val(i) &= feval(fname, pop(ibest,:), y) \\ nfeval &= nfeval + 1 \end{aligned}$$

To check if the member is best and saving it's location

$$\begin{aligned} &if\ (val(i) \leq bestval \\ &\quad ibest = i \\ &bestval = val(i) \end{aligned}$$

The best member and best value of the current iteration

$$\begin{aligned} bestmemit &= pop(ibest,:) \\ bestvalit &= bestval \end{aligned}$$

Hence, the best member ever can be expressed as

$$bestmem = bestmemit$$

5.4.4 Rotation

The next step is to rotate (**rot**) the index (**ind**) array (**rt**) and shuffle the position of vectors (**a1 - a5**) for exponential crossover. Hence to change the vector location every time throughout the crossover process. Moreover, we can choose the best member (bestmem) population of the last iteration expressed in below.

Saving the old population so as to compare it with new one afterwards

$$popold = pop$$

Shuffling the location of arrays and rotating indices by $ind(1)$ positions for each vector i.e $a1 - a5$

$$\begin{aligned}
a1 &= randperm(NP) \\
rt &= rem(rot + ind(1), NP) \\
a2 &= a1(rt + 1) \\
rt &= rem(rot + ind(2), NP) \\
a3 &= a2(rt + 1) \\
rt &= rem(rot + ind(3), NP) \\
a4 &= a3(rt + 1) \\
rt &= rem(rot + ind(4), NP) \\
a5 &= a4(rt + 1)
\end{aligned}$$

For each vector the shuffled population can be expressed as

$$\begin{aligned}
pm1 &= popold(a1, :) \\
pm2 &= popold(a2, :) \\
pm3 &= popold(a3, :) \\
pm4 &= popold(a4, :) \\
pm5 &= popold(a5, :)
\end{aligned}$$

The population filled with the best member of the last iteration is given by

$$bm(i, :) = bestmemit$$

Where pm1-pm5 denotes the modified population matrix after rotation.

5.4.5 Mutation

The old population i.e., **popold**, will participate and will remain static throughout one iteration. Further the new population i.e **pop** will appear. During the initialization of the population matrix and the bestmem matrix, there remain unsettled vectors in population, and the intermediate population of these vectors is assigned as **ui**, and **mui** represents the mask for an intermediate population.

5.4.6 Crossover

However numerous crossover methods are appropriate, binomial crossover and exponential crossover are extensively used in differential evolution.

Differential Mutation Operators

The most commonly used variant is expressed as DE/rand/1. It consists of adding a scaled difference vector between two randomly selected individuals and to a third randomly picked individual (pm3), called the base vector:

$$u_i \longleftarrow pm3 + F * (pm1 - pm2)$$

In DE/best/1, the base vector is defined as the best individual (bm) in the current population:

$$u_i \longleftarrow bm + F * (pm1 - pm2)$$

In DE/rand-to-best/1, the base vector is equivalent to the parent vector and is added to two difference vectors, a randomly generated one and a vector from the parent to the best member in the population.

$$u_i \longleftarrow popold + F * (bm - popold) + F * (pm1 - pm2)$$

The DE/rand/2 operator consists of adding two difference vectors to a randomly chosen member:

$$u_i \longleftarrow pm5 + F * (pm1 - pm2 + pm3 - pm4)$$

Hence, by applying differential mutation operators and strategies the crossover operation is executed as mentioned in Appendix A.

5.4.7 Selection

The random selection of vectors is done by shuffling the population array. Thus a particular vector can't be chosen twice in the same term of the disorder expression. The first step is to select the vectors that are allowed to enter the new population and to check the cost of competitor i.e

$$tempval = feval(fname, ui(i,:), y)$$
$$nfeval = nfeval + 1$$

If the competitor is better than the value in "cost array" the old vector gets replaced by the new one (for new iteration) and the new value gets saved in "cost array"

$$if\ (tempval \leq val(i)$$
$$pop(i,:) = ui(i,:)$$
$$bestmem = ui(i,:)$$

Freezing the best member of this interation for the coming interations as this will be needed for some of the strategies.

$$bestmemit = bestmem$$

Chapter 6

Discussion of the results, conclusion and recommendations

6.1 Introduction

In this chapter, The results for the proposed daily average and hourly optimization are presented. Firstly, the daily average optimized results for a year, i.e., from 08/Oct/2016 to 07/Oct/2017 are presented which illustrates the best values concerning flow rate, reservoir volume, and level which proffers maximum profit (maximum revenue) by retaining the reservoir levels and water flow rate within the limit. Further, it can be observed that during the actual operation the reservoir's level is controlled by decreasing or discontinue the flow rate which is restricting the generation of power and revenue. Whereas in the proposed algorithm this situation is not occurring because of the pumping system impersonating a vital role to prevent the overflow at Botsvatn reservoir.

Secondly, The hourly optimization for the randomly chosen weeks throughout the year is executed in order to test the performance and accuracy of the proposed daily average optimization. Two different policies were taken into account. Daily average release policy is taken into consideration, compared with the hourly basis release policy to show the performance of the algorithm.

Finally, The tables have presented which show the optimized and actual results. Consequently, we can observe that the optimized curves are within maximum and minimum boundaries. Furthermore, we can remark the profit that originates from the proposed differential evolution algorithm.

NOTE: All the simulations conducted in the thesis had initial conditions equivalent to the actual initial values concerning volume and level of the reservoirs.

6.2 Results and discussion

6.2.1 Daily average

To perform daily average optimization, firstly the actual flow and reservoir volume hourly data is averaged daily, i.e., for every 24 hours. Accordingly, the reservoir level and power curves were made (shown in chapter 4). Finally, the DE algorithm is implemented to generate maximum revenue. Following plots represents the optimal curves obtained by executing daily average optimization for a year, i.e., from 08 Oct 2016 to 08 Oct 2017 together with the actual operational curves.

Reservoir Level

Figure 6.1 displays the optimized and actual reservoir level. We can observe in the plot that the actual level of Botsvatn reservoir is exceeding HRV boundary during the midst of June whereas the optimized level is beneath HRV boundary. Hence the algorithm is successful in controlling the level boundaries.

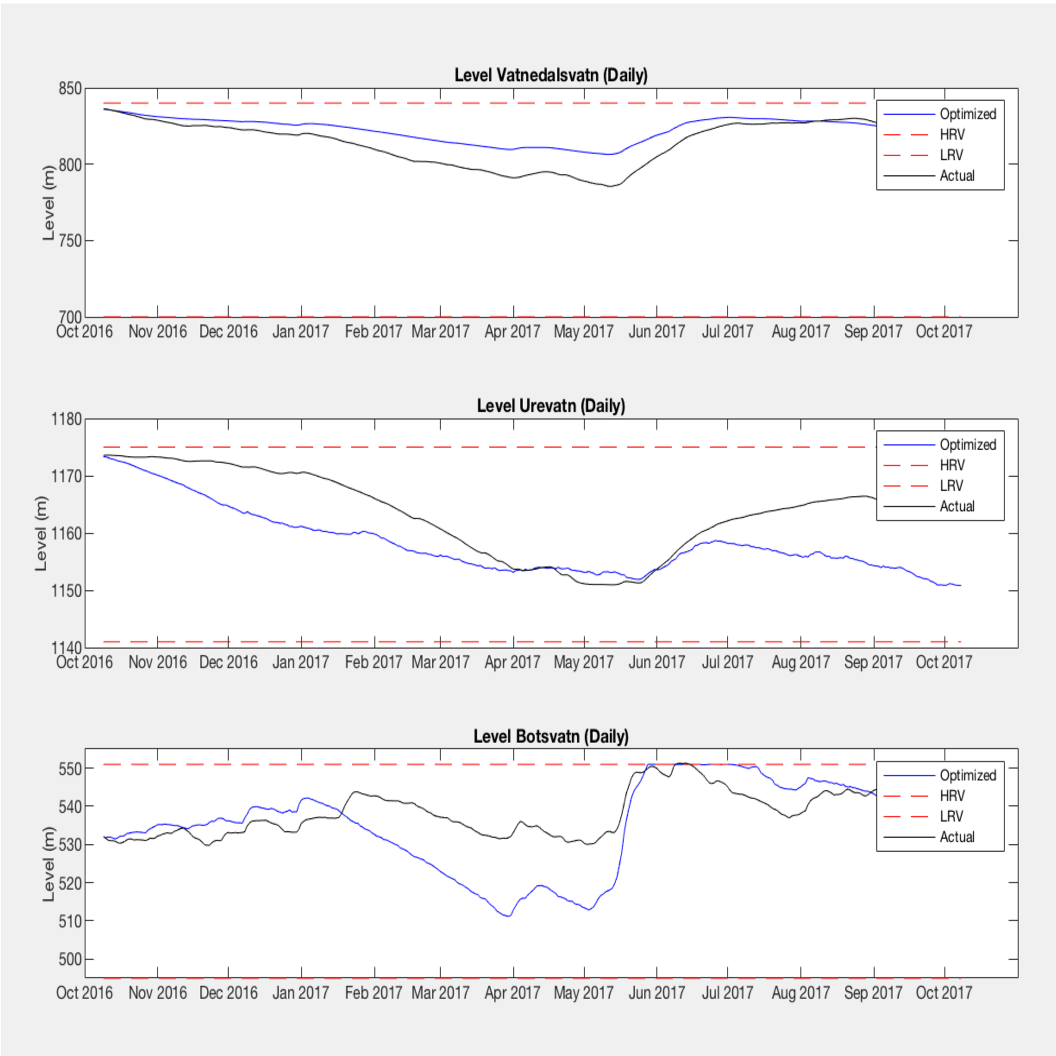

FIGURE 6.1: Optimized vs actual reservoir level

Reservoir volume

Figure 6.2 represents the optimized and actual volume of the reservoirs throughout the year which is functioning similar to the reservoir level curves. The last plot shows the corresponding inflows into the reservoirs, and we can observe in the plot during the midst of May to June the inflow at Botsvatn reservoir is too high which forces algorithm to put pumping system to operate to avoid overflow.

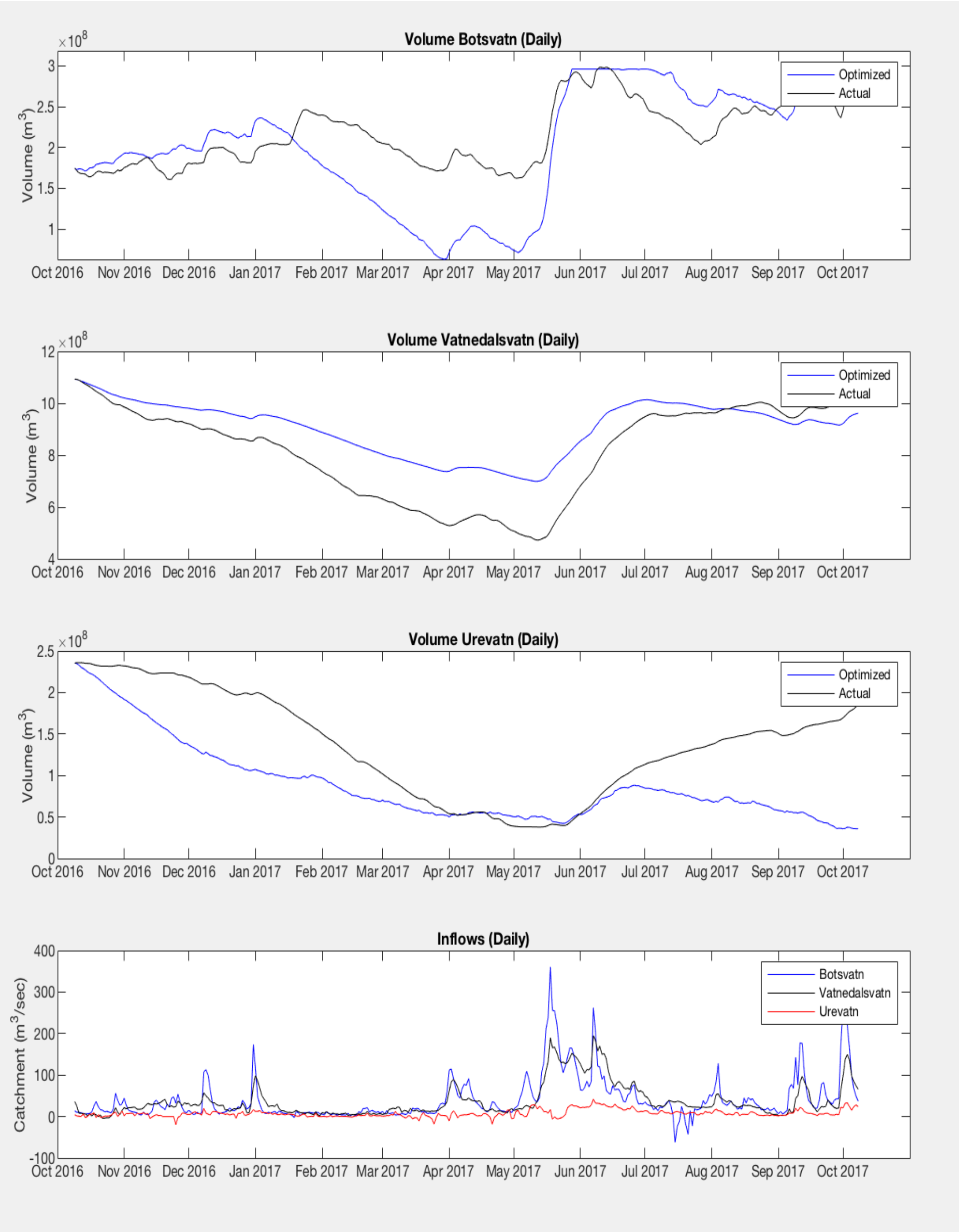

FIGURE 6.2: Optimized vs actual reservoir volume

Flow rate

Figure 6.3 displays the optimized and actual flow rate through the Pump and three distinct turbine units within the actual limits. We can recognize in the plot that flow rate from Holen G1-G2 and Holen G3 unit has stopped from period 18 May to 30 June due to an increase in corresponding Botsvatn reservoir level whereas optimized value is at its peak. So we can perceive that we do not require to discontinue flow from any of the turbine units to control the level of the reservoir for such a prolonged duration. Hence, the proposed algorithm can generate maximum revenue from the system.

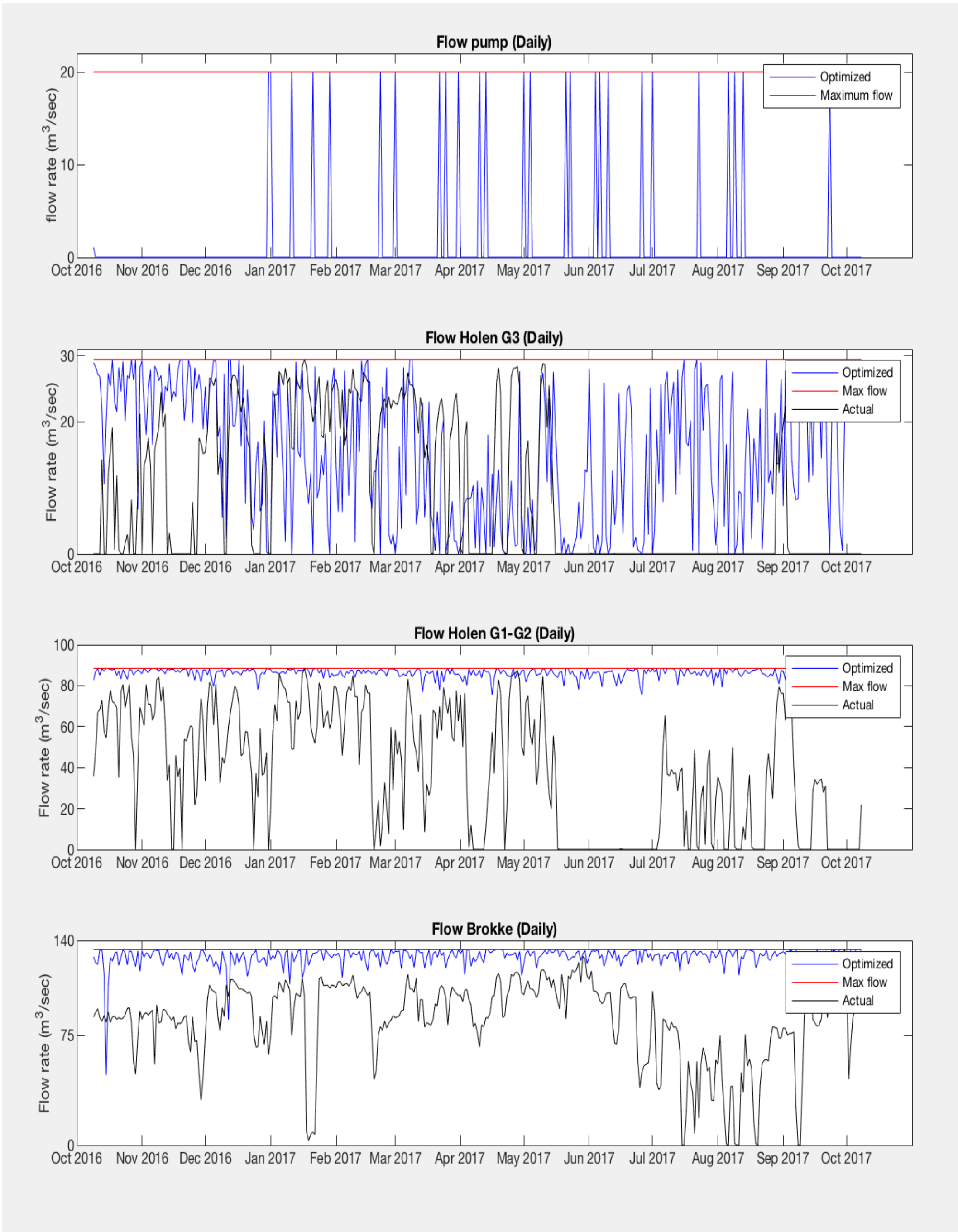

FIGURE 6.3: Optimized flow rate vs actual flow rate

Revenue

Figure 6.4 shows the functioning of a differential evolution algorithm which represents the optimized revenue originating per iteration. The red dotted line in the figure signifies the actual revenue generated throughout the year, and the black dots show the optimized revenue from the system for each iteration until the best value.

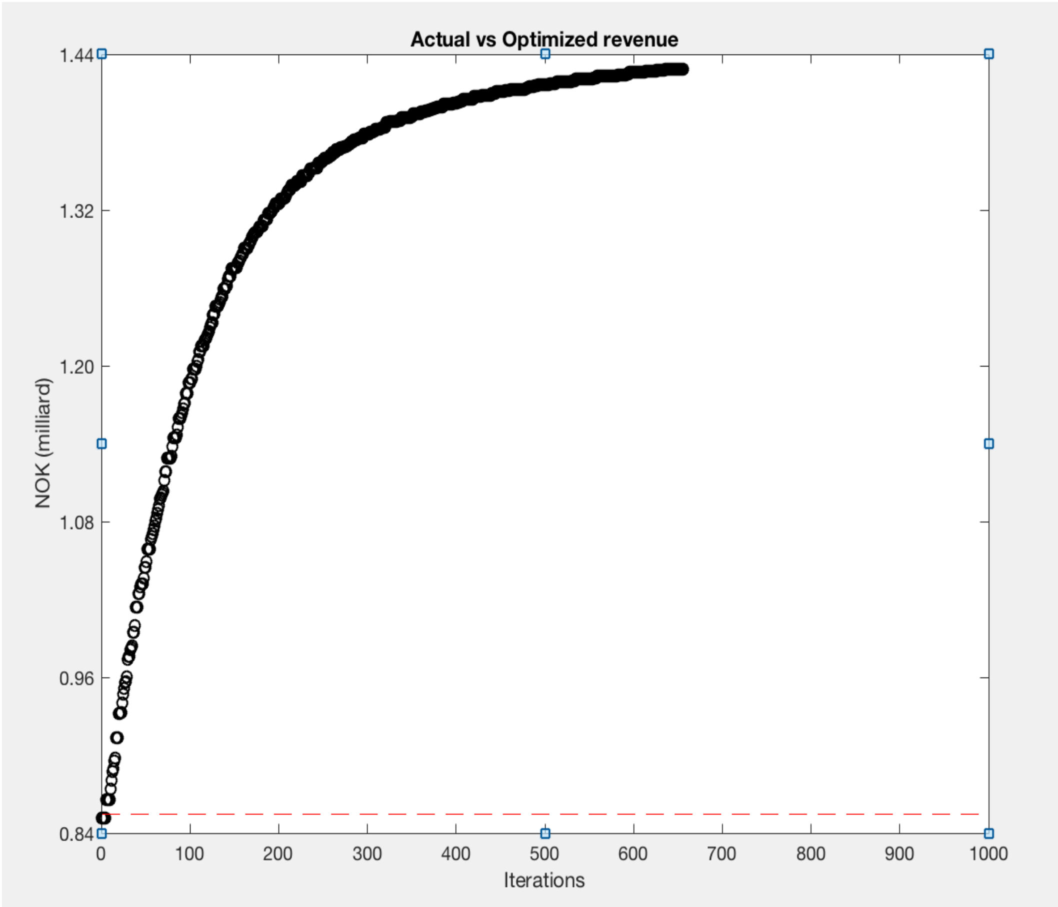

FIGURE 6.4: Optimized vs actual revenue

Table 6.1 presents the results from daily average optimization, and we can observe that the optimization model is getting a satisfactory benefit.

Actual revenue [NOK milliard]	Optimized revenue [NOK milliard]	Profit [NOK milliard]	Pumped [TWh]	Total generation [TWh]	Pump operation [days]
0.85	1.43	0.57	0.098	5.24	26

TABLE 6.1: Daily average optimization results

6.2.2 Daily and hourly optimization results: 08 Oct - 15 Oct 2016

The hourly optimization is executed for randomly chosen weeks to determine the accuracy and aspects of the results by comparing it to daily average optimization.

Flow rate

It can be observed in figure 6.5 that during hourly optimization the pump system and turbine units are working actively whereas in daily optimization these units are flat which indicates hourly optimization is more accurate if correlated with the daily average optimization.

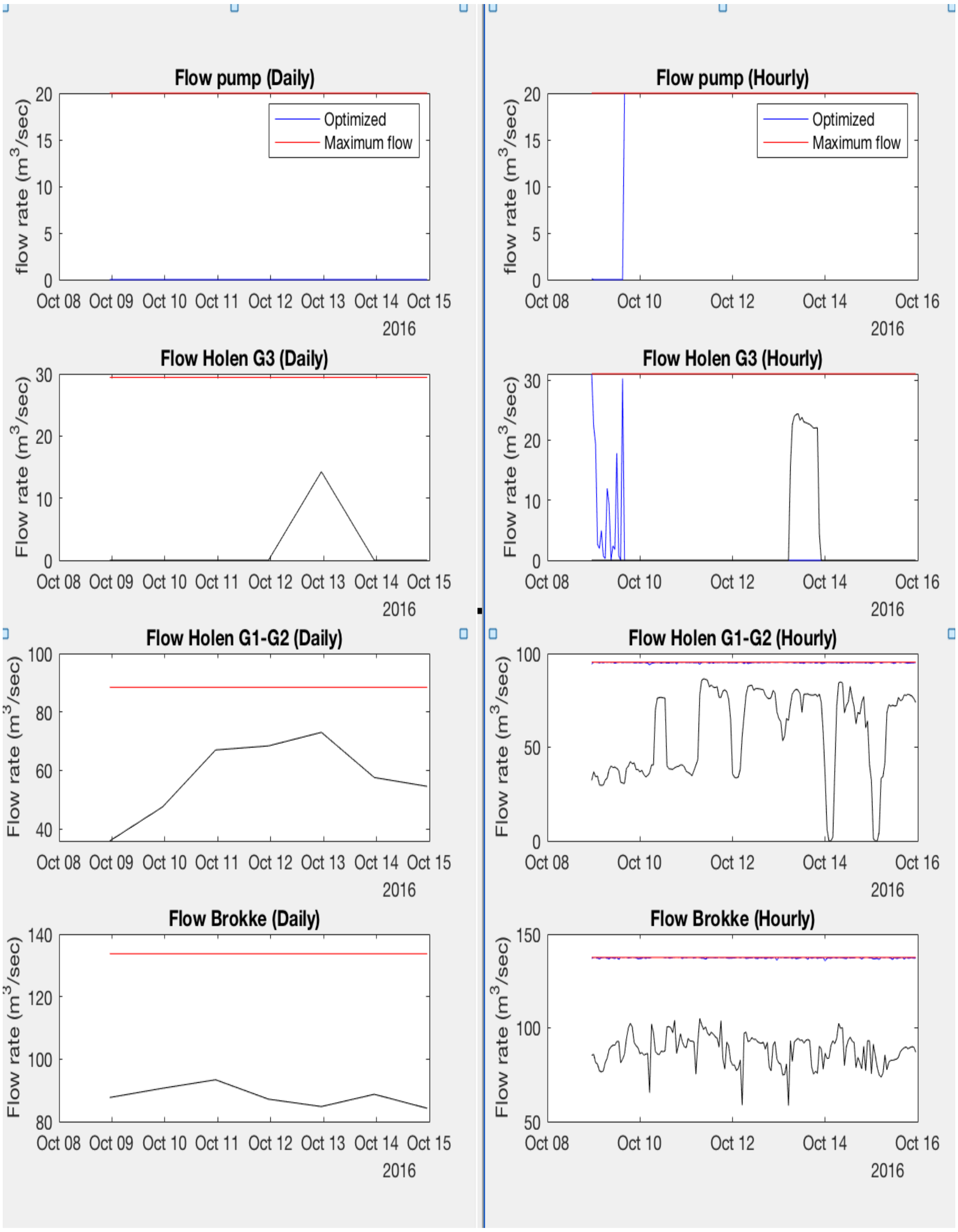

FIGURE 6.5: Hourly vs Daily average flow: 08 Oct - 15 Oct 2016

Reservoir level

Figure 6.6 represents the optimized level during hourly and daily average optimization. We can observe the change in Urevatn reservoir level because the pump is operating extravagantly during hourly optimization whereas in daily average it is not operating.

