Unterschrift des Betreuers

TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

DIPLOMARBEIT

Linear Optimization of Hydro Power in Sweden

Ausgeführt am Institut für

Analysis und Scientific Computing

der Technischen Universität Wien

unter der Anleitung von

Univ. Prof. Jens Markus Melenk, PhD

und

Mag. Dr. techn. Gerhard Totschnig

durch

Tobias Wurzer, BSc

Obere Weißgerberstraße 8/12, 1030 Wien

12. September 2013

Tobias Wurzer

ABSTRACT

This thesis contributes to the HiREPS model developed at the Vienna UT, Energy Economics Group - a highly resolved European power system, investment planning and supply security optimization program. In this work, we focus on South and Central Sweden's hydropower portfolio and the hydropower part of the HiREPS model.

Chapter 1 gives a short introduction to Sweden's energy market. In chapters 2 to 4 we build up the knowledge basis (physics of a hydropower plant, used linear optimization method, functionality of the HiREPS hydropower model) for the main chapters, Chapter 5 and Chapter 6.

In Chapter 5 we investigate possibilities to speed up the computation time of the model, involving a treatment of scaling issues, an assessment of redundant internal presolve proceedings of our used modelling language (AMPL) and a treatment of issues regarding the choice of starting values for the discharge in our implemented successive linear optimization. Through our modifications, we are able to cut off nearly 2/3 of the computation time and achieve a faster decrease of the linearization error caused by modelling non-linearity.

Finally, the last chapter shows various simulation results regarding South and Central Sweden's hydropower plants with an installed capacity equal to or greater 5 MW. We include simulated production and profit data for the whole power plant complex for the years 2001 - 2011 to compare it with real data. Furthermore, we show results regarding detailed operation plans for selected power plants. To be precise, we have a close look at some plants along river Ljungan and at one of the few pumped-storage power plants called Kymmen, situated at river Rottnan.

KURZFASSUNG

Diese Diplomarbeit wurde im Rahmen der Entwicklung des HiREPS Modells an der TU Wien, Energy Economics Group geschrieben. HiREPS ist ein hochauflösendes Optimierungsmodell des europäischen Strommarkts zur Berechnung von detaillierten Kraftwerkseinsatzplänen mit Berücksichtigung der Versorgungssicherheit und Möglichkeit zur endogenen Investitionsplanung.

Kapitel 1 gibt einen kurzen Einblick in den schwedischen Energiemarkt zu geben. Die Kapitel 2 bis 4 bauen die Wissenbasis (physikalische Grundlagen eines Wasserkraftwerks, verwendete lineare Optimierungsmethode, Funktionsweise des HiREPS Wasserkraftsmodells) für die nachfolgenden Hauptkapitel auf.

Kapitel 5 untersucht verschiedene Möglichkeiten, die Laufzeit des Modells zu verbessern. Das beinhaltet die Betrachtung von Skalierungsproblemen, die genaue Untersuchung von Redundanz in der Presolve-Phase unserer Modellierungssprache (AMPL) sowie die Suche nach geeigneten Startwerten für die verwendete sukzessive lineare Optimierung. Als Resultat konnten wir die Laufzeit des Modells um fast 2/3 senken. Weiters erzielten wir durch eine intelligente Wahl der Startwerte eine Verbesserung des Linearisierungsfehlers, der durch das Modellieren der auftretenden Nicht-Linearitäten entsteht.

Das letzte Kapitel geht auf die Simulationsergebnisse ein, die das Modell von Süd- und Zentralschweden liefert. Das integrierte Kraftwerksportfolio zählt insgesamt 140 Kraftwerke, wobei nur Kraftwerke mit einer installierten Leistung größer oder gleich 5 MW in Betracht gezogen wurden. Gezeigt werden die simulierten Produktions- und Profitergebnisse des gesamten Kraftwerksparks für die Jahre 2001 - 2011, um sie mit realen Daten zu vergleichen. Außerdem demonstrieren wir detaillierte Kraftwerkseinsatzpläne einiger Kraftwerke verschiedenen Typs. Neben dem Pumpspeicherkraftwerk Kymmen am Fluss Rottnan, haben wir den Fokus auf Kraftwerke im Fluss Ljungan gerichtet.

ACKNOWLEDGEMENTS

Contents

List of figures

List of abbreviations

AEP	Annual Energy Production
$avg.AEP$	Average Annual Energy Production
CO	Convex optimization
DP	Dual Program\Dual Problem
EEG	Energy Economics Group
$HiREPS$	High Resolution Power System
HPL	Hydropower Plant
HSD	Homogeneous Self-Dual Model
IP	Integer-programming
KKT	Karush-Kuhn-Tucker
LO	Linear Optimization
LP	Linear Program
MIP	Mixed-integer-programming
nLO	Nonlinear optimization
NPS	Nord Pool Spot
PP	Primal Program\Primal Problem
$PS-HPL$	Pumped Storage Hydropower Plant
QO	Quadratic Optimization
RES	Renewable Energy Sources
$RR-HPL$	Run-off-the-river Hydropower Plant
$S-HYPE$	Hydrological Predictions for the Environment - Sweden

SLO Successive Linear Optimization

$SMHI$ Swedish Meteorological and Hydrological Institute

$ST - HPL$ Storage Power Plant

$TH - HPL$ Threshold Hydropower Plant

$YST - HPL$ Year-storage Hydropower Plant

Chapter 1

Introduction

In the last years, energy has risen to one of the most discussed topics. Since the energy demand has increased enormously in the last decade and this increase is not likely to fade out in the future, these discussions are of course justified. The era of energy from fossil fuels definitely has a termination date. We do not exactly know when it will happen, but we can absolutely say that it will happen. The need to discuss strategies and make decisions, to prepare for the time when fossil energy sources are gone, does not depend on how long we have until the sources are dried up. Then it is already too late. Thus, we have to come up with answers now, how a future energy system might work out with a high share of renewable energy sources (RES). But decisions cannot be made blindly and without a good basis to build on. Hence, there is a tremendous need to firstly identify upcoming difficulties for the future energy market and to secondly simulate and model different scenarios to avoid these problems on the path to decarbonization.

One of the main issues right now in using RES is the volatility of wind power and solar power: wind and solar radiation are very inconsistent and hardly predictable due to the fact that they are highly dependent on the weather conditions. Hence, it is a challenge to integrate these forms of energy into the existing market and at the same time ensure the safety of supply.

One of the major possibilities to balance high shares of wind and solar power is provided by the storage capabilities of hydropower. All hydropower plants (HPL), except those which are operated as run-of-the-river power plants[1], have the ability to impound water, thus to store it as potential energy and later use it when the demand is higher. Moreover, pumped storage hydropower plants (PS-HPL) have both, the ability to store and to produce energy, which is particularly important in order to balance the electricity grid. A lot of energy, fed into the network by intermittent renewables, can throw the grid off balance. In this case PS-HPLs drain power through pumping water into the reservoirs and thus contribute to compensate this imbalance.

The model described in this thesis seizes the idea of the mentioned need to simulate the

---

[1]Run-of-the-river power plants turbinate the water as it comes down the river and do not impound it.

energy market before rushing into false decisions. Gladly, politics and economy share this opinion. Thus it is not surprising, that this thesis is part of an extensive research project funded by means of a governmental funding program.

1.1   Underlying Research Project

This thesis is part of the *AutRES100* research project at the *Energy Economics Group* (EEG), Institute of Energy Systems and Electrical Drives at Vienna University of Technology, which was funded by means of the Austrian "Klima- und Energiefonds"[2] (Climate and Energy Fund) in the context of the funding program "Neue Energien 2020" (New Energies 2020).

> The New Energies 2020 program supports innovations that make a significant contribution to reducing the burden on the climate and increasing efficiency. The key goals in this context are to develop a sustainable energy system and to increase competitiveness by gaining a technological edge with national economic benefits. The focus of funding is on energy efficiency, renewable sources of energy, smart energy systems and electromobility.[3]

AutRES100, in particular, was conducted in collaboration with ZAMG (*Central Institution for Meteorology and Geodynamics*)[4], *Wegener Center for Climate and Global Change*[5] and *VERBUND Hydro Power*[6]. The main idea was to investigate the economical and technical feasibility of a power system with a high share of RES. The target was in fact to cover the demand with 100% renewables, mostly through hydropower, wind and solar energy, and to test whether this would effect the security of supply and the reliability of the system.

Key questions that press for an answer also include the uncertain future role of pumped hydro storages as well as other storage technologies and the prospective adjustments and investments in the future power plant portfolio. Moreover, since we have now a liberalized electricity market, Austria cannot be considered isolated and we have to start thinking transnationally.

To take into account the development towards a Pan-European energy market and to simulate different RES-scenarios, the highly resolved European power system investment planning and supply security optimization model *HiREPS* was developed at the EEG. It computes a detailed optimization of the power plant usage (hydropower, thermal power plants, wind and photovoltaics) with hourly resolution. Therefore, high resolution weather data and hydrology data need to be applied.

---

[2]see www.klimafonds.gv.at (06.05.2013)

[3]see www.ffg.at/en/environment-and-energy/ffg-support (10.05.2013), website of the Austrian Research Promotion Agency (FFG)

[4]see www.zamg.ac.at (20.07.2013)

[5]see www.wegcenter.at (20.07.2013)

[6]see www.verbund.com (20.07.2013)

The AutRES100 project actually finished in December 2012, presenting a 90% RES and a 100% RES scenario for the connected Austrian-German power system.[7] However, the project to build up a European model with HiREPS is still ongoing.

Hence, a lot of data research needs to be done to achieve this goal and splitting up the workload to several graduate students is the obvious choice. Each graduate student works on one European country and one specific RES sector to gather data and information and consequently to extend the model at large.

In this extensive endeavour the topic of this thesis is to investigate the hydropower potential of southern and central Sweden, precisely from the river Ljungan to the southern parts with the smaller rivers Mörrumsån, Helgeån, Lagan, Nissan, Ätran and Viskan. Thus, we concentrated on the hydropower simulation section in the HiREPS model. The reason why we attended to these specific Swedish regions is that another diploma candidate is currently working on hydropower in northern Sweden. Clearly, according to the national view, it is meaningful to finally aim at obtaining a model for whole Sweden.

The motivation to choose Sweden in particular can be traced back to the fact that especially Nordic countries are seen to be pioneers concerning innovations in the energy sector. Furthermore, the vast hydropower opportunities of Sweden, referring to the topography and the hydrological conditions, make it quite interesting to simulate this area.

1.2   Main Objectives

We formulate this section to introduce the reader to where this thesis is headed and to give an idea about the main objectives that are addressed in this work. The aims are to:

- determine the existing hydro power portfolio of South and Central Sweden (hydro power plants with a capacity $\geq$ 5MW, dams, reservoirs)

- find possibilities to make the computation of the HiREPS hydro model faster

- investigate scaling issues regarding the model data

- investigate in detail certain aspects of the modelling to reduce model errors

- formulate a detailed mathematical description of the numerical optimization method used to solve the model

- informally state long term and short term simulation results regarding South and Central Sweden's hydropower opportunities

- compare model results with real data for years 2001 - 2011 to confirm model accuracy

---

[7]One can find the final presentation of the project at the website of the EEG. www.eeg.tuwien.ac.at/index.php?option=com_wrapper&view=wrapper&Itemid=86 (14.05.2013)

1.3 Overview Swedish Energy Market

1.3.1 History of Liberalization

The liberalization of the Austrian electricity market started in 1999. The first cornerstones that would lead to the concept of a full liberalization were established in the following few years. The inspiring example for this step was the model of the Scandinavian electricity market, which was already implemented some years before and proved to be a stable, reasonable idea.

In Sweden, the deregulation process already began in early 1992. *Deregulation* or *liberalization* means that the state cuts off its function to govern the power market, and instead open competition is introduced. Moreover, the strict separation of grid-related topics and electricity production respectively trading activities is enforced.[8] This also enabled growing transnational competition as more and more countries liberalized their markets. Norway was the first country to deregulate the market, followed by Sweden, where in 1996 a joint *Norwegian-Swedish power exchange* was established: the birthplace of today's world's largest market for the exchange of electrical energy, named *Nord Pool Spot AS* (NPS)[9]. It is the leading market for trading power in Nordic and Baltic countries, specifically Norway, Denmark, Sweden, Finland, Estonia and Lithuania. It is also present as well in Germany and the United Kingdom and was formed with the purpose of increasing efficiency, supply security and productivity.[10]
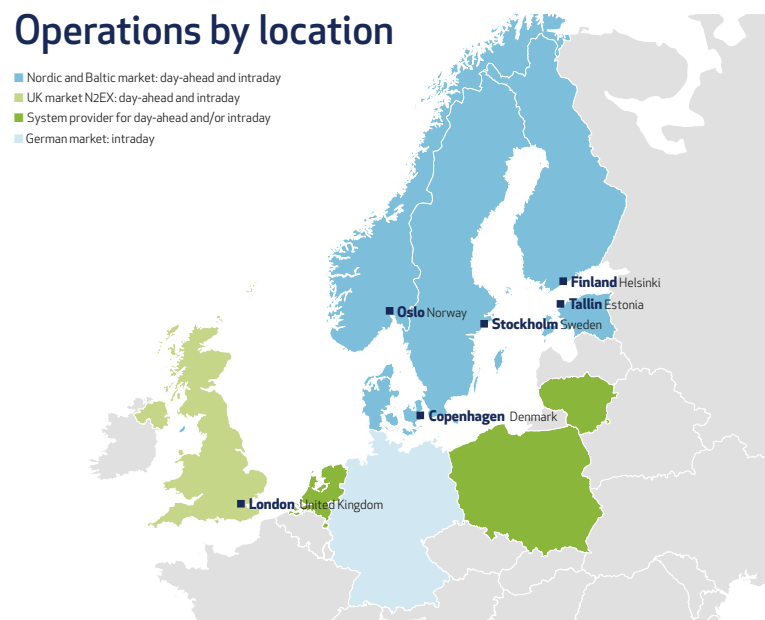
Figure 1.1 Map of operation locations of Nord Pool Spot with market type in each location. (Source: Nord Pool Spot [14])

---

[8]cf. [10, p.29]

[9]see www.nordpoolspot.com (20.07.2013)

[10]cf. www.npspot.com/About-us/History/ (10.05.2013)

Figure 1.1 shows a map with the different operations by location of Nord Pool. The term *day-ahead* refers to contracts made between seller and buyer for the delivery of electricity for the following day and the term *intraday*, as the name already suggests, refers to the short term version of this concept during each day.

For further details on the history and the basic concepts especially of the Swedish energy market see e.g. [19] or [17]. Note that the last reference, a handbook for the Swedish electricity market, is just a translation to English of the original Swedish version [18] and that only the Swedish version is updated frequently.

1.3.2   Renewable Energy Sources - Hydropower

The rising importance of renewable energies in the future energy market is coupled with the commitment of the European Union to limit greenhouse gas emissions and to counteract global warming.

> The Renewable Energy Road Map assesses the share of renewable energy in the energy mix and the progress made in this area. It also includes the target of producing 20% of total EU energy consumption from renewable energy sources by 2020, as well as measures for promoting renewable energy sources in the electricity, biofuels and heating and cooling sectors.[11]

Sweden's goal is to provide at least 50% of its energy use through RES (cf. [20, p.6]). As we can see in Figure 1.2, Sweden is already heading into the right direction to meet its target. In 1990 they were at a 33% share of RES and within 20 years they pushed this percentage to 48%.
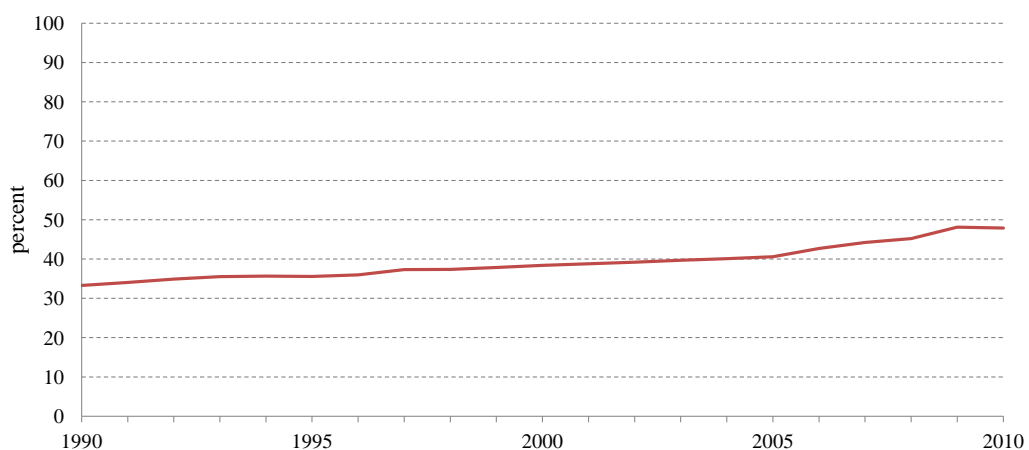
Figure 1.2 Share of renewable energy in Sweden from 1990 till 2010. (Data source: Swedish Energy Agency [21])

---

[11]see www.europa.eu/legislation_summaries/energy/renewable_energy/l27065_en.htm (08.05.2013). For further reading look at "Renewable Energy Road Map. Renewable energies in the 21st century: building a more sustainable future" COM(2006) 848 final.

Considering the total use of energy by energy carrier in 2010 in the whole of Sweden, electricity is the most used and amounts to 130 TWh, followed by oil products (112 TWh) and biofuels, peat and waste (75 TWh). Thus, Sweden needs a reliable and highly fertile electricity production. The two main pillars in this domain are nuclear power with a share of about 44%, referring to the production in total, and hydropower with a share of about 46%.[12] Hence, we see the immense importance of hydropower as a form of emission-free electricity production. Figure 1.3 further emphasizes this eminent position, which of course partly originates from the excellent geographical conditions: "Sweden has more than 100 000 lakes covering roughly 10% of the area. There are many rivers of which thirteen have a mean annual flow of more than 100 m^3/s at the mouth."[13] (cf. [4, p.9]) Thus, we can definitely say that Sweden has good natural resources at hand. The northern and western part is more mountainous and has higher precipitation than the south respectively the south-east, consequently the large-scale hydropower plants with big reservoirs are situated in the northern and western areas.
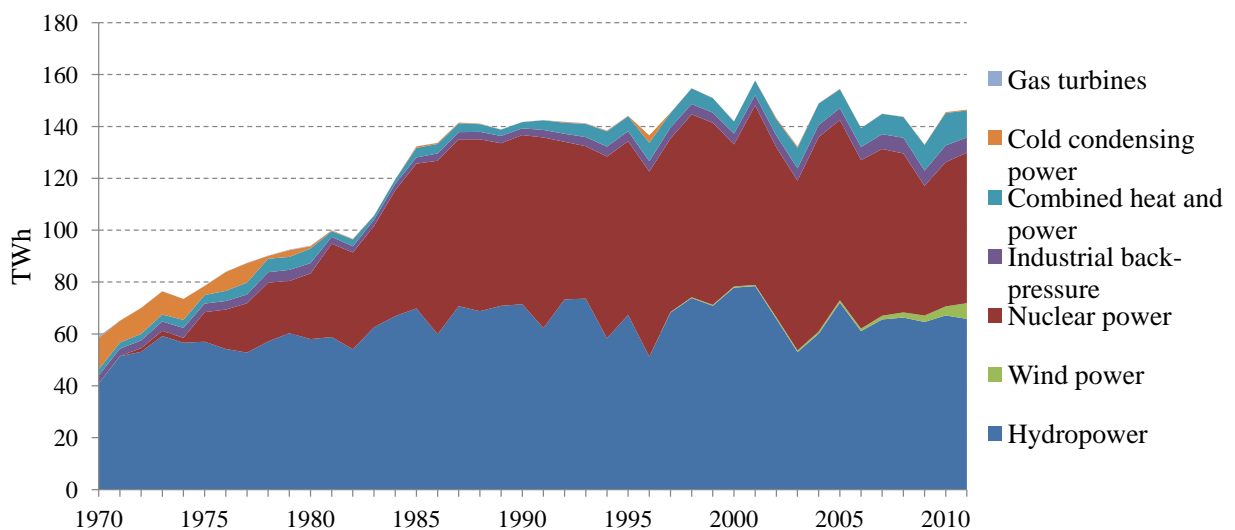
Figure 1.3 Electricity production per type in TWh in Sweden from 1970 till 2011. (Data source: Swedish Energy Agency [21])

The era of hydroelectric power started roughly in the 1880s, when the first plants were built to cover the local energy needs. We can also see in Figure 1.3 that until 1975 hydropower satisfied nearly the whole electricity demand.

Most of the production capacity of today was already built before the 1960s. Today the total number of hydropower plants in Sweden is about 2 000 of which about 1 600 have an installed capacity of maximum 10 MW. In December 2011 the installed hydropower capacity was 16 197 MW, which makes about 44% of the total installed capacity. Figure 1.4 confirms that and also shows that the hydropower capacity did not vary much in the last 16 year. The

---

[12]Computed by the mean value of the production in the years 2000 - 2011. cf. [21]

[13]100 m^3/s is equal to 100 000 l/s, to get a better grasp of the dimensions we are talking about.

same applies to the electricity production capacity according to nuclear power. Moreover, wind power increased substantially in the last years to a share of about 8% in 2011. Notably, between 2009 and 2011 it advanced by 85% compared to the installed capacity in 2009.
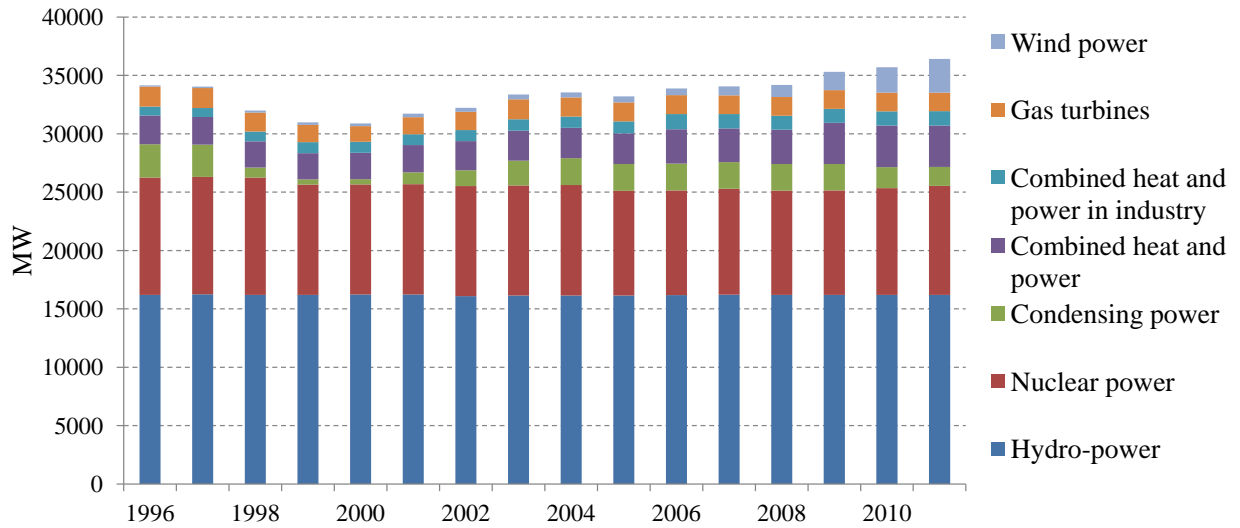
Figure 1.4 Installed electricity production capacity in MW per energy carrier in Sweden from 1996 till 2011. (Data source: Swedish Energy Agency [21])

Chapter 2

Hydropower Plants

2.1 Overview

The basic functional principle of a hydropower plant is to transform the kinetic and potential energy of water into electric energy, i.e. to make use of the hydrological and geographical conditions to produce electricity. In most cases, water is impounded by a dam and then led to a turbine. The gravitational force of the flowing water then drives the turbine, which is coupled to a generator through a mechanical shaft. The available power capacity of an HPL is basically a function of two major variables:

- *Water discharge*
  The volume rate of water flow with respect to time through the plant or a given cross-sectional area. Mostly given in cubic meters per second (m^3/s).[14]

- *Head*
  The difference of elevation when the water drops down while passing through the plant. Thus it means the difference between the *headwater* and the *tailwater* level, where headwater is the upstream water at the impoundment and tailwater is the downstream water coming from the outlet of the HPL.[15]

Other essential features of an HPL regarding the whole layout of the plant are mentioned below and for better imagination also in Figure 2.1, where we state a schematic picture of a hydropower plant with the most important terms specified.

- *Reservoir*
  In a reservoir, the water coming from the catchment area is impounded in order to be used later to produce electricity. The *catchment area* or *drainage basin* of a reservoir describes the area of land, where all the water from precipitation, snow melt or other tributaries that converges to the reservoir is coming from.

---

[14]cf. [23, p.11]
[15]see ibid.

- *Dam* and *intake*

  The dam is a massive artificial retaining wall used to store water in the reservoir. There are two main types - the embankment dam and the concrete dam. Embankment dams are made of different layers of earth and rock. Only their heavy weight suffices to resist the forces of the impounded water and their dense, waterproof core prevents from leakage of water.

  Concrete dams, as the name already indicates, are mostly made out of concrete and are either fortified by buttresses or use a natural curved form to withstand the induced forces. In Figure 2.1, we also see the intake, which is an opening in the weir to lead the water to the turbine. Another important feature of a dam are the spillways, which are not indicated in particular in the figure below. They denote gates in the dam that can be opened to be able to spill water, in case of flooding or unnaturally high tides.

- *Penstock*

  The penstock is a pressure tunnel that carries the water from the reservoir to the power house. There it leads the water to the turbines to drive them. They can be quite short in length referring to run-of-the-river power plants as in Figure 2.1, but also quite big and long, especially regarding big storage power plants in mountainous areas.

- *Power house* with *turbine* and *generator*

  The power house provides protective housing for the turbine, the generator and the control equipment. In Figure 2.1 we see the case of a power house in open air, but there is also the possibility of building it deep into the underground.

- *Headwater*

  The headwater is the upstream water of the considered hydropower plant and therefore the water upstream of the impoundment.

- *Tailwater* and *tail race*

  The tail race is the pathway through which water is transported out of the power plant and again into the river. Thus, the already mentioned tailwater level is the level of the river where it meets the tail race. Moreover, the tailwater is the downstream analog to the term headwater.

- *Power Lines*

  Big hydropower plants contribute in some cases directly to the national electricity grid via long distance power lines. Smaller power stations often only feed into local grids or even just provide energy for one industry site.

This list of power plant features in combination with Figure 2.1 should give a rough overview of how an HPL works, how it looks and what the main functionalities are. Next, we will go into detail about the different possibilities in HPL settings and present a classification.
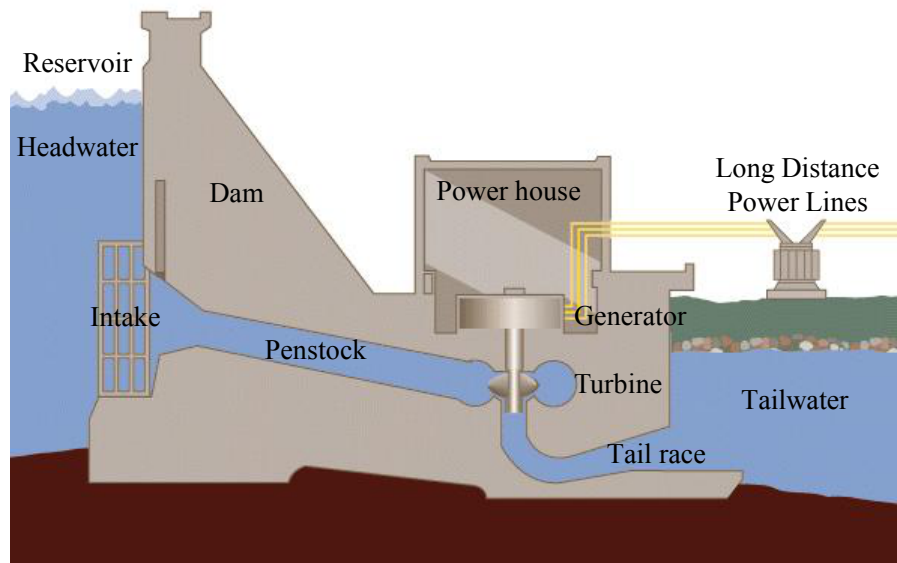
Figure 2.1 Schematic drawing of an HPL showing the main features for one layout. (Source: Tennessee Valley Authority www.tva.gov/power/hydro.htm (14.05.2013) with comments by the author)

2.2   Classification

We can distinguish between different types of power stations. Depending on what aspects we consider, be it technological, economical, capacity-related or head-related aspects, we get diverse classifications[16]:

A) **Technical Aspects**

(i) *Run-of-the-river hydropower plants* (RR-HPL)

These power plants turbinate the water as it comes down the river and have no significant possibility for impoundment, i.e. they immediately use the water supply which is provided by nature to drive the turbines. In most cases, the power house and the weir are situated directly next to each other. However, in some cases RR-HPLs do not even need a dam, like the Tazimina project in Alaska.[17] For clarity, we include a schematic drawing of one possible RR-HPL setting.

| 1 Powerhouse | 3 Generator | 5 Headwater |
| 2 Transformer | 4 Turbine | 6 Tailwater |
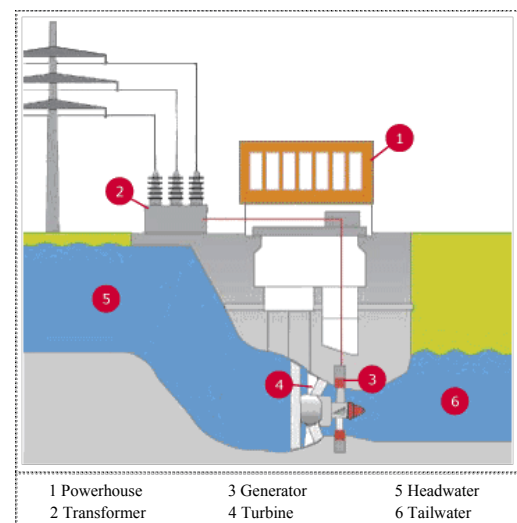
Source: VSE (Verband Schweizer Elektrizitäthersteller)

---

[16]Simplified and shortened version of the classification in [8, p.99ff].

