NTNU
Norwegian University of
Science and Technology

Optimal Bidding Strategy in the Reserve Capacity Market

Eirik Schou Grytli

Master of Energy and Environmental Engineering
Submission date: June 2016
Supervisor: Magnus Korpås, ELKRAFT
Co-supervisor: Marte Fodstad, ELKRAFT

Norwegian University of Science and Technology
Department of Electric Power Engineering

Preface

This thesis is written as part of my Master's degree in Electrical Power Engineering with the Department of Electrical Power Engineering at the Norwegian University of Science and Technology. The thesis was written during the spring of 2016. The main topic of the study was proposed by my supervisors Magnus Korpås and Marte Fodstad. I would like to thank Professor Magnus Korpås for convincing me to immerse myself in this subject. He has been very helpful and his office door has always been open. I also would like to thank Marte Fodstad for excellent guidance on this thesis and for her understanding and patience. She has always been helpful and welcoming at my frequent visits. I also want to thank her and Magnus for taking time to give me valuable and thorough feedback on my master work, multiple times. I also want to thank Arild Lote Henden for guidance on ProdRisk and Python. I also want to thank Anders Gytri and Kurt Salmi at the Department of Electrical Engineering for their help. A final thanks must be directed to Siri B. Ager-Hanssen for valuable feedback on this thesis.

Trondheim, June 14, 2016

Eirik Schou Grytli

Abstract

Good balancing services is a prerequisite for a well-functioning power market. Additionally to the day-Ahead market (DA) for electricity, there is a balancing market (BM) which provides the necessary buffers to handle short visibilities and uncertainties,such as frequency deviations, in the grid. To ensure that enough balancing reserves are available, a reserve capacity market(RKOM) has been created as an incentive for participants in the power market to reserve capacity exclusively for the BM. As part of the project "Integrating Balancing Markets in Hydro Power Scheduling Methods" was a short term hydro power scheduling model implemented,by SINTEF Energy, in the mathematical programming tool AMPL. It is a multi stage, multi scenario stochastic optimization problem.

The main purpose of this thesis has been to evaluate how participating in RKOM affects the decision making of a hydro power producer compare to only participating in the DA and BM. The weekly time resolution of the reserve capacity market makes it difficult to analyze in already excising hydro power scheduling models. Consequently, the model implemented in AMPL by SINTEF Energy has in this thesis been expanded and altered to incorporate the reserve capacity market. The main changes, in the model, were done considering the model's time horizon, and the scenario tree input used in the model.Stage wise scenario reduction was used to handled some of the challenges considering the model expansion. The model expansion was called AMPLWeek and is meant as decision-support for a hydro power producer considering bids in RKOM. Additionally, a simulation method, that includes AMPLWeek, was created to observe how simulation over multiple weeks, and seasons, affects short term scheduling. This was done by incorporating the changes in the reservoir level and water values, created by ProdRisk, over several weeks of operation. The study sought to answer how decision-making changes by participating in RKOM. A case study was done considering Statkraft's Tokke-Vinje hydro power plant in the NO2-area of Norway. To evaluate the efficiency of RKOM, and to see if there are any gains for a hydro power scheduling to participate in this market, the model was evaluated for different RKOM prices and different seasonal simulations. This thesis also intents to evaluate the benefits of using a weekly scheduling plan compared to a daily time horizon, and how doing seasonal simulations affects the decision making for the hydro power producer. It is also considered to which extent scenario reduction affects the approximation done considering DA and BM prices in this thesis.

The work presented in this thesis demonstrates that a hydro power producer's willingness to reserve capacity in RKOM increases with an increased RKOM price. The average power dispatch in the DA and the down regulation dispatch in BM decreases with a higher RKOM price. Smaller changes could be observed in the average up regulation dispatch, in form of a bell shaped curve, with increasing RKOM price. The up regulation dispatch is higher for a low RKOM price than for a high RKOM price. Results from seasonal simulations indicates that participating in RKOM is most profitable during spring and summer, when day-ahead prices and reservoir levels are low. Comparative analysis of a weekly and daily time horizon demonstrates that the optimal reserved capacity in

RKOM changes based on the time horizon. A lower objective value is obtained with a weekly time horizon.

The main finding of this thesis is that RKOM is a profitable market for a hydro power producer. Further, the bell-shaped curve of average up regulation dispatch with respect to RKOM price might indicate that incentive created by RKOM not necessarily increase the participant's dispatch in BM. Whether this is due to model simplifications or also can be observed in the real market is unknown, and recommended as further work. It was also concluded that seasonal simulations provides useful information about changes in hydro system parameters, like reservoir levels, not observed by a weekly optimization. It also follows from the results of this thesis, that scheduling model with a weekly time horizon provides better decision support than a model with a daily time horizon.

Sammendrag

Gode balansetjenester er en forutsetning for et velfungerende kraftmarked. I tillegg til spotmarkedet for strøm finnes det et regulerkraftmarkdet som er ment til å gi en nødvendig buffer for å håndtere kort forandringer og usikkerheter, som for eksempel frekvensavvik i strømnettet. For å sikre at nok reguleringskraft er tilgjengelig i markedet, finnes et ReserveKraftOpsjonsMarked, RKOM. RKOM er opprettet som et insentiv for at produsenter og konsumenter skal ta del i oppreguleringen av regulerkraftmarkdet. Som en del av prosjektet "Integrating Balancing Markets in Hydro Power Scheduling Methods", ble en kortsiktig vannkraftoptimeringsmodell implementert av SINTEF Energi i det matematisk programmeringsverktøyet AMPL. Modellen er et fler-stegs, fler-scenario, stokastisk optimaliseringproblem.

Hovedformålet med denne oppgaven har vært å vurdere hvordan deltagelse i RKOM påvirker beslutningsgrunnlaget til en vannkraftprodusent i forhold til å kun delta i regulerkraftmarkdet og spotmarkedet. Den ukentlige tidsoppløsning i RKOM gjør markedet vanskelig å analysere i allerede eksisterende vannkraftmodeller. Derfor har man, i denne oppgaven, utvidet en eksisterende kortsiktig vannkraftplanleggingsmodellen og forandret den slik at RKOM har blitt inkludert. De viktigste endringene i modellen ble gjort med tanke på tidshorisonten og scenariettreet brukt i modellen. For å håndtere utfordringer knyttet til det å utvide en stokastisk modell med flere steg, ble stegvis senarioreduksjon brukt. Modellutvidelsen ble kalt AMPLWeek og er ment som beslutningsstøtte for en vannkraftprodusent som byr inn i RKOM. I tillegg til utvidelsen har en simulerings porsedyre som inkluderer AMPLWeek blitt laget for å observere hvordan simulering over flere uker og sesonger påvirker kortsiktig vannkraftplanlegging. Simuleringen ble gjort ved å innlemme endringene i magasinfyllingen og vannverdier, skapt av ProdRisk, over flere uker med simulert drift. Studiet ønsker å besvare hvordan beslutninger i modellen endres ved å reservere forskjellige kapasitetsvolum i RKOM. Modellen har blitt evaluert for flere forskjellige RKOM priser. Dette har blitt gjort for å evaluere effektiviteten av RKOM, som marked, og for å se om det er noen gevinster for en vannkraftprodusent å delta i dette markedet. Denne masteroppgaven har også som intensjon å vurdere fordelene ved å bruke en ukentlig planleggingsmodell i forhold til daglig planleggingsmodeller med tanke på bud i RKOM. Det har også blitt vurdert hvordan senarioreduksjon påvirker prisinputen i modellen. Arbeidet, som presenteres i denne masteroppgaven, viser at en vannkraftprodusent er villig til å reservere mer kapasitet i RKOM for en høyere RKOM pris. Gjennomsnittlig kraftproduksjon i spotmarkedet og nedregulering i regulerkraftmarkdet avtar med høyere RKOM pris. Resultatene viser at oppregulering i regulerkraftmarkdet endres med økende RKOM pris, og en klokkeformet kurve kan observeres for økende pris. Et lavere volum produseres i balansemarkedet for en høy RKOM-pris enn en lav RKOM-pris Resultater fra sesongensimuleringer indikerer at å delta i RKOM er mest lønnsomt om våren og sommeren, når spottprisen og reservoarnivåene er lave. Komparative analyser av en modell med en ukentlig og daglig tidshorisont viser at objektivverdien i modellen endres med forskjellig tidshorisont. En lavere objektivverdi kan observeres med en ukentlig tidshorisont.

Det konkluderes med at den klokkeformede kurven for oppreguleringskraft i Re-

servekraftmarkedet med hensyn til RKOM prisen kan tyde på at RKOM ikke nødvendigvis fungerer som et insentiv for at mer oppregulerkraft blir bydd inn i regulerkraftmarkdet. Om dette skyldes modellforenklinger eller er en reel effekt som kan observeres i markedet er uvisst og blir oppfordret som videre studie. Det blir også konkluderte med at sesongsimuleringer gir nyttig informasjon om endringer i vannivå i magasiner for forskejllige sesonger, i forhold til en vannkraftoptimeringsmodell som ikke observert av en ukentlig optimalisering. Det blir også konkludert med at planleggingsmodeller med en ukentlig tidshorisont gir bedre beslutningsstøtte enn en modell med en daglig tidshorisont.

Problem Description

This thesis considers a hydro power producer in the NO2 area of Norway, and a short term hydro power scheduling model, meant as decision support for hydro power producers operating in a reserve capacity market. The scheduling problem is formulated as a stochastic optimization problem, bidding into sequential markets. The problem has been implemented by SINTEF Energy in AMPL 2.6,and will form the basis for this thesis. The research question of this thesis is to evaluate how participating in a reserve capacity market (RKOM) will affect a hydro power producers decisions compared to only participating in the Day-Ahead Market(DA) and the Balancing Market (BM). The weekly time resolution of the reserve capacity market makes it difficult to analyze in already excising hydro power scheduling model. Consequently, in this thesis an excising short term hydro power scheduling model has in this thesis been expanded and altered to incorporate the Reserve capacity market.

Abbreviations

- BM Balancing market
- DA Day-ahead market
- DMM Discrete Markov Method
- EV Electric vehicle
- EFI Expected Future Income
- IDM Intra-day market
- LP Linear Programming
- NPS NordPool Spot
- FCR Primary reserves(frequency-controlled reserves,)
- RKOM Norwegian word(ReguleringsKraftOpsjonsMarkedet)
- FRR-A Secondary reserves(automatic frequency restoration reserves)
- FRR-M Tertiary reserves(manual frequency restoration reserves)
- TSO Transmission System Operator

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

In Norway, 99% of the electricity production is generated by hydro power [4]. The directive 2012/27/EU of the European Parliament defines a biding target of 20 % renewable contribution to the total energy demand before 2020[5]. To reach this goal incorporation of more renewable energy resources like solar and wind power into the grid is necessary. According to Statnett's System Operation and Market development plan for 2014-20, is the frequency quality of the Norwegian power system decreasing due to faster, larger and more frequent changes in generation[1]. The integration of renewable energy sources like wind and solar power generation into the power system, makes it more difficult to keep the system in balance, and frequency deviation, as can be observed in Figure 1.1, is an increasing problem.

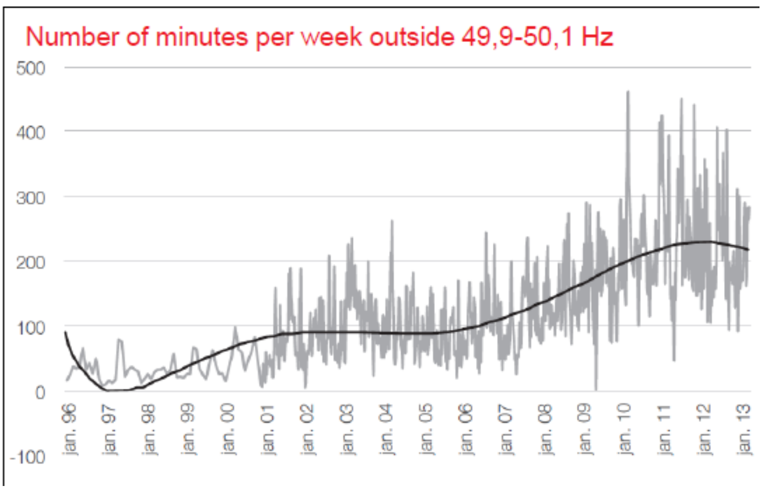


Figure 1.1: Number of minutes per week outside 49.9-50.1 HZ, from 96 to 2013[1]

Although, most of the electric power in Norway is traded in the Day-Ahead market (DA), 73% [6], there exist a Balancing market that provides the necessary buffers to handle short variability and uncertainty in the grid, such as frequency deviations. To better handle the increase in frequency deviation, Statnett introduced a regulation capacity market, RKOM, in 2000. In this market, the participants are paid to reserve capacity that is to be bid into the Balancing market (BM). This is one of several steps performed by Statnett to make the BM more efficient and to ensure that enough capacity is available in this market. Due to increased volatility in the European power market, hydro power producers might have to shift more of their capacity from the DA towards the Intra-day and BM. In the future, a robust and well function market for balancing power will be crucial to handle the energy production of tomorrow.

Optimization of short term hydro power scheduling is closely studied in literature and a brief literature review is presented in Chapter 2. This is the first time to the knowledge of the author that the RKOM is fully included in a model. The thesis is organized in the following way; Chapter 2 presents theory related to the work done in this thesis and a litterateur review of previous work done related to this subject. The short term hydro power scheduling mode presented in this thesis are described in Chapter 3. A case study of the Tokke-Vinje Hydro power system is presented in Chapter 4. The findings and discussion of results are presented in Chapter 5. The conclusion and the proposal of further work is presented in Chapter 6 and Chapter 7.

CHAPTER 2

Background Information and Theory

This thesis focus on a hydro power producer in the Nordic energy market. A literature study of previous work related to the subject is presented in Section 2.1. In this chapter brief introduction of the Nordic energy market is presented in Section 2.2. A brief introduction to Hydro Power Scheduling is also given, this can be found in Section 2.3. An overview of optimization in short term hydro scheduling is presented in Section 2.4.1. Relevant information about software tools used in this thesis is presented in Section 2.6. As this thesis is a continuation of previous work done by the author, this chapter includes some background information that also was given in [7].

2.1 Litterateur review

In this section literature relevant for this thesis are presented. The literature review is divide into three parts. On part considers the forecasting of DA and BM prices used as input to hydro power scheduling. Another part considers some work done on scenario generation and reduction. The results of the scenario generation is also used as input to the optimization model. The work considering the bidding problem is also described in this section.

$$\begin{tikzcd}
\text{Forecasting of Electricity Prices} \arrow[r] \arrow[dr, dash, bend right] & \text{Hydro Power Scheduling} \\
& \text{Scenario Reduction and Generation} \arrow[u, bend right]
\end{tikzcd}$$

Figure 2.1: Overview of Literature Review

2.1.1 Forecasting of Electricity Prices

There are a number of different ways to generate price scenarios for hydro power production. In [8] forecasting is used, by applying an ARIMA (Auto regressive Integrated Moving Average) model to historical data to generate a forecast for the DA and BM price. In [9] historical DA and BM prices was used to generate price scenarios. RKOM was also evaluated by comparing the change in objective value for a model that reserved capacity in RKOM compared to a model that could use all its capacity in DA. In [7] the effect of participating in RKOM was evaluated by comparing the shadow price of different capacities reserved in RKOM.

2.1.2 Scenario Tree Generation and Reduction

The nature of stochastic optimization problems can make them computationally hard to solve and they can be time and memory consuming. [10] uses Bender's decomposition method to a hydro power optimization-bidding model. [11] uses scenario tree approximation, in form of scenario tree reduction, to reduce computational time. Further, the scenario tree reduction is evaluated based on the objective function for different degrees of scenario reduction. They also perform an out sample and in sample analysis to evaluate the solution to the stochastic problem in this paper. This work is based on previous work done in [12]. The scenario tree reduction algorithm is based on work done in [13]. In [8] uses a stage wise scenario generation/reduction algorithm to approximate the sequential clearing of the Balancing Market for one day of operation.

2.1.3 Optimal Bidding for Hydro Power Scheduling Models

Optimal bidding strategies under uncertainties is a widely cover subject in the literature. By using a stochastic model instead of a deterministic model, uncertainties in future prices and inflow can be taken into account. The benefits of using a stochastic model compared to a deterministic model is presented in [11]. Some examples of stochastic bidding models for a price-taking hydro power producers are [14], [15] and [16]. These models uses stochastic programming to take uncertainty of electricity markets into account. They also include detailed modeling of hydro power plants. Constraints such as ramping restrictions, storage balance and capacity limits are included in the model. [16] construct a piecewise linear bidding curve for the hydro power producing in the spot market. A multi stage model is considered in this thesis. There are several works done in litterateur on multi stages models. In [17], a multi stage stochastic model is used for short-term hydro power production over multiple days with stochastic prices. The bidding problem can also be expanded to a multi stage problem by including multiple markets. [18] consider bidding in both the day-ahead market and the intraday market. [8] extends the work done in [14] to include the Balancing market, instead of the Intraday market. As part of the SINTEF project "Balancing Markets and their Impact on Hydropower Scheduling" the mathematical formulation in [8] was implemented as a model in the optimization tool AMPL. The model was further improved in [9], by including penalty functions in the model. A case study of the Tokke-Vinje power plant also was done, where both the DA, BM and Reserve capacity market (RKOM) was include in the model. The profitability of including the Balancing market and RKOM was evaluated.

2.2 Design of the Nordic Energy Market

The Nordic energy market is a common energy market for electricity in Northern Europe. The market comprise of several different sub markets, each presented in Table 2.1. The three main marketplaces for energy trading are NASDAQ OMX, markets organized by the Nordic TSOs, and Nord Pool Spot. At NASDAQ OMX, trading of Futures and Forwards is done. These financial contracts specify quantities, time and location of production. There is no physical deliverance in NASDAQ OMX, only financial agreements[19]. Nord Pool Spot is the marketplace for the physical day-ahead trading, and the Intraday market.[1]. The organization of the balancing market varies from country to country. In Norway, Statnett is the only marketplace for balancing power. The hydro power scheduling model presented in this thesis operates in three different markets. The Day-ahead market, the Balancing market and in the Reserve Capacity market. Each of these markets are described in more detail in this chapter. The model does not participate in the Intraday market or in the futures or forwards market.

Market	Market place	Commodity	Time of bidding		
			(week-1)	(day-1)	(h-1)
Financial Markets	NADAQ OMX	Futures,forwards	-	-	-
RKOM season	Statnett	Capacity	October	-	-
FRR-A	Statnett	Capacity / Energy	Thursday 12.00	-	-
FCR-Week	Statnett	Capacity	Friday 12.00	-	-
RKOM week	Statnett	Capacity	Friday 12.00	-	-
DA	NordPool	Energy	-	12.00	-
FCR-Day	Statnett	Capacity	-	18.00	-
ELBAS	NordPool	Energy	-	-	(t-1)
BM	Statnett	Energy	-	-	(t-0.45)

Table 2.1: The Norwegian Energy Market

2.2.1 The Day-Ahead Market

Elspot is Nord Pool Spot's marketplace for day-ahead trading of physical electricity production and consumption. Elspot is also called the Day-ahead market (DA). Elspot includes all the countries in Scandinavia and the Baltic states. The market is divided into

[1] In the Nordic market the Day-ahead market is defined as a physical market despite the fact that the bids are done the day ahead. The balancing market describe a physical market in a better way since the clearing of this market is based in real time data from the grid[19]

different price areas. The market clears at noon each day, as an auction of marginal pricing. Participants submit a set of price-volume bids for each hour of the following operation day. A supply and demand curve is constructed by linearly interpolation of all price-volume bids. The equilibrium of the curves is the system price of that area and is calculated by NPS (Nord Pool Spot) based on all bids given[20]. All trades are settled at this price. There are several ways to trade in the day-ahead market:

- **Single hour bid** is the most frequent bid at NPS[21]. A market participant specify the purchase and sale for a given hour and given price. The price range is -500 € to 3000 € and the bid may consist of up to 62 price points in this price interval[21]. Bids must be non-decreasing.
- **Block bids** are bids with a duration of more than three consecutive hours. In its simples form block bids are "all or nothing" bids. The whole bid is accepted if the average Elspot price for the period is higher than the bid price. If not, it is rejected. Nord Pool Spot also provides more advanced forms of block bids; these can be studied in[22].
- **Flexible hourly bid** is similar to the single hour bids, but the time of production is not specified. Only consumers in the market can use flexible hourly bid.

2.2.2 The Balancing Reserves and the Balancing Market

The purpose of Balancing Reserves is to resolve the imbalance that occurs in the system within the operational hour and to reduce bottlenecks in the power system. The Norwegian balancing reserves is operated by the TSO and is divided into three different reserve type:

- Primary reserves (Frequency-controlled reserves,FCR)
- Secondary reserves (Automatic frequency restoration reserves, FRR-A)
- Tertiary reserves (Manual frequency restoration reserves,FRR-M)

The Norwegian power system is in balance if the frequency of the system is 50 HZ[23]. If the frequency drops below or increases above this, actions has to be taken to restore balance. A graphical representation of Balancing Reserves is presented in Figure 2.2. The primary reserves is activated if an imbalance occurs and the frequency starts to deviate from 50 HZ. If the imbalance last for more than two minutes the secondary reserves will take over and relive the primary reserves so that these are available for new imbalances in the system. The primary and secondary reserves are automatically controlled. Heavy generators that are spinning due to hydro power production has a certain amount of inertia. This inertia is used to control frequency deviations in the system. Primary and secondary reserves is also known as "Spinning reserves". If the system frequency deviates for more than 15 minutes, the FRR-M is manually activated by the TSO. When activated a producer or consumer has 15 minutes to change their production or consumption according to the bid.

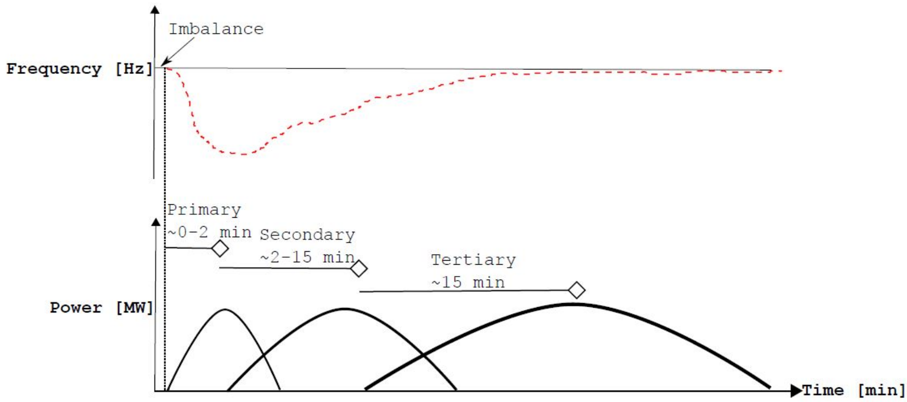


Figure 2.2: Activation of Balancing Reserves [2]

Market Structure of the Balancing Market

FRR-M is used for up and down regulation of production or consumption in the power system. The Tertiary reserve energy or FRR-M is also called the Balancing Market (BM). If the production exceeds the consumption in the power system the frequency will increases above 50 HZ and down regulation is needed. This can be done by either reducing the production or increasing the consumption. If the frequency is below 50 Hz up regulation is needed. This can be done by either reducing the consumption or increasing the production. Both consumers and producers can participate in FRR-M. Bids must be made at least 45 min before real-time and the duration of the bid must be at least one hour[21]. The price of the bid should be an integer divisible with five and should state the number of hours the bid will last. Separated bids for up regulation and down regulation can be given in the market. The bids are given in the different price areas in the same way as bids are given in the DA market. Multiple bids can be given. This will lead to a stepwise bidding curve, unlike the DA bidding curve, which is linearly interpolated between price points. The Balancing Market is in most cases cleared after the marginal price principle. The system price for the regulating hour will be the prices of the last bid needed to get the system in balance.

2.2.3 The Capacity Market(RKOM)

To make sure that sufficient balancing reserves are available for the TSO at any time Statnett, in 2000, introduced RKOM. RKOM is an option market that was made as an incentive for producers and consumers to participate in the reserve capacity market, FRR-M [21]. Currently RKOM only applies only for up-regulation, but both producers and consumers can bid in the market. As can be seen from Table 2.1, there exist two RKOM markets, RKOM week and RKOM season. The time of bids in RKOM season is 05.00-24.00. The bid is made in October and lasts for 6 months. The time horizon for the RKOM week is divided into two periods, day (05.00-24.00) and night (00.00-05.00). The bid last for a week of operation. The bids are also differentiated with respect to quality. "RKOM High quality", RKOM-H, requires full coverage in the whole duration of the bid, while "RKOM with constraints", RKOM-B, allow a down period of maximum eight hours in the time horizon of the bid[24]. The RKOM week bids has to be delivered to the TSO before Friday 12.00 for the following week. The bid state a capacity and a price given in NOK/MW/hour. The producer or consumer commits to bid the reserved capacity into the balancing market, but the producer or consumer chooses the price of the bid.

2.3 Hydro Power Scheduling

There are different tools for decision support and production planning in hydro power scheduling. A common division between these is the time horizon of different models, and their usage. Figure 2.3 describes the hydro power scheduling process divided into phases[3]. Each phase is described in this section.

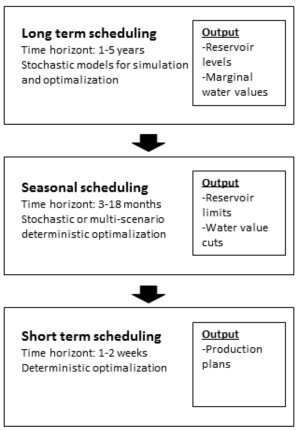


Figure 2.3: Hydro power scheduling hierarchy[3]

2.3.1 Short Term Scheduling

The short term optimization determines the actual operation of the hydro plant for the coming hours or days. The model is meant for decision support for physical operation of a hydro power system; hence, the physical modeling of the system is very important. The goal of the model is to maximize the profit of the operation period, given a stochastic market price and a deterministic inflow. The short term model has to take into account the uncertainty of the prices in the markets it operates in which it operates. The bid into the electricity market is done before the price in the market is known. To take this uncertainty into account a stochastic optimization model often is used. The short-term model gets its boundary condition from the seasonal model in form of water value cuts, described in Section 2.3.1, and initial inflow. Important physical aspects, the model has to concern, is time delay in the system, ramping rates, generators limits and, spillage. The model also has to take into account a lot of non-linearity's like the efficiency of the generators and the bid curves. Consequently, successive linear programming is often used to solve the problem[19].

Water Value Cuts

The coupling between the short term and seasonal scheduling model is done through water value cuts. The expected future income is part of the objective function in the short term model and is given by α. This variable is a function of the reservoir level in the end period of the short term optimization model. This function might be extremely difficult to calculate. It is s a non-decreasing, nonlinear concave function and therefor a piecewise linear approximation of the future income can be used to get the relation between the reservoir levels and the future income. By running the seasonal model for multiple start reservoir levels, x^* different future incomes, α^* and water values, μ^*, are obtained for different initial reservoir levels. These reservoir levels represent the end state of the short term problem and the initial reservoir for the seasonal scheduling. Figure 2.4 is a graphical representation of two cuts. The water value represent the change in the objective function of the seasonal scheduling model by a marginal increase in the reservoir level. This is called the dual value of the reservoir level. By using the calculated future income, water value and initial reservoir level, a tangent line for α can be constructed. The slope of the line is the water value and represent the change in the future income for a marginal change in the reservoir level. This tangent line as a constraint for the future income and is called a water value cut. By constructing multiple cuts for multiple reservoir levels, a piecewise linearization of the future income function is constructed.

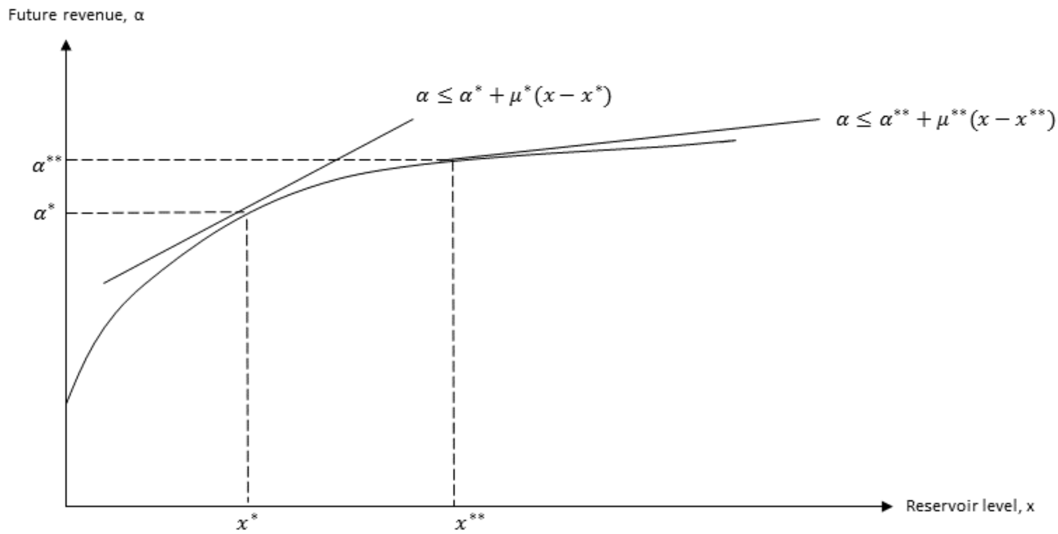


Figure 2.4: Piecewise linearization of future income

2.3.2 Seasonal Scheduling

The seasonal scheduling can be seen as the intermediate step between the long term and short term The seasonal scheduling can be seen as the intermediate step between the long term and short term scheduling. Short term scheduling needs exact reservoirs levels for each reservoir in the system input to the model. In long term, scheduling all reservoir levels are aggregated together and seasonal scheduling is needed to establish a link between the long term and the short term scheduling. The time horizon in seasonal scheduling varies between 3 to 18 months. It describes the same system as the long term model, but uses different mathematical approaches to solve the problem. This gives more exact water values and reservoir levels in each of the reservoirs in the system. These values can be used as input for the short term hydro scheduling model.

2.3.3 Long-term Scheduling

The objective of long-term scheduling is to optimize the use of resource within a time horizon of 1-5 years. Long term scheduling helps the hydro producers with the strategic management of their own resources in interaction with the whole power system. The output of a long term model is future reservoir levels and marginal water values. The input to the model is statically weather data, forecast of demand, planned out-takes and new plants in the system. The future prices are very important for a producers economic results, the model optimizes the producer's resources based on a forecast of these future prices. The EMPS-model is the most commonly used in the Nordic system. The EMPS-model is a stochastic dynamic programming model, which uses heuristic solutions to optimize the reservoir levels in different areas in the model. A challenge with the long term scheduling is physical complexity of the system. To make the model solvable different reservoir levels in a system is aggregated together[3].

2.4 Optimization in Short Term Hydro Power Scheduling

In general, everything in life is to some degree uncertain. Stochastic programming is mathematical programming under uncertainty. A decision has to be taken before a random variable is known. Stochastic programming differs from deterministic programming, where all input data is certain and optimization is done based on known parameters. Bidding in an electricity market is an optimization problem with uncertainties. The clearing of the market, and the market price, is not known at the time of bidding [8]. By using a stochastic programming approach to the problem, this uncertainty can be taken into account. In this thesis a multi stage model, that bids into multiple sequential markets are considered. A brief introduction to multi stage optimization is given in Section 2.4.1. To expand the hydro power scheduling model presented in this thesis, scenario generation and reduction was used. A brief introduction is given in Section 2.4.2 and Section 2.4.3.

2.4.1 Multi Stage Optimization Model

Short-term scheduling under uncertainty can be formulated as a multi-stage stochastic optimization model. The stages in a model represent the flow of information and each stages represent different amount of information. The information is given by the realization of random variables, $\zeta(\omega)$, in the problem. For a short-term problem, these are typically prices and inflow. In a stage, decision variables has to be decided based on the same amount of information[25].

