Dynamic Load Balancing of a Power System Portfolio

Edlund, Kristian Skjoldborg

Publication date:
2010

Document Version
Accepted author manuscript, peer reviewed version

Kristian Edlund

Dynamic Load Balancing
of a Power System Portfolio

Dynamic Load Balancing of a Power System Portfolio
Ph.D. thesis

ISBN: 978-87-92328-34-2
July 2010

Contents

Preface and Acknowledgements

The work presented in this thesis has been carried out under the Industrial PhD programme supported by the Danish Ministry of Science, Technology and Innovation. The thesis is submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy at Section for Automation and Control, Department of Electronic Systems, Aalborg University, Denmark. The work has been carried out at Department for Modelling and Optimisation at DONG Energy and at the Section for Automation and Control, Aalborg University in the period April 2007 to April 2010 under supervision of Associate Professor Jan Dimon Bendtsen, Manager of Modelling and Optimisation Tommy Mølbak, Associate Professor Palle Andersen and engineer from Production Optimisation Jan H. Mortensen.

I would like to thank all my supervisors for their invaluable support and guidance throughout the project. A special thank you to Jan Dimon Bendtsen and Palle Andersen for spending hours on academic discussions throughout the project and to Tommy Mølbak for helping me understand the process, both in terms of the power system process, but especially the process of being a PhD student.

My colleagues at both DONG Energy and Aalborg University all deserve a mention here as well, especially Simon Børrensen and Brian Astrup with whom I have discussed endless ideas with during their work on the current version of the controller.

Thank you to Associate Professor John Bagterp Jørgensen for letting me visit Department of Informatics and Mathematical Modelling at Technical University of Denmark (DTU) and for giving me many hours of valuable sparring for my research. I would also like to thank Leo Emil Sokoler whom I had the pleasure of supervising while staying at DTU Informatics. He made significant contributions to the development of efficient optmisation algorithms.

Kristian Edlund
Fredericia, April 2010

Abstract

With the recent (and ongoing) liberalisation of the energy market, increasing fuel prices, and increasing political pressure toward the introduction of more sustainable energy into the market, dynamic control of power plants is becoming highly important. More than ever, power companies must be able to adapt their production to uncontrollable fluctuations in consumer demands as well as in the availability of production resources, e.g. wind power, at a short notice.

Currently, thermal power plants in Denmark provide the necessary flexibility, which is coordinated by a load balancing controller. As the stochastic production increases, the flexibility of the power system should be increased as well. A proposal for increasing flexibility is virtual power plants (VPP). The concept of VPP is to pool smaller units together to obtain a larger unit which offers the flexibility known from thermal power plants. A virtual power plant could consist of heat pumps and electrical vehicles which has some flexibility that can be utilised. Creating such virtual power plants will increase the number of units the load balancing controller coordinates, and this will strain the design method of the current load balancing controller.

This thesis presents a new method for designing a load balancing controller which is flexible and scalable in the number of units to meet the requirement of the future power system. The developed method is based on model predictive control. In order to achieve flexibility in the controller, the method presented in this thesis utilises a two-layer hierarchical control structure using an object-oriented design. The object-oriented structure is designed so units can be added, removed and modified without redesigning the whole controller. Furthermore, the design allows freedom in the implementation of the unit in question, in order to meet the diversity of the future units.

The optimisation problem arising from the construction of the model predictive controller has been fitted into the hierarchical structure by decomposing it using Dantzig-Wolfe decomposition. Besides the benefits of the flexibility by solving the optimisation problem within the hierarchical structure, this decomposition also ensures efficient solving of the problem, thus allowing the controller to coordinate more units.

The newly developed design method has been utilised for synthesis of a controller for the current portfolio and compared to the performance of the current portfolio controller through simulations. Through simulations on a real scenario the new controller shows improvements in ability to track reference production and economic performance.

Synopsis

Den nylige (og igangværende) liberalisering af elmarkedet, stigende brændselspriser og øget politisk pres for at indføre mere vedvarende energi i markedet har gjort dynamisk regulering af kraftværker til et vigtigt emne. Elselskaberne skal i højere grad end tidligere være i stand til med kort varsel at tilpasse produktionen til de ukontrollerbare udsving i forbrugernes efterspørgsel samt tilgængeligheden af produktionsressourcer, f.eks. vindkraft.

Det er i øjeblikket de termiske kraftværker, der leverer den nødvendige fleksibilitet, koordineret af en balanceregulator. Når den stokastiske produktion øges, er der et behov for at øge fleksibiliteten. Et forslag til hvordan øget fleksibilitet kan opnås, er virtuelle kraftværker (VPP). Konceptet består i at samle mange små enheder med en smule fleksibilitet til en større enhed, som kan give samme fleksibilitet, som kendes fra de termiske kraftværker. Et par eksempler på sådanne enheder er varmepumper og elbiler. Selv om konceptet i et virtuelt kraftværk er at aggregere mange små enheder, må det stadig forventes, at de medfører en kraftig stigning i antallet af enheder, som balanceregulatoreren skal koordinere. Dette er mere, end den nuværende regulator kan håndtere.

Denne afhandling præsenterer en ny metode til at designe balanceregulatorer, som er fleksible og skalerer til mange enheder for at imødekomme de krav, som fremtidens energisystem stiller. Den udviklede metodik er baseret på en model prædiktiv reguleringsstrategi. For at opnå den ønskede fleksibilitet i regulatoren, udnytter den præsenterede metode sig af en objektorienteret to-lags hierarkisk regulatorstruktur. Den objektorienterede struktur er konstrueret, så enheder kan tilføjes, fjernes og ændres, uden at den grundlæggende struktur i regulatoren ændres. Endvidere er designet udformet, så det giver størst mulig frihed til at udforme den enkelte enhed for at imødekomme den mangfoldighed af forskellige enheder, der kommer i fremtiden.

Det underliggende optimeringsproblem, som udspringer af den modelprædiktive regulator, er blevet indpasset i den hierarkiske struktur ved at benytte Dantzig-Wolfe dekomposition. Dekomposition giver ud over at kunne indpasse løsningen af optimeringsproblemet i den hierarkiske struktur, en mere effektiv løsning af problemet, hvilket medfører, at regulatoren kan koordinere flere enheder.

Den udviklede design metode er anvendt til at syntetisere en regulator til den nuværende portefølje af kraftværker. Den nye regulator er sammenlignet med den nuværende regulator via simuleringer med rigtige produktionsdata. Simuleringerne viser en forbedring af evnen til at følge referencer og en forbedret økonomi.

1 | Introduction

This thesis is concerned with developing a method for controller design for dynamic load balancing of a portfolio consisting of multiple units connected to one common power system. The goal is to use the current operational experience to develop a new method in order to create a controller with a more modular structure which is ready to meet the future challenges that the power system will bring with the current focus on developing a sustainable energy production.

The chapter gives the motivation for developing a new method, a description of the power system as well as state of the art within power system control and the underlying theory the method utilises.

1.1 Motivation

This research project was proposed and funded by DONG Energy [DONG Energy, 2010]. DONG Energy is the largest Danish power producer with more than 4500 employees and 5500 MW installed capacity of thermal power and 654 MW of wind power in Denmark. Besides, DONG Energy has activities in most countries in Northern Europe where the focus is on development of renewable energy projects. Besides the activities in power generation, DONG Energy is active within oil and gas exploration and production as well as distribution of both gas and electricity.

Even though DONG Energy is considered a small company compared to the tycoons in the area of power generation, there has been a tradition for designing, constructing and operating the most fuel efficient thermal power plants in the world as well as a massive practical experience with wind power projects.

The massive investment in wind technology driven by the Danish Government has resulted in 30% of the installed capacity in the Danish power system comes from wind turbines in 2007, with visions to expand even further. More wind integrated in the system increases the demand for the power production by exisiting thermal units to be flexible as well as the coordination between thermal power and wind [Weber et al., 2006; Banakar et al., 2008].

The Danish system began as a monopolised system with generation based on fossil fuels. A system with a reasonable predictable production and consumption, and only slow changes in the power exchange with other regions. The development has been towards decentralisation and liberalisation along with a political incentive to introduce more renewable energy in the system which is often stochastic production such as wind turbines. In Denmark the goal is to increase the share of electrical energy from renewable sources

from 24% in 2005 to 36% in 2025 as found in [Danish Ministry of Transport and Energy, 2005].

In 2003 Energinet.dk the Danish Transmission System Operator (TSO) started constructing a controller to maintain the balance between consumption and production in Denmark. This led to the fact that DONG Energy had to design a controller which could communicate with the TSO and distribute the set point received from the TSO to the thermal power plants. This requirement was expanded within DONG Energy to include a better coordination of the power plants to minimise the deviations between the actual and sold production within the portfolio. This event was the kickoff the load balancing controller within DONG Energy.

The controller started out as an excellent idea implemented as a prototype and has proved to work well in practice. However, many years of incremental design has led to a structure which is no longer simple and easy to maintain. The purpose of this project is to take a step back and rethink the design principles for the controller in order to get an easier maintainable controller, and a controller which can cope with the challenges the future of the power system is likely to bring.

Since the PhD project started, DONG Energy has formulated a strategy called 85/15, meaning that 85% of the power production should come from carbon dioxide neutral sources within the lifespan of of generation. This is a very challenging vision. There is no grand solution where change of one technology will solve this challenge, it relies on multiple different techonologies, all cooporating to achieve this goal. An important step towards this vision is to create a flexible system such that the production and consumption can be changed depending on the resources available, such as wind.

One of the candidates for creating flexibility is Virtual Power Plants (VPP). The concept pools several, otherwise too small, production and consumption units, such as multiple smaller power plants, wind turbines and heat pumps, and make them behave as one unit providing yet another means of load balancing. If the VPP concept proves successful, an enormous amount of possibilities for load balancing becomes available, and thus increasing the importance of this project, rethinking the current load balancing controller structure to obtain a more flexible a scalable controller.

Electric vehicles is another topic which catches much attention. The electrical vehicles will introduce an additional demand for electricity, but the charging of the vehicles can be controlled thus providing an additional VPP.

This project has the objective to develop a controller design method for the next generation of load balancing controllers. In order to investigate this objective, the following hypothesis is formulated

Hypothesis: *It is possible to develop a controller design method which can be utilised to synthesise a controller which fulfils the criteria:*

> ***Scalability*** *The controller must be scalable in the number of units participating in the load balancing control.*
>
> ***Flexibility*** *The controller must be flexible, such that addition of new units and maintenance of existing ones is possible.*
>
> ***Performance*** *The controller must perform at least as well as the current controller measured on some performance criteria.*

1.2 Power Systems Control and Electricity Market

The largest of the European grids both in area but also in volume with a production capacity of 3000 GW is ENTSO-E RG Continental Europe [ENTSO-E, 2010a]. The electrical grid covers the continent of Europe, from Portugal in the west to Romania in the east. Since electricity cannot be stored for later use, there is a constant need to outbalance the consumption and the production to supply the consumers. In order to keep the balance within an area as big as ENTSO-E RG Continental Europe it is split into several regions where each region of the grid is governed by a Transmission System Operator (TSO). Western Denmark, meaning Jutland and Funen, is one region within the ENTSO-E RG Continental Europe area and is synchronous interconnected to Germany and asynchronously connected to Norway and Sweden. The area along with major production units is shown in Figure 1.1. This region is governed by the Danish TSO Energinet.dk.

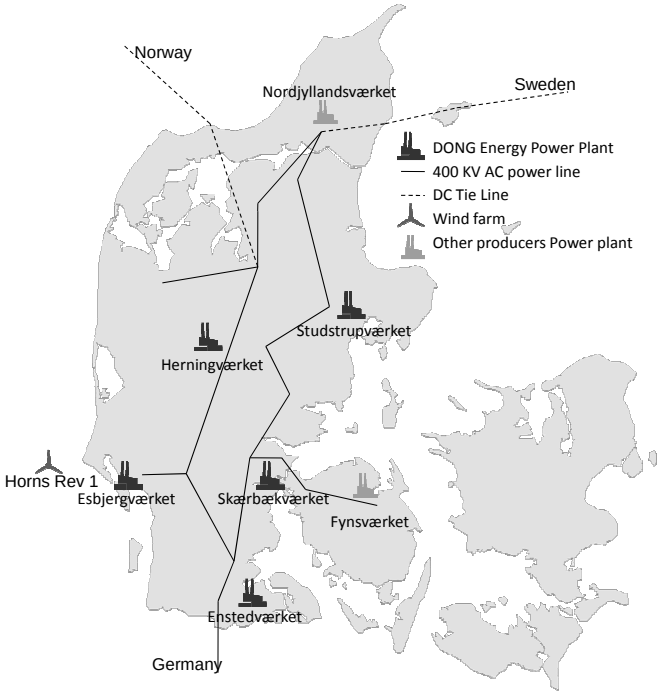


Figure 1.1: The main components of the power system in the western Denmark.

In western Denmark there are 7 sites containing large power plants covering a total of 9 units with an electrical production capacity ranging from 80 MW to 650 MW, where the most common size is around 400 MW. There are two major producers in western Denmark where DONG Energy is the largest and operates a total of 6 units in the area.

Sealand on the other hand is not part of the ENTSO-E RG Continental Europe area, instead it is synchronous interconnected with ENTSO-E RG Nordic which covers most of Scandinavia.

The electricity grid balance between consumption and production have to be main-

tained at all times. All the rotating devices connected to the grid, such as generators, have some energy and thus gives a bit of leeway to maintain the balance. If the consumption is larger than the production, energy will be pulled out of the system, making the generators slow down from the usual 50Hz, and thus a drop in the system frequency can be observed.

In order to keep balance between production and consumption, DONG Energy uses a multi hierarchical scheme as shown in Figure 1.2.

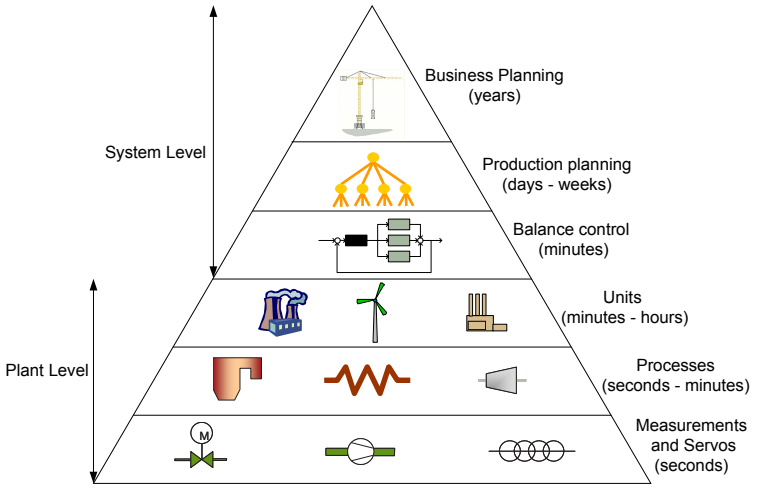


Figure 1.2: System hierarchy within DONG Energy. The hierarchy consist of a system level which coordinates the units, and a unit level that contains the control hierarchy of the individual unit. The time units on the figure show the typical time scale on which the level operates.

The upper three levels of the hierarchy are denoted *system level*, meaning that the scope of these levels covers multiple power producing units. On the highest level is the business planning where decisions on building new power plants is taken. It might not seem obvious to include this level when discussing balance between consumption and production, but the investment decision is based on the need for the capacity. During planning and construction, balance control is an essential part of the power plant design.

The next level is production planning also known as *unit commitment*. Production planning is static optimisation of load distribution among power production units, [Padhy, 2004], [Salam, 2007]. Solving the unit commitment problem means determining the combination of available generating units and scheduling their respective output to satisfy the reference production, often with a minimisation of cost under the operating constraints enforced by the power producing portfolio for a specific time - typically from 24 hours up to a week. The optimisation problem is of high dimension and combinatorial in nature, and can thus be difficult to solve in practice. Results using heuristic methods [Johnson et al., 1971], [Viana et al., 2001], Mixed Integer Programming [Dillon et al., 1978], [Jørgensen et al., 2006], Dynamic Programming [Ayuob and Patton, 1971] and Lagrangian Relaxation [Aoki et al., 1987], [Shahidehpour and Tong, 1992], have been reported in literature.

Once a solution to the unit comment problem, i.e. a static schedule has been found, the

load plans are distributed to the generating units. Each unit is responsible for following its load plan and must handle disturbances etc. locally, implying the necessity of local power plant controllers, wind farm controllers etc., which is shown as the lower three levels of the hierarchy.

The lowest system level is the balance control level. Due to deviations between the predicted and actual consumption as well as fluctuations in production, this level is added to give a dynamic correction on system level. Due to the aforementioned increased production from wind power, the fluctuations in production will increase in the future, making this layer even more important. This hierarchy level can be influenced both by the power company operating the portfolio of power generating units for minimising the deviation between sold and actual production, which is only reported in [Jørgensen et al., 2006], and by the TSO in the area, that uses a dynamic feedback approach to balance the load in the area. The latter is often referred to as a Automatic Generation Control (AGC).

The problem of designing AGCs to cooperate among multiple regions has been the subject of much research lately, both regarding optimisation and stability. However, it is often assumed that the generators within the area function as one generator. For example [Bakken and Grande, 1998] describe how to introduce an AGC in Norway, but the focus is on the main controller rather than the distribution of the error among the participating generators. Centralised AGC design under constraints is treated in [Hassan et al., 2008] both for single-area and multi-area production, but the area is treated as one generator. In [Venkat et al., 2006; Moon et al., 2000; Tyagi and Srivastava, 2006] decentralised model-based methods for multi-area AGC are developed, but without discussing how to distribute the output from the controller known as the area control error (ACE) among the multiple generators in the control area. Focusing on stability, [Azzam and Mohamed, 2002] developed a design method for generating a stabilising controller.

[Liu et al., 2003; Chen et al., 2007; Wood and Wollenberg, 1996] describe how to distribute the ACE among the participating generators in the area. [Liu et al., 2003; Chen et al., 2007] deal with control of multiple generators within an area using optimisation-based schemes. However, both treat the problem as a static rather than a dynamic problem. [Wood and Wollenberg, 1996] present an AGC for distributing the ACE to multiple generators based on a PI-controller structure with a set of distribution factors to share the contribution among multiple units. The distribution factors are based on a static optimisation of the system, [Raj, 2006] describes an updated way to use real time prices to update the distribution factors. A complete survey can be found in [Shayeghi et al., 2009].

1.2.1 Energy Market and Short Term Load Scheduler

The liberalisation of the power system has created a market which, according to [Jørgensen et al., 2006], includes two types of costumers from the power producers' point of view - The power exchanges and the TSO, the commodities traded in the power market appear in Figure 1.3. The market has influence on the production planning and balance control levels of the hierarchy, where the trades on the market is decisive for the production planning and balance control.

The different commodities traded in the market are:

1. Energy Every day an hourly based price for the next 24 hours production is set based on the producers' and buyers' forecasted demands. If the actual production

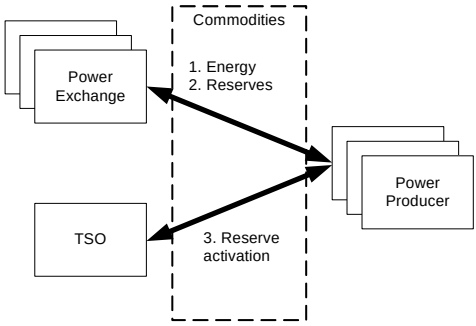


Figure 1.3: Power market commodities.

deviates from the sold production, the TSO will fine the producer, since the TSO must balance the production by activating reserves. During the day, bilateral trades among power producers are allowed through the power exchange as well, to cover foreseen production deficiencies in case of failure.

2. Reserves The TSO buys power reserves in the form of primary, secondary and tertiary reserves for a period in time to have capacity to balance out imbalances between the production and consumption. The seller must be able to activate the power reserve when required throughout the sold period. The reserves and their differences are described later in this section.

3. Secondary and tertiary reserve activation The Danish TSO can activate the bought reserves to balance production and consumption in western Denmark. The seller of the reserves will get extra payment if the reserve is activated. The primary reserve is governed by the frequency and must be automatically activated in case of deviations in the system frequency.

Each day on the energy market, which in the Danish case is Nord Pool [Nord Pool, 2010], at noon an auction is run for the forthcoming day. The production companies will submit amount and price for the energy production for each hour of the forthcoming day. The distribution companies will submit the consumption and price they are willing to pay. For each hour an intersection between consumption and production is formed, and this intersection determines the amount of energy and the price of energy .

After the auction has run, Nord Pool will announce the result to the participants of the auction which includes DONG Energy. The announced result is an amount of energy which is to be produced each hour. As depicted in Figure 1.4, the sold production is used by the *short-term load scheduler* (STLS) together with weather forecasts, district heating demand forecasts and constraints such as minimum amount of biomass fuel. The STLS solves the unit commitment problem again and the output of the short term load scheduler is a 5-minute based 24 hour ahead schedule for all production units that DONG Energy operates.

Based on the 5-minute based production plan generated by DONG Energy, The TSO generates two plans, an hourly and a quarter plan. These plans are used for settling payments for deviations.

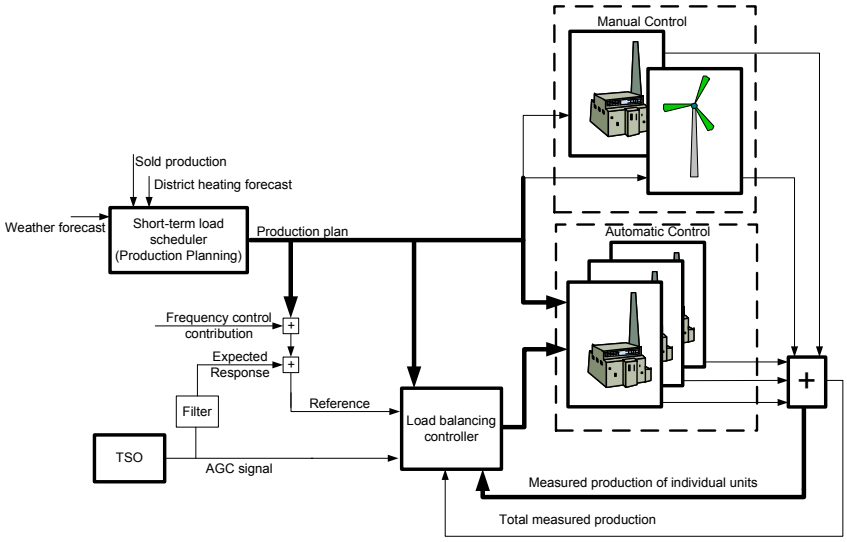


Figure 1.4: Diagram of the interconnection of the system. The bold lines show vectors of signals. The portfolio can be divided into two groups. A manual control which the load balancing controller cannot give corrections to, and an automatic control group which the load balancing controller can affect.

The first plan generated is an hourly based energy plan, referred to as the hourly plan, which defines the energy production at each hour of the forthcoming day. This plan can be changed up to 45 minutes prior to the start of each hour. When it is locked, the settlement price will be according to this plan.

The settlement price for each hour is based on the energy deviation between actual production and planned production multiplied by a price per energy unit. The price of the introduced deviations are not known in advance, and thus cannot be used for control purposes. Note that on an hourly basis, a positive deviation (production $>$ reference) is likely to generate an income rather than an expense.

The other plan generated is a quarter-based plan which must be changed according to the actual conditions during the hour. This means in case of faults on a unit, it is possible to change the quarter plan during the hour. This can result in the sum of the four quarterly plans of the hour being different from the hourly plan. This plan has been added to the market to ensure that the power balance is maintained and not just the energy balance. There is a settlement price on deviations from the quarterly plan as well. Any deviation outside $\pm 2.5MWh$ is billed at a price per energy unit. This will always result in an expense for the producer no matter if the deviation is positive or negative, although the price for positive and negative deviation is normally asymmetric. The prices for deviations on a quarterly basis are also not known in advance.

The full details of the billing and the market can be found in [Energinet.dk, 2010].

Reserves

Even though the market gives a good estimate of the demand for the following day, there will be deviations during the day for obvious reasons. Therefore, three levels of control have been established to balance production and consumption, see Figure 1.5.

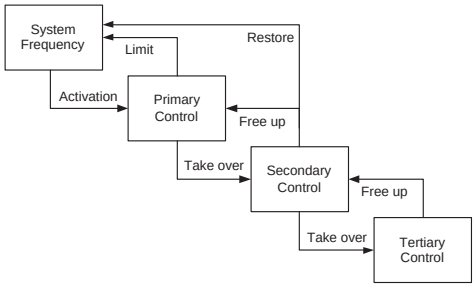


Figure 1.5: Interaction of of the tier of reserves.

In order to execute the control, it is required that a certain production capacity is reserved hence reserves. On the shortest time scale is the primary reserve which is used to avoid system collapse, and then followed up by slower reserves to bring the system back to the nominal state. The time scale for activation is shown in Figure 1.6.

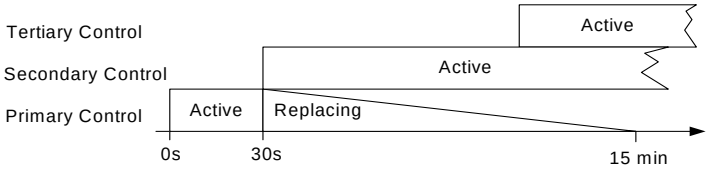


Figure 1.6: Time scale for the reserve activation. The primary reserves must be fully activated within 30 seconds. The primary reserves are then replaced with secondary reserves within 15 minutes. The secondary reserves must be maintained for as long as necessary until the tertiary reserves can take over.

Primary Reserves

When the system frequency deviates from the 50Hz, this reserve is to be activated proportionally to the system frequency deviation. The reserve must be activated within 30 seconds after a deviation occurs. Details about the reserve can be found in [ENTSO-E, 2010b]. In case of frequency deviations, the primary reserves are activated throughout the entire European grid.

In the ENTSO-E RG Continental Europe grid a total of $\pm 3000MW$ of primary reserves are maintained, of those $\pm 32.1MW$ must be maintained by western Denmark.

Primary reserve activation must be implemented as a local controller on the unit, typically on the process level. The controller measures the frequency of the system, and if it deviates from the nominal frequency of $50Hz$, the controller is activated. The primary

reserve controller must be implemented as a proportional controller with a deadband, resulting in the characteristics shown in Figure 1.7.

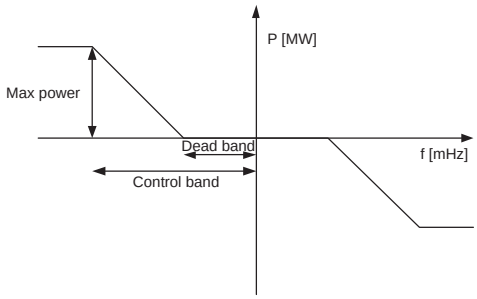


Figure 1.7: Primary reserve activation as a function of the frequency deviation. The controller activating the primary reserve must be implemented as a proportional controller.

Energinet.dk is responsible for providing the reserves, and buys them from the power producers in Denmark. In case Energinet.dk is buys reserves from DONG Energy they buy an amount from the portfolio. The distribution of the reserves among units within the portfolio can be freely chosen. The distribution is performed by the Frequency Control Scheduler which sends a set of parameters consisting of deadband, control band and max power to the local controllers to coordinate the local control with the amount sold.

On a system level the response anticipated from the primary reserve controller is added to the reference as seen in Figure 1.4 to avoid being canceled by secondary reserves.

Secondary Reserves

The secondary reserves are used to replace the primary reserves and help restore the system frequency when they are activated. Each control area e.g. western Denmark has secondary reserves. The control area which hosts an imbalance should seek to activate secondary reserves in order to reject the disturbance. This means that if an area creates a frequency deviation, all areas seek to stabilise the system with the primary reserves, but the area must bring the system back to nominal behaviour by activating secondary reserves.

The secondary reserves can in many cases be activated before a frequency deviation occurs. In western Denmark, the TSO measures the exchange with Germany, and in case of deviations from the planned exchange, secondary reserves are activated to normalise the situation.

The secondary reserves are activated automatically by a controller owned by the TSO without the interference of an operator. The TSO will send an activation signal for the secondary reserve activation which they then expect a filtered version of as a response. The distribution of the secondary reserve activation is performed by the load balancing controller shown in Figure 1.4

Tertiary Reserves

The last reserves in the battle to stabilise the system frequency are tertiary reserves. They must be activated within 15 minutes from the time of the order. They are activated by

the operator at the TSO by contacting to the operator at the central control room for the energy generation companies. The additional order of energy will most often be put into the STLS which will then generate and broadcast a new production plan to the units.

The size of the needed positive reserve is based on the $N-1$ principle, i.e. there must be enough reserves to outbalance a breakdown of the largest unit within the region. The reserves are asymmetric with $+630$MW and -160MW which must be fully delivered within 15 minutes.

1.2.2 Load Balancing Controller

The topic of the load balancing controller has already been briefly described in section 1.2.1. It serves two purposes; one of them is to distribute the secondary reserve activation signal among the units. The other purpose is to minimise the deviation between actual and sold power production.

The mechanism for determining the individual units participating in the control must contribute is proposed to be a steady state optimisation in [Wood and Wollenberg, 1996]. However, due to the conditions in western Denmark, where the boiler units are not used for base load, but rather changing load very frequently, the static optimisation approach has been deemed infeasible. Instead, the gains are determined by a logic-based mechanism, where each unit is prioritised by the operator for both negative and positive corrections. The logic then utilises the boiler unit with highest priority first, and after usage all boilers must be returned to the production plan.

Besides the main control loop, there is much logic in the controller for handling bumpless transfer between automatic and manual control and other features in an attempt to make the controller as optimal as possible. The result is a huge control structure with many cross couplings.

Figure 1.8 shows the correction signals from the load balancing controller during the morning hours. The correction amount is quite significant.

The problem with the current controller is the complexity of the cross couplings, which means that modifying one part of the controller often affects other parts of the controller in a way that the designer cannot predict. Thus, while the performance of the controller is quite adequate for the existing system, the current structure is not suited for portfolios that change structure over time. Furthermore, the complexity of the logics makes any form of rigorous stability or performance analysis virtually impossible.

To the author's knowledge no other load balancing controller for balancing the load within a portfolio has been reported in literature.

Figure 1.4 shows that the portfolio is split in two parts; an automatic control part and a manual control part. DONG Energy has the responsibility to deliver a total production from the portfolio corresponding to the reference. However, not all units have the ability to communicate with the load balancing controller - they will always be in manual control. But the units that are capable of participating are switched in and out of automatic control mode by the operator and the control systems on the unit. The result is a system that needs dynamic reconfiguration.

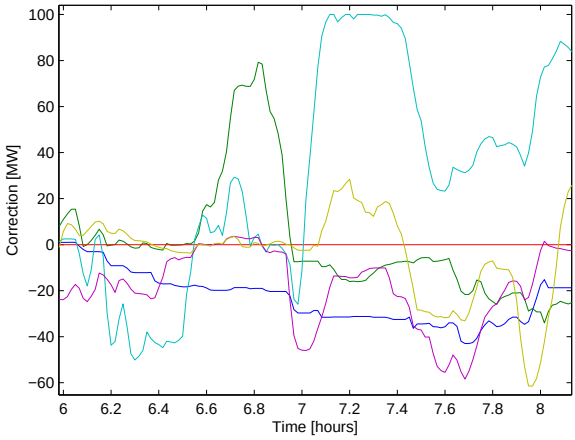


Figure 1.8: Example of control signals given by the current controller during the morning hours. Each line shows the corrective control signal to one unit in automatic control. The control signal for all six units are depicted, but only five participate in the control.

1.2.3 Power Plant Modelling and Control

The planning and dynamic coordination on system level becomes increasingly important to power systems. In order to cope with the increasing demand for flexibility, the existing power plants must be changed from base load to being able to change load fast.