[17]see www.arctic.net/~newhalen/Tazfolder/Tazimina.html (15.05.2013)

(ii) *Storage hydropower plants* (ST-HPL)

ST-HPLs can store vast amounts of water and have a natural source of inflow that fills the reservoir, as for instance rain, snow melt or a tributary river. Topographically, according to their big reservoirs, these HPLs are mostly situated in low and high mountain regions, where they can make use of the immense potential energy due to high elevation differences up to nearly 1 900 m. The Swiss power plant Bieudron holds the record with a head of 1 883m.[18] In Sweden, the highest head is achieved by the power station Stensjöfallet at river Indalsälven with 318 m followed by HPL Tåsan with a head of 269 m.[19]

1 Reservoir	4 Machine Cavern	7 Transformer
2 Dam Wall	5 Pelton-Turbine	8 Water Course
3 Penstock	6 Generator	
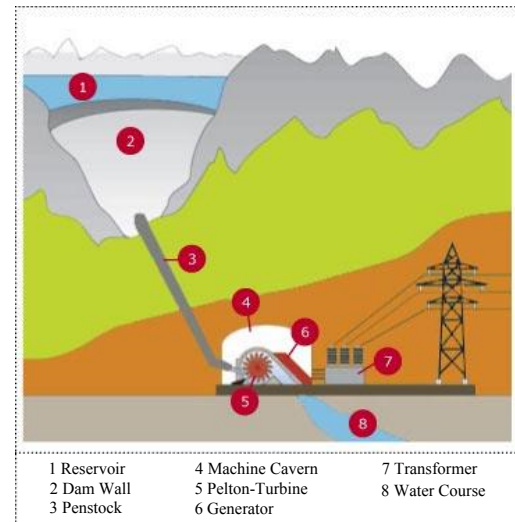
Source: VSE (Verband Schweizer Elektrizitäthersteller)

However, there are also ST-HPLs which are closer to RR-HPLs than ST-HPLs according to their operation plan, because they just minimally alternate the water level with an amplitude of e.g. 0.5 - 1 m and thus have little storage capacity - those are also called *threshold power plants* (TH-HPL).

In general, depending on how long an HPL can actually store the water, we differ between *day-storage reservoirs*, *week-storage reservoirs* and *year-storage reservoirs*. Further on, we will also specify a power station according to its reservoir's storage ability as for instance YST-HPL is short for *year-storage hydropower plant*.

To avoid confusion concerning the schematic picture above, we clarify that typical types of turbines (here Pelton-turbine) will not be addressed until the next section, which formulates a characterization according to head-related aspects.

(iii) *Pumped storage hydropower plants* (PST-HPL)

This type of storage power station has the ability to pump water from a lower reservoir to an upper reservoir. For instance in times of an energy surplus, which results in low electricity prices, water is pumped to the upper storage of the power plant and can be used again later in times of higher demand and higher prizes.

The overall efficiency of a PST-HPL, i.e. the ratio between the produced energy through turbining and the required energy for pumping, varies between 70% and 80%. Of course, also several efficiency losses according to the mechanics of the tur-

---

[18]see www.grande-dixence.ch/energie/wasserkraft/wallis/kraftwerk-bieudron-hohe.html (15.05.2013)

[19]For a list of the power stations with the highest heads see the website of Svenk Energi www.svenskenergi. se/Elfakta/Elproduktion/Vattenkraft1/De-storsta-vattenkraftstationerna/ (20.07.2013).

bines, pumps, generators and transformers are incorporated into these percentages.[20]

Since the middle 1980s, the usage of PST-HPLs in Europe shifted from refinement of energy and providing peak power to mostly balancing the grid frequency. We clarify that the term refinement describes the idea of turning non-renewable electricity, like for example electricity generated by coal-fired power plants or nuclear power, into power from RES.

Also, the opportunities for PST-HPLs to achieve profits became more lim-

| 1 Reservoirs | 3 Generator | 5 Pump |
| 2 Penstock | 4 Transformer | 6 Francis-Turbine |

Source: VSE (Verband Schweizer Elektrizitäthersteller)

ited during the last years. One of the reasons for that is the remarkable increase of photovoltaic plants, which of course have their production peak around midday, where the solar radiation is highest. Consequently, a considerable amount of energy is fed into the grid, leading to a cut off of the midday price peak at the electricity stock market. Hence, the formerly typical operation plans of PST-HPLs (turbining by day when demand and thus prices are high, pumping at night when prices are low) become increasingly restricted. For further reading on PST-HPLs see [8, p.675].
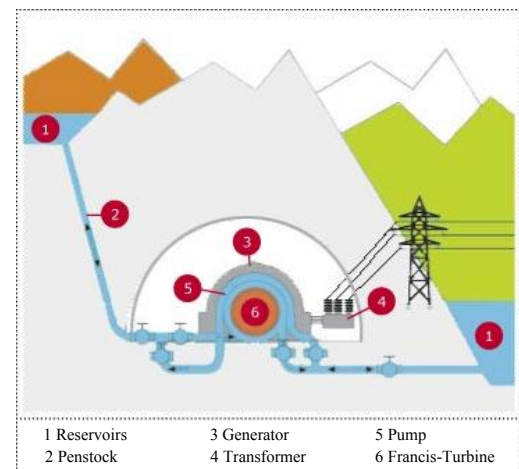
(iv) *Tidal hydropower plants*

The principal idea of a tidal power plant is to exploit the elevation difference between high and low tide. The water flowing back and forth is used to drive the turbines and hence to produce electricity. We just added this type of hydropower plant for the sake of completeness, since this technique of energy production is actually not subject of this thesis. However, for clarity we present a

Source: 2008 Encyclopedia Britannica, Inc.

schematic drawing that vividly describes the functionality of such a facility.

B) **Head-related aspects**

(i) *Low-pressure power plants*

Low-pressure power plants have a head less than 15 m. Hence, they are topographically located in the lowlands. Mostly RR-HPLs and plants with a day-storage reservoir can be found in this category.

---

[20]cf. [8, p.686]

Probably the most used turbine for this type of power station is the Kaplan-turbine. The reason is that they are especially characterized by large discharge rates and low heads up to 80 m, which fits perfectly for RR-HPLs, TH-HPLs and other small hydropower plants. They can be installed vertically as in case of Figure 2.1 or horizontally as we see in the schematic picture to the RR-HPLs above. For better understanding, we display a closer view of a Kaplan-turbine and refer to [8, p.569ff].
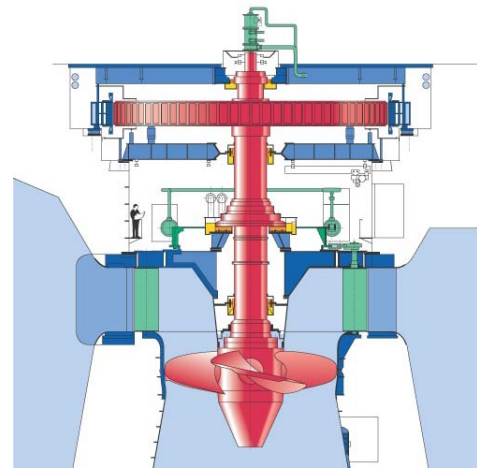
Source: Voith Siemens Hydro Power Generation

(ii) *Medium-pressure power plants*

Medium-pressure power plants have a head between 15 m and 50 m and are generally located in the low mountain range with day- and week-storage reservoirs.

Concerning the turbines, in medium-pressure plants there are typically turbines of either the Kaplan type or the Francis type installed, which is also what we see in the schematic figure in the pumped-storage section above. Thus, also PST-HPLs tend to use this type of turbine. We state a cross-sectional schematic figure to get a better view of the functional principles.

Francis-turbines are characterized by their horizontal water inflow and axial water outflow. The two big circles to the left and to the right show actually the cross-section of a tube around the turbine runner that leads the water to the blades.
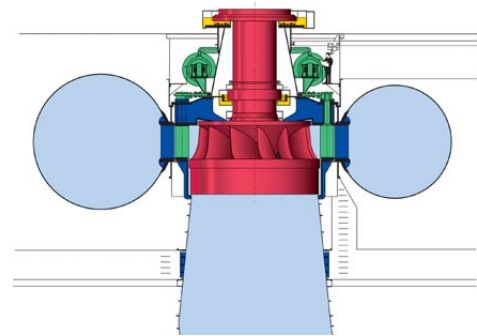
Source: Voith Siemens Hydro Power Generation

Furthermore, those turbines can process heads up to 600 m and capacities of up to 750 MW. In general, depending on the size of the power plant, we can say that the field of application of Francis-turbines overlaps with one of the other two types we mention. For further reading on Francis turbines we refer to [8, p.585ff].

(iii) *High-pressure power plants*

High-pressure power plants have a head greater than 50 m. They tend to be week-storages or year-storages situated in the high mountain range. Since high mountainous areas offer rich opportunities to achieve big heads, there has to be a turbine type that can handle even higher heads than the Francis-turbine. In fact, the Pelton-turbines are specifically designed for that purpose. We show a schematic figure that

represents a vertical angle on a Pelton-turbine.

This turbine works best for low and medium discharge and for huge heads from 550 m to 2 000 m. Characteristic are also the nozzles we see in the picture that direct streams of water against the turbine vanes. That is why a Pelton-turbine is also denominated as a *water impulse turbine*. For further reading we refer again to [8, p.592ff].
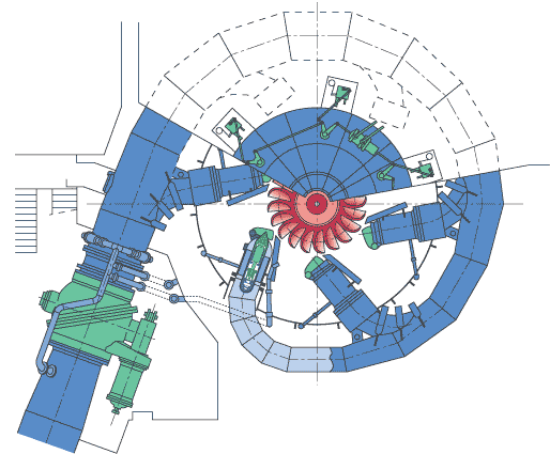
Source: Voith Siemens Hydro Power Generation

C) **Aspects of energy economics**

(i) *Base load power plants*

The energy produced by such an HPL is used to meet the essential and constantly needed portion of the electricity supply. Mostly RR-HPLs, TH-HPLs and smaller ST-HPLs are operated for this purpose.

(ii) *Medium-* and *peak load power plants*

These HPLs are applied to fulfill medium and peak demands. They can be categorized primarily as high-pressure power plants, since those HPLs are able to rapidly start and stop producing electricity compared to fossil fuel and nuclear plants. Thus, they perfectly meet the requirements to trim peaks in demand.

D) **Capacity-related aspects**

(i) *Small hydropower plants* (usually < 1 MW)

(ii) *Medium hydropower plants* (< 100 MW)

(iii) *Large hydropower plants* (> 100 MW)

2.3   Physical Parameters of Power Stations

The classification of different types of power stations in Section 2.2 and the rough overview in Section 2.1 inevitably lead to the question about how energy production via hydropower functions in detail. Hence, this section is dedicated to provide the reader with the physical parameters that play a major role in hydropower usage. Section 2.3 should also build up the physical knowledge basis for later analysis in context of modelling a hydropower system and for interpretation of simulation results of the HiREPS model.

2.3.1   Energy of Water

The total energy that lies within hydropower can be split up into a potential and a kinetic component. *Bernoulli's equation*, which can be derived from the principle of conservation of energy, forms the basis for this observation. It implies that the energy of an incompressible, frictionless fluid adds up from the energy associated with the movement of and the pressure in the fluid, as well as the elevation above a reference plane (cf. the three terms on the left hand side of equation (2.1)). This equation can be found in every standard book on fluid dynamics, see e.g. [5] or [12].

We consider the following physical values at a specific point on a streamline. Bernoulli's principle then looks as follows:

$$\frac{v^2}{2g} + \frac{p}{\rho g} + z = const. \tag{2.1}$$

v	flow velocity	$[\text{m/s}]$
g	gravitational acceleration	$[\text{m/s}^2]$
p	pressure at the specific point	$[\text{Pa}]$
ρ	density of the fluid	$[\text{kg/m}^3]$
z	geodetical height of the point above a reference plane	$[\text{m}]$.

In fluid dynamics, the concept of head relates energy terms to terms of length measurements. We define

$$h_k := \frac{v^2}{2g} \quad \text{and} \quad h_{pressure} := \frac{p}{\rho g}. \tag{2.2}$$

Since meter is actually the unit of both h_k and $h_{pressure}$, we can rewrite equation (2.1) in terms of elevations. We have

$$h_E = h_k + \underbrace{h_{pressure} + z}_{=:h_p} = const. \tag{2.3}$$

h_k	kinetic head or velocity head	$[\text{m}]$
$h_{pressure}$	pressure head	$[\text{m}]$
h_p	potential head or hydraulic head	$[\text{m}]$
h_E	energy head	$[\text{m}]$

The term h_k represents the kinetic energy and the term h_p the potential energy. Thus, Bernoulli's principle suggests that an increase of the flow velocity results in a decrease of the potential energy.

Figure 2.2 should lead to a better understanding of the terms head, potential head, kinetic head and pressure head. The head will be more important in the next section, when we discuss the electric power of an HPL. Usually, we use the term "head" for the *gross head h_{gross},*

which is the difference between the headwater level and the tailwater level. However, the *net head h_{net}* is much more important for real applications, since it already takes in account head losses due to friction. We have

$$h_{net} = h_{gross} - h_{loss_{01}} - h_{loss_{23}}. \tag{2.4}$$

Thus, also energy losses can be described in terms of elevations. In Figure 2.2 the so called *head loss* or *friction head $h_{loss_{ij}}$* is pointed out. There, the double-index in the subscript implies at which local point of the HPL the leakage occurs.
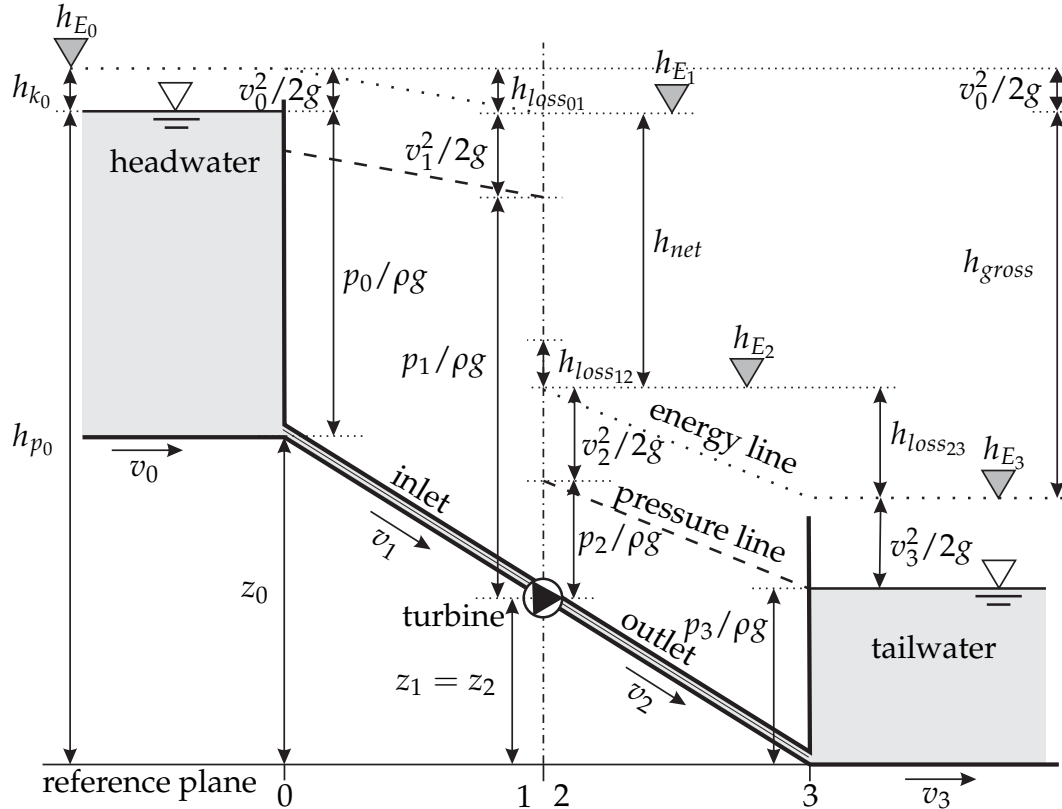
Figure 2.2 Schematic drawing of the headwater and tailwater section with a turbine in between showing the interpretation of the different heads at different points 0 to 3. (Source: [8, p.30] with translations and comments by the author)

We can now write down the different components of the energy content of water: the potential energy and the kinetic energy. They sum up to the total useable energy:

$$E_p = \frac{1}{3.6 \cdot 10^6} \cdot g \cdot m \cdot h_p \tag{2.5}$$

$$E_k = \frac{1}{3.6 \cdot 10^6} \cdot g \cdot m \cdot h_k \tag{2.6}$$

$$E_{ideal} = E_p + E_k = \frac{1}{3.6 \cdot 10^6} \cdot g \cdot m \cdot h_E, \tag{2.7}$$

where the mass equals the density times the volume,

$$m = \rho_w V.$$

m	mass of a portion of water	[kg]
ρ_w	density of water	$[\text{kg}/\text{m}^3]$
V	volume	$[\text{m}^3]$
E_p	potential energy	[kWh]
E_k	kinetic energy	[kWh]
E_{ideal}	ideal total energy	[kWh]

The factor $\frac{1}{3,6\cdot10^6}$ is due to the conversion of the energy unit from watt-seconds [Ws] to kilo-watt-hours [kWh], since this is one of the most common units when referring to electric energy.

The subscript term "ideal" in the abbreviation for the total energy already implies that this is only the energy content achieved by assuming ideal conditions. Influences of friction in pipes, turbulences, etc. were not taken into account. In practical use the *real total energy* will be diminished due to losses. Hence,

$$E_{real} < E_{ideal}. \tag{2.8}$$

2.3.2 Energy Content of a Storage Reservoir

A storage reservoir always has a specified scope for varying the water level:

- *top water level (h_{max})*

- *drawdown level (h_{min})*.

The top water level is the highest and the drawdown level the lowest water level permitted or possible. Both are given in meters above see level. Either environmental regulations force the reservoir to be operated in a certain level-range or mechanical restrictions like the height of the intake or the design of the turbine restrict this so called *regulation amplitude*, which denotes the difference between top water and drawdown level:

$$\text{regulation amplitude} = h_{max} - h_{min}. \tag{2.9}$$

From (2.5) we already know what the potential energy looks like, which is stored within a volume layer of the reservoir with respect to a reference horizon (e.g. the height of the turbine shaft). But this calculation is only justified for the amount of water in one layer. The potential energy also differs, since every layer has a different height with respect to the turbine shaft.

Thus, one interesting question is how to calculate the total energy content of a storage reservoir with *useable volume V_{use}*, whereas useable volume means the water volume between top water level and drawdown level.

In this context we introduce another important term, the *height of the barycenter* of the reservoir h_b. As Figure 2.3 suggests, h_b forms the level line that horizontally cuts the reservoir in

two sections, where each one contains half of the useable volume. Using this quantity we can state the total useable energy content as:[21]

$$EC_{total} = \frac{1}{3.6 \cdot 10^6} \cdot g \cdot \rho_w \cdot V_{use} \cdot h_b. \tag{2.10}$$

Figure 2.3 illustrates the terms mentioned above.
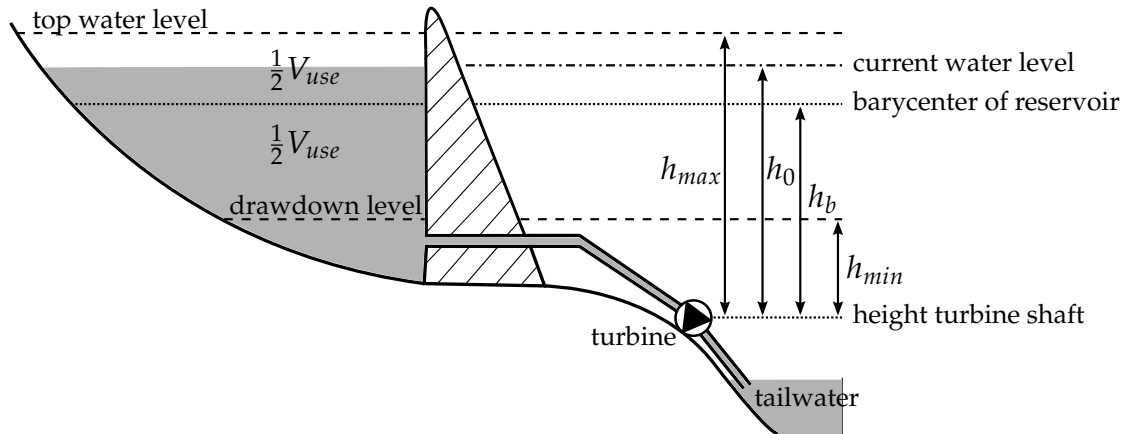
Figure 2.3 Storage reservoir with different water levels. The reference horizon is the height of the turbine shaft.

Moreover, the current water level h_0 implies that the barycenter is not a constant term but dependent on the way the HPL operates. While the turbine is running, the water level sinks and thus the barycenter also varies.

Hence, to compute the current energy content $EC_{current}$ we introduce the current barycenter h_{b_0} and the current useable volume V_{use_0}. We have

$$EC_{current} = \frac{1}{3.6 \cdot 10^6} \cdot g \cdot \rho_w \cdot V_{use_0} \cdot h_{b_0}. \tag{2.11}$$

2.3.3 Power of an HPL

Another very important term in context of HPLs is the *electric power*, which equals the rate at which electric energy is transferred. In general, *power* is defined as work per unit time with SI unit watt $[W]$.

We already mentioned that head and discharge are two of the most important concepts in the usage of hydropower. To confirm this statement we will relate the power extracted by an HPL in terms of discharge and head. We recall that discharge is the rate of volume (of a water flow). Hence,

$$Discharge = \frac{Volume}{Time}. \tag{2.12}$$

---

[21]cf. [8, p.34]

Using (2.12), the theoretical power equation looks as follows:

$$Power = \frac{Work}{Time} = \frac{Force \times Distance}{Time} = \frac{Mass \times Acceleration \times Distance}{Time}$$

$$= \frac{Density \times Volume \times Acceleration \times Distance}{Time}$$

$$= Density \times Discharge \times Acceleration \times Distance.$$

We abbreviate

$$P_{ideal} = \rho_w \cdot Q \cdot g \cdot h, \tag{2.13}$$

where we use just h for the head, since we do not specify whether we take the net head or the gross head.

The standard power equation also takes account of losses during energy production by adding an efficiency factor. Moreover, we change the unit from watt to kilo-watt. We formulate

$$P = \frac{1}{10^3} \cdot \eta_{tot} \cdot \rho_w \cdot Q \cdot g \cdot h. \tag{2.14}$$

P	power	$[\text{kW}]$
Q	discharge	$[\text{m}^3/\text{s}]$
η_{tot}	total efficiency	$[\,\text{-}\,]$

The total efficiency η_{tot} splits up into several components, since turbine and generator lose energy during processing and a plant also needs a certain amount of energy for its own consumption. We have[22]

$$\eta_{tot} = \eta_{turb} \cdot \eta_g \cdot \eta_{self}. \tag{2.15}$$

η_{turb}	turbine efficiency ($\eta_{turb} = 0.88 - 0.92$)	$[\,\text{-}\,]$
η_g	generator efficiency ($\eta_g = 0.96 - 0.98$)	$[\,\text{-}\,]$
η_{self}	self provision ($\eta_{self} = 0.990 - 0.995$)	$[\,\text{-}\,]$

These efficiency factors are applicable for state of the art turbines and generators. For older versions, these numbers can of course be much lower.

Multiplying the efficiencies yields an overall efficiency $\eta_{tot} = 0.836 - 0.897$. Using the mean value of the boundaries of η_{tot} and the facts that water in a river has a density of approximately $1000 \, \text{kg}/\text{m}^3$ and that the mean gravitational acceleration is $g = 9.81 \, \text{m}/\text{s}^2$, we can present a quick approximation to the power output of an HPL, which further underpins the importance of discharge and head:

$$P \approx \frac{1}{10^3} \cdot 0.866 \cdot 1000 \cdot Q \cdot 9.81 \cdot h = 8.495 \cdot Q \cdot h. \tag{2.16}$$

---

[22]Simplified version of equation (2.12) in [8, p.32].

We have now in hand the foundation to understand further statements and argumentations in the upcoming chapters. We will especially rely on this basic knowledge we have gathered here to explain our modelling approach in Chapter 4. Moreover, we will also need this chapter when discussing simulation results regarding detailed operational plans of different types of hydropower plants in Chapter 6.

Next, we widen our basis and provide another chapter that gathers basic information about linear optimization but also specific knowledge about the solving method we use to meet the mathematical claim of this thesis.

Chapter 3

Optimization

This chapter serves to get to know the mathematical fundamentals that build up a linear optimization problem. These basics will help to understand the modelling in Chapter 4 and certain numerical behavior investigated in Chapter 5.

One of the main objectives of this work is also to give a detailed view of the specific mathematical method used to solve our hydropower model, which will be addressed in Section 3.3. Hence, this part at the end of this chapter is dedicated to satisfy that claim.

3.1 Introduction

It is essentially important in engineering and economy as well as in everyone's everyday life to make the "right" decisions, i.e. find optimal solutions for given problems. The process of making choices clearly involves a detailed analysis of the problem and predefinition of the overall objective. Throughout the phase of analyzing restrictions to our sought solution will occur naturally. Thus, an optimal decision should satisfy our aim and at the same time also obey our constraints. The task to fulfill those two goals simultaneously can turn into a highly complex procedure. Hence, mostly it is necessary to formulate such an optimization problem by use of a mathematical model to solve it.

In general, the field of optimization can be roughly divided into two major classes: *static optimization* and *dynamic optimization*. Static means that we try to minimize some given function $f : \mathbb{R}^n \mapsto \mathbb{R}$, whereas the data and the model stay constant in the period of consideration. Contrarily, in dynamic optimization we minimize a functional $J : \mathcal{X} \mapsto \mathbb{R}$, where $\mathcal{X}$ is an appropriate function space. Moreover, the data respectively the model are dynamic, i.e. they change in time.

Static optimization can further be split up into several classes:

- Linear optimization (LO)

- Quadratic optimization (QO)

- Non-linear optimization (nLO)

- Convex optimization (CO)

- Integer-programming (IP)

- Mixed-integer-programming (MIP)

Besides the term optimization, the term *programming* is also used in literature (e.g. *static programming*, *dynamic programming*).

Since we only use linear optimization in our hydro model, we will focus on this topic. But first we will state a general formulation of a static optimization problem. There are three main components that build up the model:

- a real vector of *variables* $x \in \mathbb{R}^n$

- the already mentioned *objective function* $f : \mathbb{R}^n \mapsto \mathbb{R}$

- the *equality constraints* $g_i(x), i = 1, \ldots, p$ and the *inequality constraints* $h_i(x), i = 1, \ldots, q$

The general form then looks as follows:

$$
\begin{aligned}
\min_{x \in \mathbb{R}^n} \quad & f(x) \\
\text{subject to} \quad & g_i(x) = 0, \quad i = 1, \ldots, p \\
& h_i(x) \geq 0, \quad i = 1, \ldots, q,
\end{aligned}
\tag{3.1}
$$

where $q, p \in \mathbb{N}_0$. One can always rearrange formulations to standard form using different tactics. Those will be discussed in further detail in terms of linear programming in the next section.