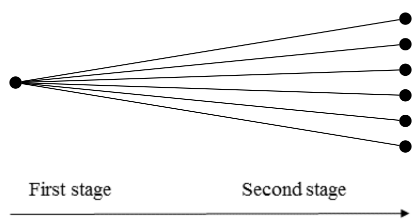


Figure 2.5: Two stage stochastic problem

Bidding into an electricity market in hydro power scheduling can be described by a two-stage stochastic problem. Figure 2.5 is called a scenario fan, and is a graphical representation of a two-stage stochastic optimization problem. The node in the first stage represent the bid made into the market before the price is known. The nodes in the second stage represent the dispatch of electricity. In stage two, the random variable, $\zeta(\omega)$ is known and has S possible outcomes. When there is a finite number of outcomes for $\omega \in \Omega$, these outcomes can be called scenarios. In Figure 2.5 each node in the second stage represent a scenario. Each scenario has a given value of unknown parameters, $\zeta(\omega)$ and a given probability[25]. For the bidding problem, a realization of the random variable corresponds to different prices in an electricity market. The dispatch of water in stage two is done according to different prices in the market. The branches in the scenario fan are called scenario threads, and each represent a determinist realization of the optimization problem. Each thread has a given probability, and the sum of the probability of all scenario threads must be one. In this way the scenario threads is a discrete approximation of reality. By solving each scenario thread deterministically and take the expected value of the solution, it is possible to solve the stochastic problem with uncertainty. This is called the *deterministic equivalent* of the stochastic problem. A two-stage problem can be expanded to a multi stage problem by adding more stages. In the hydro scheduling problem, this could correspond to adding multiple markets or multiple days.

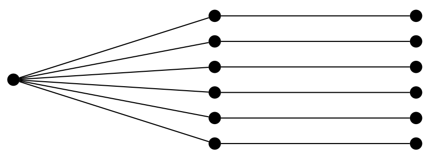


Figure 2.6: Scenario fan of three stage problem

Nonanticipativity Constraints

To ensure that the same decision is the same for all variables in stage 1 in Figure 2.5, nonanticipativity constraints has to be introduced to the problem. Nonanticipativity constraints ensure the information is gradually reveled for each stage in the model. All decision taken with the same information must be the same. The constraint enforces all decision variables in the same node to have the same value. The bracing is done in stages and the nodes in each stage represent a set of scenarios with the same amount of information. The branching structure in the scenario tree represent the flow of information in the problem. For the bidding problem nonanticipativity constraint ensure that the bid done in stage 1 is the same for all realizations of the electricity price in stage two. A scenario fan with nonanticipativity constraints is called a scenario tree. A graphical representation of a scenario tree is presented in Figure 2.7. This is the same problem as Figure 2.5, but nonanticipativity constraints has been enforced to stage two and two and two nodes are branched together.

2.4.2 Scenario Tree Generation

A scenario tree is a discrete approximation of a continuous distribution of uncertain data. It is assumed that a stochastic process can be approximated by a discrete dis-

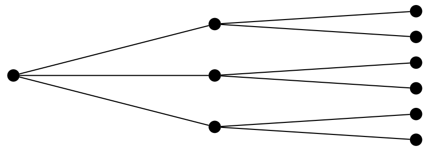


Figure 2.7: Scenario tree of three stage problem

tribution represented by a scenario tree [11]. In literature, there are several ways to generate scenario trees and to approximate the continuous scenario distribution. A short overview of scenario generation methods is presented in [12]. When considering the uncertainty of electricity prices either in the Spot market or in the Balancing market, it is possible to sample from historical data, probability distributions or from price forecasts. The latter is done in [17]. Extensions of ARMA (Auto regressive moving average) models also can be used to generate forecast for spot and balancing prices. In [8], a SARIMA(2,1,0) is used to generate spot prices while an ARMAX(1,0,0) is used to generate BM prices. Another approach is to consider the DA BM price relationship as different states. In the BM no regulation, up regulation or down regulation is possible. This represent three different states. By using discrete Markov chains, it would be possible to model the BM price according to these states. This is done in [26]. By assuming that historical DA prices provide some information about the trend of future prices, these prices can be used to create scenario trees. In [7], historical DA prices are used in the model without sampling, while the BM prices is sampled according to a discrete Markov method. Discret Markov Method is also uesd to sample BM prices in thsi thesis.

2.4.3 Scenario Tree Reduction

If dealing with a multistage stochastic programming, the number of scenarios will increase for each stage in the model.

The number of scenarios can be denoted:

$$S = \omega^n \tag{2.1}$$

S is the number of scenarios, ω is the number of possible realization of the random variable in a stage and n is the number of stages. The number of scenarios is exponentially increased, as can be seen from Equation 2.1, with the number of stages. A large number of scenarios will increase the computational time of the model and can make models with large number of stages or possible realizations of the random variable unsolvable. A method to deal with this problem is called scenario reduction.

Scenario tree reduction is briefly studied in literature. In [8], a stage wise scenario reduction is done for the balancing market. In [11], the benefits of scenario approximation is evaluated. [27] presents a method to reduce scenario tree by comparing probability metrics.

A probability metric is the statistical distance between two random variables and is used to compute probability distances between full and reduced scenario trees. Metrics can be defined in terms of the probability weighted integral (expectation) over a cost function $c_t(\zeta,\tilde{\zeta})$ [28]. Here $t \in T$ is the number of stages in the problem and ζ is the number of scenarios in the original tree, while $\tilde{\zeta}$ is the number of scenarios in the reduced tree. The cost function is a measure of the cost of approximating the full tree with a reduced tree. In this paper, the Wasserstein metric [29] is used to compute difference between scenario trees and to represent the cost function. The metric is given by equation Equation 2.2:

$$c_t(\zeta,\tilde{\zeta}) = ||\zeta - \tilde{\zeta}|| \tag{2.2}$$

The double brackets in equation Equation 2.2 represents the norm of the scenario tree with and without reduction.

An algorithm presented in [13] is based on the work done in [30] and is an algorithm used to reduce a scenario tree of spot prices for a power producer. The algorithm is called Fast Forward Selection and is presented in Algorithm 1. The algorithm uses the Wasserstein metric, Equation 2.2, to calculate the difference between scenario trees.

Algorithm 1: Fast Forward Selection

Step 0

$c_{k,u} = c_T(\xi^k,\xi^u), k,u = 1,\cdots,S$;

Step 1

$z_u^{[1]} = \sum_{k=1}^{k\neq u} p_k c_{ku}^{[1]}, u = 1,\cdots,$S;

Set $J^{[1]} = \{1,\cdots,S\}|\{u_1\}$;

Step i

Compute

$c_{ku}^{[i]} = min\{c_{ku}^{[i-1]}, c_{ku_{i-1}}^{[i-1]}\}, k,u \in J^{[i-1]}$;

and

$z_u^{[i]} = \sum_{k\in J^{[i-1]}|\{u\}} p_k c_{ku_{i-1}}^{[i]}, u \in J^{[i-1]}$;

Choose $u_i \in \arg\min z_u^{[i]}, u \in J^{[i-1]}$;

Set $J^{[i]} = J^{[i-1]}|\{u_i\}$;

Step S-s+1

$J = J^{[S-s]}$ is the index of deleted scenarios;

In step 0, the probability metric for the scenario tree calculated. $c_{k,u}$ is the distance between two scenario pairs. The distance is measured in the difference in the realized value of each scenario. For instance the different in the spot price in two different scenarios. The fast forward selection algorithm finds each scenarios closest neighbor, scenarios where Equation 2.2 is small, and makes a set of close neighbors. In step 1, the probability metric is multiplied with the probability of the corresponding scenarios. In step i, the closest neighbor with the lowest probability is bundled together with the neighboring scenario.$z_u^{[i]}$ is the minimal Monge-Kantorovich distance between the reduced scenario tree u and the original tree k[13].When scenarios are bundled together the probability of the remaining scenario increases to the sum of the bundled scenarios. In step S-s + 1, the deleted scenario is added to a list of deleted scenarios. The probability of the preserved scenarios, q_j, in the new tree is given by equation Equation 2.3,

$$q_j = p_j + \sum_{i \in J(j)} p_i \tag{2.3}$$

where $J(j)$ is the set of deleted scenario that were close neighbors to the remaining scenarios with respect to Equation 2.2. When a sufficient amount of scenarios has been removed, the probability of each of the remaining scenarios is scaled such that the sum equals one.

2.5 Marcov Processes in Hydro Power Scheduling

In short term hydro power scheduling uncertainties in both prices and inflow often is considered. stochastic processes are evolution of random variables over time and can be used to forecast the behavior of inflow or electricity prices. To generate BM prices in the model, a Discrete Markov method was used. This method is based on the properties of a Markov processes and is described in this section

A Markov process predict a future state of a random variable based on the systems present state. This can be used to predict how prices will evolve in a time interval. Let $X_n, n = 0, 1, 2, ...$, represent a stochastic process that can take n different values $i, j \in \chi$. $X_n = i$ is the state i at time n. A stochastic process is a collection of random variables, and their evolution over time.

Assume that there is a fixed probability to go from one state to another and that this probability, P_{ij}, is given by[31]:

$$P\{X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, ..., X_1 = i_1, X_0 = i_0\} = P_{ij} \tag{2.4}$$

Given a state i, P_{ij} gives the probability of reaching state j.

The following conditions also has to be satisfied:

$$P_{ij} \geq 0 \tag{2.5}$$

$$i, j \geq 0 \tag{2.6}$$

$$\sum_{j=1}^{\infty} P_{ij} = 1, for\ i = 0, 1, ... \tag{2.7}$$

The transition probability cannot be negative and the probability of going from one state to any other state including the same state must be one. A three state example has the following probability matrix:

$$\mathbf{P} = \begin{bmatrix} P_{00} & P_{01} & P_{02} \\ P_{10} & P_{11} & P_{12} \\ P_{20} & P_{21} & P_{22} \end{bmatrix} \tag{2.8}$$

In matrix Equation 2.8, the sum of each row must be one. The probability transition matrix is illustrated in Figure 2.8, the diagonal elements of Equation 2.8 is represented by an arrow that goes back into the same state that it came from. This represents the probability to go from one state to antoher.

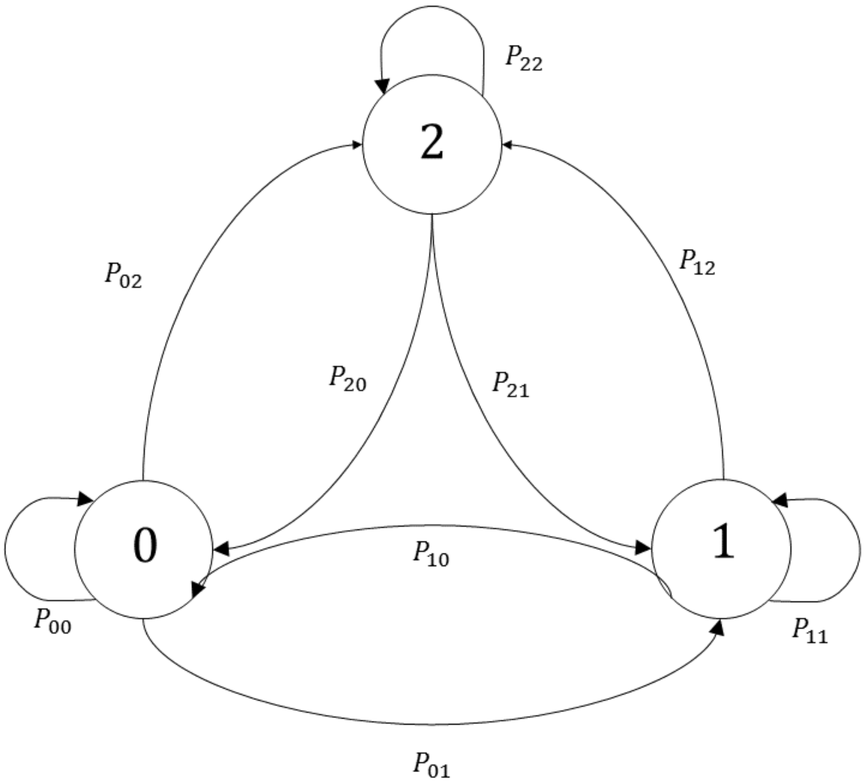


Figure 2.8: Markov chain

2.6 Software Tools

The mathematical formulation presented in Appendix A was implemented in the commercial software AMPL (A Mathematical Programming Language)[32] by Marte Fodstad and Arild Helseth as part of the SINTEF project «Integrating Balancing Markets in Hydro Power Scheduling Methods". The syntax used in AMPL is similar to the standard mathematical notation in an optimization problem. This is a clear advantage. The AMPL used CPLEX to solve the optimization problem. Cplex is an optimization package developed by IMB. The Cplex solver was solved with a MIP gap of 0.0005 and a tree limit of 128 MB. ProdRisk was used to calculate the water value cuts in the model, initial reservoir level and the inflow. It is developed by SINTEF and is used to seasonal and long term hydro power scheduling. ProdRisk is based on Stochastic Dual Dynamic Programming (SDDP) to solve problems [33]. The DMM was implemented in Matlab. This is a high level programming language. The script can be found in Appendix D. To reduce scenario trees ScenTreeGen was used. Scentreegen is a C program that implements algorithms for scenario tree reduction and tree generation. Turid Follestad at SINTEF Energy[28] created it. The implemented algorithms in the current version of the program are based on minimizing a measure of distance between the original tree and the reduced tree, and have been developed by Dupačová et al in [27] and Heitsch and Römisch in [34]. Python was used to generate the input files to ScenTreeGen and AMPL. All scrips and code can be found in Appendix E. The AMPL problem was solved on the computer eelk1656. This belongs to the Department of Electrical Power Engineering at NTNU. This has an AMD Opteron Processor 6174 with 2.2 GHZ and 53.2 GB RAM.

CHAPTER 3

Model Description

One of main objects in this thesis has been to expand a short term hydro power scheduling model with a daily time horizon to a model with a weekly time horizon. This was done to incorporate an additional market in the model, RKOM. RKOM was introduced in Section 2.2.3 This chapter is divided into three sections. Important assumption of the model is presented in the first section. The model that has been expanded in this thesis is described in Section 3.2. This model will be referred to as *AMPLDay*, from here on. The model expansion is described in more detail in Section 3.3. The model expansion will be referred to as *AMPLWeek*.

3.1 Assumptions Taken in Model

A brief overview of assumptions taken in the model:

- All the BM-bids are done in the same stage. In reality this is done sequentially since BM has to be submitted at least 45 min before the given operation hour Section 2.2.2
- For the DA only single hour bids, described in Section 2.2.1 is modeled.
- For the reserve capacity market we only consider bids with "high quality" for the day period between 06.00-24.00.
- The RKOM bid and the DA bid for the first day is modeled in the same stage, even though the RKOM bid happens 48 hours before the DA bid for the first day.

3.2 AMPLDay

In this thesis, a multi-stage, multi-scenario, short-term stochastic model is used to solve a short-term hydro power scheduling problem. The optimization problem is based on the mathematical model presented in[8], and has been implemented in the mathematical programming tool AMPL by Marte Fodstad and Arild Helseth as part of the SINTEF project "Integrating Balancing Markets in Hydro Power Scheduling Methods". The model is a *Mixed Integer Linear Programming* ($MILP$) problem, and all nonlinearities has been linearized. It was reviewed and improved by Caroline Rasmussen and Jakob Boye in [9], and by the author of this thesis in [7]. A more thorough presentation of the model has previously been given in [9] and [7]. In this section the stages, objective function, the market modeling and modeling of the physical structure of AMPLDay is presented. A full mathematical formulation of the model can be found in Appendix A.

3.2.1 Stages in AMPLDay

AMPLDay considers two sequential markets in the Norwegian Power system. The Day-ahead market and the Balancing market. These markets will henceforth be referred to as DA and BM. The time horizon of the model is one day, T^{Tot}, and it is discretized into time step of one hour, T^L. The model consist of three stages and is presented in Figure 3.1 and described in Figure 3.2.1. Each node describes a set of decisions and scenarios in the given stage.

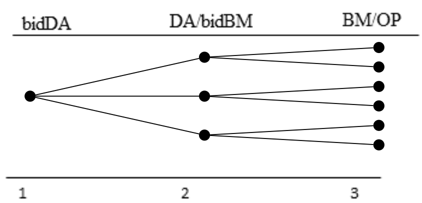


Figure 3.1: Scenario tree AMPLDay

First stage

In the first stage the bid into the DA is done without knowing the actual DA price.

Second stage

In the second stage the DA price is revealed, the dispatch in the DA market is done and the bid into the BM is done without knowing the price at the time of bidding.

Third stage

In the third stage the BM price is revealed and the dispatch in the BM is done. This is the operational stage.

This sequential clearing of markets and gradually revealing of information is formulated as a multi stage stochastic program. The inflow to the model is deterministic and the only uncertainty the model has to take into account is DA and BM prices.

3.2.2 The Objective Function

The objective function maximize the expected future profit of the remaining water in the reservoir after the operation period has ended, and the dispatch in DA and BM for each hour of operation. The cost of spillage and start-up cost for each generator is included in the objective function and minimized. The total objective value is the sum of the expected objective value in each scenario in S, that are the total number of scenarios.

3.2.3 Modeling of the Physical Structure and Hydro Power Operation

The physical structure of the system can be described by Figure 3.2. The solid line describe the flow of water and are variables in the model. These variables are denoted q with either D (Discharge), S (Spillage) or B (Bypass) as superscript. q^v describes the reservoir level. The dashed lines is the inflow to the system, regulated or unregulated. These values are given as parameters to the model. The model consist of $r \in R$ reservoirs and $g \in G$ generators.

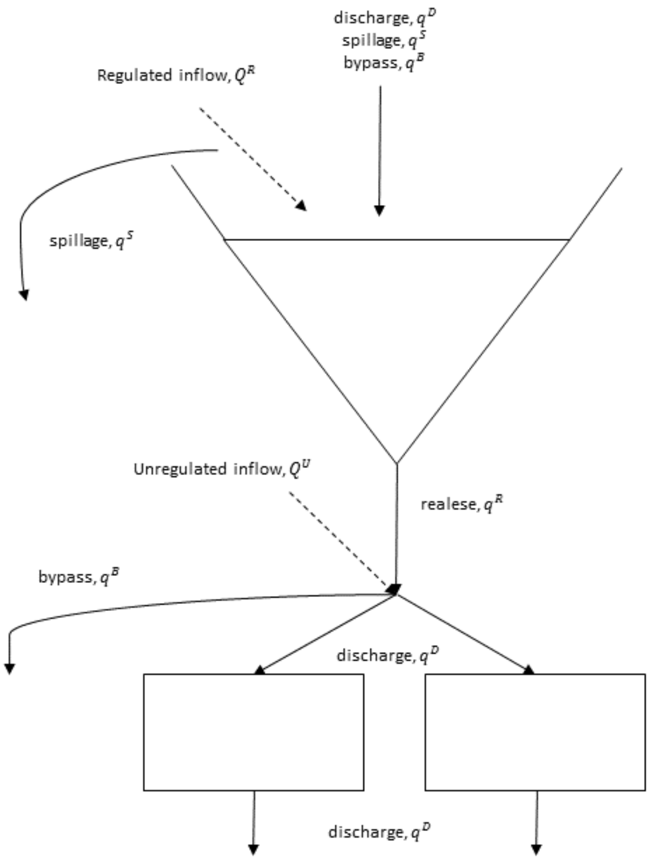


Figure 3.2: Hydro Power model

Reservoir balance,ramping constraints and discharge limits

The reservoir level, q^v, as can be seen from Figure 3.2, is depending on the amount of discharge, spillage and bypass that flows into the reservoir and how much water that is released to the generator. The reservoirs are also coupled in time and are depending on the reservoir level in previous time step. A maximum level for how fast reservoir levels can increase or decrease is given in the model. These constraints are called ramping constraints. Constraints for the maximum discharge to the generators are also given. The model also allow spillage if the reservoir levels are at its maximum.

Generators

The discharged water to the turbines, $q^D_{g,i,t,s}$ produce hydro power, $\omega_{g,t,s}$. The relation between the discharged water and the power produced in the turbine depends on the efficiency of the turbine and is not linear. The relationship is described by a PQ-curve. The curve is piecewise linearized, with different efficiencies for different discharge volumes.

Startup cost for generators in hydro power plants are usually very small[9]. To ensure a realistic production plan, startup costs are added to the objective function. In the model, binary variables are used to define if the turbine in the generators are spinning or not. Binary variables also forces the discharge and production in a generator to be zero if the generator is not spinning.

Time lag

The hydro power system is modeled with a time delay from the water is discharged, or bypassed from one reservoir to it reaches the next reservoir. The time the water spends flowing from one reservoir to another is described by the sets $\mathrm{T}^{\mathrm{D}}_{\hat{\mathrm{r}},\mathrm{t}}$, $\mathrm{T}^{\mathrm{S}}_{\hat{\mathrm{r}},\mathrm{t}}$, $\mathrm{T}^{\mathrm{B}}_{\hat{\mathrm{r}},\mathrm{t}}$, for discharge spillage and bypass.

Inflow

In the model, inflow is considered deterministic and is calculated apriori. Some of the inflow is regulated and some is unregulated. The inflow over the operation period affects the reservoir levels in the model.

Water values

The water values describes the piecewise linearization of the aggregated water value for the end state in each reservoir in the system. The calculation of cuts are described in Section 2.3.1. The expected future income in the objective function is given by piece wise linearized curves of water value cuts. The income is depending on the end state of the reservoirs in the model after the operation period.

Initial parameters

Some parameters in the model has to be set in the model before the model runs. These parameters are the initial reservoir level, initial bypass, discharge and spillage. The initial state of the generators also has to be set apriori in the model.

3.2.4 Modeling of the Markets

In AMPLDay, two different market is incorporated. Bids can be made in either the day-ahead or the balancing market. Additionally, a third market, the reserve capacity market, is included. In this market, bids are made apriori and is not a part of the decision process. Dispatch has to happen in one of these two markets or to a fixed delivery, through bilateral agreements. In this thesis fix delivery is sat to zero.

The Day-ahead market

The day-ahead market's supply and demand curve is, as described in Section 2.2.1, a linear interpolation of all the bids given for an operation period of 24 hours. All bids have to be non-decreasing for the supply curve and decreasing for the demand curve. Bids compris of a specified volume and price for a given hour. To avoid non-linearity the supply and demand curve, of the DA market, has to be piecewise linearized to fit the model. The price can be calculated in advance and it is possible to know apriori if a bid is rejected or not in a given scenario. The bid curve in this model consist of 64 price points with corresponding bid volumes. For DA prices between two price points, the dispatch is the linear interpolation between the bids in each of the two price point. The DA bid curve represent a connection between the first and second stage in the model. The bids into the DA, $x_{b,t,s}^{DA+}$, is done without any information about the spot price, $\tilde{P}_{t,s}^{DA}$, in the first stage. In the second stage the dispatch, $y_{t,s}^{DA+}$ into the market is done. It is also possible for the producer to buy eletricity in the market. Nonanticipativity constraints is included to enforce dispatch to take place after the market clearing[8].

The Balancing market

Similar to the DA bid curve the BM bid curve consist of 64 prices and corresponding volumes. The BM bid curve is not interpolated as the DA curve, but stepwise linearized. For BM prices between two price points, the lowest price point is chosen as price. If the BM price is higher than the DA price, up regulation is possible, and if the DA price is higher than the BM price down regulation is possible. Bids has to be non-decreasing for up regulation and decreasing for down regulation. As for the DA market the bids and the dispatch happens before and after the market clearing, and connects stages in the stochastic model. Nonanticipativity constraints is included to enforce dispatch to take place after the market clearing[8].

The Reserve capacity market

In AMPLDay, the reserve capacity market (RKOM) is modeled as a fixed volume sat apriori, not a decision variable. The capacity reserved in RKOM cannot be bid into the DA. The producer also commits to bid the RKOM capacity into the up regulation BM, but to an optional price. The model only considers RKOM-H bids, implying that capacity bid into the market is reserved between 05.00-24.00 for each day of a week and that this capacity has to be available in BM for these operation hours. Type of bids in RKOM is further explained in Section 2.2.3 In the project thesis written by the author, a RKOM variable (x_t^{RKOM}), and Equation 3.1 was introduced in AMPLDay.Equation 3.1 ensured that the RKOM variable was equal to the commitment. This was done to take the dual value of the RKOM commitment to measure the effect of how it affects a daily production plan. For more information read [7].

$$x_t^{RKOM} \geq X_t^{RKOM}, \quad t \in T \tag{3.1}$$

Energy balance in market

There has to be energy balance in the model. This ensures that the aggregated hydro power produced for a given hour and scenario must be consumed either to a fixed delivery, the DA market, BM market or in the imbalance settlement. In this thesis, the imbalance settlement and fixed delivery is deactivated. In the energy balance, both the DA and BM delivery is represented, hence this creates a connection between the second and third stage in the model. Nonanticipativity constraints is included to enforce dispatch to take place after the market clearing[8].

3.2.5 Nonanticipativity Constraints

To ensure that decision variables in the model with the same amount of information is the same, nonanticipativity constraints are enforce to a set of decision variables in the model. The bid into both DA and BM has to be the same for all nodes with the same information. The same applies for the dispatch in DA and BM.

3.3 Model Expansion to AMPLWeek

In this thesis, AMPLDay, described in Section 3.2, was expanded to a weekly time horizon. The new model was called AMPLWeek. the expansion was done to incorporate the fact that bids in RKOM has a duration of a week compared to DA and BM bids that only lasts for an hour. The purpose of AMPLWeek is to decide the optimal bid into RKOM. In the model, sequential bidding is done in three different markets. RKOM, DA and BM. Since the time horizon of the model is expanded to a week, bidding in DA and BM occurs multiple times in the model. The shifting between DA and BM stages is described in the sub section about stages, in Section 3.3.2 In this section, it is also described how the expansion affects the objective function, the stages and the time horizon of the model.

3.3.1 The Objective Function in AMPLWeek

The objective function includes the income from DA, Equation 3.6, and BM, Equation 3.7, and the future income from water in the reservoirs α_s. It also includes the startup cost of generators and spillage cost for reservoirs. Equation 3.5 is the valuation of final spinning state of the generators. The RKOM is included in Equation 3.8. A deterministic price for the RKOM bid is decided apriori and used as an input parameter in the model. The price is based on historical RKOM prices from [35]. This parameter is denoted $\tilde{P}^{RKOM}$. The objective function for the AMPLWeek is given by Equation 3.5 to Equation 3.8 and is sum of the expected value of all scenarios in the model. $prob_s$ is the probability of each scenario.

$$\text{Objective} = \sum_{s \in S} prob_s(-\sum_{t\in T}\sum_{r\in R} c^{Spill}_{r,s,t} \tag{3.2}$$

$$-\sum_{t\in T}\sum_{g\in G} c^{Start}_{g,s,t} \tag{3.3}$$

$$+\alpha_s \tag{3.4}$$

$$+\sum_{g\in G}(\delta^{Spin}_{g,s,t_{Max}} - \delta^{Spin}_{g,s,t_0}) \tag{3.5}$$

$$+\sum_{t\in T}\tilde{P}^{DA}_{t,s}(y^{DA+}_{t,s} - y^{DA-}_{t,s}) \tag{3.6}$$

$$+\sum_{t\in T}\tilde{P}^{BM}_{t,s}(y^{BM+}_{t,s}, y^{BM-}_{t,s}) \tag{3.7}$$

$$+\sum_{t\in T}\tilde{P}^{RKOM}_{t} x^{RKOM}_{t}, \tag{3.8}$$

$$s \in S, t \in T \tag{3.9}$$

3.3.2 Stages in AMPLWeek

To incorporate reserve capacity in RKOM as a decision variable in the model, additional stages have been added to the model and the time horizon has been expanded. The AMPLWeek has a time horizon, T^{Tot}, of seven days. This equals 168 hours of operation. Since bidding and dispatch into DA and BM is done each day, the bidding sequence in AMPLDay had to be repeated seven consecutive times. Hence, AMPLWeek consist of 15 stages instead of three stages, and it alternates between DA and BM stages.

A graphical representation is presented in Figure 3.3:

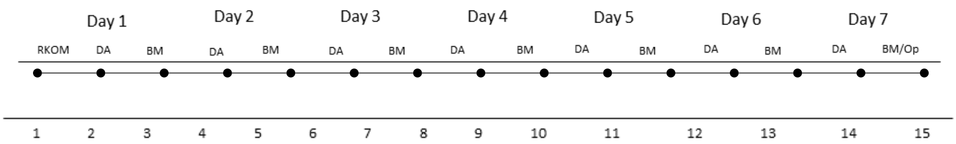


Figure 3.3: Scenario tree AMPLWeek

The nodes in Figure 3.3 represent the 15 stages in the model. In each stage, there are two decision that are taken. The dispatch in the given stage and the bid for the following stage. The operation of the hydro power plant is done in stage 15 after the BM price for the last day of operation is revealed.

An overview of the stages is given below:

Day 1

First stage
The bid into the DA and RKOM is done without knowing the actual DA price. The price of the RKOM bid is known.

Second stage
The DA price is revealed, the dispatch in DA is done and the bid into the BM is done without knowing the price at the time of bidding.

Third stage
The BM price is revealed and the dispatch in the BM is done. The bids for the second day in DA is done.

Day 2

Forth stage
The DA price is revealed, the dispatch in the DA and bid into BM is done.

Fifth stage
The BM price is revealed and the dispatch in the BM and the bid in DA is done.

⋮

Day 7

Fourteenth stage
The DA price is revealed, the dispatch in the DA and bid into BM is done.

Fifteenth stage
The BM price is revealed and the dispatch in the BM is done. In stage 15 the operation for the all 168 hours is executed.

Nonanticipativity constraints in AMPLWeek

Nonanticipativity constraints has to be enforced to some decision variables in each stage in the model. Equation 3.10 and Equation 3.12 enforces nonanticipativity constraint on the bid into the DA and BM. Since the bid is defined for all $t \in T$ a set of Days had to be defined. Each day, d, contains a sub set of hours $T(d) \subset T$. Because each stage correspond to a given period of operation, nonanticipativity constrains has to be divided into periods corresponding to the time horizon of the given stage. The nonanticipativity constrains was also enforce to the down regulation dispatch in BM and buying in DA by Equation 3.13 and Equation 3.11.

Day-Ahead market

$$x_{b,t,s}^{DA+,-} = x_{b,t,s'}^{DA+,-} \qquad , b \in B, t \in T(d), s, s' \in S, d \in Days \tag{3.10}$$

$$y_{t,s}^{DA+,-} = y_{t,s}^{DA+,-} \qquad , t \in d, s, s' \in S, d \in Days \tag{3.11}$$

Balancing market

$$x_{b,t,s}^{BM+,-} = x_{b,t,s'}^{BM+,-} \qquad , b \in B, t \in d, s, s' \in S, d \in Days \tag{3.12}$$

$$y_{t,s}^{BM+,-} = y_{t,s}^{BM+,-} \qquad , t \in d, s, s' \in S, d \in Days \tag{3.13}$$

The nonanticipativity constraints enforces decision variables in the same node to have the same value, meaning that the scenario path for two different scenarios in a node must be the same. Nonanticipativity constraints are further described in Section 2.4.2.

3.3.3 Time Horizon in AMPLWeek

The time horizon of AMPLWeek was increased from $T^{Tot} = 24$ to $T^{Tot} = 168$. Time based parameters given to the model as input, like inflow, $Q_{r,s,t}^{R}$ had to be extended from 24 to 168 hours.

3.3.4 Expansions Done Considering the RKOM Capacity in AMPLWeek

As described in Section 3.2.4, RKOM was introduced in AMPLDay. The time horizon of the RKOM bid, x_t^{RKOM}, is T^{Tot}= 168 hours and the bid has to be the same in the whole time period. The RKOM bid is only active between 05.00 and 24.00 each day. A parameter, X_t^{RKOM} is by Equation 3.14, equal to the maximum production capacity, W^{Tot}, for hours between 05.00 and 24.00 for each day, represented by $T(d)$, and zero for every other hour. $d \in Days$ is the number of days in the model.

$$X_t^{RKOM} = \begin{cases} W^{Tot}, & \text{if: } t \in T(d) \\ 0, & else \end{cases}, \quad t \in T, d \in Days \tag{3.14}$$

Equation 3.15 is an equality constraint enforcing the bids with in a day to be equal to each other.

$$x_t^{RKOM} = x_{t+1}^{RKOM}, \quad t \in T \quad \text{if: } X_t^{RKOM} > 0 \quad and \quad X_{t+1}^{RKOM} > 0 \tag{3.15}$$

To enforce x_t^{RKOM} to be equal for every day of operation, Equation 3.16 had to be included in the model. This force the first hour of operation in each day to be equal. Since Equation 3.15 forces equality between the hours with in a day, all RKOM bids are the same for the whole time of operation.

$$x_t^{RKOM} = x_{t+24}^{RKOM}, \quad t \in Days \tag{3.16}$$

CHAPTER 4

Case Study

This chapter aims to describe the hydro power system used in this model. AMPLWeek was run with data from the Tokke-Vinje hydro power system located in NO2. The psychical structure of the hydro power plant is presented in Section 4.1. A presentation of the model input of DA prices, water values and inflow is done in Section 4.2. A simulation procedure, that includes AMPLWeek, was created in this thesis , the simulation processes is described in Section 4.3. The modeling of scenario trees and generation of BM prices are presented in Section 4.5. A Discrete Markov Method(DMM) was used to generate prices in this thesis. This is described in Section 4.4 The Discrete Markov Method was developet in the project thesis written by the author, but a presentation of the method is also presented in this thesis.