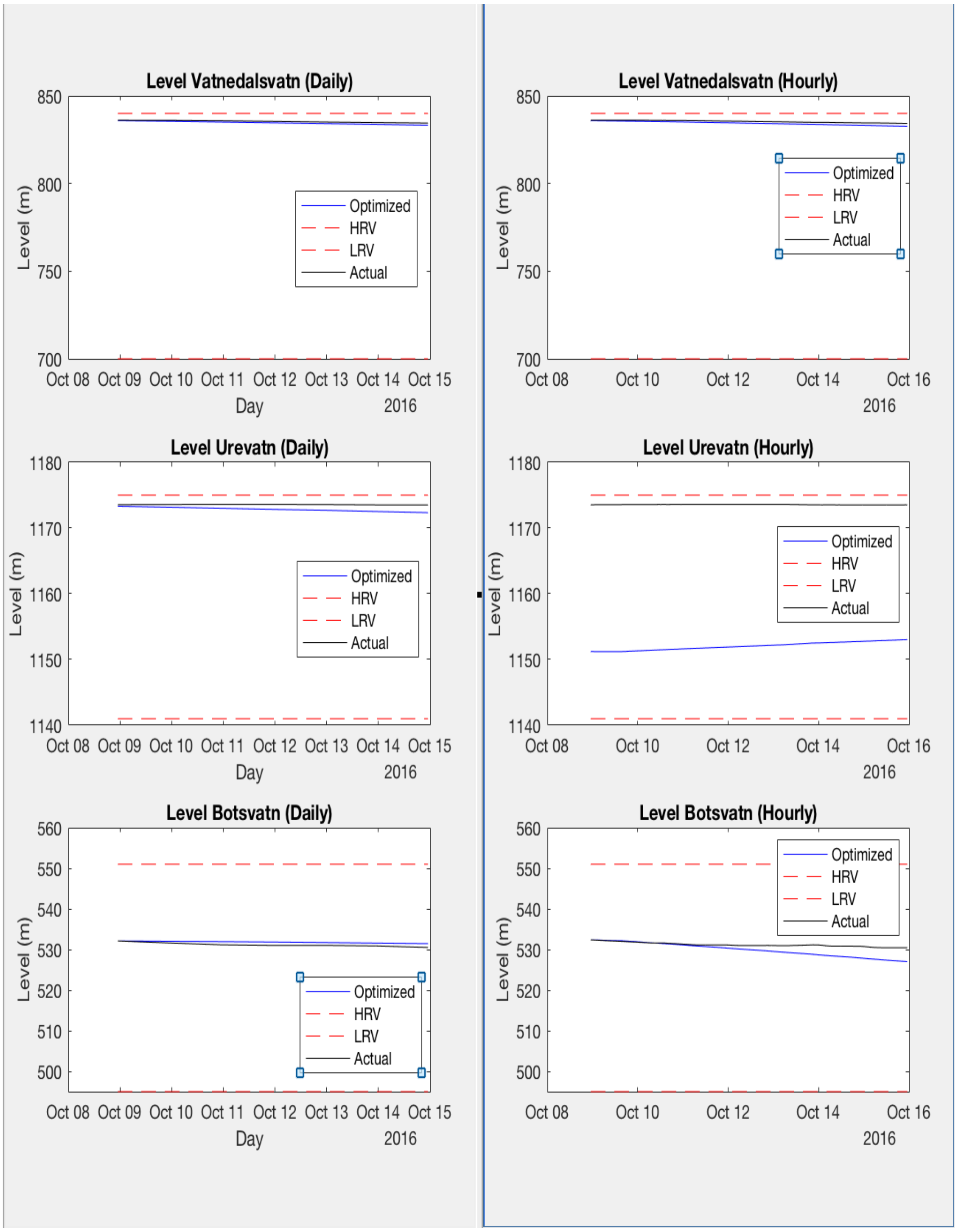

FIGURE 6.6: Hourly vs Daily average reservoir level: 08 Oct - 15 Oct 2016

Revenue

The revenue obtained from hourly optimization is lower than the daily average because of the value of decision variables. As in daily average optimization we are running the algorithm for seven days which means single value (average) per day ($24 * Avg.value$) on the other hand if we run the algorithm for hourly optimization for a week we are dealing with 168 hours means 168 different values. Hence we can say that hourly optimization allows taking advantage of the hourly price variation during each day, which of course will allow achieving a more accurate solution than the one using only daily average values.

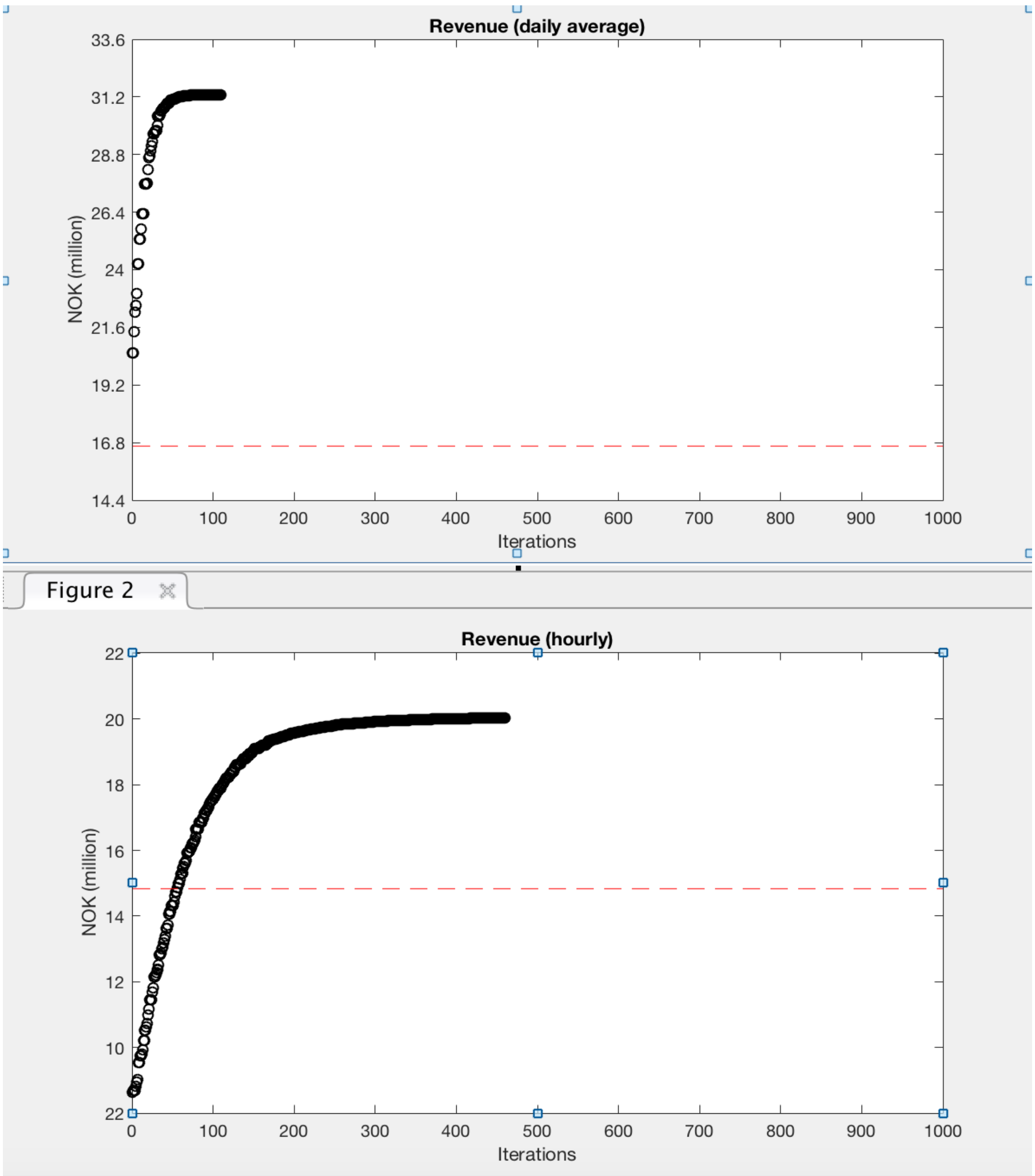

FIGURE 6.7: Hourly vs Daily revenue generated: 08 Oct - 15 Oct 2016

6.2.3 Daily and hourly optimization results: 16 May - 23 May 2017

Flow rate

Figure 6.8 shows the optimized flow rate from the pumping and turbine units. We can observe the flow results from hourly optimization is precise than the daily average results.

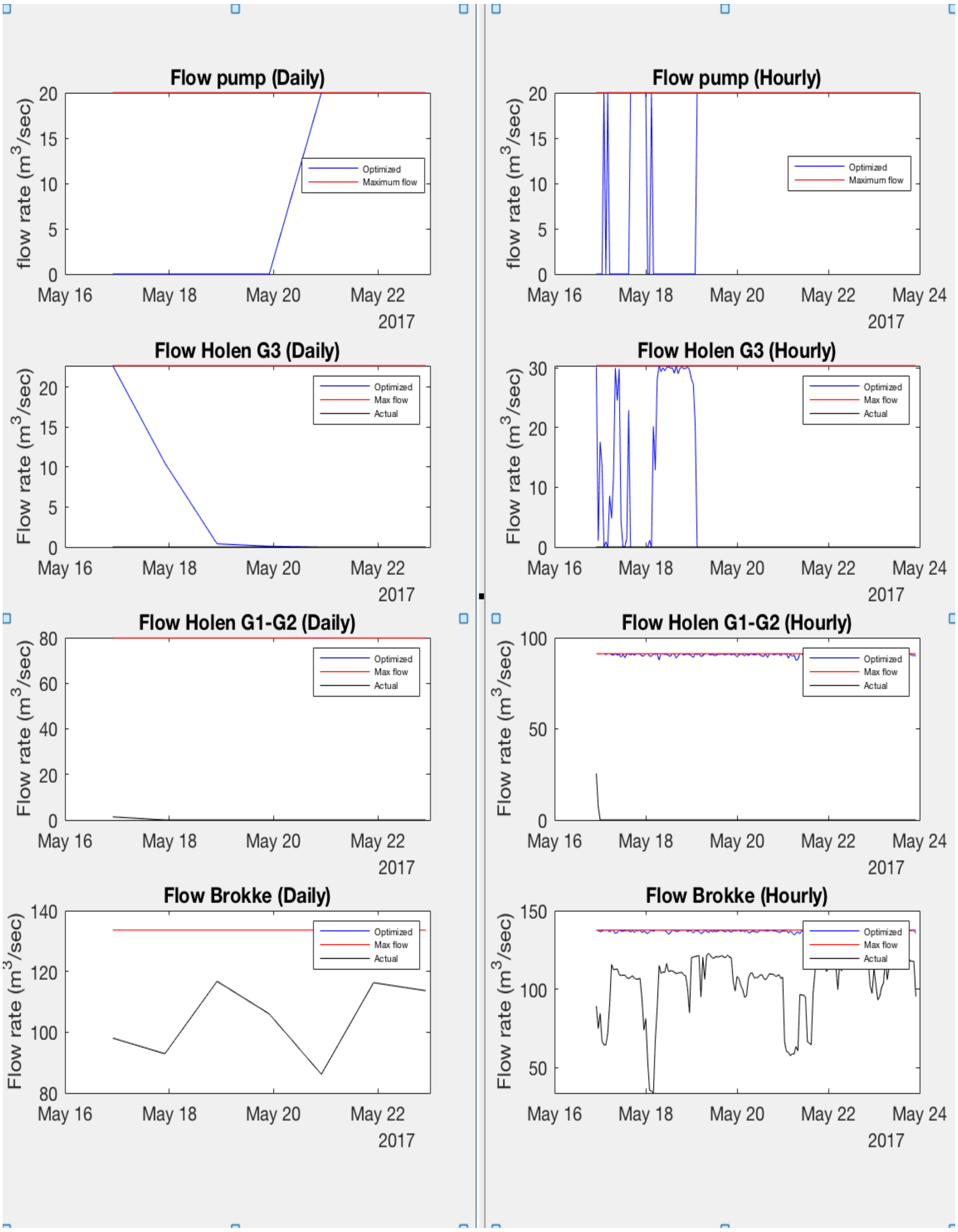

FIGURE 6.8: Hourly vs Daily average flow: 16 May - 23 May 2017

Reservoir level

In figure 6.9 we can perceive that there is not much variation between hourly and daily average results because of similar flow rate curves.

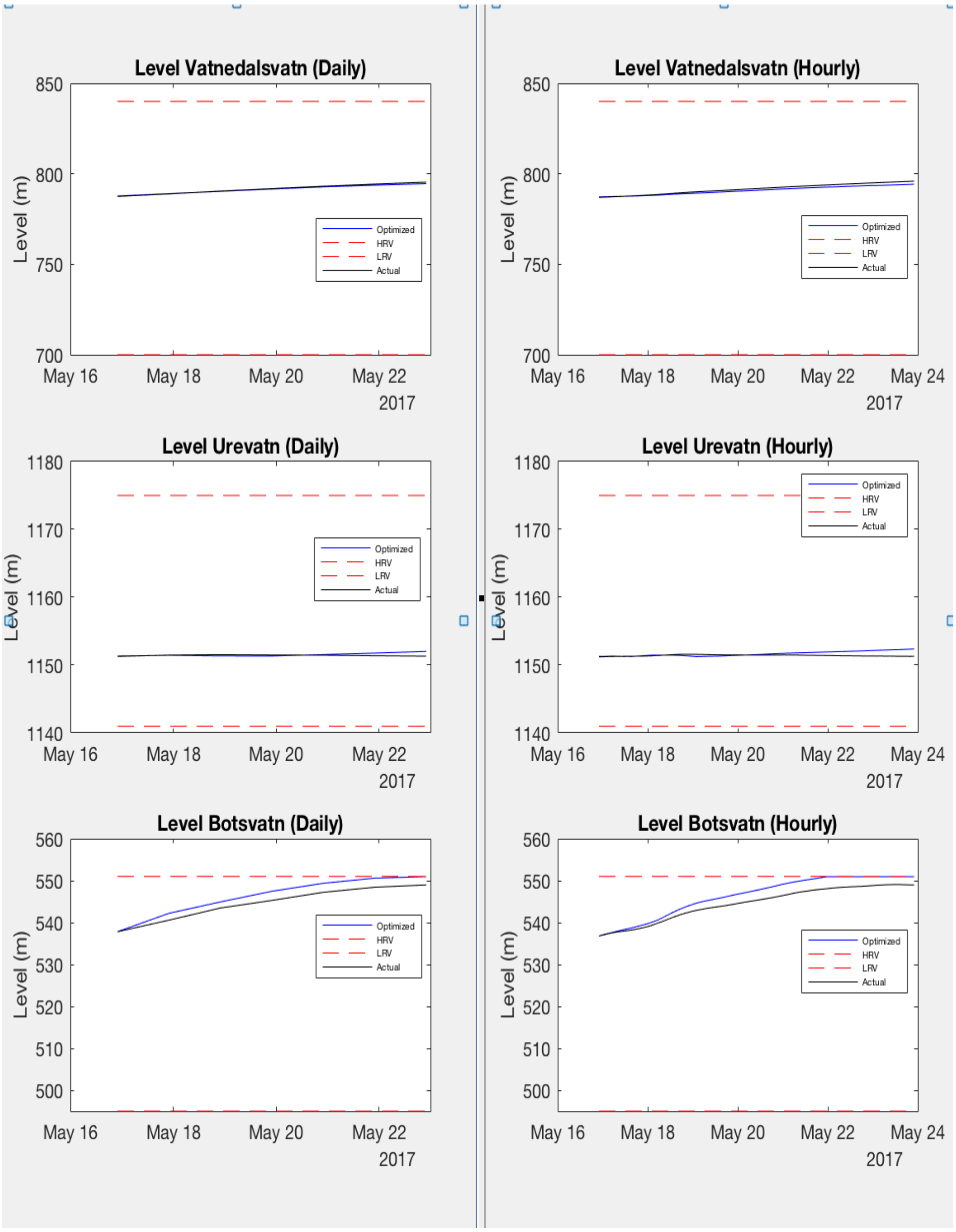

FIGURE 6.9: Hourly vs Daily average reservoir level: 16 May - 23 May 2017

Revenue

Figure 6.10 shows the revenue generated during both scenarios. Still, we are making a good profit from both situations. However hourly optimization proffers similar value concerning daily average optimization.

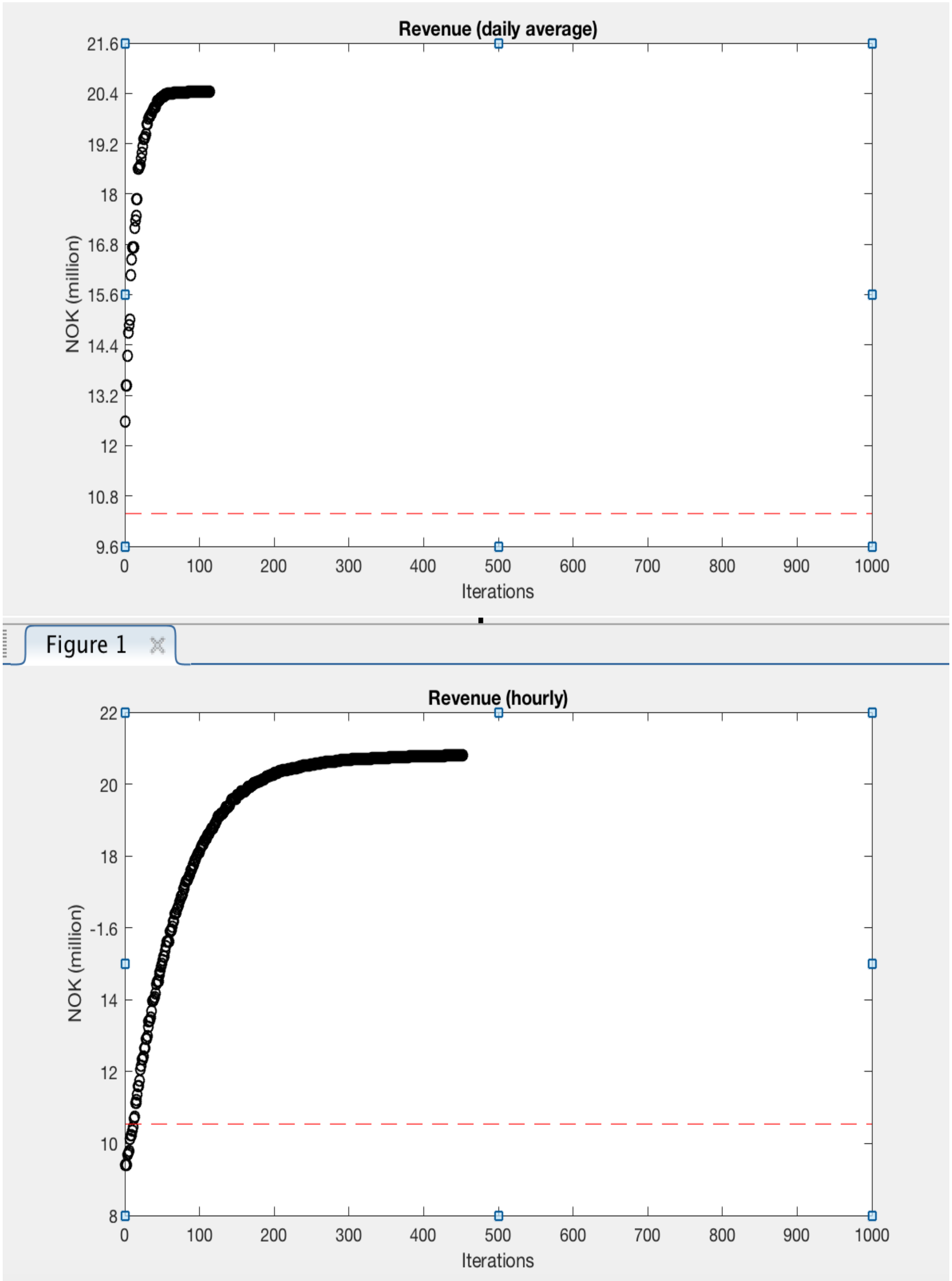

FIGURE 6.10: Hourly vs Daily revenue generated: 16 May - 23 May 2017

6.2.4 Daily and hourly optimization results: 29 June - 13 July 2017

Flow rate

Figure 6.11 shows the optimized flow rate from the pumping and turbine units. We can observe the flow results from hourly optimization is more defined compared to the daily average results.