For our purposes it is sufficient to consider $g_i(x)$ and $h_i(x)$ as affine functions. Hence we have

$$
\min_{x \in \mathbb{R}^n} \quad f(x) \tag{3.2a}
$$

$$
\text{s.t.} \quad Gx - g = 0 \tag{3.2b}
$$

$$
Hx - h \geq 0, \tag{3.2c}
$$

where G and H are matrices with $G \in \mathbb{R}^{p \times n}$, $H \in \mathbb{R}^{q \times n}$ and g and h are vectors with $g \in \mathbb{R}^p$, $h \in \mathbb{R}^d$. We call x *feasible*, if x satisfies the constraints in (3.2) and *strictly feasible*, if it satisfies the equality constraint (3.2b) and $Hx - h > 0$. Furthermore, we define the *feasible set* to be the set of all feasible vectors, i.e.

$$
\mathcal{F} := \{ x \in \mathbb{R}^n : Gx = g, Hx \geq h \}. \tag{3.3}
$$

This leads to the definition of feasibility and infeasibility in the context of a given optimization problem. We denote (3.2) as *feasible*, if there exists a feasible point x, i.e. $\mathcal{F} \neq \{\}$. Vice versa we call (3.2) *infeasible*, if the feasible set is empty, i.e. $\mathcal{F} = \{\}$. We can now define different types of solutions to problem (3.2).

Definition 3.1 (Local and global solutions) *With regards to problem* (3.2) *a vector* $x^* \in \mathcal{F}$ *is called*

(i) *a* local solution *if there exists an* $\varepsilon > 0$ *and a neighborhood* $\mathcal{U}_\varepsilon$ *of* x^* *such that* $f(x^*) \leq f(x)$ *for all* $x \in \mathcal{U}_\varepsilon \cap \mathcal{F}$.

(ii) *a* strict local solution *if there exists an* $\varepsilon > 0$ *and a neighborhood* $\mathcal{U}_\varepsilon$ *of* x^* *such that* $f(x^*) < f(x)$ *for all* $x \in \mathcal{U}_\varepsilon \cap \mathcal{F}$ *with* $x \neq x^*$.

(iii) *a* global solution *if* $f(x^*) \leq f(x)$ *for all* $x \in \mathcal{F}$.

(iv) *a* strict global solution *if* $f(x^*) < f(x)$ *for all* $x \in \mathcal{F}$ *with* $x \neq x^*$.

Next, we can state necessary optimality conditions, also known as the *Karush-Kuhn-Tucker* (KKT) *conditions*, that will drive further computations and lead us to the core problem solved by primal-dual interior-point methods.

Theorem 3.2 (KKT conditions) *Let* x^* *be a local solution to the problem* (3.2) *and let* $f \in C^1$, *then there exist vectors* $\lambda \in \mathbb{R}^p$ *and* $s \in \mathbb{R}^q$ *such that*

$$\nabla f(x^*) - G^T\lambda - H^Ts = 0, \tag{3.4a}$$

$$Gx^* = g, \tag{3.4b}$$

$$Hx^* \geq h, \tag{3.4c}$$

$$s \geq 0, \tag{3.4d}$$

$$s^T(Hx^* - h) = 0. \tag{3.4e}$$

Proof. For a proof see e.g. [13]. There the statement is proved for the general form of static optimization (3.1) with $g_i, h_i \in C^1$. Theorem 3.2 can be easily derived. Another reference for the proof would be [6].

$\square$

The vectors $\lambda \in \mathbb{R}^p$ and $s \in \mathbb{R}^q$ are *Lagrange multipliers* for the constraints $Gx = g$ and $Hx \geq h$ in (3.2). As done in [13], we could also have formulated the KKT conditions above using a *Lagrange function* $\mathcal{L}(x, \lambda, s) = f(x) - \lambda^T(Gx - g) - s^T(Hx - x)$.

In the next section we will go into linear optimization and we will see that the KKT conditions applied to LO are not just necessary for a solution but also sufficient.

3.2   Linear Optimization

As the name already implies, linear optimization deals with optimization problems, where the objective function and all constraints are linear. In literature such optimization problems

are sometimes also referred to as *linear programs* (LP). The *standard form* in this case looks as follows:

$$\min_{x \in \mathbb{R}^n} \quad c^T x$$
$$\text{s.t.} \quad Ax = b \tag{3.5}$$
$$x \geq 0,$$

where $c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, $A \in \mathbb{R}^{m \times n}$ and $m, n \in \mathbb{N}$. We can show that every LP formulation can be rearranged to have the form (3.5).

Assume a given linear maximization problem in *general form*:

$$\max_{x \in \mathbb{R}^n} \quad c_1 x_1 + \ldots + c_n x_n \tag{3.6a}$$
$$\text{s.t.} \quad a_{i1} x_1 + \ldots + a_{in} x_n = b_i, \quad i = 1, \ldots, p_1 \tag{3.6b}$$
$$a_{i1} x_1 + \ldots + a_{in} x_n \leq b_i, \quad i = p_1 + 1, \ldots, p_2 \tag{3.6c}$$
$$a_{i1} x_1 + \ldots + a_{in} x_n \geq b_i, \quad i = p_2 + 1, \ldots, m \tag{3.6d}$$
$$x_j \geq 0, \quad j = 1, \ldots, q_1 \tag{3.6e}$$
$$x_j \text{ free}, \quad j = q_1 + 1, \ldots, n. \tag{3.6f}$$

Firstly, a maximization problem can easily be rewritten as a minimization problem using that

$$\max_{x \in \mathbb{R}^n} f(x) = \min_{x \in \mathbb{R}^n} -f(x). \tag{3.7}$$

Secondly, inequality constraints as in (3.6b) and (3.6c) can be transformed to equality constraints by introducing so called *slack variables s*. We define s through the following relation:

$$a_{i1} x_1 + \ldots + a_{in} x_n + s_i = b_i, \quad i = p_1 + 1, \ldots, p_2,$$
$$a_{i1} x_1 + \ldots + a_{in} x_n - s_i = b_i, \quad i = p_2 + 1, \ldots, m. \tag{3.8}$$

Thirdly, if a variable x_j is free as in (3.6f), i.e. there is no restriction on the sign, then we can replace it by $x_j^+ - x_j^-$, with $x_j^+, x_j^- \geq 0$ and again arrive at the standard form.

3.2.1 Duality and Optimality

Every linear optimization problem has a second LP related to it, called the *dual problem* (DP). (3.5) is referred to as the *primal problem* (PP). The dual to (3.5) can be stated as:

$$\max_{x \in \mathbb{R}^n} \quad b^T \lambda$$
$$\text{s.t.} \quad A^T \lambda + s = c \tag{3.9}$$
$$s \geq 0,$$

where $\lambda \in \mathbb{R}^m$ is the vector of *dual variables* and $s \in \mathbb{R}^n$ is the vector of *dual slack variables*. The two constraints in (3.9) can be written more concisely as one constraint

$$A^T \lambda \leq c. \tag{3.10}$$

Further on, however, we will mostly work with formulation (3.9).

The in-depth relation between the primal and dual problem will become clearer when we look at KKT conditions adapted to (3.5). Furthermore, we can prove that the KKT conditions are not only necessary but also sufficient to find a solution to (3.5).

Theorem 3.3 (Optimality conditions) *The vector $x \in \mathbb{R}^n$ is a solution to the PP* (3.5) *if and only if there exist vectors $\lambda \in \mathbb{R}^m$ and $s \in \mathbb{R}^n$ such that*

$$A^T \lambda + s = c, \tag{3.11a}$$

$$Ax = b, \tag{3.11b}$$

$$x_i s_i = 0, \quad i = 1, \ldots, n \tag{3.11c}$$

$$x, s \geq 0. \tag{3.11d}$$

Proof. We assume that x is a solution to (3.5). Applying Theorem 3.2 to (3.5) we can clearly see that inserting $f(x) := c^T x$, $G := A$, $H := I$ (I is the identity matrix), $g := b$ and $h := 0$ into (3.4) leads to the equations (3.11a), (3.11b) and (3.11d). From (3.4e) we arrive at $s^T x = 0$. Since $s, x \geq 0$, we can also verify (3.11c).

Next, we assume that we have a triple (x, λ, s) that satisfies the conditions above and an arbitrary primal feasible vector $\tilde{x}$. We write

$$c^T \tilde{x} \overset{(3.11a)}{=} (A^T \lambda + s)^T \tilde{x} \overset{(3.11b)}{=} b^T \lambda + s^T \tilde{x} \overset{(3.11d)}{\geq} b^T \lambda = c^T x, \tag{3.12}$$

where the last equation follows from multiplying (3.11a) from the left side with x^T and applying (3.11c). Thus, the value of the objective function of an arbitrary primal feasible vector $\tilde{x}$ is always bigger than $c^T x$. This concludes the proof.

$\square$

It is not just a coincidence that we have used the same names for the Lagrange multipliers in Theorem 3.3 and the dual variables $\lambda \in \mathbb{R}^m$ respectively the dual slack variables $s \in \mathbb{R}^n$. They are actually equal and thus we have uncovered two different ways of looking at the vectors λ and s. We also realize that the KKT conditions state an important connection between primal and dual variables, since the conditions (3.11a), (3.11b) and (3.11d) represent *dual* and *primal feasibility*. The equalities in (3.11c) are also called the *complementarity condition*, since from $x_i > 0$ follows immediately $s_i = 0$ and vice versa.

The next theorem again relates primal and dual variables and is one of the main statements in duality theory. One could say that it takes the results of Theorem 3.3 and puts them into a different perspective.

Theorem 3.4 (Duality theorem)

  (i) *If $x \in \mathbb{R}^n$ is primal feasible and $\lambda \in \mathbb{R}^m$ is dual feasible, we have*

$$b^T \lambda \leq c^T x. \tag{3.13}$$

    Hence, if either (3.5) *or* (3.9) *has an unbounded objective, then the other is infeasible.*

  (ii) *If either* (3.5) *or* (3.9) *has a solution with finite optimal objective value, then so does the other, and in this case the objective values coincide.*

Proof. The fact that the primal objective is an upper bound for the dual objective can be proved immediately by combining (3.11b) and (3.10). For (ii), see e.g. [13, Theorem 13.1]. $\qquad\square$

We state another important theorem, that informs us about the existence of *primal-dual solutions* (a triple (x, λ, s) that solves the PP (3.5) and the DP (3.9)) and the boundedness of the set of solutions.

Theorem 3.5 *We assume that $x \in \mathbb{R}^n$ is primal feasible and $\lambda \in \mathbb{R}^m$ is dual feasible. Then the following two statements are true:*

  (i) *The set of primal-dual solutions is nonempty. That is, there exists a primal-dual solution.*

  (ii) *Provided that $\lambda \in \mathbb{R}^m$ is strictly dual feasible, the set of all primal solutions is bounded. Conversely, if the PP is strictly feasible, the set*

$$\{s \mid (\lambda, s) \text{ solves the DP (3.9) for some } \lambda \in \mathbb{R}^m\} \tag{3.14}$$

    is bounded.

Proof. For (i) see [24, Theorem 2.1 i.] and for (ii) see [24, Theorem 2.3]. $\qquad\square$

Next, we introduce the concept of strict complementarity, since we will need it later in order to understand the implemented model of the MOSEK solver (see [11]) which we will investigate in Section 3.3.1.

Definition 3.6 (Strict complementarity) *We define a primal-dual solution, i.e. a triple (x, λ, s) that solves the PP* (3.5) *and the DP* (3.9), *as a* strictly complementary *solution if*

$$x_i + s_i > 0, \quad i = 1, \ldots, n. \tag{3.15}$$

In this context it is of course interesting to know, whether there even exist solutions with this property. Goldman and Tucker [9] formulated the following theorem that provides an answer to this question.

Theorem 3.7 (Goldman-Tucker) *If the PP* (3.5) *and the DD* (3.9) *are feasible, then there exists a primal solution $x \in \mathbb{R}^n$ that solves* (3.5) *and a dual solution $(\lambda, s) \in \mathbb{R}^m \times \mathbb{R}^n$ that solves* (3.9) *such that (x, λ, s) is strictly complementary.*

Proof. For a full proof see [15, Theorem II.3], where the result is derived from Theorem I.12, or [24, Theorem 2.4]. Of course, we can also directly refer to Goldman and Tucker [9].

$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad$ $\square$

Since we have now conditions for optimality at hand, we can proceed by introducing a class of methods that uses Theorem 3.3 as a basis for its computations.

3.3 Primal-Dual Interior-Point Methods

Primal-dual interior-point methods apply variants of Newton's method[23] to the equality conditions in Theorem 3.3, where they modify search directions and step lengths to satisfy the non-negativity bounds $x, s \geq 0$ strictly in each iteration. This is exactly where the term *interior-point* originates from. One should notice that this system is in fact nonlinear due to the equations $x_i s_i = 0$, $i = 1, \ldots, n$. We can rewrite the whole system concisely as:

$$F(x, \lambda, s) := \begin{pmatrix} A^T \lambda + s - c \\ Ax - b \\ XSe \end{pmatrix} = 0, \tag{3.16a}$$

$$x, s \geq 0, \tag{3.16b}$$

where $e = (1, \ldots, 1)^T$ is the all-ones vector and X and S are diagonal matrices with

$$X := \operatorname{diag}(x_1, \ldots, x_n) \quad \text{and} \quad S := \operatorname{diag}(s_1, \ldots, s_n). \tag{3.17}$$

For reasons of clarity we state the basic algorithm in pseudo-code below. Most primal-dual interior-point methods rely on this structure and establish further specializations upon this basis. As already mentioned, a modified Newton's method is used in order to solve (3.16a) followed by a line search along the computed Newton direction to determine the maximal step-size without violating the positivity bound $x, s > 0$.

The regular Newton system for F with (x, λ, s) *strictly feasible* looks as follows

$$\underbrace{\begin{pmatrix} 0 & A^T & I \\ A & 0 & 0 \\ S & 0 & X \end{pmatrix}}_{=:J_F(x,\lambda,s)} \begin{pmatrix} \Delta x \\ \Delta \lambda \\ \Delta s \end{pmatrix} = \underbrace{\begin{pmatrix} 0 \\ 0 \\ -XSe \end{pmatrix}}_{=-F(x,\lambda,s)}, \tag{3.18}$$

where J_F is the Jacobian matrix to the mapping F.

Algorithm 3.8 (Interior-point method - a framework)

   Given $(x^0, \lambda^0, s^0) \ldots$ *strictly feasible starting point*

$\qquad\quad$ $\varepsilon_{tol} \ldots$ *tolerance parameter*

---

[23]For details on Newton's method see e.g. [16].

$k = 0 \dots$ *iteration counter*

while *termination criterion* $> \varepsilon_{tol}$ **do**

$$\textbf{solve} \quad J_F(x^k, \lambda^k, s^k) \begin{pmatrix} \Delta x^k \\ \Delta \lambda^k \\ \Delta s^k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ -X^k S^k e + \sigma_k \mu_k e \end{pmatrix},$$

where $\sigma_k \in [0,1]$ *and* $\mu_k := \left(\sum_i x_i^k s_i^k \right) / n$.

let $(x^{k+1}, \lambda^{k+1}, s^{k+1}) \leftarrow (x^k, \lambda^k, s^k) + \alpha_k \left(\Delta x^k, \Delta \lambda^k, \Delta s^k \right),$

where α_k *is chosen such that* $(x^{k+1}, s^{k+1}) > 0$.

$k \leftarrow k + 1,$

end while

As we can see, the right-hand side changed. Introducing the mean value μ_k, the so-called *duality measure*, and the additional control parameter σ_k we force the Newton step to head for a point where all the products $x_i^k s_i^k$, $i = 1, \dots, n$ take on the same value $\sigma_k \mu_k$.

Clearly, using $\sigma_k = 0$ would result in a standard Newton step, which heads directly towards a point, where the KKT conditions are satisfied. However, this could conflict with our desire to keep clear of the boundary $x_i^k s_i^k = 0$.

On the other hand, applying $\sigma_k = 1$ leads to a so-called *centering direction*, i.e. we determine to step towards an interior point where $x_i^k s_i^k = \mu_k$. (We have that $\mu_k > 0$ since our starting point (x^0, λ^0, s^0) is strictly feasible.) These centering directions benefit to stay clear of the boundary, but often do not yield much progress in reducing the duality measure μ_k and thus do not help to find a primal-dual solution.

To compromise over reducing μ_k and staying clear from $x_i^k s_i^k = 0$, σ_k usually takes on intermediate values, i.e. $\sigma_k \in (0,1)$. Note that in practical algorithms choosing σ_k adaptively is also a widely used concept. Therefore, so-called *predictor-corrector steps* are introduced. We actually calculate two different directions: the predictor and the corrector. The predictor step is computed by use of $\sigma_k = 0$ to determine, which value of σ_k is most suitable. If the resulting direction yields a big reduction of the duality measure μ_k without violating the positivity boundaries, σ_k is chosen close to 0. Otherwise, we set σ_k close to 1. Then, the corrector step solves our Newton system by applying the newly selected σ_k.

Concerning the determination of the step-size α_k, we mention that each method based on Algorithm 3.8 chooses its step-sizes differently. Clearly, we want to make as much progress towards the solution as possible. Hence, in most cases α_k is defined by another maximization problem that takes the fact into account that the boundary should not be violated.

To sum up, we can state that as we move on in the iteration, μ_k decreases (depending on the chosen value of σ_k) and thus the iterates (x^k, λ^k, s^k) converge to a primal-dual solution until the given termination criterion is sufficiently small to stop the algorithm. Hence, with interior-point methods we can never get exact solutions but only approximations.

3.3.1 MOSEK Interior-Point Method

MOSEK[24] is a commercial, high-performance optimization tool used to solve e.g. large-scale LO, QO and MIP. This section is dedicated to the special interior-point method implemented in MOSEK.

What makes the MOSEK solver special is the way infeasibilities are handled. Most interior-point methods for solving the PP (3.5) assume that the problem has an optimal solution. But what if the given LP is infeasible? A problem can be *primal infeasible* and/or *dual infeasible*, i.e. the dual objective and/or the primal objective is unbounded (see Theorem 3.4 (i)). Thus, the optimization algorithm has to deal with each one of these situations.

From Theorem 3.5 we know, if we find a strictly feasible starting point (x^0, λ^0, s^0) for our algorithm, the existence and boundedness of a primal-dual solution is assured. In practice, however, searching for a strictly feasible starting point can be costly.

A remedy to this problem was first proposed by Goldman and Tucker [9], [22]. They studied the so-called *homogeneous self-dual model* (HSD):

$$\min_{x \in \mathbb{R}^n} \quad 0 \tag{3.19a}$$

$$\text{s.t.} \quad Ax - b\tau = 0 \tag{3.19b}$$

$$A^T \lambda + s - c\tau = 0 \tag{3.19c}$$

$$b^T \lambda - c^T x - \kappa = 0 \tag{3.19d}$$

$$x, s, \tau, \kappa \geq 0, \tag{3.19e}$$

where τ and κ are scalar variables. τ is a *homogenizing parameter* and κ is the *slack variable* corresponding to the constraint $b^T \lambda - c^T x \geq 0$. The first two constraints in combination with (3.19e) obviously describe primal and dual feasibility. Most important, constraint (3.19d) contains the so-called *duality gap* $b^T \lambda - c^T x$. Equation (3.19d) actually forces the model to reach a primal-dual solution, since this is the exact opposite of inequality (3.13) in the duality theorem. So the model has to approach an optimal point, where $b^T \lambda - c^T x = 0$.

The term *homogeneous* is derived from the fact that all the right-hand sides of the constraints are zero. The model is called *self-dual*, since the DP to (3.19) is the same as the PP (see [24, p.179f] for further details). Note that the objective function of the HSD is uniformly zero, hence every feasible vector x is also a solution of the problem. Furthermore, (3.19) always has $(x, \lambda, s, \tau, \kappa) = (0, 0, 0, 0, 0)$ as a solution, not a significant one though.

The key to detecting infeasibilities lies in finding strictly complementary solutions (Definition 3.6). Recalling the complementary conditions (3.11c) and applying Theorem 3.7 yields that any solution $(x^*, \lambda^*, s^*, \tau^*, \kappa^*)$ to (3.19) satisfies

$$
\begin{aligned}
x_i^* s_i^* = 0, \quad & x_i^* + s_i^* > 0, \quad \forall i = 1, \dots, n \\
\tau^* \kappa^* = 0, \quad & \tau^* + \kappa^* > 0.
\end{aligned}
\tag{3.20}
$$

---

[24]see www.mosek.com (05.05.2013)

The next theorem again by Goldman and Tucker will prove that indeed a strictly complementary solution provides all the information needed to decide between feasibility and infeasibility. In the feasible case it also shows how to retrieve a solution to (3.5) from a solution of (3.19).

Theorem 3.9 *Let* $(x^*, \lambda^*, s^*, \tau^*, \kappa^*)$ *satisfy* (3.20) *then the following two statements are true:*

(i) $\tau^* > 0$ *and* $\kappa^* = 0$ *if and only if* (3.5) *and* (3.9) *have an optimal solution. In this case* $\left(\frac{x^*}{\tau^*}, \frac{\lambda^*}{\tau^*}, \frac{s^*}{\tau^*}\right)$ *yields an optimal primal-dual solution.*

(ii) $\kappa^* > 0$ *if and only if* (3.5) *is primal or dual infeasible. In this case if* $-b^T y^* < 0$ *then* (3.5) *is primal infeasible, if* $c^T x^* < 0$ *then* (3.5) *is dual infeasible.*

Proof. See [9] or [24, Theorem 9.2, Theorem 9.3]. □

The algorithm MOSEK uses, is in fact based on our framework Algorithm 3.8. However, it does not solve (3.19) directly. Ye, Todd and Mizuno [26] suggested to solve (3.19) by using another more complicated model that has the same features, but adds some advantages considering the numerical behavior and the actual implementation of the algorithm. We will not state this advanced HSD form, since this does not contribute to the topic of this thesis. Instead we refer to [26] and [24, p.183ff]. Xu, Hung and Ye [25] also developed a related model, which can be interpreted as a simplification of the Ye-Todd-Mizuno algorithm.

The MOSEK algorithm, discussed in explicit detail in [3] and [1], is quite similar to the one presented in [25], although it differs in the choice of certain algorithmic parameters, stopping criteria and initial values. A pseudo-code of it can be found in [1].

Termination Criteria

Since interior-point algorithms do not solve LPs exactly, we have to implement termination criteria as already mentioned in Algorithm 3.8. Therefore, the following measures are introduced (for simplicity we denote the Euclidean norm by $\|.\|$):

$$\rho_P^k := \frac{\left\|r_P^k\right\|}{\max\left(1, \left\|r_P^0\right\|\right)}, \quad \rho_D^k := \frac{\left\|r_D^k\right\|}{\max\left(1, \left\|r_D^0\right\|\right)}, \quad \rho_O^k := \left|c^T \frac{x^k}{\tau^k} - b^T \frac{\lambda^k}{\tau^k}\right|$$

$$\text{and} \quad \mu^k := \frac{(x^k)^T s^k + \tau^k \kappa^k}{(\tau^k)^2}, \tag{3.21}$$

where the residuals r_P^k and r_D^k with $k \in \mathbb{N}_0$ are defined as

$$r_P^k := \left\|b\tau^k - Ax^k\right\| \quad \text{and} \quad r_D^k := \left\|c\tau^k - A^T\lambda^k - s^k\right\|. \tag{3.22}$$

We see that ρ_O^k measures the duality gap, i.e. the gap between the primal and dual objective value and we already have seen the duality measure μ^k in Algorithm 3.8, although this form

is a little bit different from the one before.

With these definitions in hand, the termination criteria of the MOSEK algorithm in the *feasible case* look as follows:

$$\rho_P^k \leq \varepsilon_P, \quad \rho_D^k \leq \varepsilon_D, \quad \text{and} \quad \min\left(\mu^k, \rho_O^k\right) \leq \varepsilon_G \, \max\left(1, \left|\frac{c^T x^k}{\tau^k}\right|\right), \quad (3.23)$$

where $\varepsilon_P, \varepsilon_D, \varepsilon_G > 0$ are tolerance parameters chosen by the user with default value 10^{-8}. The termination criteria for the infeasible case are based on our observations in Theorem 3.9 (ii). We omit these criteria and refer to either [3] or the MOSEK manual [11].

3.4   How an Optimizer works

This section provides a brief summary of how a commercial solver like MOSEK works to give the reader a rough idea of the internal proceedings that take place. Especially in Chapter 5 we will sometimes refer to this section, since a detailed analysis of the computation time of a solver certainly involves checking these procedures.

Commonly, when an optimizer is called, it performs several steps on its way to the solution of a proposed problem. It does not not just apply the solution method of choice to the given LP but also certain presolve activities are carried out. We will need a basic understanding of those functions in Chapter 5, where we will go into detail about scaling problems and issues concerning redundancy in internal presolving processes of our modelling language AMPL.[25]

The following list shows the major tasks that are executed by the optimizer and gives a short explanation to each one: :[26]

(i) **Presolver**

The presolver is doing some preprocessing mainly to reduce the size of the problem. One often cannot formulate the LP optimally, what includes for instance implementing as few variables and constraints as possible and avoiding redundancy in the code. However, doing all that could probably make the code unreadable. Hence, due to reasons of clarity and comprehensibility, it is worthwhile to express a model in a convenient way.

Furthermore, implementing code commonly involves trial and error. Most likely, when trying to set up an LP, we happen to formulate constraints that oppose each other or we load data that violates certain set boundaries. Thus, it is an obvious idea to implement a procedure that uses simple rules to check for all these bad and redundant formulations and either rearranges them or gives a warning in case of violated constraints.

---

[25]In spite of AMPL not being an optimizer but just the mathematical programming language to formulate a LP, it also has a built-in presolver. For details on AMPL see Section 4.3.

[26]cf. MOSEK manual [11]

After solving the LP, the original form of the problem implemented by the user is restored. We list the major tasks performed by the presolve phase to give an overview:

- remove redundant constraints
- eliminate variables that are fixed to a value
- check for and remove linear dependencies
- substitute free variables
- check for violated constraints

For a detailed analysis on the presolve phase we refer to [2].

(ii) **Dualizer**
Choosing whether to solve the primal or the dual form of the problem, determined by internal heuristics that measure if the dual or the primal formulation can be solved more efficiently.

(iii) **Scaling**
Scales the input parameters of the problem to get better numerical behavior. If the occurring coefficients in the problem are either very large or very small, significant digits could be truncated during the computations and thus an inaccurate solution would be the result. Thus, it is beneficial to have data that has roughly the same overall magnitude. Normally the optimizer has built in heuristics that check for bad scaling and reformulate the problem data if the LP is not well-scaled.

(iv) **Solving**
Solves the optimization problem using a selected method, e.g. an interior-point method or a simplex method.

Chapter 4

Modelling

This part deals with the aspects of modelling a hydropower system and introduces the reader to the *HiREPS - High Resolution Power System model* that was already mentioned in the introductory chapter.

4.1   HiREPS - High Resolution Power System Model

The ultimate goal of the project behind this optimization program is to get a highly resolved dynamical modelling of the power system in all of Europe with a high share of renewables. *High resolution* means that we get hourly values for every simulation day in our output data and thus also need hourly resolved input data, involving e.g. electricity prices and hydrological data like the water inflow in m^3/s into a reservoir.

In this work, only one part of HiREPS model will be addressed in detail. We concentrate on the section that is simulating hydropower systems. In general, the model also takes wind power, solar power, biomass and other non-renewable electricity production (coal, combined-cycle gas turbine plants, ...) and storage techniques into account.

Thus, the big model describes the complexity of electricity markets to a large extent. Under certain assumptions (future demand, future investment costs for different types of plants, future prices for fossil fuels and CO_2 certificates, etc.) one can simulate prospective RES-share scenarios and their economical feasibility.[27] Naturally, this extended optimization model needs a lot more data, such as hourly resolved wind velocities or data on solar radiation.

4.1.1   Input - Output

In this section an input-output diagram will be stated to give a rough overview of the functionalities of the hydro part of the HiREPS model (see Figure 4.1). We list the outcome of the

---

[27]As already mentioned in the introduction, details on simulations regarding the Austrian and German market applying 90% and 100% share of RES can be found on the website of the EEG, www.eeg.tuwien.ac.at/index. php?option=com_wrapper&view=wrapper&Itemid=86 (14.05.2013)

model and the most important data that has to be provided to drive the implementation. In between we find the black box optimization program. We will fill this box with life in the upcoming section, where we consider the most important equations and inequalities that build up the optimization model.
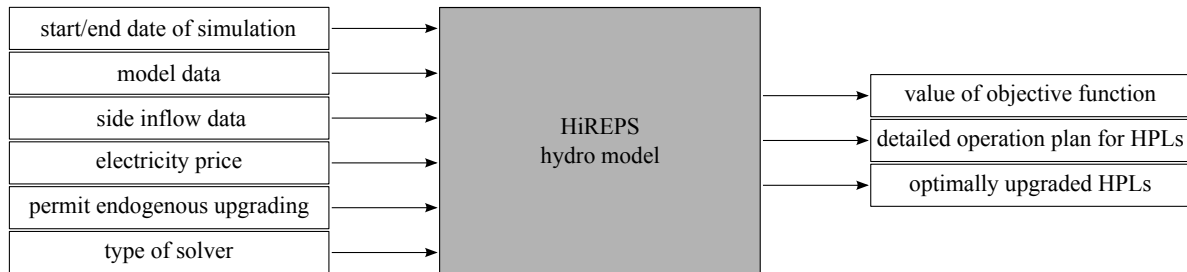
Figure 4.1 Input-output diagram of the HiREPS hydro model.