4.1 Physical Structure of Hydro Power Plant

The overview of the Tokke-Vinje hydro power system is presented in the flow chart in Figure 4.1 in addition, the system data is presented in Table 4.1. In the figure the black line represent the flow between reservoirs and the dashed lines represents the spillage and bypass that flows from a reservoir to another. The generators are represented by circles, the reservoirs by a trapezoid. The square represents gates. Gates are used to regulate reservoirs without a generator and are modeled as generators with zero generation. The generation capacity in Table 4.1 is the total capacity for the whole plant.

	Reservoir	Size [Mm^3]	Number of generators	Production capacity [MW]
	Førsvatn	122.0	1	60
	Vennemo	23	-	-
	Songa	638.6	1	120
	Totak	258.0	-	-
	Våmårsvatn	26.2	3	110
	Langeidvatn	31.8	-	-
	Vatjern	0.4	2	2.2
	Vinjevatn	11.2	4	100
	Botnedalsvatn	58.2	1	20
	Byrtevatn	75.5	1	40
	Bandak	86.9	1	16
Total	11	1331.8	14	990.4

Table 4.1: System data of hydro power plant

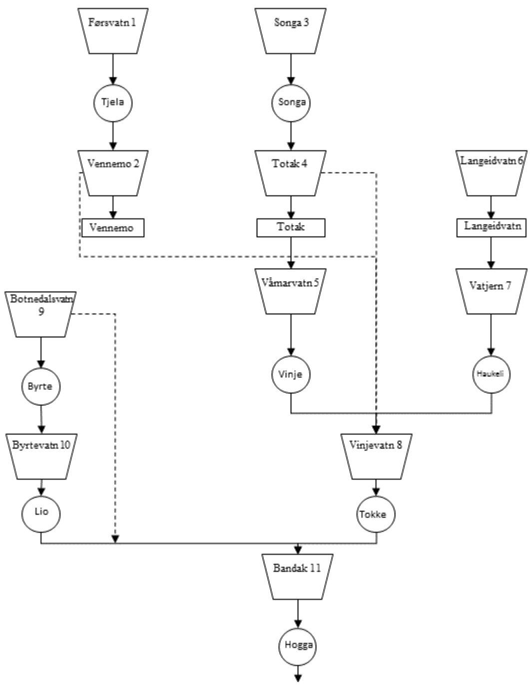


Figure 4.1: A flow chart of the Tokke-Vinje hydro power plant

4.2 Model Input

AMPLWeek has been tested with two sets of input data. The first data set was used to evaluate how different RKOM prices affects the model. The data is presented in Table 4.2 The other data set was used to evaluate seasonal changes in the RKOM bids. The data for this model is presented in Table 4.3. In this thesis, it is assumed that historical DA price is valid to predict future prices. Hence, historical DA prices is used to generate prices scenarios for the future. The BM price scenarios are generated by a Discrete Markov Method (DMM), further described in Section 4.4. The method takes into account statistical properties of historical BM prices in the period 2011-2015 and was created by the author in the project thesis[7].

4.2.1 Data Set 1

The first set of data was used to evaluate how different RKOM prices affects the AMPLWeek model. The scenario was supposed to represent the winter of 2014 and only consider one week of operation. Historical DA prices from five different weeks from the winter of 2014 were used as input.

	Reservoir Level [%]	Average DA
Winter(2014)	75	33.72764396

Table 4.2: Input data to evaluate RKOM prices

4.2.2 Data Set 2

The second set of data was used to evaluate how seasonal differences in AMPLWeek affects the bid in RKOM. Four different periods were considered, each of them representing a different season and each period lasting for four consecutive weeks. Historical DA prices from 2015,2014,2013,2012 and 2011 were used as input.

The following seasonal scenarios where considered:

- **Winter**: This is the depletion season, inflow to the system is low, the reservoirs are still full and the demand for electricity high.
- **Spring**: The end of the depletion season, reservoirs are close to empty, but inflow is rising. The demand for electricity is decreasing
- **Summer**: Inflow is high and the reservoirs are close to full. The demand for electricity is low.
- **Autumn**: The reservoirs is full and inflow is decreasing and the electricity prices are increasing.

The reservoir changes from season to season and the chosen reservoirs levels in Table 4.3 are based on work done in [9].

	Reservoir Level [%]	Average DA Price	Weeks
Winter(January)	75	41.4652619	1,2,3,4
Spring(April	35	35.68432725	14,15,16,17
Summer(June)	90	24.07482738	27,28,29,30
Autumn(October)	85	29.18723357	44,45,46,47

Table 4.3: Seasonal input data

4.2.3 Inflow

The inflow to the Tokke-Vinje hydro power system is calculated by NVE[36]. The data from these calculations has been written into a DETD file by SINTEF Energy Research. A script has been written in Python, by Arild Lote at SINTEF Energy Research to write the inflow to the Tokke-Vinje power plant for a given month and week as an input file to AMPL. Different inflow scenarios were obtained for each week presented in Table 4.3 and Table 4.2.

4.2.4 Water Values and Initial Reservoir Levels

ProdRisk was used to generate water value cuts to the AMPL model. The water values indicates the value of the future income of the water in the reservoir of the Tokke-Vinje power plant. ProdRisk is described in more detail in Section 2.6. A Python script has been created by SINTEF Energy to convert the results from ProdRisk to the input file for AMPL. The most important input to ProdRisk is the week of the year that is simulated, the average DA price for this week abd the reservoir level of the system that is simulated. To create consistency between the ProdRisk and the AMPL model the same initial reservoir levels were used in both ProdRisk and in AMPL. The initial reservoir levels and average DA price for each season is presented in Table 4.3.

4.3 Seasonal Simulation of AMPLWeek

As can be observed from Table 4.3 each season comprises four consecutive weeks. Initial reservoir levels for the first week was taken from Table 4.3. To make a connection between weeks in the model, water value cuts and the initial reservoir levels in AMPLWeek and Prodrisk were based on the end state of the system for the previous week. The end state of the system includes the expected reservoir levels, the expected discharge and spillage for the last hour of operation in AMPLWeek. These values were used as initial values for the AMPLWeek for the next week of operation. The expected value of the aggregated reservoir level for the whole system was also used as input to ProdRisk. This was done so that the water values calculated by ProdRisk incorporates the change in the reservoir levels during a season of operation, and so that AMPLWeek was depending on the water used in the previous week. DA prices used in the model was taken for data set 2. The inflow was obtained from a DETD file described in 4.2.3.

A flow chart of the process of modeling a season is presented in Figure 4.2.

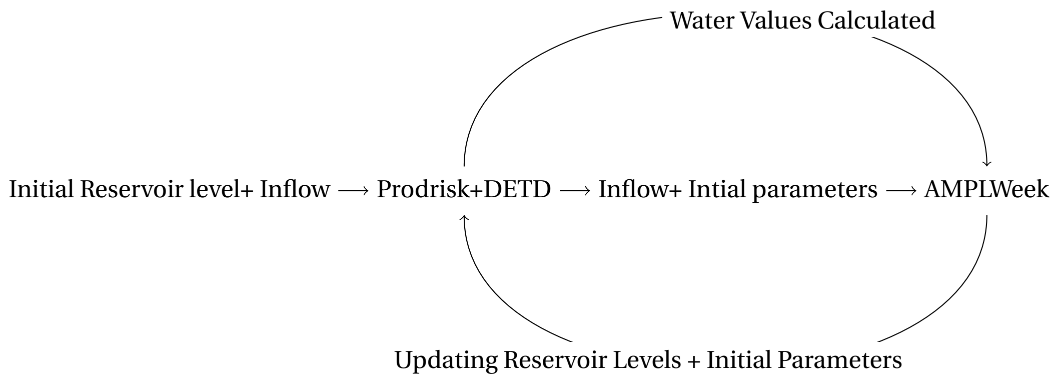


Figure 4.2: Flow chart for seasonal simulation

4.4 Discrete Markov Method

In this thesis, a discrete Markov chain was used to generate BM price scenarios. The theory of Markov chain is briefly described in Section 2.5. As described in Figure 2.2.2, a negative price deviation between the BM price and the DA price for a given hour implies down regulation and a positive deviation implies up regulation. To generate as realistic price scenario, historical balancing market prices for 2011-2014 was observed. The Winter season is in this section used as an illustration for explaining the procedure for generating BM prices . The calculations done for the other seasons can be found in Appendix B.

As an illustrative example the number of hours with deviation from spot, for the winter season, is presented in Figure 4.3.

Statistical property	value[€/MWh]
Mean	-0.74
Standard Deviation	16.74
Median	-0.1

Table 4.4: Statistical propeties of relation between DA and BM prices

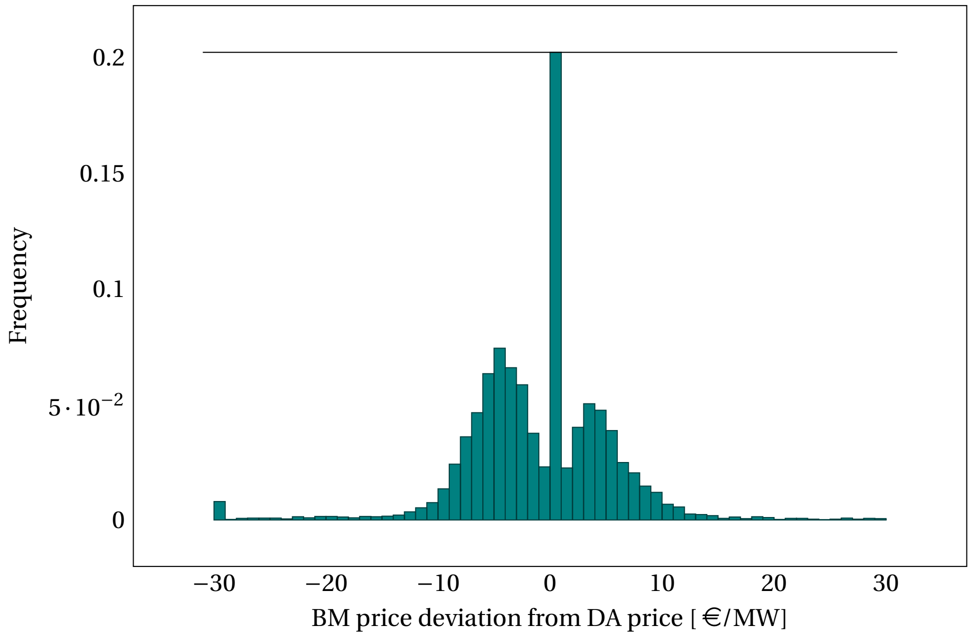


Figure 4.3: Occurrences of BM price deviating from DA price

From Figure 4.3 it can be observed that there are more occurrence of a negative deviation from the spot price than positive. As can be observed from Table 4.4, the mean BM price deviation from the DA price is -0.74. These observations implies that it is more down regulation than up regulation in the Balancing market. The median is -0.1. From Figure 4.3 it can be observed that the BM price deviation from the spot has a negative and a positive peak and that observation seems to be normally distributed around these two peaks. For down regulation the peak value is -five Euro and for the up regulation, the peak is at three Euro. The occurrence of zero regulations is the most command and happens at 20 % of the observations.

The observation from Figure 4.3 can be divided into seven price intervals, called states:

$$
X_{i,t} = \begin{cases} 1, & if: \tilde{P}^{BM}_{t,s} - \tilde{P}^{DA}_{t,s} \leq -30 \\ 2, & if: -29 \leq \tilde{P}^{BM}_{t,s} - \tilde{P}^{DA}_{t,s} \leq -15 \\ 3, & if: -14 \leq \tilde{P}^{BM}_{t,s} - \tilde{P}^{DA}_{t,s} \leq -1 \\ 4, & if: \tilde{P}^{BM}_{t,s} = \tilde{P}^{DA}_{t,s} \\ 5, & if: 1 \leq \tilde{P}^{BM}_{t,s} - \tilde{P}^{DA}_{t,s} \leq 14 \\ 6, & if: 15 \leq \tilde{P}^{BM}_{t,s} - \tilde{P}^{DA}_{t,s} \leq 29 \\ 7, & if: \tilde{P}^{BM}_{t,s} - \tilde{P}^{DA}_{t,s} \geq 30 \end{cases} \tag{4.1}
$$

These seven stages represent seven different BM price scenarios. Down regulation occurs in state 1, 2, 3 and up regulation in state 5, 6, 7. In stage 4 no regulation occurs,hence the BM price and the DA prices is the same. The difference in the BM and DA price is different for all seven stages. The probability of a state is the probability that $\tilde{P}^{BM}_{t,s} - \tilde{P}^{DA}_{t,s}$ is in the price interval given by that stage. Similar work has been done in [9], with the same number of stages, but with a different data set. This leads to different probabilities and different prices. An illustrative example of the winter season is presented in Table 4.5. The data is based on historical differences in BM and DA prices.

State	Price interval[€/MW]		Probability
	From	To	
1	–	−30	0,008
2	−29	15	0,0161
3	−14	−1	0,4609
4	0	0	0,2023
5	1	14	0,2901
6	15	29	0,0103
7	30	–	0,0124

Table 4.5: BM states for the winter period

Periods occurs where both the up regulating price and the down regulating price vary from the spot. In these cases, the direction of the market was chosen to be the direction of price that deviated the most from the spot price. As can be seen from Table 4.5, the probability of down regulation is in this case much higher than up regulation. The state probability for the other season can be observed in the Appendix B Another interesting property of the states presented in Equation 4.1 is the transition probability to go from one state to another, between two hours of operation. A discrete Markov approach was used to define the relationship between the deviations of the BM price compared to the DA price for two consecutive periods. The transition probability is given by Equation 2.4. $P_{ij}(t)$ is the probability of being in state i, given a previous state, j. Equation 2.7 ensure that the transition probability out from one state always is one. By using Equation 2.4 and historical data it was possible to calculate the transition probability from the states in Equation 4.1. This was done using Matlab and the script can be found in Appendix D.

For the winter season the transition probability is:

$$
P_{ij}^{W} = \begin{array}{cc} & \mathbf{X(t+1)} \\ \mathbf{X}(t) & \begin{pmatrix} 0.647 & 0.162 & 0.103 & 0.029 & 0.059 & 0 & 0 \\ 0.073 & 0.62 & 0.241 & 0.029 & 0.029 & 0 & 0.007 \\ 0.001 & 0.008 & 0.885 & 0.0694 & 0.036 & 0.001 & 0 \\ 0.002 & 0.002 & 0.181 & 0.651 & 0.157 & 0.005 & 0.004 \\ 0.002 & 0.002 & 0.050 & 0.109 & 0.819 & 0.013 & 0.006 \\ 0 & 0.012 & 0.012 & 0.049 & 0.366 & 0.366 & 0.195 \\ 0.011 & 0.011 & 0.103 & 0.042 & 0.137 & 0.095 & 0.600 \end{pmatrix} \end{array}
$$

Figure 4.4 is a graphical illustration of the transition matrix for the winter season. The graph plots the probability to go from state, $\mathbf{X(t)}$, to state, $\mathbf{X(t+1)}$, for all possible states. As can be seen from the graph, the diagonal elements of the matrix has the highest probability. This corresponds to staying in the same state for two consecutive periods. Moving from a low numbered state to a high numbered state has the lowest probability; this corresponds to having a negative BM price deviation in one period and a positive BM price deviation in the next. A random draw is done from this matrix in the pseudo-code presented in Algorithm 2.

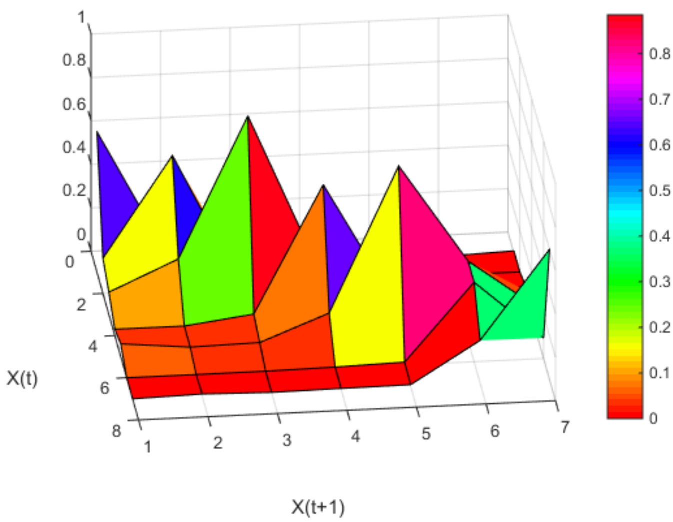


Figure 4.4: Graphical illustration of the transition matrix for the winter

The Matlab script in Appendix D generates the BM price scenarios that are used as input in the model.

A pseudo-code of the script is presented in Algorithm 2.

Algorithm 2: Discrete Markov Method

Data:
Historical DA Prices $\tilde{P}^{DA}_{t,s}$
Transition probability Matrix P^{Season}_{ij}
BM deviation from Spot I_i
Result: BM Prices $\tilde{P}^{BM}_{t,s}$
Initialization;
for $i \in 1..S$ **do**
  $X_{i,1} = I_i$;
for $t \in 2..T$ **do**
  for $i \in 1..S$ **do**
    if $X_{i,t-1} = i$ **then**
      R=random draw from $P_{i,j\in J}$;
      $X_{i,t} = R$;
for $t \in 2..T$ **do**
  for $i \in 1..S$ **do**
    $\tilde{P}^{BM}_{t,s} = X_{i,t} - (X_{Zero,1} - \tilde{P}^{DA}_{t,s})$;

The algorithm generates different BM price scenarios. The input to the algorithm is a set of 24-hour DA price scenarios and the season given as an integer between one and four. The transition probability matrix is calculated apriori for each season in the model. For each DA price given as input to the model 7 BM price scenarios is generated. For each state in Table 4.5, an average deviation from the DA price is calculated based on historical observations for the given season. For the first hour of operation, the BM price is given by the average price deviation for all seven stages presented in Table 4.5. Hence, all seven stages are generated for the first hour of operation. For the next hour of operation, a random draw is done from the transition matrix, P^{W}_{ij}, of the given state in the previous hour. The draw is weighted according to probability of each of the possible transition. The state that is drawn represent the new state of the given scenario. When the state of every scenario for a given hour is updated, the script moves on to the next hour a new random draw to decide if a scenario changes state or not. When BM scenarios for all 24 hours are generated, all BM prices are scaled to correspond to the change in DA prices during the day. For the scaling, a linear relationship between the change in DA prices and BM prices is assumed. If for instance the DA price increase from one to two [e/MWh] then corresponding BM prices also increases from one to two [e/MWh]. The method used is based on theory of discrete Markov chain and will be called the Discrete Markov method (DMM).The full Matlab script can be found in Appendix D

4.5 Modeling of Scenario generation and reduction

In this paper a method for developing scenario trees for AMPLWeek has be used to reduce scenario trees to a manageable size. AMPLWeek uses historical DA prices as input and generates BM prices by using a Discrete Markov Method. AMPLWeek consist, as described in Section 3.3.2, of 15 stages and has a time horizon of 7 days. There are three stages for the first day and two stages for each consecutive day. For each day in the time period of AMPLWeek a scenario tree with a high amount of scenarios was generated and then reduced to a manageable size. In this process, the BM prices were generated based on the DA prices and by doing the generation in a stage wise fashion, the cross correlation between the DA price and the BM price was persevered. Furthermore, a multi stage problem with 15 stages requires storage of a huge amount of temporary variables if the scenario generation were to be done separately from the scenario reduction. By sequentially reducing scenarios, a much smaller amount of temporary variables has to be saved; hence, the computational time and memory usage is reduced. Because the number of scenarios is reduced to the same amount in every second stage, on can for computational reasons allow a higher number of scenarios in each stage, than if the scenario reduction was done after the final stage. In this way it is possible to obtain input data with similar statistical properties as the input to the scenario generation. By doing the generation in a stage wise fashion the reduced scenario tree can be taken into account when generating scenarios for a new stages. In this way auto correlation between the different stages is preserved in a better way. The method of scenario generation and reduction can be described as follows:

Algorithm 3: Stage wise scenario generation

Step 1
Generate a scenario tree for the first day of operation.

Step 2
Extend the scenario three by adding a one-day scenario tree with 35 scenarios to each end node in the excising tree. The DA prices for this one-day scenario tree is scaled to match the DA price in the end node it is attached to; BM prices are generated by DMM.

Step 3
Reduce the scenario three so that the number of scenarios equals the number of scenarios in the first day of operation

Step 4
Extend the scenario tree, in the same way as described in Step 2, to include the third day of operation

Step 5
Reduced the scenario three to the number of scenarios for the first day of operation

Step 6
Continue this procedure for all the days in the weekly model

Algorithm 3 presents the scenario generation and reduction algorithm used in this thesis. A scenario tree with the desired realization of the DA price and the desired realizations of the BM price was generated for the first day of operation. For each scenario in the first day, a new scenario tree corresponding to the first day was created. As described in Algorithm 3, the DA prices is scaled to match the end node of the previous day. This was done to keep the auto correlation between different days of operation. Then the scenario tree was reduced using Scentreegen. Scentreegen is a software developed by SINTEF for scenario generation and scenario reduction. It implement Algorithm 1. The method and algorithm is further explained in Section 2.4.2. The output from the ScentreeGen was a reduced tree with the same amount of scenarios as the first day. The reduced tree was used as input to generate a new set of scenarios and the procedure was repeated. This was done for all the days in the operation period. Figure 4.5 is a graphical representation of how the stage wise scenario reduction was done. The nodes and the filled line represent the scenarios that were kept after scenario reduction. The nodes with the dashed lines represent the scenarios that were reduced by the scenario reduction algorithm. The figure only display the first three days of operation, but the procedure for the remaining days is the same.

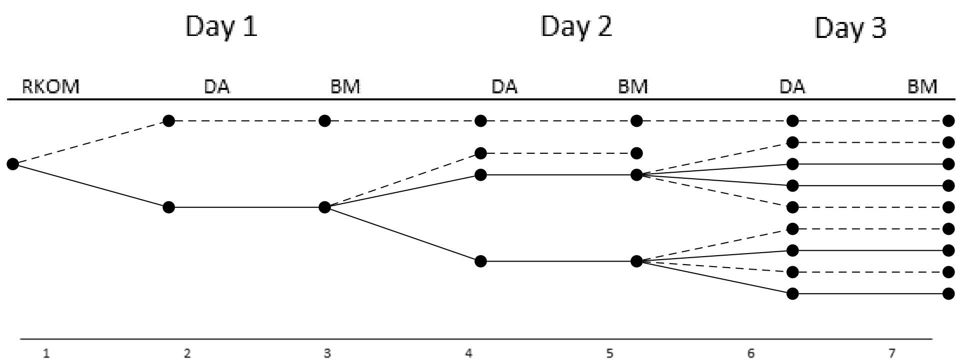


Figure 4.5: Stagewize scenario tree generation and reduction

A Python script was used to create the input files for the scenario tree reduction program. The Python code can be found in Appendix E. The DMM was implemented in Matlab script, but was ran as part of the Python program. The Matlab script can be found in Appendix D.

CHAPTER 5

Results and Discussion

This chapter aims to present and discuss the main findings in this thesis. Four major subject has been studied. First, the evaluation of the stage wise scenario tree generation and reduction method, developed in this thesis, is displayed and discussed. Secondly, AMPLWeek was run with 12 different RKOM prices. The effect of changing the RKOM price on the hydro power scheduling plant has been measured. The results and discussion is presented in Section 5.2. The RKOM prices were based on historical data that can be found in Appendix C. To measure the gains of expanding to a weekly hydro power scheduling model, a comparison of a weekly and a daily model is done in Section 5.3. Finlay, seasonal changes in the reserved capacity in RKOM and other decision variables was simulated in AMPLWeek. The results and discussion is presented in Section 5.4.1. The results were compared to historical data in Section 5.5. The validity of the results are discussed in Section 5.6.

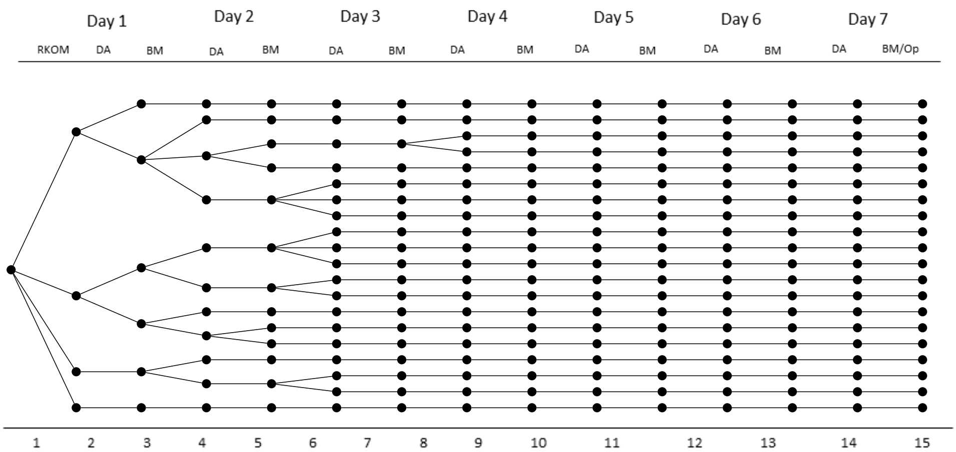


Figure 5.1: Scenario tree

5.1 Evaluation of The Scenario Tree Reduction

In this section the results and statistical analysis of scenario tree reduction performed in this thesis is presented. Data set 1, described in Section 4.2 with 20 scenarios was used to evaluate the scenario tree reduction method. The final tree after scenario reduction is presented. The statistical properties of the DA and BM prices before and after reduction is also analysed. The stability of the objective function in AMPLWeek for different types of scenario trees is also discussed. The expected DA and BM price for a 20 scenario tree is also presented and it can be observed how BM prices generated by DMM behaves compared to the DA price.

5.1.1 Final Scenario Tree After Reduction

Figure 5.1 is a graphical representation of a scenario tree for the AMPLWeek model. The tree has been created by the stage wise scenario generation and reduction algorithm described in Section 4.5. The tree has 15 stages and 20 scenarios. Branching is done twice each day, once for the DA price and once for the BM price. Each node represents 24 price points for either DA prices or BM prices. The scenario reduction algorithm has reduced the number of scenarios from 20^7 to only 20 scenarios. In the final scenario tree, the maximum number of branches is four. This occurs in the DA stages in Day 1. The maximum number of branches for the BM stage occurs in Day 2. Most of the branching can be observed early in the tree and 20 scenarios is reached after Day 4. No branching is done from Day 5 to Day 7. Since there only are eight stages with branching, this is an eight stage stochastic problem, even though the intent was to model a 15 stage stochastic program. The implication of this result is further discussed in 5.6

5.1.2 Analysis and Statistical Properties of Scenario Tree Reduction

In this subsection, the scenario generation of the DA and BM price series is compared to historical prices and evaluated.

	DA Price		
	Mean	Std.dev.	Autocor.
Pre	33.87	3.19	0.86
Post	33.90	2.76	0.88
Historical	33.04	3.41	0.88

Table 5.1: Statistical properties of DA prices

Table 5.1 display the statistical properties of DA prices used in the scenario tree prior and after the scenario reduction. The properties prior to reduction are called Pre, and after reduction Post. Historical spot prices for the winter of 2014 are displayed as comparison. The values are calculated by taking the expected value of all the scenarios in the model and then take the average value over the whole time period, which is a week. The result was as expected because the prices used to generate the scenarios are based on the same historical data as the one used in the comparison.

The historical standard deviation was close, but higher than the standard deviation before the scenario reduction. After the scenario reduction, the standard deviation decreased to 2.76. In the scenario reduction algorithm a large amount of DA scenarios were removed from the scenario tree. This can have resulted in a smaller solution space for DA prices in the final scenario tree, compared to the scenario tree before the reduction. The Post DA prices also had a smaller solution space than historical prices. The first order auto correlation was almost the same for Pre, Post and historical. This might indicate that the variation of the DA prices from one hour to the next is similar for both Pre, Post and Historical prices. Because historical prices curves are used to generate the scenario tree used in the AMPLWeek model, this result was as expected.

		BM Price	
	Mean	Std.dev.	Autocor.
Pre	32.51	3.25	0.55
Post	32.34	3.35	0.68
Historical	31.82	5.03	0.62

Table 5.2: Statistical properties of BM prices

Table 5.2 is the statistical properties of the BM prices. By comparing Pre data with historical data, a higher mean was obsberved. The standard deviation was similar for the Pre and Post prices, but were considerably higher for the historical prices. The result implies that the solution space of the generated BM prices was smaller than for historical prices. The first order auto correlation increased to 0.68 after scenario reduction. This implies that the Post prices varies less from hour to hour than the Pre prices. A reason for this behavior could be that there are much fewer scenarios in the Post tree than in the Pre tree. Hence, the first order auto correlation increases, since there are less prices to take into account. The BM prices Pre and Post was generated by the DMM-algorithm. The differences between the historical data and the prices generated by DMM implies that the BM prices that was generated by DMM, have a higher price with less variation than the historical data. The difference in the first order auto correlation on the other hand, implies that the prices generated by DMM varies more from one hour to another than the historical data. After scenario reduction, the first order auto correlation of the prices increased close to the auto correlation of the historical prices. The standard deviation also increased marginally. It should be noted that DMM used historical data from 2010-2014 to generate prices. Consequently, prices generated by this method will vary from the historical data used as comparison, because this data was collected from 2014, only.

By just taking the mean value of the first order auto correlation, some information is lost. A box and whisker plot of the first order auto correlation of the DA and BM prices is presented in Figure 5.2 and Figure 5.3.

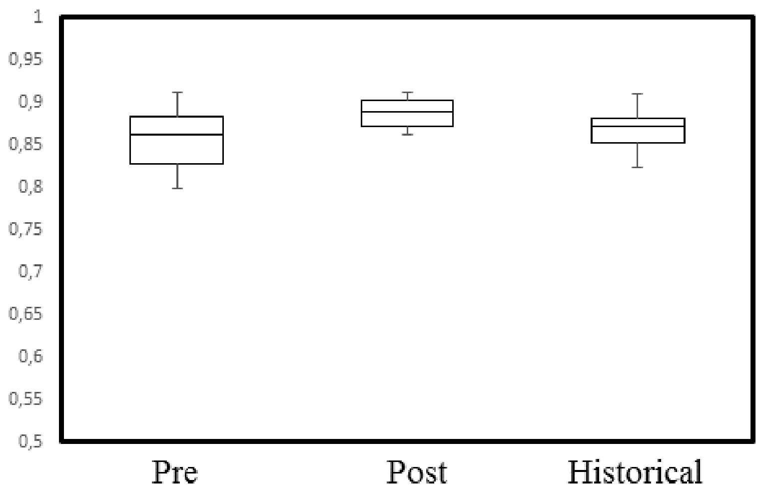


Figure 5.2: First order auto correlation of DA prices

As can be seen from Figure 5.2, the first order auto correlation of the DA prices had a very narrow distribution. The median increased after the scenario reduction and the second and third quartile were in a smaller price range than before reduction. The box plot for the Post prices was similar to the historical data.

Figure Figure 5.3 represent the auto correlation of the BM prices. It can be observed that the distribution had a lower auto correlation than the DA prices. The Pre BM prices is skewed below the median. After scenario reduction, the second and third quartile were in a much narrower price range than for the Pre prices. This comply in a higher degree with the first order auto correlation of the historical prices. This might be due to the fact that there are fewer samples in Post.