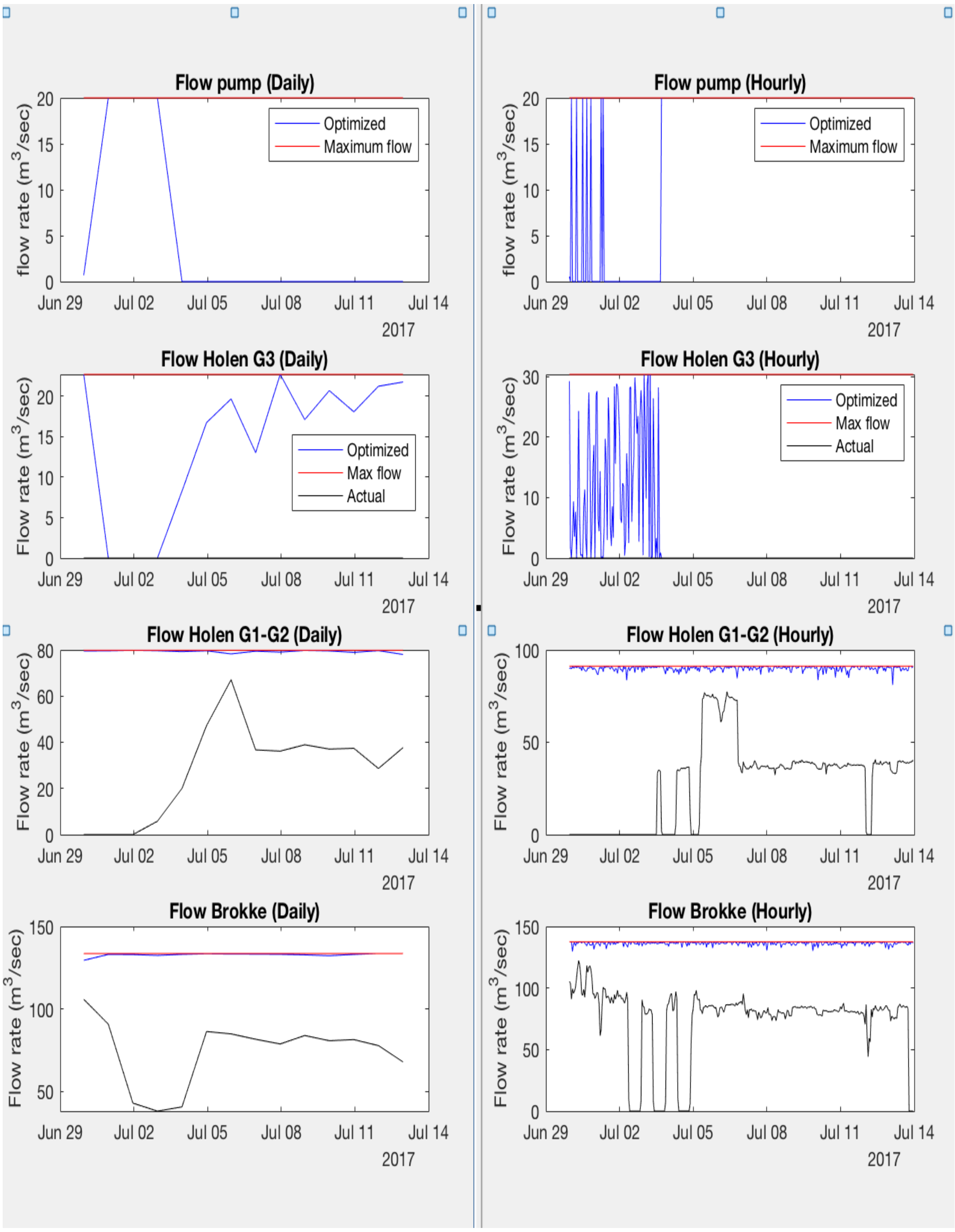

FIGURE 6.11: Hourly vs Daily average flow: 29 June - 13 July 2017

Reservoir level

In figure 6.12 we can perceive that there is not much variation between hourly and daily average results because of similar flow rate curves.

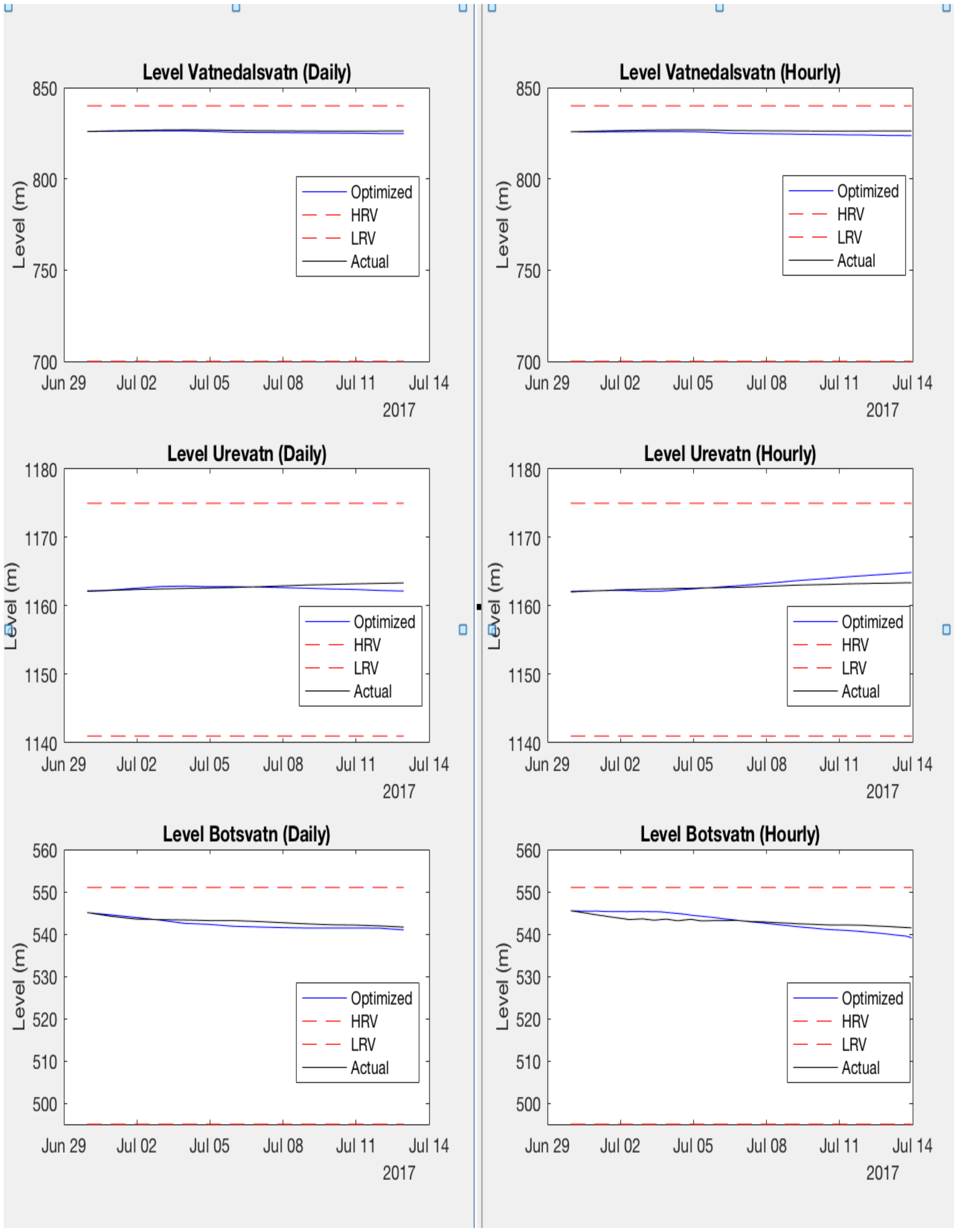

FIGURE 6.12: Hourly vs Daily average reservoir level: 29 June - 13 July 2017

Revenue

Figure 6.13 shows the revenue generated during both scenarios. Still, we are making a good profit from both situations. However, the revenue obtained from hourly optimization is lower than the daily average, since, hourly optimization allows taking advantage of the hourly price variation during each day, which of course will allow achieving a more accurate solution than the one using only daily average values.

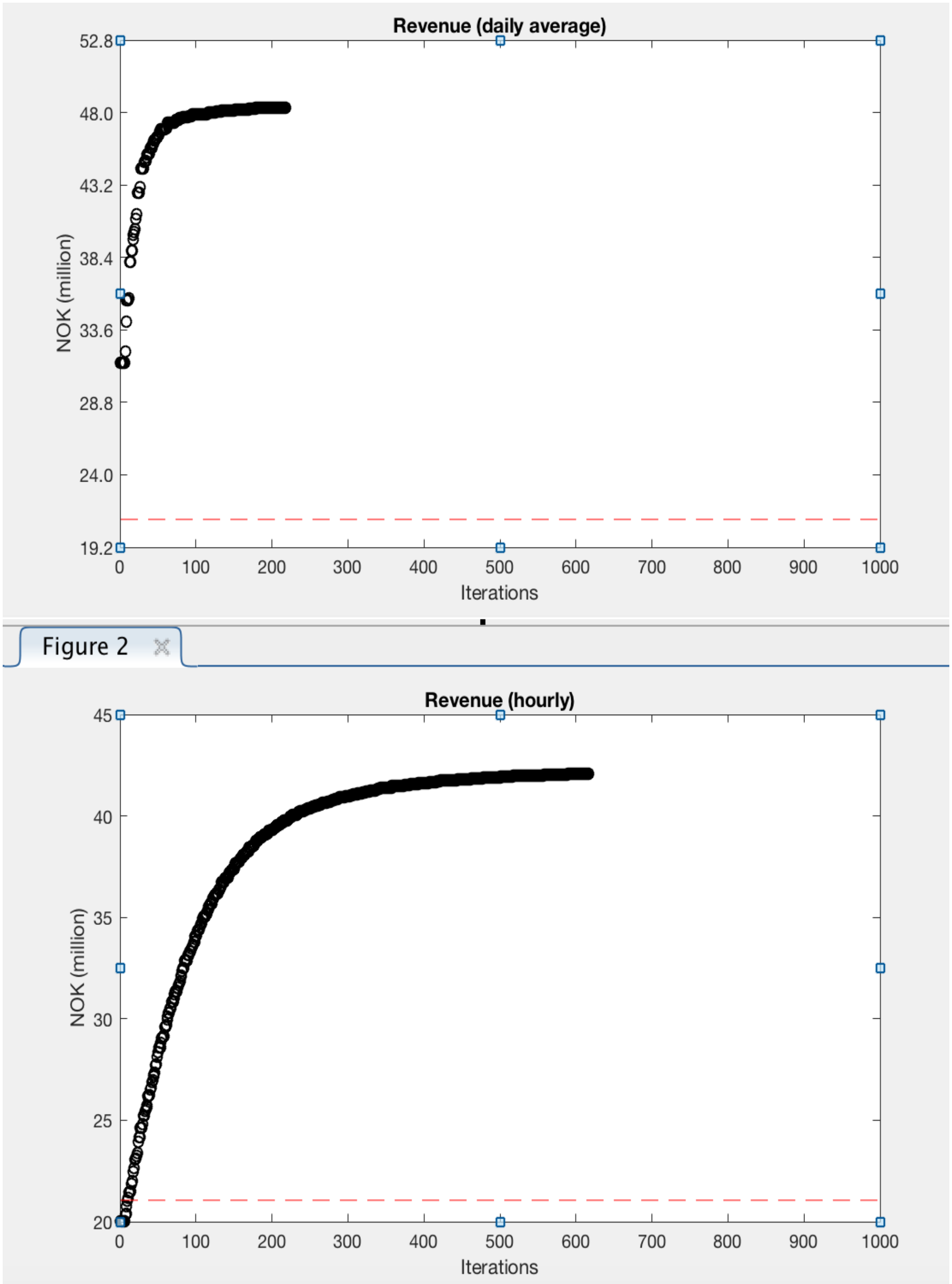

FIGURE 6.13: Hourly vs Daily revenue generated: 29 June - 13 July 2017

6.2.5 Daily and hourly optimization results: 04 Aug - 11 Aug 2017

Following decisions represents the contrast in hourly and daily average optimization results which signifies the accuracy of hourly optimization over the daily average optimization.

Flow rate

It can be observed in figure 6.14 that during both situations the pump system and turbine units are working adequately. However, hourly optimization plots are more precise.

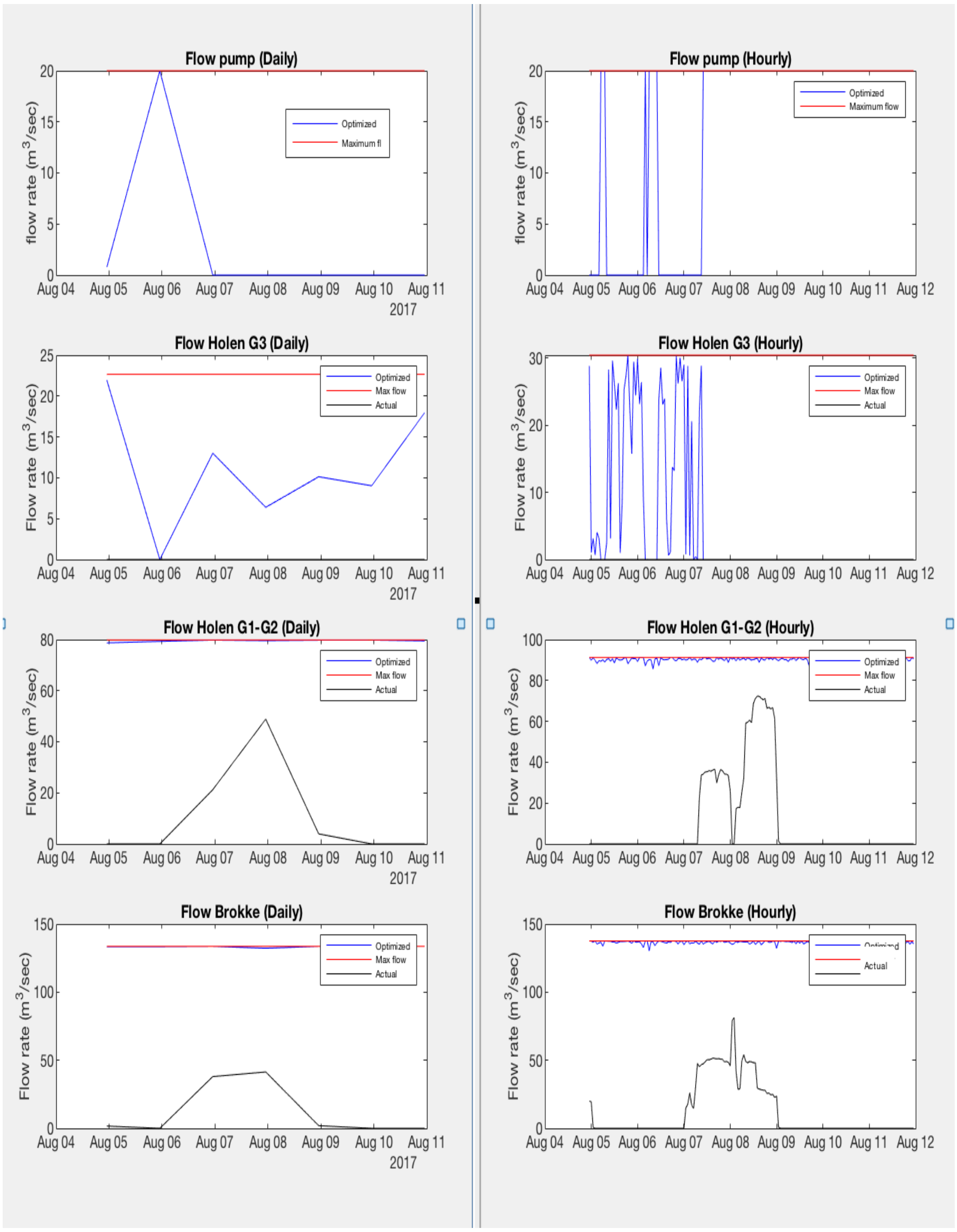

FIGURE 6.14: Hourly vs Daily average flow: 04 Aug - 11 Aug 2017

Reservoir level

Figure 6.15 represents the optimized level during both scenarios, and the plots are similar to each other because of the related flow curves.

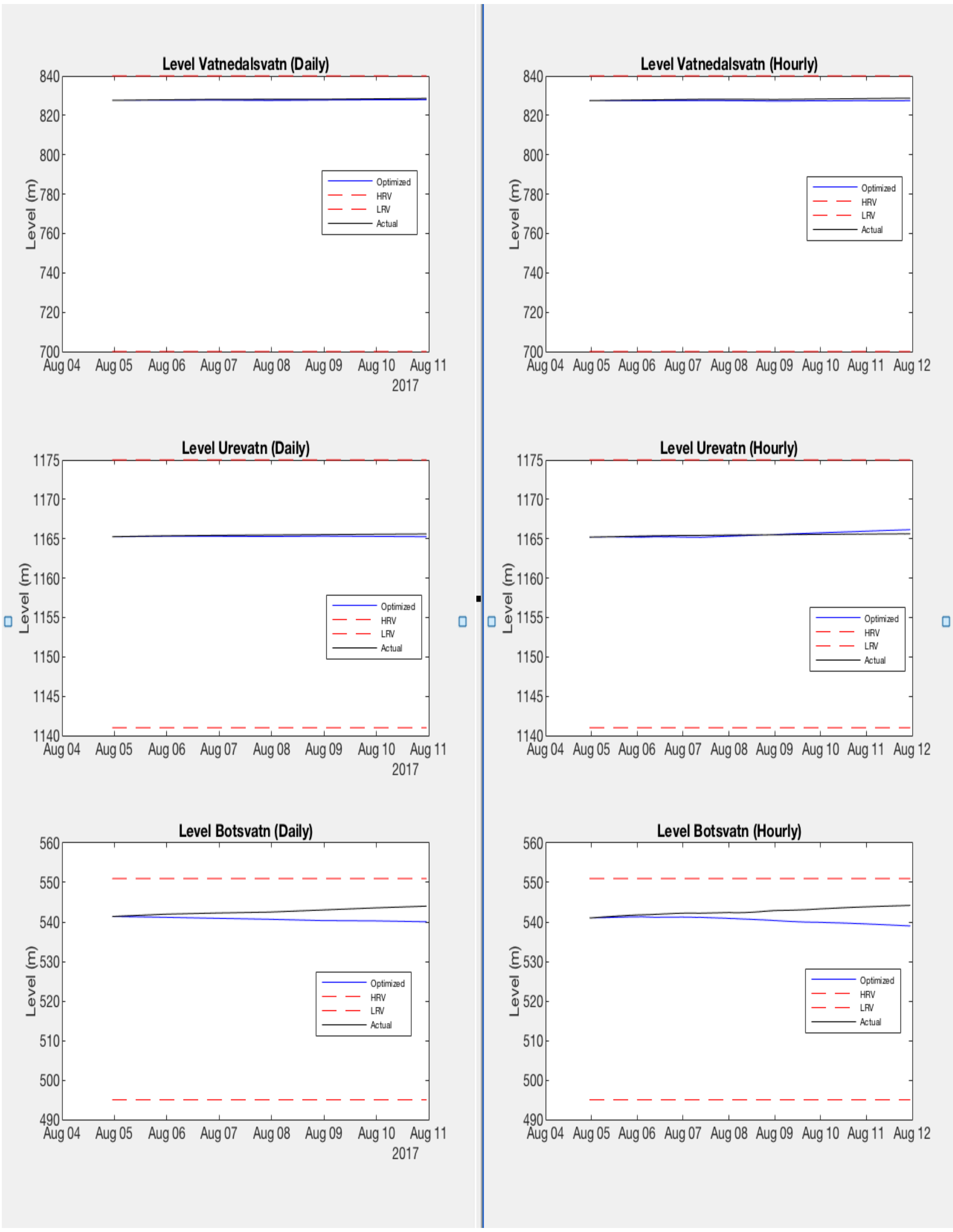

FIGURE 6.15: Hourly vs Daily average reservoir level: 04 Aug - 11 Aug 2017

Revenue

Figure 6.16 shows the revenue generation by the algorithm for both scenarios. Both curves are giving better results compared to the actual one. However, the curves show the advantage of hourly optimization over daily average to estimate the precise revenue.

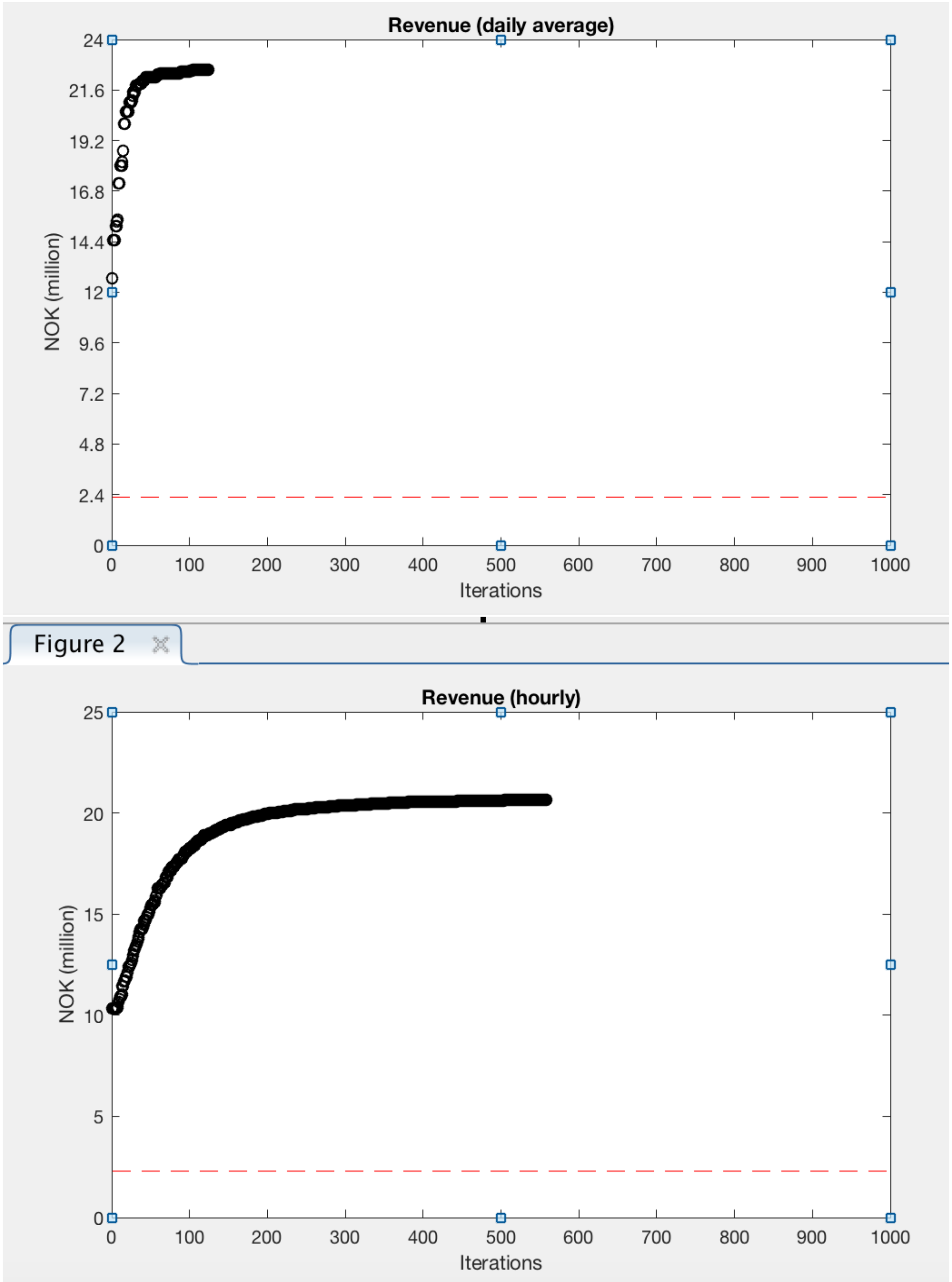

FIGURE 6.16: Hourly vs Daily revenue generated: 04 Aug - 11 Aug 2017

6.2.6 Hourly optimization results

Table 6.2 shows the results for hourly optimization during randomly chosen period and we can observe that we are getting sufficient benefit.

Duration	**Actual revenue [NOK million]**	**Optimized revenue [NOK million]**	**Profit [NOK million]**	**Pumped [GWh]**	**Total generation [GWh]**
08 Oct - 15 Oct 2016	16.8	22	5.2	22.02	96.12
16 May - 23 May 2017	10.5	20.8	10.2	18.98	97.91
29 June - 13 July 2017	21	42	21	36.81	19.41
04 Aug - 11 Aug 2017	2.3	20	18.3	17.56	98.55

TABLE 6.2: Hourly optimization results

6.3 Conclusion

In this thesis, the optimization of multi-reservoir pumped storage hydropower system is executed based on daily average values and hourly data by using differential evolution algorithm.

An algorithm was developed to manage the optimal operation of three different reservoirs with the pump system in Otra Kraft hydropower system using single speed pump unit of capacity 20 m^3/s. According to the requirement, the pump chosen is considered to be established together with Holen-3 turbine unit. Accordingly, a ON/OFF switch has been developed in the algorithm so that the turbine and pump unit cannot operate at the same time. In the study, when using a pump unit, the model exhibited improvements in both energy storage capacity and energy generation capacity. The optimization is performed by utilizing non-linear constraints which have conferred the ability of the algorithm to keep reservoir level in a controlled state, correspondingly to maximize and control discharging production flow, and to avoid overflow.

It is beneficial to use the differential evolution algorithm on the hydropower system as it can derive multiple alternatives concerning reservoir discharge, level, and volume. The results confirm the convergence performance of evolutionary algorithms varies from conventional calculus-based methods. Evolutionary algorithms show longer solution times— identified by prompt classification of the region containing the best or optimum values, with relatively moderate local convergence. DE algorithm appears beneficial for the hydropower optimization problems which are discrete, non-convex and irregular. DE algorithm is easily applied to such problems and could give direction for daily operational decisions at hydropower plants.