Input:

- *Start/end date*
  We can start and end the simulation wherever we want it to. The starting and ending dates are given in the format *DD.MM.YYYY_HH.* where the hours *H* for one day are given from 0 to 23.

- *Model data*
  Model data involves e.g. the names of the dams and HPLs and the data related to both, like top water level, drawdown level, design power of the turbines/pumps, maximal admissible discharge, which the turbines/pumps can handle, year of initial operation of the plant, useable volume in the reservoirs, height of the barycenter, gross head, etc. All of this is given in simple text files as a data matrix, which is then passed to the model.[28]

- *Side inflow data*
  This is the hydrological data needed to determine how much water is fed into the main river coming from smaller tributaries and/or ground water between two or more consecutive dams. Commonly, this is given in mean values per day. To break it down to hourly resolution we linearly interpolate in between. Hence, we simply calculate

$$S_{\text{hourly}}(t,\delta) = S_{\text{daily}}(day,\delta) + \left(S_{\text{daily}}(\text{next}(day),\delta) - S_{\text{daily}}(day,\delta)\right) \cdot \frac{t}{24} \tag{4.1}$$

$$\text{with} \quad \delta \in DAMS, \quad t \in \{0,\ldots,23\}.$$

$DAMS$	set of all implemented dams	[-]
$S_{\text{hourly}}(t,\delta)$	hourly side inflow	$\left[\text{m}^3/\text{s}\right]$
$S_{\text{daily}}(day,\delta)$	mean side inflow per day	$\left[\text{m}^3/\text{s}\right]$

The dependence of δ in $S_{\text{hourly}}(.,\delta)$ and $S_{\text{daily}}(.,\delta)$ signifies that this is the inflow going

---

[28]For an example of how data is specified see Listing 4.2 in Section 4.3.

to dam/reservoir δ.

Analogously to the model data this is also passed to the model as a data matrix in a text file, where the rows indicate the date respectively the hour and the columns the flow value.[29]

- *Electricity price*
  The electricity price in Sweden is formed by the laws of supply and demand at the Nordic stock exchange Nord Pool Spot. We have hourly prices for the years 2000 - 2010 given in €/MWh.

- *Permit endogenous upgrading*
  A Boolean parameter that tells the model if upgrading of turbines/pumps is possible. Thus, the model itself has built in investment possibilities and decides endogenously to expand existing plants.

- *Type of solver*
  The type of solver that is used to solve the resulting LP. In our case, in Section 3.3.1 we pointed out the MOSEK solver, but there are also others like CPLEX[30], XPRESS[31] or GUROBI[32].

Output:

- *Value of objective function*
  Since we apply linear optimization to our hydropower system (see next section) we have an objective function that is minimized or maximized as defined in (3.5), Section 3.2.

- *Operation plan*
  The model yields a detailed operational plan for every dam and every plant that is implemented. This includes e.g. the hourly discharge from one dam to the next one below, i.e. the amount of water per unit time that leaves one plant and enters the reservoir of the next one below after some delay due to the run time, operation curves of turbines, pumps and reservoirs that state the power output respectively the stowage height during the whole simulation period.

- *Upgrade of HPLs*
  Based on endogenous decisions the power of certain HPLs is upgraded and these extensions are seamlessly included in the whole hydropower system.

---

[29]see footnote 28

[30]see www-01.ibm.com/software/commerce/optimization/cplex-optimizer/index.html (28.05.2013)

[31]see www.fico.com/en/Products/DMTools/xpress-overview/Pages/Xpress-Optimizer.aspx (28.05.2013)

[32]see www.gurobi.com (28.05.2013)

4.1.2   Model Equations

Every dam and every plant is considered a node in the model. The relationship between different nodes has to be formulated mathematically through equations and inequalities. Moreover, certain regulations to the functionality of the HPLs and their machinery and to other node-related data will also lead to constraints in the resulting LO problem. Further on, we will use the term dam as a synonym to reservoir or impoundment. Thus, reservoir related data, such as the useable water volume will be a feature of the dam.

For convenience, we restate the general form of a constraint optimization problem:

$$\min_{x \in \mathbb{R}^n} \quad f(x)$$
$$\text{subject to} \quad g_i(x) = 0, \quad i = 1, \dots, p$$
$$h_i(x) \geq 0, \quad i = 1, \dots, q,$$

where in our case $f(x), g_i(x), i = 1, \dots, p$ and $h_i(x), i = 1, \dots, q$ are all linear.

Objective Function

The most important part of the LP is the objective function f. There are various possibilities to formulate it, depending on what exactly we want to be optimized in the hydropower system:
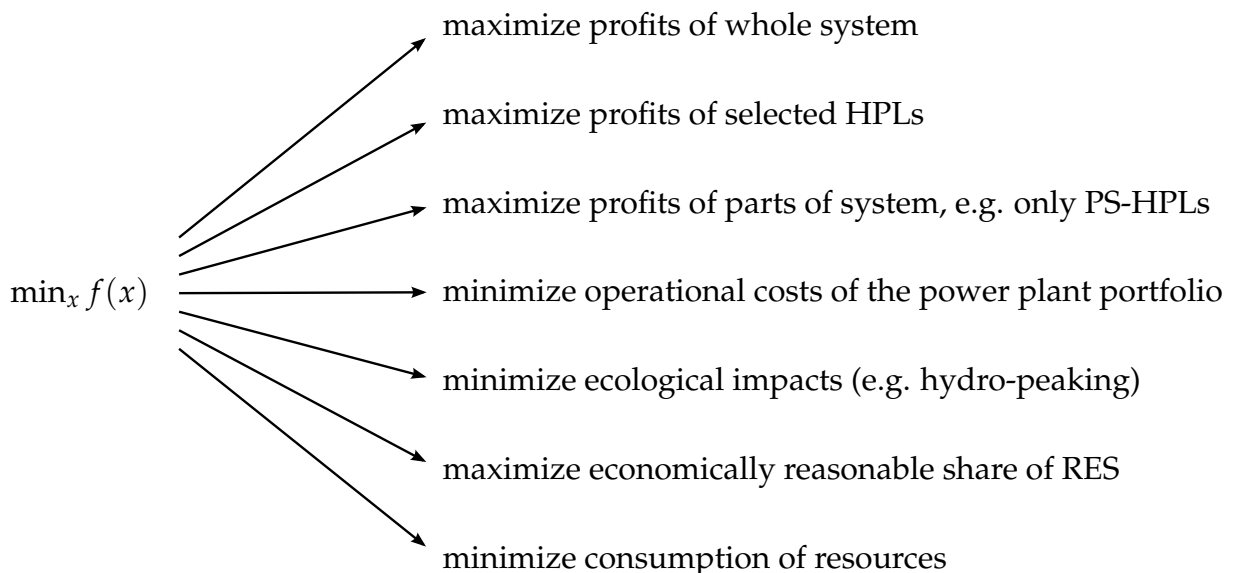
Figure 4.2 Some different possibilities to formulate the objective function $f(x)$, depending on what aspects we want to optimize.

In Figure 4.2 we mentioned the term *hydro-peaking*. This describes the alternation in water flow in a river due to impoundment and turbining processes caused by HPLs. In order to stem the resulting ecological effects one can apply optimization to e.g. mitigate the peak flow but still remain in an economically valuable region of operation.

In our LP we choose to maximize the profits of a predefined hydropower system, which is

one of the main questions in economy. Mostly we try to optimize the whole system at once, which yields the view of one central operator deciding how to run all plants with an optimal plan. In reality, every plant operator is trying to optimally run their own HPLs according to the current electricity price. Thus, it is also an eligible issue to optimize just parts of the system. Notably, if there is a lot of storage volume in the reservoirs of the HPLs, the model has a lot more degrees of freedom concerning the mode of operation.

Next, we look at our objective function in detail:

$$
\begin{aligned}
\max\,&\text{Profits}\\
&= \max \sum_{t \in HOURS} \text{NetEnergy}(t) \cdot \text{NPSPrice}(t)\\
&= \max \sum_{t \in HOURS} \left(E^{el}_{generated}(t) - E^{el}_{used}(t) - E^{el}_{lost}(t) \right) \cdot \text{NPSPrice}(t) \qquad (4.2)\\
&= \max \sum_{t \in HOURS} \left(\left(\sum_{p \in HPL} E^{el}_{turb}(t,p) - E^{el}_{pump}(t,p) \right) - E^{el}_{lost}(t) \right) \cdot \text{NPSPrice}(t).
\end{aligned}
$$

HOURS	set of all hours of the simulation time	[-]
HPLS	set of all HPLs in the considered system	[-]
NPSPrice	electricity price at Nord Pool Spot	[€/MWh]
E^{el}_{turb}	electric energy generated by turbines	[MWh]
E^{el}_{pump}	electric energy used by pumps	[MWh]
E^{el}_{lost}	energy loss by overflow	[MWh]

We clarify that $E^{el}_{turb}(t,p)$ is the total electric energy generated in hour t by all turbines installed in HPL p. This should not to be confused with the potential energy of water that we talked about in Section 2.3.1 and Section 2.3.2. We also note that by NPS price we mean the electricity price per hour at the day-ahead market. Hence, we do not consider the options of short-term *intraday trading*, long-term *electricity futures*[33] or the possibilities to trade at the *regulatory market*.[34]

By energy loss due to overflow we indicate the case that the incoming water cannot be handled any longer by the turbines, since they are dimensioned with a specific maximum discharge capacity. Furthermore, turbines are also designed for a specific head. If the head drops below this threshold the turbine has to be taken out of operation.

In the instance of flooding, one may think that this leads to perfect conditions for HPLs to generate electricity. In fact, it is just the opposite. The surplus of water that cannot be handled by the turbines is released through the spillways of the dam. Thus, the enormous

---

[33] An electricity future is a contract between two parties to buy or sell a certain quantity of electric energy for a priorly fixed price and a specified future delivery and payment date. E.g. companies in industrial production who want to cover their base load can be players in this market.

[34] Aside the electricity stock exchange there is also a market for regulatory energy, where energy producers can provide negative or positive operating reserve for a specific price. This is to stabilize the frequency of the electricity grid in case of any occurring disruption.

amount of water compensates the elevation difference between the headwater and the tail-water section, which reduces the head to a minimum. Consequently, the turbines have to be cut off. The resulting energy loss is described by the term E_{loss}^{el}.

In general, a turbine or a pump with the power of 1 MW can generate or drain 1 MWh of energy in 1 hour. Since the time in our model has already hourly resolution, i.e. $t \in HOURS$, the current power level of a turbine at time t is equal to the produced electricity. Hence, we can further split up E_{turb}^{el} and E_{pump}^{el}:

$$E_{turb}^{el}(t, p) = P_{turb}(t, p) := \sum_{j \in TURBS(p)} P_j(t) \tag{4.3}$$

and

$$E_{pump}^{el}(t, p) = P_{pump}(t, p) := \sum_{j \in PUMPS(p)} P_j(t). \tag{4.4}$$

$TURBS(p)$	set of all turbines of HPL p	[-]
$PUMPS(p)$	set of all pumps of HPL p	[-]
$P_{turb}(t)$	power of all turbines of HPL p at time t	[MW]
$P_{pump}(t)$	power of all pumps of HPL p at time t	[MW]
$P_j(t)$	power of the j-th turbine/pump at time t	[MW]

Constraints

Next, we take a closer look at the main constraints in the HiREPS hydro model. As we have already seen in Chapter 3, constraints can be equations or inequalities which form the core of the whole optimization by limiting the variables to stay in a certain region. We start with constraints regarding the discharge varible:

Discharge

One of the main variable in the LP is the discharge $Q(t, p)$, i.e. the water flow concerning HPL p due to turbining or pumping. Further on, if we want to specify whether it is the discharge of a turbine or a pump we are talking about, we will resort to the terms $Q_{turb}(t, p)$ and $Q_{pump}(t, p)$. When no index is given either it is clearly recognizable from the context if we mean turbines or pumps or we refer to both at the same time.

Furthermore, by $Q_j(t, p), j \in TURBS(p)$ we denote the average discharge through turbine j in hour t going to the reservoir of the next HPL along the river and by $Q_j(t, p), j \in PUMPS(p)$ the average discharge through pump j per unit time that is pumped into an upper reservoir. Thus, $Q_{turb}(t, p)$ and $Q_{pump}(t, p)$ describe the subsumption of the discharges of all turbines/pumps installed in HPL p, i.e.

$$Q_{turb}(t, p) = \sum_{j \in TURBS(p)} Q_j(t, p) \quad \text{and} \quad Q_{pump}(t, p) = \sum_{j \in PUMPS(p)} Q_j(t, p), \tag{4.5}$$

where $p \in HPLS$. Clearly, the discharge is a non-negative variable and bounded above by technical regulations, since every turbine and every pump has a limit to how much water can be processed. These basic considerations lead to the following inequalities:

$$0 \le Q_j(t, p) \le Q_j^{max}, \quad j \in TURBS(p) \cup PUMPS(p), \quad t \in HOURS, \qquad (4.6)$$

where Q_j^{max} is the maximal discharge that turbine/pump j of HPL p can handle.

Power of Turbines and Pumps

We reconsider the power equation (2.14) in Section 2.3.3 as well as the definition of $P_{turb}(t, p)$ and $P_{pump}(t, p)$ in (4.3) and (4.4). For the installed power in HPL p there holds

$$P_{turb}(t, p) = c_{MW} \cdot \eta \cdot \rho_w \cdot g \cdot Q_{turb}(t, p) \cdot h(t, \delta_p), \quad p \in HPLS, \quad t \in HOURS. \qquad (4.7)$$

where δ_p is the dam belonging to HPL p and $h(t, \delta_p)$ is the variable that represents the head at time t. Note that the constant factor $c_{MW} = \frac{1}{10^6}$ is due to the unit change to MW. Moreover, for the density of water we use $\rho_w = 1000 \, \text{kg}/\text{m}^3$ and for the gravitational acceleration $g = 9,81 \, \text{m}/\text{s}^2$. η denotes the average efficiency of all turbines in HPL p. Specific values for turbine efficiencies are often hard to obtain, thus if data is missing we use the mean value $\eta = 0.866$ as presented at the end of Section 2.3.3 (see equation (2.16)).

Concerning pumps one can say that (4.7) equally holds with the difference that the efficiency factor η changes to $\frac{1}{\eta}$, since a pump consumes energy instead of generating it. Hence, we have

$$P_{pump}(t, p) = c_{MW} \cdot \frac{1}{\eta} \cdot \rho_w \cdot g \cdot Q_{pump}(t, p) \cdot h(t, \delta_p), \quad p \in HPLS, \quad t \in HOURS. \qquad (4.8)$$

Note that (4.7) and (4.8) actually lead to non-linear constraints due to the fact that $Q(t, p)$ and $h(t, \delta_p)$ are both variables of the LP. Consequently, both equations have to be linearized. We will go into further detail about this topic in the upcoming Section 4.2 about successive linear optimization.

Naturally, also the power has to be limited to a certain range due to the design of the machinery. We have

$$0 \le P_j(t, p) \le P_j^{max}, \quad j \in TURBS(p) \cup PUMPS(p), \quad t \in HOURS, \qquad (4.9)$$

where P_j^{max} is the maximal power output of turbine/pump j of HPL p according to the collected data.

Energy Balance

To know how much water/energy lies within each part of the system in each time step, is one of the crucial things in modelling hydropower. For every dam we have to have a

balancing equation that tells the system about the amount of water that has been passed on by e.g. turbining, pumping or just overflow and the portion of water that has been gained in time step t, e.g. through pumping from the dam below or natural inflow. There are two different ways of doing this. Either trough balancing the energy content of each dam or the water volume which would, physically seen, be the more descriptive way. Nonetheless, in the HiREPS model the concept of energy content is implemented, which relates the volume of water to its value as potential energy.

The energy content of a dam $EC(t, \delta), \delta \in DAMS$ is also a variable in our LP since it is changing over time. We have already discussed how to calculate it in Section 2.3.2 (see equation (2.11)). Before we build up the balance equation, we have to bound EC:

$$0 \leq EC(t, \delta) \leq EC_\delta^{max}, \quad \delta \in DAMS, \quad t \in HOURS, \tag{4.10}$$

where EC_δ^{max} is the maximum energy content possible in the reservoir. Thus, EC_δ^{max} is the total potential energy of the water volume between the top water level and the drawdown level and can be computed using equation (2.10). The balance now takes the following form:

$$
\begin{aligned}
EC(\text{next}(t), \delta) = \; & EC(t, \delta) + E_{sideinflow}^{pot}(t, \delta) \\
& + \sum_{p \in HPLS_{above(\delta)}} E_{turb}^{pot}(t, p) - E_{pump}^{pot}(t, p) + E_{overflow}^{pot}(t, p) \\
& + \sum_{p \in HPLS_{below(\delta)}} E_{pump}^{pot}(t, p) - E_{turb}^{pot}(t, p) - E_{overflow}^{pot}(t, p).
\end{aligned}
\tag{4.11}
$$

$HPLS_{above(\delta)}$	set of all HPLs directly above dam δ	[-]
$HPLS_{below(\delta)}$	set of all HPLs directly below dam δ	[-]
E_{turb}^{pot}	potential energy of water from turbining	[MWh]
E_{pump}^{pot}	potential energy of water from pumping	[MWh]
$E_{overflow}^{pot}$	potential energy of overflowing water	[MWh]
$E_{sideinflow}^{pot}$	potential energy of water from side inflows	[MWh]

Note that $HPLS_{below(\delta)}$ mostly consists of the one power plant belonging to the dam δ. Reconsidering equation (2.7) that describes the total energy of water, we can state

$$E_{turb}^{pot}(t, p) = c_{MWh} \cdot g \cdot \rho_w \cdot V_{Q_{turb}(t,p)} \cdot h(t, \delta), \quad t \in HOURS, \tag{4.12}$$

where $c_{MWh} = \frac{1}{3.6 \cdot 10^9}$ is the constant factor needed for the unit change to MWh and $V_{Q_{turb}(t,p)}$ denotes the volume of the water leaving the turbines during one hour at discharge $Q_{turb}(t, p)$. That is,

$$V_{Q_{turb}(t,p)} = \Delta t \cdot Q_{turb}(t, p), \tag{4.13}$$

where $\Delta t = 1 \, \text{hour} = 3600 \, s$. Inserting (4.13) into (4.12) we can simplify

$$E_{turb}^{pot}(t, p) = \frac{1}{10^6} \cdot g \cdot \rho_w \cdot Q_{turb}(t, p) \cdot h(t, \delta), \quad t \in HOURS, \tag{4.14}$$

Analogously, we get E^{pot}_{pump}, $E^{pot}_{overflow}$ and $E^{pot}_{sideinflow}$.

To facilitate comprehension of the energy balance we did not take into account the running time of water in the river or any special modelling of the topology, e.g. communicating storage reservoirs[35] or regulated connecting passages between two dams without any HPL in between. Figure 4.3 should also ease the understanding of (4.11). It shows the different contributors to the energy content of dam δ.
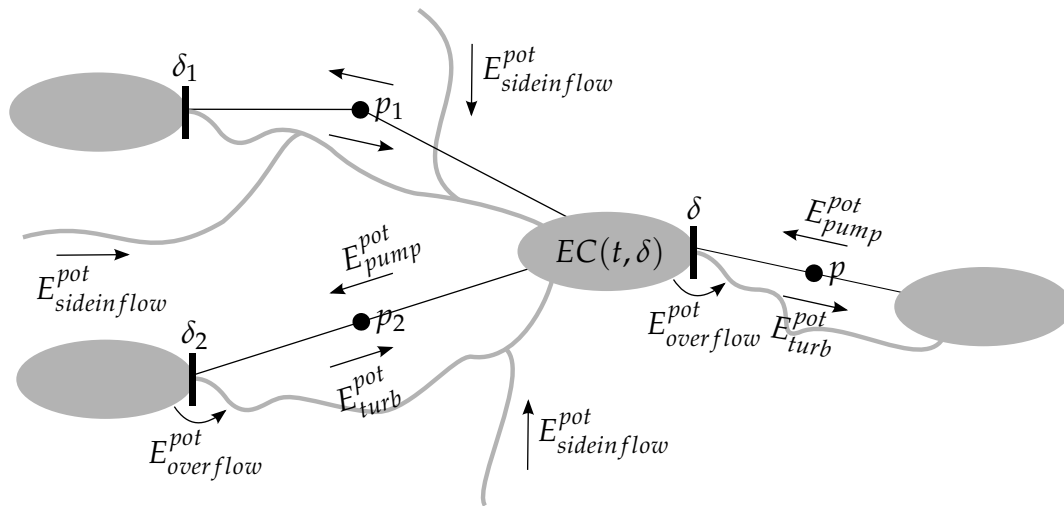
Figure 4.3 Figurative explanation of the energy balance of dam δ to see where the different terms in (4.11) originate from. Here, $p_1, p_2 \in HPLS_{above(\delta)}$ and $p \in HPLS_{below(\delta)}$.

With this section we provided an overview of the most important model equations and inequalities that contribute to build up the model - to summarize we stated the objective function (4.2) and the constraints (4.6) - (4.11).

Now we can go on to analyze different detailed aspects of the modelling as for instance the already mentioned problem concerning the non-linear terms in equations (4.7) and (4.8). This will be the topic of the next section.

4.2 Successive Linear Optimization

Successive linear optimization (SLO) is a very important concept in view of prevention of errors made by the HiREPS hydro model. We will closely investigate this concept in Chapter 5, which will lead to tremendous enhancements in the computation time of our model. Hence, this section introduces the reader to the idea behind SLO.

Successive linear optimization is an algorithmic design to approximately solve non-linear optimization problems. Linearization of the occurring non-linear terms yields an efficiently

---

[35]In some cases there exist connected storage reservoirs which work by use of the concept of communicating vessels, but here we leave those out of consideration.

solvable LP, but also inflicts a certain error. By solving a sequence of closely related LPs we hope to reduce the error and get an adequate solution for the original problem. During this process, the optimal values obtained in the foregone iteration help to solve the current LP. Applying SLO to the HiREPS model takes care of the already mentioned problematic, non-linear term $Q(t, p) \cdot h(t, \delta_p)$ in (4.7) in (4.8) as well as in (4.14). The non-linearity is due to the fact that both the discharge and the current head are variables in the LP. We solve this problem by simple means - using two-dimensional, first-order Taylor approximation. Assumed we have the function $f : \mathbb{R}^2 \mapsto \mathbb{R}$ defined by $f(Q, h) := Qh$. We linearize at some known point (Q_{n-1}, h_{n-1}). To get the value of the function at some unknown point (Q_n, h_n) we compute

$$
\begin{aligned}
Q_n h_n = f(Q_n, h_n) &\approx f(Q_{n-1}, h_{n-1}) + \partial_1 f(Q_{n-1}, h_{n-1})(Q_n - Q_{n-1}) \\
&\quad + \partial_2 f(Q_{n-1}, h_{n-1})(h_n - h_{n-1}) \\
&= Q_{n-1} h_{n-1} + h_{n-1}(Q_n - Q_{n-1}) + Q_{n-1}(h_n - h_{n-1}) \\
&= Q_n h_{n-1} + Q_{n-1} h_n - Q_{n-1} h_{n-1},
\end{aligned}
\tag{4.15}
$$

where $\partial_1 f$ and $\partial_2 f$ are the partial derivatives in the first and the second argument, respectively.

In context of our hydro-model we introduce $Q_n(t, p)$ and $h_n(t, \delta_p)$, where the index n stands for the n-th iteration. $Q_{n-1}(t, p)$ and $h_{n-1}(t, \delta_p)$ can be interpreted as the computed discharges and heads of the previously solved LP in iteration $n - 1$. Combining the linear approximation (4.15) and the power equation for turbines (4.7) we arrive at the following:

$$
P_{turb}(t, p) = \frac{\eta \cdot \rho_w \cdot g}{10^6} \cdot \left(Q_n(t, p) h_{n-1}(t, \delta_p) + Q_{n-1}(t, p) h_n(t, \delta_p) - Q_{n-1}(t, p) h_{n-1}(t, \delta_p) \right).
\tag{4.16}
$$

Note that the analogue holds for pumps when applying (4.15) to (4.8). In view of (4.16) and the preliminary considerations we give a short pseudo-code that illustrates the procedure using SLO.

By Q_n we denote the discharge variable and by h_n the head in the n-th iteration. Notably, in the first iteration, i.e. for $n = 1$, we have to set reasonable initial values Q_0 and h_0 to get the method started.

Algorithm 4.1 (SLO)

  1: ***Given*** h_0, Q_0 ... *initial data for linearization*

  2:           *iter_max* $\in \mathbb{N}$ ... *maximal number of iterations*

  3: ***let*** $q^* \leftarrow Q_0$, $h^* \leftarrow h_0$

  4: ***load*** *linearized LP, data files*

  5: ***for*** $n = 1, \ldots, iter\_max$ ***do***

  6:     ***if*** $n > 1$ & *iter_max* $\neq 1$ ***then***

  7:        ***let*** $q^* \leftarrow Q_n$, $h^* \leftarrow h_n$

     8:   **update** *linearized LP*

     9:   **end if**

  10:   **solve** $LP(h^*, q^*)$

11: **end for**

Here, the term $LP(h^*, q^*)$ indicates that the given linearized LP depends on the parameters h^* and q^*. The **update**-command in line 8 implies that the LP model has changed due to the assignments $h^* \leftarrow h_n$ and $q^* \leftarrow Q_n$, thus it has to be updated in every iteration before solving it.

4.2.1   Linearization Error

As mentioned above, in implementing SLO we pursue to minimize the error inflicted by (4.16). To substantiate this error term, we will now define concretely what we mean by linearization error:

Here, we will denote the right hand side of equation (4.16) as $P_{turb}^{lin}(t, p)$ to indicate that this is the linearized power equation. In (4.16) $Q_n(t, p)$ and $h_n(t, \delta_p)$ are to be computed by the optimization method we use. After the LP is solved we know the solution values to $Q_n(t, p)$ and $h_n(t, \delta_p)$. Hence, we can actually evaluate the original non-linear form of (4.16), which we denote by $P_{turb}^{sol}(t, p)$:

$$P_{turb}^{sol}(t, p) = \frac{\eta \cdot \rho_w \cdot g}{10^6} \cdot Q_n(t, p) \cdot h_n(t, \delta_p). \tag{4.17}$$

The absolute MW error we make concerning the turbines of HPL p in a certain iteration due to our linearization then reads as follows:

$$e_{turb}(p) := \sum_{t \in HOURS} \left| P_{turb}^{lin}(t, p) - P_{turb}^{sol}(t, p) \right|. \tag{4.18}$$

Note that this is the error throughout the whole simulation period, since we sum over all simulated hours. The pumping error $e_{pump}(p)$ takes a similar form if we insert $P_{pump}^{lin}(t)$ and $P_{pump}^{sol}(t)$ accordingly. Thus, we can also state the total error by all turbines and pumps included in the system:

$$e_{MW} := \sum_{p \in HPLS} e_{turb}(p) + e_{pump}(p). \tag{4.19}$$

We will analyze e_{MW} in Section 5.3, where we examine how to obtain a faster decrease in error per iteration by different choices concerning the starting value Q_0.

4.3   Modelling Language - AMPL

Before we go into detailed analysis of numerical aspects of our algorithm in Chapter 5, we want to give a short introduction to the modelling language we use. This is mostly for informative reasons and for the sake of completeness. Moreover, we want to give the reader

a chance to understand how the model data and side inflow data as mentioned in Section 4.1.1 is structured.

AMPL, short for *A Mathematical Programming Language,* is a modelling language to formulate and analyze optimization problems. One can use descriptive notation in building up a model that is easy to read and comprehend. Furthermore, AMPL manages the interaction with a specified solver and enables the user to display the optimal solution values. We give a simple example code of a production model that illustrates some basic functions of AMPL:

```
1  set PRODS;                           # products
2  param rate {PRODS} > 0;              # tons produced per hour
3  param profit {PRODS};                # profit per ton
4  param sell_limit {PRODS} >= 0;       # limit on tons sold in week
5  param avail >= 0;                    # hours available in week
6  var produced {p in PRODS} >= 0, <= sell_limit[p];   # tons produced
7  # Objective: total profits from all products
8  maximize Total_Profit: sum {p in PRODS} profit[p] * produced[p];
9  # Constraint: total production time may not exceed hours available
10 subject to Time: sum {p in PRODS} (1/rate[p]) * produced[p] <= avail;
```

Listing 4.1 Example code of a production model. This is the *.MOD-file, where sets, parameters and variables are defined as well as the objective function and the constraints. (PRODUCTION.MOD)

```
1  # define set of products
2  set PROD := prod1 prod2;
3  # parameters depending on products given in a matrix
4  param:          rate      profit    sell_limit :=
5        prod1     200       25        6000
6        prod2     140       30        4000  ;
7  # paramter avail defined
8  param avail := 40;
```

Listing 4.2 The data for the model is presented in a *.DAT-file. The sets and parameters are filled with values using a quite illustrative matrix form (i.e. we have that the rate of prod1 = 200 or the profit per ton of prod2 = 30 etc.). (PRODUCTION.DAT)

```
1  model production.mod;   # loads the model file
2  data  production.dat;   # loads the data file
3  option solver mosek;    # option that specifies the solver
4  solve;                  # solves the LP
```

Listing 4.3 *.RUN-file for the whole model, where the model and the corresponding data is loaded. As a option the solver can be specified and the optimization is performed by use of the solve-command in line 10. (PRODUCTION.RUN)

The HiREPS model is arranged the same way consisting of *.MOD-, *.RUN- and *.DAT-files. The *.MOD-file can be considered as the core of the modelling, since there the LO fomulation

is implemented with all constraints and objectives. The *.DAT-file is just a mere pool of data which is used to initialise the parameters defined in the *.MOD-file. The *.RUN-file is simply a script that puts everything together to solve the LP - loading the model, loading the data, iterate the solve-command and update the parameters for the SLO as shown in Algorithm 4.1. For further information and details on AMPL see [7].