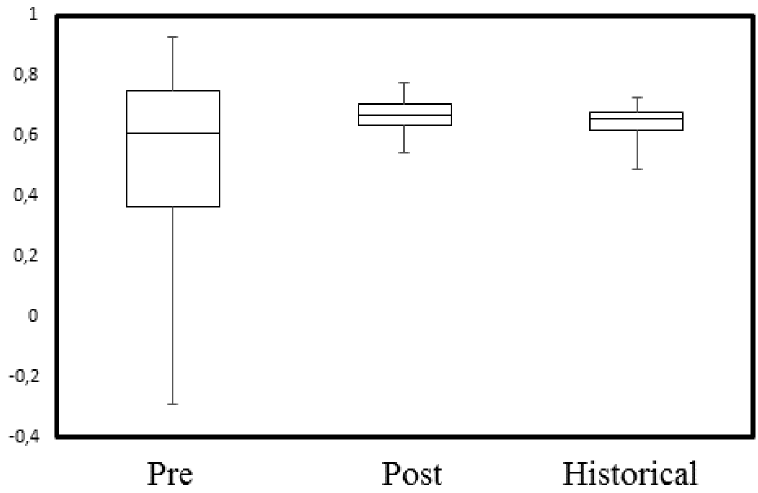


Figure 5.3: First order auto correlation of BM prices

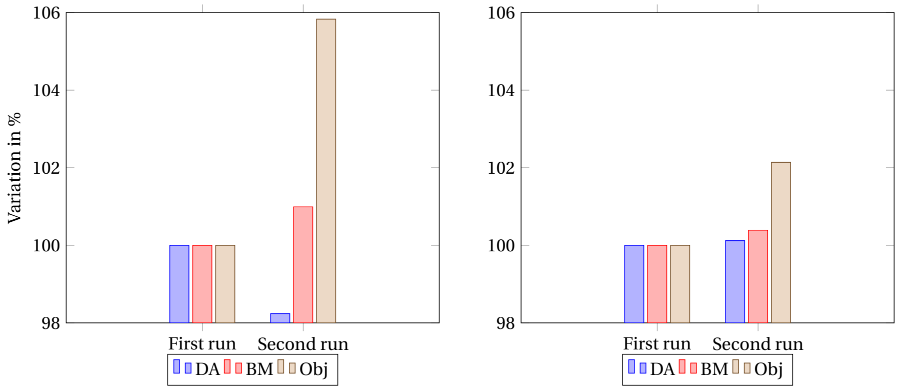
	Crossco.
Pre	0.31
Post	0.54
Historical	0.41

Table 5.3: Cross correlation

The cross correlation between the DA prices and the BM prices is presented in Table 5.3. The cross correlation for the DA and BM prices after scenario reducing was higher than the historical data. The results shows that there might be some correlation between DA prices and BM prices after scenario reduction. For historical prices and for prices before scenario reduction the correlation is weaker.

5.1.3 Stability of the Stage Wise Scenario Generation Reduction Algorithm

The scenario tree generated by the stage wise scenario generation and reduction algorithm presented in Section 4.5 was done with the same input data multiple times to test the stability of the scenario generation algorithm. Two scenario trees with 20 and 35 scenarios was tested in AMPLWeek. The results are presented in Figure 5.4. In the Figure, DA and BM is the average DA and BM price for the scenario tree for the whole operation period. Obj is the objective value from AMPLWeek.



(a) 20 Scenarios

(b) 35 Scenarios

Figure 5.4: Stability of Objective value, DA and BM prices

As can be observed from Figure 5.4, DA and BM prices and the Objective value varies more between the two runs for 20 scenarios than with 35. Scenario generation samples from a set of possible outcomes of BM prices. Consequently, the probability of getting the same scenario tree twice is very small. A tree with 20 scenario compared to 35 scenario are a result of fewer samples for a distributing of possible outcomes. Fever samples implies that the stability of the results depend more on the specific sample than on the general distribution of BM prices. By including more samples, the scenario tree is more stable because more samples converge towards the general distribution of the BM prices used in DMM. This might explain the results in Figure 5.4.

5.1.4 Evaluation of the generated DA and BM Price

The DA and BM prices used in AMPLWeek is presented in Figure 5.5. This is the prices generated by the scenario reduction algorithm presented in Section 4.5. The input to the scenario tree generation was historical DA prices from five different weeks in 2014. BM prices was generated by DMM.

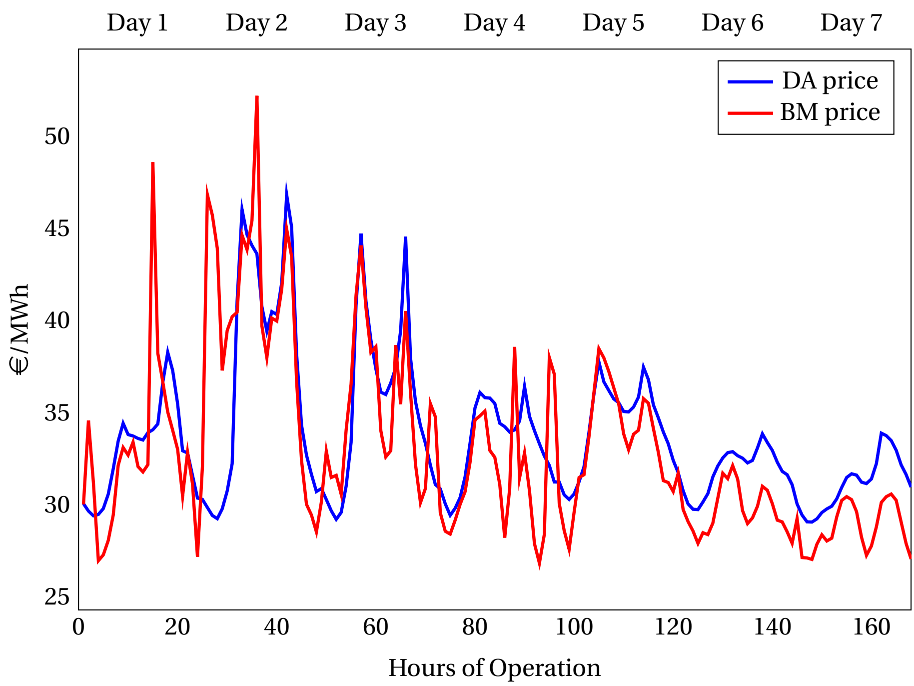


Figure 5.5: Weekly DA and BM prices

The average price for each day is presented in Table 5.4. It can be observed that the DA price, on average, was higher than the BM price in every day, except the third day. Both the DA and BM price increased from Day 1 to Day 2 and then decrease towards the end of the week. The peak in average price for both DA and BM was Day 2. The lowest price can be found on Day 7. A higher spot price can be observed for the week days than for the weekend, this complies with historical observations [1].

	Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7
DA price [€/MWh]	33.2	36.86	35.35	33.62	34.63	32.27	31.39
BM price [€/MWh]	32.3	35.68	35.57	30.93	33.66	29.56	28.69

Table 5.4: Average DA and BM price for a week

5.2 AMPLWeek Results and Discussion for Different RKOM Prices

AMPLWeek was run with 12 different RKOM prices to observe how variables in the model changes with different RKOM prices. All RKOM prices are based on historical data that can be found in the Appendix C. In this section, the input to AMPLWeek was Data set 1, described in Section 4.2 with 20 scenarios. The effect of a change in the RKOM price on the objective value is presented. The effect of an increased RKOM price on the reserved capacity in RKOM and the average power dispatch in DA and BM has been observed. The change in aggregated volume in the hydro power system for different prices also were observed and discussed. An analysis of how the RKOM price affects the up regulation dispatch in BM is also presented.

In this section, the term EFI is used for *the expected future income* and is the value of the water that is left in the reservoir at the end of the operation period. It is also assumed that the reader is familiar with the term elasticity in this section.

5.2.1 Effect of RKOM price on Objective Value

Figure Figure 5.6 is an graphical representation of the changes in all components in the objective function for 12 different RKOM prices. The same results can also be found in Table F.1 in Appendix F.

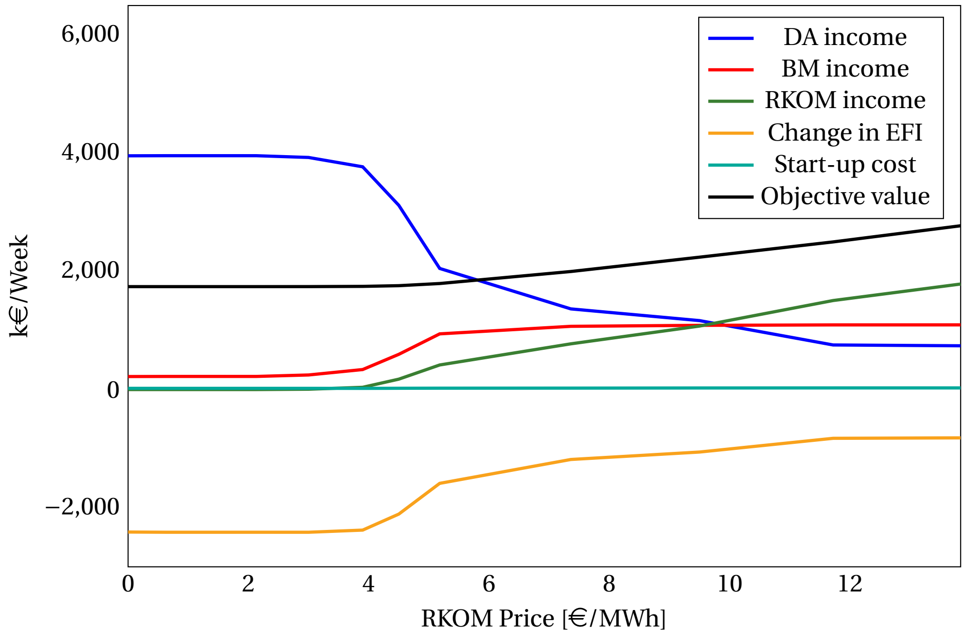


Figure 5.6: Change in objective Value with Increased RKOM Price

The results show that the objective value increased with an increased RKOM price. The objective function comprise of the expected income from the DA, BM, RKOM, the expected start-up cost and the change in EFI (Expected Future Income). The reservoir level at the beginning of the operation period is the reference value for EFI. Consequently, the change of this value is negative. As can be seen, reserving capacity in RKOM had multiple income effects in the objective function. The decrease in the DA income was steeper than the increase in BM income. Since the DA and BM prices were the same for all RKOM prices, this means that the producer relatively dispatch less in the DA than it produces more in the BM for a higher RKOM price. This can be observed by the increase in EFI, meaning that the total production was reduced. The producer increased its objective function both by the payment for reserving capacity and the change in EFI.

The results were as expected. By making RKOM more profitable by increasing the price, the producer wants to reserve more capacity in this market. The reduction of income in DA was compensated by an increase in income from the BM and the payment for reserving the capacity and the increase in future income from the water in the reservoirs. Even though the total increase from BM and RKOM was lower than the reduction in DA, the objective value increased because of the increased income from the EFI. The water value increased because the total production decreased, resulting in that the reservoir levels were relatively higher at the end of the production period than without reserving capacity in RKOM.

5.2.2 Effect of RKOM Price on Average Power Dispatch and Reservoir Levels

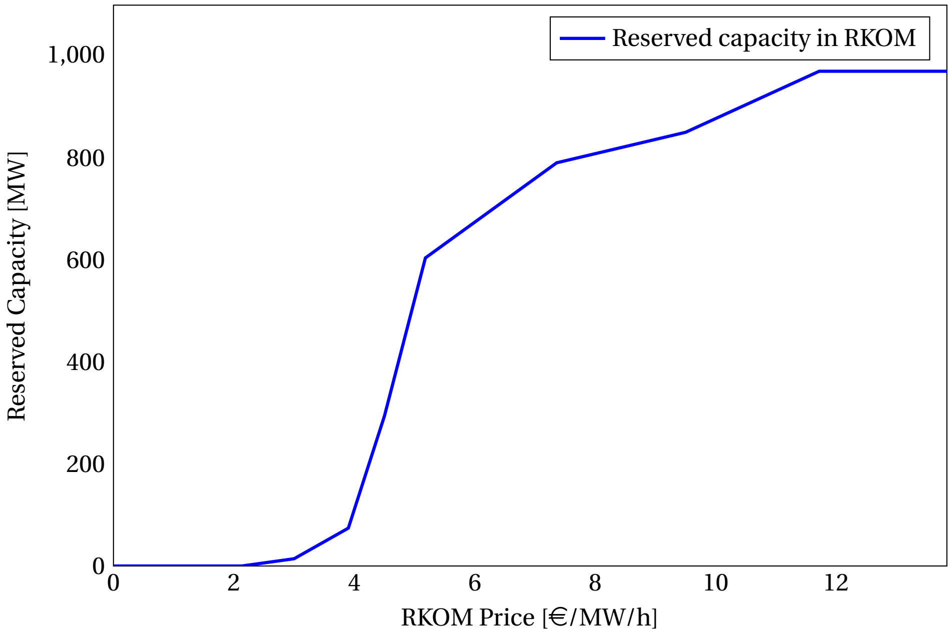


Figure 5.7: Price curve for RKOM

Figure 5.7 shows different capacities reserved in RKOM for different RKOM prices. The results shows that there is a nonlinear relationship between the RKOM price and the expected reserved capacity in RKOM.

As a profit-maximizing agent, the producer wants to increase the capacity reserved in RKOM if the price it receives for this increases. As mentioned in Section 5.2.1, a marginal increase in the RKOM price leads to both an increase in payment for reserving the RKOM capacity and an increase in the EFI. AMPLWeek also comprise of several different binary variables, so different prices can have binary variabels with different values. This might be the reason for the exponential growth and non linear relationship in RKOM volume for a higher price.

Figure 5.8 shows the price curve for DA, BM Up and Down and total expected power dispatch in AMPLWeek for different RKOM prices. The expected power dispatch in the BM comprise of both up regulation and down regulation. The elasticity for the different power dispatches changes for different prices, but are inelastic for prices below 3 €/MW/h and above 11.72 €/MW/h.

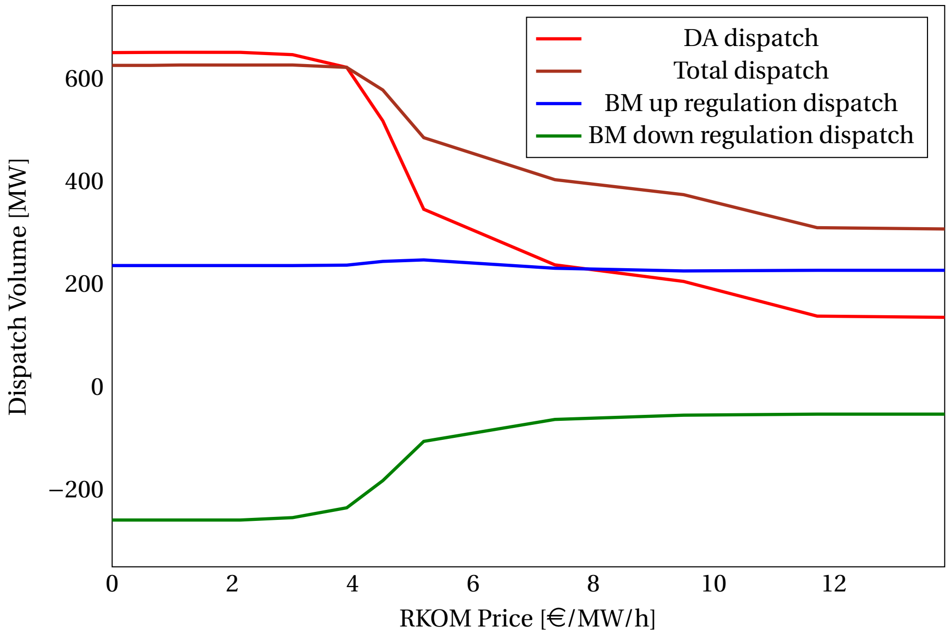


Figure 5.8: Change in DA and BM dispatch volume as function of RKOM Price

Observations from Figure 5.8 show that the reduction in the expected DA production decreased more than the expected down regulation in BM decreased. The results also show that the up regulation in BM is almost unaffected by the RKOM price, compared to the down regulation. For RKOM prices between 6 €/MW/h and 12 €/MW/h the BM down regulation dispatch shows an almost inelastic behavior, while the DA power dispatch is negatively elastic. The total dispatch in the model decreases with a higher RKOM price.

The main purpose of RKOM is, as described in Section 2.2.3, to create an incentive for hydro power producers to participate in the up regulation BM. A possible way to measure the effect of RKOM would be to observing how up regulation in BM changes with increased RKOM price. The results from Figure 5.8 shows that the down regulation BM is affected in a higher degree than the up regulation BM by an increase in RKOM price.

A table with the same data as presented in Figure 5.8 can be found in Table F.2 in Appendix F

Figure 5.9 only display the up regulation dispatch in BM from Figure 5.8. As can be observed, the expected dispatch increases for RKOM prices between 3.9 €/MW/h and 6 €/MW/h. For prices higher than 4.9 €/MW/h, the dispatch decreases below the dispatch without including RKOM. This result was not as expected. On might expect an increasing or equally average up regulation dispatch for a higher RKOM price, since more capacity is reserved for the BM. The results on the other hand implies a non-linear relationship between the increase in RKOM price and the average up regulation dispatch in

BM. AMPLWeek comprise of several binary variables, the generator state for instance, that can cause a result like this.

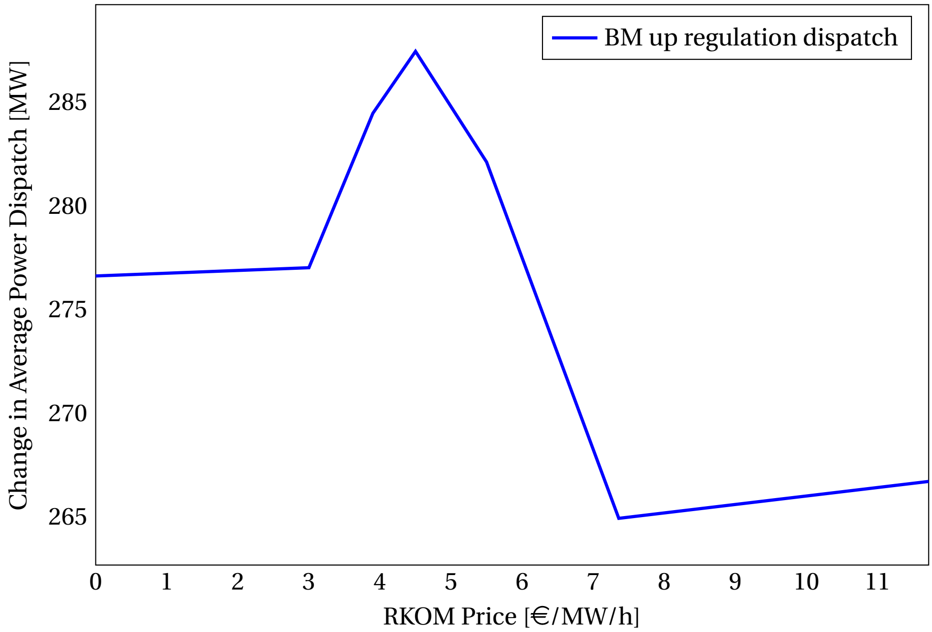


Figure 5.9: Change in up regulation dispatch for different RKOM prices

RKOM was, as described in Section 2.2.3, introduced to create an incentive for hydro power producers to increase their up regulation dispatch in the BM. The result implies that participating in RKOM not necessarily has the desired effect on the hydro power producer, as was the intent from the TSO when creating RKOM. A higher price in the market might lead to less participation in up regulation BM. The result is to some extent counter intuitive. It is hard to know if the results are an exception case or is a general observation for RKOM based only on a case study. It might also be the result of model simplifications. This is further discussed in validity of results, in Section 5.6.

The reason for this model behavior is not fully known and could be further exploder. Binary variables like the generator setting might give nonlinear relationships between power dispatch and RKOM price. The capacity reserved in RKOM reduce the power dispatch in DA. When there is need for up regulation in BM, the hydro power producer can dispatch more in BM than in DA. The relative difference in dispatch between the markets depend on the reserved capacity. If a large part of the maximum capacity is reserved, fewer generators will operate in DA. If there is a need for up regulation and the planned BM dispatch is higher than the DA dispatch, More generators has to be activated. Activating these generators have a cost. If the cost of activating generators are too high the consequence might be that participating in BM for up regulation is less profitable when a lot of capacity is reserved in RKOM. Depending on the reserved capacity in RKOM different generator settings would be optimal and this might give the non linear relationship observed in Figure 5.9. The impact of reserved capacity on generator states is not study further in this thesis. A recommendation for further work is given in Chapter 7.

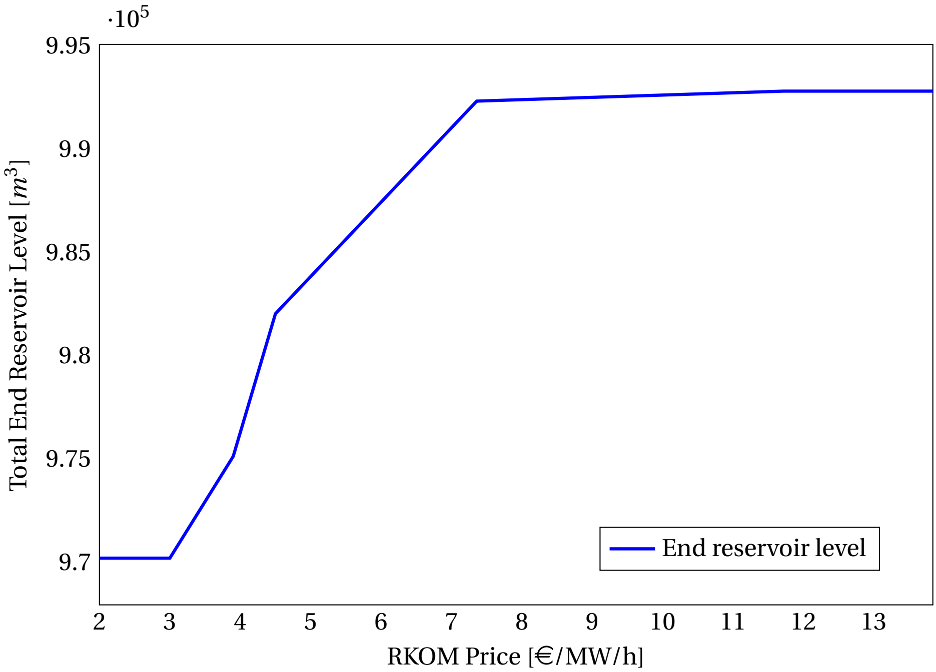


Figure 5.10: Aggregated end reservoir level for different RKOM prices

Figure 5.10 presents the aggregated end reservoir levels for different RKOM price. As can be observed, the end reservoir level increases with a higher RKOM price. This result was as expected. The RKOM price compensates the hydro power producer for not participating in the DA and by observing Figure 5.8 it can be seen that the reduction in the average dispatch in DA is higher than the increase BM volume with respect to increased RKOM price, resulting in a lower total production.

5.3 Evaluation of a Weekly Time Horizon Compared to a Daily Horizon

To evaluate how AMPLWeek performers compared to a model with a time horizon of one day, Equation 3.16 was removed from AMPLWeek and the model was run with the same price input as in Section 5.2. This constraint forces the RKOM bid, x_t^{RKOM}, to be the same for each day of the operation period. By removing this, the reserved capacity in RKOM could change from day to day, and the daily optimal reserved capacity in

RKOM could be reserved. The model without the constraint will be called AMPLWeek RKOM Free and with the constraint AMPLWeek RKOM Fixed.

Another approach to evaluate how AMPLWeek performes compared to a model with a time horizon of one day would be to use the same input parameters in AMPLWeek and AMPLDay, described in Section 3.2. The optimal reserved capacity in RKOM could be obtained from AMPLWeek. In AMPLDay the reserved capacity in RKOM is a parameter. By using the optimal reserved capacity from AMPLWeek, AMPLDay could be run for the same input data as AMPLWeek and the results could be compared. The results from AMPLWeek and seven days of simulation with AMPLDay is not necessarily comparable. Scenario trees in AMPLWeek is generated for a week and to run AMPLDay with the same information as AMPLWeek for different days is not a trivial task.

Figure 5.11, Figure F.3, Figure F.2 and Figure F.1 compare the fixed and free reserved capacity in RKOM for four different RKOM prices. Table 5.5 shows the change in the objective function for the model with and without the fixed RKOM constraint. A positive change means that the value has increased for the model without the constraint and negative values means that there has been a decrease in value from AMPLWeek RKOM Fixed to AMPLWeek RKOM Free.

As can be seen from Table 5.5,the total objective function was increased for all RKOM prices. By relaxing the constraint Equation 3.16, the model can optimize the reserve capacity in RKOM based on the DA and BM prices for that day. This behavior could be observed in Figure 5.11. RKOM Free display the RKOM bid without constraint Equation 3.16 and RKOM Fixed with the constraint. As can be observed, the RKOM Free changes from day to day while the RKOM Fixed is constant for the whole time of operation. By comparing Table 5.4 and Figure 5.11, it can be observed that there is a negative correlation between a high DA price and the volume reserved in RKOM. The difference between the DA and the BM price also affects the bid in RKOM. When capacity is reserved in RKOM it cannot be produced in the DA market, it has to be produced in the BM or be saved resulting in a higher EFI.

	Zero	Low	Average	High
RKOM Price [€/MW/h]	1.13	3	4.5	11.72
Start cost	0	-0.27	-1.93	-0.31
Day-ahead income	-1.11	-8.44	-225.2	174.85
Balancing market income	-0.36	-3.78	115.11	-1.92
Change in EFI	1.48	11.84	94.31	-97.14
RKOM income	0	0.99	34.86	-69.5
Objective value	0.01	0.88	19.98	6.59

Table 5.5: Change in objective value for different RKOM price [k€/Week]

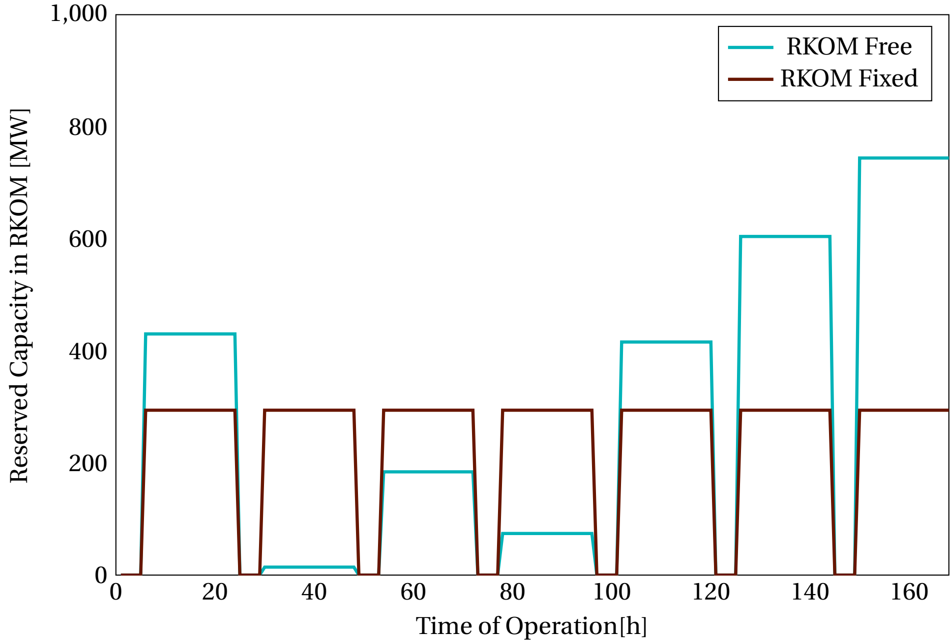


Figure 5.11: Fixed and Free RKOM volume for RKOM price= 4.5 €/MW/h

Figure F.3, Figure F.2 and Figure F.1 can be found in Appendix F and display similar results The Figures show shows that the effect of the Free constraint is highest when the reserved capacity in RKOM is now very high or very low.

Figure 5.12 compares the change in DA, BM and total average dispatch, in addition to the RKOM capacity for different RKOM prices. As can be observed, the change in dispatch varied with RKOM price and there was a clear shift in behavior for prices above and below five €/MW/h. For RKOM prices below 5 €/MW/h the total dispatch decreased and an increase in the EFI could be observed. The relative dispatch in BM increased, but decreased in DA. The reserved capacity in RKOM increases marginally. The model behavior might indicate that the hydro power producer is willing to bid into RKOM for a lower RKOM price with RKOM Free For RKOM prices above 5 €/MW/h the total dispatch increased and a decrease in the EFI could be observed. The relative dispatch in BM was unchanged, but the dispatch increased in DA.

When prices increase above 5 €/MW/h the DA and total dispatch increases. By observing Figure 5.7 it can be seen that the capacity reserved in RKOM is high for price above five €/MW/h. The RKOM capacity restricts the dispatch in DA to a higher degree for high prices than for low. By introducing RKOM Free this restriction is relaxed, resulting in a higher dispatch in DA and lower capacity reserved in RKOM.

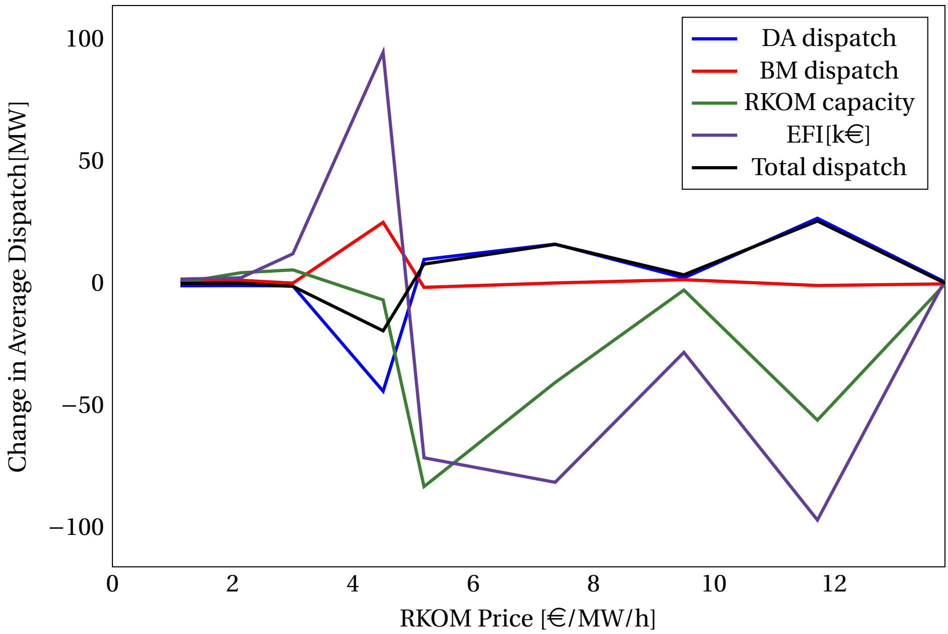


Figure 5.12: Change in average dispatch with free RKOM volume

Figure 5.13 present the change in the objective value for different RKOM prices between RKOM Fixed and RKOM Free. An interpretation of this is that the effect of a free RKOM bid is larges when the possible variation between days is large. For a very high RKOM price, the producer wants to reserve as much capacity as possible in RKOM each day in the model. Hence, the change between days is small. For a very low price, it is not profitable to reserve capacity in RKOM. Hence, the change between days are also small her.