The optimization results imply that the hourly optimization for less duration (a week or less) performs better with high-grade quality reliability and smaller vulnerability than the daily average optimization for a year. However, daily average optimization is the best alternative for the estimation of optimal average flows, reservoir level, and revenue for a long duration by using the historical data as an input. Therefore, the method mentioned above might be compatible with flow-rate prediction to provide an alternative optimal solution to increase the revenue or profit.

6.4 Recommendations

In this research, pump selection is made according to the company requirement. However, the best-recommended site for the pump installation should be within Botsvatn, and Vatnedalsvatn reservoir as the head between these two reservoirs is almost half of the head between Botsvatn and Urevatn reservoir. Figure 3.6 gives an overview of the connectivity between the reservoirs. As the objective function is directly related to the hydraulic equation i.e.

$$P_{pump} = \frac{\rho.H.g.Q}{\eta_{pump}} \tag{6.1}$$

Therefore the power required by a pump to elevate water to Vatnedalsvatn reservoir will be half of the pump that is to be installed within Botsvatn and Urevatn.

Furthermore Vatnedalsvatn reservoir has the maximum capacity compared to other reservoirs.

Bibliography

Abrenna (2019). "**Electricity imprted/exported from Norway**". In: *enerwe* 01. URL: https://enerwe.no/sa-mye-strom-eksporterte-og-importerte-norge-til-og-fra-sverige-danmark-finnland-nederland-og-russland/140263?source=facebook&campaign=utenlandskabler.

al, J.M.K.C. Donev et (2018). ***Energy Education - Hydraulic head [Online]***. URL: https://energyeducation.ca/encyclopedia/Hydraulic_head.

Ana Cláudia F. Medeiros Bragaa Richarde Marques da Silva, Celso Augusto Guimarães Santosa Carlos de Oliveira Galvãoc Paulo Nobre (2013). "**Downscaling of a global climate model for estimation of runoff, sediment yield and dam storage: A case study of Pirapama basin, Brazil**". In: *Journal of Hydrology* 498, pp. 46–58. URL: https://doi.org/10.1016/j.jhydrol.2013.06.007.

Antonio Frangioni Claudio Gentile, Fabrizio Lacalandra (2009). "**Tighter approximated MILP formulations for unit commitment problems**". In: *IEEE* 24. URL: https://ieeexplore.ieee.org/document/4682641.

Atlas, NVE (2018). ***Hydropower data***. URL: https://atlas.nve.no/Html5Viewer/index.html?viewer=nveatlas#.

Baorong Zhou Shuhua Liu, Siyu Lu1 Xiaoyu Cao Wenmeng Zhao (2017). "**Cost–benefit analysis of pumped hydro storage using improved probabilistic production simulation method**". In: *The Journal of Engineering*, pp. 1–5. URL: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8311284.

Choong, Shi-Mei and A. El-Shafie (2015). "**State-of-the-Art for Modelling Reservoir Inflows and Management Optimization**". In: *Water Resour Manag* 29. URL: https://link.springer.com/article/10.1007/s11269-014-0872-z.

Company, National Pump (2018). ***SUBMERSIBLE PUMP SELECTION***. URL: https://www.nationalpumpcompany.com/wp-content/uploads/2016/12/sub-selection.pdf.

E. Solvang, A. Hartby Å. Killingtveit (2012). "**Impacts of pumped storage hydropower on the ecosystem of reservoirs**". In: *nina.no* 27. URL: https://www.cedren.no/Portals/Cedren/Pdf/HydroBalance/6_Sundt-HansenL_Environmental\%20impacts\%20on\%20reservoirs.pdf?ver=2012-10-05-100419-213.

Ehsan Goodarzi Mina Ziaei, Edward Zia Hosseinipour (2014). ***Introduction to Optimization Analysis in Hydrosystem Engineering***. Vol. 25, pp. 01–265. URL: https://link.springer.com/chapter/10.1007%2F978-3-319-04400-2_1.

Eivind Solvang Atle Harby, Ånund Killingtveit (2012). "**Increasing balance power capacity in Norwegian hydroelectric power stations**". In: *Sintef* 1, pp. 01–86. URL: https://www.cedren.no/Portals/Cedren/Scenario/Increasing\%20balance\%20power\%20capacity-sintef\%20report\%202011.pdf.

enel (2014). "**The Benefits of Pumped-Storage Hydropower**". In: *Innovation and Sustainability* 01. URL: https://www.enel.com/media/news/d/2014/01/the-benefits-of-pumped-storage-hydropower.

Energi, Agder (2019). ***Power Stations***. URL: https://www.ae.no/en/operations/hydropower/power-stations/.

Energy facts, Norway (2019). "**The Power Market**". In: *MARKET-BASED POWER SYSTEM* 01. URL: https://energifaktanorge.no/en/norsk-energiforsyning/kraftmarkedet/.

Fikru Fentaw Abera Dereje Hailu Asfaw, Agizew Nigussie Engida and Assefa M. Melesse (2018). "**Optimal Operation of Hydropower Reservoirs under Climate Change: The Case of Tekeze Reservoir, Eastern Nile**". In: *Water* 273, pp. 46–58. URL: https://www.mdpi.com/2073-4441/10/3/273.

Groot Wina Crijns-Graus, Robert Harmsen Mats de (2017). "**The effects of variable renewable electricity on energy efficiency and full load hours of fossil-fired power plants in the European Union**". In: *Energy*, pp. 576–586. URL: https://www.researchgate.net/publication/301214060_Design_of_Future_Pumped_Storage_Hydropower_in_Norway.

Halfpap, Richard (2000). "**Selecting an appropriate pump motor**". In: *Pump selction tutorial*. URL: https://www.machinedesign.com/motorsdrives/tutorial-selecting-appropriate-pump-motor.

Han, S. P. (1977). "**A Globally Convergent Method for Nonlinear Programming**". In: *Journal of Optimization Theory and Applications* 22. URL: https://link.springer.com/article/10.1007/BF00932858.

(IFC), Mary Porter Peschka (2014). ***Hydroelectric Power***, pp. 03–115. URL: https://www.ifc.org/wps/wcm/connect/06b2df8047420bb4a4f7ec57143498e5/Hydropower_Report.pdf?MOD=AJPERES.

Inc, Lawrence Pumps (2018). "**Introduction on Pumps and selection**". In: *Pump categories*. URL: https://www.flowserve.com/en/products/pumps.

Ioannis Kougias, Sándor Szabó (2017). "**Pumped hydroelectric storage utilization assessment: Forerunner of renewable energy integration or Trojan horse?**" In: *Energy* 140, pp. 318–329. URL: https://doi.org/10.1016/j.energy.2017.08.106.

Jones, Walter V. (2013). "**Motor Selection Made Easy: Choosing the Right Motor for Centrifugal Pump Applications**". In: *IEEE Industry Applications Magazine* 19, pp. 36–45. URL: https://ieeexplore.ieee.org/document/6587280.

Julie Charmasson Michael Belsnes, Oddgeir Andersen Antti Eloranta Ingeborg Graabak Magnus Korpås Ingeborg Palm Helland Håkon Sundt Ove Wolfgang (2017). ***HydroBalance***. URL: https://www.sintef.no/globalassets/sintef-energi/arendalsuka/cedren_veikart_web.pdf.

Kaltenekker, Bela (2014). "**Making your own Hydroelectric Power**". In: *Alternative Energy*, pp. 1–1. URL: http://www.altenergy.org/renewables/producing-hydro2.html.

— (2018). "**Renewable Energy**". In: *Review of Alternative energy*. URL: http://www.altenergy.org/renewables/producing-hydro2.html.

Kjersti, Liv Mari Hatlen (2015). ***ENERGY AND WATER RESOURCES IN NORWAY***. Vol. 07, pp. 01–83. URL: https://www.regjeringen.no/contentassets/fd89d9e2c39a4ac2b9c9a95bf156089a/facts_2015_energy_and_water_web.pdf.

Kraft, Otra (2019). "**Otra Kraft**". In: *Otra* 01. URL: http://otrakraft.no/om-oss/.

M. V. F. Pereira, L. MVG. Pinto (1991). ***"Multi-stage stochastic optimization applied to energy planning,"*** vol. 52, pp. 357–376. URL: https://www.semanticscholar.org/paper/The-operation-optimization-model-of-pumped-hydro-on-Liang-Li/f532581fcc949b3e248225d6e613d43500ed8272.

M. Y. Wang H. P. Xie, Y. S. Ni (1999). ***Optimal scheduling of pumped storage units and the benefit analysis***. Vol. 11, pp. 39–44.

maps, Google (2019). ***Map***. URL: https://www.google.com/maps.

McGraw-Hill (2008). ***Thermodynamics: An Engineering Approach - chapter5***. Vol. 6, pp. 01–09. URL: https://web.statler.wvu.edu/~wu/mae320/5-control-volume.pdf.

Mosonyi, Emil (2016). ***PRINCIPLES OF HYDROPOWER ENGINEERING***. Vol. 5, pp. 01–33. URL: https://nptel.ac.in/courses/105105110/pdf/m5l01.pdf.

Munson B. R. Young, D. F. and T. H. Okiishi (1998a). ***Fundamentals of Fluid Mechanics***. Vol. 3, pp. 020–079. URL: http://civilcafe.weebly.com/uploads/2/8/9/8/28985467/fluid_mechanics.pdf.

— (1998b). ***Fundamentals of Fluid Mechanics***. Vol. 3, pp. 020–079. URL: http://civilcafe.weebly.com/uploads/2/8/9/8/28985467/fluid_mechanics.pdf.

Muste, Marian and Ton Hoitink (2017). "**The Oxford Research Encyclopedia of Natural Hazard Science**". In: *Measuring Flood Discharge* 01. URL: http://oxfordre.com/naturalhazardscience/view/10.1093/acrefore/9780199389407.001.0001/acrefore-9780199389407-e-121.

NEVINA (2019). ***NEVINA Nedbørfelt-Vannføring-INdeks-Analyse***. URL: http://nevina.nve.no.

Nordegio (2008). "**Nordics top world energy consumption**". In: *World energy*, pp. 1–2. URL: http://archive.nordregio.se/en/Metameny/About-Nordregio/Journal-of-Nordregio/Journal-of-Nordregio-2010/Journal-of-Nordregio-no-3-2010/Nordics-top-wor/index.html.

NORDPOOL (2018). ***ElectricityPrice***. URL: https://www.nordpoolgroup.com/Market-data1/Dayahead/Area-Prices/ALL1/Hourly/?view=table.

NVE (2013). "**Production and consumption of electric energy.**" In: *ENERGY IN NORWAY* 01. URL: https://www.nve.no/Media/5147/folde2014.pdf.

online (1977). ***Limitations of Dynamic Programming***. URL: https://www8.cs.umu.se/kurser/TDBA77/VT06/algorithms/BOOK/BOOK2/NODE49.HTM.

Powell, M.J.D (1978). "**A Fast Algorithm for Nonlineary Constrained Optimization Calculations**". In: *Nonlinear Programming* 22. URL: https://link.springer.com/chapter/10.1007%2FBFb0067703.

Prewitt, K. (2001). "**Darwinian Theory**". In: *International Encyclopedia of the Social Behavioral Sciences* 01. URL: https://www.sciencedirect.com/topics/computer-science/darwinian-theory.

Price, Ken (1997). "**Differential evolution (DE) algorithm** ". In: *an algorithm by Kenneth Price and Rainer Storn* 01. URL: http://www.icsi.berkeley.edu/~storn/code.html.

Pumps, Goulds (Goulds online). ***Goulds Pumps***. URL: https://https://www.gouldspumps.com/en-US/Products/VIT/.

Sharma (2018). "**Design and optimization of pumped storage hydropower system**". In: *Hydropower* 01. URL: https://www.researchgate.net/profile/Rohit_Sharma125.

Statkraft (2018). "Hydropower". In: *Hydropower*, pp. 01–02. URL: https://www.statkraft.com/globalassets/old-contains-the-old-folder-structure/documents/hydropower-09-eng_tcm9-4572.pdf.

— (online). ***The glaciers' inner energy***. URL: https://www.statkraft.com/en-GB/energy-sources/hydropower/the-glaciers-inner-energy/.

Suganthan, Dr. P. N. (2018). "**Numerical optimization using differential evolution**". In: *Institute of Mathematical sciences* 01. URL: https://dr.ntu.edu.sg/handle/10220/48011.

Warren B. Powell, Belgacem Bouzaiene-Ayari (2012). "**Strategic,tacticalandreal-time planning of locomotives at Norfolk Southerm using approximate dynamic programming**". In: *2012 Joint Rail Conference* 22. URL: https://www.researchgate.

net/publication/267648560_Strategic_Tactical_and_Real-Time_Planning_of_Locomotives_at_Norfolk_Southern_Using_Approximate_Dynamic_Programming.

Whiticar, Dr. Michael (2016). "**Large Hydro**". In: *Energy BC* 273. URL: http://energybc.ca/largehydro.html.

Wikipedia (2019a). "**Environmental impact of reservoirs**". In: *online* 01. URL: https://en.wikipedia.org/wiki/Environmental_impact_of_reservoirs#Flood_control.

— (2019b). ***Nord Pool AS***. URL: https://en.wikipedia.org/wiki/Nord_Pool_AS.

Appendix A

Syntax (Matlab)

A.1 Computations

```
% compute the reservoir volumes based on the levels and the area curves
volume_Botsvatn=volume_percentage_Botsvatn*Total_volume_Botsvatn/100;
%function to pass reservoir level to volume
volume_Urevatn=volume_percentage_Urevatn*Total_volume_Urevatn/100;
%function to pass reservoir level to volume
volume_Vatnedalsvatn=volume_percentage_Vatnedalsvatn*...
    Total_volume_Vatnedalsvatn/100;
%function to pass reservoir level to volume

nHours=numel(level_Botsvatn); %number of hours
Time=(1:nHours)'; %vector of time

% Mass balance at Botsvatn
for i=1:length(volume_Botsvatn)-1
    flow_Botsvatn_catchment(i)=(volume_Botsvatn(i+1)-volume_Botsvatn(i))/..
        ((Time(i+1)-Time(i))*60*60)-flow_Holen_G1_G2(i)-flow_Holen_G3(i)...
        +flow_Brokke(i);
end

% Mass balance at Urevatn
for i=1:length(volume_Urevatn)-1
    flow_Urevatn_catchment(i)=(volume_Urevatn(i+1)-volume_Urevatn(i))/...
        ((Time(i+1)-Time(i))*60*60)+flow_Holen_G3(i);
end

% Mass balance at Vatnedalsvatn
for i=1:length(volume_Vatnedalsvatn)-1
    flow_Vatnedalsvatn_catchment(i)=(volume_Vatnedalsvatn(i+1)-...
        volume_Vatnedalsvatn(i))/((Time(i+1)-Time(i))*60*60)+...
        flow_Holen_G1_G2(i);
end
```

FIGURE A.1: Volumes and catchment

```
% removing all the 0 levels from the BOTSVATN (Brokke) record replacing ..
%them by the previous
% value (NOTE: this can give problems if the first level is 0)
n_zeros_Botsvatn=0; % variable just to know how many 0 there were...
%in the record
for i=1:length(data_Brokke(:,1))
    if (data_Brokke(i,1)==0||data_Brokke(i,2)==0)
        level_Botsvatn(i)=level_Botsvatn(i-1);
        %if the level is 0 it takes the previous level
        volume_percentage_Botsvatn(i)=volume_percentage_Botsvatn(i-1);
        %if the level is 0 it takes the previous level
        n_zeros_Botsvatn=n_zeros_Botsvatn+1;
    else
        level_Botsvatn(i)=data_Brokke(i,1);
        volume_percentage_Botsvatn(i)=data_Brokke(i,2);
    end
end
level_Botsvatn=level_Botsvatn';
volume_percentage_Botsvatn=volume_percentage_Botsvatn';

% removing all the 0 levels from the Vatnedalsvatn (Holen G1-G2)...
%record replacing them by the previous value
n_zeros_Vatnedalsvatn=0;
% variable just to know how many 0 there were in the record
for i=1:length(data_Holen_G1_G2(:,1))
    if (data_Holen_G1_G2(i,1)==0||data_Holen_G1_G2(i,2)==0)
        level_Vatnedalsvatn(i)=level_Vatnedalsvatn(i-1);
        %if the level is 0 it takes the previous level
        volume_percentage_Vatnedalsvatn(i)=...
            volume_percentage_Vatnedalsvatn(i-1);
```

FIGURE A.2: level data

```
            volume_percentage_Vatnedalsvatn(i-1);
        %if the level is 0 it takes the previous level
        n_zeros_Vatnedalsvatn=n_zeros_Vatnedalsvatn+1;
    else
        level_Vatnedalsvatn(i)=data_Holen_G1_G2(i,1);
        volume_percentage_Vatnedalsvatn(i)=data_Holen_G1_G2(i,2);
    end
end
level_Vatnedalsvatn=level_Vatnedalsvatn';
volume_percentage_Vatnedalsvatn=volume_percentage_Vatnedalsvatn';

% removing all the 0 levels from the UREVATN (Holen G3) record...
%replacing them by the previous
% value
n_zeros_Urevatn=0;
% variable just to know how many 0 there were in the record
for i=1:length(data_Holen_G3(:,1))
    if (data_Holen_G3(i,1)==0||data_Holen_G3(i,2)==0)
        level_Urevatn(i)=level_Urevatn(i-1);
        %if the level is 0 it takes the previous level
        volume_percentage_Urevatn(i)=volume_percentage_Urevatn(i-1);
        %if the level is 0 it takes the previous level
        n_zeros_Urevatn=n_zeros_Urevatn+1;
    else
        level_Urevatn(i)=data_Holen_G3(i,1);
        volume_percentage_Urevatn(i)=data_Holen_G3(i,2);
    end
end
level_Urevatn=level_Urevatn';
volume_percentage_Urevatn=volume_percentage_Urevatn';
```

FIGURE A.3: level data

```
% 2nd order polynomial fit to have the reservoir curves
[reservoir_curve_Botsvatn, godfits_Botsvatn] = ...
    fit(volume_percentage_Botsvatn, level_Botsvatn, 'poly2');

coeff_reservoir_curve_Botsvatn=coeffvalues(reservoir_curve_Botsvatn);

[reservoir_curve_Vatnedalsvatn, godfits_Vatnedalsvatn] =...
    fit(volume_percentage_Vatnedalsvatn, level_Vatnedalsvatn, 'poly2');

coeff_reservoir_curve_Vatnedalsvatn=...
    coeffvalues(reservoir_curve_Vatnedalsvatn);

[reservoir_curve_Urevatn, godfits_Urevatn] = ...
    fit(volume_percentage_Urevatn, level_Urevatn, 'poly2');

coeff_reservoir_curve_Urevatn=coeffvalues(reservoir_curve_Urevatn);

% figure with volume vs. level BOTSVATN
figure(1)
plot(volume_percentage_Botsvatn,level_Botsvatn,'bo')
hold on
plot([0:1:100],feval(reservoir_curve_Botsvatn,[0:1:100]),'k-')
%plots the 2nd order fitted polynomial
plot([0 100],[HRV_Botsvatn HRV_Botsvatn],'r--') %plots the maximum level
plot([0 100],[LRV_Botsvatn LRV_Botsvatn],'r--') %plots the minimum level
hold off
title('BOTSVATN')
legend('data',strcat('fit curve: Z = ',num2str...
    (coeff_reservoir_curve_Botsvatn(1)),'V^2 + ',...
    num2str(coeff_reservoir_curve_Botsvatn(2)),...
    'V + ',num2str(coeff_reservoir_curve_Botsvatn(3)),...
```

FIGURE A.4: Polynomial fit

```
%--------------------------LEVEL & VOLUME------------------------
% making separate data for daily mean value
level_Botsvatn_avg = arrayfun(@(i)...
    mean(level_Botsvatn(i:i+n-1)),...
    1:n:length(level_Botsvatn)-n+1).';
volume_percentage_Botsvatn_avg = arrayfun(@(i)...
    mean(volume_percentage_Botsvatn(i:i+n-1)),...
    1:n:length(volume_percentage_Botsvatn)-n+1).';

level_Vatnedalsvatn_avg = arrayfun(@(i)...
    mean(level_Vatnedalsvatn(i:i+n-1)),...
    1:n:length(level_Vatnedalsvatn)-n+1).';
volume_percentage_Vatnedalsvatn_avg = arrayfun(@(i)...
    mean(volume_percentage_Vatnedalsvatn(i:i+n-1)),...
    1:n:length(volume_percentage_Vatnedalsvatn)-n+1).';

level_Urevatn_avg = arrayfun(@(i)...
    mean(level_Urevatn(i:i+n-1)),...
    1:n:length(level_Urevatn)-n+1).';
volume_percentage_Urevatn_avg = arrayfun(@(i)...
    mean(volume_percentage_Urevatn(i:i+n-1)),...
    1:n:length(volume_percentage_Urevatn)-n+1).';
```

FIGURE A.5: Daily average

```
%%
nDays=numel(level_Botsvatn_avg); %number of days
Time=(1:nDays)'; %vector of time
% 2nd order polynomial fit to have the average reservoir curves
[reservoir_curve_Botsvatn_avg, godfits_Botsvatn_avg] =...
    fit(volume_percentage_Botsvatn_avg, level_Botsvatn_avg, 'poly2');
coeff_reservoir_curve_Botsvatn_avg=coeffvalues...
    (reservoir_curve_Botsvatn_avg);
[reservoir_curve_Vatnedalsvatn_avg, godfits_Vatnedalsvatn_avg] =...
    fit(volume_percentage_Vatnedalsvatn_avg, level_Vatnedalsvatn_avg,...
    'poly2');
coeff_reservoir_curve_Vatnedalsvatn_avg=coeffvalues...
    (reservoir_curve_Vatnedalsvatn_avg);
[reservoir_curve_Urevatn_avg, godfits_Urevatn_avg] =...
    fit(volume_percentage_Urevatn_avg, level_Urevatn_avg, 'poly2');
coeff_reservoir_curve_Urevatn_avg=coeffvalues...
    (reservoir_curve_Urevatn_avg);
```

FIGURE A.6: Polynomial fit: average

A.2 Differential Evolution

Following differential evolution script is performed by using Price, 1997 as reference.

```
for i=1:NP
    pop(i,:) = XVmin + rand(1,D).*(XVmax - XVmin);
end

popold     = zeros(size(pop));      % toggle population
val        = zeros(1,NP);           % create and reset the "cost array"
bestmem    = zeros(1,D);            % best population member ever
bestmemit  = zeros(1,D);            % best population member in iteration
nfeval     = 0;                     % number of function evaluations

%------Evaluate the best member after initialization----------------------

ibest    = 1;                       % start with first population member
val(1)   = feval(fname,pop(ibest,:),y);
bestval = val(1);                   % best objective function value so far
nfeval   = nfeval + 1;
for i=2:NP                          % check the remaining members
  val(i) = feval(fname,pop(i,:),y);
  nfeval   = nfeval + 1;
  if (val(i) < bestval)             % if member is better
     ibest    = i;                  % save its location
     bestval = val(i);
  end
end
bestmemit = pop(ibest,:);           % best member of current iteration
bestvalit = bestval;                % best value of current iteration

bestmem = bestmemit;                % best member ever
```

FIGURE A.7: Syntax 1

A.2.1 Rotation

```
while ((iter < itermax) & (bestval > VTR))
  popold = pop;                     % save the old population

  ind = randperm(4);                % index pointer array

  a1  = randperm(NP);               % shuffle locations of vectors
  rt = rem(rot+ind(1),NP);          % rotate indices by ind(1) positions
  a2  = a1(rt+1);                   % rotate vector locations
  rt = rem(rot+ind(2),NP);
  a3  = a2(rt+1);
  rt = rem(rot+ind(3),NP);
  a4  = a3(rt+1);
  rt = rem(rot+ind(4),NP);
  a5  = a4(rt+1);

  pm1 = popold(a1,:);               % shuffled population 1
  pm2 = popold(a2,:);               % shuffled population 2
  pm3 = popold(a3,:);               % shuffled population 3
  pm4 = popold(a4,:);               % shuffled population 4
  pm5 = popold(a5,:);               % shuffled population 5

  for i=1:NP                        % population filled with the best member
    bm(i,:) = bestmemit;            % of the last iteration
  end

  mui = rand(NP,D) < CR;            % all random numbers < CR are 1, 0 otherwise
```