In existing literature there are many detailed models of parts of the energy system to describe the dynamic behaviour of individual system components, such as [de Mello, 1991; Weber and Krueger, 2008].

There is focus both on improving processes in the power plants as well as the master control level of the unit, i.e. the two upper plant levels in Figure 1.2. [Deprugney and Liters, 2004] reports improvements on the control of the air controller. [Mølbak, 1999] reports improvement in control of superheater steam temperature control using Generalised Predictive control. [Dahl-Soerensen and Solberg, 2009] implement a simple controller to improve coal mill performance, while [Niemczyk et al., 2009] work on improving nonlinear models for use in coal mill control. [Majanne, 2005] works on stabilising the steam temperature in an industrial power plant where part of the steam is used for other purposes than power production using model predictive methods, while [Gibbs et al., 1991] use nonlinear model predictive methods to improve controller design to increase availability and lower pollution of fossil fired plants. [Mortensen et al., 1998] are concerned with improving the load following capabilities of the power plants on unit level using LQG methods, while [Deprugney et al., 2006] use H_∞-control. [Welfonder, 1997; Lausterer, 1998] both report significant improvements in the the disturbance rejection capabilities and load following capabilities of single power plants by using smaller energy buffers in the power plant which can later be repaid, such as the condensate system or turbine throttling valves.

[Bjerge and Kristoffersen, 2007] share experience designing the controller for an off-

shore wind farm to be integrated in the current power system.

Another issue is the start up of plants where [Franke and Vogelbacher, 2006; Albenesi et al., 2006] report better automation and faster start up of thermal power plants and combined cycle power plants using nonlinear model predtictive methods and nonlinear programming. A highly relevant problem when increased flexibility is needed.

Most of the developed controllers and models reported are complex and unsuited for making a load balancing controller which covers a large scope and therefore needs simple models to avoid too much complexity. The load balancing controller gives a set point to the power production and measures the output from the plant. Therefore, models should be limited to capturing the main dynamics along with the constraints governing the behaviour, such as the upper and lower production bounds and constraints on the rate of change on the set point.

1.3 State of the Art and Background of Chosen Methodology

This section provides an overview of the state of the art methodology utilised for developing the design method to fulfil the hypothesis.

There are many methods for controlling a MIMO system, such as the power system portfolio. Spanning from the current PI-controller structure based on SISO theory in combination with cross couplings and feed forward ([Franklin et al., 2002; Åström and Hägglund, 2006] to mention a few) to more advanced techniques model-based multivariable controllers like LQR or H_∞-control [Skogestad and Postlethwaite, 2005]. The power system portfolio is a constrained MIMO system with knowledge of the future reference. Therefore, Model Predictive Control (MPC) is an obvious controller scheme to choose.

In this thesis a linear MPC implementation is utilised which requires repeated online solution of constrained linear optimisation problem. Therefore, the some basics of convex optimisation with the focus on linear programming is covered first in this section.

1.3.1 Convex Optimisation - Linear Programming

In MPC applications the performance and reliability of the optimisation algorithm solving the constrained optimal control problem are important elements, as the optimisation problem is solved repeatedly online. In linear MPC the performance function is usually quadratic, linear, or ℓ_1-norm based as described previously. Using these performance functions leads to a convex optimisation problem as treated in [Boyd and Vandenberghe, 2004].

The performance function used in the controller design method in this project result in a linear constrained optimisation problem, which is a special case of convex programming and will be described here. A general linear program has the structure

$$\min_{\mathbf{z}} \quad \phi = \mathbf{c}^T\mathbf{z} \tag{1.1a}$$

$$s.t. \quad \mathbf{G}\mathbf{z} \geq \mathbf{h} \tag{1.1b}$$

with $\phi \in \mathbb{R}$ being the functional to be minimised in order to find optimum, $\mathbf{z} \in \mathbb{R}^n$ are the free variables which can be manipulated in order to minimise ϕ, $\mathbf{c} \in \mathbb{R}^n$ contains the weights of the free variables, weighing their importance relative to each other. $\mathbf{G} \in \mathbb{R}^{m\times n}$ is the constraint matrix, and $\mathbf{h} \in \mathbb{R}^m$ is the affine part of the constraints.

Checking if a solution is an optimal solution to (1.1) is equivalent to finding a solution $(\mathbf{z}^*, \pi^*)$ to the corresponding Lagranian function

$$\mathcal{L}(\mathbf{z}, \pi) = \mathbf{c}^T\mathbf{z} - \pi^T(\mathbf{G}\mathbf{z} \geq \mathbf{h}) \tag{1.2}$$

with $\pi \in \mathbb{R}^m$ being the introduced Lagrange multipliers. If the solution $(\mathbf{z}^*, \pi^*)$ fulfils the *Karush-Kuhn-Tucker* (KKT) conditions

$$\begin{aligned}
\nabla_{\mathbf{z}}\mathcal{L} = \quad & c - \mathbf{G}^T\pi = 0 && && \text{(1.3a)}\\
\nabla_{\pi}\mathcal{L} = \quad & \mathbf{G}\mathbf{z} - \mathbf{h} - \mathbf{s} = 0 && && \text{(1.3b)}\\
& s_i\pi_i = 0 && i = 1, 2, \ldots, m && \text{(1.3c)}\\
& \mathbf{s}, \pi \geq 0 && && \text{(1.3d)}
\end{aligned}$$

with the slack variable $\mathbf{s}$ defined as

$$\mathbf{s} = \mathbf{G}\mathbf{z} - \mathbf{h} \geq 0. \tag{1.4}$$

then the solution $\mathbf{z}^*$ is an optimal solution to (1.1) [Nocedal and Wright, 2006].

The KKT conditions imply that the first derivative of the Lagranian with respect to $\mathbf{z}$ as well as the first derivative with respect to π must be zero. Furthermore, element wise either the constraint or Lagrange multiplier must be zero. $s_i > 0$ means that the proposed solution is not on the constraint, and thus the constraint does not affect whether or not the optimum is reach. If $s_i = 0$ the constraint is active and the lagrange multiplier π_i can be different from zero, and thus affecting (1.3a).

The special property of a linear program is that the solution will always be on a vertex of the feasible area. This property can be exploited when finding the solution. In case of a non unique solution, there will still be a valid solution on a vertex. Illustrated in Figure 1.9 is the optimsation problem

$$\begin{aligned}
\min_{\mathbf{z}} \quad & \phi = -z_1 - 2z_2 && \text{(1.5a)}\\
s.t. \quad & z_1 \leq 3 && \text{(1.5b)}\\
& z_2 \leq 3 && \text{(1.5c)}\\
& z_1 + z_2 \leq 4 && \text{(1.5d)}\\
& z_1 \geq 0, z_2 \geq 0 && \text{(1.5e)}
\end{aligned}$$

The optimum is shown in the figure and is in an extreme point of the feasible area. There are two main methods to solve this problem, either through the *Simplex* algorithm [Dantzig and Thapa, 1997], or through a *primal-dual interior point algorithm* such as Mehrotra's predictor-corrector algorithm [Mehrotra, 1992; Wright, 1997; Zhang, 1998; Czyzyk et al., 1999; Nocedal and Wright, 2006].

The Simplex algorithm starts at a feasible extreme point of the problem, and travels along the edges of the feasible region until it finds optimum. The search will always happen in the direction with the steepest decline. For the example, starting at vertex 1 there are two possibilities 2 or 5. The direction towards 2 has the steepest decline. From here the algorithm would go to 3 and conclude it to be optimal since following any vertex

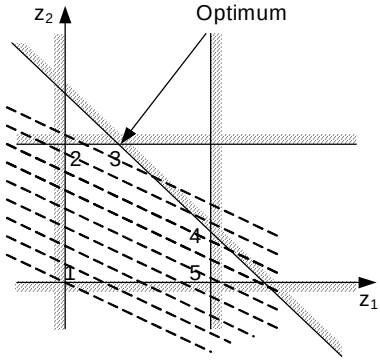


Figure 1.9: Two dimensional linear optimisation problem. The lines show the inequality constraints. The dashed lines show the contours of the performance function

would lead to an increase in the objective function. For a detailed mathematical covering of the algorithm see [Dantzig and Thapa, 1997]. The chosen path is shown in Figure 1.10. The simplex algorithm belongs to the group of active set solvers [Nocedal and Wright, 2006], a group which is not restricted to linear programming.

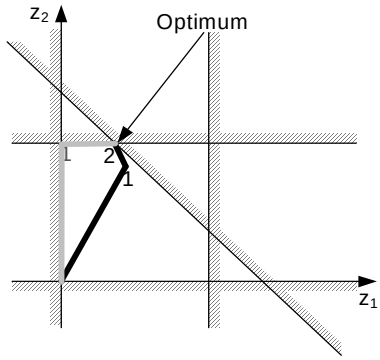


Figure 1.10: Paths to the solution for the interior point and simplex methods. The black line shows the interior point method, while the grey line shows the simplex method.

The simplex algorithm is not used in practice, but rather an implementation known as the revised simplex method [Dantzig and Thapa, 1997] which is more computationally efficient. [Klee and Minty, 1972] showed that the simplex algorithm in worst case needs to visit all extreme points of the feasible area, and thus grows exponential with the problem size [Nocedal and Wright, 2006]. In practice the algorithm works well and is widely used. However, this theoretical drawback has lead to the development of alternative methods, such as the interior point methods.

Interior point methods make a search through the interior of the feasible area to the optimum based on the gradient of the performance function. Interior point methods usually uses fewer but more computationally expensive iterations to reach optimum. The

rule of thumb says that simplex algorithm is faster on small and medium problems, while the interior point methods are competitive on large-scale problems [Nocedal and Wright, 2006]. However, this is only a rule of thumb. One of the advantages of the interior point method is that the computational complexity lies in calculation of a matrix and a Cholesky factorisation. And thus it is possible to exploit the structure of the problem and tailor the algorithm to be efficient on a certain problem. The interior point methods might find an optimum on an edge instead of a vertex in case of a non unique solution. Figure 1.10 shows the path through the interior, it makes a few iterations very close to the goal to converge completely, which is not shown in the figure.

In an MPC context [Rao et al., 1998] show how to structure a quadratic program arising from linear MPC with a quadratic performance function to make efficient use of interior point methods for solving the optimisation problem.

1.3.2 Model Predictive Control

Model Predictive Control (MPC) has successfully been applied in the process industries for more than thirty years [Qin and Badgwell, 1997, 2003; Froisy, 2006]. Regarding the use of MPC within power system, it has been applied both to single elements like boilers [Rossiter et al., 2002; Gibbs et al., 1991] and wind farms [Senjyu et al., 2009]. It has also been applied to coordination of power systems [Venkat et al., 2006; Larson and Karlsson, 2003; Negenborn et al., 2009].

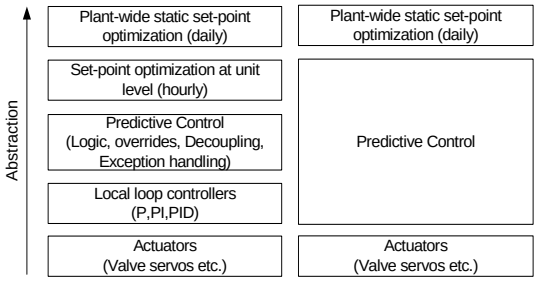
MPC refers to a group of control algorithms that makes explicit use of a process model to predict future responses from the system. In most implmentations the prediction horizon is finite and constant, these algorithms are also known as receding horizon controllers. At each controller update, measurements from the controlled plant are gathered and predictions are based on these measurements. The predictions are used to evaluate a performance function, and an optimisation is performed which seeks to find the input sequence optimising the performance function over the chosen horizon. The first input in the sequence is then applied to the plant, and the procedure is repeated at every controller update.

In this thesis the models used for prediction are linear, though both linear [Muske and Rawlings, 1993] and nonlinear models [Allgöwer et al., 1999; Tennyu et al., 2004] can be used. An overview of linear MPC is found in [Rossiter, 2003; Maciejowski, 2002; Rawlings and Mayne, 2009] among others.

MPC has a number of strengths, these are the ability to incorporate constraints, using future knowledge and not least handle MIMO systems. The most important ability with MPC is the ability to incorporate constraints both on input, output and internal states of the system with MPC. Even though it is denoted linear MPC and it has linear models and affine constraints, the resulting controller is nonlinear. Compared to a linear controller it is possible to move the system closer to the constraints without increasing the number of constraint violations.

Process Control Hierarchy

The placement in the control hierarchy is given for the controller in this thesis. However, this section briefly discusses where MPC is usually applied in the hierarchy. MPC is usually found in the middle of the hierarchy, as shown in Figure 1.11a.



(a) Typical Hierachy for MPC (b) Future trend for use of MPC.

Figure 1.11: Typical control hierarchy for MPC [Maciejowski, 2002].

The reason is mainly due to the computationally complexity, including the local control loops where P- and PI-controllers are dominant in the model predictive controller will increase the size of the problem, thus making it impossible to solve it within the time limit. [Maciejowski, 2002] suggests, as can be seen in Figure 1.11b, that the future trend is to incorporate the local control loop as well as the set point optimisation in the MPC. [Pannocchia et al., 2004, 2005] show that in some cases MPC should be considered over PID controller even in SISO systems.

Computational Aspects of MPC

Model Predictive Control is often expressed using either ℓ_2-penalty functions without economic terms [Muske and Rawlings, 1993], ℓ_1-penalty functions without economic terms [Chang and Seborg, 1983; Allwright and Papavasiliou, 1992; Rao and Rawlings, 2000] or using economic terms only [Rawlings and Amrit, 2009].

When using ℓ_2-penalty functions the result is a convex quadratic programming problem which is covered in [Boyd and Vandenberghe, 2004]. Using ℓ_1-penalty or linear terms result in a linear programming problem which is further discussed in Section 1.3.1. MPC requires repeated online solution of these optimisation problems. Therefore, the computational speed and robustness of the optimisation algorithms have limited the type of applications that can be controlled by MPC. MPC was originally developed for the process industries with relative slow dynamics and a low number of input and output (say less than 50). As MPC is developed for mechatronic applications with very fast dynamics, low state order models, and typically less than three input and output, new ways of implementing and solving the constrained optimization problem constituting the MPC have been developed. Using explicit controllers found by means of multi-parametric programming [Bemporad et al., 2002; Sakizlis et al., 2007] reduces the online problem to a look up table.

Another method to reduce the number of free variable is through input blocking [Qin and Badgwell, 1997; Maciejowski, 2002]. Input blocking is a technique to only allow the controller to change the input at a limited number of times throughout the prediction horizon, as opposed to every sample instance throughout the prediction horizon. Often this will be the first moves of the prediction horizon and for the remaining part the input

is kept constant.

Both process control and mechatronic applications use one centralised MPC to control the system. This is possible because of the low number of input and output as well as the relative low number of states in the model. The system in this thesis consists of fast dynamics with a large number of controlled inputs and outputs, therefore methods for achieving lower computationally complexity of the controller by exploiting the structure of the problem is treated later in this section.

Models

A common implementation of models in MPC is step or impulse response models. The advantage of using these convolution models is that they can represent any kind of stable dynamic process [Muske and Rawlings, 1993].

The problem with this formulation is that unstable models cannot be represented. [Morari and Lee, 1991; Eaton and Rawlings, 1992] described ways to encompass this deficit by representing the instability as an integrator. [Maciejowski, 2002] gives a way to decompose the unstable model by using coprime factorisation [Zhou et al., 1996].

The systems modelled in this thesis are all stable models, and thus impulse response models are used to represent the system dynamics.

Starting with a state space model used for N-step prediction

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{E}\mathbf{d}_k \tag{1.6a}$$

$$\mathbf{z}_k = \mathbf{C}\mathbf{x}_k \tag{1.6b}$$

an impulse response model can be derived as

$$\mathbf{z}_k = \mathbf{C}\mathbf{A}^k\mathbf{x}_0 + \sum_{i=0}^{k-1}\mathbf{H}_{u,k-i}\mathbf{u}_i + \sum_{i=0}^{k-1}\mathbf{H}_{d,k-i}\mathbf{d}_i \tag{1.7}$$

with $k = 1, 2, \ldots, N$ and the impulse response coefficients defined as

$$\mathbf{H}_{u,i} = \mathbf{C}\mathbf{A}^{i-1}\mathbf{B} \qquad i = 1, 2, \ldots, N \tag{1.8a}$$

$$\mathbf{H}_{d,i} = \mathbf{C}\mathbf{A}^{i-1}\mathbf{E} \qquad i = 1, 2, \ldots, N \tag{1.8b}$$

Define the vectors

$$\mathbf{U} = \begin{bmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \\ \vdots \\ \mathbf{u}_{N-1} \end{bmatrix} \quad \mathbf{D} = \begin{bmatrix} \mathbf{d}_0 \\ \mathbf{d}_1 \\ \vdots \\ \mathbf{d}_{N-1} \end{bmatrix} \quad \mathbf{Z} = \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_N \end{bmatrix}$$

and the matrices

$$\mathbf{\Phi} = \begin{bmatrix} \mathbf{C}\mathbf{A} \\ \mathbf{C}\mathbf{A}^2 \\ \vdots \\ \mathbf{C}\mathbf{A}^{N-1} \end{bmatrix} \quad \mathbf{\Gamma}_\alpha = \begin{bmatrix} \mathbf{H}_{\alpha,1} & \mathbf{0} & \ldots & \mathbf{0} \\ \mathbf{H}_{\alpha,2} & \mathbf{H}_{\alpha,1} & \ldots & \mathbf{0} \\ \vdots & \vdots & & \\ \mathbf{H}_{\alpha,N} & \mathbf{H}_{\alpha,N-1} & \ldots & \mathbf{H}_{\alpha,1} \end{bmatrix}$$

with $\alpha \in \{u, d\}$. Using (1.7) the stacked output, Z, may be expressed by the linear relation

$$\mathbf{Z} = \mathbf{\Phi}\mathbf{x}_0 + \mathbf{\Gamma}_u\mathbf{U} + \mathbf{\Gamma}_d\mathbf{D} \tag{1.9}$$

This model description has eliminated all internal states, except the current, and thus the size of the matrix relating to controlled input, $\mathbf{\Gamma}_u$, is only dependent on the number of input, output and prediction horizon.

Output Feedback and Offset-free Tracking

MPC assumes that the state vector is measurable in order to make correct predictions. This is often not the case, so in order to be able to achieve output feedback a state observer is needed, for instance a Kalman Filter [Grewal and Andrews, 2008].

If a step disturbance enters the system, the combination controller and observer will result in a steady state offset from the reference. The same behaviour is exhibited when the steady state gain of the model is different from the steady state gain of the system [Maciejowski, 2002].

To remove this error, an augment the system model with a disturbance model in the observer. [Pannocchia and Rawlings, 2003] suggests a Kalman filter designed for the augmented system

$$\begin{bmatrix}\mathbf{x}_{k+1}\\ \mathbf{d}_{k+1}\end{bmatrix} = \begin{bmatrix}\mathbf{A} & \mathbf{B}_d\\ \mathbf{0} & \mathbf{I}\end{bmatrix}\begin{bmatrix}\mathbf{x}_k\\ \mathbf{d}_k\end{bmatrix} + \begin{bmatrix}\mathbf{B}\\ \mathbf{0}\end{bmatrix}\mathbf{u}_k + \mathbf{w}_k \tag{1.10a}$$

$$\mathbf{y}_k = \begin{bmatrix}\mathbf{C} & \mathbf{C}_d\end{bmatrix}\begin{bmatrix}\mathbf{x}_k\\ \mathbf{d}_k\end{bmatrix} + \mathbf{v}_k \tag{1.10b}$$

with $\mathbf{d}_k \in \mathbb{R}^{n_d}$, $\mathbf{B}_d \in \mathbb{R}^{n\times n_d}$ and $\mathbf{C}_d \in \mathbb{R}^{q\times n_d}$. The noise vectors $\mathbf{w}_k \in \mathbb{R}^{n+n_d}$ and $\mathbf{v}_k \in \mathbb{R}^p$ are assumed to be zero-mean white noise disturbances for the augmented system.

For a stable estimator to exist the original system must be detectable and the following condition must hold

$$\operatorname{rank}\begin{bmatrix}\mathbf{I}-\mathbf{A} & -\mathbf{B}_d\\ \mathbf{C} & \mathbf{C}_d\end{bmatrix} = n + n_d \tag{1.11}$$

A pair of matrices ($\mathbf{B}_d$, $\mathbf{C}_d$) always exists such that (1.11) holds.

It is possible to obtain offset-free tracking for the system if

$$\operatorname{rank}\begin{bmatrix}\mathbf{I}-\mathbf{A} & -\mathbf{B}_d\\ \mathbf{C} & \mathbf{C}_d\end{bmatrix} = n + q \tag{1.12}$$

with n being the number of states in the original system, and q is the number of output. This only holds if the constraints are not active and the closed-loop system is stable. Similar results to [Pannocchia and Rawlings, 2003] are obtained in parallel in [Muske and Badgwell, 2002].

If $\mathbf{B}_d = \mathbf{0}$ and $\mathbf{C}_d = \mathbf{I}$ then the disturbance model is modelled as a constant as described in [Muske and Rawlings, 1993]. This model is often denoted an output error model.

Controller Tuning

A model predictive controller must be tuned like most other controller. The performance is based on the values of the weight functions in the performance function and the prediction horizon as well as the observer, for instance the covariance matrices in a Kalman filter. Depending on the problem size, this gives quite a number of free design variables.

In practice, for all types of controllers tuning, the controller such that the systems behaves "right" is often a matter of trial and error.

In some cases the weight matrices may be given, but in most cases this is a task for the controller designer. There are methods to aid controller tuning such as loop transfer recovery [Doyle and Stein, 1981], and least-squares methods for estimating autocovariance on noise [Åkesson et al., 2007; Odelson et al., 2006; Rajamani and Rawlings, 2009].

1.3.3 Hierarchical Control and Reconfigurable Systems

Decomposing the control problem into smaller problems, whether the structure is decentralised without communication between local controllers [Elliott and Rasmussen, 2008; Acar, 1995; Magni and Scattolini, 2006; Raimondo et al., 2007], distributed where the local controllers communicate [Mercangöz and Doyle III, 2007; Jia and Krogh, 2001; Dunbar, 2007] or hierarchical, as used in this thesis, can serve many purposes.

Complexity of power plants, power systems and most other process and traffic networks have increased due to a wish to optimise them. The systems often consist of multiple units or subsystems interacting, and it can be difficult to control with a centralised control structure. Reasons for not pursuing a centralised solution could be that the controlled system is spread over a physical area where communication could be expensive in resources such as communication bandwidth of power consumption as in multi robot coordination [Keviczky et al., 2008], or communication delays as in multi area automatic generation control [Venkat et al., 2008]. Other reasons for decomposition of the control is to achieve robustness, reliability or reconfigurability of the subsystems without having to redesign the whole controller or to achieve a lower computationally complexity.

The overall structure of the power system portfolio control is hierarchical, but on the plant level, especially in the units, numerous examples of both decentralised and distributed controllers can be found. The main focus on the rest of the section is on MPC implementations of hierarchical control and reconfigurable control.

A good classification and review of the subject of the area can be found in [Scattolini, 2009].

There are many variants of hierarchies, the main structure of the power system portfolio is a multi layered hierarchy. However, this thesis treats two layer hierarchical control for coordination. The idea of hierarchical control and the design of coordinators has been studied for a long time [Mesarovic et al., 1970]. The basic idea is that the system comprises a set of subsystems under local control with some interaction either through a common goal or through dynamic interaction.

The basic idea is described in [Scattolini, 2009] and shown in Figure 1.12. For each local system an MPC optimises the local performance function under local constraints. If the local solution for each subsystem satisfies the constraints that couples the subsystems together, the solution is accepted. If this is not the case, the coordinator will update the local control objective based on the coupling constraint. [Scattolini, 2009] suggests

that the local objective functions are updated using the Lagrange multipliers of the coupling constraint. This scheme is then pursued in an iterative manner until the coupling constraint is satisfied. How the coordinator updates the price of the common resource to guarantee convergence of this method as well as if it converges to the same optimum as a centralised solution, is not specified.

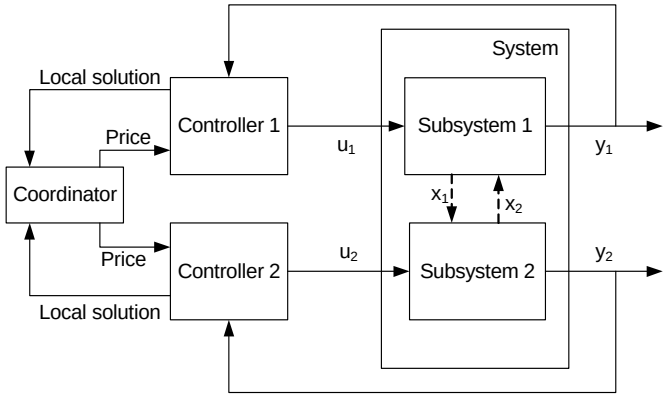


Figure 1.12: Hierarchical controller where the the local controllers are coordinated through a supervisor [Scattolini, 2009].

This coordination method has been the topic of [Negenborn, 2007; Negenborn et al., 2008a, 2009] who use it in power networks as well as traffic networks [Negenborn et al., 2008b]. Using a price updating scheme is also the approach of [Rantzer, 2009] who uses dual decomposition to decompose the system. [Marcos et al., 2009; Cheng et al., 2007] use a newton search and sensitivity analysis to update the price scheme.

As described in Section 1.2, the controllable part of the system changes topology frequently. It may only be once every few days, or several times per day depending on the scenario. It is frequent enough that the controller must be able to handle it.

There are two major research topics within this field. One of them is fault tolerant control [Blanke et al., 2006; Iserman, 2005] which relates to detection and isolation in case of failure in part of the system. However, when a plant changes from manual to automatic control or vice versa, that is not a fault, it is an occurrence that needs to be handled. Plug and Play process control [Stoustrup, 2009] is an ongoing research topic dealing with this kind of systems both with models [Michelsen and Trangbæk, 2009] and without models [Bendtsen and Trangbæk, 2009] to support the reconfiguration of the system. A related topic is found in [Chokshi and McFarlane, 2008] which treat reconfigurability of manufacturing systems.

1.3.4 Decomposition of Linear Programs

When a linear program has a structured constraint matrix, it is possible to solve it efficiently by decomposing it into smaller programs. Two methods for decomposing such a system is the *Lagrange relaxation* [Beasley, 1993] and *Dantzig-Wolfe decomposition* [Dantzig and Wolfe, 1960; Dantzig and Thapa, 2002; Lasdon, 2002]. Both methods can

significantly decrease the computation time for optimisation of linear systems with special structure in the constraint matrix, especially problems as given in (1.1) with a block-angular structure of G, such that the problem can be stated as

$$\min_{\mathbf{z}} \quad \phi = \mathbf{c}_1^T \mathbf{z}_1 + \mathbf{z}_2^T \mathbf{z}_2 + ... + \mathbf{c}_P^T \mathbf{z}_P \tag{1.13a}$$

$$s.t. \quad \begin{bmatrix} \mathbf{F}_1 & \mathbf{F}_2 & \dots & \mathbf{F}_P \\ \mathbf{G}_1 & & & \\ & \mathbf{G}_2 & & \\ & & \ddots & \\ & & & \mathbf{G}_P \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_P \end{bmatrix} \geq \begin{bmatrix} \mathbf{g} \\ \mathbf{h}_1 \\ \mathbf{h}_2 \\ \vdots \\ \mathbf{h}_P \end{bmatrix} \tag{1.13b}$$

with $\mathbf{z} = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_P]$

Both methods use an iterative scheme to solve the optimisation problem. In each iteration, the lagrange multipliers attached to the coupling constraints of the problem are assumed constant. Thus the optimisation problem is reduced to a block diagonal structure and can be treated as P independent problems. The difference in the two methods is how to find the Lagrange multipliers. Lagrange relaxation computes the multipliers through heuristic methods, while Dantzig-Wolfe decomposition finds the multipliers by solving an optimisation problem. [Gunnerud et al., 2009] showed that the computation time using Lagrange relaxation is very sensitive to changes in the problem, and even minor changes might result in a doubling of the computational time.

[Negenborn et al., 2008b; Rantzer, 2009] applied Lagrange relaxation for decomposing problems in a model predictive control context. [Gunnerud et al., 2009] used Dantzig-Wolfe decomposition for control and planning purposes on a longer time scale, while [Cheng et al., 2008] used Dantzig-Wolfe for target calculation in a distributed model predictive controller. The contribution of this thesis is to use Dantzig-Wolfe decomposition for computation of the dynamic calculations of model predictive control.

The Dantzig-Wolfe decomposition breaks the linear program (1.13) into P independent subproblems and a Master Problem (MP). The Master Problem coordinates the subproblems. The Master Problem sends Lagrange multipliers, π, of the coupling constraint to each of the subproblems. The Lagrange multiplier is often interpreted as a price of a common resource which gives rise to the coupling constraint. Using this price, π, each of the P subproblems computes their optimal solution. This interchange of information continues until convergence. This is the same description as given by [Scattolini, 2009] for how to make hierarchical MPC.

For the dual problem Benders' decomposition can be used efficiently [Benders, 1962]. Dantzig-Wolfe decomposition builds on the principles of the simplex algorithm, and the subproblems must to be solved so that the proposed solution is a vertex of the feasible area. Interior point methods for Dantzig-Wolfe decomposition has been developed in [Martinson and Tind, 1998].

1.4 Outline of Thesis

This thesis is made as a collection of publications and is divided into two parts. The first part, which has already begun with a system description and state of the art in Chapter

1, consists of an introduction and state of the art. The next chapter, Chapter 2, describes the design method developed through the contributions which fulfil the criteria of the main hypothesis as presented in Section 1.1. A summary of the contributions is given in Chapter 3. Part 1 ends with a conclusion and suggestions for future work presented in Chapter 4.

The second part is the presentation of the six publications made during the project and presented in the following order:

Paper A [Edlund et al., 2009a] Shows that the controller structure of the current controller is internally unstable, unless the gains distributing the control actions were chosen carefully. This research was driven by an observed problem where the power plant in control drifted away from the production plan. This paper serves as a motivation for developing a new and more stringent design method.

Paper B [Edlund et al., 2008] Introduces model predictive control for controlling a power plant portfolio. This paper gives a preliminary indication that MPC is a viable option to base a design method upon.

Paper C [Edlund et al., 2009b] MPC relies on models, and as the scope broadens, the required fidelity of the models are lowered. Therefore, it was necessary to develop simple models for use in a model based portfolio controller. This paper covers the modeling of different possibilities to change load within the portfolio. These possibilities are termed effectuators. The paper includes models for four different effectuators; boiler load, district heating, condensate throttling and wind turbines. Only the boiler load unit is currently operational in the portfolio.

Paper D [Edlund et al., 2009c] In this paper a primal-dual interior point algorithm based on Mehrotra's predictor-corrector algorithm is tailored to the control of a single boiler load effectuator. Even though the optimisation problem in the controller is decomposed, an efficient solution strategy still relies on solving the derived subproblems fast. This paper exploits the structure of the subproblem to reduce the number of calculations.

Paper E [Edlund and Jørgensen, nd] Introduces Dantzig-Wolfe decomposition for solving the dynamic part of the model predictive controller. A thorough description of the algorithm is presented. The main result of the paper is the experimental results showing linear scalability of the algorithm as a function of the number of effectuators.

Paper F [Edlund et al., nd] This paper gives an overview of the complete design method, including the handling of switching effectuators in and out of automatic control. A simulation based comparison with the currently implemented controller is presented based on the actual scenario of a month of portfolio operation.