Chapter 5

Numerical Analysis of Model and Simulation

This chapter is dedicated to our conducted numerical experiments concerning the computation time of the successive linear optimization as well as the effects we obtain by several improvements we made in the code to achieve better performance.

To be more precise, in the subsequent section we will introduce a certain tolerance parameter that helps to speed up the computation time to some extent and to eliminate near zero elements in the constraint matrix A of our hydropower model. It also assures that the simulation stands clear from scaling issues, which is essential to get good numerical behavior. A big leap in reducing computation time was also possible due to reordering the SLO's update phase (lines 6 to 9 in Algorithm 4.1), which will be discussed in Section 5.2.

Furthermore, we will go into detail about different choices of initial values for the discharge Q_0 (see line 3 in Algorithm 4.1) and their effects on the linearization errors we make and the computation time of the SLO algorithm. We will present two different approaches to the problem of choosing starting values which will lead to faster error convergence per iteration.

5.1 Near zero Elements in Constraint Matrix

5.1.1 Scaling

In Section 3.4 (iii) we have already emphasized that scaling is of significant importance in numerical computations. Hence, there are already built-in heuristics in the MOSEK solver to improve a badly scaled problem. However, those heuristics may not always work, thus it can be beneficial to experiment with different scaling of the LP on your own.

During simulation we experienced a problem regarding near zero elements in the matrix A that specifies the constraints of the LP. In that case the MOSEK solver issues a warning (*MOSEK Warning 710*), whereby one can adjust the tolerance for throwing the warning through the MOSEK option *MSK_DPAR_DATA_TOL_AIJ*, which holds as default value

10^{-12}. Due to the fact that the warnings did mostly appear in the iterations after the first one, we realized that the main problem lies within the SLO. Since MOSEK just gives a warning, the LP itself compiles and is solved nonetheless, however it indicates either redundancy or bad scaling.

To investigate the question where and why these small matrix elements occur, we have to understand how the optimizer rearranges the model equations by substituting variables. We therefore take a close look at our objective function in (4.2). We will now prove that in fact the right hand side of the profit equation only depends on the variables that characterize the discharge, the energy content and the energy loss by overflow. All the other variables can be substituted by terms including these three. For convenience we restate the profit formula:

$$\text{Profits} = \sum_{t \in HOURS} \left(\left(\sum_{p \in HPL} E_{turb}^{el}(t,p) - E_{pump}^{el}(t,p) \right) - E_{lost}^{el}(t) \right) \cdot \text{NPSPrice}(t), \quad (5.1)$$

Remember that E_{lost}^{el} describes the lost energy due to overflow, which is a function of the variable $\text{Overflow}_n(t,\delta_p)\ \left[\text{m}^3/\text{s}\right]$ that measures the flow rate that cannot be processed by the turbines during high water periods. Having this, we just need to confirm the dependence on discharge and energy content. We also clarify that the subscript n specifies the current iteration in the SLO.

We have already indicated in (4.3) and (4.4) that E_{turb}^{el} and E_{pump}^{el} can be rephrased by use of $P_{turb}(t,p)$ and $P_{pump}(t,p)$ due to our hourly time resolution:

$$E_{turb}^{el}(t,p) = P_{turb}(t,p) \qquad \text{and} \qquad E_{pump}^{el}(t,p) = P_{pump}(t,p). \quad (5.2)$$

Further considerations will just be concerning turbines. The analogue holds for pumps. The linearized version of the power equation for turbines (4.16) gives a representation of $P_{turb}(t,p)$. If we plug this into the statement above, we immediately see the dependence on the discharge variable $Q_n(t,p)$. Doing so, we also bring in the variable h_n. We have the possibility to express the current head h_n in terms of the energy content $EC_n(t,\delta_p)$ through

$$\begin{aligned} h_n(t,\delta_p) &= \text{StowageHeight}(t,\delta_p) - h_{outlet}^{\delta_p} \\ &= \frac{h_{max}^{\delta_p} - h_{min}^{\delta_p}}{EC_{\delta_p}^{max}} \cdot EC_n(t,\delta_p) + h_{min}^{\delta_p} - h_{outlet}^{\delta_p}, \end{aligned} \quad (5.3)$$

where $h_{max}^{\delta_p}$ and $h_{min}^{\delta_p}$ are the top water respectively the drawdown water level and $h_{outlet}^{\delta_p}$ is the sea level of the outlet of dam δ_p. Inserting the combination of (5.2), (5.3) and (4.16) into (5.1) yields our proposed dependencies.

Clearly, (5.3) illustrates only one special power plant setting where there is a free-flowing section in between two dams. For simplicity we omit other possible cases as for instance the setting of two TH-HPLs in close range to each other, where the head of the upper plant

depends on the stowage height of the lower reservoir.[36]

Since we have shown that the profit equation depends on discharge, energy content and energy loss by overflow, we can in fact rewrite (5.1) as

$$\text{Profits} = \sum_{\substack{t \in HOURS, \\ p \in HPLS}} c_{t,p}^{ec} EC_n(t, \delta_p) + c_{t,p}^{q} Q_n(t, p) + c_{t,p}^{over} \text{Overflow}_n(t, \delta_p), \tag{5.4}$$

which is actually the form in which the solver receives the profit equation from AMPL. That is, equation (5.4) expresses the appearance of the objective function after going through AMPL's presolve phase while (5.1) shows our implementation of it in AMPL.

With this knowledge in hand we examine Figure 5.1 that shows the coefficients $c_{t,p}^{ec}$, $c_{t,p}^{q}$ and $c_{t,p}^{over}$ belonging to $EC_n(t, \delta_p)$, $Q_n(t, p)$ and $\text{Overflow}_n(t, \delta_p)$, $\forall t \in HOURS, p \in HPLS$. Notably, in Figure 5.1 we have $n = iter\_max$.
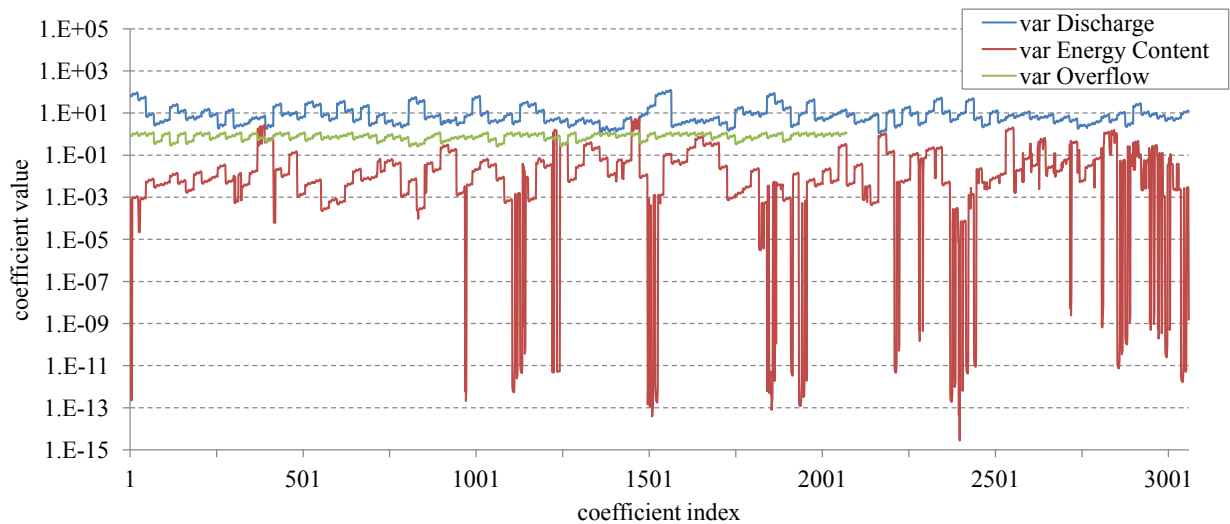

Figure 5.1 Values of the coefficients $c_{t,p}^{q}$, $c_{t,p}^{ec}$ and $c_{t,p}^{over}$ in the profit equation on a logarithmic y-scale. The x-axis just specifies the index of the coefficients. The simulation period is just one day, since otherwise there would be too many coefficients - here we have in total about 8000. $\big[$ Simulation period: 01.05.2011 - 02.05.2011, $max\_iter = 2 \big]$[37]

We see that the coefficients concerning the discharge and the overflow are roughly in the same order of magnitude and thus we have to focus on the red graph. The observed values in the magnitude band of 10^{-9} to 10^{-14} indicate that the problem concerning the MOSEK warning lies within the coefficients to $EC_n(t, \delta_p)$.

---

[36]In this case the stowage area of the lower reservoir reaches back to the dam upstream. Hence, if the headwater level of the lower HPL increases, also the tailwater level of the upper HPL will increase, which leads to a decrease of the head of the upper HPL. In such a case we will denote the upper HPL as *impounded*.

[37]We clarify that by the simulation period 01.05.2011 - 02.05.2011, we actually mean the time span from 01.05.2011 00:00 until 02.05.2011 00:00. Furthermore, the total number of coefficients of about 8000 is achieved, since we have about 3000 coefficients for the energy content, 3000 for the discharge and according to the green graph about 2000 for the overflow.

Hence, we look at them in detail. Since the first and the third addend of (4.16) are independent of $EC_n(t, \delta_p)$, we just have to consider the second one to compute $c_{t,p}^{ec}$. Filling in (5.3) into the second term of (4.16) and inserting it into (5.1) yields the terms

$$\underbrace{\frac{\eta \cdot \rho_w \cdot g}{10^6} \cdot Q_{n-1}(t, p) \cdot \left(\frac{h_{max}^{\delta_p} - h_{min}^{\delta_p}}{EC_{\delta_p}^{max}} \right) \cdot \text{NPSPrice}(t)}_{=c_{t,p}^{ec}} \cdot EC_n(t, \delta_p) + const. \qquad (5.5)$$

This just shows those parts of (5.1) that are important to calculate $c_{t,p}^{ec}$. In *const* we subsume several constants that arise when including (5.3) in the second term of (4.16).

Further on, we abbreviate $c_{ec} = c_{t,p}^{ec}$. From the form of c_{ec} we derive that scaling the data to say kWh instead of MWh will only lead to minor improvements in this case because changing to kWh would just mean to multiply c_{ec} by 10^3 which leaves us with still quite small coefficients in the range of 10^{-6} to 10^{-11}. Moreover, the overall range of all the coefficients in Figure 5.1 (from about 10^2 to 10^{-14}) would not change but only shift upwards by a factor of 10^3. Clearly, scaling is not the wished-for solution to the problem.

Analogously, this whole analysis can also be made for other HPL settings and all the constraints in our LP, leading to similar results. Notably, in the case of an impounded[38] TH-HPL p, we get the following coefficients to $EC_n(t, \delta_p)$, which we will need later on:

$$c_{ec} = \frac{\eta \cdot \rho_w \cdot g}{10^6} \cdot (Q_{n-1}(t, p) - Q_{n-1}(t, p-1)) \cdot \left(\frac{h_{max}^{\delta_p} - h_{min}^{\delta_p}}{EC_{\delta_p}^{max}} \right) \cdot \text{NPSPrice}(t), \qquad (5.6)$$

where $p - 1$ denotes the HPL directly above p.

5.1.2 SLO Tolerance Parameter

A remedy to the problem with the near zero elements becomes obvious when considering $Q_{n-1}(t, p)$ in (5.5) and the fact that the algorithm solving the LP is an approximation method. The case that a turbine is not running can be expressed by zero discharge. Since we do not get an exact solution from our solver, the discharges we obtain will not be equal zero but mostly take on very small values, say around 10^{-8}. Hence, $Q_{n-1}(t, p)$ dominates the order of the coefficients c_{ec} in (5.5). Furthermore, the factor $\frac{\eta \cdot \rho_w \cdot g}{10^6}$ is of order 10^{-3} and for big reservoirs the order of factor $EC_{\delta_p}^{max}$ can be as high as 10^6. The combination of those facts causes the outliers in the red graph in Figure 5.1.

To get rid of those, we introduce a tolerance parameter ε_{SLO} that gives a threshold to where the discharge parameters of iteration $n - 1$ are treated as zero. We mostly use $\varepsilon_{SLO} = 10^{-2}$, since we suppose that discharges lower than this value are practically irrelevant. In view of that, we can restate a modified version of the SLO - algorithm:

---

[38]cf. footnote 36

Algorithm 5.1 (modified SLO)

1: ***Given*** h_0, Q_0 ... *initial data for linearization*
2:       *iter_max* $\in \mathbb{N}$ ... *maximal number of iterations*
3:       $\varepsilon_{SLO} > 0$ ... *SLO tolerance parameter*
4: ***let*** $h^* \leftarrow h_0$, $q^* \leftarrow Q_0$
5: ***load*** *linearized LP, data files*
6: ***for*** $n = 1, \ldots, iter\_max$ ***do***
7:     ***if*** $n > 1$ & *iter_max* $\neq 1$ ***then***
8:         ***if*** $Q_n < \varepsilon_{SLO}$ ***then***
9:             ***let*** $q^* \leftarrow 0$
10:        ***else***
11:            ***let*** $q^* \leftarrow Q_n$
12:        ***end if***
13:        ***let*** $h^* \leftarrow h_n$
14:        ***update*** *linearized LP*
15:     ***end if***
16:     ***solve*** $LP(h^*, q^*)$
17: ***end for***

Figure 5.2 shows the coefficients c_{ec} that we get from implementing the suggested tolerance parameter as in Algorithm 5.1. To be able to compare more easily with the version without using ε_{SLO}, we restate the red graph of Figure 5.1.
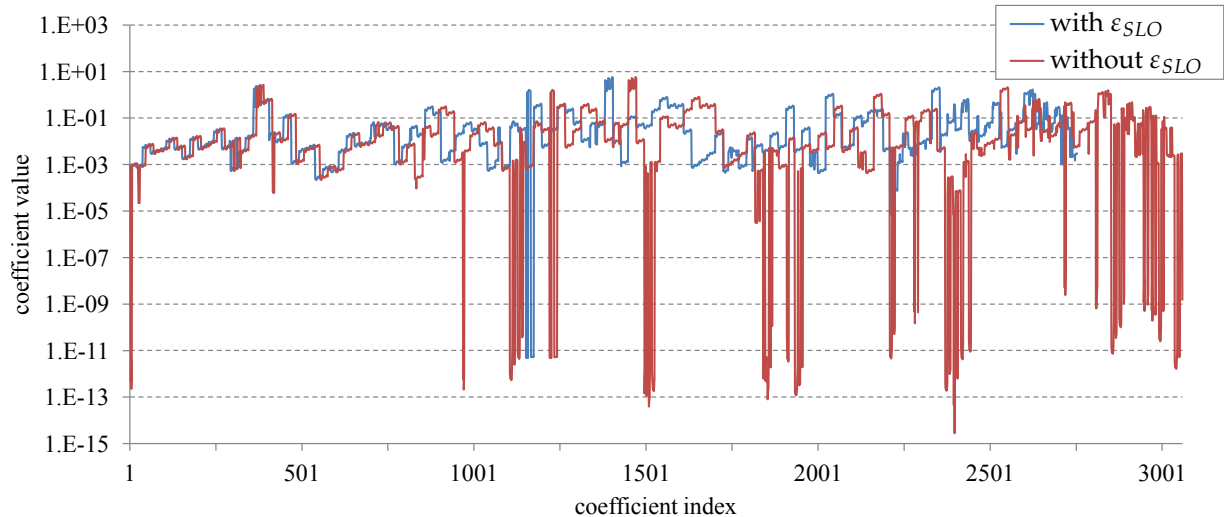
Figure 5.2 Comparison of the values of the coefficients c_{ec} with using ε_{SLO} and without it. We used the same simulation period as in Figure 5.1 and again the x-axis just specifies the index of the coefficients. $\big[$ Simulation period: 01.05.2011 - 02.05.2011, *max_iter* $= 2$ $\big]$.

It suggests that by zeroing redundant factors, we can eliminate most of the outliers. We notice that the blue line still has a peak close to index 1200. The coefficients in (5.6) give an

explanation to this behavior. Since we have the term $Q_{n-1}(t,p) - Q_{n-1}(t,p+1)$ in (5.6), the possibility arises that those two cancel out. Again, due to numerical inexactness instead of a cancellation we obtain very small values for $Q_{n-1}(t,p) - Q_{n-1}(t,p+1)$ and thus for c_{ec}, which explains the peak. This happens most likely when HPL p and $p+1$ have the same Q^{max}, p is impounded and both are turbining at approximately maximum discharge in time step t.

If we want to eliminate it, we either trust the accuracy of the turbine data (data to maximal discharge Q^{max}), in which case we have to set $Q_{n-1}(t,p) - Q_{n-1}(t,p+1) = 0$ or we pertube the data by an insignificant value, e.g. $Q_j^{max} + 1, j \in TURBS(p)$, such that the difference will be of order 1 in this particular time step t.

5.1.3   Model Size

We also note in Figure 5.2 that the number of coefficients diminished by about 250, since the blue line stops close to index 2750. This observation lets us hope that the computation time can be sped up due to a reduction in model size. When talking about the size of a model we refer to the number of constraints and variables. These counts can be given in two different ways. First, we can refer to the number of variables and constraints that are passed to the solver by our modelling language after the presolve phase[39]. The second possibility is to state them before a presolver is started, i.e. the number of constraints and variables that are given by the implemented model.

The following two figures verify that at large scale, i.e. when simulating over a much longer time span of say one year, we really get a smaller model when using Algorithm 5.1 instead of Algorithm 4.1.

In each of the runs in Figure 5.3 and Figure 5.4 we iterate 4 times. Looking at *iter_1* in Figure 5.3, we see that independent of the use of ε_{SLO} the difference between the numbers before and after the presolver is very low - just about 20 000 constraints can be dropped by the presolve phase.[40] We also clarify that the first iteration always has the same number of constraints with and without ε_{SLO}, since we always start both algorithms with the same initial model data and since the scope of our tolerance parameter just starts in the second iteration. Thus, we can concentrate on *iter_2*, *iter_3* and *iter_4*.

We see that before the presolver we get about $3.9 \cdot 10^6$ constraints without ε_{SLO} and $3.3 \cdot 10^6$ constraints with ε_{SLO}. Hence, using Algorithm 5.1 we decreased the count of constraints by 15% on average, which is quite a significant reduction in model size. Furthermore, the red and the blue graph stay fairly stable when comparing the different runs with variations in between *iter_2*, *iter_3* and *iter_4* in the magnitude of just 10^4.

---

[39]cf. Section 3.4

[40]In Figure 5.3 we cannot really see a difference between the four dots in *iter_1* due to the scaling of the y-axis that is not suitable to show a difference of magnitude 10^4. However, there is a small gap.
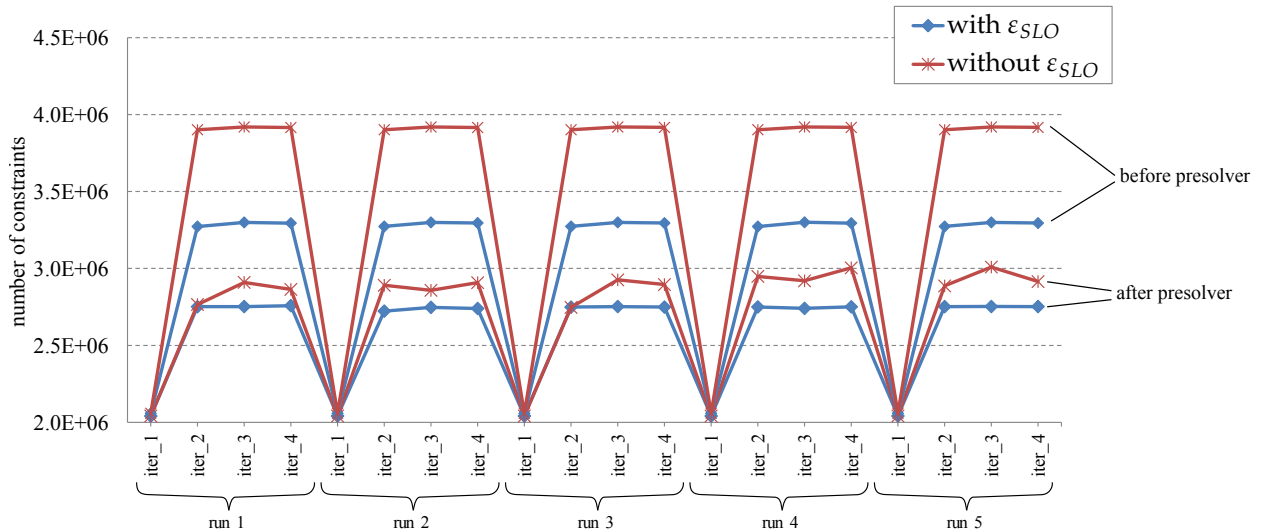
Figure 5.3 Comparison of the number of constraints in every iteration before and after AMPL's presolve phase as well as with and without implemented tolerance ε_{SLO}. We performed 5 runs for each of the two algorithms with the exact same simulation period and data for each run. $\big[$ Simulation period: 01.01.2011 - 01.01.2012, $max\_iter = 4$ $\big]$

After the presolve phase, due to diverse rearrangements of the model, we get less constraints than before. As we have already mentioned, the presolver tries to build a more compact version of the model. We also notice that the number of constraints without ε_{SLO} varies to some extent and that the blue graph behaves much more stable during those 5 runs. Hence, cancelling out our redundant near zero coefficients helped the presolver to drop even more constraints and thus to stabilize the count of constraints in higher iterations.

Considering Figure 5.4, the first thing we observe is that before presolving the red and the blue line are identical. We actually expected this behavior, since zeroing coefficients in certain constraints is not going to alter the number of variables. This quantity is fixed by the implementation, which becomes clear if we consider for instance the variable that describes the energy content $EC(t, \delta), t \in HOURS, \delta \in DAMS$. A missing variable would imply that the energy content of the dam δ at the specific time t does not have any value, which is not admissible. Hence, preliminary to presolving, there cannot be any reduction in model variables due to the use of ε_{SLO}.

After the presolver, the first iteration also has identical values with and without ε_{SLO}, which is once more due to identical initial data and the limited scope of ε_{SLO}. Moreover, the number of variables increases significantly after the presolve phase, since the presolver introduces a lot of slack variables to set up the standard form (3.5) of the given LP. That is, for every constraint, which is declared an inequality constraint, there will be assigned a slack variable to generate an equality constraint.[41]

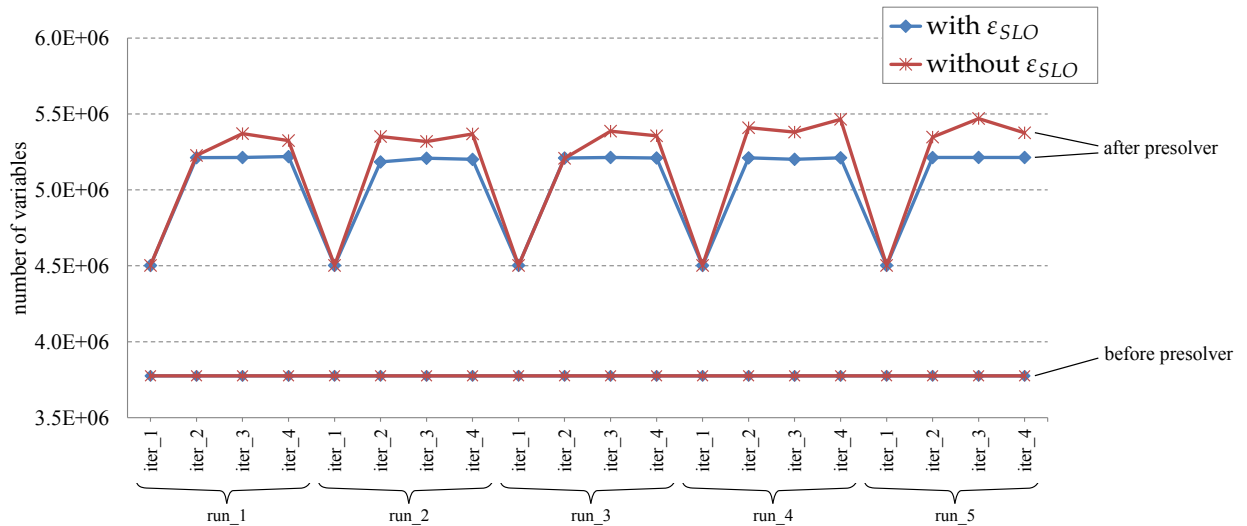[41] cf. equation (3.8) in Section 3.2

Figure 5.4 Comparison of the number of variables in every iteration before and after AMPL's presolve phase as well as with and without implemented tolerance ε_{SLO}. Again, the exact same simulation period and data for each run was used. $\big[$ Simulation period: 01.01.2011 - 01.01.2012, $max\_iter = 4\,\big]$

Leaving the first iteration out of consideration, we see again that, as in the case of the constraints, implementing the tolerance ε_{SLO} helps to keep the number of variables close to a certain level - namely $5.2 \cdot 10^6$ on average. Clearly, since Figure 5.4 shows that after the presolver we have less constraints by applying ε_{SLO} than without it, the associated slack variables do not need to be inserted into the model. This also explains the similarity of the after-presolving-graphs in Figure 5.3 and Figure 5.4.

5.1.4 Computation Time

Reduction in size of the model inevitably leads to the question if we can exploit this fact to make the computation time faster. We examine the effects concerning the time MOSEK's interior-point algorithm needs to solve the given LP. To be clear, this is not the overall computational time, since AMPL also needs time to translate the given model code, set up the LP in standard form and update it during every iteration. Regarding Algorithm 5.1, we just look at the time to process line 16.