FIGURE A.8: Syntax 2

A.2.2 Crossover

```
if (strategy > 5)
  st = strategy-5;          % binomial crossover
else
  st = strategy;        % exponential crossover
  mui=sort(mui');           % transpose, collect 1's in each column
  for i=1:NP
    n=floor(rand*D);
    if n > 0
       rtd = rem(rotd+n,D);
       mui(:,i) = mui(rtd+1,i); %rotate column i by n
    end
  end
  mui = mui';           % transpose back
end
mpo = mui < 0.5;                % inverse mask to mui

if (st == 1)                     % DE/best/1
  ui = bm + F*(pm1 - pm2);       % differential variation
  ui = popold.*mpo + ui.*mui;    % crossover
elseif (st == 2)                 % DE/rand/1
  ui = pm3 + F*(pm1 - pm2);      % differential variation
  ui = popold.*mpo + ui.*mui;    % crossover
elseif (st == 3)                 % DE/rand-to-best/1
  ui = popold + F*(bm-popold) + F*(pm1 - pm2);
  ui = popold.*mpo + ui.*mui;    % crossover
elseif (st == 4)                 % DE/best/2
  ui = bm + F*(pm1 - pm2 + pm3 - pm4);  % differential variation
  ui = popold.*mpo + ui.*mui;           % crossover
elseif (st == 5)                 % DE/rand/2
  ui = pm5 + F*(pm1 - pm2 + pm3 - pm4);  % differential variation
  ui = popold.*mpo + ui.*mui;            % crossover
```

FIGURE A.9: Syntax 3

A.2.3 Selection

```
%-----Select which vectors are allowed to enter the new population------------
for i=1:NP
  tempval = feval(fname,ui(i,:),y);   % check cost of competitor
  nfeval  = nfeval + 1;
  if (tempval <= val(i))  % if competitor is better than value in "cost array"
     pop(i,:) = ui(i,:);  % replace old vector with new one (for new iteration)
     val(i)   = tempval;  % save value in "cost array"

     %----we update bestval only in case of success to save time-----------
     if (tempval < bestval)     % if competitor better than the best one ever
        bestval = tempval;      % new best value
        bestmem = ui(i,:);      % new best parameter vector ever
     end
  end
end %---end for imember=1:NP

bestmemit = bestmem;       % freeze the best member of this iteration for the coming
                           % iteration. This is needed for some of the strategies.
```

FIGURE A.10: Syntax 4

A.2.4 Syntax - computation and DE implementation

```
% 2nd order polynomial fit to have the reservoir curves
[reservoir_curve_Botsvatn, godfits_Botsvatn] = ...
    fit(volume_percentage_Botsvatn, level_Botsvatn, 'poly2');

coeff_reservoir_curve_Botsvatn=coeffvalues(reservoir_curve_Botsvatn);

[reservoir_curve_Vatnedalsvatn, godfits_Vatnedalsvatn] =...
    fit(volume_percentage_Vatnedalsvatn, level_Vatnedalsvatn, 'poly2');

coeff_reservoir_curve_Vatnedalsvatn=...
    coeffvalues(reservoir_curve_Vatnedalsvatn);

[reservoir_curve_Urevatn, godfits_Urevatn] = ...
    fit(volume_percentage_Urevatn, level_Urevatn, 'poly2');

coeff_reservoir_curve_Urevatn=coeffvalues(reservoir_curve_Urevatn);
```

FIGURE A.11: Second order polynomial fit syntax

```
% Mass balance at Botsvatn
for i=1:length(volume_Botsvatn)-1
    flow_Botsvatn_catchment(i)=(volume_Botsvatn(i+1)-volume_Botsvatn(i))/...
        ((Time(i+1)-Time(i))*60*60)-flow_Holen_G1_G2(i)-flow_Holen_G3(i)...
        +flow_Brokke(i);
end

% Mass balance at Urevatn
for i=1:length(volume_Urevatn)-1
    flow_Urevatn_catchment(i)=(volume_Urevatn(i+1)-volume_Urevatn(i))/...
        ((Time(i+1)-Time(i))*60*60)+flow_Holen_G3(i);
end

% Mass balance at Vatnedalsvatn
for i=1:length(volume_Vatnedalsvatn)-1
    flow_Vatnedalsvatn_catchment(i)=(volume_Vatnedalsvatn(i+1)-...
        volume_Vatnedalsvatn(i))/((Time(i+1)-Time(i))*60*60)+...
        flow_Holen_G1_G2(i);
end
```

FIGURE A.12: Mass balance syntax

```
inflow1 = flow_Botsvatn_catchment_avg(1:n_days).';
%inflow from catchment into Botsvatn
inflow2 = flow_Vatnedalsvatn_catchment_avg(1:n_days).';
%inflow from catchment into Vatnedalsvatn
inflow3 = flow_Urevatn_catchment_avg(1:n_days).';
%inflow from catchment into Urevatn
volume1_initial=volume_Botsvatn_avg(1);
volume2_initial=volume_Vatnedalsvatn_avg(1);
volume3_initial=volume_Urevatn_avg(1);
c_Res1_avg = coeff_reservoir_curve_Botsvatn_avg;   % constants
c_Res2_avg = coeff_reservoir_curve_Vatnedalsvatn_avg;   % constants
c_Res3_avg = coeff_reservoir_curve_Urevatn_avg;   % constants
Total_volume1=Total_volume_Botsvatn;
% This value is taken equivalent to hourly data
Total_volume2=Total_volume_Vatnedalsvatn;
Total_volume3=Total_volume_Urevatn;
c_Pow1_avg = coeff_power_curve_Brokke_avg;   % constants
c_Pow2_avg = coeff_power_curve_Holen_G1_G2_avg;   % constants
c_Pow3_avg = coeff_power_curve_Holen_G3_avg;   % constants
priceKris_avg=price_avg(1:n_days).';
pricePump_avg = 0.60*priceKris_avg;
% price for pumping is 1% more that the turbining
```

FIGURE A.13: Symbolize syntax

```
% constraints with bounds
XVmin = zeros(1,D); %min hourly flow at brokke,holen G1_G2, holen_G3, pump
XVmax = [max(flow_Brokke_avg)*ones(1,D/4) max(flow_Holen_G1_G2_avg)*...
    ones(1,D/4) max(flow_Holen_G3_avg)*ones(1,D/4) flow_Pump*ones(1,D/4)];
%max hourly flow at brokke,holen G1_G2, holen_G3, pump

% XVmin,XVmax   vector of lower and bounds of initial population
y=[]; %coefficients or parameters in the objective function
NP = 10*D; %Number of population members
VTR = -1.e20; %tolerance error on the objective function value
strategy = 7; % means DE/rand/1/bin.
refresh = 100; %intermediate output will be produced after "refresh"...
%iterations. No intermediate output will be produced...
%if refresh is < 1.

itermax = 1000; % maximum number of iterations (generations)

F = 0.1; %DE-stepsize F ex [0, 2]

CR = 0.5; %crossover probabililty constant ex [0, 1]
```

FIGURE A.14: Bound constraint syntax

```
for ii=1:D/4
    level1(ii)=c_Res1_avg(1)*(volume1(ii)/Total_volume1*100).^2+...
        c_Res1_avg(2)*(volume1(ii)/Total_volume1*100)+...
        c_Res1_avg(3);
    level2(ii)=c_Res2_avg(1)*(volume2(ii)/Total_volume2*100).^2+...
        c_Res2_avg(2)*(volume2(ii)/Total_volume2*100)+...
        c_Res2_avg(3);
    level3(ii)=c_Res3_avg(1)*(volume3(ii)/Total_volume3*100).^2+...
        c_Res3_avg(2)*(volume3(ii)/Total_volume3*100)+...
        c_Res3_avg(3);
end
% Power (regression curves)
Power1=c_Pow1_avg(1)*(pop(ibest,1:D/4).*level1)+c_Pow1_avg(2);
Power2=c_Pow2_avg(1)*(pop(ibest,D/4+1:D/2).*level2)+c_Pow2_avg(2);
Power3=c_Pow3_avg(1)*(pop(ibest,D/2+1:3/4*D).*level3)+c_Pow3_avg(2);
Power4=rho*g*H*pop(ibest,3/4*D+1:D)/eta/1e6; %power in MW
%NOTE pop(ibest,1:D/4) refers to all hours of Q_Brokke
%     pop(ibest,D/4+1:D/2) refers to all hours of Q_Holen_G1_G2
%     pop(ibest,D/2+1:3/4*D) refers to all hours of Q_Holen_G3
%     pop(ibest,3/4*D+1:D) refers to all hours of pump
```

FIGURE A.15: Level and power curves syntax

A.3 Conditions and bounds implementation

```
%computation of the reservoir volumes
volume1=[];
volume1(1) = volume1_initial; %initial volume Botsvatn
volume2=[];
volume2(1) = volume2_initial; %initial volume Vatnedalsvatn
volume3=[];
volume3(1) = volume3_initial; %initial volume Urevatn
volume3_spill=0;

for ii = 2:D/4
    volume1(ii) = volume1(ii-1) + (inflow1(ii)-(pop(ibest,ii)-...
        pop(ibest,D/4+ii)-pop(ibest,D/2+ii)+pop...
        (ibest,3/4*D+ii)))*interval;
    if volume1(ii)>Total_volume_Botsvatn
        volume1_spill=volume1(ii)-Total_volume_Botsvatn;
        volume1(ii)=Total_volume_Botsvatn;
    end
     if pop(ibest,D/2+ii)>0

        pop(ibest,3/4*D+ii)=0;
    else
        pop(ibest,3/4*D+ii)=20;

    end
    volume3(ii) = volume3(ii-1) + (inflow3(ii)-(pop(ibest,D/2+ii)...
        -pop(ibest,3/4*D+ii)))*interval;
    if volume3(ii)>Total_volume_Urevatn
        volume3_spill=volume3(ii)-Total_volume_Urevatn;
        volume3(ii)=Total_volume_Urevatn;
    end
```

FIGURE A.16: Script A

```
    if volume3(ii)<=min(volume_Urevatn_avg)
        pop(ibest,3/4*D+ii)=max(pop(ibest,3/4*D+ii));
        pop(ibest,D/2+ii)=0;
    end
     if pop(ibest,D/2+ii)>0

        pop(ibest,3/4*D+ii)=0;
    else
        pop(ibest,3/4*D+ii)=20;

    end
    volume2(ii) = volume2(ii-1) + volume3_spill+(inflow2(ii)...
        -pop(ibest,D/4+ii))*interval;
    if volume2(ii)>Total_volume_Vatnedalsvatn
        volume2_spill=volume2(ii)-Total_volume_Vatnedalsvatn;
        volume2(ii)=Total_volume_Vatnedalsvatn;
    end
    %NOTE pop(ibest,ii) refers to ii hour of Q_Brokke
    %     pop(ibest,D/4+ii) refers ii hour of Q_Holen_G1_G2
    %     pop(ibest,D/2+ii) refers ii hour of Q_Holen_G3
    %     pop(ibest,3/4*D+ii) refers ii hour of pump
end
```

FIGURE A.17: Script B

```
level1=[];
level2=[];
level3=[];
for ii=1:D/4
    level1(ii)=c_Res1_avg(1)*(volume1(ii)/Total_volume1*100).^2+...
        c_Res1_avg(2)*(volume1(ii)/Total_volume1*100)+...
        c_Res1_avg(3);
    level2(ii)=c_Res2_avg(1)*(volume2(ii)/Total_volume2*100).^2+...
        c_Res2_avg(2)*(volume2(ii)/Total_volume2*100)+...
        c_Res2_avg(3);
    level3(ii)=c_Res3_avg(1)*(volume3(ii)/Total_volume3*100).^2+...
        c_Res3_avg(2)*(volume3(ii)/Total_volume3*100)+...
        c_Res3_avg(3);
end
% Power (regression curves)
Power1=c_Pow1_avg(1)*(pop(ibest,1:D/4).*level1)+c_Pow1_avg(2);
Power2=c_Pow2_avg(1)*(pop(ibest,D/4+1:D/2).*level2)+c_Pow2_avg(2);
Power3=c_Pow3_avg(1)*(pop(ibest,D/2+1:3/4*D).*level3)+c_Pow3_avg(2);
Power4=rho*g*H*pop(ibest,3/4*D+1:D)/eta/1e6; %power in MW
%NOTE pop(ibest,1:D/4) refers to all hours of Q_Brokke
%     pop(ibest,D/4+1:D/2) refers to all hours of Q_Holen_G1_G2
%     pop(ibest,D/2+1:3/4*D) refers to all hours of Q_Holen_G3
%     pop(ibest,3/4*D+1:D) refers to all hours of pump


val(1)  = sum(Power4.*pricePump_avg-(Power1+Power2+Power3)...
    .*priceKris_avg); %objective function

bestval = val(1);                  % best objective function value so far

nfeval  = nfeval + 1;
```

FIGURE A.18: Script C

```
for i=2:NP                              % check the remaining members
    %computation of the reservoir volumes
    % volume1 = cell(D/4,1); %hourly volume at Botsvatn
    % volume2 = cell(D/4,1); %hourly volume at Vatnedalsvatn
    % volume3 = cell(D/4,1); %hourly volume at Urevatn
    % volume1{1} = volume1_initial; %initial volume Botsvatn
    % volume2{1} = volume2_initial; %initial volume Vatnedalsvatn
    % volume3{1} = volume3_initial; %initial volume Urevatn

    for ii = 2:D/4
        volume1(ii) = volume1(ii-1) + (inflow1(ii)-(pop(i,ii)-...
            pop(i,D/4+ii)-pop(i,D/2+ii)+pop(i,3/4*D+ii)))*interval;
        if volume1(ii)>Total_volume_Botsvatn
            volume1_spill=volume1(ii)-Total_volume_Botsvatn;
            volume1(ii)=Total_volume_Botsvatn;
        end
          if pop(i,D/2+ii)>0

            pop(i,3/4*D+ii)=0;
        else
            pop(i,3/4*D+ii)=20;

        end
        volume3(ii) = volume3(ii-1) + (inflow3(ii)-(pop(i,D/2+ii)-...
            pop(i,3/4*D+ii)))*interval;
        if volume3(ii)>Total_volume_Urevatn
            volume3_spill=volume3(ii)-Total_volume_Urevatn;
            volume3(ii)=Total_volume_Urevatn;
        end
```

FIGURE A.19: Script C

```
        if volume3(ii)<=min(volume_Urevatn_avg)
            pop(i,3/4*D+ii)=max(pop(i,3/4*D+ii));
            pop(i,D/2+ii)=0;
        end
          if pop(i,D/2+ii)>0

            pop(i,3/4*D+ii)=0;
        else
            pop(i,3/4*D+ii)=20;

        end

        volume2(ii) = volume2(ii-1) + volume3_spill +(inflow2(ii)...
            -pop(i,D/4+ii))*interval;
        if volume2(ii)>Total_volume_Vatnedalsvatn
            volume2_spill=volume2(ii)-Total_volume_Vatnedalsvatn;
            volume2(ii)=Total_volume_Vatnedalsvatn;
        end

    end
    % reservoir levels (regression curves)
    % level1 = cell(D/4,1);
    % level2 = cell(D/4,1);
    % level3 = cell(D/4,1);
```

FIGURE A.20: Script D

```
    for ii=1:D/4
        level1(ii)=c_Res1_avg(1)*(volume1(ii)/Total_volume1*100).^2+...
            c_Res1_avg(2)*(volume1(ii)/Total_volume1*100)+...
            c_Res1_avg(3);
        level2(ii)=c_Res2_avg(1)*(volume2(ii)/Total_volume2*100).^2+...
            c_Res2_avg(2)*(volume2(ii)/Total_volume2*100)+...
            c_Res2_avg(3);
        level3(ii)=c_Res3_avg(1)*(volume3(ii)/Total_volume3*100).^2+...
            c_Res3_avg(2)*(volume3(ii)/Total_volume3*100)+...
            c_Res3_avg(3);
    end
    % Power (regression curves)
    Power1=c_Pow1_avg(1)*(pop(i,1:D/4).*level1)+c_Pow1_avg(2);
    Power2=c_Pow2_avg(1)*(pop(i,D/4+1:D/2).*level2)+c_Pow2_avg(2);
    Power3=c_Pow3_avg(1)*(pop(i,D/2+1:3/4*D).*level3)+c_Pow3_avg(2);
    Power4=rho*g*H*pop(i,3/4*D+1:D)/eta/1e6; %power in MW
    %NOTE pop(i,1:D/4) refers to all hours of Q_Brokke
    %     pop(i,D/4+1:D/2) refers to all hours of Q_Holen_G1_G2
    %     pop(i,D/2+1:3/4*D) refers to all hours of Q_Holen_G3
    %     pop(i,3/4*D+1:D) refers to all hours of pump
    val(i)  = sum(Power4.*pricePump_avg-(Power1+Power2+Power3).*...
        priceKris_avg);  %objective function

    nfeval  = nfeval + 1;
    if (val(i) < bestval)           % if member is better
        ibest   = i;                 % save its location
        bestval = val(i);
    end
end
```

FIGURE A.21: Script E

```
bestmemit = pop(ibest,:);          % best member of current iteration
bestvalit = bestval;               % best value of current iteration

bestmem = bestmemit;               % best member ever
%%
%------DE-Minimization---------------------------------------------
%------popold is the population which has to compete. It is--------
%------static through one iteration. pop is the newly--------------
%------emerging population.

pm1 = zeros(NP,D);                 % initialize population matrix 1
pm2 = zeros(NP,D);                 % initialize population matrix 2
pm3 = zeros(NP,D);                 % initialize population matrix 3
pm4 = zeros(NP,D);                 % initialize population matrix 4
pm5 = zeros(NP,D);                 % initialize population matrix 5
bm  = zeros(NP,D);                 % initialize bestmember  matrix
ui  = zeros(NP,D);                 % intermediate population of perturbed vec
mui = zeros(NP,D);                 % mask for intermediate population
mpo = zeros(NP,D);                 % mask for old population
rot = (0:1:NP-1);                  % rotating index array (size NP)
rotd= (0:1:D-1);                   % rotating index array (size D)
rt  = zeros(NP);                   % another rotating index array
rtd = zeros(D);                    % rotating index array for exponential cro
a1  = zeros(NP);                   % index array
a2  = zeros(NP);                   % index array
a3  = zeros(NP);                   % index array
a4  = zeros(NP);                   % index array
a5  = zeros(NP);                   % index array
ind = zeros(4);
```

FIGURE A.22: Script F

```
ui(find(flow_Brokke_avg==0))=0;
ui(ui<0)=0;
ui(ui(:,1:D/4)>max(flow_Brokke_avg))=max(flow_Brokke_avg);
for i=D/4+1:D/2
    for j=1:length(ui)
        if ui(j,i)>max(flow_Holen_G1_G2_avg)
            ui(j,i)=max(flow_Holen_G1_G2_avg);
        end
    end
end
for i=D/2+1:3/4*D
    for j=1:length(ui)
        if ui(j,i)>max(flow_Holen_G3_avg)
            ui(j,i)=max(flow_Holen_G3_avg);
        end
    end
end
for i=3/4*D+1:D
    for j=1:length(ui)
        if ui(j,i)>20
            ui(j,i)=20;
        end
    end
end
```

FIGURE A.23: Script G

```
%-----Selecting which vectors are allowed to enter the new population---
for i=1:NP
    % Evaluating............
    %computation of the reservoir volumes
    % volume1 = cell(D/4,1); %hourly volume at Botsvatn
    % volume2 = cell(D/4,1); %hourly volume at Vatnedalsvatn
    % volume3 = cell(D/4,1); %hourly volume at Urevatn
    % volume1{1} = volume1_initial; %initial volume Botsvatn
    % volume2{1} = volume2_initial; %initial volume Vatnedalsvatn
    % volume3{1} = volume3_initial; %initial volume Urevatn
    for ii = 2:D/4
        volume1(ii) = volume1(ii-1) + (inflow1(ii)-(ui(i,ii)...
            -ui(i,D/4+ii)-ui(i,D/2+ii)+ui(i,3/4*D+ii)))*interval;
        if volume1(ii)>Total_volume_Botsvatn
            volume1_spill=volume1(ii)-Total_volume_Botsvatn;
            volume1(ii)=Total_volume_Botsvatn;
        end
         if ui(i,D/2+ii)>0

            ui(i,3/4*D+ii)=0;
        else
            ui(i,3/4*D+ii)=20;

        end
        volume3(ii) = volume3(ii-1) + (inflow3(ii)-(ui(i,D/2+ii)...
            -ui(i,3/4*D+ii)))*interval;
        if volume3(ii)>Total_volume_Urevatn
            volume3_spill=volume3(ii)-Total_volume_Urevatn;
            volume3(ii)=Total_volume_Urevatn;
        end
```

FIGURE A.24: Script H

```
    if volume3(ii)<=min(volume_Urevatn_avg)
        ui(i,3/4*D+ii)=max(ui(i,3/4*D+ii));
        ui(i,D/2+ii)=0;
    end
     if ui(i,D/2+ii)>0

        ui(i,3/4*D+ii)=0;
    else
        ui(i,3/4*D+ii)=20;

    end
    volume2(ii) = volume2(ii-1) + volume3_spill+(inflow2(ii)-ui(i,D/4+
    if volume2(ii)>Total_volume_Vatnedalsvatn
        volume2_spill=volume2(ii)-Total_volume_Vatnedalsvatn;
        volume2(ii)=Total_volume_Vatnedalsvatn;
    end

  %NOTE pop(i,ii) refers to ii hour of Q_Brokke
    %     pop(i,D/4+ii) refers ii hour of Q_Holen_G1_G2
    %     pop(i,D/2+ii) refers ii hour of Q_Holen_G3
    %     pop(i,3/4*D+ii) refers ii hour of pump
end
% reservoir levels (regression curves)
% level1 = cell(D/4,1);
% level2 = cell(D/4,1);
% level3 = cell(D/4,1);
for ii=1:D/4
    level1(ii)=c_Res1_avg(1)*(volume1(ii)/Total_volume1*100).^2+...
        c_Res1_avg(2)*(volume1(ii)/Total_volume1*100)+...
        c_Res1_avg(3);
```

FIGURE A.25: Script I

```
for ii=1:D/4
    level1(ii)=c_Res1_avg(1)*(volume1(ii)/Total_volume1*100).^2+...
        c_Res1_avg(2)*(volume1(ii)/Total_volume1*100)+...
        c_Res1_avg(3);
    level2(ii)=c_Res2_avg(1)*(volume2(ii)/Total_volume2*100).^2+...
        c_Res2_avg(2)*(volume2(ii)/Total_volume2*100)+...
        c_Res2_avg(3);
    level3(ii)=c_Res3_avg(1)*(volume3(ii)/Total_volume3*100).^2+...
        c_Res3_avg(2)*(volume3(ii)/Total_volume3*100)+...
        c_Res3_avg(3);
end
% Power (regression curves)
Power1=c_Pow1_avg(1)*(ui(i,1:D/4).*level1)+c_Pow1_avg(2);
Power2=c_Pow2_avg(1)*(ui(i,D/4+1:D/2).*level2)+c_Pow2_avg(2);
Power3=c_Pow3_avg(1)*(ui(i,D/2+1:3/4*D).*level3)+c_Pow3_avg(2);
Power4=rho*g*H*ui(i,3/4*D+1:D)/eta/1e6; %power in MW
%NOTE pop(i,1:D/4) refers to all hours of Q_Brokke
%     pop(i,D/4+1:D/2) refers to all hours of Q_Holen_G1_G2
%     pop(i,D/2+1:3/4*D) refers to all hours of Q_Holen_G3
%     pop(i,3/4*D+1:D) refers to all hours of pump
tempval  = sum(Power4.*pricePump_avg-(Power1+Power2+Power3).*priceKris

nfeval   = nfeval + 1;
 if (tempval <= val(i))% if competitor is better than value in "cost a
  if (tempval <= val(i))&(max(level1)<=max(level_Botsvatn_avg(1:n_days
          (max(level2)<=max(level_Vatnedalsvatn_avg(1:n_days)))&(min(l
          (max(level3)<=max(level_Urevatn_avg(1:n_days)))&(min(volume3
    pop(i,:) = ui(i,:);  % replacing old vector with new one (for new
    val(i)   = tempval;  % save value in "cost array"
```

FIGURE A.26: Script J

```
        pop(i,:) = ui(i,:);  % replacing old vector with new one (for new
        val(i)   = tempval;  % save value in "cost array"

        %----we update bestval only in case of success to save time-------
        if (tempval < bestval)     % if competitor better than the best on
            bestval = tempval;      % new best value
            bestmem = ui(i,:);    % new best parameter vector ever
        end
    end
end %---end for imember=1:NP

bestmemit = bestmem;       % freeze the best member of this iteration for
% iteration. This is needed for some of the strategies.
%%
put section

 (refresh > 0)
if (rem(iter,refresh) == 0)
   fprintf(1,'Iteration: %d,  Best: %f,  F: %f,  CR: %f,  NP: %d\n',iter,b
   for n=1:D
     fprintf(1,'best(%d) = %f\n',n,bestmem(n));
   end
end
d
 if (rem(iter,10) == 0)
```

FIGURE A.27: Script K

Appendix B

Data Plots: Hourly and weekly average

The Figure below illustrates the level and power cover formed by second order polynomial fit that is determined by using hourly data. **NOTE:** The hourly data is converted into daily data in accordance to its mean value for every 168 hours.