2 | Design Method

The load balance controller as seen in Figure 1.4 is the focus of this thesis. The objective of the controller is to minimise deviations between sold and actual production as well as activating secondary reserves when ordered by the TSO.

In order to accept the main hypothesis from Section 1.1, the design method for the new load balancing controller must result in a a controller that meets the criteria:

Scalability Is scalable in the number of units it can coordinate.

Flexibility Must be flexible, so that addition of new units and maintenance of existing units is possible. This means that the design must have a modular structure with good encapsulation of information and clear communication interfaces between modules.

Performance Perform as least on par with the current controller measured on some performance criteria.

Balance between the production and consumption is currently maintained by changing the production, but in some cases consumption could be changed as well to achieve the goal. For the sake of generalisation all power producing and power consuming units that are capable of participating in load balancing control are termed *effectuators* which has the definition:

Definition 1. An effectuator is a process or part of a process in a power system that represents control actions with associated dynamics and actuation costs allowing the power output to be manipulated.

2.1 Proposed Controller Structure

The structure of the proposed controller is a two layer hierarchical structure as shown in Figure 2.1. All parts referring to the individual effectuators in the controller are placed in the lower layer separated from one another, allowing them to be modified, removed or adding new ones without affecting the other units. Above is a coordination layer coordinating the units in question to achieve the portfolio goal of minimising deviations.

Model Predictive Control (MPC) has been chosen as the controller scheme, since the system is a constrained MIMO system where knowledge of the future references are available.

The design framework relies on a set of assumptions:

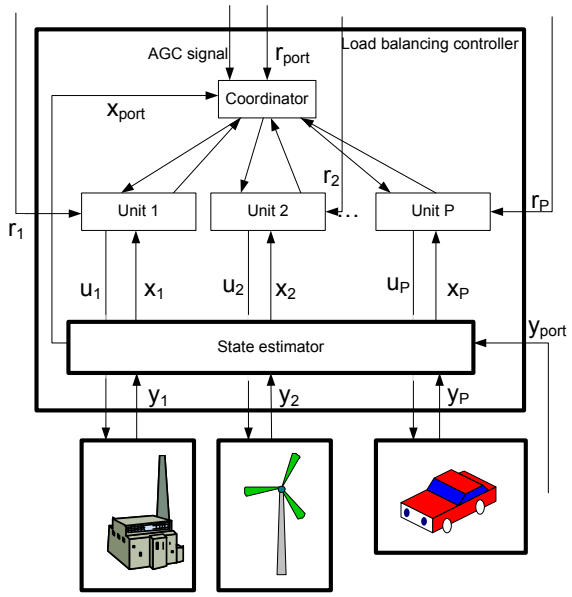


Figure 2.1: Sketch of the modular structure of the load balancing controller. Communication with the individual effectuator is handled by the independent subsystems, and portfolio communication is handled on the upper layer of the hierarchy. r_i is the reference to effectuator $i \in \{1, 2, \ldots, P\}$, x_i is the state estimate, y_i is the measured output, and u_i is the controller correction. For the portfolio there is a reference r_{port}, state estimate x_{port} and a total measured production y_{port}. The references r_i and r_{port} come from the STLS as seen in Figure 1.4.

- The effectuators can be modelled as independent of each other, so that a change in one effectuator does not directly affect another effectuator.
- The effectuators can be modelled as a linear dynamic model with affine constraints. The investigated models in [Edlund et al., 2009b] can all, with minor modifications, be modelled with the structure shown in Figure 2.2. However, other kinds of linear

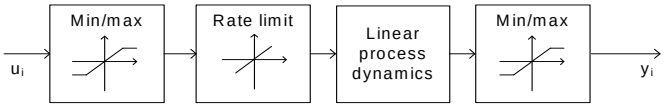


Figure 2.2: General structure of the effectuators

  input, output and state constraints fit into the modelling framework as well.
- The underlying optimisation problem in the MPC can be stated as a linear program, which means the corresponding objective function must consist of linear and ℓ_1-norm elements.

Each effectuator in the lower layer of the hierarchy contains a constrained linear model and an objective function for the optimal operation of the effectuator which together form a constrained linear programming problem. Furthermore, it contains all communication with the physical unit. The information that must be sent to the upper layer is how the output of the effectuator will affect the portfolio output, meaning a prediction of the power production/consumption of the unit.

The upper layer contains a constrained linear model of the portfolio excluding the individually modelled effectuators, as well as an objective function of the optimal operation of the portfolio. The upper layer also handles communication with surrounding systems, for instance obtaining the portfolio reference (the load schedule).

2.1.1 Solving the Optimisation Problem

The hierarchical structure encapsulates the information referring to each unit. However, one challenge persists: MPC relies on solving an optimisation problem at each sample. This is a challenge of the MPC framework, since solving the optimisation problem usually grows cubically with the size of the problem. Therefore, one of the design challenges has been to create an optimisation problem which can be encapsulated in the same hierarchical structure as well as being scalable.

To solve the optimisation problem, a Dantzig-Wolfe decomposition approach has been applied [Dantzig and Wolfe, 1960; Dantzig and Thapa, 2002]. The decomposition technique has been adapted to the MPC context in [Edlund and Jørgensen, nd] where details of the algorithm are also described.

Dantzig-Wolfe decomposition can only be applied to linear problems. The performance function for the whole problem is assumed to be chosen as a mixture of linear and ℓ_1-norm terms which can be rewritten into a linear program suitable for Dantzig-Wolfe decomposition.

An important consequence of this forced choice of performance function and constraints is the solution, i.e. the point where the performance function attains its extremum, must either be at an extreme point of the feasible set, or the solution of an unconstrained problem.

When the optimisation problem is composed from the effectuator optimsation problems and the portfolio optimisation problem, it can be rewritten into a linear program with the structure

$$\min_{\mathbf{z}} \quad \phi = \mathbf{c}_1^T\mathbf{z}_1 + \mathbf{c}_2^T\mathbf{z}_2 + ... + \mathbf{c}_P^T\mathbf{z}_P \tag{2.1a}$$

$$s.t. \quad \begin{bmatrix} \mathbf{F}_1 & \mathbf{F}_2 & \dots & \mathbf{F}_P \\ \mathbf{G}_1 & & & \\ & \mathbf{G}_2 & & \\ & & \ddots & \\ & & & \mathbf{G}_P \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_P \end{bmatrix} \geq \begin{bmatrix} \mathbf{g} \\ \mathbf{h}_1 \\ \mathbf{h}_2 \\ \vdots \\ \mathbf{h}_P \end{bmatrix}. \tag{2.1b}$$

with $\mathbf{z} = [\mathbf{z}_1, \mathbf{z}_2, \dots \mathbf{z}_P] \in \mathbb{R}^n$, $\mathbf{z}_i \in \mathbb{R}^{n_i}$, $\phi \in \mathbb{R}$, $\mathbf{F}_i \in \mathbb{R}^{m \times n_i}$, $\mathbf{G}_i \in \mathbb{R}^{p_i \times n_i}$, $\mathbf{g} \in \mathbb{R}^m$ and $\mathbf{h} \in \mathbb{R}^{p_i}$. ϕ is the functional which needs to be minimised in order to find optimum, $\mathbf{z}_i$ are the free variables, $\mathbf{c}_i$ are weight factors, weighing the importance of the corresponding $\mathbf{z}_i$. The constraint matrix has a block-angular structure where the block diagonal elements

come from the effectuator optimisation problem, and the coupling constraint comes from the portfolio linking the problem together. $\mathbf{F}_i$ is unit i's contribution to the coupling constraint. $\mathbf{G}_i$ originates from the individual effectuators optimisation problem. $\mathbf{g}$ and $\mathbf{h}_i$ are the affine parts of the constraints. Ignoring the coupling constraints the program consist of P independent problems

$$\min_{\mathbf{z}_i} \quad \phi = \mathbf{c}_i^T \mathbf{z}_i \tag{2.2a}$$

$$s.t. \quad \mathbf{G}_i \mathbf{z}_i \geq \mathbf{h}_i \tag{2.2b}$$

Dantzig-Wolfe decomposition builds on the theorem of convex combinations

Theorem 1. *Let $\mathcal{Z} = \{\mathbf{z} \in \mathbb{R}^n \mid \mathbf{G}\mathbf{z} \geq \mathbf{h}\}$ with $\mathbf{G} \in \mathbb{R}^{m \times n}$ and $\mathbf{h} \in \mathbb{R}^m$ be nonempty, closed and bounded, i.e. a polytope. The extreme points of $\mathcal{Z}$ are denoted $\mathbf{v}^j$ with $j \in \{1, 2, ..., M\}$.*

Then any point $\mathbf{z}$ in the polytopic set $\mathcal{Z}$ can be written as a convex combination of extreme points

$$\mathbf{z} = \sum_{j=1}^{M} \lambda_j \mathbf{v}^j \tag{2.3a}$$

$$s.t. \quad \lambda_j \geq 0, \qquad j = 1, 2, ..., M \tag{2.3b}$$

$$\sum_{j=1}^{M} \lambda_j = 1 \tag{2.3c}$$

Proof. See [Dantzig and Thapa, 2002] □

Using the theorem on (2.2) and substituting it into (2.1) yields

$$\min_{\lambda} \quad \phi = \sum_{i=1}^{P} \sum_{j=1}^{M_i} f_{ij} \lambda_{ij} \tag{2.4a}$$

$$s.t. \quad \sum_{i=1}^{P} \sum_{j=1}^{M_i} \mathbf{p}_{ij} \lambda_{ij} \geq \mathbf{g} \tag{2.4b}$$

$$\sum_{j=1}^{M_i} \lambda_{ij} = 1, \qquad i = 1, 2, ..., P \tag{2.4c}$$

$$\lambda_{ij} \geq 0, \qquad i = 1, 2, ..., P;\ j = 1, 2, \ldots, M_i \tag{2.4d}$$

With M_i being the number of extreme points of subproblem i. f_{ij} and p_{ij} are defined as

$$f_{ij} = \mathbf{c}_i^T \mathbf{v}_i^j \tag{2.5a}$$

$$\mathbf{p}_{ij} = \mathbf{F}_i \mathbf{v}_i^j \tag{2.5b}$$

Equation (2.4) is denoted the Master Problem. The idea is to only generate the extreme points needed for the optimisation instead of generating all extreme points which can be

even more computationally complex due to the size of the problem. Assuming that an initial feasible solution is available for (2.4), a Reduced Master Problem can be set up and expanded through iteration with more extreme points. At iteration l the Reduced Master Problem is defined as

$$\min_{\lambda} \quad \phi = \sum_{i=1}^{P} \sum_{j=1}^{l} f_{ij} \lambda_{ij} \tag{2.6a}$$

$$s.t. \quad \sum_{i=1}^{P} \sum_{j=1}^{l} \mathbf{p}_{ij} \lambda_{ij} \geq \mathbf{g} \tag{2.6b}$$

$$\sum_{j=1}^{l} \lambda_{ij} = 1, \qquad i = 1, 2, ..., P \tag{2.6c}$$

$$\lambda_{ij} \geq 0, \qquad i = 1, 2, ..., P;\ j = 1, 2, \ldots, l \tag{2.6d}$$

in which $l \leq M_i$ for all $i \in \{1, 2, \ldots, P\}$. Obviously, the Reduced Master Problem can be regarded as the Master Problem with $\lambda_{i,j} = 0$ for $j = l + 1, \ldots, M_i$ and all $i \in \{1, 2, \ldots, P\}$.

Solving the Reduced Master Problem yields a Lagrange multiplier, π, for the coupling constraint (2.6b). This can be interpreted as a 'price' for the portfolio deviation. New extreme points are generated by solving subproblems defined as

$$\min_{\mathbf{z}_i} \quad \phi = \left[\mathbf{c}_i - \mathbf{F}_i^T \pi\right]^T \mathbf{z}_i \tag{2.7a}$$

$$s.t. \quad \mathbf{G}_i \mathbf{z}_i \geq \mathbf{h}_i \tag{2.7b}$$

for $i \in \{1, 2, \ldots, P\}$. These originate from (2.2), but the objective function is updated with $-\mathbf{F}_i^T \pi$ where π is given by the Reduced Master Problem in order to generate different extreme points based on the updated price. $\mathbf{F}_i$ is the effect the effectuator will have on the portfolio output.

The algorithm will then iterate over these steps until convergence at the global optimum is reached. One of the strengths of Dantzig-Wolfe decomposition is that there is a well-defined stop criterion, and convergence is ensured. For a thorough description of the algorithm, see [Edlund and Jørgensen, nd].

Algorithm 1 summarises the Dantzig-Wolfe Algorithm for solution of the block-angular linear program (2.1). The subproblems (2.9) may be solved in parallel. This is advantageous when the number of subproblems, P, is large.

One of the properties of the Dantzig-Wolfe decomposition is that the computationally complexity grows linearly with the number of effectuators in control, and thus good scalability is ensured when compared to a centralised solution which would typically grows cubically with the problem size. However, it is still a numeric algorithm, and the exact execution time of the algorithm cannot be guaranteed as it is dependent on the system states. However, the Dantzig-Wolfe Algorithm preserves feasibility of (2.1) at each iteration, as well as having a monotonically decreasing performance function. In predictive control applications, this implies that the algorithm can be stopped prematurely and the output of the last iteration will be a suboptimal solution that can be applied to the system without violating the constraint. Thereby it is ensured that a solution can be found within

Algorithm 1 The Dantzig-Wolfe Algorithm for a Block-Angular LP (2.1).

1: Compute a feasible vertex of the Master Problem (2.4). If no such point exists the original problem is infeasible, so stop.
2: $l = 1$, Converged = false.
3: **while** Not Converged **do**
4: Solve the l'th Reduced Master Problem, RMP(l):

$$\min_{\lambda_{ij}} \quad \phi = \sum_{i=1}^{P} \sum_{j=1}^{l} f_{ij} \lambda_{ij} \tag{2.8a}$$

$$s.t. \quad \sum_{i=1}^{P} \sum_{j=1}^{l} \mathbf{p}_{ij} \lambda_{ij} \geq \mathbf{g} \tag{2.8b}$$

$$\sum_{j=1}^{l} \lambda_{ij} = 1 \qquad i = 1, 2, \ldots, P \tag{2.8c}$$

$$\lambda_{ij} \geq 0 \qquad i = 1, 2, \ldots, P;\ j = 1, 2, \ldots, l \tag{2.8d}$$

and let π be the computed Lagrange multiplier associated to the linking constraint (2.8b). Let ρ_i be the computed Lagrange multiplier associated with (2.8c).
5: Solve all the subproblems ($i \in \{1, 2, \ldots, P\}$)

$$\min_{\mathbf{z}_i} \quad \phi_i = \left[\mathbf{c}_i - \mathbf{F}_i \pi\right]^T \mathbf{z}_i \tag{2.9a}$$

$$s.t. \quad \mathbf{G}_i \mathbf{z}_i \geq \mathbf{h}_i \tag{2.9b}$$

and let $(\psi_i, \mathbf{v}_i^{l+1}) = (\phi_i^*, \mathbf{z}_i^*)$ be the optimal value-minimiser pair.
6: **if** $\psi_i - \rho_i \geq 0\ \forall i \in \{1, 2, \ldots, P\}$ **then**
7: Converged = true. The optimal solution is

$$\mathbf{z}_i^* = \sum_{j=1}^{l} \lambda_{ij} \mathbf{v}_i^j \qquad i = 1, 2, \ldots, P \tag{2.10}$$

8: **else**
9: Compute the coefficients for the new columns in the RMP

$$f_{i,l+1} = \mathbf{c}_i^T \mathbf{v}_i^{l+1} \tag{2.11a}$$

$$\mathbf{p}_{i,l+1} = \mathbf{F}_i \mathbf{v}_i^{l+1} \tag{2.11b}$$

10: $l \leftarrow l + 1$
11: **end if**
12: **end while**

the sample time. Under mild condition this implies that stability can be guaranteed, even if the algorithm is stopped prematurely [Scokaert et al., 1999].

2.1.2 Communication and Data Encapsulation

When applying a Dantzig-Wolfe decomposition, the Master Problem and the subproblems are defined using the exact same structure as shown in Figure 2.1. The Reduced Master Problem (2.6) is solved in the upper layer, and the subproblems (2.7) are solved in the lower layer of the hierarchy.

The Dantzig-Wolfe algorithm grows almost linearly as a function of the number of subproblems, rather than cubically when solving one centralised problem as shown in [Edlund and Jørgensen, nd].

Currently, a standard Kalman filter is used for state estimation; it communicates the states to each subproblem and the Master Problem. The current solution means that the modelled units which are not in control need to send an output prediction at the beginning of each sample for use in the Master Problem. The communication between upper and lower layer is shown in Figure 2.3.

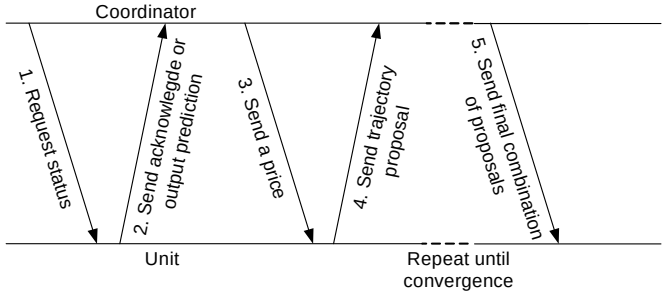


Figure 2.3: Communication timeline between coordinator and each unit during each sample.

The developed controller has an object-oriented structure with a clear interface between the layers and a clear communication scheme. The controller structure can be described as a UML diagram as shown in Figure 2.4. As long as the implementations of the effectuators adhere to the defined interface, the implementations can be chosen freely without having to change the framework. The interface is defined by the communication needs of the Dantzig-Wolfe decomposition.

If the information of one unit needs to be updated, it is easy to shut down this particular part of the controller, update it and set it back into control without having to shut down the entire controller. If the coordination layer needs maintenance, the controller will clearly lose its ability to minimise the deviation, but the communication with the effectuators can be maintained, and thus information of the states and input can be maintained.

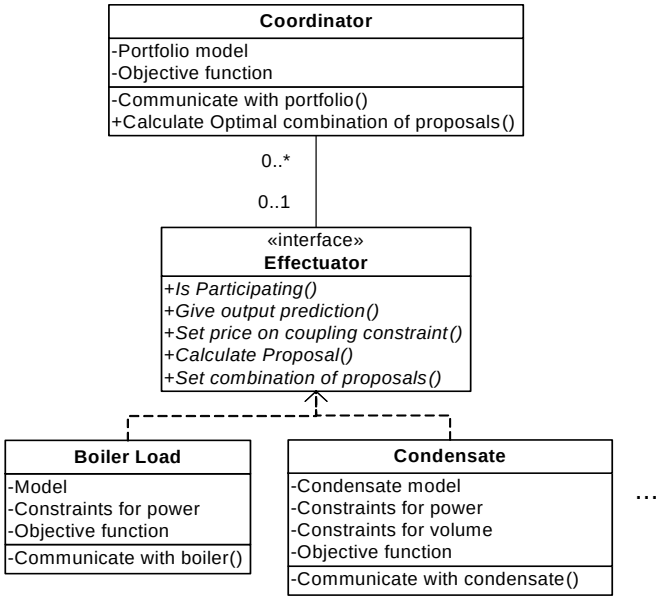


Figure 2.4: UML diagram of the controller structure. The defined interface allows for a flexible implementation of the specific effectuators.

2.2 Specific Controller Implementation

In the current system, only boiler load units are available for control purposes, and the specific implementation in this thesis is limited to include those; however, other effectuators can be included in a straightforward manner.

As outlined above, the individual boilers can be modelled separately, as the actions in one boiler do not affect the other boilers. They are only coupled through the objective to follow the overall portfolio reference and activating secondary resources. A constrained linear model for each boiler is derived in the following, along with a performance function for each boiler.

2.2.1 Boiler load units

In the current controller there are between 0 and 6 power plant units in control. These will all be modelled in a similar fashion.

The boiler load effectuator is activated by offsetting the production reference. The boiler has an operating range, shown in the PQ diagram in Fig 2.5. The district heating production (Q) is plotted along the x-axis and the power production (P) along the y-axis. There are upper and lower limits on the power production which depends on the current district heating production.

When using the boiler for control purposes, the district heating production is maintained, meaning that the changes in production happen vertically in the PQ-diagram as shown in Figure 2.5.

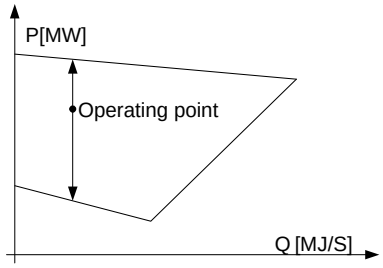


Figure 2.5: Movement in the PQ-diagram when changing the boiler load.

A simple model of the boiler has been derived in [Edlund et al., 2009b], but in order to fit it into the linear control scheme developed here, some assumptions must be made. The modelling concept is shown in Figure 2.6. The model derived here is for the use in the controller, and thus all constraints are formulated to fit into the controller which gives corrective signals to the boiler units.

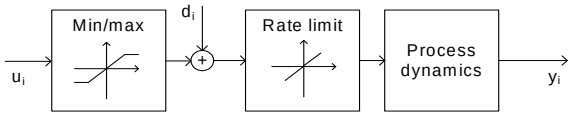


Figure 2.6: Concept of the boiler modelling.

The model has two input signals, d_i is the input signal coming from the production plan, and u_i is the input signal coming from the load balancing controller. Thus, in the nominal case, u_i is zero, since no corrective signals are needed.

The process dynamics is modelled as the third order system

$$H(s) = \frac{1}{(T_i s + 1)^3} \tag{2.12}$$

where T_i is the time constant of effectuator i.

In order to gain offset-free tracking, the linear models are augmented with a disturbance model under the assumption that the disturbance output is constant, so that the constrained augmented discrete time state space model becomes

$$\mathbf{x}_{i,+1k} = \begin{bmatrix} a_{1,1,i} & 0 & 0 & 0 \\ a_{2,1,i} & a_{2,2,i} & 0 & 0 \\ a_{3,1,i} & a_{3,2,i} & a_{3,3,i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{i,k} + \begin{bmatrix} b_{1,i} \\ b_{2,i} \\ b_{3,i} \\ 0 \end{bmatrix} \mathbf{u}_{i,k} + \begin{bmatrix} e_{1,i} \\ e_{2,i} \\ e_{3,i} \\ 0 \end{bmatrix} \mathbf{d}_{i,k} \tag{2.13a}$$

$$\mathbf{y}_{i,k} = \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix} \mathbf{x}_{i,k} \tag{2.13b}$$

$$\underline{u_i} \leq u_i \leq \overline{u_i} \tag{2.13c}$$

$$\max\{\underline{\Delta u_i} - \Delta d_i, 0\} \leq \Delta u_i \leq \min\{\overline{\Delta u_i} - \Delta d_i, 0\} \tag{2.13d}$$

The elements in $\mathbf{A}_i$, $\mathbf{B}_i$ and $\mathbf{E}_i$ are dependent on T_i and the sample time. Symbols with a bar beneath, e.g. $\underline{u}$ mean the lower bound, while $\overline{u}$ denotes the upper bound. The upper

rate of change constraint is modelled, so that it is always non-negative and vice versa, to avoid forcing the controller to take actions in case the production plan violates the rate of change constraint. The upper and lower limits for the controller (2.13c) are set in the control system by the operator.

The rate of change constraint is dependent on the boiler load. A typical form of the rate of change constraint as a function of the boiler load is depicted in Figure 2.7.

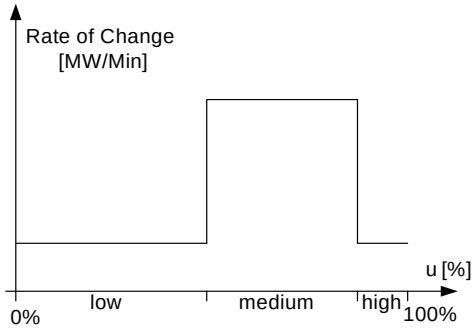


Figure 2.7: Actual rate of change constraint as a function of boiler load. This state dependency is not captured in the constraint (2.13d), but a linearisation based on the prediction is used in the model.

To linearise, the constraint the prediction of u is used to generate rate of change constraints throughout the prediction horizon. If no prediction of u exists, it is assumed to be zero.

In case the operator changes the upper or lower bound, so that the current control signal violates the limits, the limit is ramped down with the maximum allowed rate of change. This measure is taken to avoid infeasible optimisation problems.

2.2.2 Optimisation Problem for a Boiler Load Effectuator

The optimisation problem for each boiler unit is formulated as

$$\min_{\mathbf{U}_i} \quad \phi_i = \sum_{k=0}^{N-1} p_{i,k+1} y_{i,k} + ||y_{i,u,k+1}||_{1,q_{i,k+1}} + ||\Delta u_{i,k}||_{1,s_{i,k}} \tag{2.14a}$$

$$s.t. \quad \mathbf{x}_{i,k+1} = \mathbf{A}_i \mathbf{x}_{i,k} + \mathbf{B}_i u_{i,k} + \mathbf{E}_i d_{i,k}, \qquad k = 0, 1, \ldots, N-1 \tag{2.14b}$$

$$y_{i,k} = \mathbf{C}_i \mathbf{x}_{i,k}, \qquad k = 1, 2, \ldots, N \tag{2.14c}$$

$$\underline{u_{i,k}} \leq u_{i,k} \leq \overline{u_{i,k}}, \qquad k = 0, 1, \ldots, N-1 \tag{2.14d}$$

$$\underline{\Delta u_{i,k}} \leq \Delta u_{i,k} \leq \overline{\Delta u_{i,k}}, \qquad k = 0, 1, \ldots, N-1 \tag{2.14e}$$

where $\mathbf{U}_i = [u_{i,0}, u_{i,1}, ..., u_{i,N-1}]^T$, and $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i, \mathbf{E}_i$ given in (2.13a) and (2.13b).

The first term in the performance function $p_{i,k-1} y_{i,k}$ is a linear term representing the cost of the boiler unit. The weight $p_{i,k+1}$ is the marginal cost, i.e. the cost for producing energy in the boiler unit. The price is calculated based on the fuel prices and boiler efficiency. The efficiency is state-dependent. For the calculations of $p_{i,k+1}$, it is based on the production plan alone.

It is assumed that the production plan from the STLS is optimal, and thus in the nominal case the correction signal from the load balancing controller should be zero. In order to avoid the power controller maximise the production of the cheapest units, and minimise the production of the most expensive units the term $||y_{u,k-1}||_{1,q_{y,k-1}}$ is added. This term penalises the part of the output coming from the controller.

The last term of the performance function is a penalty on rapid changes on the correction signal.

This optimisation problem is the controller for unit i, and this information is stored in each of the effectuators on the lower layer of the hierarchy.

Primary Reserve Handling

Figure 2.8 shows an example of the maximum reserve available in both up and down direction as a function of the unit load. Reserves available for the positive and negative corrections are shown in the right and left half planes respectively.

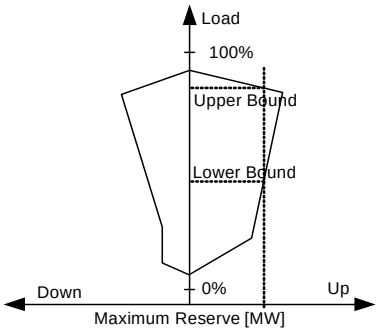


Figure 2.8: Primary Reserves as a function of unit load. On the y-axis is the unit load. On the x-axis the maximum possible primary reserve that can be delivered at the boiler load in both positive and negative direction. Dotted lines show a positive reserve reservation and the derived upper and lower input bounds for the controller. Similar reservation can be made at the same time for negative reserves.

Currently the Frequency Control Scheduler makes reservations of the reserves periodically. It means it can reserve 5MW of positive correction and 10MW of negative correction power from a specific unit at a given time.

It is chosen to give first priority to the Frequency Control Scheduler, and let it make the reservations. Once the reservations are known, the upper and lower bound for the unit can be determined, so that the reserved primary reserve can be delivered. These upper and lower bounds are enforced on the power controller along with upper and lower bounds set by the operator.

Automatic / Manual Control and Fall-back Strategies

The boiler load effectuators can be in either automatic control or manual control mode. As explained earlier, in automatic control mode the controller can give corrective control

signals to the unit. The units can also switch from manual to automatic control and vice versa. This event is assumed external and non-predictable; however, it is observable.

If the effectuator switches from automatic to manual control while $u \neq 0$ the strategy is to ramp the control signal toward zero with a predefined slope. This is done on both unit and in the controller, so in case of communication errors, the behaviour of the unit can be predicted. The same fall-back strategy is used in case of faults in the effectuator.

These fall-back strategies along with the control status are all handled in the lower layer of the hierarchy.

2.2.3 Portfolio Modelling

The portfolio is comprised of the boiler load units modelled previously and a mixture of other production units. These other production units consist of various small thermal power plants and some wind turbines. They have a production reference, and their production is measurable, but little is known about their dynamical behaviour. They are considered a disturbance in this context.

In order to include them in the controller, the model of the *other units* consists only of a disturbance model assuming that the output is constant, so that the other models are modelled as

$$x_{other,k+1} = x_{other,k} \tag{2.15a}$$

$$y_{other,k} = x_{other,k} \tag{2.15b}$$

The total portfolio output is then

$$y_{port,k} = y_{other,k} + \sum_{i=1}^{P} y_{i,k} \tag{2.16}$$

The optimisation problem for the portfolio is based on reference tracking and is given as

$$\min_{U} \phi = \sum_{k=1}^{N} \|y_{port,k} - r_{port,k}\|_{1,q_{port,k}} \tag{2.17a}$$

$$\tag{2.17b}$$

where r_{port} is the portfolio reference which is the sum of references to all units in the system plus the demand from the TSO as shown in Figure 1.4. k is the sample number and N is the prediction horizon.

This optimisation problem is placed in the upper layer of the hierarchy.

2.2.4 The Centralised Control Problem

The individual effectuators as well as the portfolio have been modelled, and the optimisation problem for each of them has been defined. The designed controller is equivalent to solving the centralised problem

$$
\begin{aligned}
\min \quad & \phi && && (2.18a)\\
s.t. \quad & \mathbf{x}_{i,k+1} = \mathbf{A}_i\mathbf{x}_{i,k} + \mathbf{B}_i\mathbf{u}_{i,k} + \mathbf{E}_i\mathbf{d}_{i,k}, && i = 1,2,...,P && (2.18b)\\
& \mathbf{y}_{i,k} = \mathbf{C}_i\mathbf{x}_{i,k}, && i = 1,2,...,P && (2.18c)\\
& \underline{\mathbf{u}_{i,k}} \leq \mathbf{u}_{i,k} \leq \overline{\mathbf{u}_{i,k}}, && i = 1,2,...,P && (2.18d)\\
& \underline{\Delta\mathbf{u}_{i,k}} \leq \Delta\mathbf{u}_{i,k} \leq \overline{\Delta\mathbf{u}_{i,k}}, && i = 1,2,...,P && (2.18e)\\
& x_{other,k+1} = x_{other,k} && && (2.18f)\\
& y_{other,k} = x_{other,k} && && (2.18g)
\end{aligned}
$$

with

$$
\begin{aligned}
\phi = & \sum_{k=1}^{N} \left\| y_{other,k} + \sum_{i=1}^{P} y_{i,k} - r_{port,k} \right\|_{1,q_{port,k}} + \\
& \sum_{i=1}^{P} \left[\sum_{k=0}^{N-1} p_{i,k+1} y_{i,k} + ||y_{i,u,k+1}||_{1,q_{i,k+1}} + ||\Delta u_{i,k}||_{1,s_{i,k}} \right]
\end{aligned}
\tag{2.19}
$$

This optimisation problem can be rewritten into a linear program with the structure of (2.1).