Figure 5.5 shows that without using ε_{SLO} the time per iteration is mostly a little bit higher than the one with ε_{SLO}. The difference of the mean values for those 10 runs is more than 5 minutes, although we have to be careful with this statement, since the peaks in the red graph have a big impact on the average value. However, during all the simulations we did, we experienced that these peaks, where the interior-point algorithm needs unnaturally long to compute the optimal solution, quite regularly occur without ε_{SLO}. With the implementation of the SLO tolerance there still happen to be some peaks as for instance in run 2 but they do not appear as often and are not as high. Hence, we conclude that we managed to modify the LP to be numerically more stable and easier to solve.
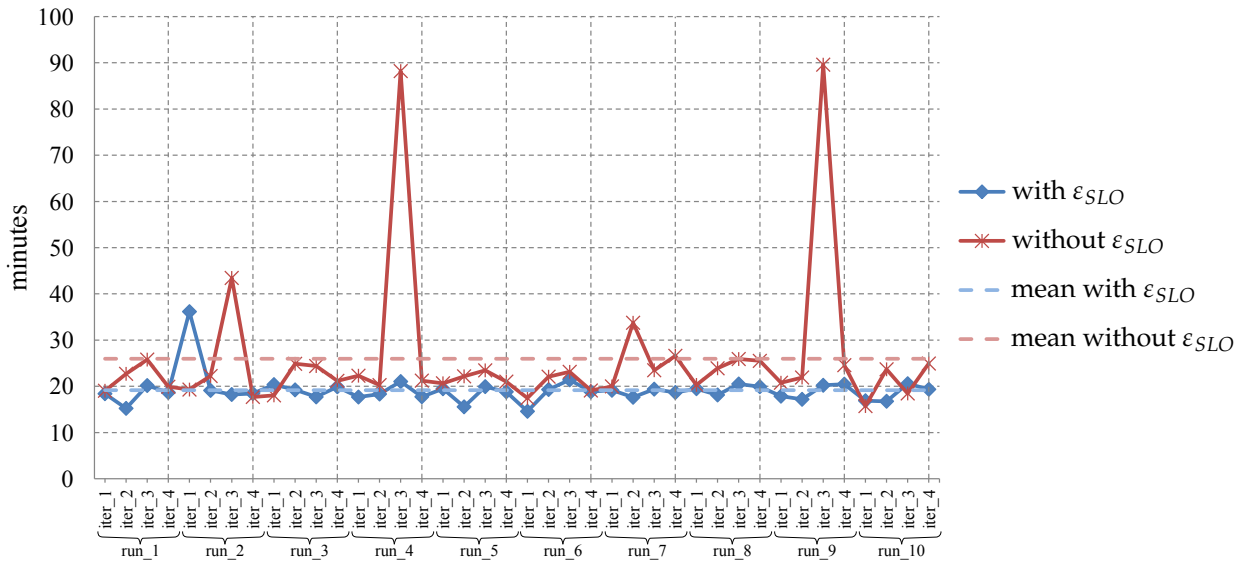
Figure 5.5 Comparison of the computational time the MOSEK interior-point method needs per iteration. $\left[\,\text{Simulation period: 01.01.2011 - 01.01.2012, } max\_iter = 4\,\right]$

Next, Figure 5.6 presents the accumulated time of the MOSEK solver per run, i.e. summing up the times for the four iterations of each run in Figure 5.5. It underlines the fact that we gain computation time by use of ε_{SLO}. With the two peaks we already referred to, the average difference between the two versions is about 30 minutes. Neglecting the peaks would still lead to an average time gap of about 15 - 20 minutes.
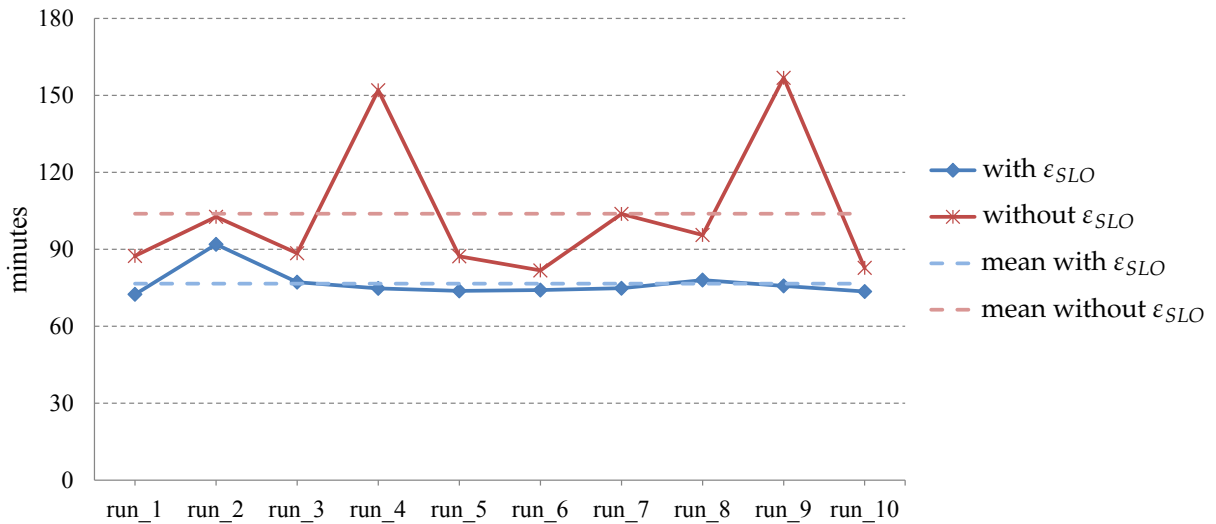

Figure 5.6 Comparison of the accumulated computational time of the MOSEK interior-point method. I.e., we summarized the times for each run in Figure 5.5. $\left[\,\text{Simulation period: 01.01.2011 - 01.01.2012, } max\_iter = 4\,\right]$

5.2 Speed up SLO Computation

One of the main tasks in this work was to investigate, if the computation could be made somehow faster. Through installing the tolerance parameter ε_{SLO}, we already achieved some improvement, though still not satisfyingly much. Throughout simulating we noticed that most of the computation time is actually needed by our modelling language AMPL. Figure 5.7 shows that, especially in iterations 3 and 4, AMPL needs as much as 12 times the computation time of the MOSEK solver for its calculations. Moreover, the fact that AMPL needs significantly more time in the second, third and fourth iteration indicates that updating the model in the SLO algorithm (see line 14 in Algorithm 5.1) comes quite costly.
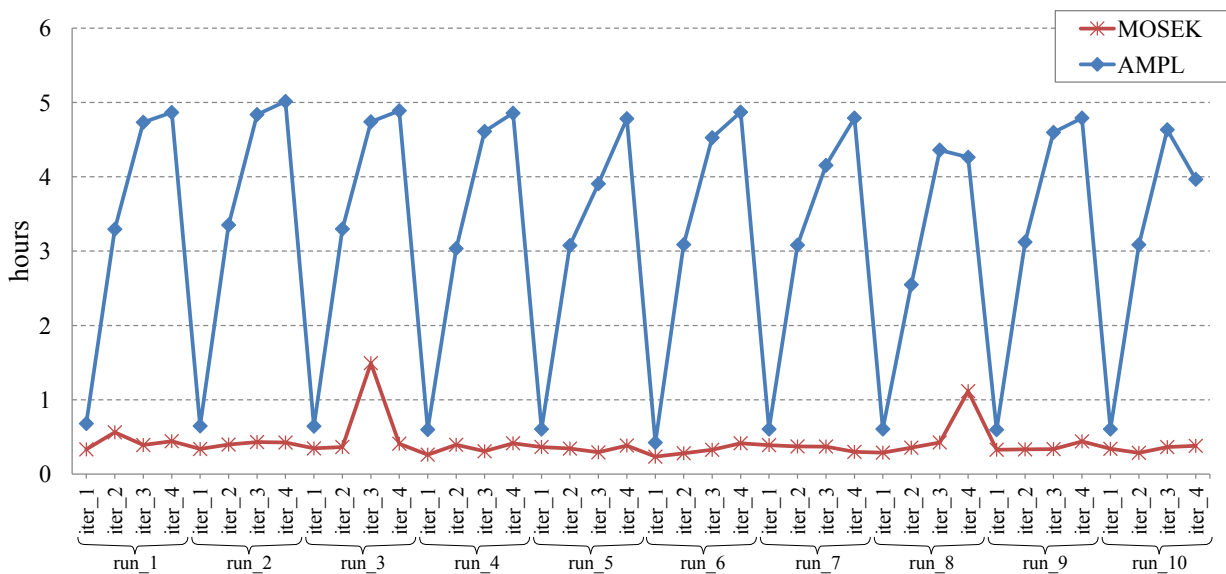

Figure 5.7 Comparison of the computation time for the MOSEK solver and the time for AMPL's calculations per iteration without using ε_{SLO}. The red line is the same as in Figure 5.5. $\left[\, \text{Simulation period: 01.01.2011 - 01.01.2012}, \mathit{max\_iter} = 4 \,\right]$

The AMPL calculations that actually consume this much time are first of all the process of generating respectively updating the model, i.e. translate the given code and remove variables that are just specified to make the code more readable, and secondly the AMPL-internal presolver[42]. However, the first process mentioned devours most of the computation time. Further on we will call it the *genmod-phase*, short for generating the model.

5.2.1 Identification of Redundancy

Investigation of AMPL's internal proceedings[43] revealed that there was some redundancy during the update phase respectively during the assignment phase of Algorithm 5.1 (see

---

[42]AMPL itself has a built-in presolver, which functions similarly to MOSEK's presolver. For further information on the functionality of a presolver see Section 3.4.

[43]One can investigate AMPL-intern calculations through the AMPL options `times` and `gentimes`, which request a summary of the time and resources consumed to generate the model. For further details see [7].

lines 8 - 13). We experienced unnaturally long computation times when AMPL processed the assignments

$$q^* \leftarrow Q_n \quad \text{and} \quad h^* \leftarrow h_n, \tag{5.7}$$

due to an intermediate start of the time-intensive genmod-phase for some model components. Normally, the genmod-phase is just triggered by the **solve**-command. This situation can be explained as follows:

AMPL offers some commands, like the **let**-command to provide access to our specified data. In the lines 8 to 13 of Algorithm 5.1 we change given parameter values by assignment of variables belonging to the current optimal solution. In some cases, trying to assign a variable to a parameter can trigger a re-run of the genmod-phase. Namely, in context of our example code, the assignment order is particularly of interest. If we assign at first $q^* \leftarrow Q_n$ and afterwards $h^* \leftarrow h_n$ this causes the genmod re-run referred to. This behavior is due to the fact that the values of h_n depend on $q^* = Q_{n-1}$.

To show this, we recall equation (5.3), which states that the head h_n depends on the energy content EC_n. The values of EC_n are determined through the energy balance constraint in (4.11), which depends on E_{turb}^{pot}, E_{pump}^{pot}, $E_{overflow}^{pot}$ and $E_{sideinflow}^{pot}$. For convenience we restate equation (4.14) that specifies the potential energy of a turbine:

$$E_{turb}^{pot}(t, p) = \frac{1}{10^6} \cdot g \cdot \rho_w \cdot Q(t, p) \cdot h(t, \delta), \quad j \in TURBS(p), \quad t \in HOURS.$$

This shows that E_{turb}^{pot} depends on the non-linear term $Q(t, p) \cdot h(t, \delta)$. Using our linearization of Section 4.2, we conclude that Q_{n-1} actually has influence on h_n. Consequently, changing the values of $q^* = Q_{n-1}$ affects the values of h_n and will force AMPL to update those parts of the model in reach of influence to determine up to date values for h_n before assigning $h^* \leftarrow h_n$.

Notably, this automatic update of h_n leads not only to unnecessary calculations but yields in fact incorrect values for h^*. We want h^* to take on the computed stowage height h_n of the preliminary optimal solution, but as mentioned assigning $q^* \leftarrow Q_n$ leads to an update of h_n and thus to different values overwriting the correct ones.

In summary, it can be stated that updating q^* and h^* in the particular order as presented in (5.7) leads to starting the genmod-phase twice in one iteration - once when trying to assign $h^* \leftarrow h_n$ and a second time when executing the **solve**-command. In the next section, we will try to make the implementation more efficient by calling the genmod-phase just once, which will yield great economy of runtime.

5.2.2 Modification in SLO Parameter Update

There are actually two quite simple approaches to the problem outlined above. The first one involves rearranging the order of assignment such that the sequence of changes to the model

does not affect each other. That is, in our case, since Q_n does not depend on $h^* = h_{n-1}$ we assign first $h^* \leftarrow h_n$ and afterwards $q^* \leftarrow Q_n$. This will prevent AMPL from running the genmod-phase in between the assignments section but only start it when calling the **solve**-command.

In some cases it may not suffice to just flip the order of the **let**-commands to get the result we want. Assume that analogously to h_n depending on q^* also Q_n depends on h^*. Then, the order would not matter at all, since in both cases the supposed dependencies pose an obstacle and a re-run would be triggered either way. In this instance we suggest to introduce a temporary parameter tmp that holds either Q_n or h_n. For now, we assume that tmp takes the values of h_n. If we choose Q_n the analogue of the following considerations is valid.

We assign $q^* \leftarrow Q_n$, which is perfectly possible, since Q_n just depends on h^* but not on tmp. Last, we set $h^* \leftarrow tmp$, which also cannot activate the genmod-phase since tmp does not depend on anything that was previously changed. In view of this, we can state our final version of the SLO algorithm:

Algorithm 5.2 (final SLO)

 1: **Given** $h_0, Q_0 \dots$ *initial data for linearization*
 2: $\qquad$ *iter_max* $\in \mathbb{N} \dots$ *maximal number of iterations*
 3: $\qquad$ $\varepsilon_{SLO} > 0 \dots$ *SLO tolerance parameter*
 4: $\qquad$ *tmp* $\dots$ *temporary parameter*
 5: **let** $h^* \leftarrow h_0, q^* \leftarrow Q_0$
 6: **load** *linearized LP, data files*
 7: **for** $n = 1, \dots, iter\_max$ **do**
 8: $\quad$ **if** $n > 1$ & $iter\_max \neq 1$ **then**
 9: $\qquad$ **let** $tmp \leftarrow h_n$
10: $\qquad$ **if** $Q_n < \varepsilon_{SLO}$ **then**
11: $\qquad\quad$ **let** $q^* \leftarrow 0$
12: $\qquad$ **else**
13: $\qquad\quad$ **let** $q^* \leftarrow Q_n$
14: $\qquad$ **end if**
15: $\qquad$ **let** $h^* \leftarrow tmp$
16: $\qquad$ **update** *linearized LP*
17: $\quad$ **end if**
18: $\quad$ **solve** $LP(h^*, q^*)$
19: **end for**

Next, we present the outcome we obtain from implementing Algorithm 5.2 compared to the runtime of the two former versions in three figures. We put the total computation time per iteration and per run for all three versions next to each other and present the analogue of Figure 5.7 for our final version of the SLO algorithm to see how the shares in computation time for MOSEK and AMPL have changed.
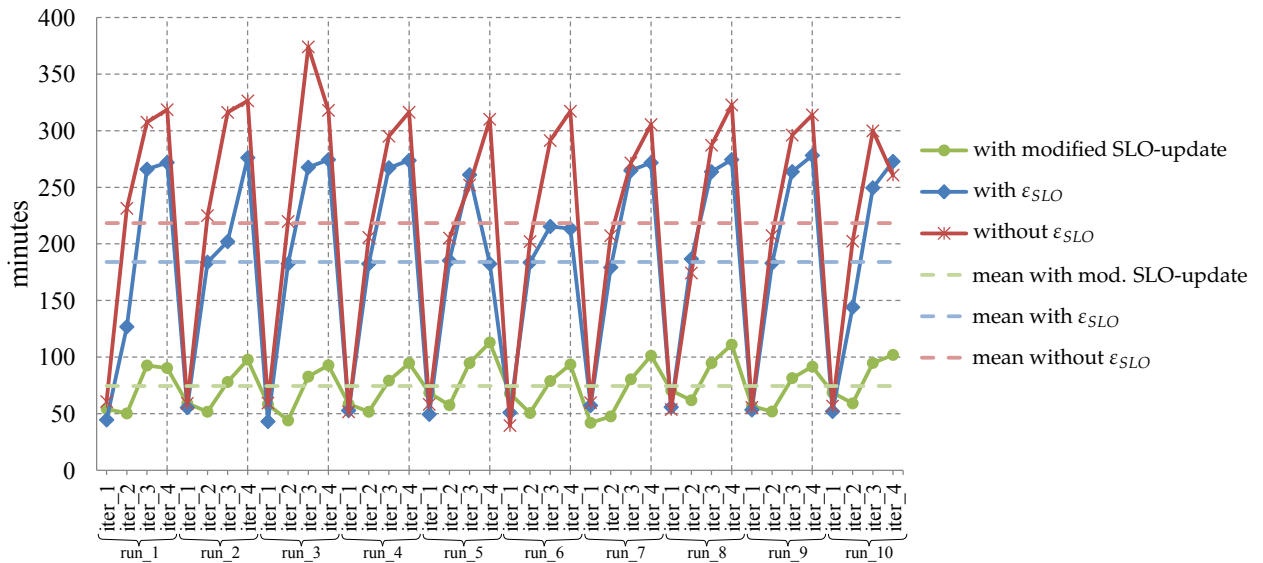
Figure 5.8 Comparison of the total computation time per iteration of all three versions of the SLO algorithm. The data presented shows the computation times of the MOSEK solver plus the times AMPL needs for its calculations. $\big[$ Simulation period: 01.01.2011 - 01.01.2012, $max\_iter = 4$ $\big]$

Figure 5.8 and 5.9 give a good overview of all the runtimes of the three different algorithms. At first, we investigate Figure 5.8: Compared to the basic version of the SLO, Algorithm 5.1 already yields an average reduction of close to 35 minutes per iteration indicated by the difference between the red and the blue dotted line. We recall from Figure 5.5 that in case of comparing Algorithm 4.1 and 5.1 the solver runtime could be reduced by about 5 minutes per iteration.

Hence, we conclude that the usage of ε_{SLO} also has moderate effects on the time AMPL needs for its calculations, namely a reduction of about 30 minutes. This is not surprising, since the model size is reduced due to the implementation of the tolerance parameter, which results to some extent in a faster set up and update of the model.

Probably the first thing that leaps to the eye in Figure 5.8 are the outstanding results we get by our modified assignment section as stated in Algorithm 5.2 (green graph). On average, contrasting the green and the red line, we experience a decrease in computation time per iteration by more than 2 hours, which tells us that we managed to cancel out quite a big section of redundant calculations.

Figure 5.9 especially demonstrates the enormous economy of time we get by use of Algorithm 5.2. Here, we sum up the runtimes of 4 iterations. Thus, by multiplying the mean time differences of Figure 5.8 by 4 we approximately arrive at the average values of Figure 5.9. The gap between the dotted blue and the dotted red line amounts to more than 2 hours per run. Using the modified SLO-update adds as much as 7.5 hours to this value. Hence, we manage to save 9.5 hours of total computation time per run on average, which is quite impressive, considering the simplicity of the modifications we made in the code.
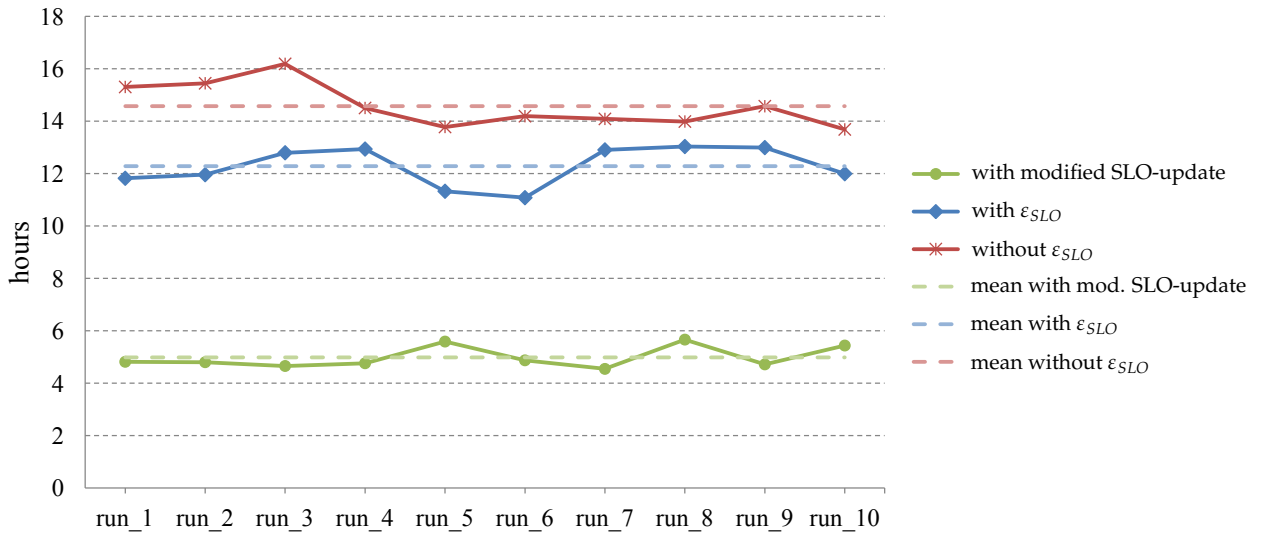
Figure 5.9 Comparison of the total computation time per run of all three versions of the SLO algorithm. That is, we show the total calculation time for a whole run using 4 iterations. $\left[\text{ Simulation period: 01.01.2011 - 01.01.2012, } max\_iter = 4 \right]$

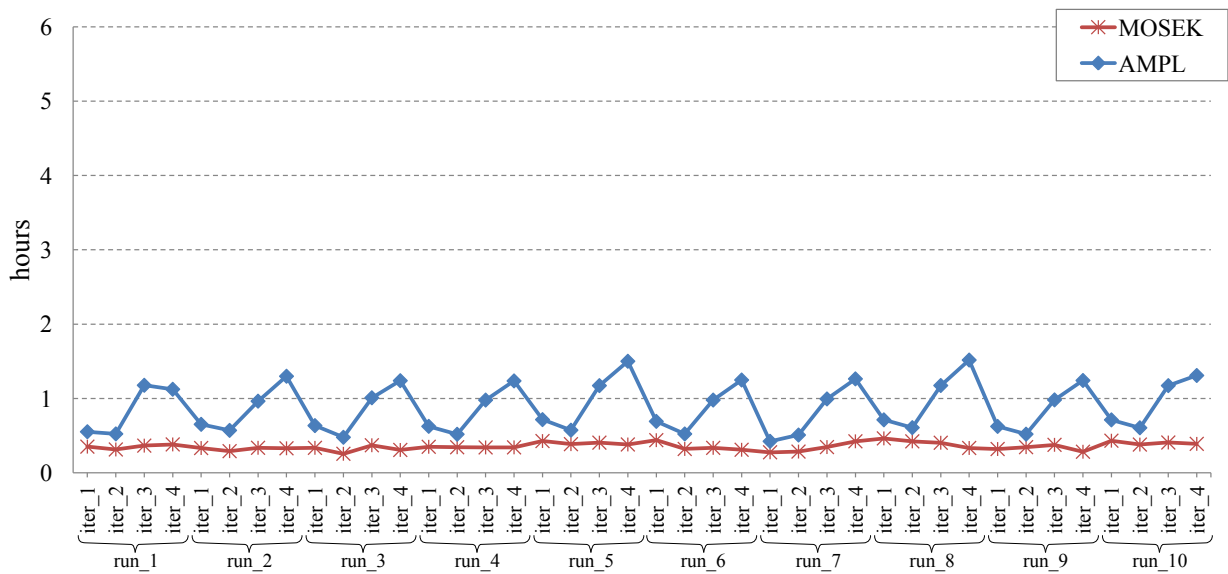Next, we compare the times of the MOSEK solver and those of AMPL for our final version of the SLO algorithm:

Figure 5.10 Comparison of the computation time for the MOSEK solver and the time for AMPL's calculations per iteration using Algorithm 5.2. $\left[\text{ Simulation period: 01.01.2011 - 01.01.2012, } max\_iter = 4 \right]$

To better see the big improvement we achieved by Algorithm 5.2, we display the same y-axis scale as in Figure 5.7. Here, AMPL's calculation time hovers around 1 hour per iteration. Noticeable is also that the second iteration needs about the same or even less time than the first one, which is obviously not the case in Figure 5.7.

5.3 Choice of Initial Values for Discharge

In this section we want to investigate how changing the initial values that are passed to the model can affect our computations. This fits in seamlessly with this chapter so far, since our main objective to save calculation time will be also achieved in this part. However, we will not reach the economy of time directly, but through a significant reduction in linearization errors and the consequential possibility to perform less iterations than before to get to the wished for accuracy.

As already indicated in line 1 of Algorithm 5.2, we need initial data for our parameters h_0 and Q_0. Further on, we will only concentrate on the initial discharge Q_0. When a calculation depends on starting values, the question often arises, if we can find a better choice of values to improve the algorithm's behavior, be it concerning the computation time, faster decrease in error, numerical stability, etc. In our case, the parameter Q_0 particularly affects the linearization error we make in the first iteration of our model. Since one iteration run significantly depends on the foregone iteration, we hope that making a sophisticated selection of the initial discharge will lead to a faster overall error convergence.

5.3.1 Approach

We clarify that we will only consider the default discharge for turbines since PST-HPLs are actually not that important in the Sweden model (there exist only two of them).

Constant Default Values

Our first approach to choosing an appropriate initial value for the parameter $Q_0(t, p)$ is to start at constant zero and work our way up trying different increasing values proportional to the maximal discharge possible of each HPL. According to the bound on the discharge of each turbine/pump in (4.6), we can define the maximum discharge of an HPL as

$$Q_p^{max} := \sum_{j \in TURBS(p)} Q_j^{max}. \tag{5.8}$$

Hence, in the next section we will show the simulation results regarding the three selected values

$$Q_0(t, p) = 0, \; \frac{Q_p^{max}}{10}, \; Q_p^{max}.$$

$Q_0(t, p) = 0$ means that the turbines are not running at all during the whole simulation period while $Q_0(t, p) = Q_p^{max}$ describes a state where the turbines are at full load at all times. $Q_p^{max}/10$ can be interpreted as something in between those two - every power plant operates in constant partial load. Neither of those states makes practically any sense but we will see how they affect the linearization error.

Default Values According to Electricity Price

The second idea we want to suggest is a little bit more complicated. Normally, the index $n-1$ in $Q_{n-1}(t,p)$ describes the optimal solution in the preliminary iteration. Since we maximize the profits in our model, every optimal solution depends to a big extent on the electricity price. Thus, if $n=1$, the logical idea would be to find starting parameters $Q_0(t,p)$ that also take on values according to the electricity price in a similar way as an optimal solution does. A possible method to achieve that works as follows:

At first, we generate a trendline or regression curve from our used electricity data points, which will be denoted as $y_{trend}(t), t \in HOURS$. The electricity price is given in hourly resolution, i.e. for every hour of a day we have a data point $y_{NPS}(t)$, where NPS indicates that this is the electricity price at the Nord Pool Spot electricity stock market. We define $y_{trend}(t)$ as follows:

$$y_{trend}(t) := \begin{cases} \frac{1}{\frac{n}{2}-t} \sum_{i=1}^{\frac{n}{2}+t} y_{NPS}(i), & t \leq \frac{n}{2} \\ \frac{1}{N-(t-\frac{n}{2})} \sum_{i=t-\frac{n}{2}}^{N} y_{NPS}(i), & t \geq N - \frac{n}{2} \\ \frac{1}{n} \sum_{i=-\frac{n}{2}}^{\frac{n}{2}} y_{NPS}(t+i), & \text{else,} \end{cases} \tag{5.9}$$

where $n = 24$ and $N = card(HOURS)$, the cardinality of the set $HOURS$, i.e. the total number of elements in $HOURS$. Definition (5.9) yields that we calculate $y_{trend}(t)$ by the mean value of $n/2$ points before and $n/2$ points after the current hour t. The first and the second case in (5.9) are to determine the values of $y_{trend}(t)$ on the right and the left boundary of the set $HOURS$. For better understanding of (5.9) we schematically show in Figure 5.11 which elements are taken in each of the average values.
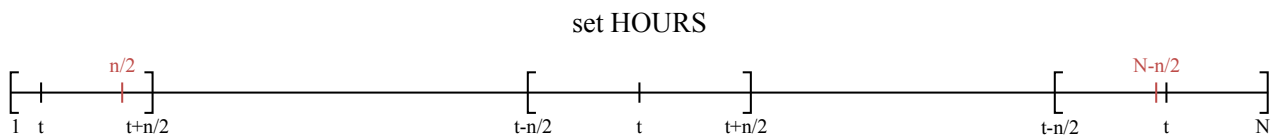

Figure 5.11 Schematic drawing of the *HOURS* set and the elements we use in the different mean values to build up the regression line $y_{trend}(t)$.

Comparing $y_{trend}(t)$ with the actual electricity price we see that we obtain a curve that represents the tendency of the price by smoothing the peaks (see Figure 5.12). Of course this is not surprising, since we apply average values to set up $y_{trend}(t)$.

Each of the blue dots in Figure 5.12 represents the mean NPS price of one specific hour and the red graph shows our trendline $y_{trend}(t)$ according to the price data of the year 2011. Moreover, the straight red dotted line presents the annual average price value minus the standard deviation of the NPS price. We will explain the use of this later on.
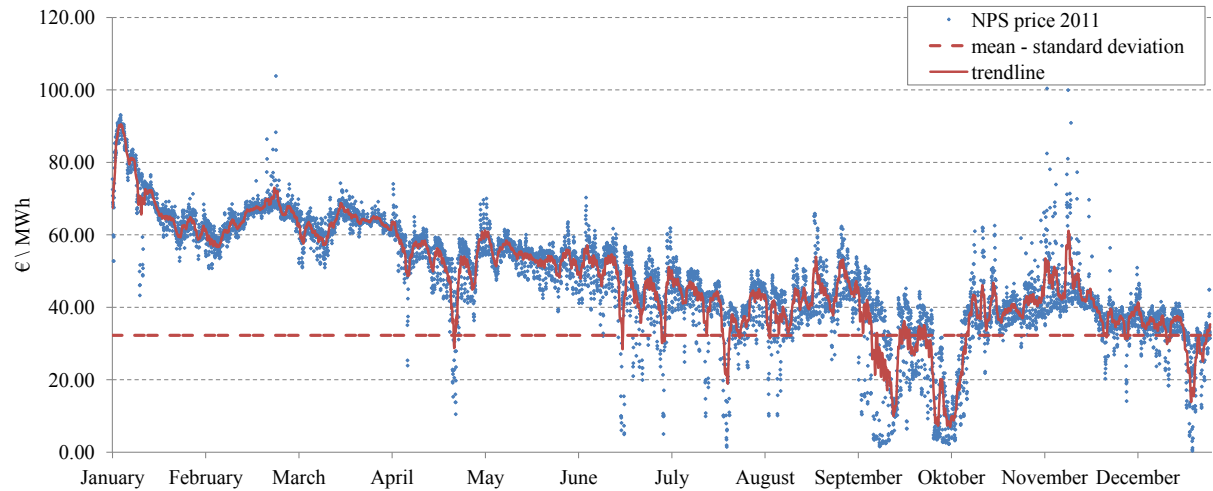
Figure 5.12 Hourly NPS price of the year 2011 with the trendline $y_{trend}(t)$ accordingly. For later analysis we also plotted the annual average value minus the standard deviation of the NPS price data.