B.1 Hourly Power data curves

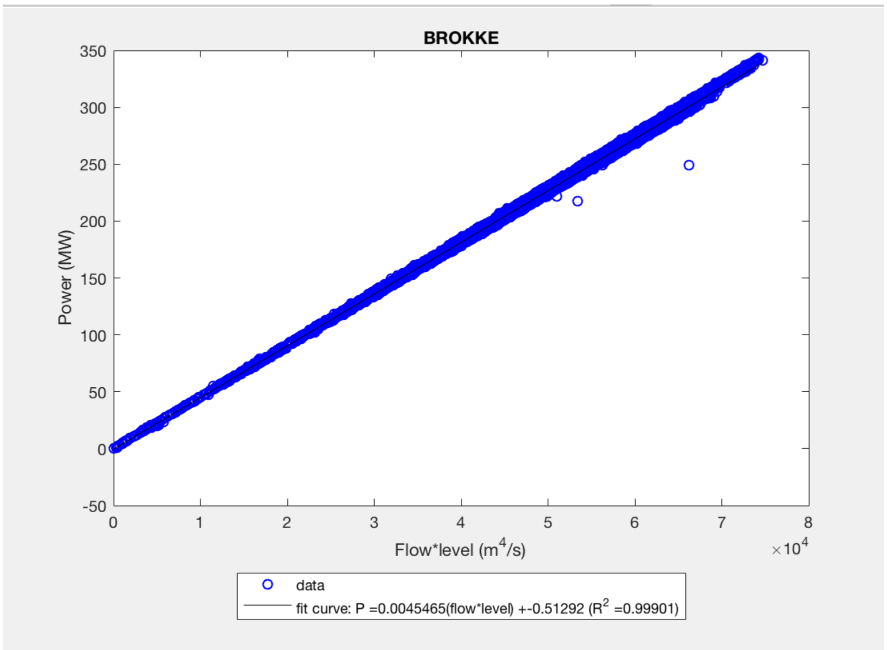

FIGURE B.1: Power Brokke

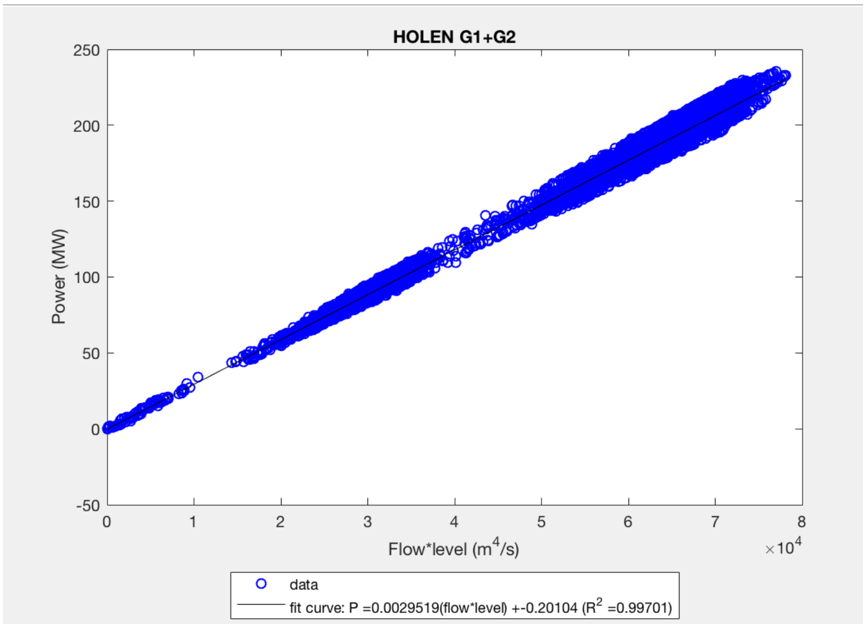

FIGURE B.2: Power Holen G1-G2

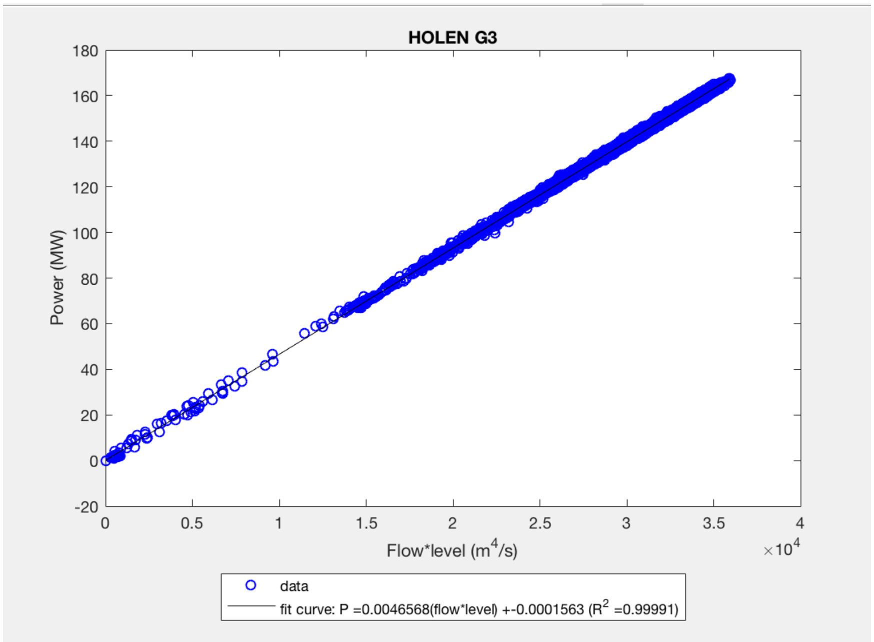

FIGURE B.3: Power Holen G3

B.2 Hourly Level data curves

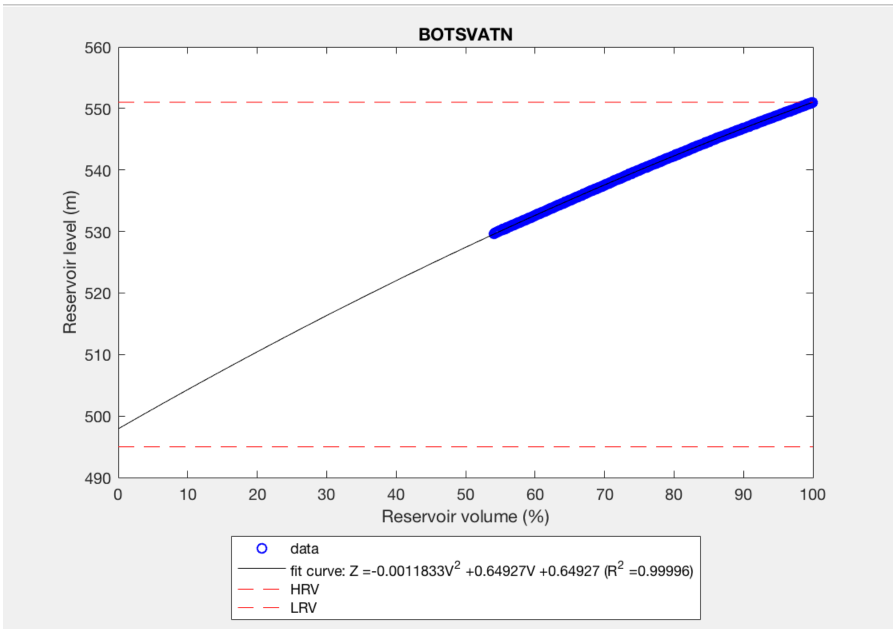

FIGURE B.4: Botsvatn: Level vs Volume %

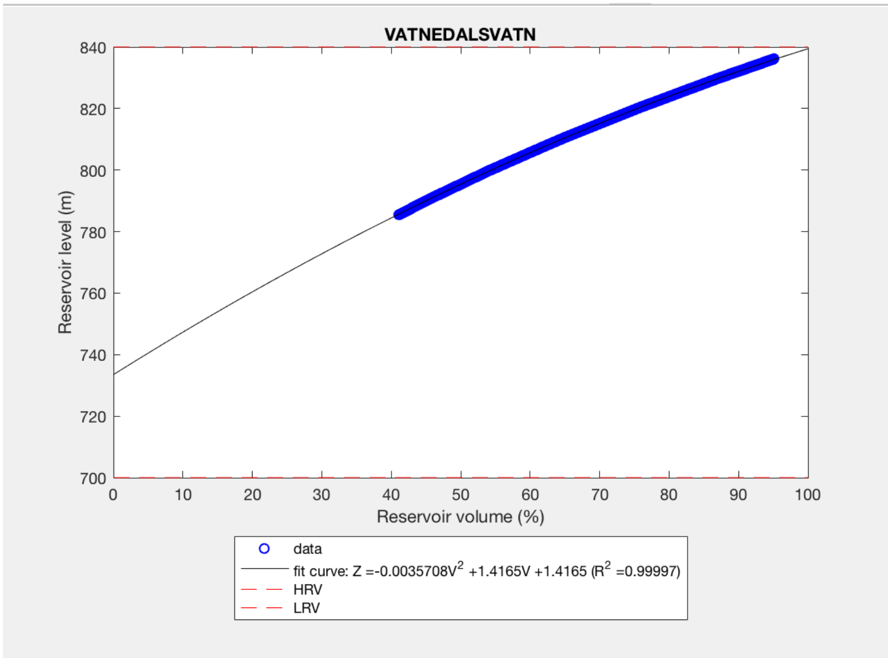

FIGURE B.5: Vatnedalsvatn: Level vs Volume %

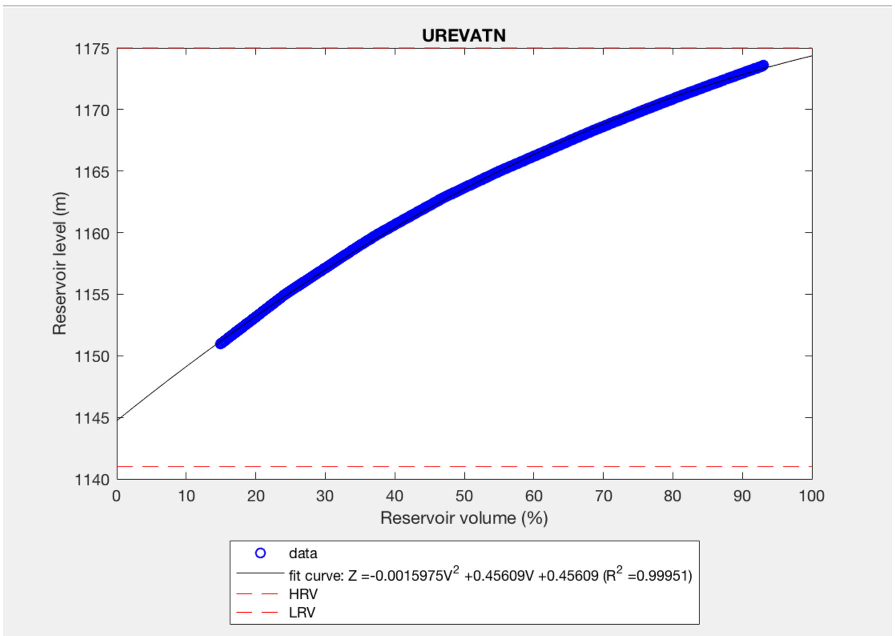

FIGURE B.6: Urevatn: Level vs Volume %

B.3 Power and level curves (weekly average)

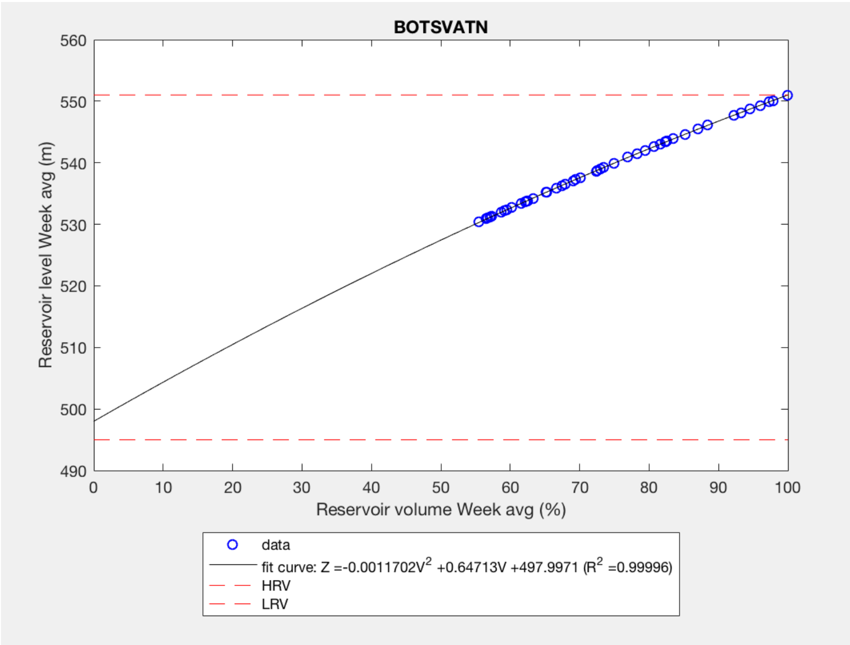

FIGURE B.7: Botsvatn level (weekly average)

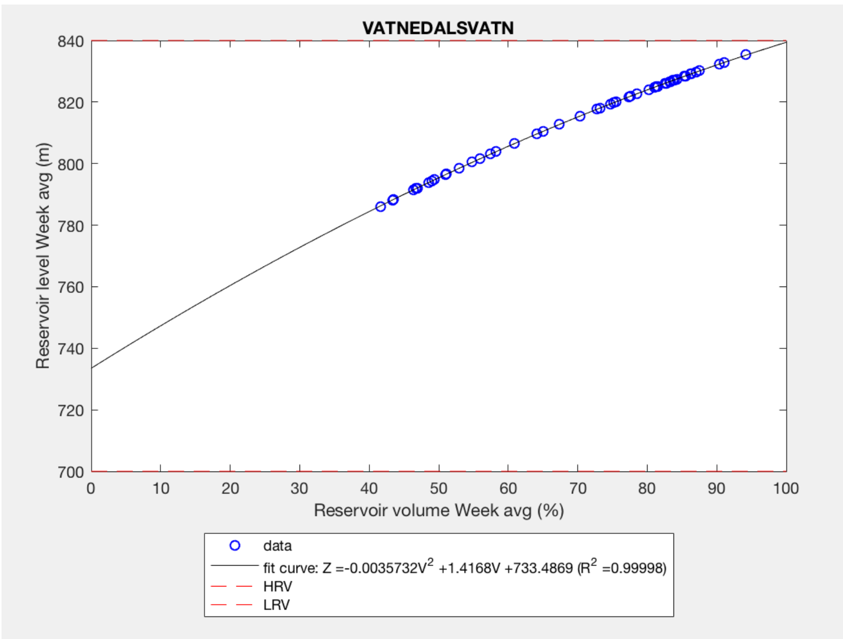

FIGURE B.8: Vatnedalsvatn level (weekly average)

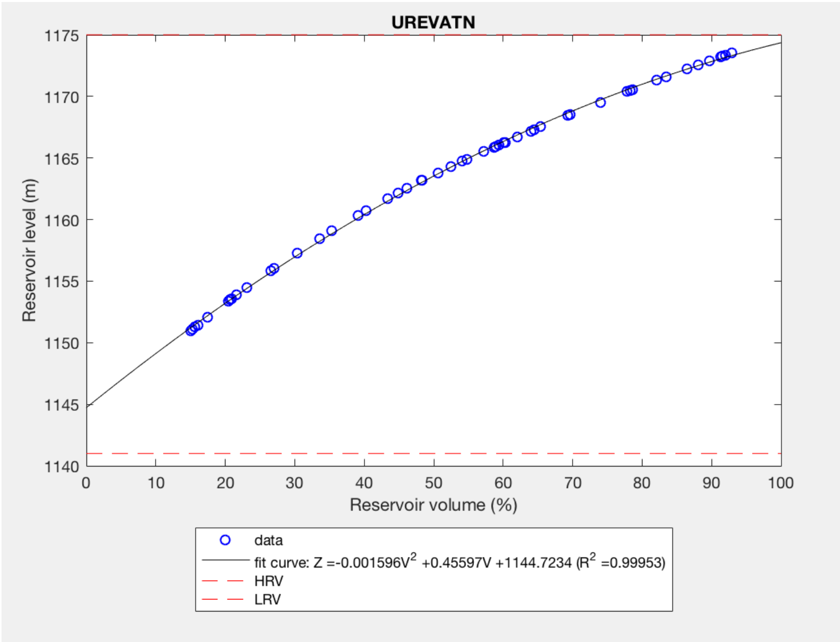

FIGURE B.9: Urevatn level (weekly average)

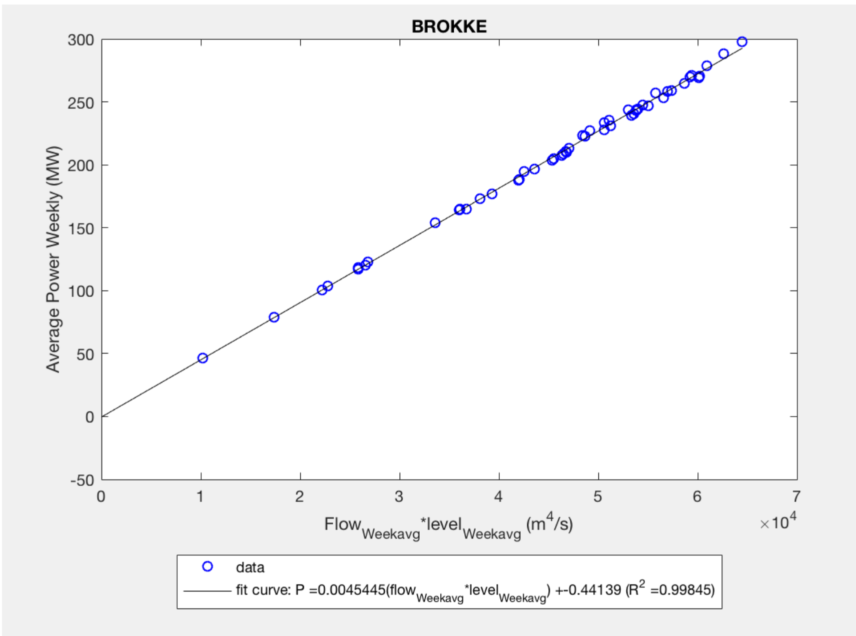

FIGURE B.10: Power Brokke (weekly average)

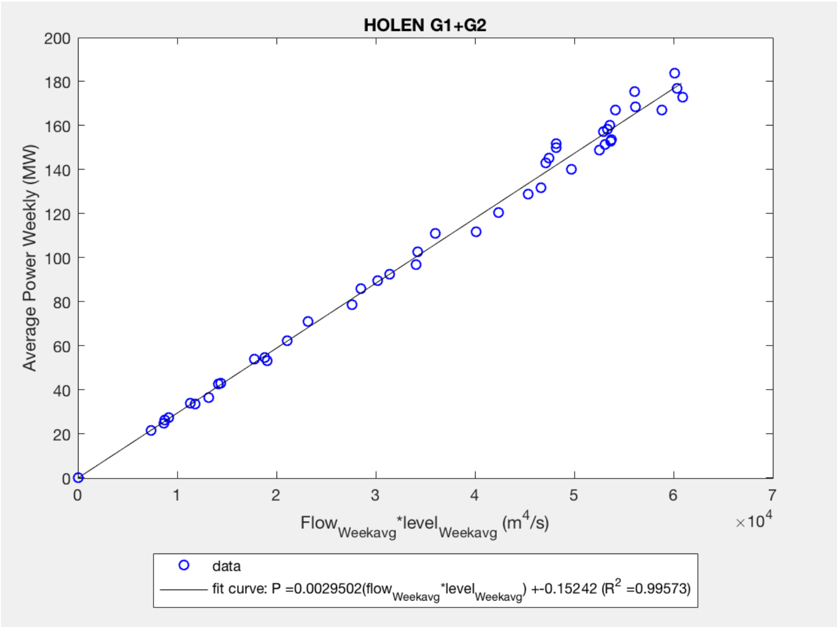

FIGURE B.11: Power Holen G1-G2 (weekly average)

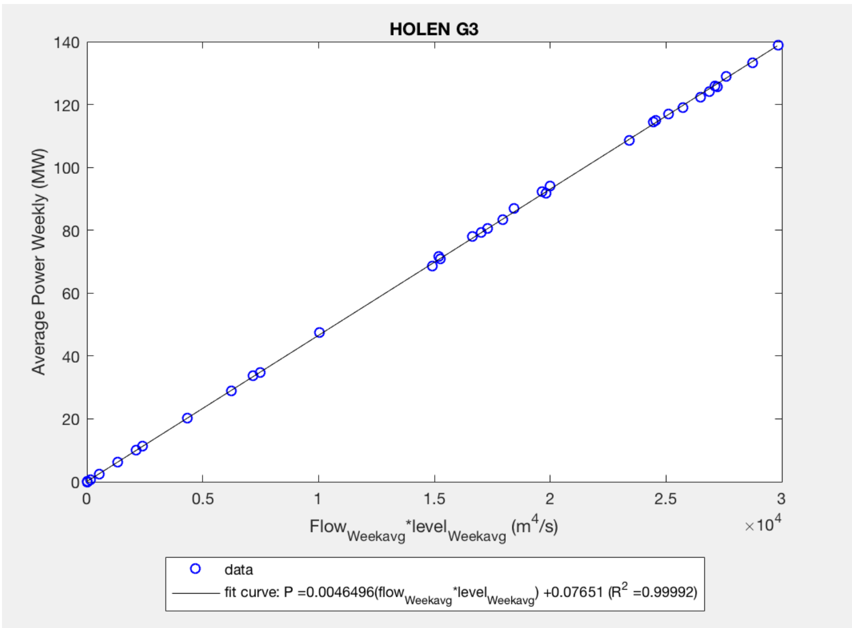

FIGURE B.12: Power Holen G3 (weekly average)

B.4 Inflow plot

Figures showing average inflows throughout the year.