2.2.5 Finding an Initial Feasible Solution

Step 1 of Algorithm 1 states that an initial feasible solution needs to be calculated. This can in all cases be done by using a Phase I simplex algorithm as described in [Edlund and Jørgensen, nd]. It should be noted that computating a feasible vertex of (2.4), may be just as expensive as computing the optimal solution. Therefore, if a feasible vertex is readily available, it should be used directly instead of applying a phase I simplex procedure.

Rewriting (2.17) into a linear program will add an extra set of decision variables to the Master Problem called $\mathbf{z}_{tot}$. These variables act similar to slack variables in the sense that if they are large enough, the problem will become feasible. In this case, it means that if a feasible solution can be found to all subproblems, a feasible solution to the Master Problem exists.

The task of finding an initial feasible solution to the Master Problem is thereby reduced to finding a feasible solution to all subproblems with $\pi = \mathbf{0}$. Once a solution to all subproblems are found, $\mathbf{z}_{tot}$ must fulfil $z_{tot,k} \geq |y_{other,k} + \sum_{i=1}^{P} y_{i,k} - r_{port,k}|$. Since the right hand side is known, finding a solution for this inequality is trivial and result in an initial feasible solution to the Master Problem.

2.3 Simulations

In order to evaluate the new load balancing controller, it will be tested against the currently running controller through simulation in a scenario stretching throughout a month of real operation. Figure 2.9 shows the simulated system. The simulated system consists

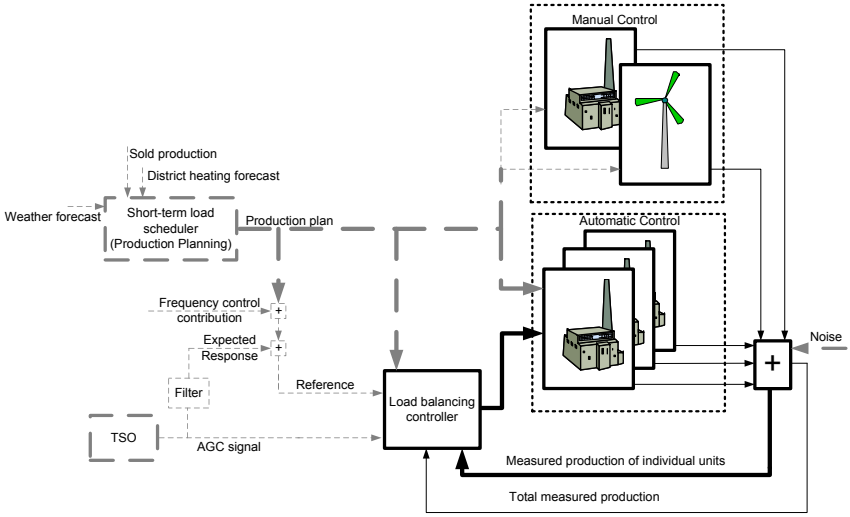


Figure 2.9: Simulation of the load balancing controller on the system level. The bold lines show vectors of signals. The dashed lines show signals with boundary conditions for the simulation. The portfolio is a simulation model.

of the load balancing controller and models of the power plants containing boiler load effectuators. All dashed lines, i.e. the signals from the short-term load scheduler and TSO are boundary conditions for the simulation.

The current controller is implemented in Simulink^{TM}([Mathworks, 2010]) and compiled so it is able to be executed in the central control room. In other words, it is the *actual* controller and not some simplified implementation of the controller the comparison is performed against. In order to test the new developments and maintenance on the current controller, models have been developed in Simulink^{TM}to be able to test the whole system. Since a test environment already exists, it is an obvious choice to make the comparison in Simulink^{TM}.

The new controller is implemented in mixture of Java for all the data handling such as reading measurement data and constructing constraints, and Matlab^{TM}[Mathworks, 2010] for solving the optimisation problem.

The dynamic part of the boiler unit models are implemented as linear models or linear parameter varying models. Besides the dynamics of the boiler unit, parts of the control system operating the boiler unit have been implemented. It means that all upper/lower bounds, rate of change constraints, correction for district heating and parasitic consumption are implemented in the models along with a lot of the logic controlling the switch from manual to automatic mode and vice versa.

The simulation environment runs at the same sample time as the current controller, i.e. 0.5s, and since the sample rate of the newly developed controller is 5s, a ZOH approach will be taken. The data is saved with a 5s sample time for both controllers for analysis purposes.

Simulations cover 25-hour sequences start from 23:00 to midnight the following day. In the analysis section, the first hour is discarded, so the analysis covers 24-hour se-

quences from midnight to midnight. The first hour is used to avoid start-up and settling issues influencing the analysis, allowing to string together several sequences for more extensive analysis.

In this thesis only the main results from a noisy scenario are treated for full details, see [Edlund et al., nd]. For each boiler unit the input and output sample sequences of the original measured scenario are known. One can thus estimate a noise sequence for the scenario as

$$y_n = y_{meas} - y_{sim} \tag{2.20}$$

This noise is applied to the output of the model of the boiler unit. Since the noise is generated based on closed loop measurements, it is likely filtered by the controller in the loop rather than white noise.

This noise generation is chosen, so that the simulation scenario resembles the actual scenario as closely as possible including failures. The measurements from the units modelled as the portfolio are applied directly to the simulation without filtering.

The analysed scenario contains three entities. There are the measurements from the actual operation, a simulation of the current controller and a simulation of the new controller.

For the simulations standard deviation and mean error are used as quantitative measurements for the evaluation. Figure 2.10 shows the mean error

$$\mu = \frac{1}{N_s} \sum_{k=1}^{N_s} y_{port,k} - r_{port,k} \tag{2.21}$$

with N_s being the number of samples in the simulation. Figure 2.11 shows the standard deviation

$$\sigma = \sqrt{\frac{1}{N_s} \sum_{k=1}^{N_s} ((y_{port,k} - r_{port,k}) - \mu)^2} \tag{2.22}$$

on a daily basis. Analysis shows that the constraints from primary reserves limit the controller in periods.

The standard deviation on the new controller is higher than the current controller. A significant change in the current and the new controller is that the new controller adheres to fulfilling the primary reserve reservations at all times. This adds an extra constraint to the controller which is active for longer periods and this degrades the performance of the new controller. A comparison of of the new controller with and without the primary reserve constraint on a noise-free scenario is found in [Edlund et al., nd]. Both are significantly higher than the measurement data which are likely to be caused by the noise generation scheme as shown in Figure 2.11. The trends in standard deviation is the same for both controllers and measurement data. As seen in the noise-free scenario, the current has a slightly better performance than the proposed.

The mean error is larger in the noisy scenario compared to the noise-free. Though not consistently lower, the average shown in Table 2.1 shows that the new controller is an order of magnitude closer to zero mean error compared to the current controller.

Figure 2.12 shows the price difference between the two controllers. Analysing the price shows that on most days the new controller performs better. On day 20, the primary reserves limit the controller, so that a large deviation occurs over a long period of time

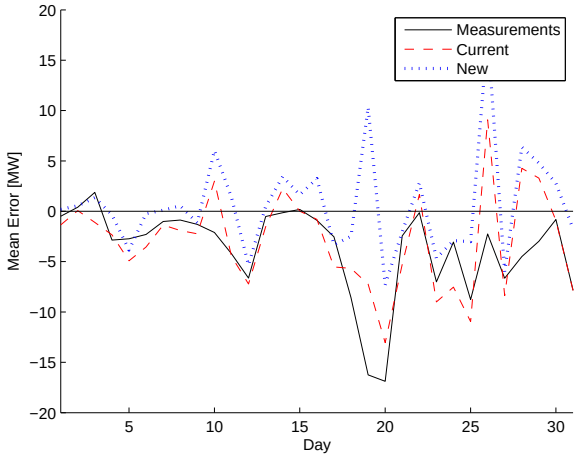


Figure 2.10: 24-hour mean error for the controllers in a noisy scenario. The figure shows the measurements (solid), the current controller (dashed) and the new controller with (dotted). The results are for the individual days. Day 19 is omitted from the analysis due to missing measurement data for the scenario.

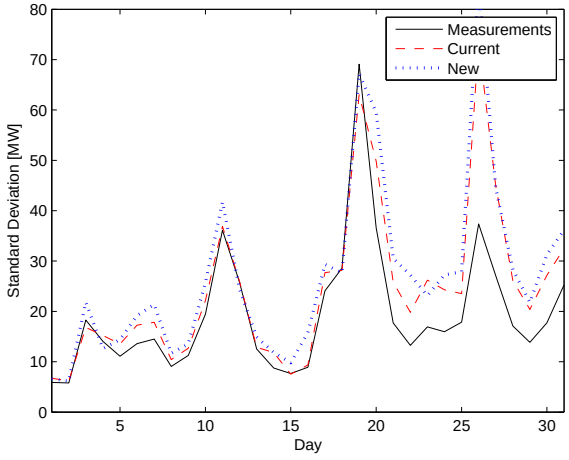


Figure 2.11: 24-hour standard deviation for the controllers in a noisy scenario. The figure shows the measurements (solid), the current controller (dashed) and the new controller (dotted). The results are for the individual days. Day 19 is omitted from the analysis due to missing measurement data for the scenario.

	σ	μ
Measurements	17.74	-3.27
Current	23.11	-2.78
New	25.72	0.29

Table 2.1: Standard deviation and mean throughout the whole month of simulation. Measurements are the measured values with the controller running at that time. Current is a simulation with the current controller and new is the simulation with the new controller.

which is detrimental for the earnings of the controller. On average the difference is 240 €per day, which means an earning of almost 90,000 €per year.

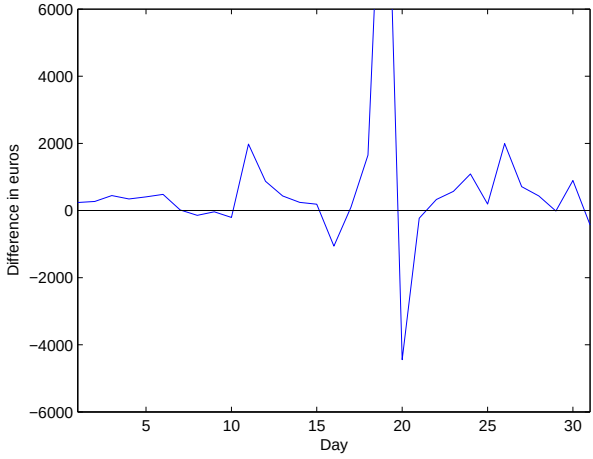


Figure 2.12: Price difference between the current controller and the new controller. Positive difference means that the new controller is cheaper (earns more money for DONG Energy).

2.4 Fulfilling the Design Criteria

Three design criteria has been established which the new controller needs to meet in order to fulfil the predicted demands from such a controller.

Scalability Through the application of Dantzig-Wolfe decomposition, it is shown experimentally in [Edlund and Jørgensen, nd] that the computational complexity of the controller grows almost linearly with the the number of effectuators while still converging to the same optimum as the centralised solution. This is a significant improvement in lowering the computationally complexity over the centralised solution. It is also shown that already at 2-3 effectuators, the Dantzig-Wolfe decomposition is faster than the centralised solution. Furthermore, it is possible to

distribute the optimisation problem among multiple processors, giving an advantage which exploits the trends toward computers with multiple processors.

Flexibility The design method fulfils the objective of flexibility through an object-oriented design with data encapsulation and clear interfaces. It ensures that the controller is easily maintainable in case of updates of the controller such as adding and removing effectuators.

Performance The design method itself does not ensure that the performance criterion is met, so that a controller design with the developed method performs as well as the current controller. However, the developed method ensures that if an ℓ_1-norm based MPC can be constructed to fulfil the performance criterion, the design hierarchical design will also fulfil the criterion. In Section 2.3 as well as [Edlund et al., 2008, nd], simulations show that an ℓ_1-norm based MPC can be constructed, which improves the performance in terms of standard deviation and mean value, as well as improve the economic performance by a better distribution of the control actions.

The scalability and flexibility criteria have both been treated and fulfilled by the design method, while the performance criterion is dependent on the specific implementation. The design method has been utilised for controller synthesis for the current power plant portfolio and the resulting controller fulfills the performance criterion. Thus, all criteria established in the hypothesis can be fulfilled by the design method.

3 | Summary of Contributions

The main contributions of this thesis consist of six papers regarding different aspects of the portfolio control. This chapter summarises the contributions made in the project. The papers are not included in a chronological order, but in an order that takes the reader from the motivation to the solution in a logical way.

3.1 Stability of the Current Controller

In [Edlund et al., 2009a] the current load balancing controller structure was analysed. [Wood and Wollenberg, 1996] give the structure of an automatic generation controller which is used for balance control by the TSOs and is very similar in structure to the load balancing controller treated in this thesis. The stucture in [Wood and Wollenberg, 1996] is expanded to include rate of change constraints which are required for the controller to meet the requirements for operating in the western Denmark.

A linear approximation of the implemented structure is shown in Figure 3.1.

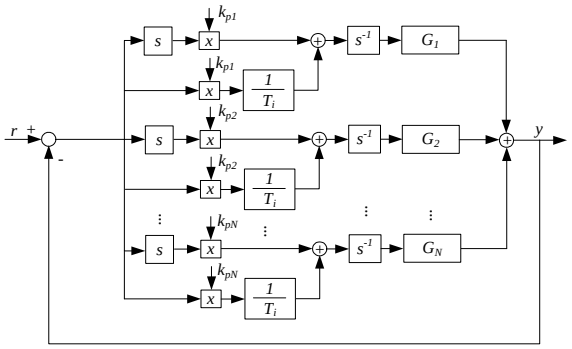


Figure 3.1: Linear approximation of the load balancing controller structure.

Using the definition of internal stability [Zhou et al., 1996; Skogestad and Postlethwaite, 2005], it is proven that the current controller structure is internally unstable. The instability cannot be seen in the portfolio output, but result in the effectuators drifting away from the production plan in opposite directions. The instability can be shown through numeric simulations as well as in real data as shown in Figure 3.2. In the figure two

effectuators are shown which at 3 hours drift away from the optimal production plan and stay there for many hours.

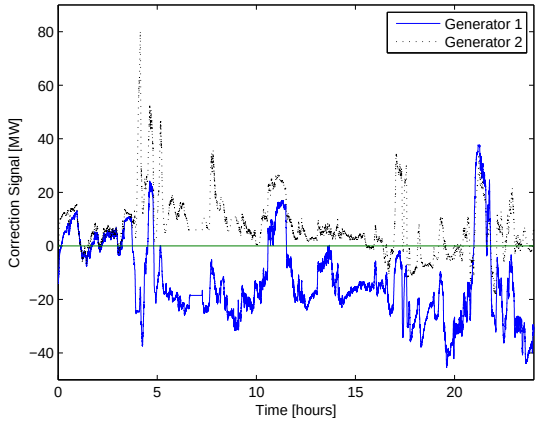


Figure 3.2: Correction signals from the controller. The correction signals drift apart and stay on opposite sides of the optimal production plan for several hours.

[Edlund et al., 2009a] give two proposals for salvaging the controller. One of them is a stronger parallel run which forces all effectuators towards the same correction. Parallel run is commonly used within the power plant industry when two or more similar subsystems have to work in parallel to complete a task. When using PI-controllers in parallel, the subsystems will in general not contribute equally when equilibrium is reached. In order to obtain this behaviour, parallel run is introduced. The parallel run drives the integrators towards the same value, thereby ensuring that the behaviour seen in the example above, where two generators drift in opposite directions, is avoided. However, increasing the strength of the parallel run will decrease the ability to make fast changes. The other proposed solution is to make a smarter distribution of the amount of correction signal each effectuator receives. The latter has been implemented in the current system as a result of this paper.

3.2 Showing MPC is Viable for Portfolio Control

The first contribution towards developing a design method is in [Edlund et al., 2008]. A Model Predictive controller with the structure

$$\min_{\mathbf{u}} \quad \phi = \sum_{k=0}^{N-1} \|\mathbf{y}_{k+1} - \mathbf{r}_{k+1}\|_{1,q_{e,k+1}} + \mathbf{q}_{u,k+1}^T \mathbf{y}_{k+1} + \|\Delta \mathbf{u}_k\|_{1,q_{\Delta u,k}} \tag{3.1a}$$

$$s.t. \quad \mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{E}\mathbf{d}_k, \qquad k = 0, 1, \ldots, N \tag{3.1b}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k, \qquad k = 1, 2, \ldots, N \tag{3.1c}$$

$$\underline{\mathbf{u}_k} \leq \mathbf{u}_k \leq \overline{\mathbf{u}_k}, \qquad k = 0, 1, \ldots, N \tag{3.1d}$$

$$\underline{\Delta \mathbf{u}_k} \leq \Delta \mathbf{u}_k \leq \overline{\Delta \mathbf{u}_k}, \qquad k = 0, 1, \ldots, N \tag{3.1e}$$

is proposed to to improve the control compared with the currently implemented PI-controller. (3.1b) and (3.1c) is a MIMO state space model of the whole portfolio including the effectuators, so that $y = [y_1, y_2, \ldots, y_P, y_{port}]$ with y_i being output from effectuator $i = 1, 2, \ldots, P$ and y_{port} being the portfolio output. The first term of (3.1a) describes the deviation between output and reference both for the whole portfolio and for the individual effectuators. The second term is an economic term measuring the cost of producing power on each plant. The third term smooths the solution by penalising rapid movements in the manipulated variables, thereby avoiding wear on the effectuators.

Simulations showed that this controller is capable of lowering the deviation as shown in Figure 3.3 as well as increasing the income of the portfolio.

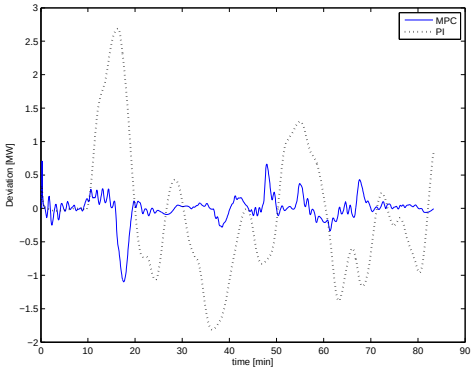


Figure 3.3: Portfolio deviation comparison between a PI-controller based and an MPC based load balancing controller.

Besides the contribution of formulating an MPC-based controller, the introduction of economic terms in the portfolio controller is a novel approach. From distributing the control actions to all effectuators based on possible rate of change to an economic distribution is a significant change of behaviour. This can be seen in the input signals in Figure 3.4. This contribution has, parallel to this project, been successfully implemented in a rule-based logic scheme in the current PI-controller.

[Rao and Rawlings, 2000] demonstrate the Model Predictive Controllers containing ℓ_1-norm penalty functions may give rise to either dead-beat or idle control. While theoretically (3.3) result in this behavior, simulations in [Edlund et al., 2008] demonstrate that the controller performs well and provides the desired portfolio control.

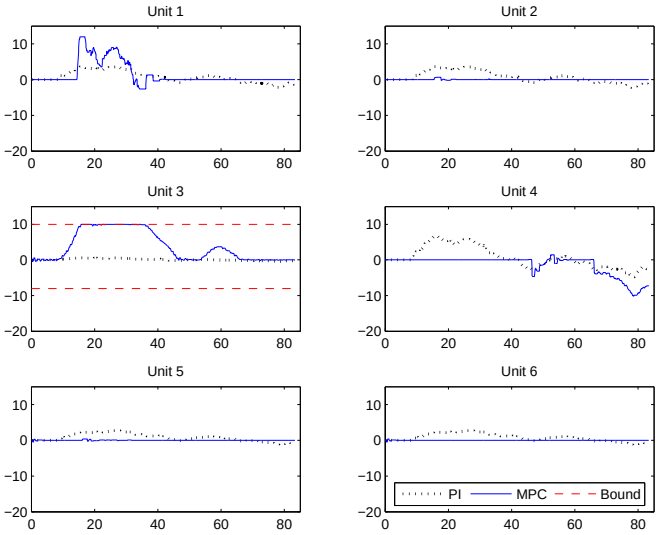


Figure 3.4: Comparison of input signals between the PI-controller and the MPC. The MPC controller distribution is based on economic terms rather than a share based on the rate of change capability.

3.2.1 Developing Models for Effectuators

The effectuator model used for MPC in [Edlund et al., 2008] was based on assumptions that the current model in the simulation environment used for the currently implemented controller is an adequate description of the system dynamics. And the results of simulations with a controller based on these models showed good performance. In [Edlund et al., 2009b] simple dynamic models with constraints of several effectuators were derived and validated against available measurement data.

Three of the derived effectuator models are located in a power plant, and their dynamics affect the other models constraint or dynamics. However, the effect of the cross couplings induced by the control actions is so small that it can be ignored by the controller, but the model parameters likely needs to be tracked and updated outside the controller.

In [Edlund et al., 2008] all power plant units were considered to have no district heating production. The derived model for the boiler load effectuator in [Edlund et al., 2009b] had dynamics corresponding to the dynamics of the models used in [Edlund et al., 2008], but the constraints were modelled as a function of the district heating production.

The fourth derived effectuator model describes a wind turbine farm which can also be used for load balancing.

3.3 Hierarchical Controller Structure

In order to obtain a modular and flexible controller, a hierarchical approach has been taken in [Edlund et al., nd] as shown in Figure 2.1. The developed control structure is

a two-layer hierarchical structure designed with an object-oriented design approach and can be represented using UML diagrams as in Figure 2.4.

One important design decision is the usage of an interface between the upper and lower layer of hierarchy. The interface allows multiple implementations of effectuators without changing the design framework. The interface is designed as minimalistic as possible to allow as much freedom to the implementation of effectuators as possible. The structure has a high coherence inside each module and a low coupling. This, coupled with the usage of the interface, ensures a modular and flexible controller.

The hierarchical controller design has the following requirements to the lower layer controller implementation

- The effectuators must be modelled as independent, so that a change in one effectuator does not directly affect another effectuator.
- The underlying optimisation problem can be stated as a linear program, which in terms means the objective function consist of linear and ℓ_1-norm terms and linear constraints.
- The interface requires that the controller can generate control proposals for the effectuator based on an updated price from the upper layer.

All effectuators that can be fitted into these requirements can be used in the control structure without changing the design framework, thereby giving the design flexibility to incorporate future effectuators, such as a virtual power plant or electric vehicles.

The hierarchical controller structure relies on a decomposition of the underlying optimisation problem to ensure the same optimality conditions as the centralised version of the controller.

3.4 Efficient Solution of Optimisation Problems

In [Edlund et al., 2008] an MPC was derived, and the results obtained with the controller were good. However, it was clear that the time needed for finding a solution was critical, and the chosen centralised implementation has a computation time on a normal PC close to the sample time. Even if faster hardware is utilised, the complexity grows cubically with the problem size, and thus the number of effectuators that can be added to the controller is limited.

Two contributions for lowering the computationally complexity are considered. A hierarchical decomposition using Dantzig-Wolfe is proposed in [Edlund and Jørgensen, nd], and a customisation of an interior point solver for the problem [Edlund et al., 2009c]. Both contributions exploit the structure of the problem to reduced the required computations.

3.4.1 Dantzig-Wolfe Decomposition

Dantzig-Wolfe decomposition has been used for decomposing large-scale linear problems in optimisation for a long time. However, using it for the dynamical calculations of a model predictive controller is a novel approach.

The benefits of using Dantzig-Wolfe decomposition are that the hierarchical structure is possible to implement with a clear interface with a small level of communication. Besides the clear interface design, it also makes it possible to spread the optimisation over multiple computers connected by a network, without much overhead.

Dantzig-Wolfe decomposition also ensures better scalability. Scalability in computationally complexity is shown in Figure 3.5.

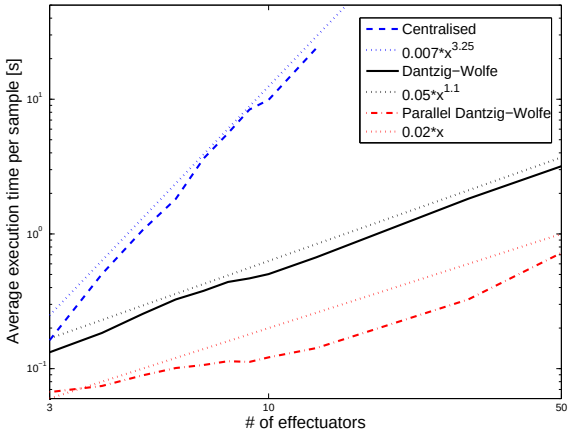


Figure 3.5: Computation time as a function of the effectuators in the optimisation.

Computation time grows almost linearly with the number of effectuators compared to the cubic growth of a centralised solution. This coupled with the algorithm being faster with a small number of effectuators makes MPC a viable controller scheme for the load balancing controller.

Theoretically, the Dantzig-Wolfe decomposition converges very slowly, the test problem in [Edlund and Jørgensen, nd], and the simulation on a realistic scenario [Edlund et al., nd] shows good convergence in practice.

3.4.2 Customising an Interior Point Method

The other contribution toward increasing the efficiency of the optimisation algorithm was customising an interior point method which is reported in [Edlund et al., 2009c]. The interior point method is tailored to solve the local optimisation related to one of the boiler load effectuators. However, many of the techniques applied can be reused in MPC applications.

The optimisation problem for a single boiler load effectuator

$$\min_{u} \quad \phi = \sum_{k=0}^{N-1} \|y_{k+1} - r_{k+1}\|_{1,q_{e,k+1}} + q_{u,k+1}^T y_{k+1} + \|\Delta u_k\|_{1,q_{\Delta u,k}} \tag{3.2a}$$

$$s.t. \quad \mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k + \mathbf{E}\mathbf{d}_k, \qquad k = 0, 1, \ldots, N-1 \tag{3.2b}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k, \qquad k = 1, 2, \ldots, N \tag{3.2c}$$

$$\underline{\mathbf{u}_k} \leq \mathbf{u}_k \leq \overline{\mathbf{u}_k}, \qquad k = 0, 1, \ldots, N-1 \tag{3.2d}$$

$$\underline{\Delta\mathbf{u}_k} \leq \Delta\mathbf{u}_k \leq \overline{\Delta\mathbf{u}_k}, \qquad k = 0, 1, \ldots, N-1 \tag{3.2e}$$

can be rewritten into a linear program with the form

$$\min_{x} \quad \phi = \mathbf{c}^T\mathbf{z} \tag{3.3a}$$

$$s.t. \quad \mathbf{G}\mathbf{z} \geq \mathbf{h} \tag{3.3b}$$

by rewriting the optimisation problem using different rules such as the model description as given in (1.9).

The program has a constraint matrix, G, with the structure

$$\mathbf{G} = \begin{bmatrix} \mathbf{I} & \mathbf{0} & \mathbf{0} \\ -\mathbf{I} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \\ \mathbf{\Psi} & \mathbf{0} & \mathbf{0} \\ -\mathbf{\Psi} & \mathbf{0} & \mathbf{0} \\ \mathbf{\Psi} & \mathbf{I} & \mathbf{0} \\ -\mathbf{\Psi} & \mathbf{I} & \mathbf{0} \\ \mathbf{\Gamma}_u & \mathbf{0} & \mathbf{I} \\ -\mathbf{\Gamma}_u & \mathbf{0} & \mathbf{I} \end{bmatrix} \tag{3.4}$$

The individual elements of this matrix are defined in [Edlund et al., 2009c]. The important thing to note is that the constraint matrix is highly structured, i.e. many zero entries, and this can be exploited.

One of the most costly operations in the interior point solver in [Edlund et al., 2009c] is the calculation of $\overline{\mathbf{H}} = \mathbf{G}^T\mathbf{D}\mathbf{G}$, where $\mathbf{D}$ is a diagonal matrix. Performing this calculation with normal matrix multiplication is very expensive. Implementing the interior point algorithm in Matlab™, 79% of the time was spent on this operation when implemented with standard notation. Using Matlab's sparse techniques, a significant speed-up of the algorithm was obtained, but the most expensive operation was still the calulation of $\overline{\mathbf{H}}$ with 55% of the computation time. Exploiting the structure of the matrix along with a few other customisations of the algorithm gave a speed-up of the algorithm of approximatly 20 times compared to the standard nonsparse implementation.

Comparing the algorithm in Figure 3.6 with an active set solver implemented in FORTRAN and Matlab's Linprog shows a significant improvement compared to other algorithms handling general problems.

The important contribution is not that it is possible to speed up a solver for this specific problem, but most optimisation problems arising from MPC have a highly structured constraint matrix and thus can be exploited in a similar way.

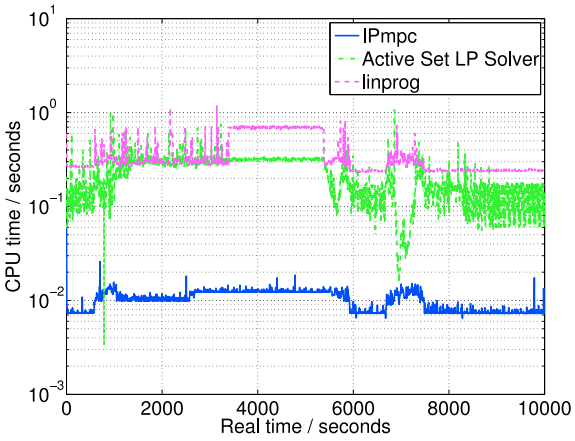


Figure 3.6: CPU times for the different LP algorithms.

Dantzig-Wolfe Decomposition relies on the subproblems to generate solutions in a vertex of the feasible area. In the rare case that a solution to the subproblem exists on an edge of the feasible area, the interior point method might not converge to a vertex. So using the customised interior point method to efficiently solve the subproblems generated by Dantzig-Wolfe decomposition could lead to a Dantzig-Wolfe algorithm which will not converge to an optimum. In practice this has not proved to be a problem.

3.5 Implementation and Benchmarking of the Controller

Throughout the project, implementations of the controller have been made. The first implementation is reported in [Edlund et al., 2008] with the purpose of showing that MPC is a viable option for a control scheme as described in Section 3.2.

In [Edlund et al., nd] the design method was used to synthesis of a controller for use in the current system. The performance of the controller designed with the developed method was compared based on three factors; mean error, standard deviation and price. The comparison was performed through a simulation in a scenario based on data from a month of actual operation.

The new method incorporates a method to ensure that the load balancing controller does not violate the primary reserve constraints, something that has not been handled previously.

Table 3.1 shows a comparison of the new controller compared with the current implementation. There are three tested entities; there are the measurements from the actual operation, a simulation of the currently implemented controller and a simulation of the proposed new controller. The new controller includes constraints to ensure that primary reserves are maintained. For comparison purposes the new controller has been simulated without this constraint as well.

In both the noise-free and the noisy scenario, the new controller performs close to but slightly worse than the current controller. The hypothesis was that it was caused by the constraints enforced by the primary reserves. The comparison with the new controller

	Noiseless		Noise	
	σ	μ	σ	μ
Measurements	-	-	17.74	-3.27
Current	11.98	-1.26	23.11	-2.78
New	12.21	-0.12	25.72	0.29
New no primary	11.41	-1.02	-	-

Table 3.1: Standard deviation σ and mean error μ throughout the whole month of simulation. The measurements are the actual production data, current is a simulation of the current controller and new is a simulation of the new controller. For comparison purposes, the new controller is also simulated with the constraint for maintaining the primary reserves are removed.

without this constraint in the noise-free scenario shows a slightly improved result compared to the current controller.

Comparing the economy of the two implementations, the new controller performs slightly better and is estimated to yield a gain of approximately 90,000 euro per year.

One remark to make is that the currently implemented controller has matured over the cause of years. In comparison, the new controller has been implemented and tested through simulation for a very short time. It is therefore likely that the implementation and further development of the newly developed method will yield improved performance compared to the results of this thesis.