Our principle to arrive at starting points for Q_0 is made up of two criteria. Before we examine them, we introduce a tolerance band around $y_{trend}(t)$ with the band width ε_{band}. Since the discharge concerning an optimal solution mostly jumps between the maximal and zero discharge according to high and low electricity price, we want to simulate the same. Hence, we set $Q_0(t, p) = 0$ if the price drops below the lower boundary of the band and $Q_0(t, p) = Q_p^{max}$ if it stays above it. Of course, we could also just use $y_{trend}(t)$ as our cutting line to decide between zero and maximal discharge but the benefit of implementing ε_{band} is simply that we have another variation parameter at hand. Otherwise the criterion would be quite static and hardly customizable.

The second criterion we add takes into account the case of unnaturally low electricity prices compared to the average price during the simulation period. In such an instance it will be favorable for TH-HPLs and ST-HPLs to cut off the turbines and impound water as much as possible to use it later on when the prices are increasing again.

To capture this phenomena, we installed the line that shows the mean value μ minus the standard deviation σ. For all hours in which the price is lower than that line we also set $Q_0(t, p) = 0$. Since σ represents a measure for the dispersion of the data, we figure that this is a good way to get hold of such outliers in the NPS price. To sum up, we state the whole criterion in pseudo code:

if $y_{NPS}(t) < y_{trend}(t) - \varepsilon_{band} \ || \ y_{NPS}(t) < \mu - \sigma$ **then**
    let $Q_0(t, p) \leftarrow 0, \quad t \in HOURS, p \in HPLS$
else
    let $Q_0(t, p) \leftarrow Q_p^{max}, \quad t \in HOURS, p \in HPLS$
end if

To allow a closer look at the electricity price data in connection with our regression curve we restate one segment of Figure 5.12. Clipping out the month August and September and
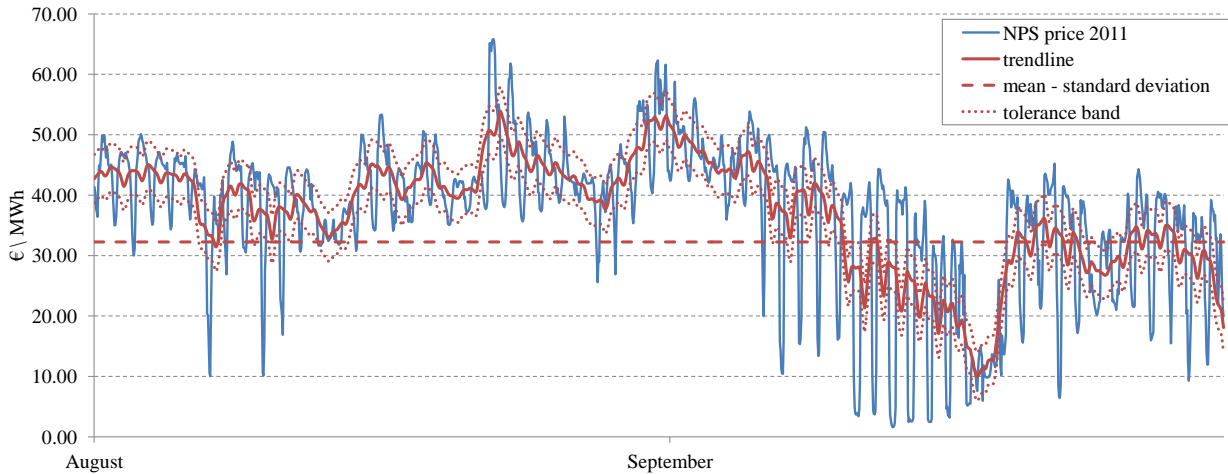
Figure 5.13 Hourly NPS price in the month of August and September with the trendline $y_{trend}(t)$ and the tolerance band $y_{trend}(t) \pm \varepsilon_{band}$.

including the tolerance band with a band width of $\varepsilon_{band} = 4$ yields Figure 5.13.

5.3.2   Results

The previous section discussed how we are going to choose our initial points $Q_0(t, p)$. Now, we will examine the results we achieve by varying the initial values. To be precise, we will exclusively investigate the linearization error and the computation times we get.

In Section 4.2.1, more exact in (4.19) we have already defined the overall error e_{MW} that we make through linearization in one iteration. Next, we present the error e_{MW} we obtain by applying the different cases of Q_0 discussed above.
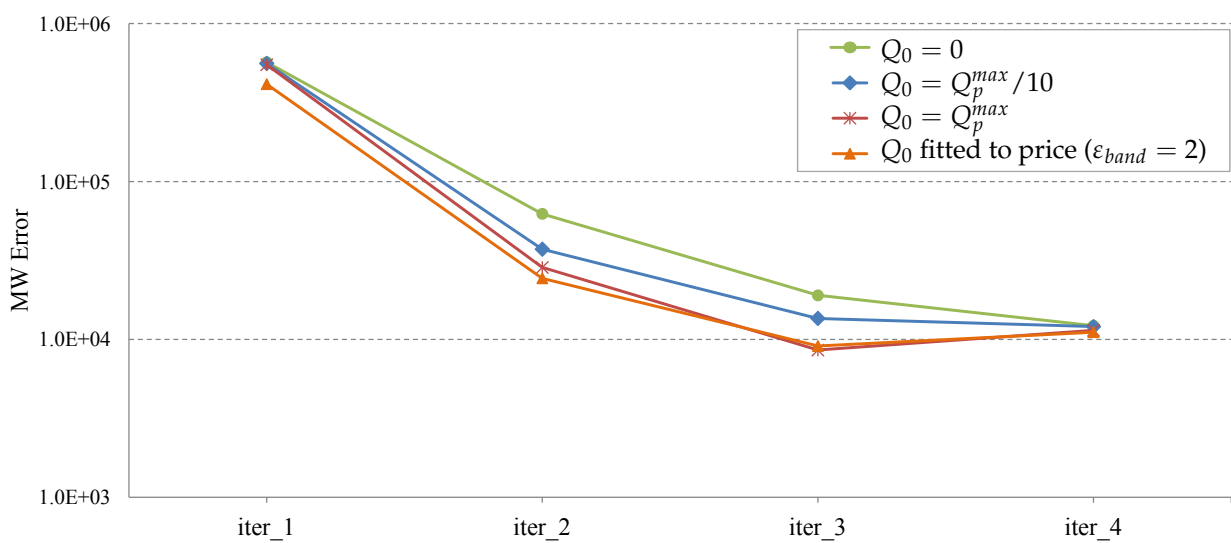

Figure 5.14 Comparison of the absolute linearization errors e_{MW} per iteration for different choices for the initial values $Q_0(t, p)$. [Simulation period: 01.01.2011 - 01.01.2012, $max\_iter = 4$]

For convenience, we also state the table of values used to generate Figure 5.14. This should make it easier to actually quantify the improvements achieved.

	$Q_0 = 0$	$Q_0 = Q_p^{max}/10$	$Q_0 = Q_p^{max}$	Q_0 fitted to price
iter_1	568923	559360	548606	413447
iter_2	62385.5	37264.6	28546.3	24392.1
iter_3	19006.6	13578.7	8558.18	9087.8
iter_4	12201.9	12047.6	11420.4	11116.2

Table 5.1 The values of the absolute linearization errors e_{MW} per iteration for different choices of the initial values $Q_0(t, p)$

Figure 5.14 and Table 5.1 tell us that already choosing $Q_0(t, p)$ constant greater than 0 improves the error. Moreover, the closer we get to the maximal discharge, the better the performance in the second and third iteration. In the first iteration we do not see any big enhancement for $Q_0(t, p) = \frac{Q_p^{max}}{10}$ and $Q_0(t, p) = Q_p^{max}$, which is hardly surprising since constant discharge over a whole year is definitely far from any optimal solution. This is different in case of adapting our initial data to the electricity price. We can see a gap between the orange and the other graphs in *iter_1*. Although the decrease in error does not seem much due to the scale, the relative difference between the green and the orange line is more than 27% in *iter_1*. We will state the percentages concerning the error decrease per iteration compared to the basic case of $Q_0(t, p) = 0$ in Table 5.2.

	$Q_0 = Q_p^{max}/10$	$Q_0 = Q_p^{max}$	Q_0 fitted to price
iter_1	1.68%	3.57%	27.33%
iter_2	40.27%	54.24%	60.90%
iter_3	28.56%	54.97%	52.19%
iter_4	1.26%	6.40%	8.90%

Table 5.2 The relative difference between the error values to $Q_0(t, p) = 0$ and the other instances, i.e. for instance in iteration 2 the error in case of $Q_0 = Q_p^{max}$ compared to the basic case of $Q_0(t, p) = 0$ has decreased by 54.24%.

Clearly, referring to the orange graph, we also tried selecting other values for ε_{band} than just $\varepsilon_{band} = 2$. However, we just state this case, since otherwise we did only achieve similar but worse performance. The sensitivity for changes in parameter ε_{band} ranged for the first iteration in the magnitude of 10^4 and for the second in the magnitude of 10^3. In the third and fourth iteration, due to variations in ε_{band}, the error values changed by less than 10^2. In Table 5.1 we notice also that although we get better performance in the first iteration when fitting Q_0 to the NPS price, the improvement fades in higher iterations. In iteration 3, the red

line indicating $Q_0(t,p) = Q_p^{max}$ even yields a smaller error than the orange line. Examining red and orange, we also experience a kind of error barrier at about 10^4, since after *iter_3* the error grows again.

Next, in Figure 5.15 we have a look at the computation times according to the different initial values. Note that the given times are the sum of the time AMPL needs to generate the LP and the time MOSEK needs to solve the LP. In general, we can say that the increase of computational time per iteration coincides with the decrease of the error level - the less the error the more time is used. Only the green graph is not growing monotonically. The second iteration is quite fast compared to the others. This behavior can be explained by a kind of start-up phase that is needed when using $Q_0(t,p) = 0$ in order to get close to a similar error level as other choices of Q_0. That is, in Figure 5.14 we see that the error in *iter_3* for $Q_0(t,p) = 0$ has actually similar magnitude as the red and the orange graph already indicate in *iter_2*. Furthermore, the runtimes according to the green graph in iterations 3 and 4 are of the same magnitude as those concerning the other cases in iterations 2 and 3. Thus, we conclude that due to bad choices of starting values we introduce redundant iterations that can be avoided if we select Q_0 more sophisticatedly.
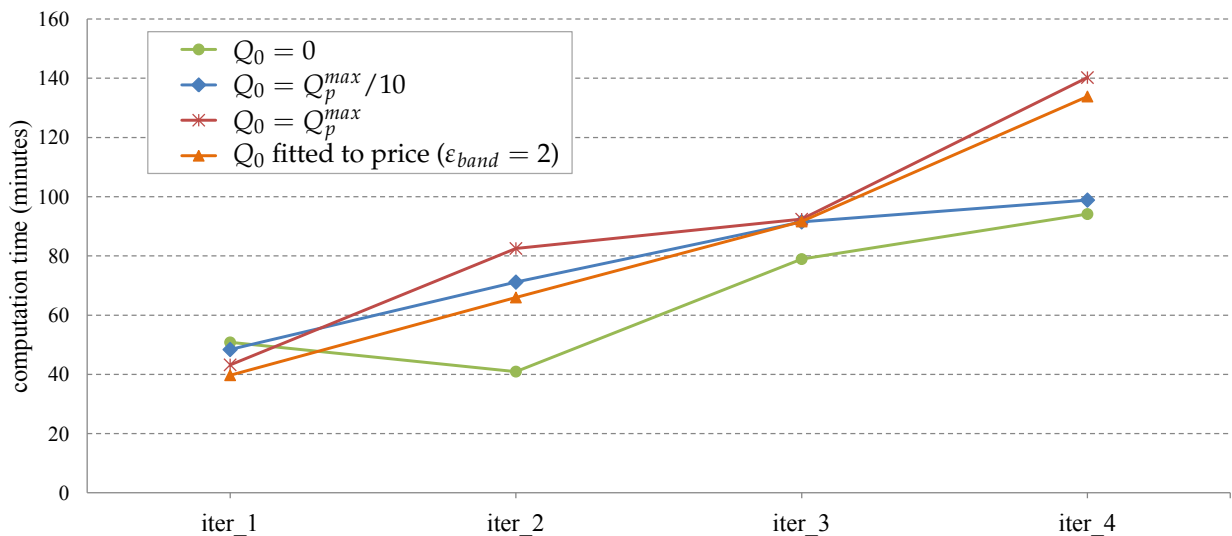

Figure 5.15 Comparison of the computation time per iteration for different choices for the initial values $Q_0(t,p)$. $\Big[$ Simulation period: 01.01.2011 - 01.01.2012, *max_iter* $= 4$ $\Big]$

Referring to our choice to fit Q_0 to the electricity price, we should consider to stop the algorithm after the third iteration, since after that the error stays similar or gets worse[44] and the runtime becomes unnaturally long. Neglecting the fourth iteration yields a total computation time of 197.3 minutes. If we compare this to setting Q_0 constant equal zero, firstly we need four or more iterations to arrive at approximately the same error level (compare

---

[44]In higher iterations (higher than 4) we experienced that the error zig-zags around the already mentioned error barrier at about 10^4.

the green and the orange graph in Figure 5.14) and secondly the total time needed for four iterations is 264.8 minutes.

Hence, if we calculate the difference between both total computation times, we discover that we can save more than one hour of computation time due to a faster error convergence and consequently less required iterations.

Chapter 6

Simulation Results

We conclude this work by stating some basic simulation results from the years 2001 - 2011, regarding our implemented power plant portfolio as presented in the appendix in Figure A.1. Although the main parts of this thesis are actually the numerical experiments in the previous chapter, we want to give the reader an idea of the optimization results we get by the HiREPS model concerning the operation plan of HPLs in South and Central Sweden as well as the annual production and the profits gained by the whole system due to following this optimized mode of operation. In the end, we will also give a prospect of what simulations can be further done concerning the South Sweden model.

To avoid confusion, we clarify that all the model data given in the upcoming figures in this chapter originate from the same simulation. Hence, data with equal dates coincide.

6.1 Production and Profits 2001 - 2011

6.1.1 Electricity Production

We start by giving various simulation results concerning the years 2001 - 2011. At first, we want to see if the model of South and Central Sweden works accurately and if it correctly depicts the reality. Thus, we compare the simulated production with the actual production in those years. Since we do not have any detailed annual production data of our special implemented HPL portfolio, we compare to the total electricity production due to hydropower of all of Sweden. Data regarding that is easily accessible through the Swedish Energy Agency, see e.g. [21]. Then, from the trend in the data concerning whole Sweden, we can deduce if the model results are plausible.

However, we have to be careful when comparing to actual data, since our point of view is different from reality. In reality, we have lots of power plant operators, where supposedly each of them tries to optimize their chain of HPLs. Our model in fact yields the perspective of one global operator that optimizes the whole system of HPLs at once, which of course leads to a different outcome.

For simplicity, we introduce abbreviations for the *annual energy production* (AEP), which simply is the total electricity produced in a specific year, and the *average annual energy production* (avg. AEP), which is the average electricity production based on e.g. the last 10 years.
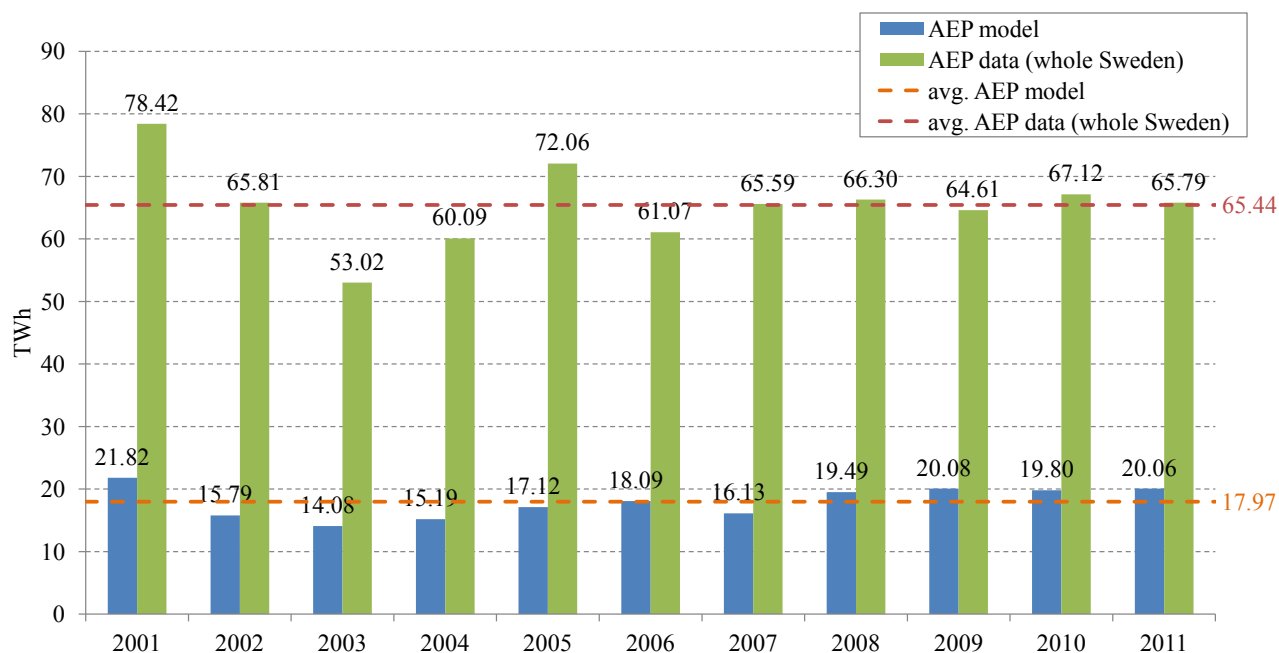

Figure 6.1 Comparison of the total electricity production by hydropower in whole Sweden to the total production values of the model of South and Central Sweden for the years 2001 - 2011. We also indicate the avg. AEG in both cases, where the mean value is formed by incorporating the whole simulation period 2001 - 2011.

We know from Figure 6.1 that the 140 HPLs we considered constitute approximately 27% of the total hyropower output of whole Sweden. Referring to the trend in both AEP data we can argue that they actually coincide quite well. In both cases we encounter high production values in the year 2001, followed by the break in the trend in 2003 and the moderate results after that from 2007 - 2011. According to this overall view, we conclude that the HiREPS model actually does what we expect it to do.

Next, we have a closer look at the AEP of our model. We break down the blue pillars of Figure 6.1 to obtain the specific energy production for different types of hydropower plants. Figure 6.2 considers the base-load production of RR-HPLs and TH-HPLs, the sum of the AEP of all the implemented ST-HPLs as well as the combination of PST-HPLs and YST-HPLs. There is no use in further splitting up for example the YST-HPLs and the PST-HPLs, since in South Sweden there are only two pumped storage power stations installed and also quite few year-storage reservoirs, which are mostly situated in the western, more mountainous areas.

Notably, Figure 6.2 unravels the big importance of RR-HPLs and TH-HPLs in South-Sweden. They make up more about 60% of the energy production whilst storage power plants provide more than 25%. In fact, there is also a considerable amount of small hydropower plants with capacity less than 5 MW in the southern areas. However, in Figure 6.2 we only refer to

HPLs with power ≥ 5 MW.

We also observe that our simulations obtain a higher avg. AEP than the data (red dotted line) indicates. For clarity, we explain the meaning of *avg. AEP data* - we have the avg. AEP to each HPL in the model, thus the red dotted line represents the sum of those average production values. We conclude that through optimizing the whole system at once, we can achieve some improvements in the amount of energy produced.
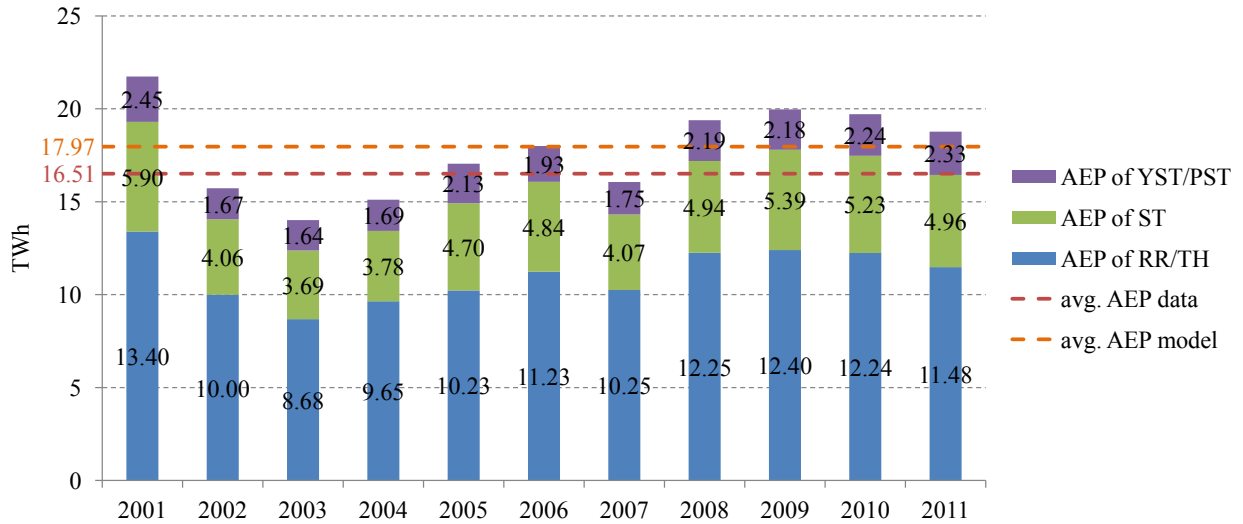

Figure 6.2 AEP of different types of hydropower plants for the years 2001 - 2011. By the orange and the red dotted lines we also show the avg. AEP according to the collected data and the model simulation.

6.1.2 Profits

Besides the production, also the profit which the whole system generates, is of utmost interest. Not only because it is the value of the objective function of our model, but also because profit is one of the big incentives in economy. We recall the objective function specified in 4.2 and the definition of the profits therein:

$$\text{Profits} = \sum_{t \in HOURS} \text{NetEnergy}(t) \cdot \text{NPSPrice}(t). \tag{6.1}$$

From this, it appears that the profit of the system highly depends on the produced energy and the electricity price. Thus, before we take a look at the results, we examine the progression of the electricity price for the years 2001 - 2011 at the Nord Pool Spot market in Figure 6.3. Of course, keeping (6.1) in mind, we expect to see a connection between the curve of the electricity price and the yearly profits.

Comparison of Figure 6.3 and Figure 6.4 yields the anticipated behavior according to these preliminary considerations. We observe price peaks in the years 2003, 2006, 2008 and 2010, which are reflected by higher profits in Figure 6.4. Furthermore, we recognize a slight

Figure 6.3 Electricity price at the Nord Pool Spot stock exchange for the years 2001 - 2011 given in weekly mean values.

upward-trend of the NPS price from left to right, which also reappears in the profit diagram.

In Figure 6.4 we split up again into different HPL types. Remarkable is the fact that comparing the years 2001 - 2005 to later years such as 2010 or 2011, each HPL type was able to more than double the profit made. Putting Figure 6.2 and Figure 6.4 next to each other, we can also see that the influence of the production value on the profit is actually dominated by the electricity price. For instance, high energy production but low prices in 2001 compared to low production but high prices in 2003 yields better performance in the second case.
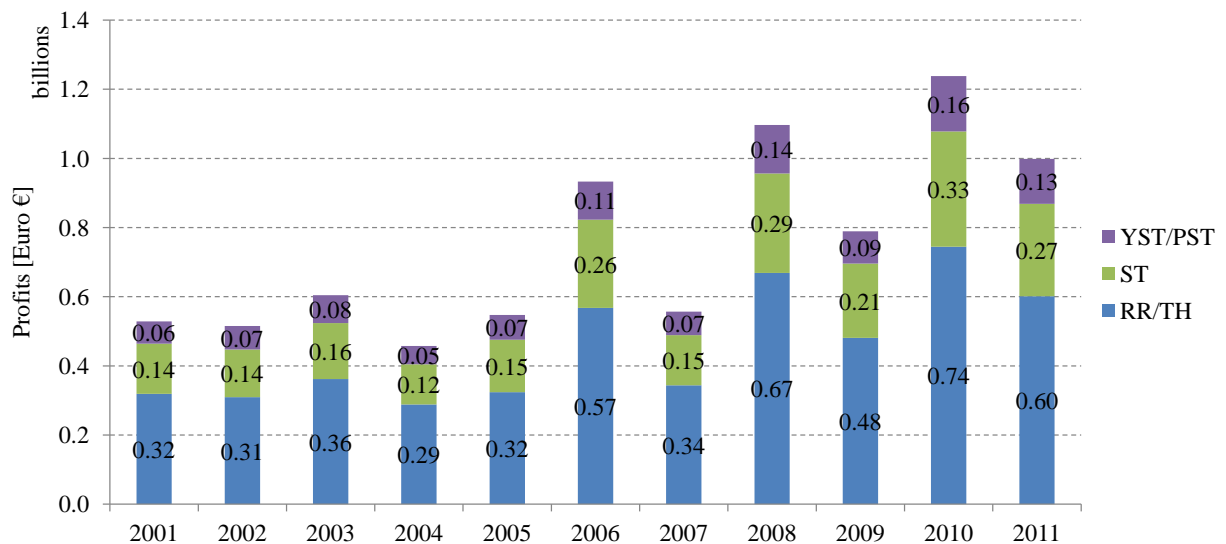

Figure 6.4 Profits gained by different types of hydropower plants for the years 2001-2011. The profits are given in billion €.

6.1.3   River Ljungan

For the sake of providing a detailed view of profits and energy production, we pick one of the rivers in our model and present a profound diagram for a chain of power plants along the river. We choose river Ljungan, the most northern river in our system. Later on, we will also state optimized operation plans of two of Ljungan's HPLs, namely of the uppermost ST-HPL Flåsjö and the TH-HPL Trångfors.