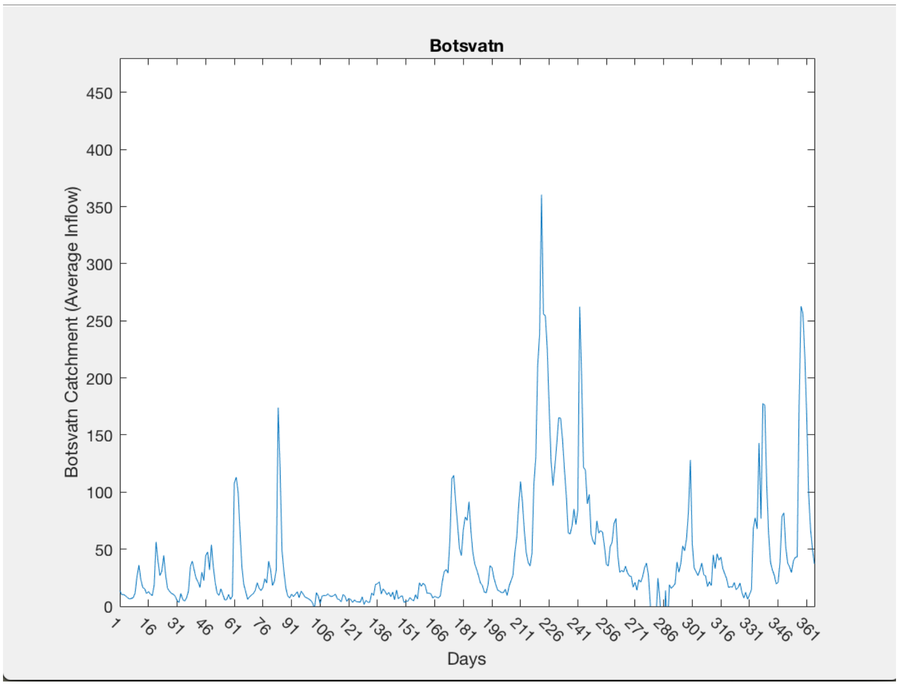

FIGURE B.13: Botsvatn Catchment (Average)

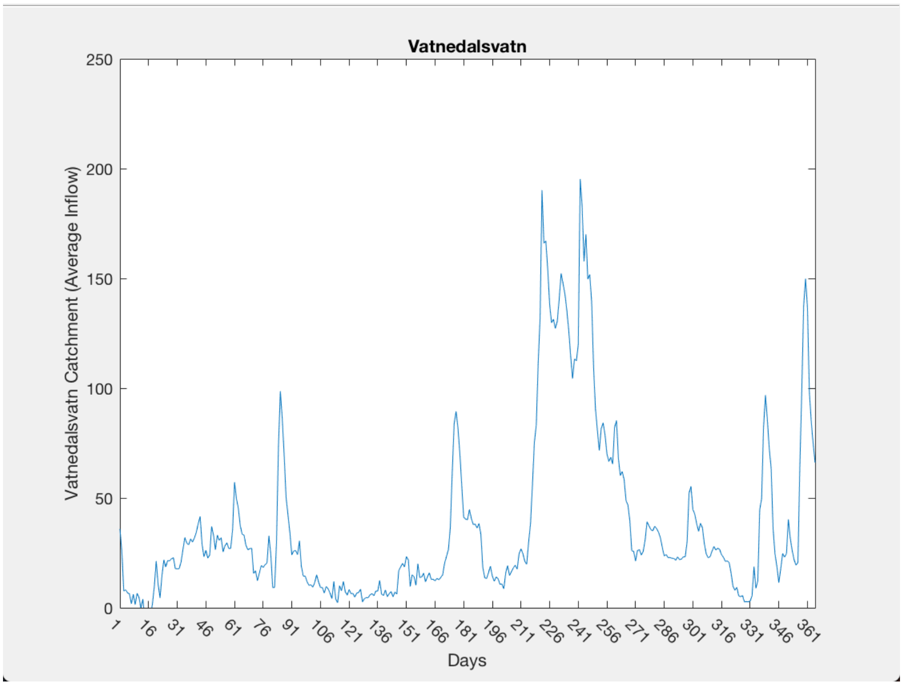

FIGURE B.14: Vatnedalsvatn Catchment (Average)

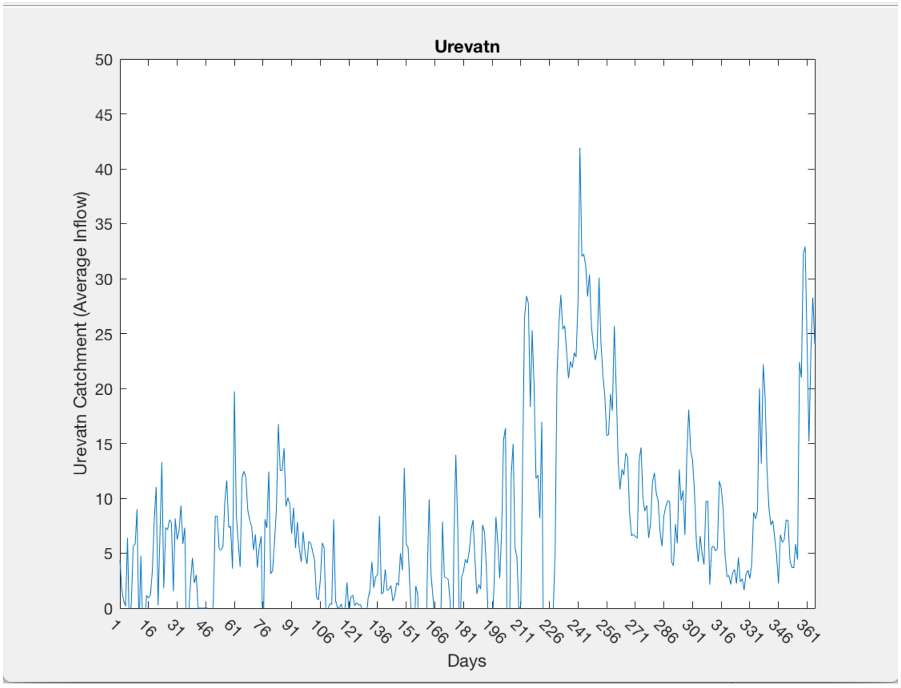

FIGURE B.15: Urevatn Catchment (Average)

Appendix C

Weekly Optimization results

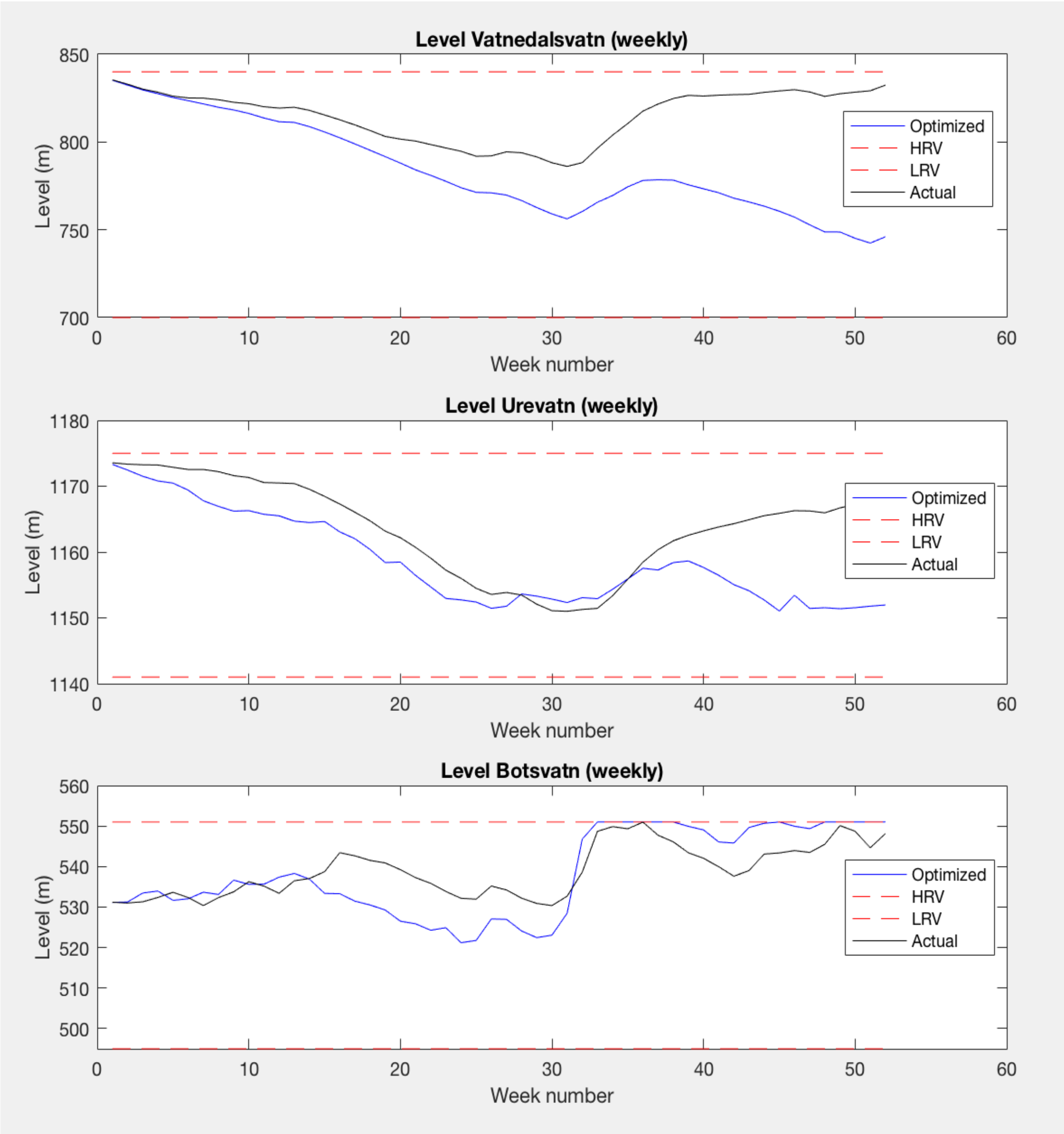

FIGURE C.1: Weekly average reservoir level

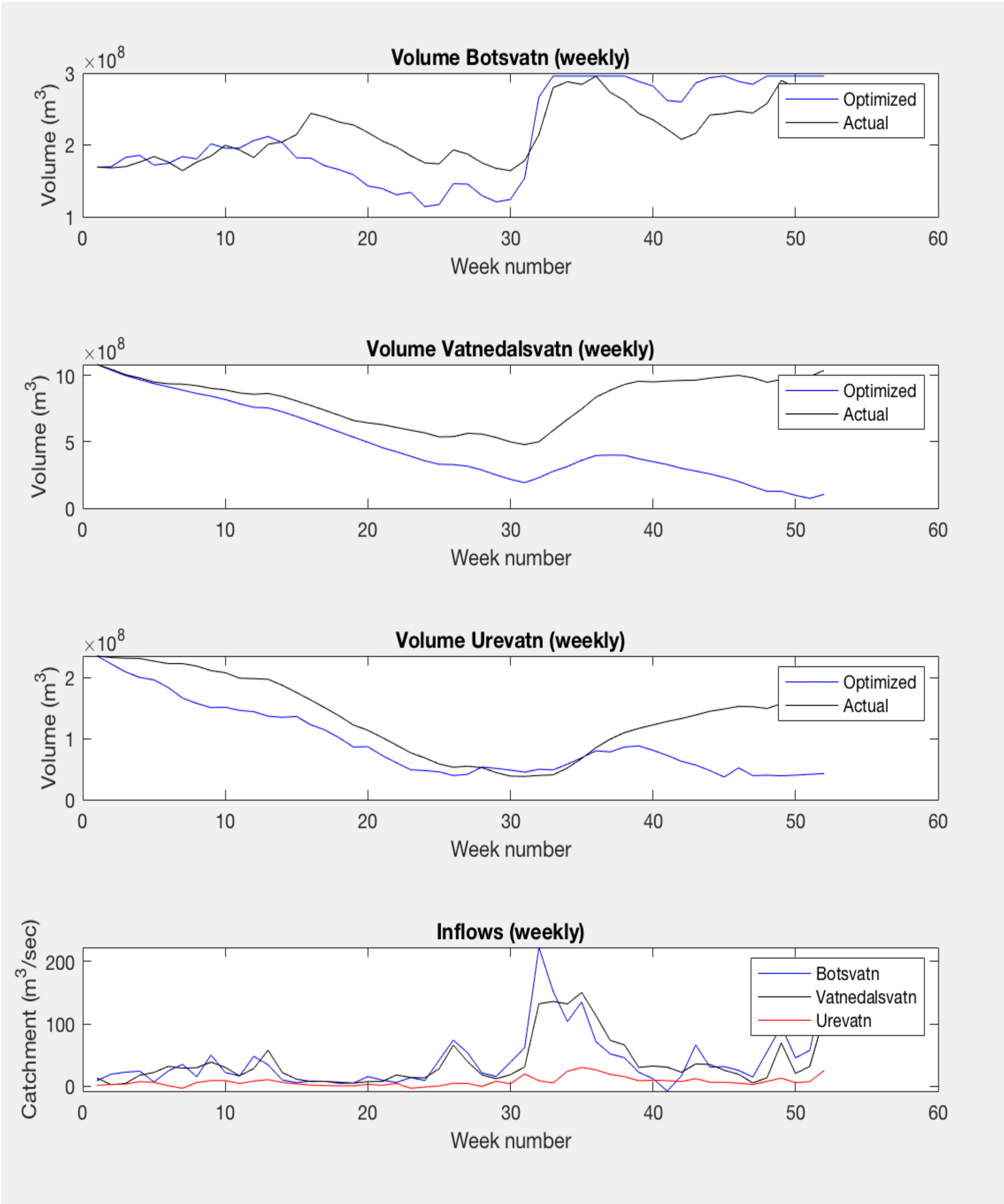

FIGURE C.2: Weekly average reservoir volume

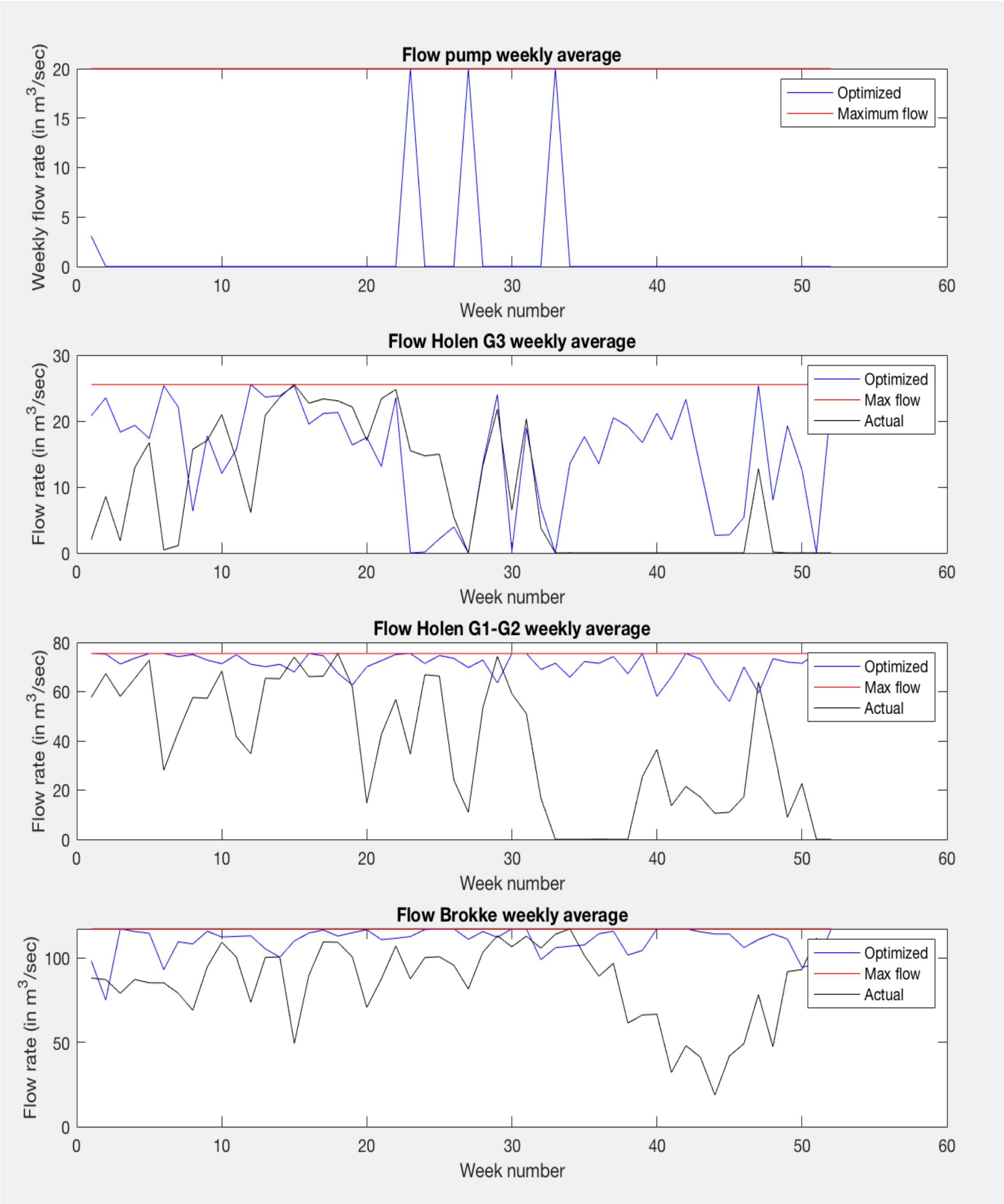

FIGURE C.3: Weekly average reservoir volume

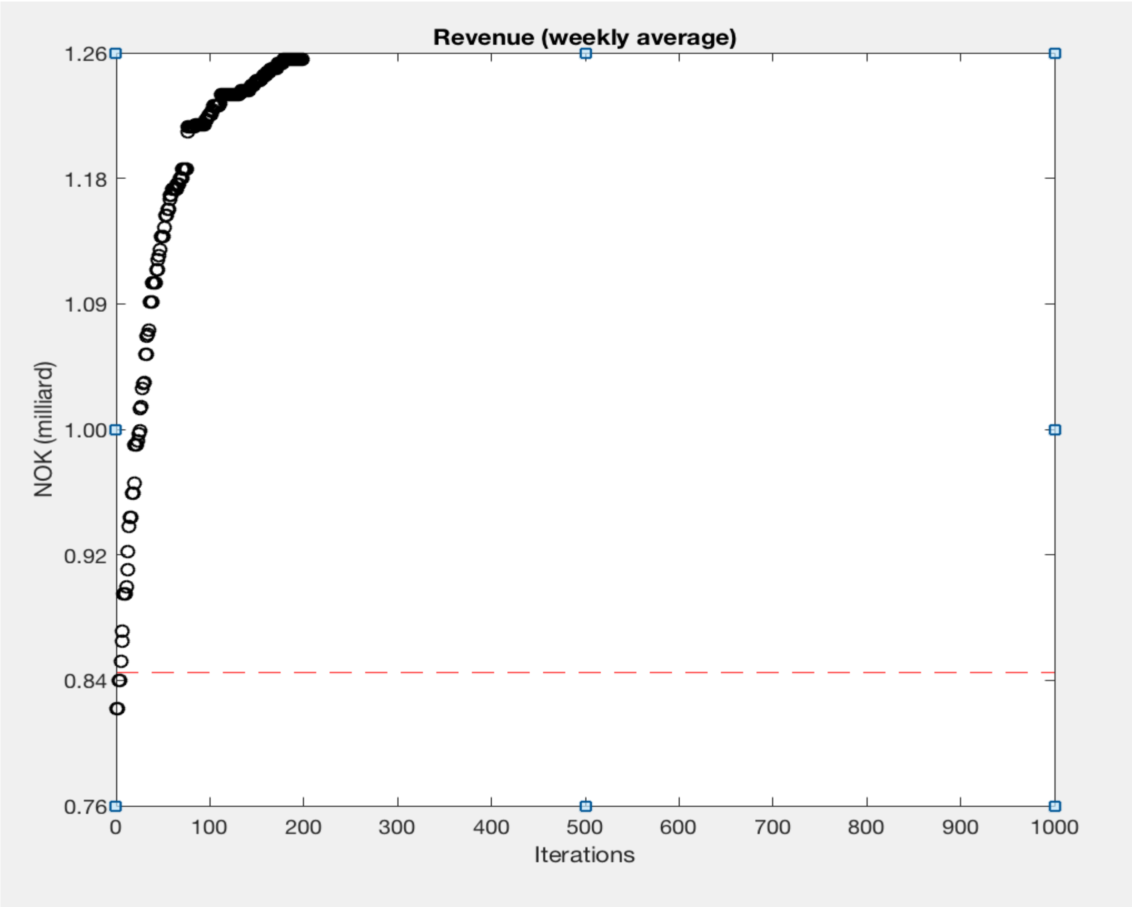

FIGURE C.4: Weekly average revenue vs actual revenue

Appendix D

Optimization results - daily average without ON/OFF condition (200 days)

The Figure below illustrates the level and power cover formed by second order polynomial fit.

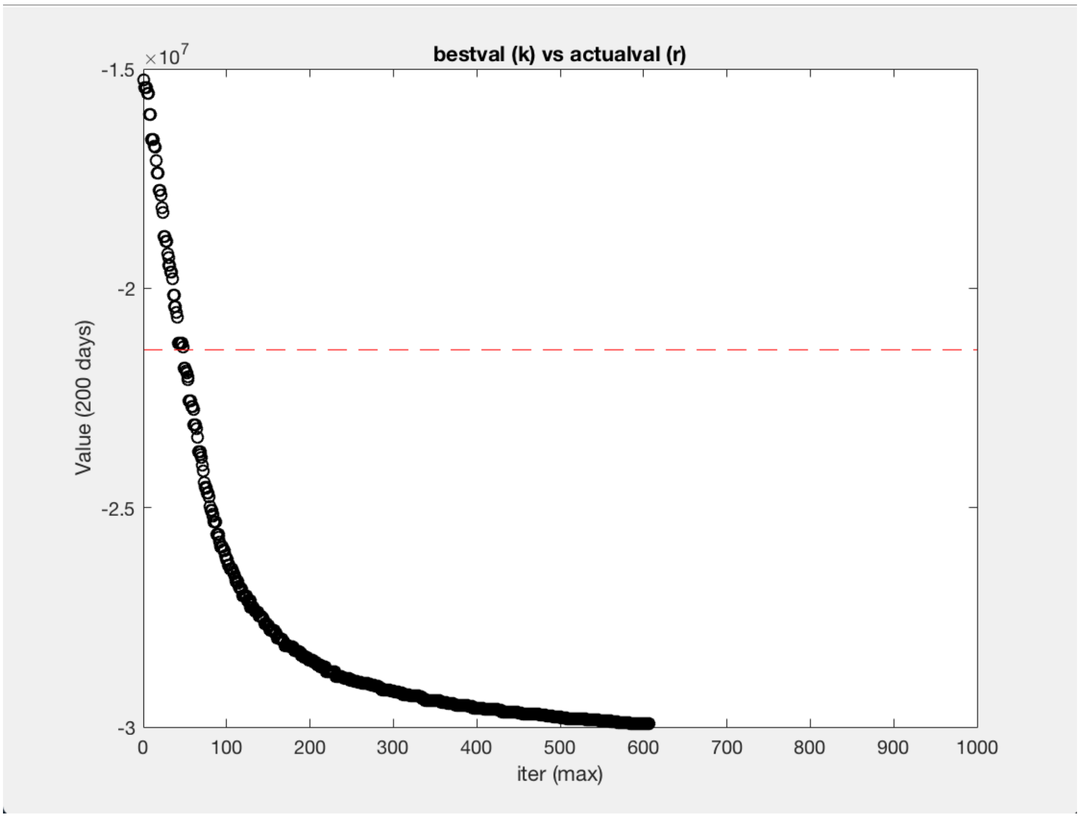

FIGURE D.1: Best value or Profit for 200 days

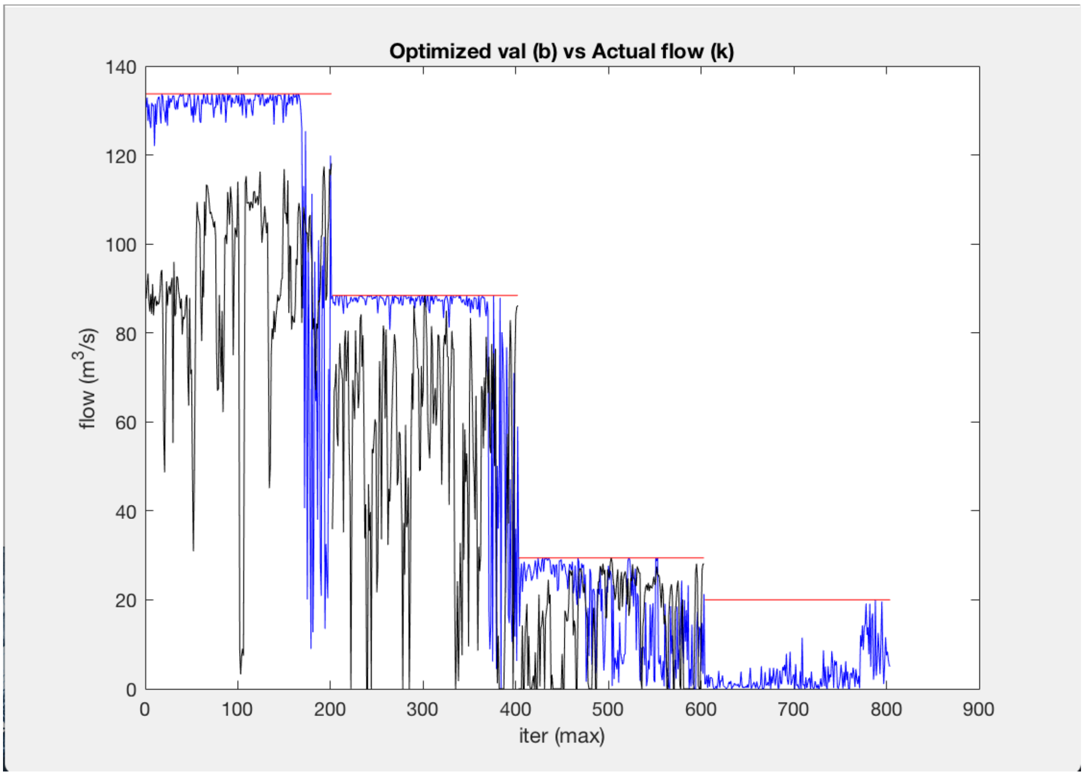

FIGURE D.2: Optimize value (blue) vs Actual value (black)