4 | Conclusion

When the project started, the scalability of the controller was a feature which might be required at some point in the future, but the main focus was the performance improvement on the current portfolio. During the course of the project the drive towards more sustainable energy production has changed the energy system, so that scalability has become absolutely crucial to the controller design. An example of this change is the rapid development of virtual power plants and thereby a potential increase in number of effectuators in control in the very near future.

The power system is very complex, and with the increasing number of effectuators complexity will increase even more. One of the aims in this thesis has been to develop a design method which is clear and relatively simple to ensure that it can be implemented in the production at some point. This includes a very high level of abstraction.

To test whether it was possible to develop a design method, a hypothesis was formulated stating that it is possible to develop a controller design method which leads to synthesis of a controller which fulfils the criteria of scalability, flexibility and performance.

A design method has been developed which results in synthesis of a controller with an object-oriented structure with clear interfaces, thus ensuring the *flexibility* of the control structure. The modular design ensures that the controller is easily maintainable in case of updates of the controller such as adding and removing effectuators. The use of interfaces in the design offers flexibility in the implementation of the control formulation for the individual effectuator.

By using Dantzig-Wolfe decomposition, a computational efficient solution can be obtained which grows linearly with the number of effectuators in control. Furthermore, decomposition allows distribution of the optimisation among several computers, thus improving the *scalability* significantly compared to a centralised solution. A customised interior point solver can be utilised to efficiently solve the smaller distributed problems in the hierarchical structure.

Using the design method for synthesis of a controller for the current portfolio, the simulated *performance* is comparable with the simulated performance of the currently implemented PI-controller structure.

The developed design method fulfils all three criteria of the hypothesis, therefore the hypothesis is accepted.

The contributions of the work consist of six papers describing different aspects of portfolio control as well as the development of the method. In summary, the contributions can be listed as:

- Analysis of the current controller structure, based on the structure used in literature. The structure was proven unstable. There are, however, methods for stabilising the controller. [Edlund et al., 2009a]
- Formulation of the load balancing problem as a model predictive controller, and showing that the current state of the art within load balancing control can be improved by introducing Model predictive control theory. [Edlund et al., 2008]
- Developing a method for synthesising a load balancing controller with a modular design and clear interfaces using model predictive control. The design is a two-layer hierarchical structure which has a performance equal to a centralised solution. The design method requires that the underlying optimisation problem can be formulated as a linear program. [Edlund et al., nd]
- Customisation of a general interior point method to exploit the structure of the specific problem in order to speed up the optimisation, by exploiting the structure of the problem. [Edlund et al., 2009c]
- Introduction of Dantzig-Wolfe decomposition for the dynamic calculations of MPC for use in portfolio load balancing control. The result is an independent problems corresponding to the physical subsystems. [Edlund and Jørgensen, nd]
- Implementation of the method on the existing portfolio and compared the performance with the current controller through simulations. The comparison showed that the developed method has potential to improve performance in the existing portfolio. [Edlund et al., nd]
- Developing a method for ensuring that primary reserves are maintained when the load balancing controller performs internal balance control. It is a prerequisite that the reserves are available from the beginning. [Edlund et al., nd]

4.1 Future Work

The design method for a load balancing controller presented in this thesis fulfils the criteria to be able to accept the hypothesis. There are two paths two be followed after this project is finished. One path is the industrial implementation maturing and implementation of the controller in a suitable application. The other is further development of the controller design to remove some of the limitations the current design method imply.

- A design method and a prototype of an implementation has been provided in this thesis, but in order to implement and use the controller in operation it must be matured and implemented to fit in the platform of the control system. Furthermore, only the controller core has been developed, in order to have a fully fledged controller suitable for operation it has to be expanded to include elements such as communication and operator interface.
- The design method enables inclusion of new effectuators in the portfolio in a relativity straight forward manner. Identification and construction of new effectuators are not directly related successors of this thesis. But the framework for including them in balancing control is now established and ready for this task.

- The proposed controller design relies on one Kalman filter for state estimation of the whole system. The complexity of the matrix multiplications grows cubically with the problem size. This could prove to be a limiting factor for the scalability in the proposed control design. However, the computation time spent on the Kalman filter in the implementation is insignificant compared to the computation time spent by the controller. A logical extension of the design method would be to include a distributed state estimator as well. Distributed state estimation has already been treated in [Alriksson and Rantzer, 2006; Farina et al., 2009] among others, but needs to be incorporated in the method. Incorporating distributed estimation into the method would ensure that the estimator would fit into the framework, thereby ensuring that each of the modules for the individual effectuators contain all information needed to add the effectuator to the control.

- Currently, the design method requires that the controller can be stated as a linear program to be able to perform the optimisation. Expansion of the controller design is a topic for further research. Inclusion of nonlinearities in both performance function and constraint is the ultimate goal, but the next logical step is to include quadratic terms in the performance function since it would enable the framework to handle the most common MPC formulation.

- Simulations show that the discrete events generated by effectuators being added to or removed from the controller do not induce large deviations or cause instabilities. However, there is not established firm theoretical guarantees that these events will not destabilise the system. A theoretical foundation ensuring these properties would be a clear enhancement of the robustness of controller.

- Currently, the events of adding and removing effectuators are generated externally. Expanding the controller to predict when it is safe and/or optimal to add and remove effectators is another topic of future research closely related to the stability analysis above. This topic can be generalised to the controller being able to handle discrete events, and thus removing the limitation that the effectuators must be continuous to be included in the control. This would significantly expand the types of effectuators that can be handled by the controller.

References

L. Acar. Boundaries of the Receding Horizon Control for Interconnected Systems. *Journal of Optimization Theory and Applications*, 84(2):251–271, 1995.

B. M. Åkesson, J. B. Jørgensen, N. K. Poulsen, and S. B. Jørgensen. A Generalized Autocovariance Least-Squares Method for Covariance Estimation. *Proceedings of the 2007 American Control Conference, New York, USA*, pages 3713–3714, 2007.

C. Albenesi, M. Bossi, L. Magni, J. Paderno, Pretolani, P. Kuehl, and M. Diehl. Optimization of the Start-up Procedure of a Combined Cycle Power Plant. *Proceedings of the 45th IEEE Conference on Decision and Control, San Diego, USA*, pages 1840–1845, 2006.

F. Allgöwer, T. A. Badgwell, J. B. Rawlings, and S. J. Wright. Nonlinear model predictive control. In *Perspectives in Control. Plenary Lectures and Mini-Courses at the 5th European Control Conference ECC'99*, pages 391–449. Springer-Verlag, London, 1999.

J. Allwright and G. Papavasiliou. On Linear Programming and Robust Model-Predictive Control Using Impulse-Responses. *Systems & Control Letters*, 18(2):159–164, 1992.

P. Alriksson and A. Rantzer. Distributed Kalman Filtering Using Weighted Averaging. In *Proceedings of the 17th International Symposium on Mathematical Theory of Networks and Systems, Kyoto, Japan*, Kyoto, Japan, 2006.

K. Aoki, T. Satoh, M. Itoh, T. Ichimori, and K. Masegi. Unit Commitment in a Large-Scale Power System including Fuel Constrained Thermal and Pumped-Storage Hydro. *IEEE Transactions on Power Systems*, 2(4):1077–1084, 1987.

K. J. Åström and T. Hägglund. *Advanced PID Control*. ISA - Instrumention, Systems and Automation Society, 2006. ISBN 1-556-17942-1.

A. K. Ayuob and A. D. Patton. Optimal thermal generating unit commitment. *IEEE Transactions on Power Apparatus and Systems*, 90(4):1752–1756, 1971.

M. Azzam and Y. S. Mohamed. Robust controller design for automatic generation control based on Q-parameterization. *Energy Conversion and Management*, 1(43):1663–1673, 2002.

B. H. Bakken and O. S. Grande. Automatic Generation Control in a Deregulated Power System. *IEEE Transactions on Power Systems*, 13(4):1401–1406, 1998.

H. Banakar, C. Luo, and B. T. Ooi. Impacts of Wind Power Minute-to-Minute Variations on Power System Operation. *IEEE Transactions on Power Systems*, 23(1):150–160, 2008.

J. E. Beasley. *Modern heuristic techniques for combinatorial problems*. John Wiley & Sons, Inc., New York, NY, USA, 1993. ISBN 0-470-22079-1.

A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numer. Math.*, 4(3):238–252, 1962.

J. D. Bendtsen and K. Trangbæk. Stable controller reconfiguration through terminal connections A practical example. *2009 IEEE International Conference on Control and Automation, Christchurch, New Zealand*, pages 2037–2042, 2009.

C. Bjerge and J. R. Kristoffersen. Run an offshore wind farm like a power plant. *VGB Powertech*, 87(4):63–66, 2007.

M. Blanke, M. Kinnaert, J. Lunze, M. Staroswiecki, and J. Schröder. *Diagnosis and Fault-Tolerant Control*. Springer-Verlag New York, Inc., 2006. ISBN 3-540-01056-4.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004. ISBN 0-521-83378-7.

T. S. Chang and D. E. Seborg. A linear programming approach to multivariable feedback control with inequality constraints. *International Journal of Control*, 37(3):583–597, 1983.

L. Chen, J. Zhong, and D. Gan. Optimal Automatic Generation Control (AGC) Dispatching and Its Control Performance Analysis for the Distribution Systems with DGs. *Power Engineering Society General Meeting, 2007, Tampa, USA*, pages 1–6, 2007.

R. Cheng, J. F. Forbes, and W. S. Yip. Price-driven coordination method for solving plant-wide MPC problems. *Journal of Process Control*, 17(5):429–438, 2007.

R. Cheng, J. F. Forbes, and W. S. Yip. Dantzig-Wolfe decomposition and plant-wide MPC coordination. *Computers and Chemical Engineering*, 32(7):1507–1522, 2008.

N. N. Chokshi and D. C. McFarlane. *A Distributed Coordination Approach to Reconfigurable Process Control*. Springer-Verlag, 2008. ISBN 1-848-00059-9.

J. Czyzyk, S. Mehrotra, M. Wagner, and S. J. Wright. PCx: An Interior-Point Code for Linear Programming. *Optimization Methods and Software*, 11(1):397–430, 1999.

M. J. Dahl-Soerensen and B. Solberg. Pulverized Fuel Control using Biased Flow Measurements. *Proceedings of the IFAC Symposium on Power Plants and Power Systems Control, Tampera, Finland*, 2009.

Danish Ministry of Transport and Energy. Energistrategi 2025 - Perspektiver frem mod 2025. Technical report, Transport- og Energiministeriet, 2005.

G. B. Dantzig and M. N. Thapa. *Linear programming 1: Introduction*. Springer Verlag, 1997. ISBN 0-387-94833-3.

G. B. Dantzig and M. N. Thapa. *Linear Programming 2: Theory and Extensions*. Springer-Verlag, 2002. ISBN 0-387-98613-8.

G. B. Dantzig and P. Wolfe. Decomposition Principle for Linear Programs. *Operations research*, 8(1):101–111, 1960.

F. P. de Mello. Boiler models for system dynamical performance studies. *IEEE Transactions on Power Systems*, 6(1):66–74, 1991.

L. Deprugney and J.-B. Liters. New excess air controller at Le Havre Power Plant. *14th Annual Joint ISA Power Industry Division and EPRI Controls and Instrumentation Conference/47th Annual ISA Power Industry Symposium, Proceedings, Colorado Springs, USA*, pages 1–14, 2004.

L. Deprugney, S. Maurin, A. Girard, and H. Jestin. New advanced coordinated controller for once-through boiler and turbine of coal-fired power plant. *16th Annual Joint ISA POWID/EPRI Controls and Instrumentation Conference and 49th Annual ISA Power Industry Division, POWID Symposium 2006, San Jose, USA*, pages 36–59, 2006.

T. S. Dillon, K. W. Edwin, H. D. Kochs, and R. J. Taud. Integer Programming Approach to the Problem of Optimal Unit Commitment with Probabilistic Reserve Determination. *IEEE Transactions on Power Apparatus and Systems*, PAS-97(6):2154–2166, 1978.

DONG Energy. `http://www.dongenergy.dk`, 2010.

J. Doyle and G. Stein. Multivariable feedback Design. *IEEE Transactions on Automatic Control*, 26(1):4–16, 1981.

W. B. Dunbar. Distributed Receding Horizon Control of Dynamically Coupled Nonlinear Systems. *IEEE Transactions on Automatic Control*, 52(7):1249–1263, 2007.

J. W. Eaton and J. B. Rawlings. Model-Predictive Control of Chemical Processes. *Chemical Engineering Science*, 47(4):705–720, 1992.

K. Edlund and J. B. Jørgensen. Dantzig-Wolfe Algorithm for Power Plant Portfolio Control. *Submitted for International Journal of Control*, nd.

K. Edlund, J. D. Bendtsen, S. Børresen, and T. Mølbak. Introducing Model Predictive Control for Improving Power Plant Portfolio Performance. *Proceedings of the 17th IFAC World Congress, Seoul, South Korea*, pages 6986–6991, 2008.

K. Edlund, J. D. Bendtsen, and P. Andersen. Structural Stability Analysis of a Rate Limited Automatic Generation Control System. *Proceedings of European Control Conference, Budapest, Hungary*, pages 4534–4539, 2009a.

K. Edlund, T. Mølbak, and J. D. Bendtsen. Simple Models for Model-based Portfolio Load Balancing Controller Synthesis. *Proceedings of IFAC Symposium on power plants and power systems control 2009, Tampera, Finland*, 2009b.

K. Edlund, L. E. Sokoler, and J. B. Jørgensen. A Primal-Dual Interior-Point Linear Programming Algorithm for MPC. *Proceedings of 48th IEEE Conference on Decision and Control, Shanghai, China*, pages 351–356, 2009c.

K. Edlund, J. D. Bendtsen, and J. B. Jørgensen. Hierarchical model-based predictive control of power plant portfolio. *Submitted for Control Engineering Practice*, nd.

M. S. Elliott and B. P. Rasmussen. Model-based predictive control of a multi-evaporator vapor compression cooling cycle. *Proceedings of the 2008 American Control Conference, Seattle, USA*, pages 1463–1468, 2008.

Energinet.dk. Common market regulations for electricity in Eastern and Western Denmark. `http://www.energinet.dk/en/servicemenu/Library/Rules+and+Regulations/Common+market+regulations+for+electricity+in+Eastern+and+Western+Denmark`, 2010.

ENTSO-E. `http://www.entsoe.eu/`, 2010a.

ENTSO-E. ENTSO-E Operational handbook. `http://www.entsoe.eu/index.php?id=57`, 2010b.

M. Farina, G. Ferrari-Trecate, and R. Scattolini. A moving horizon scheme for distributed state estimation. *Proceedings of the 48h IEEE Conference on Decision and Control, Shanghai, China*, pages 1818–1823, 2009.

R. Franke and L. Vogelbacher. Nonlinear model predictive control for cost optimal startup of steam power plants. *Automatisierungstechnik*, 54(12):630–637, 2006.

G. F. Franklin, J. D. Powell, and A. Emami-Naeini. *Feedback control of Dynamic Systems*. Prentice Hall, 4th edition, 2002. ISBN 0-130-32393-4.

J. B. Froisy. Model predictive control Building a bridge between theory and practice. *Computers and Chemical Engineering*, 30(10–12):1426–1435, 2006.

B. P. Gibbs, D. S. Weber, and D. W. Porter. Application of Nonlinear Model-Based Predictive Control to Fossil Power Plants. *Proceedings of the 30th IEEE Conference on Decision and Control, Brighton, England*, pages 1850–1856, 1991.

M. S. Grewal and A. P. Andrews. *Kalman Filtering Theory and Practice using Matlab*. John Wiley & Sons, inc, 3rd edition, 2008. ISBN 0-471-39254-5.

V. Gunnerud, B. Foss, B. Nygreen, R. Vestb, and N. C. Walberg. Dantzig-Wolfe decomposition for real-time optimization - applied to the Troll west oil rim. *Proceedings of International Symposium on Advanced Control of Chemical Processes, Istanbul, Turkey*, 2009.

M. F. Hassan, A. A. Abouelsoud, and H. M. Soliman. Constrained Load-frequency Control. *Electric Power Components and Systems*, 36(3):266–279, 2008.

R. Iserman. *Fault-Diagnosis Systems: An Introduction from Fault Detection to Fault Tolerance*. Spring-Verlag, 2005. ISBN 3-540-24112-4.

D. Jia and B. Krogh. Distributed model predictive control. *Proceedings of the 2001 American Control Conference, Arlington, USA*, pages 2767–2772, 2001.

R. Johnson, H. Happ, and W. Wright. Large Scale Hydro-Thermal Unit Commitment-Method and Results. *IEEE Transactions on Power Apparatus and Systems*, PAS-90(3): 1373–1384, 1971.

C. Jørgensen, J. H. Mortensen, T. Mølbak, and E. O. Nielsen. Modelbased Fleet Optimization and Master Control of a Power Production System. In *Proceedings of IFAC Symposium on Power Plants and Power Systems Control 2006, Kananaskis, Canada*, 2006.

T. Keviczky, F. Borelli, K. Fregene, D. Godbole, and G. J. Balas. Decentralised Receding Horizon Control and Coordination of Autonomous Vehicle Formations. *IEEE Transactions on Control Systems Technology*, 16(1):19–33, 2008.

V. Klee and G. J. Minty. How good is the simplex algorithm? In O. Shisha, editor, *Inequalities*, volume 3, pages 159–175. Academic Press, New York, 1972.

M. Larson and D. Karlsson. Coordinated System Protection Scheme Against Voltage Collapse Using Heuristic Search and Predictive Control. *IEEE Transactions on Power Systems*, 18(3):1001–1006, 2003.

L. S. Lasdon. *Optimization theory for large systems*. Dover Publications Inc., 2nd edition, 2002. ISBN 0-486-41999-1.

G. K. Lausterer. Improved maneuverability of power plants for better grid stability. *Control Engineering Practice*, 6(12):1549–1557, 1998.

F. Liu, Y. Song, S. M. J. Ma, and Q. Lu. Optimal load-frequency control in restructured power systems. *IEE Proceedings of Generation, Transmission and Distribution*, 150 (1):87–95, 2003.

J. M. Maciejowski. *Predictive Control with Constraints*. Pearson Education Limited, 2002. ISBN 0-201-39823-0.

L. Magni and R. Scattolini. Stabilizing decentralized model predictive control of nonlinear systems. *Automatica*, 42(7):1231–1236, 2006.

Y. Majanne. Model predictive pressure control of steam networks. *Control Engineering Practice*, 13(12):1499–1505, 2005.

N. I. Marcos, J. F. Forbes, and M. Guay. Coordination of Distributed Model Predictive Controllers for Constrained Dynamic Processes. *Proceedings of International Symposium on Advanced Control of Chemical Processes, Istanbul, Turkey*, 2009.

R. K. Martinson and J. Tind. An interior point method in Dantzig-Wolfe decomposition. *Computers & Operation Research*, 26(12):1195–1216, 1998.

Mathworks. `http://www.mathworks.com/`, 2010.

S. Mehrotra. On the Implementation of a Primal-Dual Interior Point Method. *SIAM J. Optimization*, 2(4):575–601, 1992.

M. Mercangöz and F. J. Doyle III. Distributed model predictive control of an experimental four-tank system. *Journal of Process Control*, 17(3):297–308, 2007.

M. D. Mesarovic, D. Macko, and Y. Takahara. *Theory of Hierarchical Multilevel Systems*. Academic Press, New York, 1970.

A. G. Michelsen and K. Trangbæk. Local controller synthesis for plug and play processes by decomposition. *2009 IEEE International Conference on Control and Automation, Christchurch, New Zealand*, pages 2223–2228, 2009.

T. Mølbak. Advanced control of superheater steam temperatures - an evaluation based on practical applications. *Control Engineering Practice*, 7(1):1–10, 1999.

Y.-H. Moon, H.-S. Ryu, and J.-K. Park. A new paradigm of automatic generation control under the deregulated environments. *Proceedings of IEEE Power Engineering Society Winter Meeting, 2000.*, pages 21–25, 2000.

M. Morari and J. Lee. Model Predictive Control: The Good, the Bad, and the ugly. *Proc. 4th Int. Conf. Chemical Process Control, Columbia, USA*, pages 3575–3584, 1991.

J. H. Mortensen, T. Mølbak, P. Andersen, and T. S. Pedersen. Optimization of boiler control to improve the load-following capability of power plant units. *Control Engineering Practice*, 6(12):1531–1539, 1998.

K. R. Muske and T. A. Badgwell. Disturbance modeling for offset-free linear model predictive control. *Journal of Process Control*, 12(5):617–632, 2002.

K. R. Muske and J. B. Rawlings. Model Predictive Control with Linear Models. *AIChE Journal*, 39(2):262–287, 1993.

R. R. Negenborn. *Multi-agent model predictive control with applications to power networks*. PhD thesis, Univeristy of Delft, 2007.

R. R. Negenborn, A. G. Beccuti, T. Demiray, S. Leirens, G. Damm, B. De Schutter, and M. Morari. Supervisory hybrid model predictive control for voltage stability of power networks. *Proceedings of the 2008 American Control Conference, Seattle, USA*, pages 5444–5449, 2008a.

R. R. Negenborn, B. De. Schutter, and J. Hellendoorn. Multi-agent model predictive control for transportation networks: Serial versus parallel schemes. *Engineering Applications of Artificial Intelligence*, 21(3):353–366, 2008b.

R. R. Negenborn, S. Leirens, B. De. Schutter, and J. Hellendoorn. Supervisory nonlinear MPC for emergency voltage control using pattern search. *Control Engineering Practice*, 17(7):841–848, 2009.

P. Niemczyk, P. Andersen, J. D. Bendtsen, T. S. Pedersen, and A. P. Ravn. Derivation and validation of a coal mill model for control. *Proceedings of the IFAC Symposium on Power Plants and Power Systems Control, Tampera, Finland*, 2009.

J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, 2nd edition, 2006. ISBN 0-387-98793-2.

Nord Pool. `http://www.nordpool.com`, 2010.

B. J. Odelson, M. R. Rajamani, and J. B. Rawlings. A new autocovariance least-squares method for estimating noise covariance. *Automatica*, 42(2):303–308, 2006.

N. P. Padhy. Unit Commitment - A Bibliographical Survey. *IEEE Transactions on Power Systems*, 19(2):1196–1205, 2004.

G. Pannocchia and J. B. Rawlings. Disturbance Models for Offset-Free Model-Predictive Control. *AIChE Journal*, 49(2):426–437, 2003.

G. Pannocchia, N. Laachi, and J. B. Rawlings. A Fast, Easily Tuned, SISO, Model Predictive Controller. In *DYCOPS 7, 5-7 July 2004, Boston, USA*, 2004.

G. Pannocchia, N. Laachi, and J. B. Rawlings. A Candidate to Replcace PID Control: SISO-Constrained LQ Control. *AIChE Journal*, 51(4):1178–1189, 2005.

S. J. Qin and T. A. Badgwell. An Overview Of Industrial Model Predictive Control Technology. In *Proceedings of AIChE Symposium Series*, pages 232–256, 1997.

S. J. Qin and T. A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, 11(7):733–764, 2003.

D. M. Raimondo, L. Magni, and R. Scattolini. Decentralized MPC of nonlinear systems: An input-to-state stability approach. *International Journal of Robust and Nonlinear Control*, 17(17):1651–1667, 2007.

H. Raj. A Distributed Control Algorithm for AGC in Hydrothermal Power Plants. *Proceedings of 2006 IFAC Symposium on Power Plants and Power Systems Control, Kananaskis, Canada*, 2006.

M. R. Rajamani and J. B. Rawlings. Estimation of the disturbance structure from data using semidefinite programming and optimal weighting. *Automatica*, 45(1):142–148, 2009.

A. Rantzer. Dynamic Dual Decomposition for Distributed Control. *Proceedings of the 2009 conference on American Control Conference, St. Louis, USA*, pages 884–888, 2009.

C. V. Rao and J. B. Rawlings. Linear Programming and Model Predictive Control. *Journal of Process Control*, 10(2–3):283–289, 2000.

C. V. Rao, S. J. Wright, and J. B. Rawlings. Application of Interior-Point Methods to Model Predictive Control. *Journal of Optimization Theory and Applications*, 99(3): 723–757, 1998.

J. B. Rawlings and R. Amrit. Optimizing Process Economic Performance Using Model Predictive Control. In L. Magni, D. M. Raimondo, and F. Allgöwer, editors, *Nonlinear Model Predictive Control*, pages 119–138. Springer, 2009. ISBN 3-764-36297-9.

J. B. Rawlings and D. Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, 2009. ISBN 0-975-933770-9.

J. A. Rossiter. *Model-based Predictive Control: A Practical Approach*. CRC Press LLC, 2003. ISBN 0-849-31291-4.

J. A. Rossiter, P. W. Neal, and L. Yao. Applying predictive control to a fossil-fired power station. *Transactions of the Institute of Measurement & Control*, 24(3):177–194, 2002.

V. Sakizlis, K. I. Kouramas, and E. N. Pistikopoulos. Linear Model Predictive Control via Multiparametric Programming. In E. Pistikopoulos, M. Georgiadis, and V. Dua, editors, *Multi-Parametric Model-Based Control*, pages 1–21. WILEY-VCH Verlag GmbH & Co, 2007. ISBN 3-527-31692-2.

S. Salam. Unit Commitment Solution Methods. *Proceedings of the World Academy of Science, Engineering and Technology, Barcelona, Spain*, 26:600–605, 2007.

R. Scattolini. Architectures for distributed and hierarchical Model Predictive Control. *Journal of Process Control*, 19(5):723–731, 2009.

P. O. M. Scokaert, D. Q. Mayne, and J. B. Rawlings. Suboptimal Model Predictive control (Feasibility implies Stability). *IEEE Transactions on Automatic Control*, 44(3):648–654, 1999.

T. Senjyu, T. Kaneko, A. Uehara, A. Yona, H. Sekine, and C.-H. Kim. Output power control for large wind power penetration in small power system. *Renewable Energy*, 34(11):2334–2343, 2009.

S. M. Shahidehpour and S. K. Tong. A scheduling model for the daily operation of an electric power system. *Applied Mathematical Modelling*, 16(5):226–244, 1992.

H. Shayeghi, H. A. Shayanfar, and A. Jalili. Load frequency control strategies: A state-of-the-art survey for the researcher. *Energy Conversion and management*, 50(2):344–353, 2009.

S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control, Analysis and Design*. John Wiley & Sons, Ltd, 2nd edition, 2005. ISBN 0-470-01168-8.

J. Stoustrup. Plug & Play Control: Control Technology Towards New Challenges. *European Journal of Control*, 15(3-4):311–330, 2009.

M. J. Tennyu, J. B. Rawlings, and S. J. Wright. Close-Loop Behaviour of Nonlinear Model Predictive Control. *AIChE Journal*, 50(4):2142–2154, 2004.

B. Tyagi and S. C. Srivastava. A decentralized automatic generation control scheme for competitive electricity markets. *IEEE Transactions on Power Systems*, 21(1):312–320, 2006.

A. N. Venkat, I. A. Hiskens, J. B. Rawlings, and S. J. Wright. Distributed Output Feedback MPC for Power System Control. In *Proceedings of the 45th IEEE Conference on Decision & Control, San Diego, USA*, pages 4038–4045, 2006.

A. N. Venkat, I. A. Hiskens, J. B. Rawlings, and S. J. Wright. Distributed MPC strategies with application to power system automatic generation control. *IEEE Transaction on Control Systems Technology*, 16(6):1192–1206, 2008.

A. Viana, J. P. de Sousa, and M. Matos. Simulated annealing for the unit commitment problem. *Proceedings of 2001 IEEE Porto Power Tech Proceedings*, pages 4–8, 2001.

H. Weber and M. Krueger. Dynamic investigation of network restoration by the pumped-storage plant Markersbach in Germany. *Proceedings of the 17th IFAC World Congress, Seoul, Korea*, 2008.

H. Weber, T. Hamacher, and T. Haase. Influence on the power station park and the grid. *Proceedings of 2006 IFAC Symposium on Power Plants and Power Systems Control, Kananaskis, Canada*, 2006.

E. Welfonder. Least-cost dynamic interaction of power plants and power systems. *Control Engineering Practice*, 5(9):1203–1216, 1997.

A. J. Wood and B. F. Wollenberg. *Power Generation, Operation and control*. John Wiley & sons, inc., 2nd edition, 1996. ISBN 0-471-58699-4.

S. J. Wright. *Primal-Dual Interior-Point Methods*. SIAM, Philadelphia, 1997. ISBN 0-898-71382-4.

Y. Zhang. Solving Large-Scale Linear Programs by Interior-Point Methods under the Matlab Environment. *Optimization Methods and Software*, 10(1):1–31, 1998.

K. Zhou, J. C. Doyle, and K. Glover. *Robust and Optimal Control*. Prentice Hall, 1996. ISBN 0-134-56567-3.

Contributions

Paper A: Structural Stability Analysis of a Rate Limited Automatic Generation Control System

Paper B: Introducing Model Predictive Control for Improving Power Plant Portfolio Performance

Paper C: Simple Models for Model-based Portfolio Load Balancing Controller Synthesis

Paper D: A Primal-Dual Interior-Point Linear Programming Algorithm for MPC

Paper E: A Dantzig-Wolfe MPC Algorithm for Power Plant Portfolio Control

Paper F: Hierarchical model-based predictive control of power plant portfolio

Paper A

Structural Stability Analysis of a Rate Limited Automatic Generation Control System

Kristian Edlund, Jan Dimon Bendtsen & Palle Andersen

This paper was published in:
Proceedings of the 2009 European Control Conference.

Abstract

This paper deals with the the stability analysis of a controller structure used for automatic generation control (AGC). The AGC structure is derived from the litterature and augmented to handle generation rate constraints. The derived structure is then compared to the structure of the AGC used within DONG Energy, which removes deviations between reference and production. The analysis is conducted on a linear approximation of the closed loop system, with focus on the structure of the controller. We show that with the chosen structure, the closed loop system will be internally unstable independently of the controller parameters. The analysis shows that the different units in the portfolio might drift in opposite directions, a behavior also observed in the real system. The drifting ruins the economical optimisation. The instability issue can, however, be alleviated by introducing so-called parallel run.

1 Introduction

Controllers in general, and in particular controllers used for automatic generation control (AGC), are often implemented with the primary motivation to achieve economic benefit from good operation of the plants, which can then be translated into a number of performance criteria, of which the most fundamental is stabilisation of the system.

The problem of designing AGCs to cooperate among multiple regions has been the subject of much research lately, both regarding optimality and stability. However, it is most often assumed that the generators within the area function as one generator. For example [Bakken and Grande, 1998] describes how to introduce an AGC in Norway, but the focus is on the main controller rather than the distribution of the error among the participating generators. Centralised AGC design under constraints is treated in [Hassan et al., 2008] both for single-area and multi-area production, but the area is treated as one generator. In [Venkat et al., 2006; Moon et al., 2000; Tyagi and Srivastava, 2006] decentralised model-based methods for multi-area AGC are developed, but without discussing how to distribute the output from the controller known as the area control error (ACE) among the multiple generators in the control area. Focusing on stability, [Azzam and Mohamed, 2002] developed a design method for generating a stabilising controller.

[Liu et al., 2003; Chen et al., 2007; Wood and Wollenberg, 1996] describe how to distribute the ACE among the participating generators in the area. [Liu et al., 2003; Chen et al., 2007] deal with control of multiple generators within an area using optimisation-based schemes. However, both treat the problem as a static rather than a dynamic problem. [Wood and Wollenberg, 1996] presents an AGC for distributing the ACE to multiple generators based on a PI-controller structure. This structure for distributing the ACE is very common, and the principle is also used in the internal AGC within DONG Energy, who operates in Denmark. The work in [Wood and Wollenberg, 1996] will be used as a base for the structure in this paper and will be further discussed in section 3. However, as will be shown, in case of load disturbances, it is required to modify the scheme in order to maintain stability.