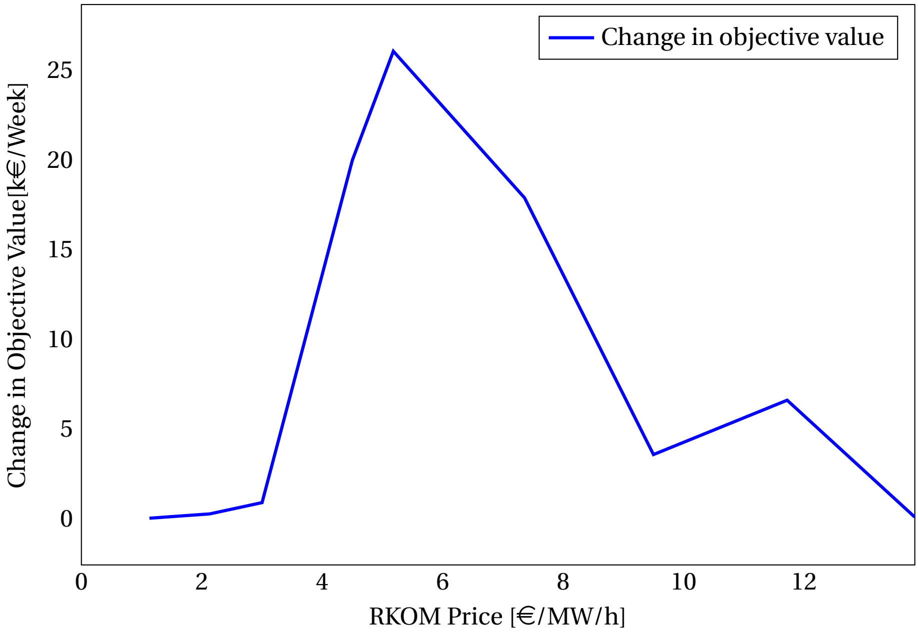


Figure 5.13: Change in objective value with free RKOM volume

A higher objective value for AMPLWeek RKOM Free than RKOM Fixed might indicate that modeling RKOM with a daily time horizon gives results that overestimates the profitability of bidding into RKOM.The difference in objective value between Free and fixed capacity is highest for RKOM prices corresponding to volumes with a high degree of flexibility. That is when the reserved capacity is not close to zero or the maximum capacity of the hydro power plant. It can also be observed that it is profitable to reserve capacity in RKOM for lower prices in AMPLWeek Free than in AMPLWeek Fixed. By running AMPLWeek, a lower a more realistic objective value is obtained. The results show that by using AMPLWeek a hydro power producer will obtain a more realistic and pessimistic forecast of the profitability of reserving capacity in RKOM, compared to a model with a daily horizon.

5.4 A comparative Analysis of the Seasonal Effects on the Reserve Capacity Market

In this section, a comparative analysis of how seasonal changes affects the reserved capacity in RKOM is done. A seasonal simulation, described in 4.3, was done for four periods. RKOM price of 4.5 €/MW/h was used in all periods to measure the effect of seasonal changes. The simulation procedure uses data set 2 to simulate different seasons. Data set 2 is described in 4.2.2.The results of the scenario generation and simulation of four consecutive weeks of operation considering water values and reservoir levels are presented in this section. The seasonal changes in the objective value, the dispatch in DA and BM, and reserved capacity in RKOM is also presented and discussed. For week 17 AMPLWeek was infeasible, consequently the values used for this week in the results are the same as for week 16 week in the spring season. This week is also marked black in the graphical representations.

5.4.1 Seasonal Differences in Input Prices and Reservoir Level

In this subsection, the DA and BM price curve and the reservoir level for four different seasons are presented. The DA price is high for the winter season, and decreases in the spring. The lowest DA and BM price can be found in the summer. For the autumn, the price increases from the summer. The results were as expected. During winter the demand for electricity is high, hence the price increases in this period. During the spring reservoir levels are low, but the demand for electrify decrease resulting in lower prices than in the winter season. During summer, the reservoir is filling up due to the increase in inflow and the demand for electricity is low, resulting in a low price. For the autumn season, the demand starts to increase and the price increases.

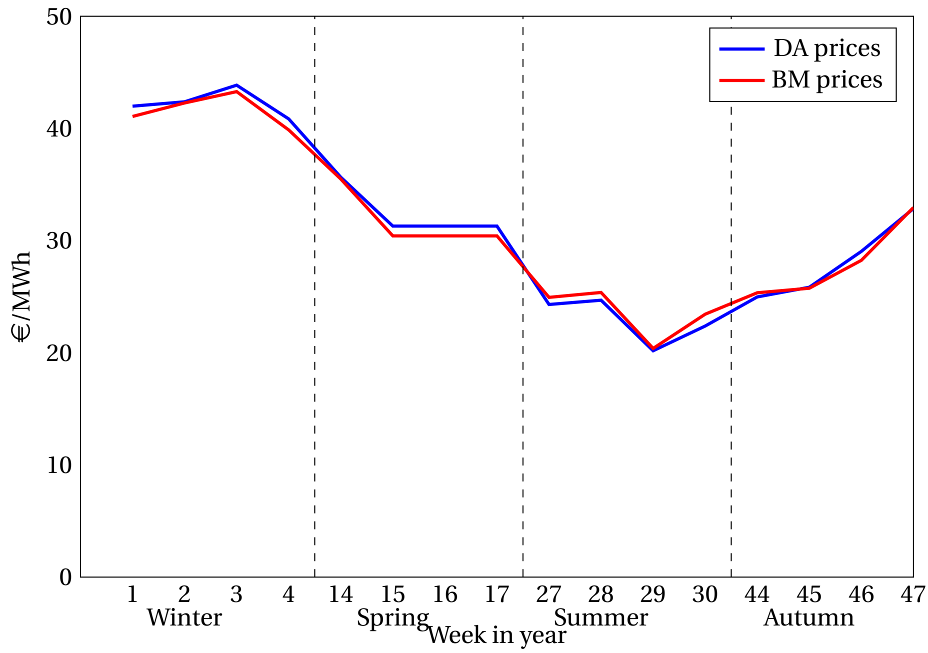


Figure 5.14: Seasonal changes in input prices

The aggregated reservoir level for the Tokke-Vinje hydro power system is presented in Figure 5.15. The results were as expected and to a high degree follows the scenario description describe in Section 4.2.2. Week 17 is black due to infeasiblity. The low initial reservoir levels used as model input is thought to be the reason for this infeasibility. The results was as expected. During the winter and spring the reservoir levels decrease because production is high and inflow is low. During the summer and autumn the inflow is high, and DA prices are increasing.

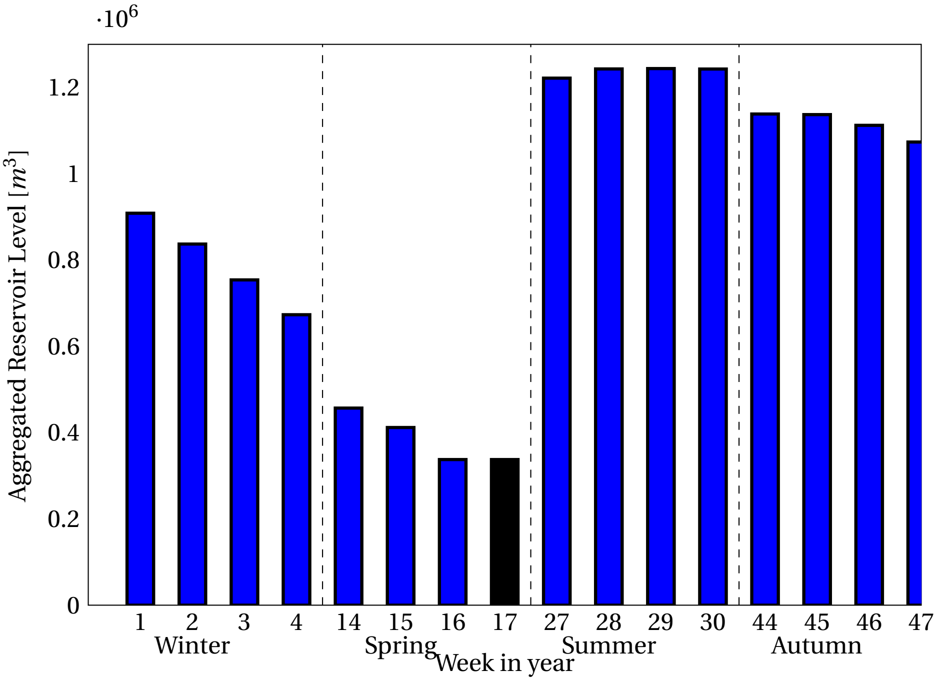


Figure 5.15: Seasonal changes in total aggregated reservoir level

Figure 5.16 describe the reservoir level for each week of operation in the winter season. The reservoir curve for the other seasons can be found in Appendix F. From the graph it can be observed the the reservoir level is steadily declining. The results were as expected since the end state of one week of operation is used as initial state for the next week of operation. It can be observed that by running the coupled AMPLWeek model more information about the reservoir levels in the magazine for a longer time horizon is provided. This gives more information about how decisions taken in the model now will affect the future reservoir levels.

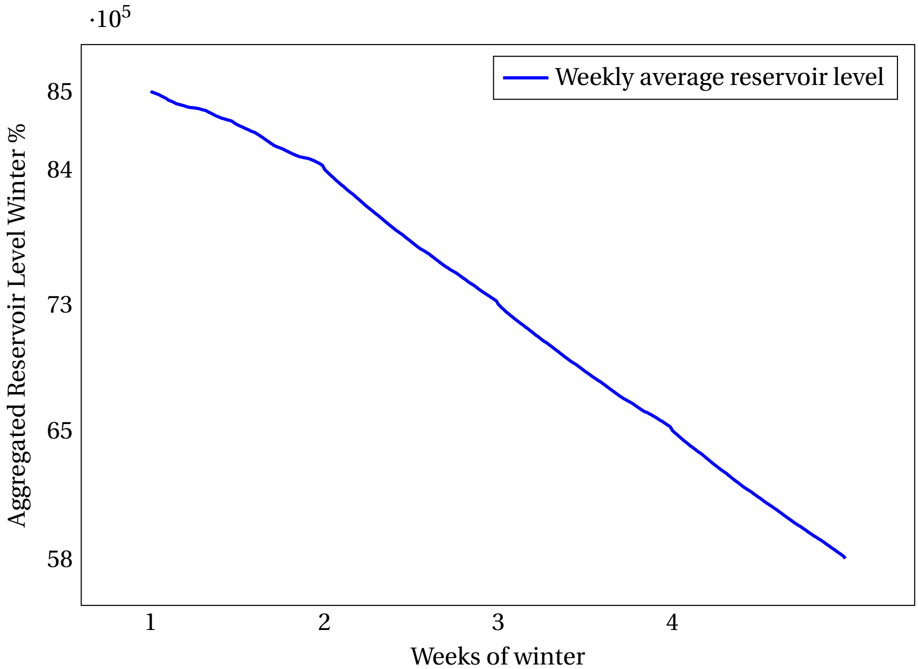


Figure 5.16: Weekly change in reservoir level for the winter season

5.4.2 Seasonal Effect on Objective Value

Figure 5.17 display the seasonal changes in the different components of the objective function. Week 17 is marked with black because the AMPLWeek model was infeasible for this week and the objective value is approximated by taking the same objective value as the previous week. A correlation between the DA price observed in Figure 5.14 and the objective value, DA and BM income can be seen. The objective value, as the DA price is higher for the winter season than the other seasons, with the lowest value in the summer and increasing value for the winter. The RKOM income is displayed, but this value is too small to notice in this graph. The change in the EFI might indicate that low DA prices results in an increase in the income from EFI because more water is stored in the reservoirs. Taking Figure 5.14 into consideration, the results were as expected.

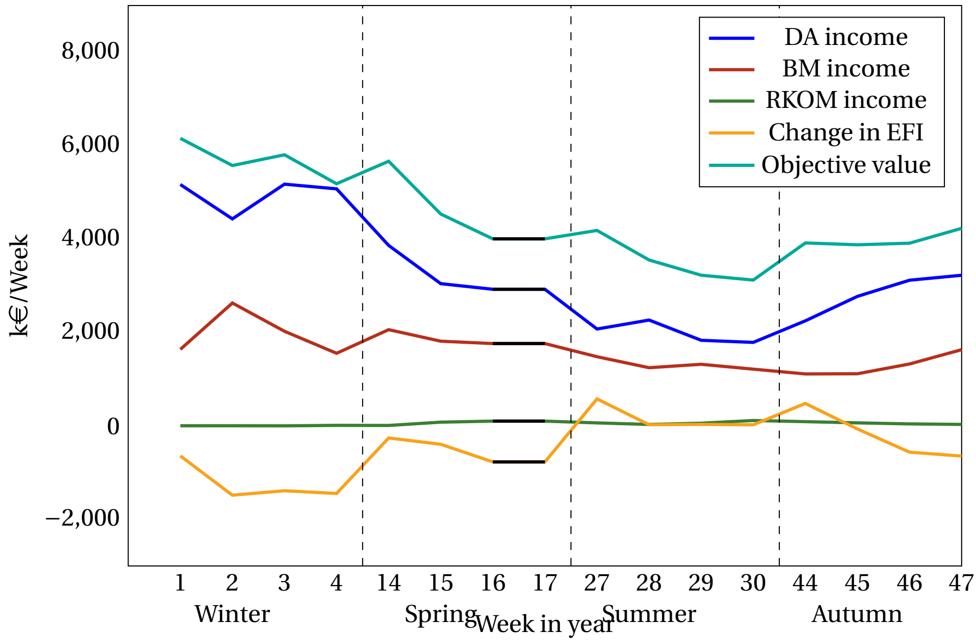


Figure 5.17: Seasonal changes in objective value

5.4.3 Reserved capacity in RKOM for different seasons

Figure 5.18 presents the seasonal changes in the reserved capacity in RKOM. During the winter season the high DA prices and the relatively high reservoir levels makes it very profitable for a hydro power producer to bid in DA. Since the capacity in RKOM is reserved from the DA, the RKOM bids are very low for the winter period. For the spring, the prices are lower than for the winter period, and the reservoir levels are very low. From Figure 5.8 it could be observed that the total dispatch volume decreased with an increased RKOM volume, since the reduction in DA is higher than the increase in BM, hence a behavior where the producer uses RKOM to save water can be observed for the spring season. For the summer season, the reserved capacity in RKOM is lower than the spring for each week except week 30. This is also the week when the DA price is at its lowest; hence, RKOM is at its most profitable. The model also wants to minimize spillage in the model, resulting in a lower reserved capacity during summer. For the autumn, the RKOM is very high for the first week and decreases for all consecutive weeks. This might be due to the increasing RKOM price in the same period.

The results display that the RKOM volume is closely linked to the price in DA since the reserved capacity cannot be bid into this market. The reservoir level in the hydro power system also affects the RKOM bid since the producer is paid to reserve a capacity for the BM market, hence the producer can dispatch a smaller volume into BM than in DA because the RKOM price makes it more profitable. This behavior is can be observed during the spring season for the hydro producer in Figure 5.8.

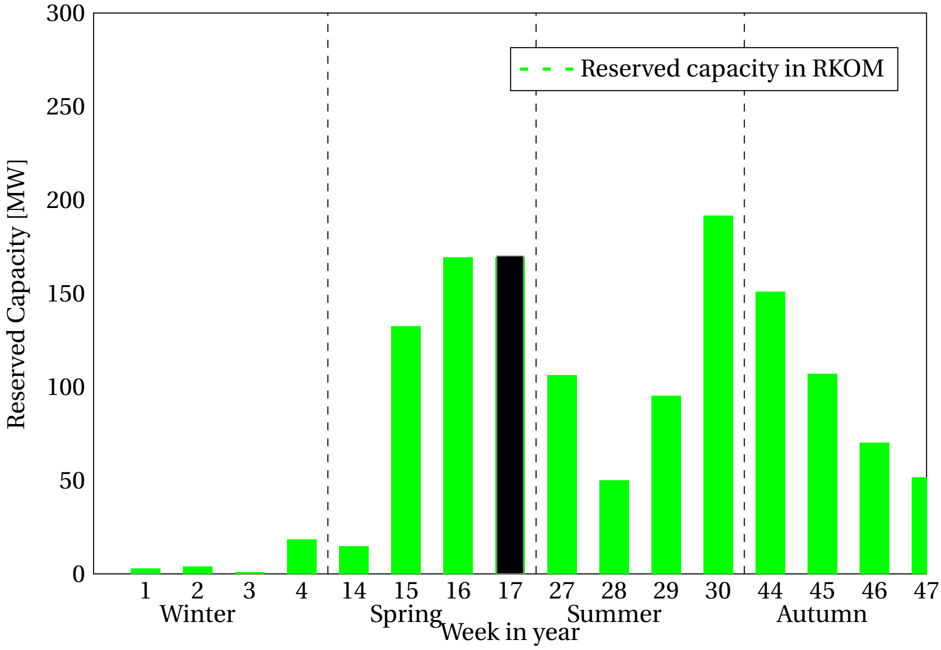


Figure 5.18: Seasonal changes in reserved capacity in RKOM

5.4.4 The seasonal change in the dispatch in DA and BM

Figure 5.19 display the total dispatch for the hydro power producer divided between average power dispatch in DA and average power dispatch for up and down regulation in BM. Down regulation in BM is presented as a negative value. The black bar is the unknown production of the infeasible model in week 17. A higher total production and average dispatch in DA can be seen for the winter and autumn seasons. This is when the DA price is high. Up regulation is decreasing for the winter season, stable for the spring season, decreasing for the summer season, and increasing for the autumn season. The down regulating dispatch is low for the winter season, and increases for the last part of the summer season and the autumn. When the average down regulation dispatch is high, total dispatch is low. This is because down regulation reduces the total production. The result were as expected. The total dispatch is highest when the DA price is high and lowest when the DA price is low. For the BM up and down regulation no clear trend could be observed, but the need for down regulation is higher for the two last week for summer and the two first week of autumn. In the same period the up regulation dispatch is low. This might imply that there is a greater need for unexpected higher need for electricity during winter than during summer. The demand for up and down regulation is a stochastic process. Hence, a clear trend in either down and up regulation in BM should not be observed.

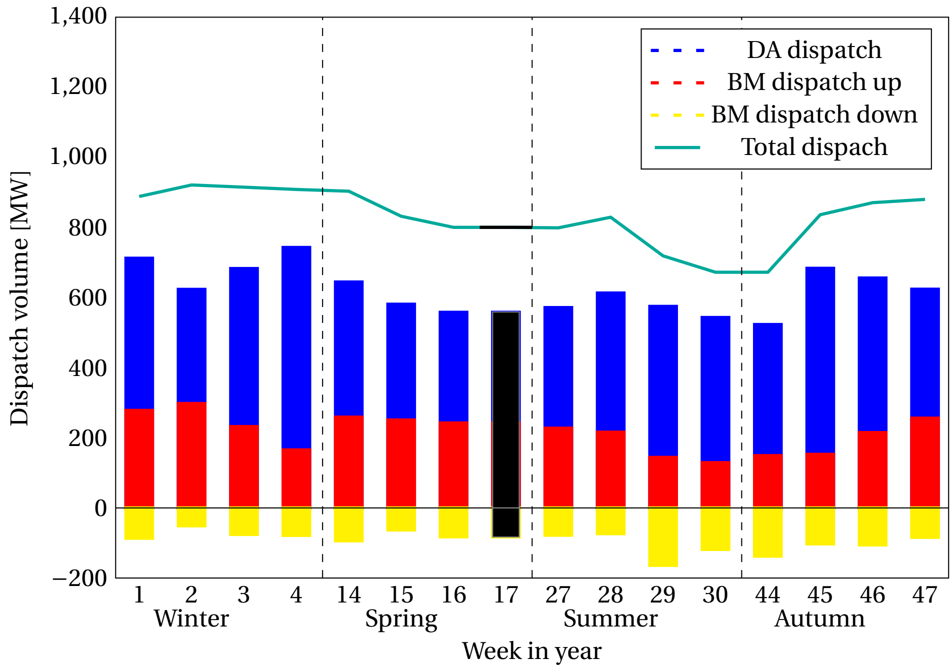


Figure 5.19: Seasonal changes in dispatch

5.5 Historical Comparison

In this section, historical RKOM prices for the winter season is displayed. The RKOM bid curve from AMPLWeek is compared to the historical RKOM bid curve. Historical seasonal difference in RKOM volumes is also presented and compared to the seasonal changes in the reserve capacity presented in AMPLWeek.

5.5.1 Historical reserved capacity for different RKOM prices compared to capacity reserved in AMPLWeek

Historical RKOM prices has been observed for the winter of 2014,2015 and 2016. The observations was based on the values from Appendix C. Figure 5.20 display the comparison between the historical bid curve for reserved capacity in RKOM and the bid curve for reserved capacity in RKOM from AMPLWeek.

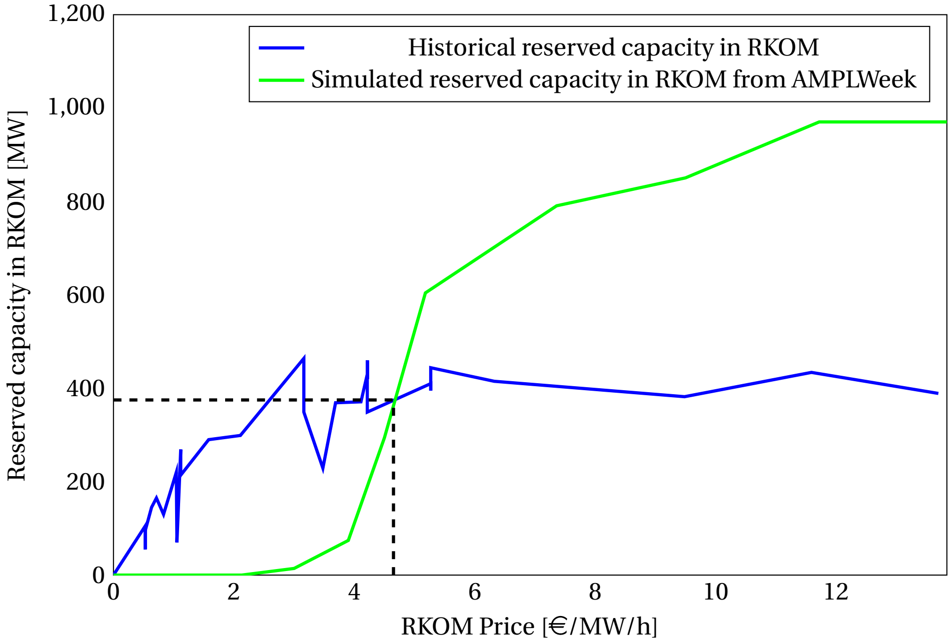


Figure 5.20: Price curve for historical RKOM bids

The dashed lines marks the cross section between the bid curve of RKOM in AMPLWeek and the historical bid curve for reserved capacity in RKOM. For prices higher than the equilibrium, the AMPLWeek is willing to bid a much greater volume into the model than the historical volume that is bid into the market. For price below the equilibrium, the AMPLWeek is willing to bid a much smaller volume into the model than historical volumes for the same price. It should be noted that the blue line, the historical volume bid into the market is the aggregated volume of all participants in the market, but a single market participants as is the case for AMPLWeek.

The result might indicate that AMPLWeek reserves more capacity in RKOM than would ever be accepted in the real market. In Section 5.2.2, it was observed that AMPLWeek reduces its dispatch for up regulation in BM and increases its reservoir levels instead of participating in BM and DA, if the RKOM price gets high enough. The results from Figure 5.20 indicates that based on historical RKOM prices and the aggregated reserved capacity in RKOM some of the results observed for high RKOM prices in this thesis would never occur in a reel market.

5.5.2 Historical Season Differences in RKOM Volume Compared to Seasonal Differences in AMPLWeek

In Figure 5.21 the reserved capacity in RKOM for AMPLWeek and historical data is presented together with the historical RKOM price. The historical reserved capacity in

RKOM is from 2015 and 2016 and display each week with RKOM commitments in this period. The simulated reserved capacity displays the RKOM commitment for the four simulated seasons presented in Figure 5.18. As can be observed, historical reserved capacity is high for the winter and autumn season and lower for the rest of the year. It should be noted that the historical data represents bids from every hydro power producer in the market while the simulated results only display the bids for one producer, hence the absolute values of the volume is not comparable.

It can be observed that the seasonal differences in the simulated reserve capacity not correlates with the historical reserved capacity in RKOM. The reason for this is not known, and recommendations for further work is described in Chapter 7. The reasons might be model weaknesses or assumptions that makes RKOM, especially during winter, less profitable in the model than in the real market.

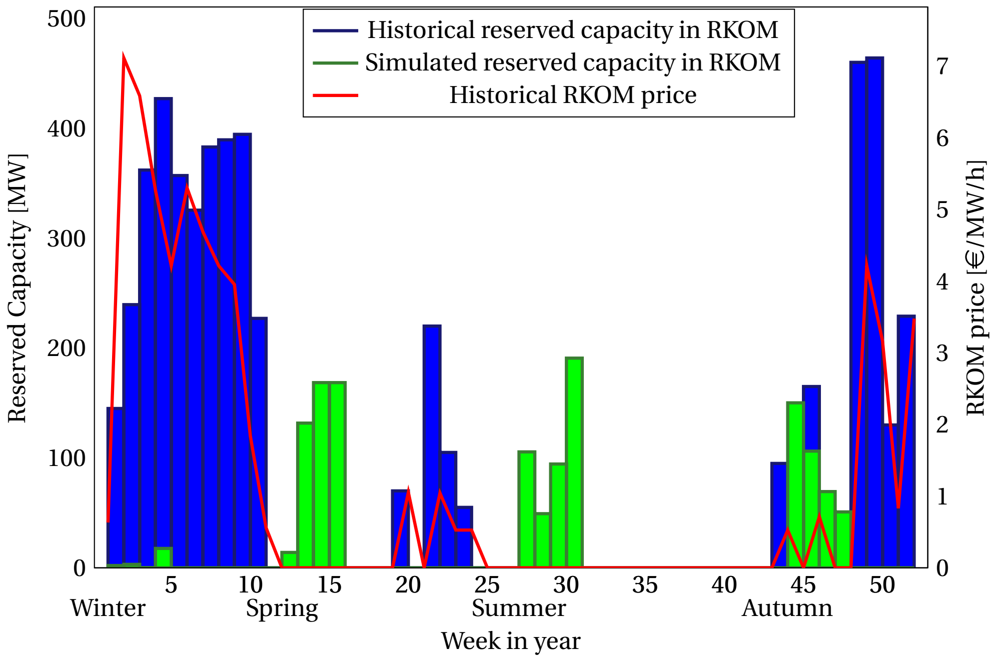


Figure 5.21: Historical RKOM volume for 2015 and 2016

5.6 Validity of Results

In this section the validity of the results are presented. It is discussed how the scenario size affects the real world approximation and how the number of scenarios affects the computational time. The use of historical DA and RKOM prices and how the uncertainty of the input data would affect the results if AMPLWeek were to be used as an actual scheduling tool is discussed in this section.

5.6.1 Reduced Computational Time and Robustness and Size of Scenario Trees

The results observed in Figure 5.4 indicates that the stability of the objective function increases with the number of scenarios. The number of scenarios used in this thesis was chosen based on computational reason and a higher number of scenarios could give better robustness of the results in the model. Additionally, as was observed in Figure 5.1, since AMPLWeek consist of 15 stages and only 20 scenarios the problem turns more and more deterministic for each day of operation. To incorporate more uncertainty in the scheduling problem, more scenarios should be included in the model. The number of tested variables in the problem can be found in Table 5.6. With 35 scenarios, a solution was never found. Either the computer ran out of memory or the solution time was too long. A stochastic hydro power scheduling problem has what is known as a flat objective. It has a large amount of possible solution that have a marginal effect on the objective function. A cause of this can be generator status. This is usually modeled by binary variables and each generator in the system can be either 1 or 0. When operating with a long time horizon and multiple scenarios, the number of possible generator combination increases. The solver used in this problem has to check each of these combinations to find the combination that has the highest objective value. There is many combinations of generators for a week of operation and the change in the objective value is often marginal. To reduce computational time, 20 scenarios were used. By reducing the scenario, the number of variables, as can be seen from Table 5.6, was reduced to a more manageable size. In addition, the MIP gap in the CPLEX solver used in this thesis was increased from 0.0001 to 0.0005. This was done to better deal with the flat objective in a hydro power scheduling problem. By doing this, the solver terminates when it finds an integer solution that is within 0.05 % of the relaxed solution, instead if 0.01 %. This greatly reduced the computational time of the model. If the model ever were to be used for commercial purpose, it would be recommended to reduce the MIP-gap so that a better objective value would be found. Alternative approaches to increase the number of scenarios in the model is described in Section 7.2.

Scenarios	Variables	Solution time [h]
20	1228734	35
35	2141234	4

Table 5.6: Computational differences between 20 and 35 scenarios

5.6.2 Scenario Reduction of Multi Stage Stochastic Programs

The scenario tree reduction program presented in [28] is designed for a two stage problem, for which the problem only has one branching point (a scenario fan). This is due to the fact that the information on the structure of the scenario tree is not taken into account for multistage problems[30].How this problem affects the scenario reduction algorithm is unknown.

5.6.3 Time Horizon of AMPLWeek in a Commercial Perspective and validity of RKOM prices

The objective of the AMPLWeek model is to optimize the bids in RKOM. The model has to be ran before 12.00 Friday the week before deliverance to get the optimal volume to bid into RKOM before the RKOM is cleared. As a potential commercial software, the solution time and uncertainty of price in DA and BM would affect the results. Assuming a solution time of 24 hours the model has to be run Thursday morning to optimize the bid in RKOM. The input data is forecasting prices over 10 days in advance. Since the model runs at a much earlier stage than the deadline for bids in DA and BM and because the model has a long time horizon, there is a high degree of uncertainty concerning the price forecast for the DA and BM.

The historical DA and RKOM prices used in this thesis is not necessarily from the same year. It might affect the result in thesis.

CHAPTER 6

Conclusion

Well-functioning balancing markets are crucial for the security of the future electricity supply for Norway and Europe. In Norway, there is a reserve capacity market, designed to ensure that a sufficient amount of capacity is bid into the balancing market. The market is called RKOM.

The main purpose of this thesis has been to evaluate how participating in RKOM affects the decision-making of a hydro power producer compare to only participating in the day-ahead and balancing market. The general theoretical literature on this subject is inconclusive and decision-making and hydro power scheduling considering these three markets has yet to be considered. The weekly time resolution of the reserve capacity market makes it difficult to analyse in already excising hydro power scheduling models. Consequently, an excising short term model has been expanded and altered to incorporate the reserve capacity market. AMPLWeek is meant as decision support for a hydro power producer considering bids in RKOM. The study sought to answer how decision making changes by participating in RKOM. The model was evaluated for different RKOM prices and for different seasons, and the gain of participating in market was measured. This thesis also had intentions to evaluate the benefits of using a weekly scheduling plan compared to a daily time horizon and how doing seasonal simulations affects the decision making for the hydro power producer. It is also considered to which extent scenario reduction affects the approximation done considering DA and BM prices in this thesis. This thesis demonstrates that a hydro power producer's willingness to reserve capacity in RKOM increases with an increased RKOM price. The average power dispatch in the DA and the down regulation dispatch in BM decreases with a higher RKOM price. Smaller changes could be observed in the average up regulation dispatch, in form of a bell shaped curve, with increasing RKOM price. The up regulation dispatch is higher for a low RKOM price than for a high RKOM price. This might indicate that RKOM not necessarily works as an incentive for hydro power producers to increase their bids in BM. Whether this is due to model simplifications or actually apply to the real market is unknown, and is recommend as further work. Observations done on seasonal effects on participating in RKOM demonstrates that the DA price and reservoir level display strong correlation to the reserved capacity in RKOM. The results show that participating in RKOM is most profitable during Spring and summer, when DA prices and reservoir levels are low. It was also observed that season simulations of a weekly scheduling model provides useful information about the change in the reservoir levels in the hydro power system, compared to only doing weekly scheduling. Comparative analysis of AMPLWeek compared to a model with a daily time horizon demonstrates that reserved capacity varies for each day in the time horizon, optimizing they daily reserved capacity depending on the price in the market for the given day. This results in a lower objective value if the reserved capacity is fixed for the whole week, compared to one day; hence, a weekly scheduling model provides a more pessimistic, but more realistic, estimate of the profitability of participating in RKOM. The results also show that the scenario reduction in this thesis provides an acceptable approximation of a real life scenarios.