To convey an overall view of the chain of HPLs along the river to the reader, we display a schematic picture of the power stations and the biggest reservoirs:
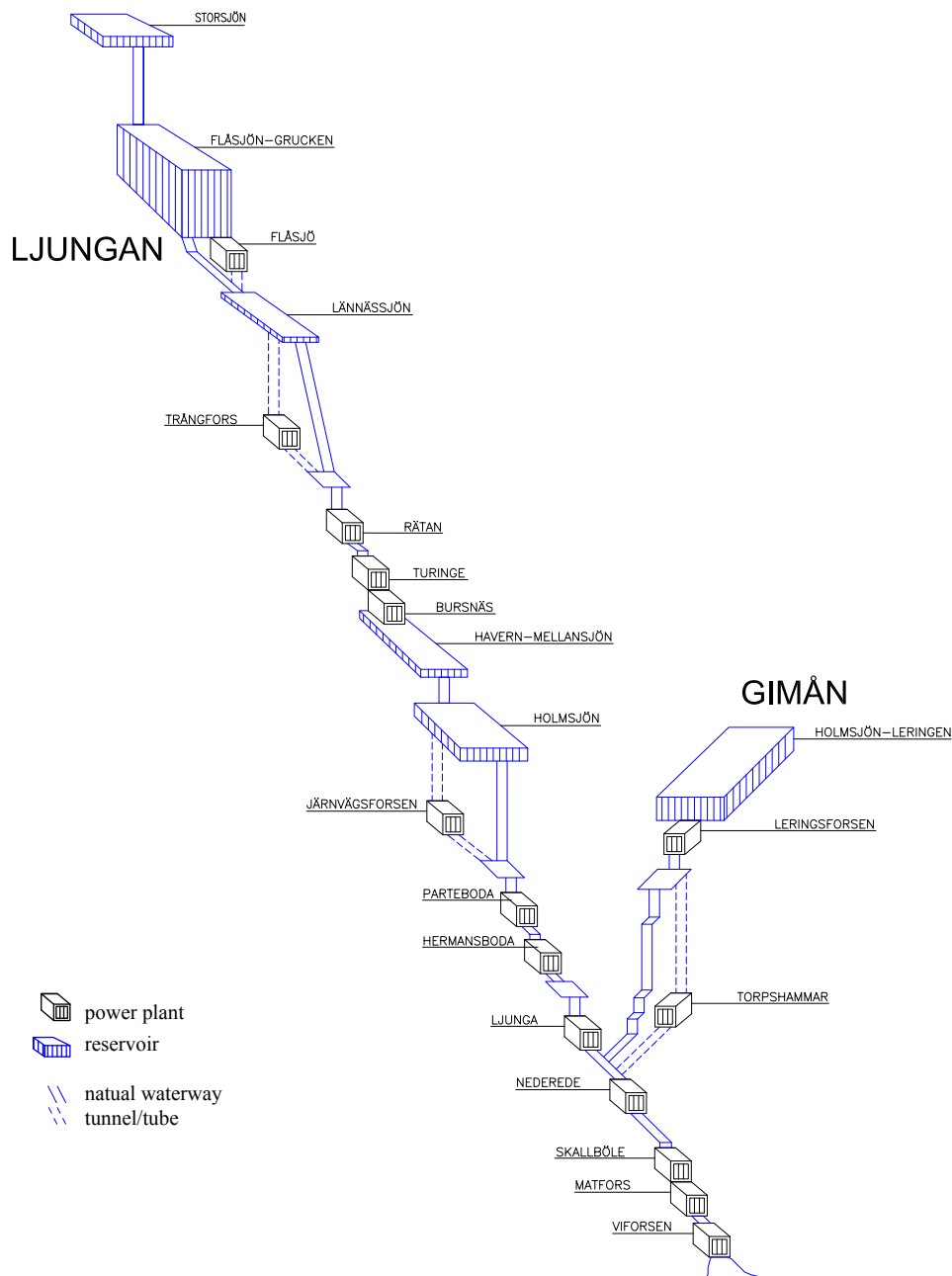

Figure 6.5 Schematic picture of the river Ljungan and its HPLs and reservoirs. (Source: Vattenregleringsföretagen www.vattenreglering.se (19.07.2013))

Figure 6.6 reflects similar data as we can see in Figure 6.4, but in a smaller scale. Concerning

the AEP, we discover that mostly the big ST-HPLs like Järnvägsforsen, Torpshammar or Rättan and some TH-HPLs with quite big reservoirs like Trångfors or Ljunga contribute primarily in years with high production as for instance 2001, 2005 and 2010. Notably, the power station Flåsjö (although having the biggest useable storage volume in this HPL-chain with close to 500 million m^3) is rather irrelevant, which is most likely due to its low turbine capacity and maximal discharge of 20 MW and 60 m^3/s. However, in the next section we will discuss Flåsjö's stowage height during one year, which is interesting since due to its big reservoir it can be used as a year-storage.
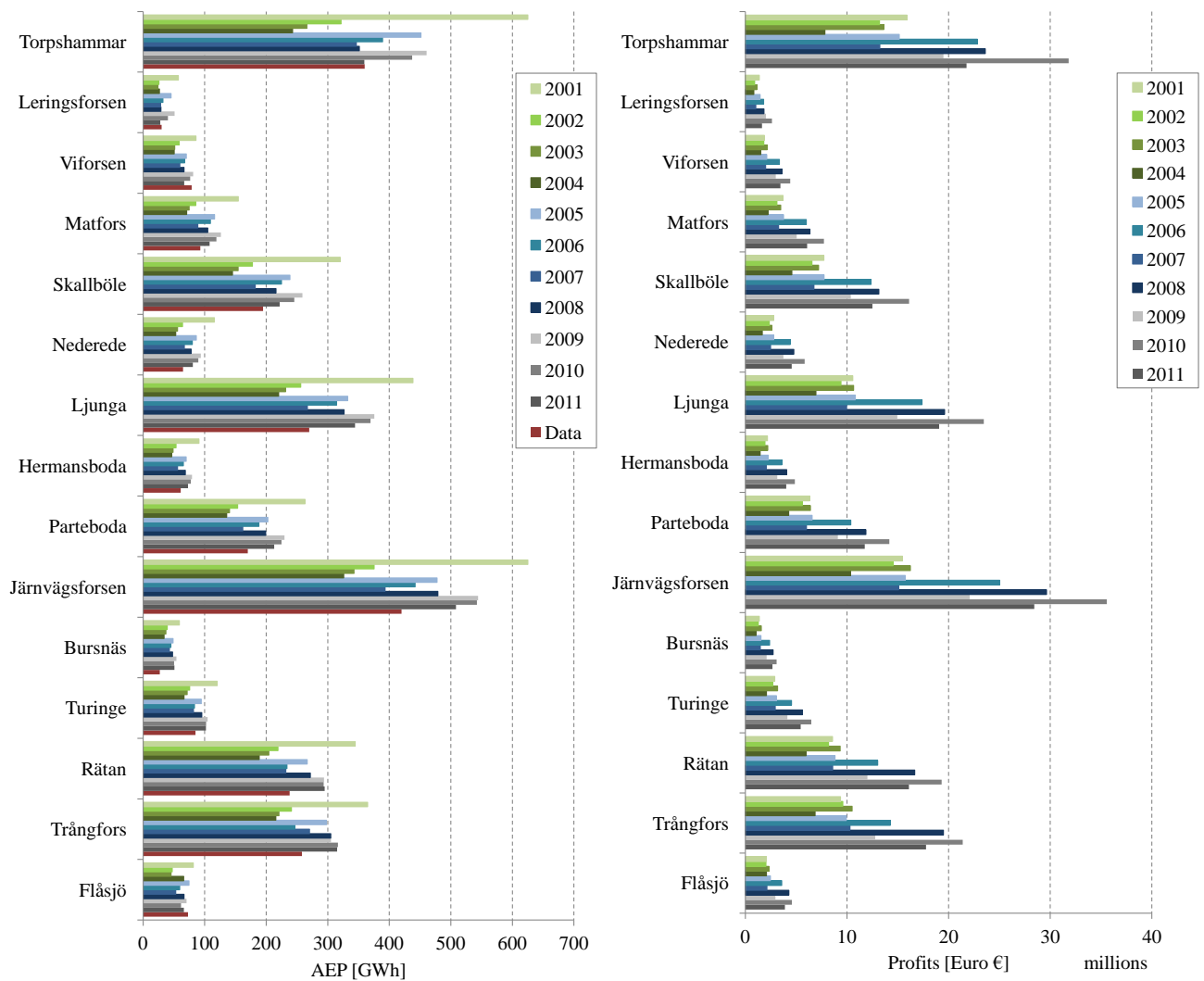

Figure 6.6 Detailed view of profits and electricity production per HPL in the river Ljungan for the years 2001 - 2011. Concerning the production, the data column in red is the actual avg. AEP of each HPL as specified by our collected data.

6.2 Operation Plans of different Types of HPLs

As already mentioned previously, we finish this thesis by stating the simulated operation plans of different types of HPLs, namely of one YST-HPL, one TH-HPL and one PST-HPL.

Of course, we can only cut out a certain segment of the operation mode during one year in order to really get a grasp of the functionality of each type. Otherwise, the sheer load of data would hardly contribute to clarify things.

6.2.1 YST-HPL Flåsjö

At first, we have a look at Figure 6.7, which shows the stowage height in the year 2010 of the extensive reservoir to the Flåsjö power station:
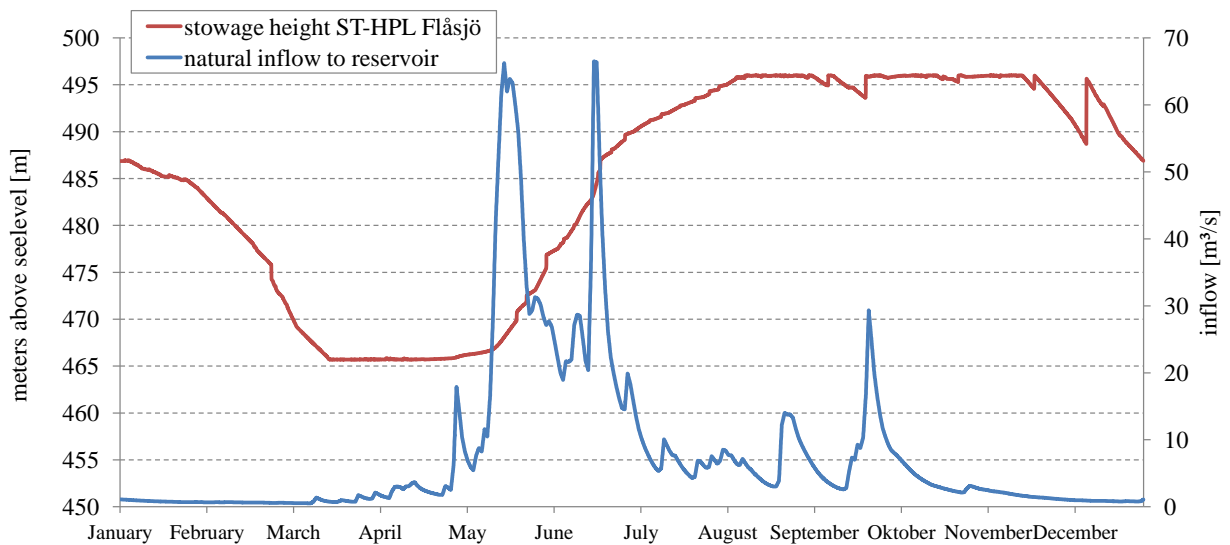

Figure 6.7 Stowage height of the lake Flåsjön during year 2010 and the natural inflow to the reservoir during this time.

For better understanding we note that the top water level of the catchment lake equals 496 m and the drawdown level 465.7 m, so the current water level is allowed to vary in a range of about 30 m. We make out typical characteristics of a year-storage reservoir, since the storage reservoir is drained in the winter months, starting in November/December until March and fills up again in summer from May to September, where the natural inflow due to the thawing period and ordinary precipitation is much higher.

The big jag we see in begin of December is due to the fact that the reservoir of HPL Flåsjö actually consists of two lakes, which are modelled separately.[45] The upper one (Storsjön) with a useable storage volume of about 90 million m^3 is regulated by a dam and the lower one (Flåsjön) has a useable volume of approximately 400 million m^3. Hence, the jag signifies that water is spilled from Storsjön. The reason for this behavior can be found in the NPS price values. Chronologically right after the time the jag occurs, we have an enormous price peak (see end of year 2010 in Figure 6.3). Thus, since the prices are this high, it seems to be advantageous to spill water from the upper reservoir for the use of producing energy instead of saving it.

---

[45]cf. the schematic picture of the river in Figure 6.5

6.2.2 TH-HPL Trångfors

HPL Trångfors is situated in the river right after HPL Flåsjö. The top water level of its dam is 438.7 m and the drawdown level is 436.2 m, leaving us with a regulation amplitude (recall definition in (2.9)) of 2.5 m. Figure 6.8 presents the stowage height of the reservoir in red and the electricity price accordingly. The time period shown is just one week in August, namely the week Mon. 02.08.2010 - Sun. 08.08.2010.

We experience the typical zig-zagging of the stowage height during the week with a linear trend from Monday to Friday to drain the reservoir. This happens due to running the turbine during day time where the price is higher and shutting it off at night time, where the NPS price collapses to a local minimum. On the weekends there happens to be less demand and consequently slightly lower prices. Hence, the reservoir is being refilled for the upcoming week.
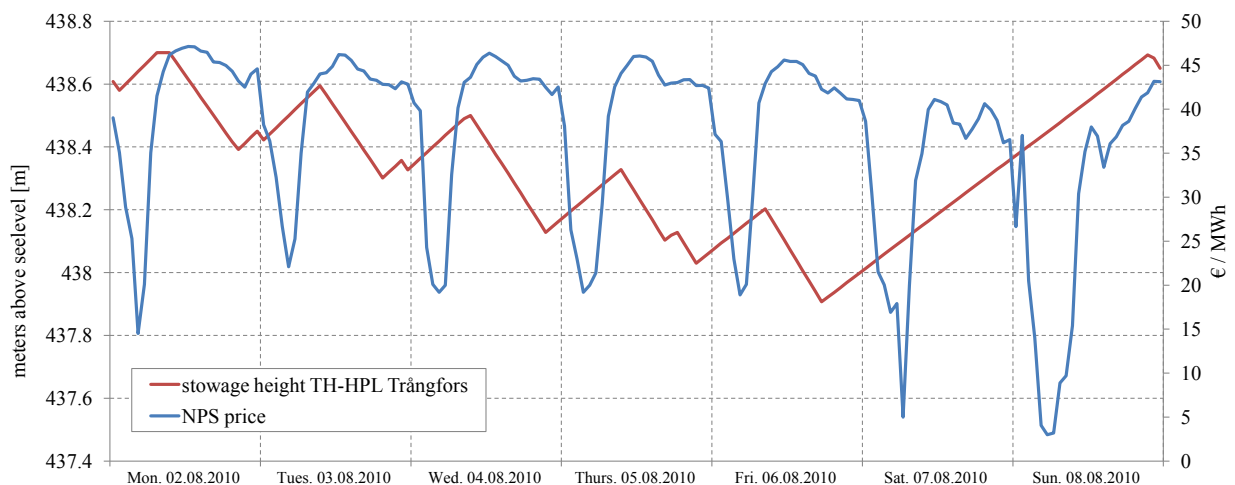

Figure 6.8 Stowage height of the reservoir to HPL Trångfors compared to the electricity price during one week of August 2010.

6.2.3 PST-HPL Kymmen

Last, we examine the typical functionality of a pumped storage power station. Kymmen is one of the two existing PST-HPLs and is situated at the river Rottnan, which is a tributary to the biggest lake in Sweden, lake Vättern. Figure 6.9 indicates the setting of the whole facility, showing all significant reservoirs (blue pins) in the area near the power station (red pin).

So, the HPL turbinates and pumps between the two lakes Kymmen and Rottnen, where the three lakes Skallbergssjön, Lilla Gransjön and Stora Gransjön tribute via a pipe to lake Kymmen.

Concerning the operation mode, we note that Figure 6.10 is a little bit overly packed with data. Nonetheless, every graph shown is interesting to examine in combination with all the others. As in case of the TH-HPL, we show again the stowage height (violet) along with the electricity price (blue). But this time, we also add graphs that display the power of the
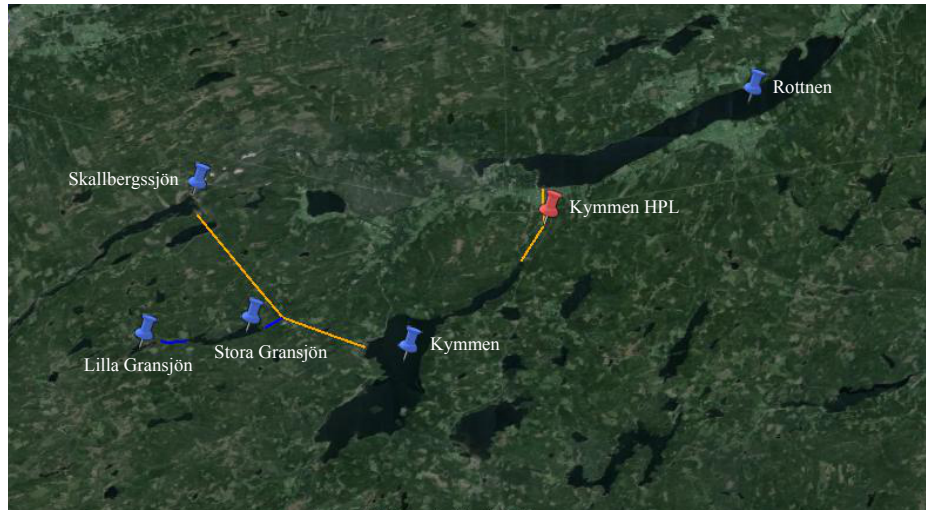
Figure 6.9 Satellite picture of HPL Kymmen and the surrounding reservoirs belonging to it. Orange lines indicate canals or tubes. (Source: Google Earth)

turbines (green) and pumps (red) per time step.

Figure 6.10 Operation plan of the PST-HPL Kymmen during one week in September 2011. The left y-axis actually has two axes overlapping - one for the NPS price and one for the power of the turbines/pumps. The right y-axis indicates the values of the stowage height.

Note that the turbines and the pumps do not work simultaneously, which derives from the variation of the electricity prices.[46] At night we have low prices and thus perfect conditions for pumping and at day time we have higher prices due to greater demand and thus we turbinate. Again, we also find the zig-zagging of the stowage height, which is perfectly the behavior we expected.

---

[46]In fact, this is quite astonishing, since our model does not prevent turbines and pumps to run at the same time. However, the optimal solution to the LP behaves as if we had prohibited simultaneous turbining and pumping.

6.3   Prospect and further Simulations

As already mentioned, the main part of this thesis lies within Chapter 5. Still, we gave a short insight in the simulation results we obtained by the HiREPS model. This section is now dedicated to the question what could be done further, using this model in the context of Sweden. The incentives to complete the model for whole Sweden are still ongoing. Thus, one of the future projects could involve examining all of Sweden at once.

In this thesis, we also did not go into simulations that permit endogenous upgrades of the power stations installed, which is also one of the features of the HiPRES hydro model as indicated in the input-output diagram in Figure 4.1. Investigation of results regarding this concept can probably lead to promising ideas for further advancement in South Sweden's HPL portfolio.

What already has been done in the underlying research project to this thesis (see Section 1.1), could of course also be conducted in case of South Sweden or whole Sweden. Namely, since we have already seen in Section 6.1 that our model indeed sufficiently simulates the reality, prospective simulations could establish production and profit prognosis' of the future. Moreover, by implementation of the electricity production due to different types (nuclear power, wind power, ...) and demand data, there would arise the possibility to simulate Sweden's total future electricity system. Clearly, in such a case we have to come up with certain meaningful assumptions concerning the electricity price and the side inflow as well as demand data, which definitely is not trivial.

Appendix A

Data Research

This appendix clarifies where we gathered the model data from and it yields a quick overview which HPLs, dams and rivers we consider in the model.

To get good results in the simulation, one has to be very careful and as precise as possible regarding the input data. This is actually the most crucial part in implementing a hydropower model, since the results are highly dependent on the input parameters. Thus, concerning non reliable sources, one always has to cross-reference with other sources and sometimes also rely on one's own judgement to make sure the data given is correct and not outdated or inaccurate.

Clearly, some information cannot be found on open resources. Most companies only publish some basic data of their HPLs online or in their annual reports, like capacity, average annual energy production or gross head. More detailed data is often kept under closure due to reasons of economical competition and data security policies.

In some cases we also had to just use estimates based on observations, context and similarities to other HPLs or reservoirs. Simple approximative calculations also helped in filling gaps in the data matrix. For example, considering a shallow reservoir, we can simplify by assuming a rectangular cross-sectional area. Hence, if we do not have the data to say the barycenter of the reservoir, we can calculate an estimate to it through

$$h_b = \frac{h_{max} - h_{min}}{2},$$ (A.1)

where h_{max} and h_{min} are the top water level and the drawdown level, respectively.

A.1 Data Resources

A.1.1 Power plants, Dams, Reservoirs

The identification of different power plants was performed by use of *Google Earth*[47]. First, we should mention that we just collected data to HPLs with a capacity $\geq 5MW$. To be pre-

---

[47]see www.google.com/earth/index.html (05.06.2013)

cise, in some cases we summarized smaller HPLs with low capacities to one HPL, where the circumstances made it possible not to distort the reality too much. That is, if there are multiple HPLs in between two reservoirs with similar heads and the same year of construction[48], or if there are some small HPLs in sequence that add up to an HPL with capacity $\geq 5MW$ without making too much error.

In Figure A.1 one can see a Google Earth picture of all the power plants that are implemented in the Sweden model. The total amount of HPLs in the map is 140. The northernmost river considered is river Ljungan. In the south we have the smaller rivers Mörrumsån, Helgeån, Lagan, Nissan, Ätran and Viskan. In between we included the main rivers Ljusnan, Dalälven, Göta Älv and Kolbäcksån and their tributaries.
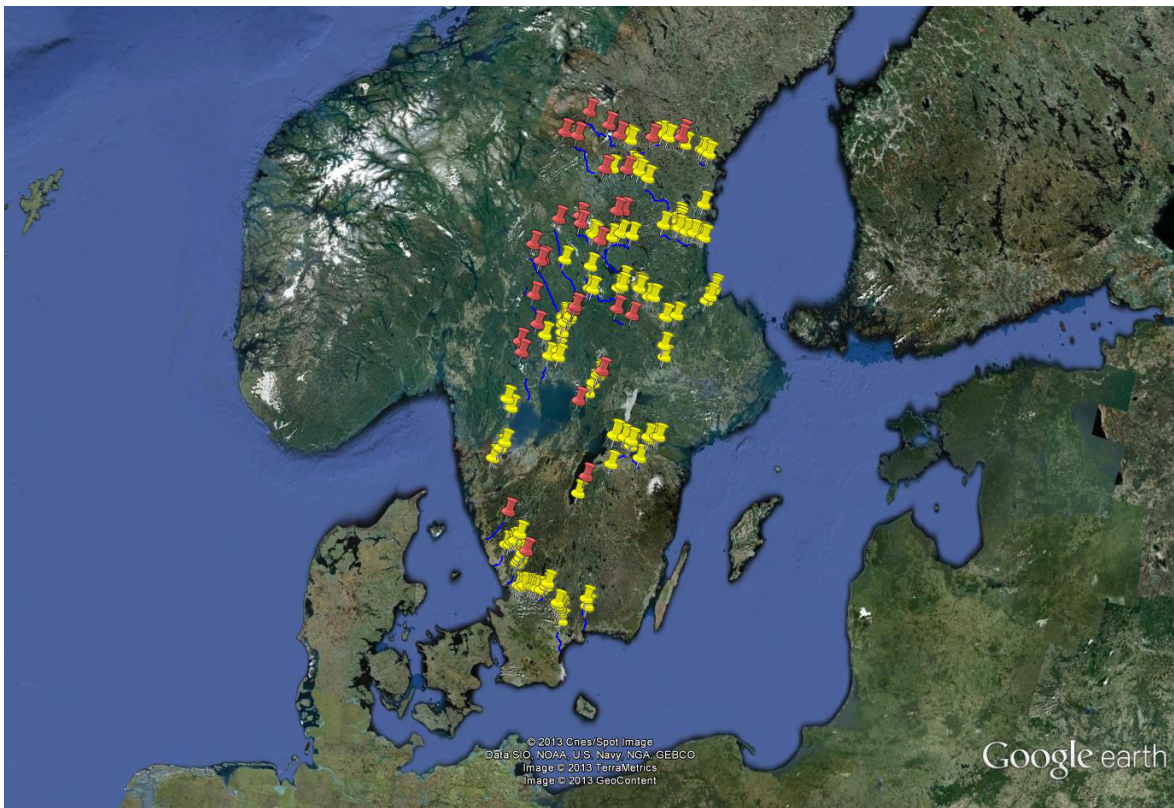

Figure A.1 Map of central and southern Sweden with implemented HPLs and dams. The yellow pins indicate RR-HPLs or TH-HPLs and the red pins show bigger ST-HPLs and PST-HPLs. (Source: Google Earth)

Further information was taken from

- several websites of HPL operators and their annual reports
- ICOLD database (Europe)[49]

  ICOLD is short for *International Commission on Large Dams*. A worldwide database for dams and power plants with a lot of specific information. However, there are some blank gaps that need to be filled.

---

[48]For clarity, the model does not distinguish between construction years before 2000, i.e. all years of construction before 2000 count as the same year, since we only have inflow data and price data starting from 2000.

[49]see www.icold-cigb.net (05.06.2013)

A.1.2 Hydrology

Concerning hydrological data, we especially need side inflow data. The *Swedish Meteorological and Hydrological Institute* (SMHI)[50] provides an online interactive model[51] called S-HYPE, short for *Hydrological Predictions for the Environment - Sweden*. It presents amongst other things "daily simulations of the discharge for 37 787 subbasins defined in the *Swedish Water Archive* (SVAR) version 2010:2".[52] We used the following setup during data collection:

HYPE Modelsetup version:	S-HYPE2010_version_1_0_2
HYPE version:	HYPE_version_3_5_3
Coastal zone version:	Kustzonsmodellen_SVAR_2010_2
Simulation start time:	1960-01-01
SVAR version:	SVAR_2010_2

The HYPE model gives daily discharge values for a specific subbasin at a specific point. Supposed we have the discharge $Q(t, \delta_b)$, $t \in DAYS$ at dam δ_b. Since we do not want the total discharge but the side inflow to dam δ_b, we have to calculate

$$S_{daily}(t, \delta_b) = Q(t, \delta_b) - Q(t, \delta_a), \, t \in DAYS,$$

where δ_a is the dam directly above δ_b. Hence, we have to subtract the discharge values of one lower dam from one upper dam to get the side inflow in between the two dams.

A.1.3 Electricity prices

As already mentioned in Chapter 4, the hourly electricity prices are given in €/MWh and reflect the prices at the day-ahead trading market. The source is Nord Pool Spot[53], the Nordic stock exchange operator.

---

[50]see www.smhi.se (05.06.2013)
[51]see vattenweb.smhi.se/ (05.06.2013)
[52]see www.smhi.se/en/Research/Research-departments/Hydrology/hype-in-sweden-s-hype-1.7891 (05.06.2013)
[53]see www.nordpoolspot.com (20.07.2013)

References

[1] E. D. ANDERSEN, *The homogeneous and self-dual model and algorithm for linear optimization. MOSEK Technical report: TR-1-2009*, tech. report, MOSEK ApS, April 2009.

[2] E. D. ANDERSEN AND K. D. ANDERSEN, *Presolving in linear programming*, Mathematical Programming, 71 (1995), pp. 221–245.

[3] ——, *The MOSEK Interior Point Optimizer for Linear Programming: An Implementation of the Homogeneous Algorithm*, High Performance Optimization, 33 (2000), pp. 197–232.

[4] S. ANGELIN, L. BERNELL, ET AL., *Hydro Power in Sweden*, The Swedish Power Association and The Swedish State Power Board, Stockholm, Sweden, 1987. written in co-operation by engineers from The Swedish Power Association, The Swedish State Power Board, VATTENFALL, Skanska Cementgjuteriet, SKANSKA, VBB.

[5] G. K. BATCHELOR, *An Introduction to Fluid Dynamics*, Cambridge University Press, 2000.

[6] R. FLETCHER, *Practical Methods of Optimization*, Wiley-Interscience, New York, 2nd ed., 1987.

[7] R. FOURER, D. M. GAY, AND B. W. KERNIGHAN, *AMPL: A Modelling Language for Mathematical Programming*, Brooks/Cole Cengage Learning, 2nd ed., November 2002.

[8] J. GIESECKE, E. MOSONYI, AND S. HEIMERL, *Wasserkraftanlagen: Planung, Bau und Betrieb*, Springer, 5th ed., 2009.

[9] A. J. GOLDMAN AND A. W. TUCKER, *Theory of Linear Programming*, Linear Inequalities and Related Systems, 38 (1956), pp. 53–97.

[10] I. KLEE, *The Electricity Market in Sweden and the Role of Svenska Kraftnät*, Svenska Kraftnät, May 2011. brochure, www.svk.se/Global/10_Press_Info_eng/Pdf/Densvenskaelmarknaden_GB.pdf (08.05.2013).

[11] MOSEK APS, *The MOSEK optimization tools manual. Version 6.0 (Revision 148)*, Fruebjergvej 3, 2100 Copenhagen, Denmark, 2012.

[12] R. MULLEY, *Flow of Industrial Fluids: Theory and Equations*, CRC PressINC, 2004.

[13] J. NOCEDAL AND S. J. WRIGHT, *Numerical Optimization*, Springer Science+ Business Media, 2006.

[14] NORD POOL SPOT AS, *The Nordic Blueprint - Annual Report 2011*, 2012. annual report, www.nordpoolspot.com/Global/Download%20Center/Annual-report/annual-report_Nord-Pool-Spot_2011.pdf (10.05.2013).

[15] C. ROOS, T. TERLAKY, AND J. VIAL, *Theory and Algorithms for Linear Optimization: An Interior Point Approach*, John Wiley & Sons, Chichester, New York, 1997.

[16] J. STOER AND R. BULIRSCH, *Introduction to Numerical Analysis*, Texts in Applied Mathematics, Springer, 2002.

[17] SVENSKA KRAFTNÄT, *Handbook for the Swedish Electricity Market - Routines and exchange of information for trading and settlement*, May 2005. brochure, www.svk.se/Global/10_Press_Info_eng/Pdf/EMHB_eng_ut.pdf (08.05.2013).

[18] SVENSKA KRAFTNÄT, SVENSK ENERGI AND OBEROENDE ELHANDLARE, *Svensk Elmarknadshandbok*, 13a ed., April 2013. brochure, www.svenskenergi.se/elmarknadshandboken/Dokumentation/Texter/NEMHB.pdf (08.05.2013).

[19] SWEDISH COMPETITION AUTHORITY, *Deregulation of the Swedish Electricity Market*, November 1996. report, www.kkv.se/upload/Filer/ENG/Publications/rap_1996-3eng.pdf (08.05.2013).

[20] SWEDISH ENERGY AGENCY, *Energy in Sweden 2012*, 2013. annual report, www.energimyndigheten.se/Global/Engelska/Factsandfigures/Energy_in_sweden_2012.pdf (08.05.2013).

[21] ——, *Energy in Sweden 2012 facts and figures*, 2013. annual report, www.energimyndigheten.se/Global/Engelska/Factsandfigures/EnergyinSwedenfactsandfigures2012(2).pdf (08.05.2013).

[22] A. W. TUCKER, *Dual Systems of Homogeneous Linear Relations*, Linear Inequalities and Related Systems, 38 (1956), pp. 3–18.

[23] C. C. WARNICK, *Hydropower Engineering*, Prentice-Hall Incorporated, Englewood Cliffs, NJ, USA, 1984.

[24] S. J. WRIGHT, *Primal-Dual Interior-Point Methods*, vol. 54, Society for Industrial and Applied Mathematics, 1987.

[25] X. XU, P.-F. HUNG, AND Y. YE, *A simplified homogeneous and self-dual linear programming algorithm and its implementation*, Annals of Operations Research, 62 (1996), pp. 151–171.

[26] Y. YE, M. J. TODD, AND S. MIZUNO, *An $O(\sqrt{n}L)$-iteration homogeneous and self-dual linear programming algorithm*, Mathematics of Operations Research, (1994), pp. 53–67.