The outline of the rest of the paper is as follows: Section 2 describes the set of assumptions and requirements for the controller and the system. This is followed by a discussion of the augmentation of the structure needed to fulfill the requirements in section 3. Section 4 presents an analysis of the stability of the augmented structure. A numerical

example of the problem is given in section 5, and a discussion of the possible remedies for the problem in Section 6. At last in Section 7 data is shown from the real system, illustrating that the problem highlighted in section 4 exists in reality.

1.1 The Danish power system

The Danish power system is split into two areas, Western Denmark and Eastern Denmark, which are not synchronously connected. The western part of Denmark is,however, synchronous with the UCTE grid. This paper will focus only on the Western Danish Area.

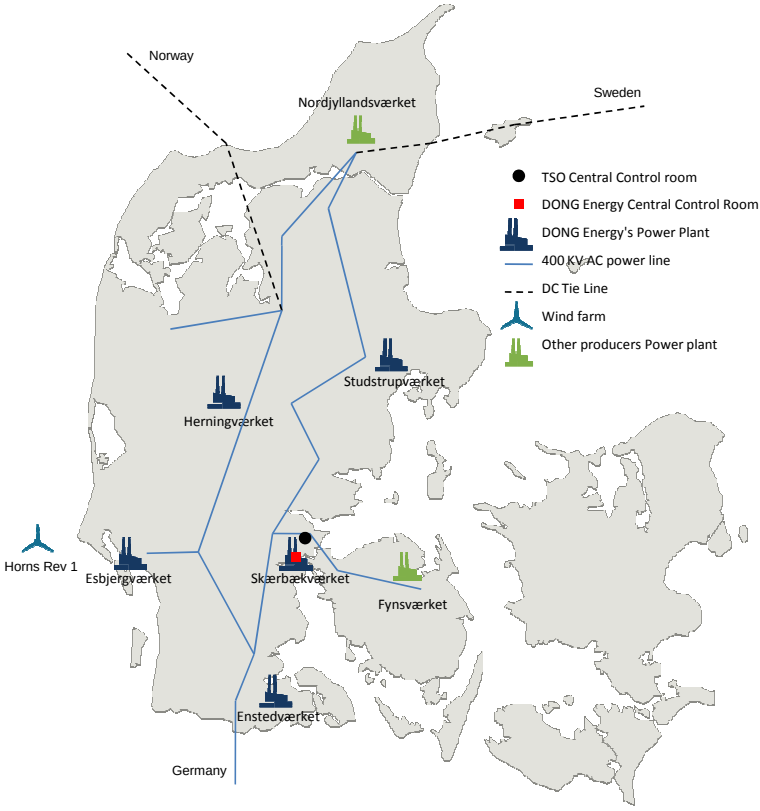


Figure 5.1: Generators participating in balance control in Western Denmark

The Danish power market has been decentralised and liberalised in the recent years. This includes the market of reserves and balance control. The transmission system operator (TSO) in the area is separated from the power producers in order to avoid conflicts of interest. Therefore the Danish TSO has to buy production, reserve and load balancing services from independent power producers. Figure 5.1 shows the generators in Western Denmark that typically participate in the power balancing in the region.

In Denmark, much of the electricity production is based on wind power. In November 2007, the installed wind capacity amounted to more than 25% of the total installed capac-

ity. The Western Danish area functions as one area with multiple generators, which means that a primary concern when designing an AGC is how to distribute the ACE between the generators. Therefore an AGC is applied to regulate the fast fluctuations caused by the wind power [UCTE, 2007].

The commissioned AGC works in a deregulated market and is split up between the TSO and multiple independent power generating companies. The Danish TSO, generates an area control error (ACE) and distributes it among the companies participating in the balance control of the Danish power system. The individual producer is then responsible for distributing its share of the ACE among the generators operated by the company. Each producer also has the responsibility for maintaining the balance between the actual and ordered production within the portfolio of generators.

The fast fluctuations in the power system must be suppressed by thermal generators, which are dominant in the power system. Therefore the thermal generators will often reach their rate limit at typically 4% load/minute, and therefore the AGC must be able to handle the fact that the generators reach the generation rate constraints a significant percentage of the time. It is noted in [Wood and Wollenberg, 1996] that an AGC should be able to handle this, but no specific method is given. In practice, it is often done with a PI-controller structure augmented with a set of so-called *participation factors* to split the load among the participating plants.

Due to the fast fluctuations in the Danish power system, the set of participation factors is changed very often. Existing literature does not seem to have investigated what influence a rate limit constraint has on the stability of the distribution of the ACE within the power system.

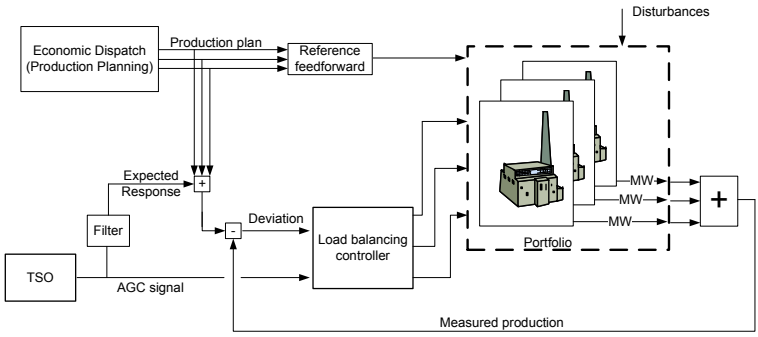


Figure 5.2: System level control within DONG Energy.

The system level control within DONG Energy is depicted in Figure 5.2 and further described in [Jørgensen et al., 2006]. This paper concentrates on the load balancing controller. It gets an input from the production sold on the Power Exchange and the share of the ACE coming from the TSO. Adding these gives the total portfolio reference which has to be followed.

This paper deals with analysis of the stability of an AGC when compensation for generation rate constraints are added to the controller. The analysis is conducted based on a linearised model of the power system.

2 Preliminaries

This section will define the stability condition as well as list the assumptions used when analysing the structure and stability of the AGC.

During the analysis the definition of internal stability is used as defined in [Zhou et al., 1996] and [Skogestad and Postlethwaite, 2005].

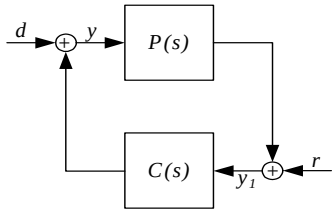


Figure 5.3: Block diagram used to check internal stability

Definition 2. The interconnection of LTI systems $P(s)$ and $C(s)$ depicted in Figure 5.3 is internally stable if and only if all four transfer functions comprising $H(s)$ in (5.1) are stable

$$\begin{bmatrix} y \\ y_1 \end{bmatrix} = H(s) \begin{bmatrix} d \\ r \end{bmatrix} \tag{5.1}$$

$$H(s) = \begin{bmatrix} (I - C(s)P(s))^{-1} & C(s)(I - P(s)C(s))^{-1} \\ P(s)(I - C(s)P(s))^{-1} & (I - P(s)C(s))^{-1} \end{bmatrix}$$

The following assumptions are made:

- The AGC is based on PI-controllers.
- The AGC should be able to compensate for generation rate constraints and generation limits of the generators.
- Each generator has it own local loop controller to ensure reference tracking. With the local loop controller, the generator response can be approximated by a third order model

  $$G_i(s) = \frac{1}{(T_i s + 1)^3} \tag{5.2}$$

  where T_i is the time constant for generator i. The time constants of the generators used in DONG Energy's AGC ranges from 15 to 90s. In this work, each of the generators will be modelled by a stable transfer function.

  $$G_i(s) = \frac{\alpha_{Gi}(s)}{\beta_{Gi}(s)}. \tag{5.3}$$

  This means that the polynomial $\beta_{Gi}(s)$ has all roots in the open left half plane, and that $s = 0$ is not a root in $\alpha_{Gi}(s)$.
- The controller in the system is tuned in such a way that the system is stable from reference to output.

3 Structural Considerations

This section first discusses the structure of the AGC proposed in [Wood and Wollenberg, 1996] and what augmentations must be made in order to fullfill the requirements given in the previous section. Secondly it discusses the structure of the controller used within DONG Energy.

3.1 AGC proposed in the literature

In essence, the structure of the AGC described in [Wood and Wollenberg, 1996] is shown in Figure 5.4. It consists of one PI-controller and a set of participation factors ($p_{fi}, i = 1, ..., N$) where N is the number of generators. The participation factors can then be found in a number of different ways; [Wood and Wollenberg, 1996] suggests to use the economic dispatch for finding the set of participation factors.

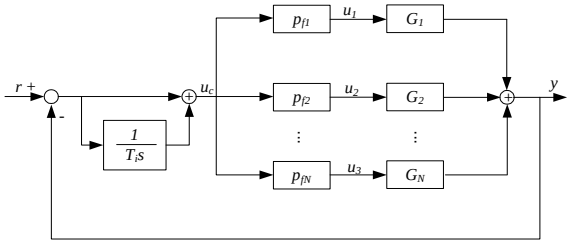


Figure 5.4: Proposed AGC structure[Wood and Wollenberg, 1996]

When phenomena such as reaching a generation rate constraint occurs, it is necessary to change the set of participation factors so that additional disturbances are not introduced in the system due to the rate limit.

Changing the participation factors arbitrarily in the structure shown in Figure 5.4 might cause abrupt changes in the control signals ($u_1, ..., u_N$), unless $u_c = 0$. Changing the control inputs to the generators abruptly will also introduce additional disturbances due to the dynamic behavior of the generators themselves.

To ensure that the generation rate constraints are not violated, the control signals should not change more rapidly than the generation rate constraints allow. This means that p_{fi} should be changed to accommodate bumpless transfer. This, in turn, means that memory of the value at the previous sample instant is needed in order to track the change in u_i. One easy way to achieve this is using an incremental formulation (described in [Åström and Wittenmark, 1990]), as shown in Figure 5.5.

Furthermore memory is needed when bumpless transfer from manual to automatic is wanted in the switch as described in [Åström and Wittenmark, 1990]. This is an additional argument for having several parallel integrators.

3.2 AGC used in DONG Energy

The controller for distributing the ACE within DONG Energy has a slightly different structure from the one presented in above. It is shown in Figure 5.6.

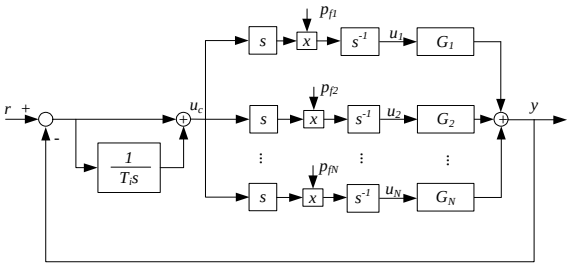


Figure 5.5: Alternative structure of the AGC with incremental implementation.

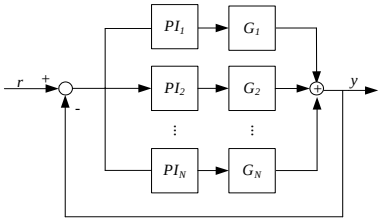


Figure 5.6: Structure of the AGC within DONG Energy

The reference is distributed to a set of PI-controllers with gain scheduling. The gains are recomputed every 0.5 seconds according to factors such as generation rate constraints.

In order to compensate for generation rate constraints, the AGC structure used within DONG Energy has to be augmented in a similar fashion. The result is shown in Figure 5.7. The structure has a behaviour similar to the structure shown previously.

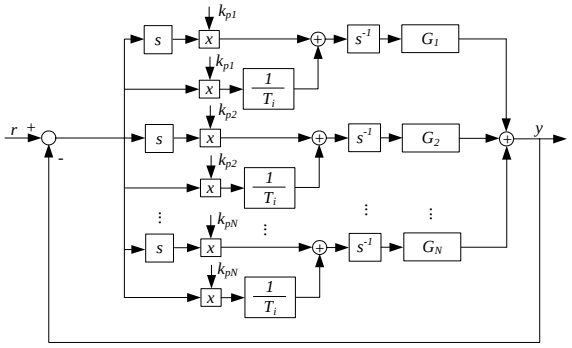


Figure 5.7: Current structure of the AGC

4 Stability of an AGC

In the previous section it was argued that, in order to compensate for the generation rate constraints, it is necessary to have memory about each generator, which can for instance be added in terms of integrators. In this section, the stability of a structure with parallel integrators will be analysed. The participation factors are calculated at each sample, and is based on the constraints and limits. Therefore the participation factors often change rapidly, in particular whenever Δu_c changes sign. This behaviour will be treated as a disturbance entering the system close to p_{fi}.

4.1 AGC with compensation for changes in p_{fi}

For the analysis the system is grouped into subsystems as shown in Figure 5.8.

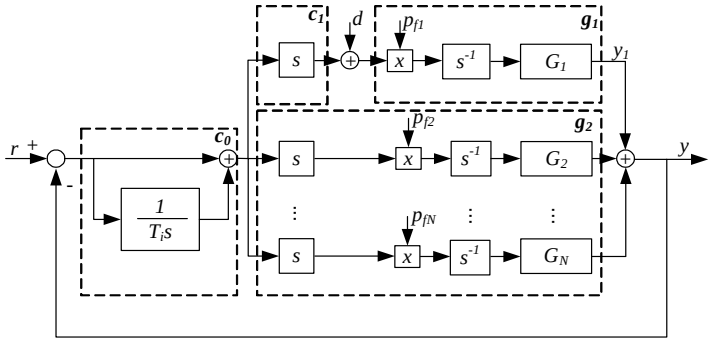


Figure 5.8: Subsystems used for analysis of AGC with compensation for changes in p_{fi}

where the transfer function of each subsystem can be described as

$$g_1(s) = \frac{\alpha_{G1}}{s\beta_{G1}}, g_2(s) = \frac{\alpha_{G2}}{\beta_{G2}}, c_1(s) = s, c_0(s) = 1 + \frac{1}{T_i s} \tag{5.4}$$

$g_2(s)$ contains the generators G_i where $i = 2, ..., N$. α_{G1} is assumed not to have a root in $s = 0$. We now have the following result.

Theorem 2. *The structure shown in Figure 5.8 is internally unstable for any choice of controller parameters under the assumptions listed in section 2.*

Proof. By showing that at least one of the transfer functions in (5.1) cannot be stabilised for any choice of controller parameter, it is proven that the system is internally unstable.

The transfer function matrix is shown in (5.5) on the next page. □

All transfer functions except the bottom left has the same denominator. Since $\frac{y(s)}{r(s)}$ is stable by assumption, all three transfer functions are stable. However, $\frac{y_1(s)}{d(s)}$ has a zero in $s = 0$, since $s = 0$ is not a root in the nominator polynomial it is not cancelled. This closed right half plane pole in the transfer function cannot be eliminated by any choice of controller parameters, and the structure is therefore internally unstable.

$$H(s) = \begin{bmatrix} \frac{g_1}{1+(g_1c_1+g_2)c_0} & \frac{(g_1c_1+g_2)c_0}{1+(g_1c_1g_2)c_0} \\ \frac{g_1+g_1c_0g_2}{1+(g_1c_1+g_2)c_0} & \frac{g_1c_1c_0}{1+(g_1c_1+g_2)c_0} \end{bmatrix} =$$

$$\begin{bmatrix} \frac{\alpha_{G1}\beta_{G2}T_i}{\beta_{G1}\beta_{G2}T_is+(\alpha_{G1}\beta_{G2}+\alpha_{G2}\beta_{G1})(1+T_is)} & \frac{(\alpha_{G1}\beta_{G2}+\alpha_{G2}\beta_{G1})(1+T_is)}{\beta_{G1}\beta_{G2}T_is+(\alpha_{G1}\beta_{G2}+\alpha_{G2}\beta_{G1})(1+T_is)} \\ \frac{\alpha_{G1}\beta_{G2}T_is+\alpha_{G1}T_is\alpha_{G2}+\alpha_{G1}\alpha_{G2}}{s(\beta_{G1}\beta_{G2}T_is+(\alpha_{G1}\beta_{G2}+\alpha_{G2}\beta_{G1})(1+T_is))} & \frac{\alpha_{G1}\beta_{G2}(1+T_is)}{\beta_{G1}\beta_{G2}T_is+(\alpha_{G1}\beta_{G2}+\alpha_{G2}\beta_{G1})(1+T_is)} \end{bmatrix} \tag{5.5}$$

4.2 Current Structure

For the analysis the system is grouped into subsystems as shown in Figure 5.9.

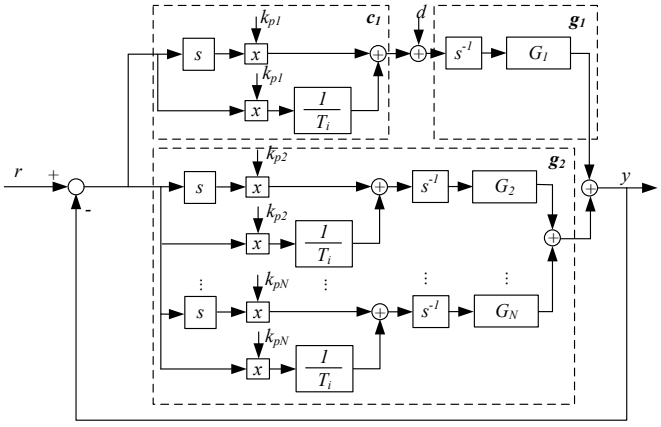


Figure 5.9: Subsystems used for analysis of the currently used structure

The transfer function for each subsystem can be described as

$$g_1(s) = \frac{\alpha_{G1}}{s\beta_{G1}}, g_2(s) = (\alpha_{c2} + \frac{1}{T_is})\frac{\alpha_{G2}}{\beta_{G2}}, c_1(s) = (s + \alpha_{c1}) \tag{5.6}$$

Theorem 3. *The structure shown in Figure 5.9 is internally unstable for any choice of controller parameters under the assumptions listed in section 2.*

Proof. For the sake of brevity, we only show the unstable transfer function from disturbance (d) to output of generator 1 (y_1) which is

$$\frac{y_1(s)}{d(s)} = \frac{g_1 + g_1g_2}{1 + g_1c_1 + g_2}. \tag{5.7}$$

By inserting the expressions from (5.6), it can be written as

$$\frac{y_1(s)}{d(s)} = \frac{\alpha_{G1}\beta_{G2}+T_is\alpha_{G1}\beta_{G2}+T_is\alpha_{c2}\alpha_{G1}\alpha_{G2}}{s(\beta_{G1}\beta_{G2}T_is+\alpha_{G2}s\beta_{G1}s(\alpha_{c2}T_is+1)+\alpha_{G1}\alpha_{c1}\beta_{G2}s^2(1+T_is))}. \tag{5.8}$$

This transfer function has a root in $s = 0$ in the denominator, and since $s = 0$ is not a root in the nominator polynomial it is not cancelled. The root in $s = 0$ cannot be cancelled by any choice of controller parameters, and the structure is therefore internally unstable. □

5 Numerical Example

In this section, a numerical example is shown for the structure shown in Theorem 2. It corresponds to the structure proposed in [Wood and Wollenberg, 1996], except that it is modified to accommodate generation rate constraints. The system consists of two generators, where the dynamics can be modelled as two third order systems as shown in Figure 5.10.

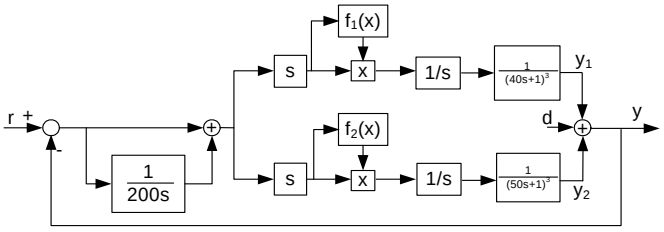


Figure 5.10: Example AGC with compensation for changes in p_{fi}

The participation factors are calculated using the simple functions $f_1(x) = \begin{cases} 0.4 & x \leq 0 \\ 0.6 & x > 0 \end{cases}$ and $f_2(x) = \begin{cases} 0.6 & x \leq 0 \\ 0.4 & x > 0 \end{cases}$ The changing participation factors are used in order to exploit the full rate of change possible when there are asymmetric rate limits. d is zero-mean white noise added to the output with variance $\sigma^2 = 0.001$.

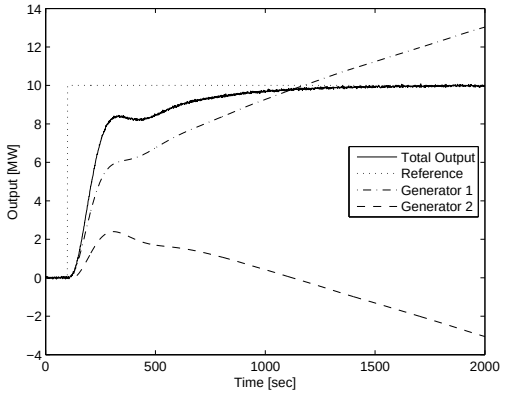


Figure 5.11: Simulation of AGC with compensation for changes in p_{fi}

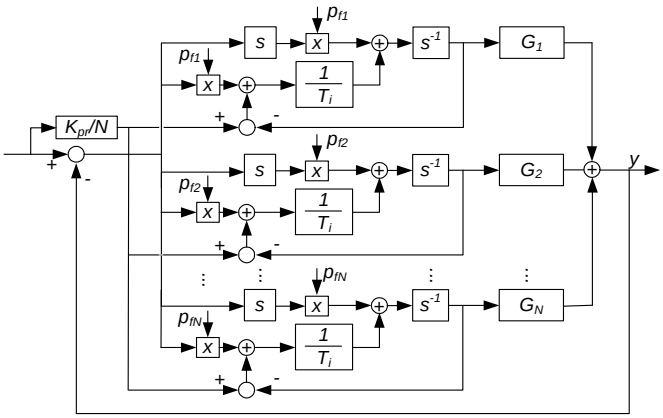


Figure 5.12: Controller with parallel run

Figure 5.11 show the result of a simulation of the system shown in Figure 5.10 using a sample time of 1 second. The results shows that the controller is capable of following the reference, but the generators drift in opposite directions, as predicted by the internal instability results shown in the previous section.

6 Stabilisation of the structure

In this section, we present two different ways to remedy the internal instability of the system as well as a discussion of pros and cons of the two solutions. Both solutions have been applied to the controller within DONG Energy.

6.1 Parallel run

Parallel run is commonly used within the power plant industry when two similar subsystems have to work in parallel to complete a task. When using PI-controllers in parallel, the subsystems will in general not contribute equally when the equilibrium is reached. In order to obtain this behaviour, parallel run is introduced. In the AGC system, the generators are treated as parallel subsystems, which contribute to achieve an overall goal. Figure 5.12 shows how the parallel run is introduced.

The parallel run drives the integrators towards the same value, and thereby ensures that the behaviour seen in the example above where to generators drift in opposite directions is avoided. Choosing the parallel run, parameter K_{pr} is a trade off between forcing the integrators towards each other, and obtaining good reference tracking. If the gain K_{pr} is too large, the parallel run may in some situations dominate the controller behaviour, ruining the reference tracking capability.

Gain scheduling of the parallel run as a function of the change in ACE might be a solution to overcome balancing between not interfering on the control power and no effect from the parallel run.

6.2 Participation factors

Within DONG Energy, the calculation of participation factors has been changed to a rule based scheme, which takes the current state of controller and the change of the ACE into account. The result of the algorithm is a prioritised list. The generator with the highest priority gets as much of the ACE as possible within the generation rate constraints.

One of the rules in the algorithm is to force generators back towards zero when possible, so that if a generator gives a positive contribution and the ACE value decreases, the generator gets higher priority. This has resulted in a significant improvement to the AGC in DONG Energy and the phenomona of internal instability of the AGC has been removed. However, the complexity of the system has been significantly increased.

7 Actual System

The linear analysis of the controller showed that the units are likely to drift even though the total portfolio output remains stable. The analysis was conducted on a simplified linear model of the controller, since the real system cannot be analysed in detail through any of the methods described in the introduction. However, general trend data shows good agreement with the analysis, as will be discussed in the following.

Figure 5.13 shows the correction signals sent to the units in a period of 24 hours.

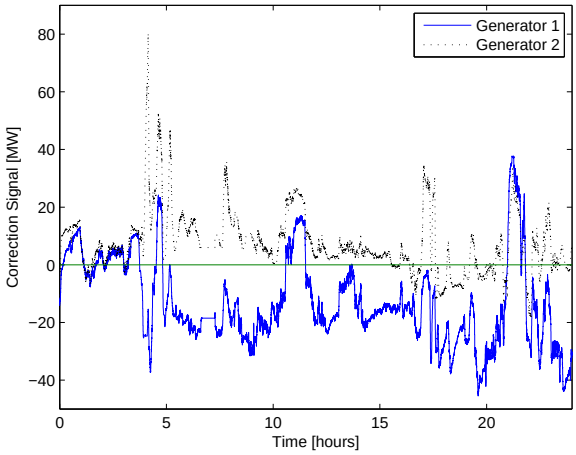


Figure 5.13: Correction signals from the controller

After approximately three hours, the correction signal of generator 1 rapidly drops to a negative value while generator 2 stays positive. The correction signals keep their opposite signs as for the most of the time within the next seven hours. Using standard PI-controllers, the output is kept steady once the error is zero, meaning that the PI-controllers output might settle with opposite signs as in Figure 5.13. The parallel run in the PI-controllers is the only mechanism trying to pull the two units towards a com-

mon correction, so that they contribute equally. However, the parallel run is too weak compared to the noise and the change in reference to succeed in this case.

The explanation for the units drifting from the production plan is found in the distribution of the controller gain and the rate limiter. The controller has two different gains, one for positive error, and one for negative error. The result is that one unit ends up with a large part of the contribution in one direction and the other unit ends up with a large part of the contribution in the other direction. When the error is positive, both units receive commands to follow a step up, but one unit gets a larger step than the other. When the error changes sign, both units correspondingly receive orders to follow a step down, but the other unit then gets a much larger step than the first. When the error is close to zero, it often changes sign, which over time results in one unit accumulating a large positive correction and one unit accumulating a large negative correction. Overall, this behaviour is quite similar to the behaviour observed by introducing disturbances in the linaer approximation treated in section 4.

The units are saved from drifting too far apart for three main reasons. The systems are limited, so that one unit cannot deviate more than a certain amount from the production plan without manual intervention from the operator. These limits ensure that the units are not drifting too much. The parallel run is a gain multiplied with an error, so the larger the span between two units, the stronger the parallel run is. And last but not least, there are steady periods where the reference from the TSO is close to steady state, giving the parallel run a chance of pulling the units together again.

8 Discussion

A set of assumptions and requirements commonly found in literature for an AGC has been summarised and analysed. In order to fulfill this set of requirements, it is argued that the system must be structured in a certain way. However, it was also found that such a system is internally unstable. The instability shows in fact that generators participating the balance control has a tendency to drift in opposite directions. Data from a real production environment supports this conclusion. The drifting of the generators ruin the economical optimisation of the system.

Two different ways to remedy the problem have been proposed - parallel run and changes in the calculation of participation factors. It is found that in practice, the parallel run is insufficient to force the drifting generators towards the same value.

Another solution is to change the calculation of the participation factors. A rule-based system has been set up within DONG Energy, and rules have been added to ensure that the generators are forced back to the production plan when possible. The system works well, and the drifting is no longer seen. However, the system behavior is now far more difficult to analyse in case of problems.

The favoured remedy is the last proposed, but due to the complexity of this solution, a better approach might be to reduce the requirement that the AGC should be based on PI-controllers. This is an indication that it may prove advantageous to introduce model-based MIMO control in the future.

References

K. J. Åström and B. Wittenmark. *Computer-controlled Systems*. Prentice Hall, 3rd edition, 1990. ISBN 0-133-14899-8.

M. Azzam and Y. S. Mohamed. Robust controller design for automatic generation control based on Q-parameterization. *Energy Conversion and Management*, 1(43):1663–1673, 2002.

B. H. Bakken and O. S. Grande. Automatic Generation Control in a Deregulated Power System. *IEEE Transactions on Power Systems*, 13(4):1401–1406, 1998.

L. Chen, J. Zhong, and D. Gan. Optimal Automatic Generation Control (AGC) Dispatching and Its Control Performance Analysis for the Distribution Systems with DGs. *Power Engineering Society General Meeting, 2007, Tampa, USA*, pages 1–6, 2007.

M. F. Hassan, A. A. Abouelsoud, and H. M. Soliman. Constrained Load-frequency Control. *Electric Power Components and Systems*, 36(3):266–279, 2008.

C. Jørgensen, J. H. Mortensen, T. Mølbak, and E. O. Nielsen. Modelbased Fleet Optimization and Master Control of a Power Production System. In *Proceedings of IFAC Symposium on Power Plants and Power Systems Control 2006, Kananaskis, Canada*, 2006.

F. Liu, Y. Song, S. M. J. Ma, and Q. Lu. Optimal load-frequency control in restructured power systems. *IEE Proceedings of Generation, Transmission and Distribution*, 150 (1):87–95, 2003.

Y.-H. Moon, H.-S. Ryu, and J.-K. Park. A new paradigm of automatic generation control under the deregulated environments. *Proceedings of IEEE Power Engineering Society Winter Meeting, 2000.*, pages 21–25, 2000.

S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control, Analysis and Design*. John Wiley & Sons, Ltd, 2nd edition, 2005. ISBN 0-470-01168-8.

B. Tyagi and S. C. Srivastava. A decentralized automatic generation control scheme for competitive electricity markets. *IEEE Transactions on Power Systems*, 21(1):312–320, 2006.

UCTE. European Wind Integration Study (EWIS) Towards a successful integration of wind power into european electricity grids. Technical report, UCTE, 2007.

A. N. Venkat, I. A. Hiskens, J. B. Rawlings, and S. J. Wright. Distributed Output Feedback MPC for Power System Control. In *Proceedings of the 45th IEEE Conference on Decision & Control, San Diego, USA*, pages 4038–4045, 2006.

A. J. Wood and B. F. Wollenberg. *Power Generation, Operation and control*. John Wiley & sons, inc., 2nd edition, 1996. ISBN 0-471-58699-4.

K. Zhou, J. C. Doyle, and K. Glover. *Robust and Optimal Control*. Prentice Hall, 1996. ISBN 0-134-56567-3.

Paper B

Introducing Model Predictive Control for Improving Power Plant Portfolio Performance

Kristian Edlund, Jan Dimon Bendtsen, Simon Børresen & Tommy Mølbak

This paper was published in:
Proceedings of the 17th IFAC World Congress

The layout has been revised

Abstract

This paper introduces a model predictive control (MPC) approach to construction of a controller for balancing power generation against consumption in a power system. The objective of the controller is to coordinate a portfolio consisting of multiple power plant units in an effort to perform reference tracking and disturbance rejection in an economically optimal way. The performance function is chosen as a mixture of the ℓ_1-norm and a linear weighting to model the economics of the system. Simulations show a significant improvement of the performance of the MPC compared to the current implementation consisting of a distributed PI controller structure, both in terms of minimising the overall cost but also in terms of the ability to minimise deviation, which is the classical objective.

1 Introduction

This paper focuses on the power system in the western part of Denmark, where DONG Energy [DONG Energy, 2007] is the largest power producer. Being a major power producer also means that DONG Energy provides a major part of the balancing reserves for the transmission system operator (TSO), who has the overall load balancing responsibility. Load balancing means making the production equal to the consumption. In 2007 approximately 30% of the installed capacity in West Denmark was wind turbines - a large share compared to other areas of Europe. This makes balance control a difficult issue due to the stochastic behaviour of the wind-based production. Today, balancing can be done partly by exporting the electricity to other parts of Europe and partly by adjusting the load of the other production units. However, with the increasing integration of wind power the current balance control system must be improved in order to be able to handle the changing conditions in the future.