In this thesis, a method for sequentially generating scenario trees for a week of hydro power operation has been developed. Additionally, a simulation procedure for a seasonal short term hydro power scheduling model, that incorporate the changes in the reservoir level and water values over several weeks of operation, has been developed. This is, to the author's knowledge, the first time a reserved capacity market has been incorporated in short term hydro power scheduling model. The seasonal effects, as reservoir level, has also been observed for a short term hydro power scheduling model. Balancing markets are crucial to secure the electric supply. Precise models of the reserve capacity market is an important step for securing the supply of electricity. The work presented in this thesis can be used as help for decision making for hydro power producers in the future.

CHAPTER 7

Recommendations for Further Work

In this chapter the recommendations for further works is presented. Several different subjects can be studied more extensively with the existing AMPLWeek model and seasonal simulation procedure done in this thesis. Some possible approaches are presented in Section 7.1. Possible model expansions and improvements are presented in Section 7.2.

7.1 Extensive Analysis of AMPLWeek

In this section, recommendations for deeper analysis of the work done in this thesis are presented. In Figure 5.21, the historical reserved capacity in RKOM was compared to the simulated reserve capacity done in the model for different season. Even though these values not are directly comparable because they are bids from a price taker and the whole market, large differences could be observed between the historical and the simulated results. Because the model only was run for selected months of the year, comparing the simulated results to yearly historical data is difficult because model behavior for all moths except the simulated once are unknown. All the simulated results was also done with the same RKOM price, and as can be observed from Figure 5.21, this is clearly not the case for the historical data. A possible further analysis could be to do extensive research on the difference in seasonal bids in RKOM. Yearly simulations could be done for each week in the year and simulations could be done for different prices to observe the price sensitivity of the model for different seasons. Another interesting observation would be to find out which market participants that are reserving capacity for the historical data in Figure 5.21. The simulated reserve capacity in RKOM shows that reserving capacity in the winter season is less profitable than for other seasons given the same price. This is because the DA price is high for this period. The historical data from the same period, on the other hand, show that this is when the reserved capacity accepted in RKOM is highest. Since both producers and consumers can participate in RKOM, it would be interesting to find out if there are any differences in when consumer or producers are reserving capacity in RKOM. Further work could be to study how market mechanisms in RKOM could be change to make it more appealing to hydro power producers. The reason of the bell-shaped curve observed in Figure 5.9 could be further explained. As described in Section 5.2.2, the author of this thesis has a hypothesis that it is caused by the state of and startup cost of the generators in the hydro power system. It could be possible to test this hypothesis by running AMPLWeek for different RKOM price to observe the generator settings of the system and see if this affects the up regulation dispatch. Simulation in AMPLWeek with and without the startup cost could give further explanation of the model behavior.

7.2 Possible Model Expansions

In this section, possible model expansions are considered and recommends as further work. As described in Section 2.2.2, there exists multiple reserve market. One of these are the primary reserves. In this market, hydro power producers are paid to reserves a certain amount of capacity for primary reserves. Either this could be included in the model as an additional market and it could be studied if reserving capacity in the primary reserves or RKOM would be most profitable. At the moment there are no uncertainty considering the RKOM price given in AMPLWeek, the decision variable that decide the reserved capacity do this based on a deterministic RKOM price. By including the RKOM price as a random variable instead of a deterministic variable, similar to what has already been done to the DA and BM, it would be possible to optimize the hydro power scheduling based on different RKOM price scenarios. In AMPLWeek, it is assumed, as described in Section 3.1, that all BM bids happens at the same time just as with the DA bids. By including more stages in the model, it would be possible to approximate the sequential nature of the BM market, where bids are done 45 min before each hour of operation.

As described in Section 5.6 increasing the number of scenarios in the model could be profitable, since more scenarios better approximate statistical properties of DMM and more uncertainty could be included in the model. Using a computer with a higher capacity than the one used in this thesis could enable more scenarios in AMPLWeek. An alternative approach would be to evaluate how the time horizon of the model affects the optimal reserved capacity in RKOM. If reducing the time horizon of the model do not affect the decision variable that reserves capacity in RKOM, this will greatly reduce the number of variables in the model and enable more scenarios. It would also be possible to include more model assumptions to reduce the number of equations in the model. An example would be to have a daily time resolution compared to an hourly resolution. This would reduce the number of variables in the problem considerably.

References

[1] Statnett. System operations and market development plan 2014-20. Technical report, Statnett, April 2015.

[2] A. Helseth, M. Fodstad, and A.L. Henden. Balancing Markets and their Impact on Hydropower Scheduling. Technical Report TR XX, SINTEF Energy AS, August 2015.

[3] G. Doorman. *Hydro Power Scheduling*. NTNU, 1nd edition, 2015.

[4] Statkraft. About statkraft. http://www.statkraft.com/about-statkraft/, (2015-10-05).

[5] OJ. Directive 2012/27/eu of the european parliament and of the council of 25 october 2012 on energy efficiency, amending directives 2009/125/ec and 2010/30/eu and repealing directives 2004/8/ec and 2006/32/ec text with eea relevance. 20012.

[6] NordPool Spot. About us. http://www.nordpoolspot.com/About-us/, (2015-12-02).

[7] Eirik Schou Grytli. Optimal spot market bidding considering reserve power and reserve power options. 2015.

[8] T. Boomsma, N. Juul, and SE. Fleten. Bidding in sequential electricity markets: The nordic case. *European Journal of Operational Research*, 238, 2014.

[9] C. Rassmusen and J. Boye. Profitability of a hydro power producer biding in multiple markets. 2014.

[10] G. Klæbo. On the gains of stochastic bidding optimization for the day-ahead market for a hydro power producer. *IEEE Trondheim PowerTech*, 2011.

[11] M. Belsnes, O. Wolfgang, T. Follestad, and E Assgård. Applying successive linear programming for stochastic short-term hydropower optimization. *Electric Power System Research*, 1, 2015.

[12] M. Kaut and S. Wallace. Evaluation of scenario-generation methods for stochastic programming. 2003.

[13] W. Romisch, N Growe-Kuska, and W Romisch. Scenario reduction and scenario tree construction for power management problems. *IEEE Bologna Power Tech Conferance*, 2003.

[14] SE. Fleten and T. Kristoffersen. Stochastic programming for optimizating bidding strategies of a nordic hydropopower producer. *European Journal of Operational Reaserach*, 181, 2006.

[15] D. De ladurantaye, M. Gendreau, and J. Potvin. Optimizing profits from hydroelectricity production. *Computer and Operations Research*, 36, 2007.

[16] SE. Fleten and E. Pettersen. Constructing bidding curves for a price-taking retailer in the norwegian electricity market. *IEEE Transactions on Power System*, 20, 2006.

[17] SE. Fleten and T. Kristoffersen. Short-term hydropower production planning by stochastic programming. *Computer and Operations Research*, 35, 2006.

[18] E. Faria and SE. Fleten. Day-ahead market bidding for a nordic hydro-power producer: taking the elbas market into account. *Computer Management Science*, 8, 2009.

[19] I. Wangensteen. *Power System Economics the Nordic Electricity Market*. Tapir Academic Press, 2nd edition, 2011.

[20] A. Helsteth, M. Fodstad, and A. Henden. Balancing markets and their impact on hydropower scheduling, restricted. *SINTEF*, 2004.

[21] A. Helseth, M. Fodstad, and A. Henden. Balancing markets and their impact on hydropower scheduling. *SINTEF Energy reasearch*, 2015, Confidential.

[22] NordPool. Block order on nordpool spot. http://www.nordpoolspot.com/TAS/Day-ahead-market-Elspot/Order-types/Block-bid/, (2015-12-02).

[23] Statnett. About Statnett - Statnett. http://www.statnett.no/en/About-Statnett/, (2015-10-05).

[24] Statnett. Vilkår for tilbud, aksept/prissetting og håndtering av bud i regulerkraftopsjonsmarkedet (rkom). http://statnett.no/Drift-og-marked/Markedsinformasjon/RKOM1/Om-RKOM/, (2015-11-12).

[25] J.R. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer Science+ Business Media, 1nd edition, 2011.

[26] A. Eriksrud and J. Braathen. Spot market bidding taking the balancing rower market into account. 2012.

[27] J. Dupacova, N Growe-Kuska, and W Romisch. Scenario reduction in stochastic programming. *Mathematical Programming Series A*, 95, 2003.

[28] Turid Follestad. Documentation of the c program scentreegen for scenario tree generation and reduction. 2012.

[29] Wasserstein metric, July 2015. Page Version ID: 672202513.

[30] H. Heitsch and W. Romisch. Scenario reduction from multistage programming. *Computer Management Science*, 6, 2008.

[31] S. Ross. *"Introduction to Probability Models"*. Elsivier Academic Press, 11nd edition, 2014.

[32] AMPL Optimization Inc. AMPL . http://ampl.com/products/ampl/, 2015-12-06.

[33] SINTEF. Prodrisk. http://www.sintef.no/sintef-energi/programvare/alle-programmer/prodrisk/, (2015-12-08).

[34] H. Heitsch and W. Romisch. Scenario reduction algorithms in stochastic programming. *Computational Optimization and applications*, 24, 2003.

[35] Statnett. Om regulerkraftopsjoner (RKOM). http://statnett.no/Drift-og-marked/Markedsinformasjon/RKOM1/Om-RKOM/, (2015-11-20).

[36] NVE. Historiske vannføringsdata til produksjonsplanlegging. https://www.nve.no/hydrologi/hydrologiske-data/historiske-data/historiske-vannfoeringsdata-til-produksjonsplanlegging/, (2015-12-10).

Appendices

APPENDIX A

A Mathematical description of the model

A mathematical formulation of the model can be found in this section. Definitions of Sets, Parameters and Variables **Sets:**

S	Scenarios
M	Stages, markets in the model
N	Scenario three nodes
N_m	Scenario three nodes within stage *m* in *M*
S_n	Set of scenarios that all goes through *n*
T	Number of operational periods
T^M	Number of market periods
$T^{Tot} = T + T^M$	Number of all periods
R	Number of reservoirs
R^{Tar}	Target reservoir level
P	Break points on penalty curve
D	Days
$R_r^D, R_r^S, R_r^B, R_r^P$	Set of reservoirs with discharge, spillage, bypass and pumping going to the reservoir indexed by *r*
G_r	Set of all hydropower generators for reservoir *r*
$G = \cup_{r \in R} G_r$	Set of cuts in the water value approximation
J	Set of cuts in the water value approximation
I	Line segments in PQ-curve
B_{DA}, B_{BM}	Set of breakpoints in the day-ahead market and the balancing market
$T^D_{\hat{r},t}, T^S_{\hat{r},t}, T^B_{\hat{r},t}, T^D_{\hat{r},t}$	Time spent flowing from one reservoir to another

Indexes:

s	Scenarios in S
$t / \hat{t}$	Time periods and time to arrival in downstream reservoir from *r*
$r / \hat{r}$	Reservoirs in R to reservoir r and from reservoir $\hat{r}$

j	Water value cuts in J
g	Hydro Power generators in G
i	Line segments in the PQ-curve
p	Break points on penalty function
b	Break points in B_{DA} and B_{BM}
d	Days in a Week,1 to 6

Parameters :

Topology

$Prob_s$	Probability of each scenario
	Time length
$T_{\hat{r}}^{D,Lag}, T_{\hat{r}}^{S,Lag}, T_{\hat{r}}^{B,Lag}$	Time lag for discharge, spillage, bypass $[h]$
$Q_{r,s,t}^{R}, Q_{r,s,t}^{U}$	Regulated and unregulated inflow $[m^3/s]$
$\bar{Q}_r^V, \underline{Q}_r^V$	Upper and lower reservoir level $[km^3]$
Q_{0r}^V	Initial reservoir level$[km^3]$
$\bar{Q}_r^R, \underline{Q}_r^R$	Maximum and minimum release from reservoir $[m^3/s]$
$\bar{Q}_r^B, \underline{Q}_r^B$	Maximum and minimum bypass from reservoir $[m^3/s]$
Q_r^S	Spillage above this limit causes spillage cost to incure $[m^3/s]$
C_r^{Spill}	Cost of spillage damage $[ecent/m^3]$
$\bar{Q}_r^{\Delta V}, \underline{Q}_r^{\Delta V}$	Maximum increase and decrease in reservoir level within an hour
$\bar{Q}_r^{Flow}, \underline{Q}_r^{Flow}$	Maximum increase and decrease in flow level within an hour

Generators

C_g^{Start}	Cost of starting the production [Euro]
δ_0^{Spin}	Initial spinning mode of the generators [0,1]
$\underline{Q}_g^D$	Minimum discharge from each generator $[m^3/s]$
$\bar{Q}_{g,i}$	Maximum discharge volume for each line segment on PQ-curve, for each generator
$\bar{Q}_{g,i}, \underline{Q}_g^D$	Maximum and minimum hydro production in generator g $[MW/h]$
$\eta_{g,i}$	Production efficiency on line segment i on PQ-curve for generator g $[MW/m^3 s]$
$\bar{W}_g, \underline{W}_g$	Maximum and minimum hydro production in generator g $[MW/h]$
$\bar{W}^{Tot}$	Maximum total production

Water value function

$\mu_{r,j}^*$	Slope of water value cut[Euro$/km^3$]
$Q_{r,j}^{WV}$	Reservoir level, percentage of $\bar{Q}_r^V$

A_j	Total water value for water value cut $j [10kEuro]$
A^0	Initial water value level

Penalty function

P_r^{Max}	Last break point on penalty function
$Q_{r,p}^{Break}$	Break point reservoir level as function of total capacity
$C_{r,p}$	Slope of the penalty function
$C_{r,p}^{Fix}$	Fixed part of the penalty function
Q_r^{Tar}	Index of target reservoir

Fixed obligations

D_t^{Fix}	Fixed delivery in each time period

Day-ahead market

$\tilde{P}_{t,s}^{DA}$	Day-ahead market price [Eur/MWH]
$\bar{P}_t^{DA}, \underline{P}_t^{DA}$	Maximum and minimum bid price in day-ahead market [Euro/MWH]
$P_{b,t}^{DA+}, P_{b,t}^{DA-}$	Price points on the bid curve for supply and demand in day-ahead market [Euro]
$B^{DA,MAX}$	Last break point in the day ahead bid curve

Balancing market

$\tilde{P}_{t,s}^{BM}$	Balancing market price [Eur/MWH]
$\bar{P}_t^{BM}, \underline{P}_t^{BM}$	Maximum and minimum bid price in balancing market [Euro/MWH]
$P_{b,t}^{BM+}, P_{b,t}^{BM-}$	Price points for supply and demand in balancing market [Euro]
$B^{BM,MAX}$	Last break point in the balancing market bid curve

RKOM

X_t^{RKOM}	Reserved up regulation in RKOM [MWh/h]

Risk handling

Obj_s^{DA}	Result from each day-ahead scenario
λ	Acceptable negative deviation from DA result

Continuous variables

$q_{r,s,t}^{v}$	Reservoir level at end of time period t

$q^{v+}_{r,s}$ Aggregated flow that will arrive in reservoir after T_{Max}
$q^R_{r,s,t}, q^S_{r,s,t}, q^B_{r,s,t}$ Released, spillage and bypass from reservoir $[m^3/s]$
$q^D_{g,t,s}$ Discharge to hydro power generators $[m^3/s]$
$q^D_{g,i,t,s}$ Discharge volume on line segment of linearized PQ-curve $[m^3/s]$
$w_{g,t,s}$ Hydro power production in generator g [MWh]
c^{TotP}_s Cost of total penalty function [EURO]
$c^{TotR}_{s,r}$ Cost penalty function of each reservoir in R^{Tar} [EURO]
$c^{Spill}_{r,s,t}$ Cost of spillage [Euro]
$c^{Start}_{g,s,t}$ Start up cost for generator not spinning [EURO]
α_s Water value of final reservoir level [EURO]
x^{RKOM}_t Capacity bid in RKOM [Mw]
$x^{DA+}_{b,t,s}, x^{DA-}_{b,t,s}$ Supply and demand bid curve for break point b in DA market [Mwh]
$y^{DA+}_{t,s}, y^{DA-}_{t,s}$ Accepted supply and demand [Mwh]
$x^{BM+}_{b,t,s}, x^{BM-}_{b,t,s}$ Supply and demand bid curve for break point b for BM [Mwh]
$y^{BM+}_{t,s}, y^{BM-}_{t,s}$ Accepted supply and demand [Mwh]
$y^{IB+}_{t,s}, y^{IB-}_{t,s}$ Positive and negative imbalance volume [Mwh]
Obj_s Optimal objective function for each scenario

Binary variables

$\delta^{Spill}_{r,s,t}$ 1 if spill is possible
$\delta^{Spin}_{g,s,t}$ 1 if generator is running

Equation for calculating time-lag sets: Same equation for D,B and S exemplified here with D

$$T^D_{\hat{r},t} = \{\hat{t} \in T : t-1 < \hat{t} - \frac{1}{2} + \frac{T^{D,Lag}_{\hat{r}}}{T^L} \geq t\} \tag{A.1}$$

Objective function

$$
\begin{aligned}
\text{Objective} = \sum_{s\in S} prob_s(&- \sum_{t\in T}\sum_{r\in R} c^{Spill}_{r,s,t} && \text{(A.2)}\\
&- \sum_{t\in T}\sum_{g\in G} c^{Start}_{g,s,t} && \text{(A.3)}\\
&+\alpha_s && \text{(A.4)}\\
&+ \sum_{g\in G}(\delta^{Spin}_{g,s,t_{Max}} - \delta^{Spin}_{g,s,t_0}) && \text{(A.5)}\\
&+ \sum_{t\in T} \tilde{P}^{DA}_{t,s}(y^{DA+}_{t,s} - y^{DA-}_{t,s}) && \text{(A.6)}\\
&+ \sum_{t\in T} \tilde{P}^{BM}_{t,s}(y^{BM+}_{t,s}, y^{BM-}_{t,s}) && \text{(A.7)}\\
&+ \sum_{t\in T} \tilde{P}^{RKOM}_{t} x^{RKOM}_{t}, && \text{(A.8)}\\
s\in S, t\in T & && \text{(A.9)}
\end{aligned}
$$

Reservoir mass balance

$$q^{v}_{r,s,t} = q^{v}_{r,s,t-1} - T^{L}(q^{R}_{r,s,t} - q^{S}_{r,s,t}) + \sum_{\hat{r}\in R^D_r}(\sum_{g\in G^R_r}\sum_{t\in T^D_{\hat{r},t}} q^{D}_{g,\hat{t},s} + \sum_{\hat{r}\in R^S_r}\sum_{t\in T^S_{\hat{r},t}} q^{S}_{\hat{r},\hat{t},s} + \sum_{\hat{r}\in R^B_r}\sum_{t\in T^B_{\hat{r},t}} q^{D}_{\hat{r},\hat{t},s}) \quad , r\in R, t\in T, s\in S \tag{A.10}$$

Water value cuts

$$\alpha_s \leq \sum_{r\in R} \mu^{*}_{r,j}(q^{v}_{r,s,T^{max}} + q^{v} +_{r,s} - Q^{WV}_{r,j}\bar{Q}^{V}_{r}) + (A_j - A^0) \quad , j\in J, s\in S \tag{A.11}$$

End flow calculation

$$
\begin{aligned}
q^{v+}_{r,s,t} = \sum_{r\in R^D_r}\sum_{t\in T-T^{D,Lag}_{\hat{r}}} T^L_t q^D_{\hat{r},t,s} + \sum_{r\in R^S_r}\sum_{t\in T-T^{S,Lag}_{\hat{r}}} T^L_t q^S_{\hat{r},t,s} && \text{(A.12)}\\
+ \sum_{r\in R^B_r}\sum_{t\in T-T^{B,Lag}_{\hat{r}}} T^L_t q^B_{\hat{r},t,s} && \text{(A.13)}\\
, r\in R, t\in T, s\in S && \text{(A.14)}
\end{aligned}
$$

Penalty function

$$c^{TotR}_{s,r} \geq -C_{r,p} q^{v}_{r,s,T^{max}} + C^{Fix}_{r,p}, \quad s\in S, r\in R^{Tar}, p\in Q^{Tar}_r \tag{A.15}$$

Penalty total

$$c^{TotP}_{s} = \sum_{r\in R^T ar} c^{TotR}_{s,r}, \quad s\in S \tag{A.16}$$

Spillage cost

$$c^{Spill}_{r,s,t} \geq C^{Spill}_{r}(q^{S}_{r,s,t} - Q^{S}_{r}), \quad r\in R, t\in T, s\in S \tag{A.17}$$

Restriction to only allow spill when reservoir is at max level

$$q^{S}_{r,s,t} \leq \bar{Q}^{V}_{r}(1 - \delta^{Spill}_{r,s,t}), \quad r\in R, t\in T, s\in S \tag{A.18}$$

Spillage not allowed unless release is at max

$$\delta^{Spill}_{r,s,t} \geq \frac{(\bar{Q}^{R}_{r} - q^{R}_{r,s,t})}{\bar{Q}^{R}_{r}}, \quad r\in R, t\in T, s\in S \tag{A.19}$$

Spillage not allowed unless reservoir is full

$$\delta^{Spill}_{r,s,t} \geq \frac{(\bar{Q}^V_r - q^v_{r,s,t})}{\bar{Q}^V_r},, r \in R, t \in T, s \in S \tag{A.20}$$

Ramping constraint on reservoir

$$-T^L \bar{Q}^{\Delta V} \bar{Q}^V_r \leq q^v_{r,s,t} - q^v_{r,s,t-1} \leq T^L \bar{Q}^{\Delta V} \bar{Q}^V_r \tag{A.21}$$

Discharge limits

$$-T^L \underline{Q}^{Flow}_r \leq \sum_{g \in G_r} (q^D_{g,t,s} - q^D_{g,t-1,s}) \leq \bar{Q}^{Flow}_r T^L, \quad r \in R, t \in T, s \in S \tag{A.22}$$

Power generation on each line segment of PQ-curve

$$w_{g,t,s} = \sum_{i \in I_g} \eta_{g,i} q^D_{g,i,t,s}, \quad g \in G, t \in T, s \in S \tag{A.23}$$

Water flow balance between reservoir and generator

$$\sum_{g \in G} q^D_{g,t,s} = q^R_{r,s,t} - q^B_{r,s,t} + Q^U_{r,s,t}, \quad r \in R, t \in T, s \in S \tag{A.24}$$

Aggregated discharge across all line segments

$$q^D_{g,t,s} = \sum_{i \in I_g} q^D_{g,i,t,s}, \quad g \in G, t \in T, s \in S \tag{A.25}$$

Minimum discharge and production if spinning

$$q^D_{g,t,s} \leq \underline{Q}^D_g \delta^{Spin}_{g,s,t} \tag{A.26}$$

$$w_{g,t,s} \leq \underline{Q}^D_g \delta^{Spin}_{g,s,t} \tag{A.27}$$

No production if the no spinning

$$q^D_{g,t,s} \geq \sum_{i \in I_g} \bar{Q}_{g,i} \delta^{Spin}_{g,s,t}, \quad g \in G, t \in T, s \in S \tag{A.28}$$

Start up cost for generator

$$c^{Start}_{g,s,t} \geq C^{Start}_g (\delta^{Spin}_{g,s,t} - \delta^{Spin}_{g,s,t-1}), \quad g \in G, t \in T, s \in S \tag{A.29}$$

Energy balance in market

$$\sum_{g \in G} w_{g,t,s} = D^{Fix}_t + (y^{DA+}_{t,s} - y^{DA-}_{t,s}) + (y^{BM+}_{t,s} - y^{BM-}_{t,s}) + (y^{IB+}_{t,s} - y^{IB-}_{t,s}), \quad t \in T, s \in S \tag{A.30}$$

Day-ahead market

Total activated supply in the day-ahead market

$$y^{DA+}_{t,s} = \frac{\tilde{P}^{DA}_{t,s}) - P^{DA+}_{b,t}}{P^{DA+}_{b+1,t} - P^{DA+}_{b,t}} x^{DA+}_{b+1,t,s} + \frac{P^{DA+}_{b,t} - \tilde{P}^{DA}_{t,s}}{P^{DA+}_{b+1,t} - P^{DA+}_{b,t}} x^{DA+}_{b,t,s}, \quad \text{if}: P^{DA+}_{b,t} \geq \tilde{P}^{DA}_{t,s} \geq P^{DA+}_{b+1,t}, \quad b = 2,..,B^{DA,MAX} - 1, t \in T, s \in S \tag{A.31}$$

Total activated demand in the balancing market

$$y^{DA-}_{t,s} = \frac{\tilde{P}^{DA}_{t,s}) - P^{DA-}_{b,t}}{P^{DA-}_{b+1,t} - P^{DA-}_{b,t}} x^{DA-}_{b+1,t,s} + \frac{P^{DA-}_{b,t} - \tilde{P}^{D}A_{t,s}}{P^{DA-}_{b+1,t} - P^{DA-}_{b,t}} x^{DA-}_{b,t,s}, \quad \text{if}: P^{DA-}_{b,t} \geq \tilde{P}^{DA}_{t,s} \geq P^{DA-}_{b+1,t}, \quad b = 2,..,B^{DA,MAX} - 1, t \in T, s \in S \tag{A.32}$$

Monotonly increasing bid curve for day ahead market

$$x^{DA+}_{b,t,s} \leq x^{DA+}_{b+1,t,s}, \quad b = 1,..,B^{DA,MAX} - 1, t \in T, s \in S \tag{A.33}$$

$$x_{b,t,s}^{DA-} \leq x_{b+1,t,s}^{DA-}, \quad b = 1,..,B^{DA,MAX} - 1, t \in T, s \in S \tag{A.34}$$

Balancing market

Total activated supply in balancing market

$$y_{t,s}^{BM+} = \begin{cases} x_{b,t,s}^{BM+}, & \text{if: } P_{b,t,s}^{BM+} \leq \tilde{P}_{t,s}^{BM} \leq P_{b+1,t,s}^{BM+} \cap \tilde{P}_{t,s}^{DA} \leq \tilde{P}_{t,s}^{BM} \\ 0, & \tilde{P}_{t,s}^{DA} \geq \tilde{P}_{t,s}^{BM} \end{cases}, \quad b = 1,..,B^{DA,MAX} - 1, t \in T, s \in S \tag{A.35}$$

$$y_{t,s}^{BM-} = \begin{cases} x_{b,t,s}^{BM-}, & \text{if: } P_{b,t,s}^{BM-} \leq \tilde{P}^B M_{t,s} \leq P_{b+1,t,s}^{BM-} \cap \tilde{P}^D A_{t,s} \leq \tilde{P}^B M_{t,s} \\ 0, & \tilde{P}^D A_{t,s} \geq \tilde{P}^B M_{t,s} \end{cases}, \quad b = 1,..,B^{DA,MAX} - 1, t \in T, s \in S \tag{A.36}$$

Monotony increasing bid curve for balancing market

$$x_{b,t,s}^{BM+} \leq x_{b+1,t,s}^{BM+}, \quad b = 1,..,B^{DA,MAX} - 1, t \in T, s \in S \tag{A.37}$$

$$x_{b,t,s}^{BM-} \leq x_{b+1,t,s}^{BM-}, \quad b = 1,..,B^{DA,MAX} - 1, t \in T, s \in S \tag{A.38}$$

Regulating Capacity market, RKOM

Maximum bid in balancing market must be higher than the reserved capacity

$$x_{B^{BM,MAX},t,s}^{DA-} \geq x_t^{RKOM}, \quad t \in T, s \in S, \quad if: \tilde{P}_{t,s}^{DA} \leq \tilde{P}_{t,s}^{BM} \tag{A.39}$$

Reduction in spot capacity due to RKOM

$$y_{t,s}^{DA+} \leq \bar{Q}_{g,i} - x_t^{RKOM}, \quad t \in T, s \in S \tag{A.40}$$

The variable RKOM has to be larger than the Reserve capacity

$$x_t^{RKOM} \geq X_t^{RKOM}, \quad t \in T \tag{A.41}$$

Imbalance settlement write about this in the model description Nonanticipativity constraints

Day-Ahead market

$$x_{b,t,s}^{DA+,-} = x_{b,t,s'}^{DA+,-} \qquad b \in B, t \in T, s, s' \in S \tag{A.42}$$

$$y_{t,s}^{DA+,-} = y_{t,s}^{DA+,-} \qquad , t \in T, s, s' \in S \tag{A.43}$$

Balancing market

$$x_{b,t,s}^{BM+,-} = x_{b,t,s'}^{BM+,-} \qquad b \in B, t \in T, s, s' \in S \tag{A.44}$$

$$y_{t,s}^{BM+,-} = y_{t,s}^{BM+,-} \qquad , t \in T, s, s' \in S \tag{A.45}$$

APPENDIX B

Scenario generation of BM prices

Probabilites of States

State	Price interval From	To	Probability
1	–	−30	0,0027
2	−29	15	0,0128
3	−14	−1	0,4670
4	0	0	0,2080
5	1	14	0,2740
6	15	29	0,0025
7	30	–	0,0028

Table B.1: BM states for the spring period

State	Price interval From	To	Probability
1	–	−30	0,0012
2	−29	15	0,0087
3	−14	−1	0,3489
4	0	0	0,3112
5	1	14	0,3256
6	15	29	0,0041
7	30	–	0,0003

Table B.2: BM states for the summer period

State	Price interval From	To	Probability
1	−	−30	0,002
2	−29	15	0,0119
3	−14	−1	0,3833
4	0	0	0,3100
5	1	14	0,2890
6	15	29	0,0042
7	30	−	0,0010

Table B.3: BM states for the autumn period

Transition Matrices

The transition matrix for the spring:

$$\mathrm{P}_{ij}^{Sp} = \quad \begin{array}{cc} & \mathbf{X(t+1)} \\ \mathbf{X}(t) & \begin{pmatrix} 0.5 & 0.375 & 0.1250 & 0.0 & 0.0 & 0 & 0 \\ 0.0531 & 0.5487 & 0.3717 & 0.0177 & 0.088 & 0 & 0 \\ 0.002 & 0.0084 & 0.8745 & 0.0754 & 0.0415 & 0 & 0 \\ 0.0011 & 0.0016 & 0.1847 & 0.6747 & 0.1368 & 0.005 & 0.005 \\ 0.008 & 0.0008 & 0.0664 & 0.109 & 0.8163 & 0.004 & 0.003 \\ 0 & 0.0 & 0.0455 & 0.0 & 0.4091 & 0.4545 & 0.091 \\ 0.04 & 0.0 & 0.1600 & 0.0 & 0.120 & 0.080 & 0.600 \end{pmatrix} \end{array}$$

Transition matrix for the summer:

$$\mathrm{P}_{ij}^{Su} = \quad \begin{array}{cc} & \mathbf{X(t+1)} \\ \mathbf{X}(t) & \begin{pmatrix} 0.6364 & 0.1818 & 0.1818 & 0 & 0 & 0 & 0 \\ 0.0519 & 0.5195 & 0.3636 & 0.026 & 0.039 & 0 & 0 \\ 0 & 0.0094 & 0.8333 & 0.1139 & 0.044 & 0 & 0 \\ 0 & 0.0011 & 0.1301 & 0.7461 & 0.122 & 0.0007 & 0 \\ 0 & 0.0010 & 0.0431 & 0.1196 & 0.831 & 0.005 & 0 \\ 0 & 0 & 0.0278 & 0.0833 & 0.361 & 0.5 & 0.028 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \end{array}$$

Transition matrix for the autumn:

$$
\mathrm{P}_{ij}^{A} = \quad \begin{array}{cc} & \mathbf{X(t+1)} \\ \mathbf{X}(t) & \begin{pmatrix} 0 & 0 & 0 & 0.5 & 0.5 & 0 & 0 \\ 0 & 0.5673 & 0.3942 & 0.0385 & 0 & 0 & 0 \\ 0 & 0.0122 & 0.8343 & 0.1088 & 0.0438 & 0.0009 & 0 \\ 0.0004 & 0.0015 & 0.1421 & 0.7331 & 0.1226 & 0 & 0.0004 \\ 0.0004 & 0.0004 & 0.0502 & 0.1392 & 0.8043 & 0.0047 & 0.0008 \\ 0 & 0 & 0 & 0.0541 & 0.3784 & 0.54 & 0.0270 \\ 0 & 0 & 0 & 0 & 0 & 0.22 & 0.5556 \end{pmatrix} \end{array}
$$

APPENDIX C

RKOM Market Data

In Table C.1 the volume and price bid in the RKOM-week for 2015. This table only consider RKOM-H bids. A conversion ratio of 9.49 NOK/ €was used to convert NOK to Euros. The average for the whole price range is 4.5 €/MWh.