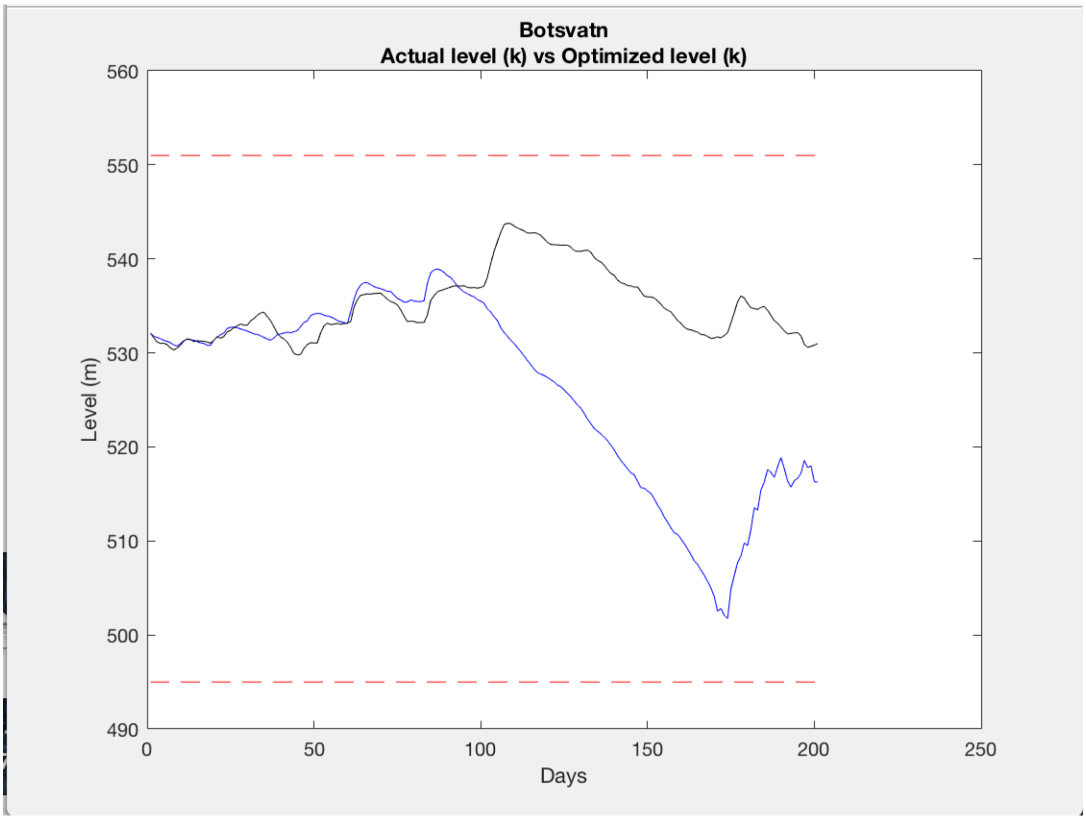

FIGURE D.3: Botsvatn: Optimized level (blue) vs Actual level (black)

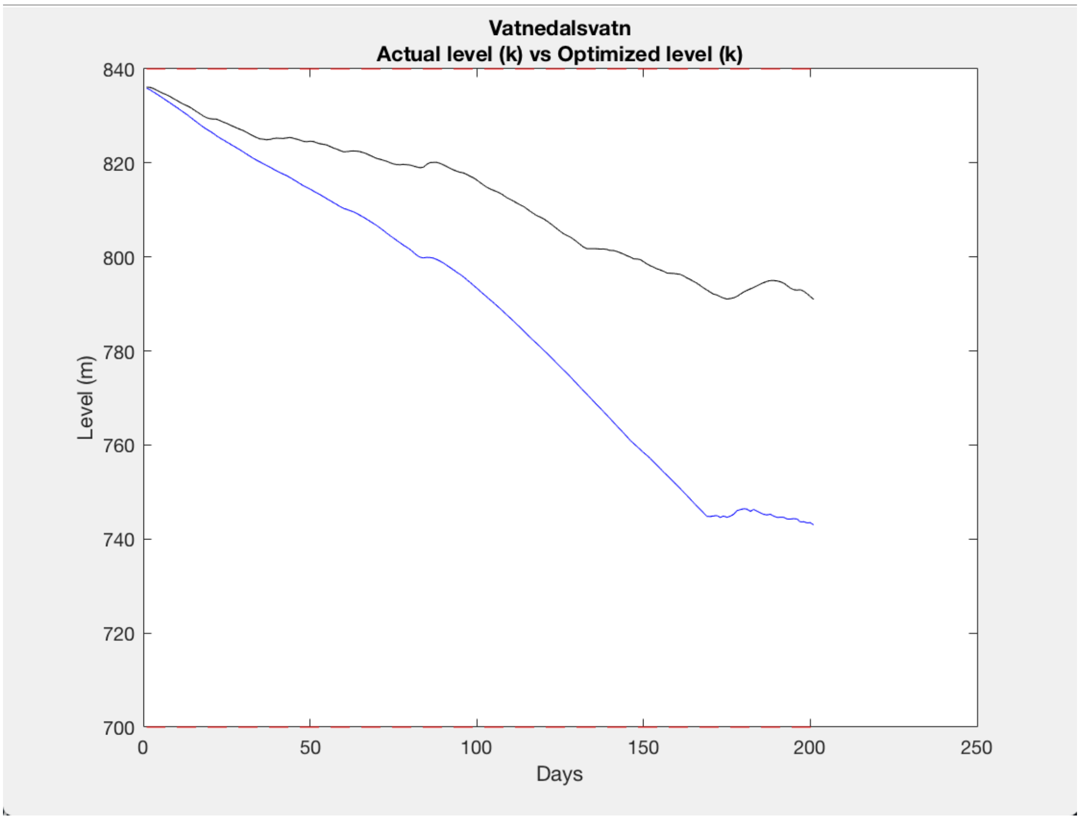

FIGURE D.4: Vatnedalsvatn: Optimized level (blue) vs Actual level (black)

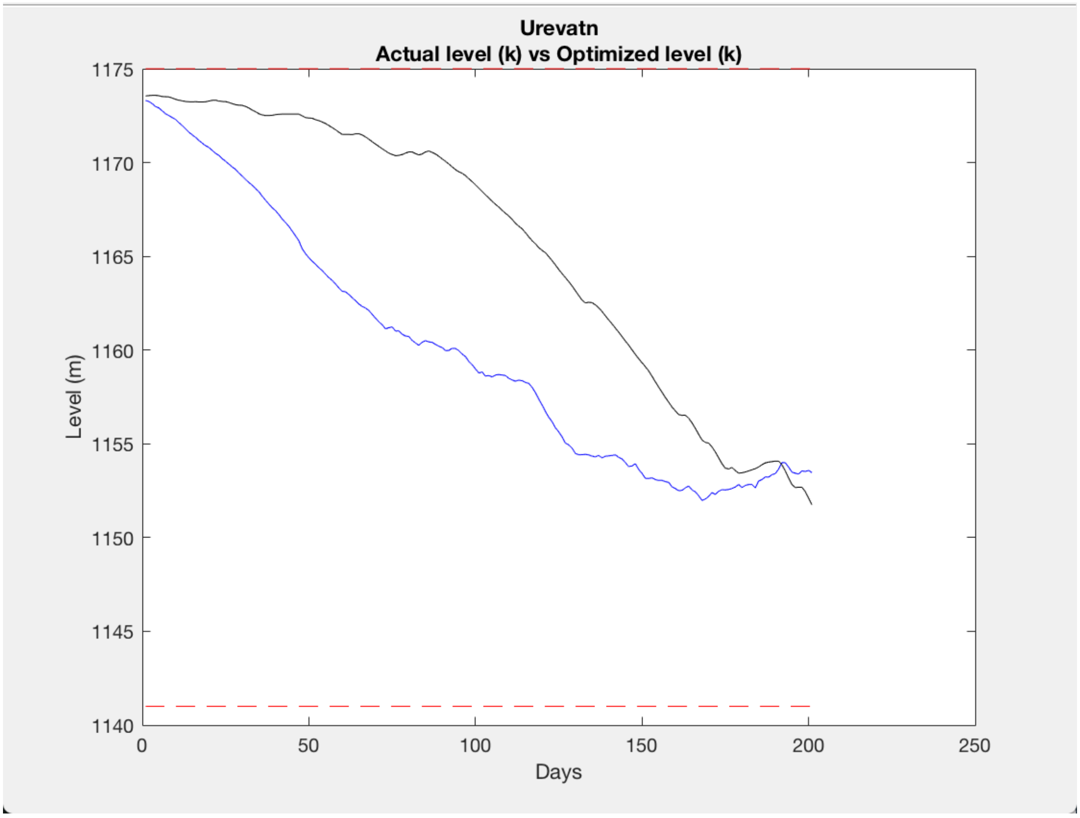

FIGURE D.5: Urevatn: Optimized level (blue) vs Actual level (black)

Appendix E

Optimization results using FMINCON

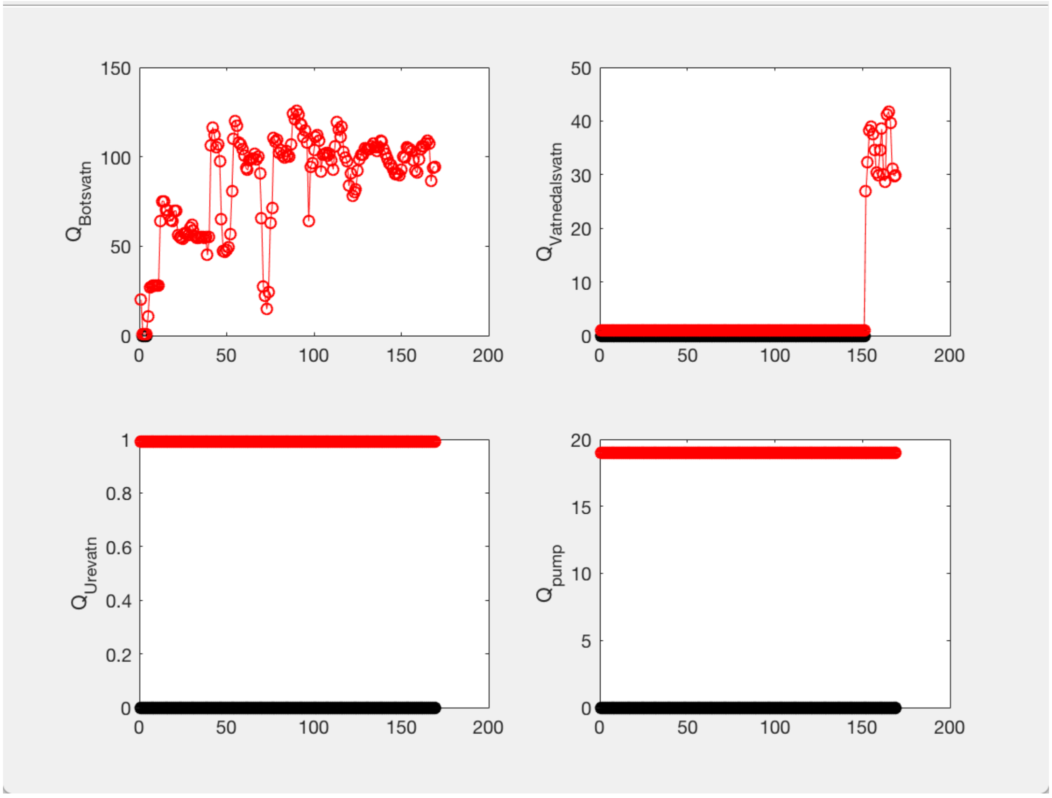

FIGURE E.1: Results using FMINCON

Appendix F

Pump Selection

For Otra Kraft Hydropower system Vertical Submersible (Multistage) pump shown in Figure F.1 is suitable because of its operation flow potential and head which is upto 1067m/3500 ft.(Statkraft, 2018, Ioannis Kougias, 2017).

FIGURE F.1: Model VIT - Vertical submersible pump (Statkraft, online)

Technical Specification of Pump

The main consideration for choosing pump are as follow.

Hydraulic Coverage

The hydraulic coverage chart (per stage) for estimating the head at the respective rpm is presented in Figure F.2 (Statkraft, online).

Flows $\longrightarrow$	70,000 [GPM] or 4.4163 [$m3/s$]
Head $\longrightarrow$	up to 1060 [m]
Bowl size $\longrightarrow$	152.4 to 1400 [mm]
Temperature $\longrightarrow$	up to 260 °C
Power Supply $\longrightarrow$	3ph./ 50 [Hz]
Pumping Liquid $\longrightarrow$	Water

TABLE F.1: Technical specification Model VIT

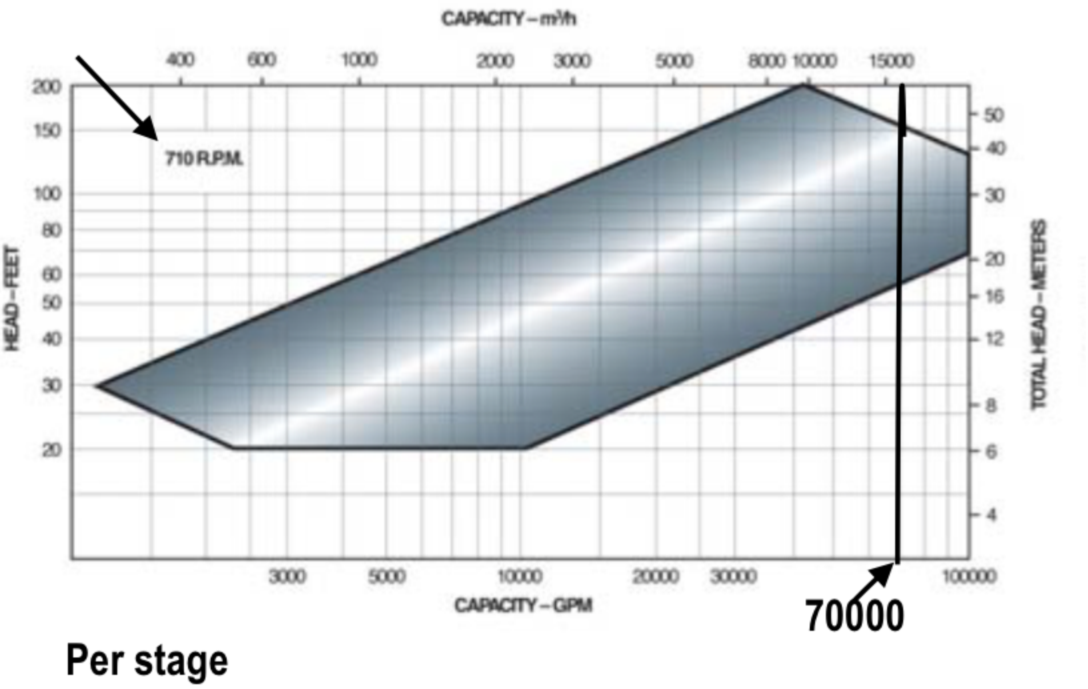

FIGURE F.2: Hydraulic coverage per stage, (Statkraft, online)

As marked in the Figure F.2 the required flow rate according to the capacity of the pump is 70000 [GPM] or 4.42 [m^3/s] at 710 [$R.P.M$]. We can further regulate the flow rate by attaching more pumps in parallel, for example, if we require the flow rate higher than 4.42 [m^3/s] than we can combine more than one pump together in parallel.

Efficiency of the pump

The pump efficiency and the corresponding power can be computed from Gould's software (Julie Charmasson, 2017). From Figure F.3 we can observe that at 740 rpm the efficiency of pump per stage is 83% and the corresponding power is 4146 kW or 5637 hp.

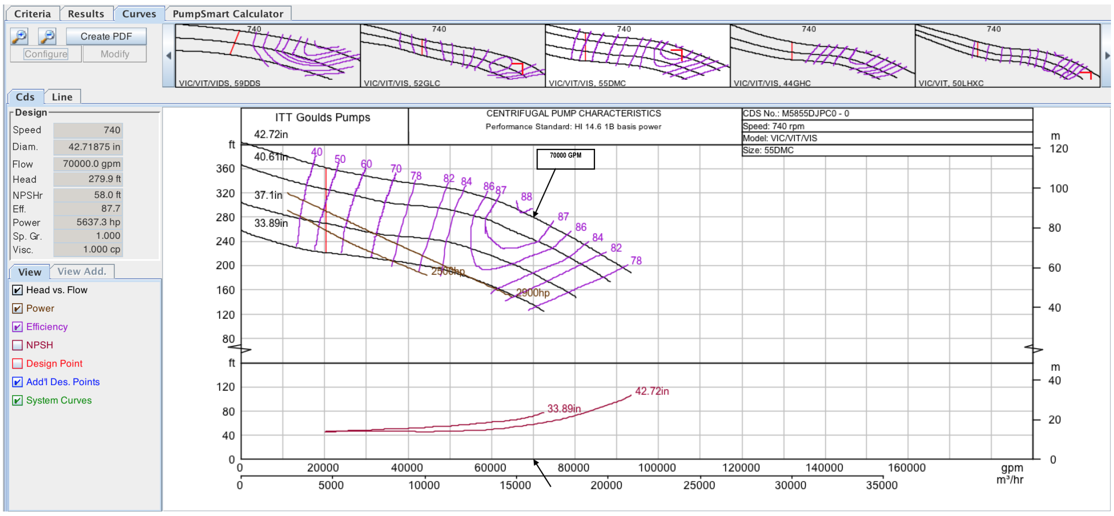

FIGURE F.3: Model VIT efficiency curve

Since we can observe that for the selected pump we can get upto 90000 [GPM] flow rate at the efficiency of 78 % or 70000[GPM] at 88 %.Now, we have the pump according to the head at site.

the total hydraulic power needed to operate the pump when pumping 4.42 m^3/s of water up to a height of 650 m and η is 0.83, can be determined by equation 3:

$$P_{shaft} = \frac{1000 * 9.81 * 650 * 4.42}{0.83} = 33956.8kW \tag{F.1}$$

Appendix G

Statistics

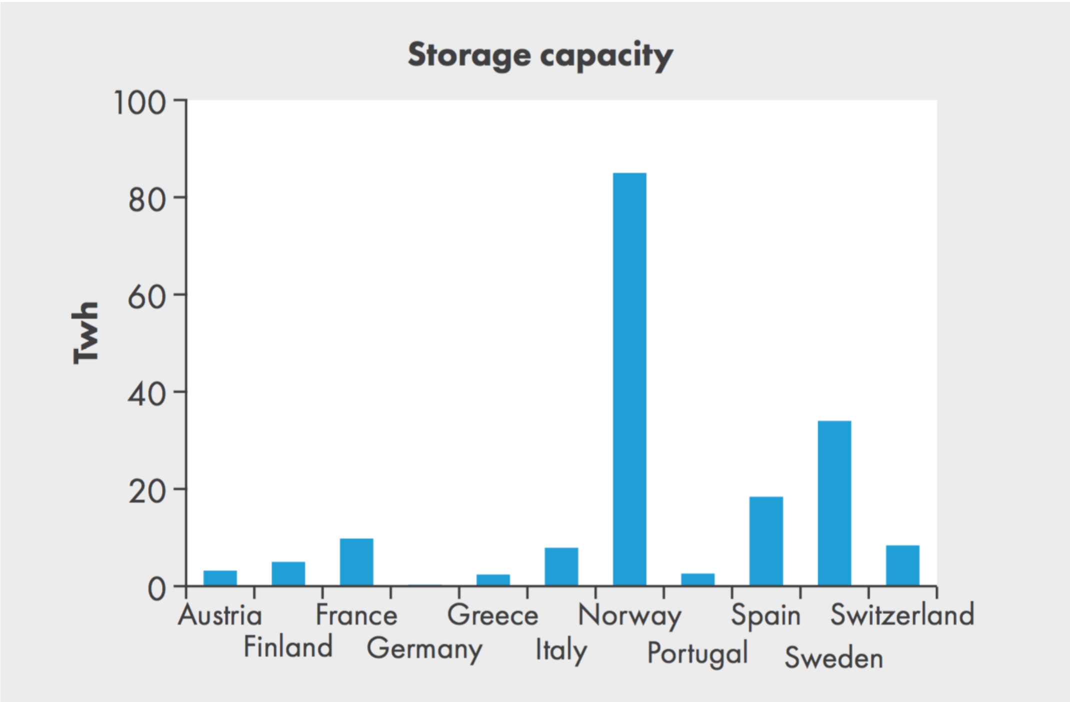

FIGURE G.1: Storage capacity

Appendix H

Reservoirs data

H.1 Urevatn

Vatn løpenummer	
Magasinnummer	297
Magasinnavn	URARVATN
Magasin kategori	1
LRV	1141 m.o.h
HRV	1175 m.o.h
Status	Drift
I drift	1997
Konsesjonsstatus	
Konsesjons dato	
Magasin formål	Kraftproduksjon
Areal ved HRV	13.15 km²
Magasinvolum	253.4 Mill. m³
Delfeltnummer	2560
Vannkraftverk nr	373
Vannkraftverk navn	Holen III
Vassdragsnummer	
Elvehierarki	URAÅNI/LØYNINGSÅNI/OTRA
gå til	Konsesjonssak

FIGURE H.1: Urevatn parameters

H.2 Vatnedalsvatn

Magasin ×

Vatn løpenummer	
Magasinnummer	290
Magasinnavn	VATNEDALSVATN
Magasin kategori	1
LRV	700 m.o.h
HRV	840 m.o.h
Status	Drift
I drift	1984
Konsesjonsstatus	
Konsesjons dato	
Magasin formål	Kraftproduksjon
Areal ved HRV	14.42 km²
Magasinvolum	1150 Mill. m³
Delfeltnummer	2616
Vannkraftverk nr	158
Vannkraftverk navn	Holen I-II
Vassdragsnummer	
Elvehierarki	LØYNINGSÅNI/OTRA
gå til	Konsesjonssak

FIGURE H.2: Vatnedalsvatn parameters

H.3 Botsvatn

Vatn løpenummer	
Magasinnummer	289
Magasinnavn	BOSSVATN
Magasin kategori	1
LRV	495 m.o.h
HRV	551 m.o.h
Status	Drift
I drift	1976
Konsesjonsstatus	
Konsesjons dato	
Magasin formål	Kraftproduksjon
Areal ved HRV	7.7 km^2
Magasinvolum	296 Mill. m^3
Delfeltnummer	1143
Vannkraftverk nr	45
Vannkraftverk navn	Brokke
Vassdragsnummer	
Elvehierarki	BOSSVASSÅI/OTRA
gå til	Konsesjonssak

FIGURE H.3: Botsvatn parameters