The deregulation and decentralisation of the European power system complicate coordination of control actions. In [UCTE, 2007] it is predicted that a significant number of wind turbines will be installed in Europe. This will introduce more stochastic production, which calls for improved control of the power system to balance production and consumption in order to avoid large blackouts in the future.

The fluctuation in Denmark introduced by the wind turbines has made it necessary to commission an AGC (Automatic Generation Control) which is able to activate a reserve of ± 140MW to take care of small and quick deviations [UCTE, 2007]. This AGC is controlled by the TSO, which sends an activation signal to the balancing participants, who are then responsible for activating the required reserves. The reserves are activated by changing the load distribution among a portfolio of power plants.

The controller which distributes the reserve activation within the DONG Energy portfolio is responsible for maintaining the load balance within the portfolio on a seconds-to-minutes horizon, until the economic dispatch can take over and handle the short term (minutes-to-hours) load balancing. The controller is therefore referred to as the load balancing controller, not to be mistaken for the AGC at the TSO.

The portfolio is a very complex system and the optimisation and control are therefore ordered in a hierarchical fashion in order to break down the complexity as described in [Jørgensen et al., 2006]. The economic dispatch as well as the load balancing are handled mainly on the system level as described in [Jørgensen et al., 2006].

The load balancing controller is currently based on a distributed PI controller structure and ad hoc methods to obtain the desired behaviour of the system. The requirements and wishes for the balance controller keep increasing, rapidly pushing the method to the limit of what is possible. In particular, requests for optimality according to a performance function have arisen, which cannot be guaranteed with the current system setup. Also, operation closer to the limits is required. The first versions of the load balancing controller were focused only on minimising the deviation from the reference production, but recently focus has shifted towards minimising the deviation as economically as possible. This calls for methods which are better suited for handling large multiple-input-multple-outpout (MIMO) systems with multiple performance measures.

Much work has been done to enhance the disturbance rejection capabilities of single power plants, see [Welfonder, 1997] and [Lausterer, 1998]. However, real-time coordination of several units in an effort to perform disturbance rejection has not been reported in the literature before.

This paper presents the first step towards establishing a more stringent method for handling balance control of a power system portfolio. A model-based MIMO control solution based on Model Predictive Control (MPC) offering inherent constraint handling and systematic utilisation of feed forward is presented. A short introduction to the system is given in section 2 followed by the derivation of a simple state space model and constraints, which are used in the construction of the controller (section 3). Based on the stated optimisation goals, a performance function consisting of linear weighting and ℓ_1-norms is derived in section 4 and used to construct a controller for the system. In section 5, the controller is tested in two different scenarios illustrating the reference tracking ability and disturbance rejection capabilities of the solution. The results are compared to the current implementation of the balance controller.

2 System Description

A quick introduction to the Danish power production system and the highest level of the hierarchy of DONG Energy is given here. For more details, see [Jørgensen et al., 2006].

The Danish power production can be split into two categories; *planned production* and *reserves*. The planned production is the production known ahead of time, which means long-term contracts and power sold on the power exchange 8-36 hours ahead of production time. Reserves are power production which can be activated quickly and which is used to compensate for imbalances. The reserves are a service requested by the Danish Transmission System Operator (TSO) who has the responsibility for maintaining the balance within the Danish region. There are different kinds of reserves, see [Jørgensen et al., 2006] for details. For the sake of simplicity, this paper only considers the automatic reserves.

An overview of the interaction of the different subsystems is presented in Fig. 6.1. A short-term load scheduler (STLS) performs optimisation for the distribution of power generation resulting in a 5-minute based 24-hour production plan, which is sent to the production units. To compensate for the dead time of the production units, the production plan is issued as a reference feed-forward.

Based on the imbalances within the Danish region, the TSO generates a reference signal, and the portfolio is then expected to respond to the reference with a given dynamic

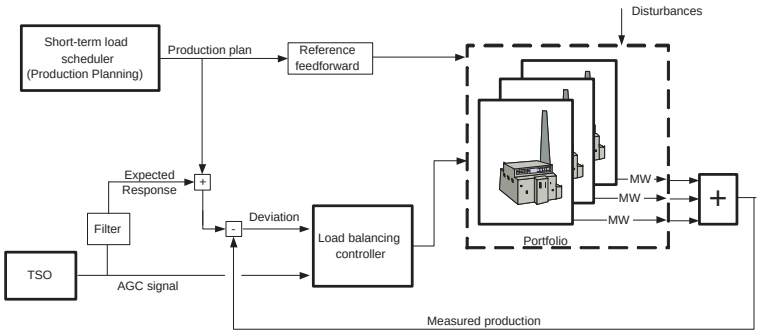


Figure 6.1: Overview of the portfolio at system level

response.

The load balancing controller shown in Fig. 6.1 serves two purposes. It manages the coordination of the production units to obtain the expected response to the TSO reference signal. The second purpose is to minimise the deviation between the actual total production and the reference production. This way, the power controller compensates for the unavoidable discrepancies that will occur due to the feed-forward nature of the STLS.

The current active portfolio consists of six units placed at five different power plants. The maximum power output of the units in the portfolio ranges from 80MW to 650MW, and the units also vary in terms of dynamic behaviour.

Note that the term 'unit' covers the physical process from fuel input to generator as well as the control system controlling the process.

3 Modelling

As the focus of this paper is to establish a model-based method for constructing a load balancing controller, this section describes the model of the portfolio used by the controller. Fig. 6.2 is a schematic view of the model of a single unit. The input to the model is a reference given to the control system, and the output is the measured power output from the unit.

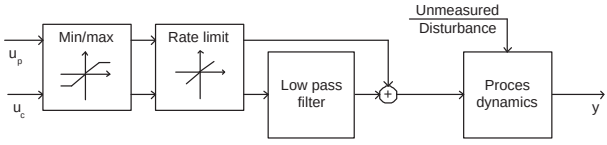


Figure 6.2: Schematic of the unit model used in the controller

The input to the model can be divided into two; the production plan (u_p), which is uncontrollable by the controller and therefore regarded as a disturbance, and the balance controller input (u_c) to the system.

The unit model consists of two parts; the control system and the system process. The part of the control system influencing the model the most is the limits on the reference signals, both in form of a max/min bound and a limit on the rate of change. The dynamics of the process are approximated by a third order model with real stable poles, which in this context describes the process dynamics reasonably well.

The max/min bound in Fig. 6.2 is to be interpreted as

$$u_{min,k} \leq u_{c,k} + u_{p,k} \leq u_{max,k} \tag{6.1}$$

meaning that the sum of the inputs are bounded at each sample k.

The same applies for the rate limiter namely

$$\Delta u_{min,k} \leq \Delta u_{c,k} + \Delta u_{p,k} \leq \Delta u_{max,k} \tag{6.2}$$

where

$$\begin{aligned} \Delta u_{c,k} &= u_{c,k} - u_{c,k-1} \\ \Delta u_{p,k} &= u_{p,k} - u_{p,k-1} \end{aligned}$$

The low pass filter is implemented in the control system in order to avoid abrupt changes from the currently implemented PI controllers. The actual filter implemented in the control systems varies slightly between the units, but it is typically a third order filter with three time constants of 10s.

The dynamic parts of the model are formulated as a state space model as shown in (6.3). The rate limitation and the max/min bound are formulated as input constraints. That is,

$$\begin{aligned} & x_{j,k+1} = A_j x_k + B_j u_k + E_j d_k \\ & y_k = C_j x_k \\ s.t. \quad & \\ & u_{min,k} \leq u_{c,k} + u_{p,k} \leq u_{max,k} \\ & \Delta u_{min,k} \leq \Delta u_{c,k} + \Delta u_{p,k} \leq \Delta u_{max,k} \end{aligned} \tag{6.3}$$

for each unit $j = 1, \ldots, 6$ and $x_j \in \mathbb{R}^{N_j}$, etc.

The production plan is treated as a disturbance since it cannot be controlled by the load balancing controller. Thus each unit model has one input, one disturbance and one output.

The individual unit models can be compiled into one portfolio model by constructing large block diagonal matrices containing the individual unit models. The output matrix is also constructed as a block diagonal matrix, but has to be expanded to contain an output describing the total portfolio output, ie,

$$A = \begin{bmatrix} A_1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & A_6 \end{bmatrix}, B = \begin{bmatrix} B_1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & B_6 \end{bmatrix}, \tag{6.4}$$

$$E = \begin{bmatrix} E_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & E_6 \end{bmatrix}, C = \begin{bmatrix} C_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & C_6 \\ C_1 & \dots & C_6 \end{bmatrix} \tag{6.5}$$

This means that the input, state and output vectors of the portfolio model are

$$u_k = \begin{bmatrix} u_{1,k} \\ u_{2,k} \\ \vdots \\ u_{6,k} \end{bmatrix}, x_k = \begin{bmatrix} x_{1,k} \\ x_{2,k} \\ \vdots \\ x_{6,k} \end{bmatrix}, y_k = \begin{bmatrix} y_{1,k} \\ y_{2,k} \\ \vdots \\ y_{6,k} \\ y_{total,k} \end{bmatrix} \tag{6.6}$$

where $y_{total,k} = \sum_{j=1}^{6} y_{j,k}$.

4 The Load Balancing Optimisation Problem

The structure of the problem is as follows:

$$\begin{aligned} & \min_U J \\ s.t. & \\ & x_{k+1} = Ax_k + Bu_k + Ed_k \\ & y_k = Cx_k \\ & U_{min} \leq U \leq U_{max} \\ & \Delta U_{min} \leq \Delta U \leq \Delta U_{max} \end{aligned} \tag{6.7}$$

where J is a performance function which has to be minimised without violating the constraints, U is a vector containing all inputs over the prediction horizon $k = 0, \dots, N$ such that $U = [u_0^T, u_1^T, \dots, u_N^T]^T$.

4.1 Choosing a performance function

The load balancing problem has two main objectives from which the performance function should be constructed; the deviation from the reference production should be minimised, and this should be done as economically as possible.

Definition 3. In the following the weighted ℓ_1-norm is denoted as $||\cdot||_{1,q}$ with the weight q. For a vector $x = [x_1, x_2, \dots x_N]^T$ the weighted ℓ_1-norm is defined as

$$||x||_{1,q} = q_1|x_1| + q_2|x_2| + \dots + q_N|x_N|.$$

The deviation can be posed as a financial objective as well, since imbalances are fined by the TSO, who has the overall load balancing responsibility in Denmark. Posing the deviation as a financial optimisation problem entails that the overall performance function has to describe the expenses for obtaining the reference production. The cost of deviations

can be described by the weighted ℓ_1-norm such that

$$J_e = \sum_{k=1}^{N} ||y_k - r_k||_{1,q_{e,k}} \tag{6.8}$$

where y_k is the system output vector, r_k is the system reference vector and $q_{e,k}$ is a cost vector for the deviations. The cost is summed up over the prediction horizon $k = 1, \ldots, N$.

The production plan for each plant is assumed to be economically optimal, therefore the output reference for each of the units is the production plan reference (denoted $r_{j,k}$ for unit j at sample k).

The reference to the total production is combined by two sources. One source is the summed production plans for all the power plants, and the other source is the signal from the TSO ($r_{TSO,k}$). This results in a reference vector

$$r_k = \begin{bmatrix} r_{1,k} \\ r_{2,k} \\ \vdots \\ r_{6,k} \\ r_{total,k} \end{bmatrix} \tag{6.9}$$

where $r_{total,k} = r_{TSO,k} + \sum_{j=1}^{6} r_{j,k}$. Since $r_{total,k}$ has an addition from $r_{TSO,k}$, it is impossible to track all references without error in all cases where $r_{total,k} \neq 0$.

The other part of the expenses is the production costs. Intuitively, these costs should be placed on the input since they are dominated by the fuel cost. However, placing the weight on the input will make it seem beneficial to lower the input since the output deviation will not occur until some time into the future, due to the phase lag through the system. When the cost is summed over the finite horizon, the cost of lowering the input would thus yield a greater benefit than the penalty of the deviation - an unintended behaviour. Therefore the weight is placed on the output to avoid this phase lag. The production cost function is described as

$$J_u = \sum_{k=1}^{N} q_{u,k}^T y_k \tag{6.10}$$

where $q_{u,k}$ is the marginal cost factor and y_k is the system output.

A cost on input change is added to the performance function in order to dampen the input signals to the system. Even though changing the input should not have any significant cost, leaving this part out yields a significantly degraded performance, due to rapidly changing control signals that will expose the discrepancies between model and the real system. This is formulated as a weighted ℓ_1-norm yielding

$$J_{\Delta u} = \sum_{k=0}^{N-1} ||\Delta u_k||_{1,q_{\Delta u,k}} \tag{6.11}$$

where Δu_k is the change in input and $q_{\Delta u,k}$ is the penalty for changing the input.

These functions can be combined into one objective function

$$J = J_e + J_u + J_{\Delta U}. \tag{6.12}$$

This performance function can be transformed into a linear program by variable substitution as described in [Maciejowski, 2002]. The linear program can be efficiently minimised by a standard Linear Programming (LP) solver.

4.2 Notes on tuning the performance function

Since the price of deviating from the total production is not known until a few hours after the deviation, the price has to be estimated. The price of reserve activation at the power exchange Nordpool ranged from 0.5 to 93€/MWh over the period from 6 to 13 July 2007. Choosing it too low will make it beneficial to deviate from the total production creating steady-state offsets which should be avoided. Therefore an estimate of 80€/MWh is chosen for both positive and negative deviations.

A unit's deviation from the production plan is not penalised financially, only the deviation of total portfolio output is. However, it is desired to adhere to the production plan, which is why a weight is put on the individual unit's deviation from the production plan.

The weight on the deviation must be chosen such that it is not in conflict with the overall optimisation goal. However, as it is kept within an upper and lower bound, the actual weight does not influence the result. The upper bound on the unit deviation penalty is equal to the penalty for deviating from the portfolio reference. Otherwise, it would be optimal to deviate from the total output in case of disturbances.

The lower bound on the unit deviation penalty is equal to the price difference between the production costs of the cheapest and most expensive unit. Otherwise, it would always be beneficial to bring the cheapest unit to the maximum and the most expensive units to the minimum in steady state. And this would in turn compromise the assumption that the production plans are optimal when in steady state

5 Implementation and Results

The controller environment and the simulation models are implemented in Matlab/Simulink. The controller is formulated as a linear program, which means that it can be solved by an LP solver. For this purpose GLPK from [Makhorin, 2007] with the GLPKMEX matlab interface from [Giorgetti, 2007] is chosen.

5.1 Bounds and limits

Due to the formulation of constraints on the input it is possible for the production plan to move outside the operator set bounds such that the upper bound on input becomes negative or the lower bound becomes positive. The controller should not try to compensate for poorly chosen limits, so the bounds in the implementation are formulated such that

$$u_{min} \leq 0 \leq u_{max} \tag{6.13}$$

The rate limits on the units are load dependent since the process is significantly easier to control in some areas than in others; therefore, a higher rate of change is allowed in

these areas. To linearise the constraint, it is assumed that the rate limit is constant over the prediction horizon $k = 0, \ldots, N$ with the value obtained at $k = 0$.

5.2 Reference signals

The production plans are known ahead of time and are therefore used in a feed-forward manner. Unlike the production plan the reference signal r_{TSO} is generated in real time and is therefore not known. The best guess is that it will be constant into the future. However, it is known that the portfolio is supposed to respond with a filtered version of the reference signal from the TSO, and therefore a filtered version of the signal from the TSO is added to the controller reference.

5.3 Simulations

The controller is evaluated and compared to the current implementation, which consists of a PI controller structure, via simulation against a nonlinear model of the portfolio. The controller will be evaluated in two different scenarios, and each scenario will be evaluated based on two different parameters. The first parameter is the ability to perform reference tracking and disturbance rejection, formulated as:

$$\delta = \sum_{k=0}^{K} ||r_{total,k} - y_{total,k}||_1 \tag{6.14}$$

which is the portfolio deviation from the reference, summed over the whole period.

The second parameter is the production costs and deviation penalties

$$c_x = \sum_{k=0}^{K} q_{e,total}(|r_{total,k} - y_{total,k}|) + \sum_{k=0}^{K} q_f^T y_k \tag{6.15}$$

where y_k is a vector of plant output, q_f is a vector of fuel costs and $q_{e,total}$ is the cost of deviation of the portfolio. Since the deviation cost (q_e) fluctuates, the controllers are compared with a deviation cost of €0, €13 and €80 per MWh denoted with the subscripted x.

The production prices used in the evaluated scenarios are fictive but based on the different types of fuel present in the portfolio. The prices used in the evaluation are shown in Table 6.1. The prices are assumed to contain all load dependent costs of producing power on a particular unit.

Unit	1	2	3	4	5	6
Cost	22.9	24.6	18.0	43.0	26.9	28.1

Table 6.1: Price in €/MWh

5.4 Scenario 1: Output disturbance

This scenario evaluates how well the controllers perform with regard to disturbance rejection. At $t = 500s$, zero-mean gaussian noise with variance 33.2 is added to the output

of the portfolio (y_{total}). The noise emulates process disturbances, which should be suppressed by the controller. The production plans are constant throughout the scenario. Fig. 6.3 shows the scenario results with the PI controller as well as the MPC.

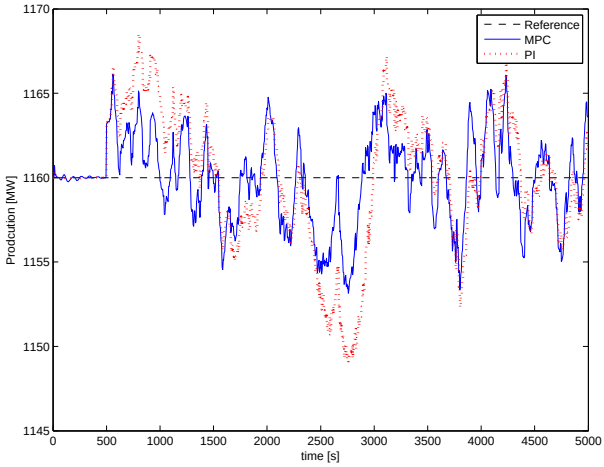


Figure 6.3: Scenario 1 - Portfolio output

The results of the objective functions (6.14) and (6.15) are found in Table 6.2.

	δ	c_0	c_{13}	c_{80}
PI	4.29 MWh	€47782	€47837	€48125
MPC	2.84 MWh	€47677	€47714	€47904
Improvement	34%	0.22%	0.26%	0.46%

Table 6.2: Scenario 1 - Comparison

The input signals from the controllers are shown in Fig. 6.4. The MPC control signals change rapidly compared to the control signals from the PI controllers. In general, this results in a better disturbance rejection for the MPC, which reduces the deviation by 34% compared to the PI controllers as seen in Table 6.2. There is a large difference in the coordination of the input signals to the units. The PI controllers distribute correction signals among all units, where the MPC exploits the knowledge on economics, using the cheapest unit when extra power is needed, and using the most expensive unit when too much power is produced. This result cannot be obtained by retuning the current implementation of the PI controllers.

5.5 Scenario 2: Signal from the TSO

This scenario evaluates the controller's capabilities of reference tracking of the signal issued by the TSO. The production plan is the same as in the previous scenario, meaning that the production plans for the individual units are constant throughout the scenario. At

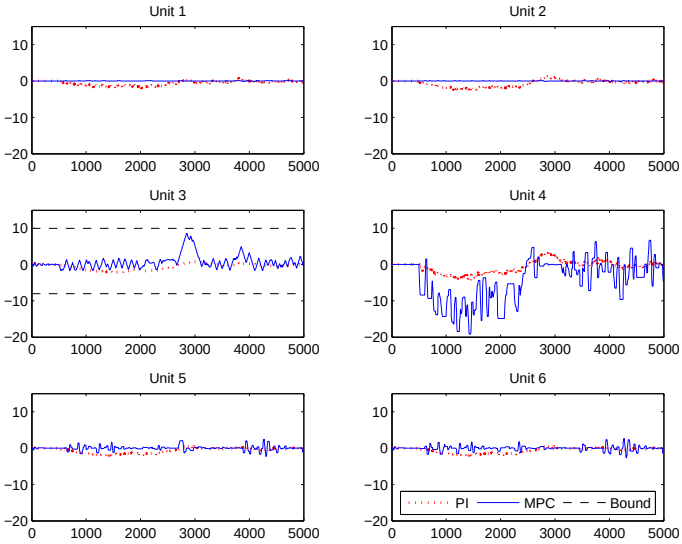


Figure 6.4: Scenario 1 - Input signals

$t = 500$ a signal applied from the TSO results in a total portfolio reference as seen in Fig. 6.5.

Table 6.3 shows the scenario results with the PI controller as well as the MPC.

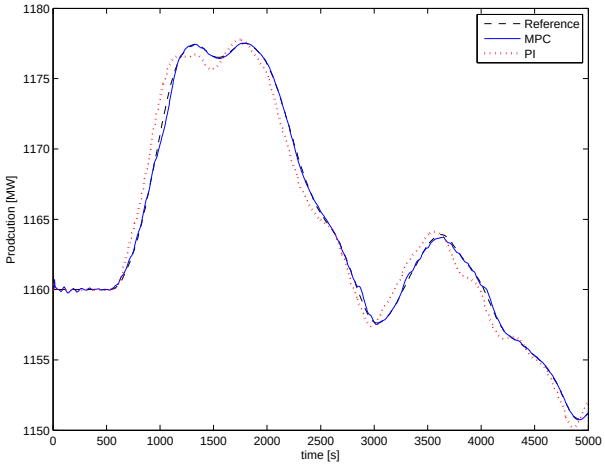


Figure 6.5: Scenario 2 - Portfolio output

The results of the performance functions (6.14) and (6.15) are found in Table 6.3. The results show that the MPC significantly reduces the deviation. The peak deviation

	δ	c_0	c_{13}	c_{80}
PI	1.02 MWh	€48081	€48095	€48163
MPC	0.16 MWh	€47994	€47996	€48007
Improvement	84%	0.18%	0.20%	0.32%

Table 6.3: Scenario 2 - Comparison

is reduced from 2.7MW to 1.1MW as shown in Fig. 6.6, and the summed deviation is reduced by 84%. This improvement originates from the MIMO approach of the MPC, which has a superior coordination of the portfolio that takes dynamics and constraints into account, unlike the ad hoc coordination used by the PI controllers. A result that is very difficult if not impossible to obtain by retuning the current configuration of PI controllers.

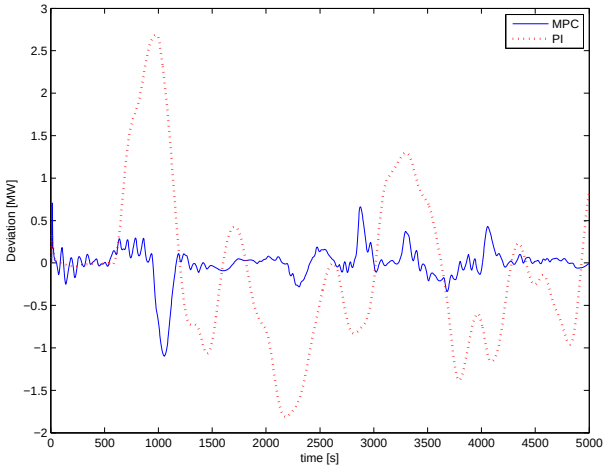


Figure 6.6: Scenario 2 - Deviation

The input signals from the controllers are shown in Fig 6.7. Once again it is seen that when extra power is needed the MPC uses the cheapest units first, and when there is an overproduction the most expensive units are lowered in order to minimise expenses. In both cases the controller returns to the production plan when possible.

6 Conclusion

This paper has introduced a model-based control approach to balance control of a portfolio of power generating units. The model-based controller uses MPC, which allows constraint handling within its framework. The MPC seeks to optimise the system based on financial considerations, thus performing reference tracking and disturbance rejection in an economically optimal way.

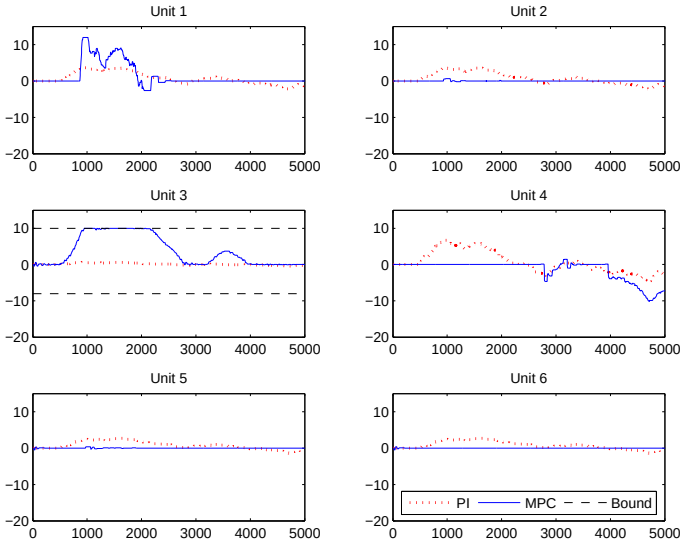


Figure 6.7: Scenario 2 - Input signals

One of the advantages of MPC is the MIMO approach, which improves the coordination of the units. The construction of the cost function as a mixture of ℓ_1-norms and linear weighting is well suited to describe the economics of the system. The choice yields a cost function, which is asymmetric around the reference, allowing for different control depending on whether the deviation is positive or negative.

Through simulations, the MPC is compared with the currently implemented load balancing controller, which is a PI controller structure. The MPC shows significant improvements both for disturbance rejection and reference tracking, and it also results in significant savings. Based on the simulations, savings of €600,000 or more per year seem likely. This improvement is the result of choosing a MIMO based approach and of the modelling of the economic behaviour in the MPC.

The portfolio in the paper is the currently active portfolio, but the goal is to incorporate more entity types than just power plant units, eg wind farms and district heating production. The model predictive controller is the first step towards developing a stringent method for portfolio control with a system containing many units, which all need to be controlled. To handle such a system, a stringent method for handling subsystems entering and leaving the portfolio will be required as well.

References

DONG Energy. `http://www.dongenergy.dk`, 2007.

N. Giorgetti. GLPKMEX - A Matlab MEX Interface for the GLPK library. http://glpkmex.sourceforge.net/, 2007.

C. Jørgensen, J. H. Mortensen, T. Mølbak, and E. O. Nielsen. Modelbased Fleet Optimization and Master Control of a Power Production System. In *Proceedings of IFAC Symposium on Power Plants and Power Systems Control 2006, Kananaskis, Canada*, 2006.

G. K. Lausterer. Improved maneuverability of power plants for better grid stability. *Control Engineering Practice*, 6(12):1549–1557, 1998.

J. M. Maciejowski. *Predictive Control with Constraints*. Pearson Education Limited, 2002. ISBN 0-201-39823-0.

A. Makhorin. Gnu Linear Programming Kit - GLPK. http://www.gnu.org/software/glpk/, 2007.

UCTE. European Wind Integration Study (EWIS) Towards a successful integration of wind power into european electricity grids. Technical report, UCTE, 2007.

E. Welfonder. Least-cost dynamic interaction of power plants and power systems. *Control Engineering Practice*, 5(9):1203–1216, 1997.

Paper C

Simple Models for Model-based Portfolio Load Balancing Controller Synthesis

Kristian Edlund, Jan Dimon Bendtsen & Tommy Mølbak

This paper was published in:
Proceedings of IFAC Symposium on power plants and power systems control
2009

The layout has been revised

Abstract

This paper presents a collection of models of so-called 'effectuators', i.e., subsystems in a power plant portfolio that represent control actions with associated dynamics and actuation costs. These models are derived in order to facilitate higher-level model-based control synthesis of a portfolio of generation units existing in an electrical power supply network, for instance in model-based predictive control or declarative control schemes. We focus on the effectuators found in the Danish power system. In particular, the paper presents models for boiler load, district heating, condensate throttling and wind turbine effectuators. Each model is validated against actual measurement data. Considering their simplicity, the models fit the observed data very well and are thus suitable for control purposes.

1 Introduction

Currently a large part of the world is deeply concerned about global warming and the consequences that might follow from emission of green house gases. This has among other things led to the signing of the Kyoto Protocol [United Nations, 1998]. Throughout Europe, this has given use to a very ambitious project to increase the share of energy delivered by renewable sources such as wind [UCTE, 2007]. In Denmark the goal is to increase the share of electrical energy coming from renewable sources from 24% in 2005 to 36% in 2025 as found in [Danish Ministry of Transport and Energy, 2005].

Due to the geography of Denmark much of this renewable energy has to come from wind turbines. Given the stochastic behaviour of the wind turbines, a flexible power system and good load balancing control is needed, in order to avoid blackouts.

DONG Energy owns and operates a portfolio of power plants in Western Denmark as shown in Fig. 7.1. With respect to the communication with the Transmission System Operator (TSO), this portfolio is considered as one entity both regarding deviations and activation of reserves. To accommodate this, DONG Energy has created a load balancing controller to minimise the deviation of the portfolio from the reference as well as distribute the ordered reserve activation.

In [Edlund et al., 2008] we showed through simulations that it was possible to significantly decrease the deviation between the portfolio output and the reference by introducing a model-based control scheme for the balance controller. The focus was to show that it is possible to gain better economics and decrease the deviation, and only little attention was paid to the models used in the control scheme. In this paper we focus on the models of the generation units. A control strategy for how to coordinate the individual effectuators is not discussed in this paper.

In the existing literature there are many detailed models of parts of the energy system, used to describe the dynamic behaviour of individual system components, such as [de Mello, 1991; Weber and Krueger, 2008; Welfonder, 1997]. However, the aim in this paper is to construct simple models which are suitable for controller synthesis of a model-based control scheme for load balancing control.

There are many ways to manipulate the output from the portfolio to follow the references. Here we shall use the term *effectuator* as a unifying term for all of them.

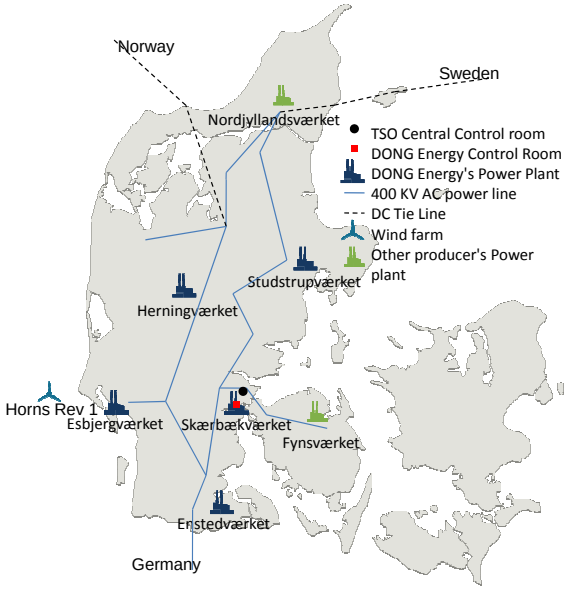


Figure 7.1: Generators participating in balance control in Western Denmark

Definition 4. An effectuator is a process or part of a process in a power system that represents control actions with associated dynamics and actuation costs allowing the power output to be manipulated.

Some parts of the power system, eg power plants, contain multiple ways of changing the power output and will therefore be treated as providing more than one effectuator. Regarding the effectuators as individual system is a novel approach when comparing to eg [Lausterer, 1998].

The paper is structured such that the description and modelling of the effectuators are in Section 2. This is followed by a validation of each effectuator in Section 3, and a discussion in Section 4.