RKOM-H day

Winter	Week	Volume [MW]	Price [EURO/MWh]
2015	48	460	4.214963119
	48	460	4.214963119
	49	464	3.161222339
	50	130	0.831401475
	51	229	3.477344573
	53	145	0.632244468
2016	2	389	13.69863014
	3	434	11.59114858
	4	444	5.268703899
	5	299	2.10748156
	6	269	1.116965227
	7	371	4.109589041
	8	349	4.214963119

Table C.1: RKOM-day market data for RKOM-H, Winter 2015 and 2016

APPENDIX D

Matlab script for scenario generation

Transition matrix This script generates the transition matrix for each of the presented scenarios in the model. The input to the function is the spot price for a given season. The output is 35 BM price scenarios for a 24 hour period.

```
function [ RKPRICE, T_sp, T_su, T_a ]=TransMat(VarName,season)
%% Import data from spreadsheet
% Script for importing data from the following spreadsheet:
%
%    Workbook: C:\Users\admin\Desktop\Prosjektoppgave\DATASETFORMARKOV.xlsx
%    Worksheet: Sheet1
% Auto-generated by MATLAB on 2015/11/13 10:06:52

%% Import the data
[~, ~, raw] = xlsread('C:\Users\admin\Desktop\Prosjektoppgave\DATASETFORMARKOV.xlsx','Shee
raw(cellfun(@(x) ~isempty(x) && isnumeric(x) && isnan(x),raw)) = {''};

%% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells

%% Create output variable
data = reshape([raw{:}],size(raw));

%% Allocate imported array to column variable names
RK44= data(:,1);
RK34 = data(:,2);
RK24 = data(:,3);
RK14 = data(:,4);
q=size(RK44);
counter1=1;
counter2=1;
counter3=1;
counter4=1;
for i=1:q(1)
    if (~isnan(RK44(i)))
    RK4(counter4,1)=RK44(i);
```

```
    counter4=counter4+1;
    end
    if (~isnan(RK34(i)))
    RK3(counter3,1)=RK34(i);
    counter3=counter3+1;
    end
    if (~isnan(RK24(i)))
    RK2(counter2,1)=RK24(i);
    counter2=counter2+1;
    end
    if (~isnan(RK14(i)))
    RK1(counter1,1)=RK14(i);
    counter1=counter1+1;
    end
end
%% Clear temporary variables
clearvars data raw R;
% Transission Matrix
%Parameters
w1=1418;     %End of winter
w2=8735;     %End of winter 2
sp=3625;     %End of Spring
su=5832;     % End of summer
a=8017;      % End of autumn
RK4=RK4(1:end-1); % Need the same size
RK3=RK3(1:end-1);
% Matrix with all input data
Tot=[RK4 RK3 RK2 RK1];
% Elments of Transission Matrix
Bin=[-200 -30 -15 -0.01 0.01 15 30 200];
Bin2=[-30:1:30];% Price intervall for RK prices
A_test=zeros(size(Bin2,1));
A4=zeros(7);     % Number of probability
MS=size(A4);
ATOT=zeros(7);   % Empty transission matrix
ATOT(:,:,2)=A4; %
ATOT(:,:,3)=A4;
ATOT(:,:,4)=A4;
% For Winter part 1
for l=1:4
    for i=2:w1
        for j=1:MS(1)
            for k= 1:MS(1);
                    if(Tot(i-1,l)<Bin(j+1)&& Tot(i-1,l)>=Bin(j)  && Tot(i,l)>=Bin(k)  && Tot
                    ATOT(j,k,1)=ATOT(j,k,1)+1;
                    end
            end
        end
    end
end
% For winter part 2
for l=1:4
    for i=a:w2
        for j=1:MS(1)
            for k= 1:MS(1);
                    if(Tot(i-1,l)<Bin(j+1)&& Tot(i-1,l)>=Bin(j)  && Tot(i,l)>=Bin(k)  && Tot
                     ATOT(j,k,1)=ATOT(j,k,1)+1;
                    end
            end
```

```
        end
    end
end
% For Spring
for l=1:4
    for i=w1:sp
        for j=1:MS(1)
            for k= 1:MS(1);
                    if(Tot(i-1,l)<Bin(j+1)&& Tot(i-1,l)>=Bin(j) && Tot(i,l)>=Bin(k) && Tot
                    ATOT(j,k,2)=ATOT(j,k,2)+1;
                    end
            end
        end
    end
end
% For Summer
for l=1:4
    for i=sp:su
        for j=1:MS(1)
            for k= 1:MS(1);
                    if(Tot(i-1,l)<Bin(j+1)&& Tot(i-1,l)>=Bin(j) && Tot(i,l)>=Bin(k) && Tot
                    ATOT(j,k,3)=ATOT(j,k,3)+1;
                    end
            end
        end
    end
end
% For Autumn
for l=1:4
    for i=su:a
        for j=1:MS(1)
            for k= 1:MS(1);
                    if(Tot(i-1,l)<Bin(j+1)&& Tot(i-1,l)>=Bin(j) && Tot(i,l)>=Bin(k) && Tot
                    ATOT(j,k,4)=ATOT(j,k,4)+1;
                    end
            end
        end
    end
end
% Calculating the probability of each state
% calculate the value of each observation
% Elements of the probability state.
p_w=zeros(1,7);
p_sp=zeros(1,7);
p_su=zeros(1,7);
p_a=zeros(1,7);
% For Winter part 1
Av_w=zeros(1,7);
for l=1:4
    for i=1:w1
        for j=1:MS(1)
            if(Tot(i,l)<Bin(j+1)&& Tot(i,l)>=Bin(j))
                Av_w(j)=Av_w(j)+ Tot(i,l);
                p_w(j)=p_w(j)+1;
            end
        end
    end
end
% For winter part 2
```

```
for l=1:4
    for i=a:w2
        for j=1:MS(1)
            if(Tot(i,l)<Bin(j+1)&& Tot(i,l)>=Bin(j))
                Av_w(j)=Av_w(j)+ Tot(i,l);
                p_w(j)=p_w(j)+1;
            end
        end
    end
end
% For Spring
Av_sp=zeros(1,7);
for l=1:4
    for i=w1:sp
        for j=1:MS(1)
            if(Tot(i,l)<Bin(j+1)&& Tot(i,l)>=Bin(j))
                Av_sp(j)=Av_sp(j)+ Tot(i,l);
                p_sp(j)=p_sp(j)+1;
            end
        end
    end
end
% For Summer
Av_su=zeros(1,7);
for l=1:4
    for i=sp:su
        for j=1:MS(1)
            if(Tot(i,l)<Bin(j+1)&& Tot(i,l)>=Bin(j))
                Av_su(j)=Av_su(j)+ Tot(i,l);
                p_su(j)=p_su(j)+1;
            end
        end
    end
end
% For Autumn
Av_a=zeros(1,7);
for l=1:4
    for i=su:a
        for j=1:MS(1)
            if(Tot(i,l)<Bin(j+1)&& Tot(i,l)>=Bin(j))
                Av_a(j)=Av_a(j)+ Tot(i,l);
                p_a(j)=p_a(j)+1;
            end
        end
    end
end

% Final transission matrix
% Absoulte probability
A_w=ATOT(:,:,1);
A_sp=ATOT(:,:,2);
A_su=ATOT(:,:,3);
A_a=ATOT(:,:,4);
% Average price for each state
% Caluclate the average price for each intevall given above.
AP_w=Av_w./p_w;
AP_sp=Av_sp./p_sp;
AP_su=Av_su./p_su;
AP_a=Av_a./p_a;
```

```
% Constructuion of Actuall probability matrix
WINTER=p_w/sum(p_w)
SPRING=p_sp/sum(p_sp)
SUMMER=p_su/sum(p_su)
AUTUMN=p_a/sum(p_a)
% P_w=A_w/sum(sum(A_w));
% P_sp=A_sp/sum(sum(A_sp));
% P_su=A_su/sum(sum(A_su));
% P_a=A_a/sum(sum(A_a));
% Construct the transsion matrix
% Divid by all obeserved transsisons in a row.
T_w=bsxfun(@rdivide, A_w, sum(A_w,2));
T_sp= bsxfun(@rdivide, A_sp, sum(A_sp,2));
T_su=bsxfun(@rdivide, A_su, sum(A_su,2));
T_a=bsxfun(@rdivide, A_a, sum(A_a,2));
% Calculating the RK-prices for given season
if(season==1) % Winter
% Creating the first colum of prices
% Uses the spot values taken inn as a parameter in teh function.
RK(:,1)=VarName(:,1);
Counter=[1:7 1:7 1:7 1:7 1:7];
RKs=size(RK);
NumRK=size(AP_w,2);
RK(1:NumRK,1)=RK(1:NumRK,1)+AP_w.';
RK(NumRK+1:2*NumRK,1)=RK(NumRK+1:2*NumRK,1)+AP_w.';
RK(2*NumRK+1:3*NumRK,1)=RK(2*NumRK+1:3*NumRK,1)+AP_w.';
RK(3*NumRK+1:4*NumRK,1)=RK(3*NumRK+1:4*NumRK,1)+AP_w.';
RK(4*NumRK+1:5*NumRK,1)=RK(4*NumRK+1:5*NumRK,1)+AP_w.';
RK(:,1)
end
if(season==2) % Spring
% Creating the first colum of prices
RK(:,1)=VarName(:,1);
Counter=[1:7 1:7 1:7 1:7 1:7];
RKs=size(RK);
NumRK=size(AP_sp,2);
RK(1:NumRK,1)=RK(1:NumRK,1)+AP_sp.';
RK(NumRK+1:2*NumRK,1)=RK(NumRK+1:2*NumRK,1)+AP_sp.';
RK(2*NumRK+1:3*NumRK,1)=RK(2*NumRK+1:3*NumRK,1)+AP_sp.';
RK(3*NumRK+1:4*NumRK,1)=RK(3*NumRK+1:4*NumRK,1)+AP_sp.';
RK(4*NumRK+1:5*NumRK,1)=RK(4*NumRK+1:5*NumRK,1)+AP_sp.';
RK(:,1)
end
if(season==3) % Summer
% Creating the first colum of prices
RK(:,1)=VarName(:,1);
Counter=[1:7 1:7 1:7 1:7 1:7];
RKs=size(RK);
NumRK=size(AP_su,2);
RK(1:NumRK,1)=RK(1:NumRK,1)+AP_su.';
RK(NumRK+1:2*NumRK,1)=RK(NumRK+1:2*NumRK,1)+AP_su.';
RK(2*NumRK+1:3*NumRK,1)=RK(2*NumRK+1:3*NumRK,1)+AP_su.';
RK(3*NumRK+1:4*NumRK,1)=RK(3*NumRK+1:4*NumRK,1)+AP_su.';
RK(4*NumRK+1:5*NumRK,1)=RK(4*NumRK+1:5*NumRK,1)+AP_su.';
RK(:,1)
end
if(season==4) % Autumn
% Creating the first colum of prices
RK(:,1)=VarName(:,1);
```

```
Counter=[1:7 1:7 1:7 1:7 1:7];
RKs=size(RK);
NumRK=size(AP_a,2);
RK(1:NumRK,1)=RK(1:NumRK,1)+AP_a.';
RK(NumRK+1:2*NumRK,1)=RK(NumRK+1:2*NumRK,1)+AP_a.';
RK(2*NumRK+1:3*NumRK,1)=RK(2*NumRK+1:3*NumRK,1)+AP_a.';
RK(3*NumRK+1:4*NumRK,1)=RK(3*NumRK+1:4*NumRK,1)+AP_a.';
RK(4*NumRK+1:5*NumRK,1)=RK(4*NumRK+1:5*NumRK,1)+AP_a.';
RK(:,1)
end
% Generating the random RK prices for the given season
    for i=2:24
        for j=1:RKs(1)
            for k=1:RKs(1)
                if (RK(j,i-1)==RK(k,1))
                PreRow=k;
                end
            end
            if (season==1)
            Prob=T_w(Counter(PreRow),:);
            Y=randsample([1:7],1,true,Prob);
            var1=7*(floor((j-1)/7));
            RK(j,i)=RK(Y+var1,1);
            end
             if (season==2)
            Prob=T_sp(Counter(PreRow),:);
            Y=randsample([1:7],1,true,Prob);
            var1=7*(floor((j-1)/7));
            RK(j,i)=RK(Y+var1,1);
             end
             if (season==3)
            Prob=T_su(Counter(PreRow),:);
            Y=randsample([1:7],1,true,Prob);
            var1=7*(floor((j-1)/7));
            RK(j,i)=RK(Y+var1,1);
             end
             if (season==4)
            Prob=T_a(Counter(PreRow),:);
            Y=randsample([1:7],1,true,Prob);
            var1=7*(floor((j-1)/7));
            RK(j,i)=RK(Y+var1,1);
            end
        end
    end
% Test figur of surface plot of the
% figure
% surf(A_test)
% colormap hsv
% colorbar
% Writing the results to an excel file
% filename='DATASETFORMARKOV1.xlsx';
% A_W=xlswrite(filename,A_w,2,'B3:H9');
% p_w=xlswrite(filename,p_w,2,'J3:P11');
% A_sp=xlswrite(filename,A_sp,2,'B13:H19');
% p_sp=xlswrite(filename,p_sp,2,'J13:P21');
% A_su=xlswrite(filename,A_su,2,'B23:H29');
% p_su=xlswrite(filename,p_su,2,'J23:P31');
% A_a=xlswrite(filename,A_a,2,'B33:H39');
% p_a=xlswrite(filename,p_a,2,'J33:P41');
```

```
RK;
RK_Out=zeros(size(RK,1),size(RK,2));
RK_Out(:,1)=RK(:,1);
% Scale the prices for every time of the day
for j=2:size(RK,2)
    for i=1:size(RK,1)
            if( i<8)
            RK_Out(i,j)=(RK(i,j)/RK(4,1))*VarName(i,j);
            elseif (i>=8 && i<15)
            RK_Out(i,j)=(RK(i,j)/RK(11,1))*VarName(i,j);
            elseif( i>=15 && i<22)
            RK_Out(i,j)=(RK(i,j)/RK(18,1))*VarName(i,j);
            elseif( i>=22 && i<29)
            RK_Out(i,j)=(RK(i,j)/RK(25,1))*VarName(i,j);
            elseif (i>=29 && i<36)
                if(RK(i,j)<0)
                    RK_Out(i,j)=(RK(i,j)/RK(32,1))*VarName(i,j);
                elseif(RK(i,j)>0)
                    RK_Out(i,j)=(RK(i,j)*(1-VarName(i,j)/RK(32,1)));
                end
            end
    end
end

% Using absolute value
for j=2:size(RK,2)
    for i=1:size(RK,1)
            if( i<8)
            RK_Out(i,j)=RK(i,j) + (RK(4,1)-VarName(i,j));
            elseif (i>=8 && i<15)
            RK_Out(i,j)=RK(i,j) + (RK(11,1)-VarName(i,j));
            elseif( i>=15 && i<22)
            RK_Out(i,j)=RK(i,j) + (RK(18,1)-VarName(i,j));
            elseif( i>=22 && i<29)
            RK_Out(i,j)=RK(i,j) + (RK(25,1)-VarName(i,j));
            elseif (i>=29 && i<36)
            RK_Out(i,j)=RK(i,j) + (RK(32,1)-VarName(i,j));
            end
    end
end

% if(season==1)
%     T=T_w;
% elseif(season==2)
%     T=T_sp;
% elseif(season==3)
%     T=T_su;
% elseif(season==4)
%     T=T_a;
% end

RKPRICE=RK_Out;
return
end
```

Import of DA prices This script import the DA prices for the each season and printes the BM prices scenarios generated by Transmat.m to a textfile.

```
clc
clear all
%% Import data from text file.
% Script for importing data from the following text file:
%
%    C:\Users\admin\Desktop\Model_inProgress\Uke1\TokkeVinje_misc_Seasons.dat
%
% To extend the code to different selected data or a different text file,
% generate a function instead of a script.

% Auto-generated by MATLAB on 2015/11/14 15:11:44

%% Initialize variables for week 1
filename1 = 'C:\Users\admin\Desktop\Model_inProgress\Uke1\TokkeVinje_misc_Seasons.dat';
filename2 = 'C:\Users\admin\Desktop\Model_inProgress\Uke14\TokkeVinje_misc_Seasons.dat';
filename3 = 'C:\Users\admin\Desktop\Model_inProgress\Uke27\TokkeVinje_misc_Seasons.dat';
filename4 = 'C:\Users\admin\Desktop\Model_inProgress\Uke44\TokkeVinje_misc_Seasons.dat';




%% Open the text file.
%UKE1
fileID1 = fopen(filename1,'r');
%UKE14
fileID2 = fopen(filename2,'r');
%UKE27
fileID3 = fopen(filename3,'r');
%UKE44
fileID4 = fopen(filename4,'r');
Names=[fileID1 fileID2 fileID3 fileID4];
for i=1:4
%% Read columns of data according to format string.
delimiter = {'\t',','};
startRow = 136;
endRow = 174;
%% Read columns of data as strings:
% For more information, see the TEXTSCAN documentation.
formatSpec = '%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s%s

textscan(Names(i), '%[^\n\r]', startRow-1, 'ReturnOnError', false);
dataArray = textscan(Names(i), formatSpec, endRow-startRow+1, 'Delimiter', delimiter, 'Ret
fclose(Names(i));
%% Convert the contents of columns containing numeric strings to numbers.
% Replace non-numeric strings with NaN.
raw = repmat({''},length(dataArray{1}),length(dataArray)-1);
for col=1:length(dataArray)-1
    raw(1:length(dataArray{col}),col) = dataArray{col};
end
numericData = NaN(size(dataArray{1},1),size(dataArray,2));

for col=[1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,
    % Converts strings in the input cell array to numbers. Replaced non-numeric
    % strings with NaN.
    rawData = dataArray{col};
    for row=1:size(rawData, 1);
        % Create a regular expression to detect and remove non-numeric prefixes and
        % suffixes.
        regexstr = '(?<prefix>.*?)(?<numbers>([-]*(\d+[\,]*)+[\.]{0,1}\d*[eEdD]{0,1}[-+]*\
```

```
        try
            result = regexp(rawData{row}, regexstr, 'names');
            numbers = result.numbers;

            % Detected commas in non-thousand locations.
            invalidThousandsSeparator = false;
            if any(numbers==',');
                thousandsRegExp = '^\d+?(\,\d{3})*\.{0,1}\d*$';
                if isempty(regexp(thousandsRegExp, ',', 'once'));
                    numbers = NaN;
                    invalidThousandsSeparator = true;
                end
            end
            % Convert numeric strings to numbers.
            if ~invalidThousandsSeparator;
                numbers = textscan(strrep(numbers, ',', ''), '%f');
                numericData(row, col) = numbers{1};
                raw{row, col} = numbers{1};
            end
        catch me
        end
    end
end


%% Replace non-numeric cells with NaN
R = cellfun(@(x) ~isnumeric(x) && ~islogical(x),raw); % Find non-numeric cells
raw(R) = {NaN}; % Replace non-numeric cells
%% Create output variable
TokkeVinjemiscSeasons(:,:,i) = cell2mat(raw);
%% Clear temporary variables
clearvars filename delimiter startRow endRow formatSpec fileID dataArray ans raw col numer

end

TokkeVinjemiscSeasons=TokkeVinjemiscSeasons(3:end-2,2:25,:);
% Winter=TokkeVinjemiscSeasons(:,:,1);
% Spring=TokkeVinjemiscSeasons(:,:,2);
% Summer=TokkeVinjemiscSeasons(:,:,3);
Autumn=TokkeVinjemiscSeasons(:,:,4);

% RKW=TransMat(Winter,1);
% RKSp=TransMat(Spring,2);
% RKSu=TransMat(Summer,3);
[ RKPRICE, T_sp, T_su, T_a ]=TransMat(Autumn,4);
format short
Sen(1:35,1)=1:35;
% RKW=[Sen RKW];
% RKSp=[Sen RKSp];
% RKSu=[Sen RKSu];
%RKWA=[Sen RKWA];
% T_w=RKW(2);
% T_sp=RKSp(2);
% T_su=RKSu(2);
% T_a=RKWA(2);

% %% Initialize variables for Export
% filename = 'C:\Users\admin\Desktop\Model_inProgress\Uke1\TokkeVinje_misc_Seasons1.dat';
% delimiter = {'\t',','};
```

```
% startRow = 136;
% endRow = 174;
 % Write the RK price to a text file.
senarios=35;
hours=24;
% % Winter
%  fileID=fopen('RKW.txt','w');
% fprintf(fileID,'%s\n', '# Balancing market', 'param RKPrice (tr):');
% for i=1:hours
%     fprintf(fileID,'\t');
%     fprintf(fileID,'%d\t',i);
%     if(i==hours)
%         fprintf(fileID,'%s\n',':=');
%     end
% end
%  fprintf(fileID,['s' repmat('%.2f\t',1,size(RKW,2)) '\n'],RKW');
%  fprintf(fileID,'%s',';');
%  fclose( fileID);
%  % Spring
%  fileID=fopen('RKSp.txt','w');
% fprintf(fileID,'%s\n', '# Balancing market', 'param RKPrice (tr):');
% for i=1:hours
%     fprintf(fileID,'\t');
%     fprintf(fileID,'%d\t',i);
%     if(i==hours)
%         fprintf(fileID,'%s\n',':=');
%     end
% end
%  fprintf(fileID,['s' repmat('%.2f\t',1,size(RKSp,2)) '\n'],RKSp');
%  fprintf(fileID,'%s',';');
%  fclose( fileID);
%  % Summer
%  fileID=fopen('RKSu.txt','w');
%  fprintf(fileID,'%s\n', '# Balancing market', 'param RKPrice (tr):');
% for i=1:hours
%     fprintf(fileID,'\t');
%     fprintf(fileID,'%d\t',i);
%     if(i==hours)
%         fprintf(fileID,'%s\n',':=');
%     end
% end
%  fprintf(fileID,['s' repmat('%.2f\t',1,size(RKSu,2)) '\n'],RKSu');
%  fprintf(fileID,'%s',';');
%   fclose( fileID);
 % Autumn
%  fileID=fopen('RKA.txt','w');
% fprintf(fileID,'%s\n', '# Balancing market', 'param RKPrice (tr):');
% for i=1:hours
%     fprintf(fileID,'\t');
%     fprintf(fileID,'%d\t',i);
%     if(i==hours)
%         fprintf(fileID,'%s\n',':=');
%     end
% end
%  fprintf(fileID,['s' repmat('%.2f\t',1,size(RKWA,2)) '\n'],RKWA');
%  fprintf(fileID,'%s',';');
%   fclose( fileID);
```

APPENDIX E

Python Scripts for input files to ScenTreeGen and AMPL

inAMPL

inAMPL.py is the main script that is run each time the Python program is executed. It is possible to executed three different kinds of script form the main file. Scenario generation E, Prodrisk and a script that rewrites the initial inflow, reservoir levels, discharge in the data input file to AMPL.

```
from shopWV2ampl import convertSHOP
#from Writingscenario import write_Scenario
import numpy
from Write2scengen import Write2scengen
from Write2scengen2 import Write2scengen2
import os
from Write2scengenVer2 import Write2scengenVer2
from Write2scengen2Ver2 import Write2scengen2Ver2
from Write2scengenUnion import Write2scengenUnion
from XML import XMLreader
from Writenewday import WriteNewDay
from xml.dom.minidom import parse
import xml.dom.minidom
from Writingscenario import write_Scenario
from Runprog import openFile
from WriteFirstDay import WriteFirstDay
import subprocess
import time
import os.path
from InnFlow import writingInflow
from InnFlow import nameChange
import InnFlow
from ShortFinal import shortFinal
from GUI import GUI
from xlwt import Workbook
from ScenarioTree import scenarioreduction
from SENGEN import write_Scenario1
from detd import detd
from os.path import join, dirname, abspath
```

```
from DAPriceGen import *
import xlrd  as WorkbookREAD
import numpy as np
import detd2ampl as DETD
from InnFlow import *
import collections
import wvfunc
import shopWV2ampl as shop2A
from VanSam import runprog
from Bidcurve import Bidcurvewrite
import sys
def sumColumn(m, column):
    total = 0
    for row in range(len(m)):
        total += m[row][column]
    return total



#Innput to detd2ampl
#week = 4       # the week number that should be used for data that has time variation in
#day = 1
#mnd = 'JAN'
#tmax = 24   # TMP! number of time periods. Need to be coordinated with time/tree-input
#smax = 20   # TMP! number of scenarios.  Need to be coordinated with time/tree-input
#startaar = 1980 #First year with inflow in AMPL


ScenGen=raw_input("Run Scenario Generation? YES/NO ")
Prod=raw_input("Run ProdRisk? YES/NO ")
Data=raw_input("Generate Data set Tokke Vinje? YES/NO ")


## Input parameters to modell

#Write final file to
# Overview of functions to run
if ScenGen == 'YES':
    Scenarios = input("Number of Scenarios? ")
    RKOMPrice=input("RKOMPrice? ")
    week=raw_input("WEek number?")
    runScenGen=True
    scenarioRK=Scenarios
    scenarioDA=5
else:
    runScenGen=False
if Prod=='YES':
    runProdRisk=True
else:
    runProdRisk=False
if Data=='YES':
    week=int(raw_input("Week number?"))
    runDataGen=True
else:
    runDataGen=False

if runScenGen:
    rkom=open("inputprice/tokkevinje_misc_seasonsrkomweek.txt","w")
    rkomtxt=open('inputprice/inputrkomtxt.txt','r')
```

```
    scenarioreduction(scenarioRK,scenarioDA,week)
    stringtoxml="inputfileday7red.xml"
    day=xml.dom.minidom.parse(stringtoxml)
    scenarios,pricesfin,nodes,start,stops=XMLreader(day,7)
    tmax=168
    thelist=write_Scenario(rkom,scenarioRK,scenarios,pricesfin,nodes,tmax,rkomtxt,start,st


# writing innflowfile
if runDataGen:
    inndetd="skienvassdraget"
    DETD.convert(inndetd)
    ###writing new inflow file
    datafile=open('skienvassdraget.dat','r')
    dataout=open('inputprice/tokkevinje_data1.txt','w')
    nameChange(datafile,dataout)
    fromampl=open('inputprice/autumn/UKE1/resultsrkomout.txt','r')
    time.sleep(6)
    datafile1=open('inputprice/tokkevinje_data1.txt','r')
    out=open('inputprice/tokkevinje_data.txt','w')
    init(datafile1,out,fromampl)


if runProdRisk:
    innDETD="skienvassdraget"
    runprog('prodrisk','prodrisk.inn')
    shop2A.convertSHOP(innDETD,reductionOfSHOPWV=False)


#innx='10Scenario.xls'
#out='bidcurve3.9.xls'
#sheet='Sheet2'
#Bidcurvewrite(innx,out,sheet)
```

ScenarioTree

ScenariTree.py contains the code that is used to generate the full scenario tree for the AMPLweek model. This is described in Section 5.1. It uses the script E.1 to generate the first day of operation and the script E.1 to generate all the consqutive days after the first day.

```
from XML import XMLreader
from Writenewday import WriteNewDay
from xml.dom.minidom import parse
import xml.dom.minidom
from Writingscenario import write_Scenario
from WriteFirstDay import WriteFirstDay
import subprocess
import time
import os.path
import time
import matlab.engine
import os
```

```
def scenarioreduction(scenarioRK,scenarioDA,week):
    # Open Matlab Engine
    eng=matlab.engine.start_matlab()
    [data1, data2]=eng.dataInput(nargout=2)
    # Probability Input
    file='Prices.xlsx'
     #Number of days
   # Write first day
    DA=open('inputPrice/Autumn/autumn'+str(week)+'1.txt','r')
    f=open('InputfileDay1.txt','w')
    probRK=WriteFirstDay(f,file,DA,data1,data2,eng)
    subprocess.Popen('scentreegen/scentreegen.exe InputfileDay1.txt')
    Daycounter=1
    for i in range(6):
        time.sleep(3)
        stringToXML="InputfileDay" +str(Daycounter)+"Red.xml"
        DayInnString="InputfileDay" +str(Daycounter)+"Red.txt"
        DayOutString="InputfileDay" +str(Daycounter+1)+".txt"
        DAPriceString='inputPrice/Autumn/autumn'+str(week)+str(Daycounter+1)+'.txt'
        DAPrice=open(DAPriceString,'r')
        while not(os.path.isfile(DayInnString)):
            time.sleep(2)
        DayInn=open(DayInnString,"r")
        DayOut=open(DayOutString,"w")
        DAY=xml.dom.minidom.parse(stringToXML)
        scenarios,pricesFin,nodes,Start,Stops=XMLreader(DAY,Daycounter)
        start_time=time.time()
        WriteNewDay(DayInn,DayOut,DAPrice,scenarios,pricesFin,Daycounter,scenarioRK,scenar
        subprocess.Popen('scentreegen/scentreegen.exe '+DayOutString)
        print("--- %s seconds ---" % (time.time() - start_time))
        Daycounter+=1
    time.sleep(4)
    Daycounter=1
    for i in range(6):
        os.remove("InputfileDay" +str(Daycounter)+"Red.xml")
        os.remove("InputfileDay" +str(Daycounter)+"Red.txt")
        os.remove("InputfileDay" +str(Daycounter)+".txt")
        os.remove("InputfileDay" +str(Daycounter)+"Out.txt")
        Daycounter+=1
```

E.1 WriteFirstDay

WriteFirstDay.py generates the scenario tree for the first day of operation in AMPLWeek.

```
import math
import sys
from matlab.mlarray import double
import matlab.engine

def WriteFirstDay(f, prices,DA,data1,data2,eng):
    scenariosRK=35
    workbook=xlrd.open_workbook(prices)
    sheet=workbook.sheet_by_index(1)
    rows=sheet.nrows
```

```
probRK=[]
probDA=[]
for i in range(4):
    probDA.append(0.2)
for i in range(35):
    probRK.append(sheet.cell(i+1,5).value)
countRk=0
for i in range(len(probRK)):
    countRk=countRk+probRK[i]
#Here you can create a list maker for the number of random variables
RK_Scen=7
scenarioDA=5
days=24
DANodes=5*days
RKnodes=scenariosRK*days
nodes=1+DANodes+RKnodes
DABranchID=[ 1, 9, 17,25, 33]
RKBranchID=[]
RKnodes=scenariosRK*days
DA_Array2=[]
RK_Array2=[]
DA2BranchID=[]
RK2BranchID=[]
DA_Array=numpy.zeros((35,24))
for i in DABranchID:
    for j in range(7):
        RKBranchID.append(i+1+j)
for i in range(scenariosRK):
        inner_listDA =DA.readline().split('\t')
        inner_list2DA=inner_listDA[0].split()
        for j in range(len(inner_list2DA)):
            DA_Array[i,j]=float(inner_list2DA[j])

data_list = DA_Array.tolist()
toMatlab=matlab.double(data_list)
RK1=eng.TransMat(toMatlab,1,data1,data2)
RK_Array=RK1
f.write('# tree specification')
f.write("\n")
f.write('TYPE TREE')
f.write("\n")
f.write("NODES ")
f.write("%d" % 961)
f.write("\n")
f.write("RANDOM "+ "%d" % 1)
f.write("\n")
f.write("DATA\n")
f.write("NODE PRED PROB Price TIME BRANCHID\n")
count=1;
# First Write Spot price
# Writes the first 961 nodes, that is the first day of the production
for i in range(nodes):
    if i<DANodes+1:
        if i<1:
            f.write("%d  " % 1)
            f.write("%d  " % 1)
            f.write("%d  " % 1)
            f.write("%d  " % 200)
            f.write("%d  " % 111111111111)
```

```
                f.write("%d  " % 1)
                f.write("\n")
            else:
                a=i+1
                f.write("%d  " % a)
                if i<6 and i>=1:
                    f.write("%d  " % 1)
                else:
                    b=i-4
                    f.write("%d  " % b)
                c=probDA[0]
                f.write("%s  " % c)
                d=DA_Array[int(7*(i-1)-7*int(math.floor((i-1)/scenarioDA)*scenarioDA))][in
                f.write("%s  " % d)
                f.write("%d  " % 11111111111)
                e=DABranchID[int((i-1)-math.floor(i/5)*5)]
                f.write("%d  " % e)
                f.write("\n")
        else:
          count+=1
          ff=i+1
          f.write("%d  " % ff)
          if i<156 and i>=DANodes+1:
            pre=DANodes+2-scenarioDA+math.floor((i-1-DANodes)/RK_Scen)
            f.write("%d  " % pre)
          else:
            g=i+1-35
            f.write("%d  " % g)
          h=probRK[int(i-121-math.floor((i-121)/scenariosRK)*scenariosRK)]
          f.write("%s  " % h)
          j=RK_Array[int(i-121-int(math.floor((i-121)/scenariosRK)*scenariosRK))][int(math
          f.write("%s  " % j)
          f.write("%d  " % 111111111111)
          k=RKBranchID[int((i-1)-DANodes-math.floor(((i-1)-DANodes)/len(RKBranchID)*len(RK
          f.write("%d  " % k)
          f.write("\n")
    #Here you can create a list maker for the number of random variables
    f.write("END")
    f.close()
    return(probRK)
```

Writenewday WritenewDay.py is used to append a new day of operation to an all ready existing scenario tree. The script uses a Matlab-engine to run the Matlab scripts described in Appendix D. WritenewDay is used before each of the stage wise reductions of the scenario tree presented in Section 5.1.

```
from xlwt import Workbook
from numpy.ma.core import reshape
from matlab.mlarray import double
import array
import sys
import matlab.engine
def WriteNewDay(innfile, f,DA,scentree, pricefin,dayCounter,scenariosRK,scenarioDA,probRK,
    SceGEN=35
    #book = Workbook()
    redfile=innfile.readlines()
    innfile.close()
```

```
#Get number of nodes
for i in range(10):
    if  redfile[i]!='\n':
        nodes=str(redfile[i])
        if str(nodes.split()[0])=="NODES":
            number=int(nodes.split()[1])
            newNumber=number+ 960*scenariosRK
# Scaling list for the DA prices
ScalingList=scalePrices(scentree, pricefin)
# Number of previouse nodes
# Number of branch ID
rows=scentree.shape[0]
col =scentree.shape[1]
#List of branch ids
branchList=list()
prevList=list()
ListOfBID=list(range(1,rows+1))
UpdatedList=list(range(1,rows+1))
ProbList=list()
count=1
for i in range(rows):
    if scentree[i,col-1]>0:
        branchList.append(int(count))
        ProbList.append(scentree[i,col-1])
    count +=1
RC=1
# Updates BRanchID list
for i in range(rows):
    if ListOfBID[i] in branchList:
        for j in range(rows):
            if j>i:
                UpdatedList[j]+=40
count1=1
# Read inputfile and updates branchlist
for lines in redfile:
    if count1 >9:
        nodeInfo=lines.split()
        nodeNR=nodeInfo[0]
        if nodeInfo[len(nodeInfo)-1] != "END":
            branchID=int(nodeInfo[len(nodeInfo)-1])
            if branchID-1 in branchList:
                if branchID != branchIDOld:
                   prevList.append(nodeNROld)
            if int(nodeNR) == number:
               prevList.append(nodeNR)
            firstline=lines.split()
            nodeNROld=firstline[0]
            branchIDOld=int(firstline[len(firstline)-1])
    count1+=1
UpEndBranch=list()
for i in range(rows):
    if i +1 in branchList:
        UpEndBranch.append(UpdatedList[i])
# Write the old file with new BRanchID
WC=0
for lines in redfile:
    if WC==4:
        f.write(lines.split()[0]+" "+ str(newNumber) + "\n")
    if WC>9:
```

```
            if lines.split()[0] != "END":
                lastString=int(lines.split()[5])
                if lastString in ListOfBID:
                    for i in range(5):
                        f.write(lines.split()[i]+" ")
                    f.write(str(UpdatedList[ListOfBID.index(lastString)]))
                    f.write("\n")
                else:
                    f.write(lines)
        elif WC!=4:
            f.write(lines)
        WC+=1
    RKprob1=probRK
    days=24
    DANodes=scenarioDA*days
    RKnodes=scenariosRK*days
    DA_Array=NP.zeros((35,24))
    # Read prices from files both RK and DA
    for i in range(SceGEN):
        inner_listDA =DA.readline().split('\t')
        inner_list2DA=inner_listDA[0].split()
        for j in range(len(inner_list2DA)):
            DA_Array[i,j]=float(inner_list2DA[j])

    ## Number of nodes after the first node
    Nodes=number
## Write the second day of production
    # These numbers has to be set again and red by
    DAscenario=scenariosRK*scenarioDA
    RKscenario=scenariosRK*scenariosRK
    DAnodeNew=DAscenario*days
    RKnodesNew=RKscenario*days
    nodesnew=DAnodeNew+RKnodesNew
    DAprob=1/float(DAscenario)
    numDays=days*8
    counter=0
    countdown=0
    a=Nodes+1
    DA=False
    TotalProbDA=0
    TotalProbRK=0
    # Have to make an updated list of branchID

    #Creates Excel sheets to write PRE prices to file
    #sheet1 = book.add_sheet('Sheet 1')
   # sheet2 = book.add_sheet('Sheet 2')
    for i in range(scenariosRK):
        ProbCounter=0
        DAINN=DA_Array
        RKhour=0
        DAhour=0
        DAcount=0
        RKcount=0
        RPC=0
        counter=0
        # ScaledDAlist
        DA1=Scaler(ScalingList[i],DAINN)
        XLROW=DA1.shape[0]
        XLCOl=DA1.shape[1]
```

```
        data_list = DA1.tolist()
        toMatlab=matlab.double(data_list)
        RK1=eng.TransMat(toMatlab,1,data1,data2)
        for k in range(XLROW):
            for l in range(XLCOl-1):
               # sheet1.write(k +36*i,l,DA1[k,l])
                to2=RK1[k][l]
     #         sheet2.write(k +36*i,l,to2)
        for j in range(960):
            RKcount=int(math.floor(RKhour/24))
            DAcount=int(math.floor(DAhour/24))
            if countdown !=0:
                countdown -=1
                if countdown==1:
                    LastNode=a+1
            else:
                DA=False
            f.write("%d  " % a)
            if j % 192==0:
                pre=int(prevList[i])
                f.write("%d  " % pre)
            else:
                if (RPC-25) % 24==0:
                    pre=LastNode
                    f.write("%d  " % pre)
                else:
                    b=a-1
                    f.write("%d  " % b)
            if j % numDays==0:
                DA=True
                countdown=23
                RPC=1
            if DA:
                ProbDA=float(ProbList[i]*0.2)
                if DAhour % 24==0:
                    TotalProbDA +=ProbDA
                f.write("%s  " % ProbDA)
                d=DA1[DAcount*7,int(j % 24)]
                f.write("%s  " % d)
                DAhour +=1
            else:
                CO= int(math.floor(ProbCounter/24))
                ProbRK=float(ProbList[i]*RKprob1[CO])
                if RKhour % 24==0:
                    TotalProbRK+=ProbRK
                f.write("%s  " % ProbRK)
                j=RK1[RKcount][int(j % 24)]
                f.write("%s  " % j)
                RKhour+=1
                ProbCounter+=1
            f.write("%d  " % 11111111111)
            abc=UpEndBranch[i]+math.floor(counter/days)+1
            f.write("%d  " % abc)
            f.write("\n")
            a+=1
            counter+=1
            RPC+=1
    print("Sannsynlighet for DA: " +str(TotalProbDA))
    print(" Sannsynlighet for RK: " + str(TotalProbRK))
```

```
    f.write("END")
    f.close()
    return (RK1)
    #book.save('PreDay'+str(dayCounter+1)+'.xls')

        # Can use mod for the DA RK question.


def scalePrices(scentree, pricefin):
    RowS=scentree.shape[0]
    ColS=scentree.shape[1]
    RowP=pricefin.shape[0]
    ColP=pricefin.shape[1]
    DAlist=list()
    for i in range(RowS):
        if scentree[i,ColS-2]>0:
            if scentree[i,ColS-1]>0:
                a=filter(lambda a: a != 0,pricefin[i,:])
                DAlist.append(a[len(a)-25])
            else:
                b=filter(lambda a: a != 0,pricefin[i,:])
                tempValue=b[len(b)-25]
        else:
            if scentree[i,ColS-1]>0:
                DAlist.append(tempValue)
    return DAlist

def Scaler(DAprice,DAInn):
    Row=len(DAInn)
    Col=len(DAInn[0])
    DAOut=NP.zeros((Row,Col))
    DATest=list()
    for i in range(Row):
        a=float(DAInn[i][0])
        b=float(DAprice)
        DATest.append(abs(b-a))
    index=int(DATest.index(min(DATest)))
    for i in range(Row):
        for j in range(Col):
            if DAInn[i][j]!="\n":
                a=float(DAInn[i][0])
                b=float(DAInn[index][0])
                c=float(DAInn[i][j])
                DAOut[i,j]=(c/a)*b
    return DAOut
```

Writingscenario

When the full scenario tree for AMPLWeek is created Writingscenario.py is used to write the scenario tree to .dat-file that is run as input in the AMPLWeek model.

```
from writeAMPL import write_1DParam
from writeAMPL import write_2DParam
import numpy
```

```
from collections import defaultdict
import collections
from tempfile import TemporaryFile
from xlwt import Workbook

# Writes scenario generation to AMPL,
# Input=
#Spot prices,
#scenarios,
#Bm pries,
#
# Reduces all alphas (constant terms) with "min{c in Cuts} alpha[c]-sum{r}pi[c,r]*v[c,r]"
# Input parameters:
#    - wv     - wvfunc object containing water value function coefficients
#    - f      - file object to be written to
def zerolistmaker(n):
    listofzeros = [0] * n
    return listofzeros



def write_Scenario(f,scenarios,Scentree,priceInput,nodes,Tmax,txt,Starts,Stops,RKOMPrice):
    Scentree=Scentree[:,1:len(Scentree)]
    nodeNames=[]
    d = defaultdict(list)
    # Write the number of scenarios in S
    f.write("set S:=")
    for j in range(scenarios):
        if j<9:
            f.write("s0" + str(j+1) + " ")
            d["n00"].append("s0" + str(j+1))
        else:
            f.write("s" + str(j+1) + " ")
            d["n00"].append("s" + str(j+1))
    f.write(";\n")



    # Writing the map for the nodes
    rows=Scentree.shape[0]
    col =Scentree.shape[1]
    node=0

    for k in range(col-1):
        nrSce=1
        for i in range(rows):
            first=Scentree[i,k]
            if first>0:
                if node < 9:
                    a="n0"+str(node +1)
                else:
                    a="n"+str(node +1)
                node +=1
                for j in range(i,rows):
                        last=col-1
                        sec=Scentree[j,last]
                        if  sec !=0:
                            if nrSce<10:
                                d[a].append("s0"+str(nrSce))
```

```
                            else:
                                d[a].append("s"+str(nrSce))
                            nrSce +=1
                        if j<(rows-1):
                            if Scentree[j+1,k] !=0:
                                break
    # Write number of Nodes
f.write("set N:=")
for j in range(node +36):
    if j<10:
        f.write("n0" + str(j) + " ")
        nodeNames.append("n"+ str(0)+str(j+1))
    else:
        f.write("n" + str(j) + " ")
        nodeNames.append("n"+str(j))
f.write(";\n")

for i in range(node+1):
    if i>=0 and i<=9:
        nrNode="n0"+str(i)
        listOfS=d.get("n0"+str(i))
    elif i>0:
        nrNode="n"+str(i)
        listOfS=d.get("n"+str(i))
    f.write("set NASet["+nrNode +"]:= ")
    for j in d[nrNode]:
        f.write(" "+str(j))
    f.write(";\n")
#OBS OBS Hardcode!!!!!!!

counterS=1
for i in range(node+1,node +scenarios+1):
    a="n"+str(i)
    if counterS<10:
        f.write("set NASet["+a +"]:= "+"s0"+str(counterS)+";\n")
    else:
        f.write("set NASet["+a +"]:= "+"s"+str(counterS)+";\n")
    counterS+=1
f.write("set M:= RKOM ")
listen=["bidDA","DA","bidRK","RK"]
ferdiglist=list()
for i in range(1,(col/2)+1):
    for j in range(4):
            text=listen[j] + str(i)
            ferdiglist.append(text)
            f.write(" "+text)
f.write( " op ;\n")
f.write("set Nm[RKOM]:= n00;\n ")
nodecount=1
columne=0
alter=1
for i in ferdiglist:
    if i =="bidDA1":
        f.write("set Nm["+i +"]:= ")
        f.write(" n00")
        f.write(";\n")
    else:

        if alter==1:
```

```
            listtwo=list()
            f.write("set Nm["+i +"]:= ")
            for k in range(rows):
                    if Scentree[k,columne]>0:
                        if nodecount<10:
                            string=" n0"+str(nodecount)
                            f.write(string)
                            listtwo.append(string)
                        else:
                            string=" n"+str(nodecount)
                            f.write(string)
                            listtwo.append(string)
                        nodecount+=1
            if i=="RK7":
                f.write(";\n")
                f.write("set Nm[op]:= ")
                for l in listtwo:
                     f.write(l)
            f.write(";\n")
            alter+=1
        elif alter ==2:
             f.write("set Nm["+i +"]:= ")
             for l in listtwo:
                 f.write(l)
             f.write(";\n")
             alter-=1
             columne+=1
f.write("param TMax:= " +str(Tmax)+" ; # number of time periods \n")
f.write("param Prob := \n")
probcount=1
for i in range(rows):
    if Scentree[i][col-1]>0:
        if probcount<10:
            f.write("s0"+str(probcount)+" "+ str(Scentree[i][col-1])+ "\n")
        else:
            f.write("s"+str(probcount)+" "+ str(Scentree[i][col-1])+ "\n")
        probcount+=1
f.write(';')
RKOMlines=txt.readlines()
FixCount=0
Fix=False
for RKOMl in RKOMlines:
    if "ss" in RKOMl:
       f.write(RKOMl.split("ss")[0])
    else:
        f.write(RKOMl)
    if "param FixedDelivery :=" in RKOMl:
        Fix=True
    if Fix:
        FixCount+=1
    if FixCount==25:
        for i in range(25,Tmax+1):
            f.write("   "+str(i) +"    "+ str(0)+"\n")
        FixCount=0
        Fix=False
f.write("\n")
for i in range(Tmax):
    f.write("   "+str(i+1))
f.write(" := \n")
```

```
 PriceRow=priceInput.shape[0]
 PriceCol=priceInput.shape[1]
# print(priceInput)
 Starts.remove(0)
 Stops.remove(0)
 DASTART=Starts[::2]
 DASTOP=Stops[0::2]
 RKSTART=Starts[1::2]
 RKSTOP=Stops[1::2]
 #print(DASTART)
 #print(DASTOP)
 DAprices=numpy.zeros((rows,Tmax))
 RKprices=numpy.zeros((rows,Tmax))
 book = Workbook()
 sheet1 = book.add_sheet('Sheet 1')
 sheet2 = book.add_sheet('Sheet 2')
 sheet3 = book.add_sheet('Sheet 3')
 sheet4 = book.add_sheet('Sheet 4')
 #print(priceInput)
 DA1=DASTART
 DA2=DASTOP
 RK1=RKSTART
 RK2=DASTART

 index=zerolistmaker(rows)
 DAscenC=0
 RKscenC=0
 colcount=0
 a=numpy.delete (priceInput[0,:],0)
 priceInput[0,:]=numpy.append(a,0)
 for j in range(col):
     zerocount=0
     DA1=DASTART
     DA2=DASTOP
     RK1=RKSTART
     RK2=RKSTOP
     for i in range(rows):
             if Scentree[i,j]>0:
                     tempListDA=list()
                     tempListRK=list()
                     number=index[i]
                     a=Starts[number]
                     if a!=0:
                         a-=1
                     b=Stops[number]
                     DAcount=0
                     RKcount=0
                     for k in range(a,b):
                         if  Scentree[i,col-1]>0:
                             if j%2==0:
                                     sheet3.write(i,DAcount+DAscenC,priceInput[i,k])
                                     DAprices[i,DAcount+DAscenC]=priceInput[i,k]
                                     DAcount+=1
                             else:
                                 sheet4.write(i,RKcount+RKscenC,priceInput[i,k])
                                 RKprices[i,RKcount+RKscenC]=priceInput[i,k]
                                 RKcount+=1
                             index[i]= number +1
                         else:
```

```
                            if j%2==0:
                                    tempListDA.append(priceInput[i,k])
                                    DAcount+=1
                                    index[i]= number +1
                            else:
                                tempListRK.append(priceInput[i,k])
                                RKcount+=1
                                index[i]= number +1

            elif Scentree[i,col-1]>0:
                DAcount=0
                RKcount=0
                number=index[i]
                if j%2==0:
                    for num in tempListDA:
                        sheet3.write(i,DAcount+DAscenC,num)
                        DAprices[i,DAcount+DAscenC]=num
                        DAcount+=1
                else:
                    for num1 in tempListRK:

                        sheet4.write(i,RKcount+RKscenC,num1)
                        RKprices[i,RKcount+RKscenC]=num1
                        RKcount+=1
    if j%2==0:
        DAscenC+=24
    else:
        RKscenC+=24
for i in range(rows):
    for j in range(col):
        sheet1.write(i,j,Scentree[i,j])
ScenCount=1
for i in range(rows):
        if DAprices[i,0]>0:
            if ScenCount<10:
                f.write("s0"+str(ScenCount)+"    ")
            else:
                f.write("s"+str(ScenCount)+"    ")
            # Only for six days
            for j in range(Tmax):
                   f.write(str(DAprices[i,j]-5) +"    ")
            f.write(" \n")
            ScenCount+=1
f.write(";\n")
f.write("\n")
f.write("param DAMinBidPrice default 0;\n")
f.write("param DAMaxBidPrice default 150;\n")
f.write("# Balancing market\n")
f.write("param RKPrice (tr):\n")
for i in range(Tmax):
    f.write("    "+str(i+1))
f.write(" := \n")
ScenCount=1
print(RKprices[7,0])
for i in range(rows):
        if RKprices[i,0] !=0:
            if ScenCount<10:
                f.write("s0"+str(ScenCount)+"    ")
            else:
```

```
                f.write("s"+str(ScenCount)+"    ")
            for j in range(Tmax):
                    f.write(str(RKprices[i,j]-5) +"    ")
            f.write(" \n")
            ScenCount+=1
f.write(";\n")
f.write("param RKMinBidPrice default -50;\n")
f.write("param RKMaxBidPrice default 200;\n")
f.write("\n")
f.write("# RKOM market\n")
f.write("param RKOMPrice:=\n")
ScenCount=1
for i in range(scenarios):
    if ScenCount<10:
        f.write("s0"+str(ScenCount)+"    ")
    else:
        f.write("s"+str(ScenCount)+"    ")
    f.write(str(RKOMPrice))
    f.write(" \n")
    ScenCount+=1
f.write(";\n")
book.save('ScenPrice.xls')
return (ferdiglist)
```

APPENDIX F

Results from AMPLWeek

Figure F.1 shows that, for very high RKOM prices, the difference in daily RKOM bids are similar as for Figure 5.11, but smaller. This might indicate that there is another binding constraint in the model than Equation 3.16. In the case of a high RKOM price, this can be the production capacity of the plant.

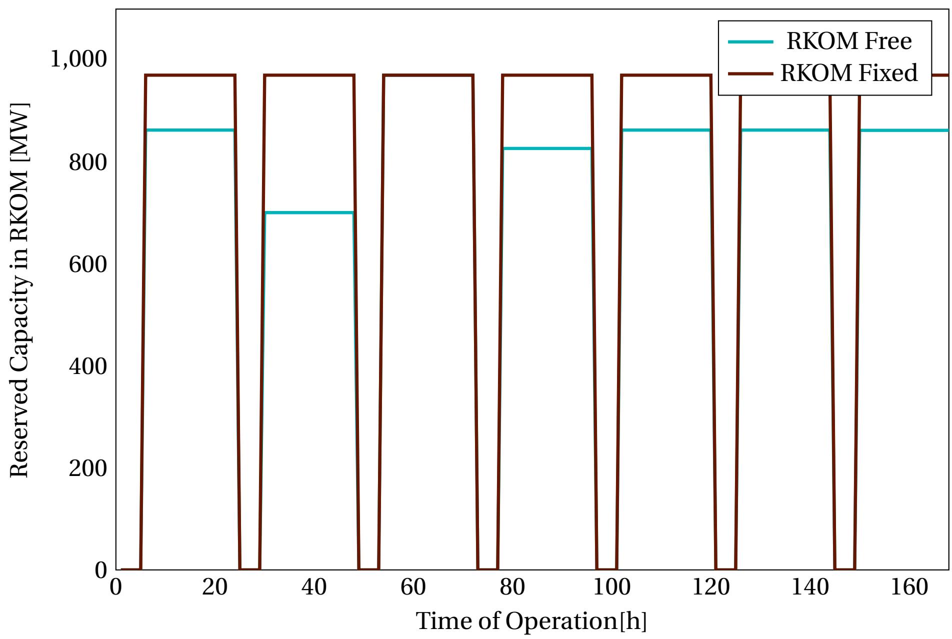


Figure F.1: Fixed and Free RKOM volume for RKOM price= 11.72 €/MW/h

Figure F.2 shows the RKOM bid for a low RKOM price. For a low price, RKOM Free was the same for three first day and zero for day 4, 6 and 7. RKOM Free increased above the

RKOM fixed for day 5.

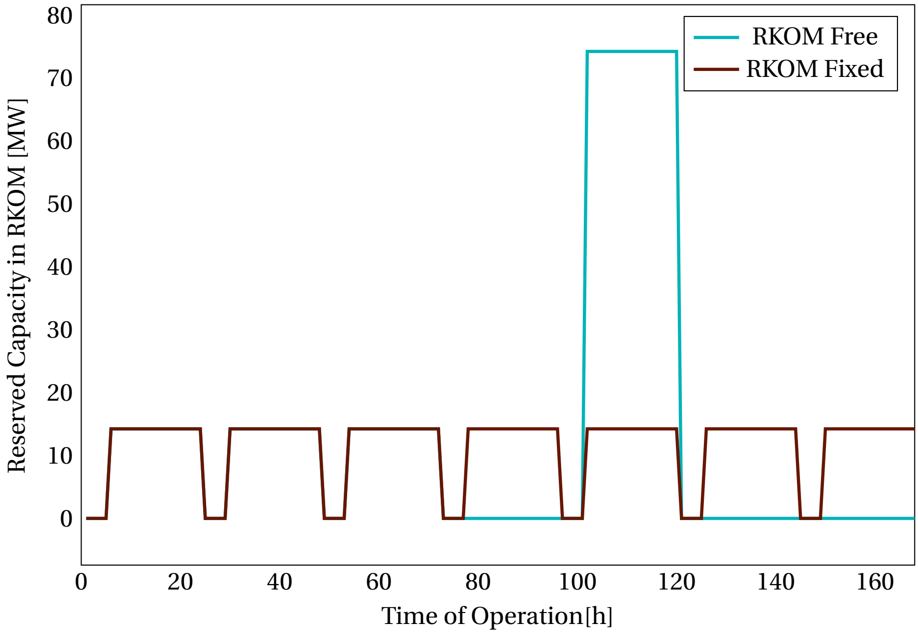


Figure F.2: Fixed and Free RKOM volume for RKOM price= 3 €/MW/h

Figure F.3

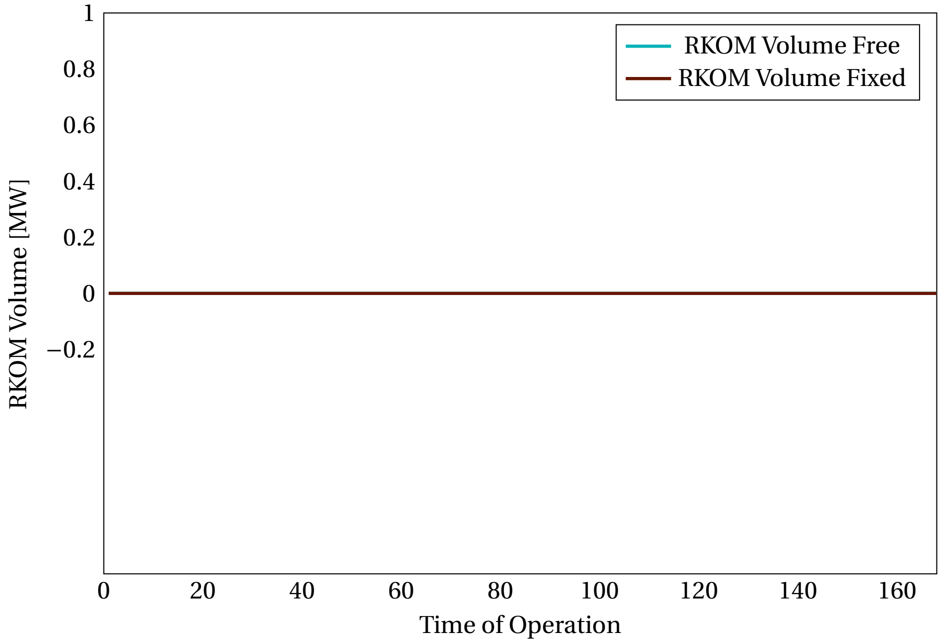


Figure F.3: Fixed and free RKOM volume for RKOM price= 1.13 €/MWh

In Table F.1, the objective value for AMPLWeek for 4 different RKOM prices is shown. The income is given in k€ pr week.

	Zero	Low	Average	High
RKOM Price [€/MW/h]	1.13	3	4.5	11.72
Start cost	20.8	20.8	22.0	27.5
Day-ahead income	3958.2	3928.4	3118.4	754.8
Balancing market income	222.2	247.0	596.3	1094.2
Increased water value	-2415.1	-2415.1	-2107.9	-825.1
RKOM income	0.0	5.7	176.1	1506.9
Objective value	1742.1	1742.7	1758.2	2498.9

Table F.1: Objective value for different RKOM price [k€/Week]

Table F.2 shows weekly average expected delivery in DA, BM and total delivery pr hour for different RKOM prices. The reserved capacity in RKOM is also displayed in the same table.

Table 5.5 display the change in objective value for different RKOM prices between RKOM Fixed and RKOM Free.

	Zero	Low	Average	High
RKOM Price [€/MW/h]	1.13	3	4.5	11.72
Volume RKOM [MW]	0.0	14.2	294.2	970.4
Volume BM [MW]	-24.8	-20.1	60.2	172.4
Volume DA [MW]	651.3	646.7	517.6	137.1
Volume TOT [MW]	626.6	626.6	577.8	309.5

Table F.2: Volume per hour for different RKOM price [MWh/h]

Figure F.4, Figure F.5 and Figure F.6 display the change in the weekly reservoir level for 3 different seasons that were calculated from the seasonal simulation describd in Section 4.3.

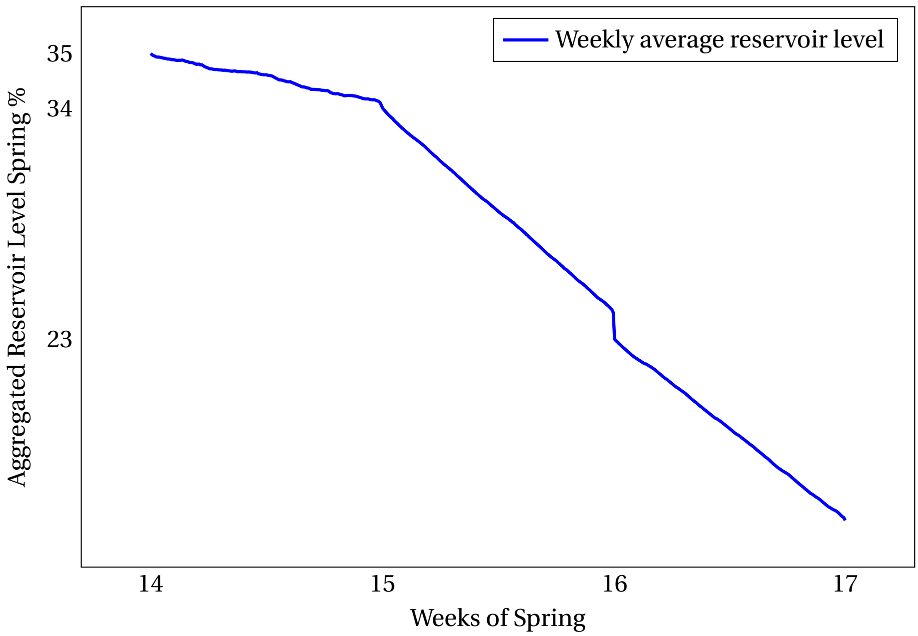


Figure F.4: Weekly change in reservoir level for the Spring season

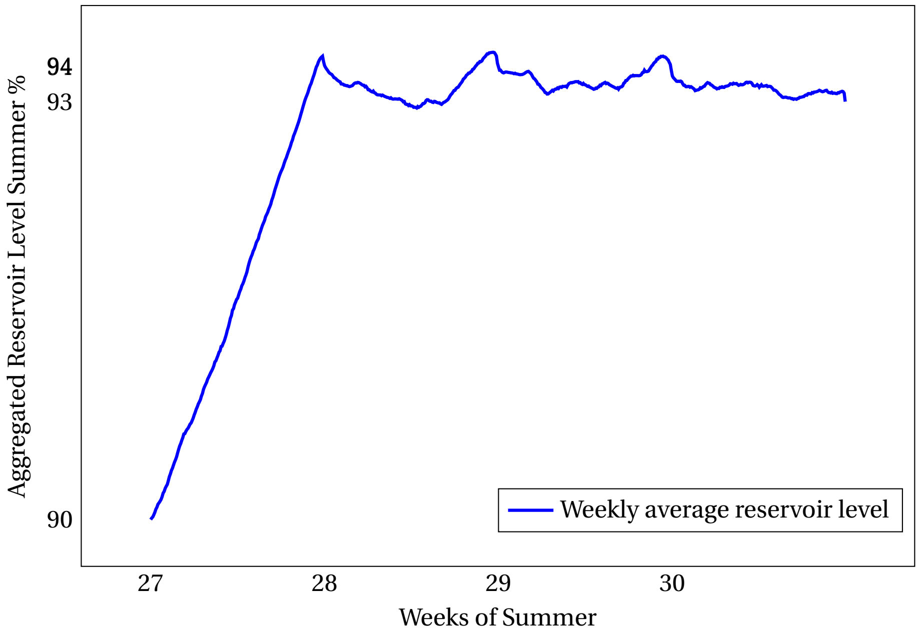


Figure F.5: Weekly change in reservoir level for the summer season

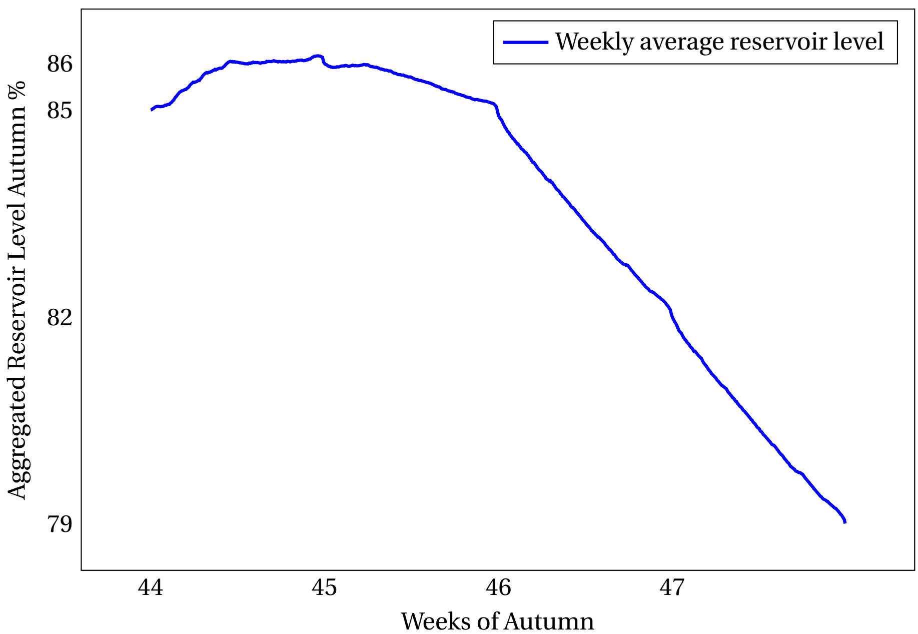


Figure F.6: Weekly change in reservoir level for the autumn season