2 Modelling

A description and mathematical model of each of the four types of effectuator is presented in this section. The first three effectuators are physically parts of the thermal power plants, while the fourth - wind turbines - are located elsewhere eg off-shore.

2.1 Boiler Load Effectuator

The boiler load effectuator affects the whole steam cycle. It is activated by offsetting the production reference. The boiler has an operating range, shown in the PQ-diagram in Fig.7.2. The district heating production (Q) is plotted along the x-axis and the power production (P) along the y-axis. There are upper and lower limits on the power production, which dependent on the current district heating production.

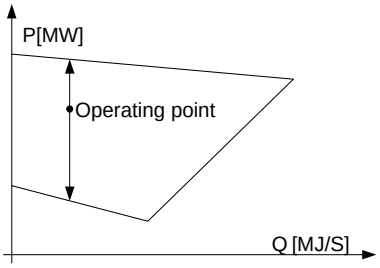


Figure 7.2: Movement in the PQ-diagram when changing the boiler load

When using the boiler for control purposes the district heating production is maintained, meaning that the changes in production happens vertically in the PQ-diagram as shown in Fig. 7.2.

This effectuator is slow in a load balancing context (minutes), but the potential energy production is unlimited, meaning that the corrections can be maintained by this effectuator for an unlimited time period. There is a large amount of power available in this type of effectuator.

Besides the behaviour of the boiler, there is a communication delay between the load balancing controller and the power plant. However, this delay is so small compared to the dynamics that it can be neglected for this effectuator.

Effectuator Model

There are two communication methods for activating the boiler load effectuator in a power plant, either through the production plan or through an input used by the balance controller to give real time corrections to the boiler load effectuator. The production plan is not controllable and is therefore modelled as a disturbance. The output of the model is the produced power from the unit caused by the boiler load effectuator.

The model is formulated as a greybox model and the dynamics are assumed to be adequately described as the following state space system

$$\begin{aligned} \dot{x}_b &= \begin{bmatrix} -T_b^{-1} & 0 & 0 \\ T_b^{-1} & -T_b^{-1} & 0 \\ 0 & T_b^{-1} & -T_b^{-1} \end{bmatrix} x_b + \begin{bmatrix} T_b^{-1} \\ 0 \\ 0 \end{bmatrix} u + \begin{bmatrix} T_b^{-1} \\ 0 \\ 0 \end{bmatrix} d \\ y &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} x_b. \end{aligned} \tag{7.1}$$

where T_b is the time constant for the effectuator. u is the input given by the balance controller, d is the production plan and additional manually ordered corrections.

Upper and Lower limits

The limits can be set by the operator, or can be given by the process. The limits are applied to the input such that $\underline{P}_b \leq u \leq \overline{P}_b$. It is assumed that the upper limit is non-negative and the lower limit is non-positive.

Only the limits derived from the process are described here. The upper limit can be approximated by a linear constraint as a function of the district heating production.

$$\overline{P_b} = -\alpha_{max} Q + \beta_{max} - d \tag{7.2}$$

where α_{max} is an approximation of the C_v value (see below) at maximum load, Q is the current district heating production and β_{max} will be the maximum power production at no district heating production.

The lower limit can be described as piecewise linear function

$$\underline{P_b} = max \begin{cases} \alpha_{min} Q + \beta_{min} - d \\ C_m Q + \beta_b - d \end{cases} \tag{7.3}$$

where α_{min} is an approximation of the C_v value at minimum load and β_{min} will be the minimum power production in condensation mode. The lower equation is a linear approximation of the pure back pressure line as shown in Fig. 7.3.

Rate Constraints

The rate limits are all piecewise linear functions of power and district heating production. The rate limits are set in the control system, and can therefore be determined precisely.

The rate limit for the balance controller is the absolute rate limit minus the disturbance (d). It is assumed that zero is always in the interval between the lower and upper rate limit.

2.2 District Heating Effectuator

The centralised Danish power plants with district heating production have a possibility to bypass part of the power generation process and instead use the energy to produce district heating. Unlike power production, district heating is not a just-in-time product since it can easily be stored. Usually there are accumulator tanks close to the thermal power plants which can store multiple hours worth of production at maximum capacity.

The district heating production can be exchanged for power production while maintaining boiler load. When doing so the production will move in the PQ-diagram as illustrated in Fig. 7.3.

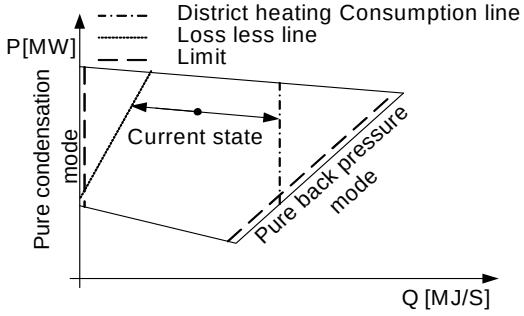


Figure 7.3: Movement in the PQ-diagram when changing the District heating

The loss of steam for power production is highly dependent on the state of the unit. Lowering the power output with $1MW$ will typically increase the district heating production with 2-7 MW. The exchange factor is called the C_v-value and is a nonlinear function of the boiler load, typically the function is in the range $0.1 - 0.4$.

When used as an effectuator, there are several constraints which should not be violated, since it results in changes in the physical process. Fig. 7.3 shows an example of such a constraint the district heating consumption line. Crossing the consumption line means that the accumulator tank goes from charging to discharging, and that requires a large discrete change in the physical process.

The district heating effectuator is characterised as a medium fast effectuator (approx. 30 sec). The potential energy is limited due to the accumulator tanks. The potential power that can be drawn from this effectuator is big, but it is still smaller than the power in the boiler load effectuator.

Besides the behaviour of the district heating system, there is a communication delay between the load balancing controller and the power plant.

Effectuator Model

The desired input to the model is a reference to the power output, and the desired output is the actual power production from the effectuator. The model constructed here is a nonlinear model.

The change in district heating is modeled as a first order linear system with a time delay. The time constant in the system is typically around $30s$ while the communication delay t_d is a result of interacting computer systems communicating over a network with no real time guarantees. The delay is therefore treated as being stochastic, but it typically ranges between 5 and 10 seconds. To convert the system so the output is power production the input and output values are multiplied and divided by C_v, thus yielding a nonlinear state space model.

$$\begin{aligned} \dot{x}_{dh}(t) &= -\frac{1}{30}x_{dh}(t) + \frac{1}{30C_v}u(t-t_d) \\ y(t) &= -C_v x_{dh}(t). \end{aligned} \tag{7.4}$$

Upper and Lower Limits

This model will only include limits for the charging lines and the plants physical limitations, such as a minimum production of no district heating.

The limits are implemented in the local control system so it is reasonable to apply the limits to the input of the model. The input to the effectuator should be within the limits $\underline{P_{dh}} \leq u \leq \overline{P_{dh}}$, and it is assumed that 0 is always a valid solution for the inequality.

The upper limit is given as the minimum of two constraints

$$\overline{P_{dh}} = min \begin{cases} C_v Q_{plan} \\ C_v(Q_{plan} - Q_{consump}) \end{cases} \tag{7.5}$$

where Q_{plan} is the production plan for district heating, and $Q_{consump}$ is the district heating consumption. The lower constraint is only active if the unit is charging the accumulator tank

The minimum limit is given by the back pressure line. The equivalent condensation mode production corresponding to the current load is found as

$$P_{con} = C_v Q + P \tag{7.6}$$

The distance between the district heating plan and a a linear approximation of the back pressure line as a function of P_{con} is used to find the maximum district heating production possible at that load. Thus the minimum power correction is found as

$$\underline{P_{dh}} = min \begin{cases} -C_v(\alpha_{back} P_{con} + \beta_{back} - Q_{plan}) \\ C_v(Q_{plan} - Q_{consump}) \end{cases} \tag{7.7}$$

The lower constraint is only active if the unit is discharging the accumulator tank.

Rate Constraints

There are rate limits on how fast the district heating production can be changed, which means that the changes from the district heating effectuator plus the district heating production should not be changes faster than a given limit. This limit is assumed constant throughout the whole state space. The rate limits can be expressed as

$$\Delta P_{dh} = \alpha_{\Delta dh} C_v \tag{7.8}$$

where $\alpha_{\Delta dh}$ is the district heating production rate of change constraint constant. The limit is typically $30 - 50 MJ/s/min$, which with a typical C_v value of 0.2 gives a rate limit of $6 - 10 MW/min$ in the power production.

2.3 Condensate Throttling Effectuator

The condensate system is the system that preheats the condensate and transports it from the condenser to the feedwater tank.

An example of a condensate system is depicted in Fig. 7.4. By changing the condensate flow, the steam demand can be changed and thereby affect the power output quickly. For more details on this effectuator see [Lausterer, 1998] as well as [Welfonder, 1997], where the authors also propose to use this effectuator to control the power output from the power plant unit.

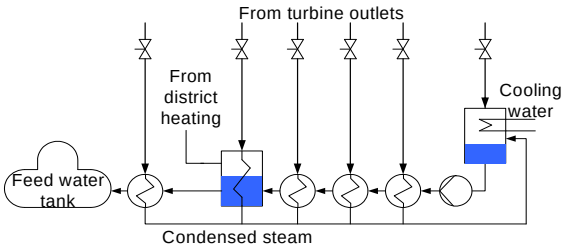


Figure 7.4: Example of a condensate System

The nominal flow in the condensate system varies with the boiler load, therefore energy and power limits varies as a function of this. At all times neither the condenser nor

the feed water tank should be completely emptied or filled. This means that the effectuator can deliver a limited amount of energy.

There is typically very little energy stored in the condensate throttling system, so it can only be used for a few minutes to manipulate the power production. On the other hand it reacts in a matter of seconds. The available power is in general much smaller than the energy the available power in the boiler load effectuator.

Besides the behaviour of the condensate system, there is a communication delay between the load balancing controller and the power plant.

Effectuator Model

The dynamics of the system from reference to power output can be approximated by a lineaer low-pass filter and a time delay. The time constant is 20 seconds for the whole operating range. The time delay is equal to the one found in the district heating effectuator.

The ratio between flow and power is a nonlinear function $f(P,Q)$ dependent on the power and district heating production.

The model of the system can be put in the form of a parameter varying state space system.

$$\dot{x}_c = \begin{bmatrix} -1/20 & 0 \\ f(P,Q) & 0 \end{bmatrix} x_c(t) + \begin{bmatrix} 1/20 \\ 0 \end{bmatrix} u(t-t_d) \tag{7.9}$$

$$y = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_c(t). \tag{7.10}$$

The output from the statespace model is

$$y = \begin{bmatrix} P(MW) \\ V(m^3) \end{bmatrix} \tag{7.11}$$

where the first term is the current power output from the system, and the second term is the water volume displaced from the setpoint of the tank. The model is valid as long as the states are kept within the described bounds.

Upper and Lower Limits

The limits are described in the control system as a piecewise linear function dependent on the current power and district heating production.

It is assumed that the condenser and tank where the district heating condensate enters the condensate system are balanced, such that the flow between condenser and feedwater tank is coordinated according to the flow entering from the district heating. The system consists of three tanks, where the water is moved between, but it will be modelled as one tank, where it is possible pour water into, or drain water. As long as the level is kept within certain bounds. These level bounds in this one virtual tank, originates from the most restricting of the three tanks for upper and lower bounds.

The limits of the volume used in the model are defined as

$$\overline{V}_c = min \begin{cases} \overline{V}_{cond} - V_{0,cond} \\ \overline{V}_{lph4} - V_{0,lph4} \\ V_{0,fwt} - \underline{V}_{fwt} \end{cases} \quad \underline{V}_c = min \begin{cases} \underline{V}_{cond} - V_{0,cond} \\ \underline{V}_{lph4} - V_{0,lph4} \\ V_{0,fwt} - \overline{V}_{fwt} \end{cases} \tag{7.12}$$

with V_0, xx being the initial volume in tank xx.

2.4 Wind turbines

The focus here is the wind turbine farms. Wind turbine farms are a collection of wind turbines, which are controlled as one entity. The turbines are gathered in a small geographical area, which means that the wind condition is approximately the same for all turbines.

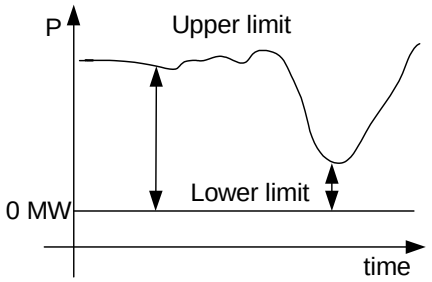


Figure 7.5: Limits on wind turbines power production

On Fig. 7.5 the possible production is shown for a wind farm. Different control strategies are described in [Bjerge and Kristoffersen, 2007] with some of them based on the possibility to incorporate the windfarm in the balance control. It is assumed that the setpoint follows the upper limit with a constant offset, which might be $0MW$

A wind turbine can react very quickly to set point changes. But there will be a maximum rate of change, this rate of change can either originate from the mechanical limits of the construction or an artificial limit enforced in order to preserve the mechanical parts in the wind turbines. The experience from the offshore wind farms is an effectuator which has a rate of change approximately as slow as the boiler load effectuator but much faster dynamics (few seconds). If the wind speed is maintained the corrections can be maintained for an unlimited amount of time.

In addition to the dynamics of each wind turbine, there is a communication delay between the load balancing controller and the wind turbines.

Effectuator

A simple first order linear low-pass filter with a time delay is used. The input to the model is the desired offset from the setpoint, and the output is the current offset from setpoint.

The model is described in state space as

$$\begin{aligned} \dot{x}_w(t) &= -\frac{1}{T_w}x_w(t) + \frac{1}{T_w}u(t-t_d) \\ y(t) &= x_w(t). \end{aligned} \tag{7.13}$$

Where T_w is the time constant for the wind park. A model of the wind turbine gives a $T_w \approx 0.1s$. The communication delay t_d is the same as in the district heating effectuator.

Limits

The upper limit is given by the distance from the setpoint to the upper limit, which is set by the operator. The negative correction is bounded by zero.

Rate Limits

The described rate limit is applied to the input of the model. This yields the following rate limits

$$-\Delta u_w \leq \Delta u \leq \Delta u_w \tag{7.14}$$

where u is the input to the model.

3 Verification

The models consist of both a dynamic part, and a constraint set. The constraints are either found and verified using a static tool such as Turabs [Johansen, 2004], or derived from the implementation in the control system. The parameters in the model are estimated by hand.

3.1 Boiler Load Effectuator

The models are likely going to be used in a predictive control scheme. However since the prediction horizon of the scheme is unknown it is chosen to evaulate the models open-loop performance.

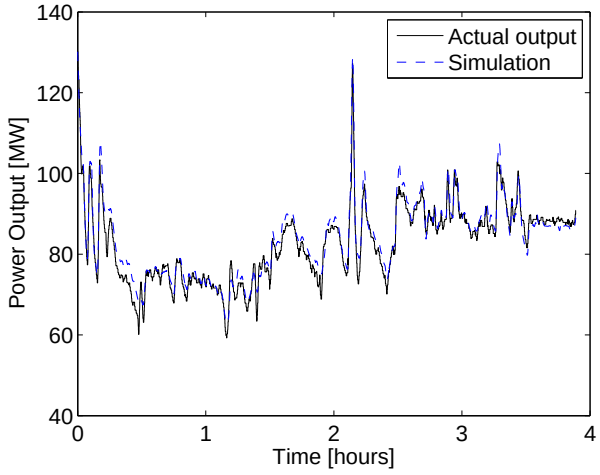


Figure 7.6: Verification of the dynamics of the power plant, scenario 1. The solid line shows the actual output, and the dashed line shows the open-loop simulation

Fig. 7.6 shows the first evaluation scenario. The model is a little offset from the actual output. This is most likely caused by the measurements used for model validation being

slightly different than the ones used for the actual control. The trends in the dynamics are followed well for the prediction.

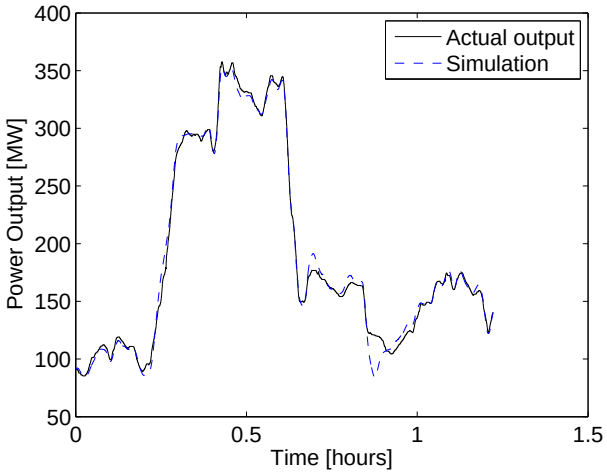


Figure 7.7: Verification of the dynamics of the power plant, scenario 2. The solid line shows the actual output, and the dashed line shows the open-loop simulation

Fig. 7.7 shows the first evaluation scenario and the model is a little offset from the actual output, but the trends in the dynamics are followed decently, both for the prediction and the simulation. Between 0.8 hours and 1 hour, the model to diverges from the actual output, which has been tracked to using different limits in different parts of the system. This should be corrected in the control system.

3.2 District Heating Effectuator

For this verification the boiler load has been maintained at a constant level. At $t = 14s$ a step has been applied to the system and the response shown in the figure was observed.

A first order linear model with time delay gives a good approximation to the measured response. A time constant of $10s$ and a delay of $7s$ is found to give the best fit.

3.3 Condensate Effectuator

The best currently available data for estimation and validation are shown in Fig. 7.9. This data shows a short series of step test applied to the system while the boiler load is maintained in steady state.

When the effectuator is activated it responds to the input. However the measurements are strongly affacted by process noise from the power plant causing the small series of steps to be insufficient for parameter estimation. The standard variation of the noise is estimated to $\sigma = 2.70$. When plotting the model output $\pm 3\sigma$, 98% of the measurements should lie within the the resulting band if the noise is normally distributed. In this sequence, 7.8% of the samples lie outside the band. From this result the model is accepted as describing the behaviour well enough.

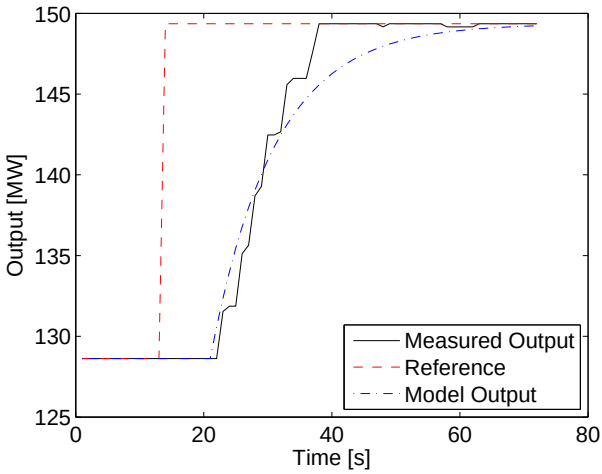


Figure 7.8: Validation for the District heating response. The solid line shows the actual output, and the dashed line shows the open-loop simulation.

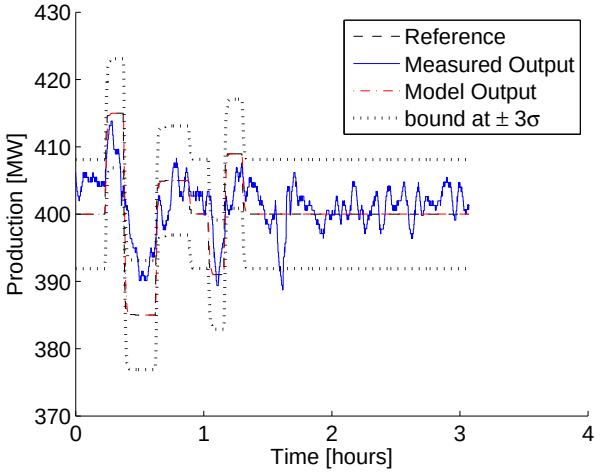


Figure 7.9: Validation for the condensate throttling system

3.4 Wind Turbine Effectuator

The model dynamics are approximately as fast as the sample time of the available data, so it has not been possible to make a numeric validation. However there are various complex models available for a single wind turbine. An unpublished model developed within DONG Energy has been used to identify the dynamic response.

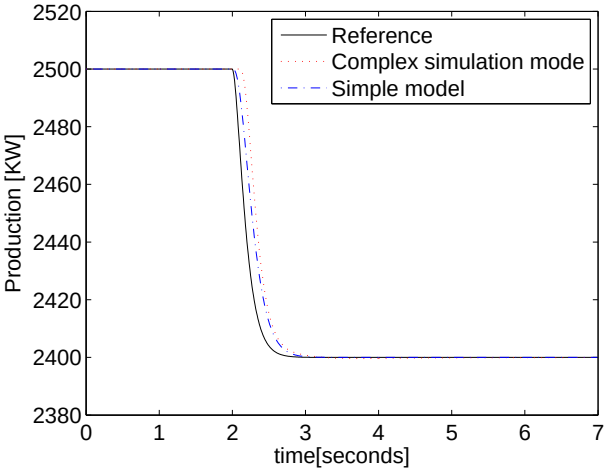


Figure 7.10: Validation for the wind turbine model

Fig. 7.10 shows a comparison of a complex windturbine model and the simple model when the windturbine makes a change in the reference. As can be seen, the dynamics are very fast, and therefore the rate limiter in the controller will be the dominating the dynamical factor of a wind turbine system. It is assumed that a wind turbine park, will behave approximately like a single turbine.

4 Conclusion

The paper characterise four different so-called *effectuators* which can be used for load balancing control.

The different effectuators are characterised as summarised in Table 7.1. Some of the effectuator are limited in the energy available for load balancing purposes, while other are capable of maintaining the correction for a long period. Simple models have been derived of each effectuator with the aim of making them suitable for controller synthesis, in particular in an model predictive control scheme.

The aim has been to create a set of simple models which describe the dynamics adequately for use in synthesis of a load-balancing controller. The models for the effectuators are described as nonlinear state space systems with up to three states and a time delay. Such simple models are well suited for controller synthesis in a model based controller scheme such as model predictive control.

effectuator	Energy	Power	Dynamics
Boiler load	Unlimited	Large	$> 1min$
District heating	Large	Medium	$20 - 40s$
Condensate throttling	Small	Medium	$10 - 15s$
Wind turbines	Unlimited	Medium	$< 5s$

Table 7.1: Comparison of the effectuators.

References

C. Bjerge and J. R. Kristoffersen. Run an offshore wind farm like a power plant. *VGB Powertech*, 87(4):63–66, 2007.

Danish Ministry of Transport and Energy. Energistrategi 2025 - Perspektiver frem mod 2025. Technical report, Transport- og Energiministeriet, 2005.

F. P. de Mello. Boiler models for system dynamical performance studies. *IEEE Transactions on Power Systems*, 6(1):66–74, 1991.

K. Edlund, J. D. Bendtsen, S. Børresen, and T. Mølbak. Introducing Model Predictive Control for Improving Power Plant Portfolio Performance. *Proceedings of the 17th IFAC World Congress, Seoul, South Korea*, pages 6986–6991, 2008.

A. O. Johansen. Simulation and optimisation of thermal power stations by use of Turabs. Technical report, DONG Energy, 2004.

G. K. Lausterer. Improved maneuverability of power plants for better grid stability. *Control Engineering Practice*, 6(12):1549–1557, 1998.

UCTE. European Wind Integration Study (EWIS) Towards a successful integration of wind power into european electricity grids. Technical report, UCTE, 2007.

United Nations. Kyoto protocol to the United Nations framework convention on climate change. Technical report, United Nations, 1998.

H. Weber and M. Krueger. Dynamic investigation of network restoration by the pumped-storage plant Markersbach in Germany. *Proceedings of the 17th IFAC World Congress, Seoul, Korea*, 2008.

E. Welfonder. Least-cost dynamic interaction of power plants and power systems. *Control Engineering Practice*, 5(9):1203–1216, 1997.

Paper D

A Primal-Dual Interior-Point Linear Programming Algorithm for MPC

Kristian Edlund, Leo Emil Sokoler & John Bagterp Jørgensen

This paper was published in:
Proceedings of 48th IEEE Conference on Decision and Control

Abstract

Constrained optimal control problems for linear systems with linear constraints and an objective function consisting of linear and l_1-norm terms can be expressed as linear programs. We develop an efficient primal-dual interior point algorithm for solution of such linear programs. The algorithm is implemented in Matlab and its performance is compared to an active set based LP solver and linprog in Matlab's optimization toolbox. Simulations demonstrate that the new algorithm is more than one magnitude faster than the other LP algorithms applied to this problem.

1 Introduction

In MPC applications, the performance and reliability of the optimization algorithm solving the constrained optimal control problem is important as the optimization problem is solved repeatedly online. In this paper we develop a primal-dual interior-point algorithm for model predictive control (MPC) with input and input-rate constraints and an objective function consisting of linear stage costs as well as l_1-norms penalizing deviation from target and movements [Maciejowski, 2002; Boyd and Vandenberghe, 2004; Edlund et al., 2008]. The primal-dual interior point algorithm is based on Mehrotra's predictor-corrector algorithm [Mehrotra, 1992; Wright, 1997; Zhang, 1998; Czyzyk et al., 1999; Nocedal and Wright, 2006]. Linear programs for MPC have previously been considered by [Morshedi et al., 1985; Allwright and Papavasiliou, 1992; Rao and Rawlings, 2000]. Interior-point algorithms based on Riccati iterations for solution of an l_2 constrained regulation problem [Rao et al., 1998] and a robust l_1 constrained regulation problem [Vandenberghe et al., 2002] have been reported. In this paper, we use state elimination to construct a structured linear program with upper and lower limits on the decision variables, and highly structured general constraints. The special structure of the constraints in this linear program is utilized by the primal-dual interior-point algorithm.

1.1 Power Portfolio Control

DONG Energy is the main power generating company in Denmark. It operates a portfolio of power plants and wind turbine farms for electricity and district heating production. The wind turbines constitute a large share: 30% of the installed generation capacity in Western Denmark. The share is expected to increase even further as a new wind turbine park is added to the portfolio at the end of 2009. In addition a large pool of electric cars are added to the power network.

In a liberalized electricity market, such an interconnected power and heating system with significant stochastic generators and consumers needs an agile and robust control system to coordinate the most economic power generation respecting constraints, long-term contracts, and short-term demand-fluctuations.

By simulation Model Predictive Control has been demonstrated as a very promising technology for dynamic regulation and coordination of power generation in the DONG Energy portfolio [Edlund et al., 2008]. This controller is called the DONG Energy portfolio balance controller. The controller reduces the deviation between sold and actual production in the most economical way. This is an example of Model Predictive Control

with an economical rather than a traditional target deviation objective function [Rawlings and Amrit, 2009].

The models used in this paper has been derived in [Edlund et al., 2009]. To test different optimization algorithms, and the possibility to exploit the structure of the problem, we consider a single subsystem of the entire power generation portfolio. The subsystem is a single boiler load effectuator with the simplification that rate-of-movement limits can be specified as parameters [Edlund et al., 2009].

1.2 Paper Organization

In Section 2 we state the constrained optimal control problem with a linear cost and l_1-norm penalties. We derive the LP problem used to compute the solution of the constrained optimal control problem. Section 3 describes the interior-point algorithm for an inequality constrained linear program. Section 4 specializes the operations in this algorithm to the LP problem for the constrained optimal control problem with linear cost and l_1-penalties. Section 5 compares the developed interior-point algorithm for the MPC-LP to off-the-shelf LP solvers. Section 6 concludes on the results.

2 Problem Definition

We state the control problem that is to be used in the power balance controller in controlling one power generating unit (a power plant). The problem and the models are described in detail in [Edlund et al., 2008, 2009]. The power balance controller is a Model Predictive Controller in which a constrained optimal control problem is solved at each sampling instant. Only the input associated to the first time period is implemented and the computations are repeated at the next sampling instant. We consider long horizons to have economic performance as well as stability. This implies that the constrained optimal control problem solved at each sampling instant is relatively large. It is important that this large constrained optimal control problem is solved robustly and fast as it is embedded in a real-time system.

The objective function used to measure the quality of a power trajectory is

$$\phi = \sum_{k=0}^{N-1} c'_{k+1} z_{k+1} + \|z_{k+1} - r_{k+1}\|_{1,q_{k+1}} + \|\Delta u_k\|_{1,s_k} \tag{8.1}$$

z_k is the output (power production), r_k is the reference (planned power production), and u_k is the input (modified power production to meet short term fluctuations in demand). k is a time index and we consider these cost for a finite period, $\mathcal{N} = \{0, 1, \ldots, N-1\}$, characterized by the control and prediction horizon, N.

The first term represent the production costs, i.e. the cost of fuel, emission taxes etc. The second term describes the costs for deviating from the production plan computed by the production planner. The last term is a cost related to plant wear that penalizes excessive movement of the input.

The models describing the dynamics of the system are linear. The inputs have bound and rate-of-movement constraints [Edlund et al., 2009]. Therefore, the constrained opti-

mal control problem solved at each sampling period is

$$\min_{\{u_k\}_{k=0}^{N-1}} \quad \phi = \phi(\{u_k\}_{k=0}^{N-1}; x_0, u_{-1}, \{d_k, r_{k+1}\}_{k=0}^{N-1}) \tag{8.2a}$$

$$s.t. \quad x_{k+1} = Ax_k + Bu_k + Ed_k \qquad k \in \mathcal{N} \tag{8.2b}$$

$$z_{k+1} = Cx_{k+1} \qquad k \in \mathcal{N} \tag{8.2c}$$

$$u_{\min,k} \leq u_k \leq u_{\max,k} \qquad k \in \mathcal{N} \tag{8.2d}$$

$$\Delta u_{\min,k} \leq \Delta u_k \leq \Delta u_{\max,k} \qquad k \in \mathcal{N} \tag{8.2e}$$

$\mathcal{N} = \{0, 1, \ldots, N-1\}$. Note that the input bounds and the rate-of-movement constraints are time varying.

Combination of (8.2b) and (8.2c) yields

$$z_k = CA^k x_0 + \sum_{i=0}^{k-1} H_{u,k-i} u_i + \sum_{i=0}^{k-1} H_{d,k-i} d_i \tag{8.3}$$

with $k = 1, 2, ..., N$ and the impulse response coefficients defined in the usual way

$$H_{u,i} = CA^{i-1}B \qquad i = 1, 2, \ldots, N \tag{8.4a}$$

$$H_{d,i} = CA^{i-1}E \qquad i = 1, 2, \ldots, N \tag{8.4b}$$

Define the vectors

$$U = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix} \quad D = \begin{bmatrix} d_0 \\ d_1 \\ \vdots \\ d_{N-1} \end{bmatrix} \quad \Delta U = \begin{bmatrix} \Delta u_0 \\ \Delta u_1 \\ \vdots \\ \Delta u_{N-1} \end{bmatrix}$$

$$Z = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_N \end{bmatrix} \quad R = \begin{bmatrix} r_1 \\ r_2 \\ \vdots \\ r_N \end{bmatrix} \quad V = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_N \end{bmatrix} \quad W = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix}$$

and the matrices

$$\Phi = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{N-1} \end{bmatrix} \quad \Gamma_\alpha = \begin{bmatrix} H_{\alpha,1} & 0 & \ldots & 0 \\ H_{\alpha,2} & H_{\alpha,1} & \ldots & 0 \\ \vdots & \vdots & & \\ H_{\alpha,N} & H_{\alpha,N-1} & \ldots & H_{\alpha,1} \end{bmatrix}$$

with $\alpha \in \{u, d\}$. Using (8.3) the stacked outputs, Z, may be expressed by the linear relation

$$Z = \Phi x_0 + \Gamma_u U + \Gamma_d D \tag{8.5}$$

Introduce the matrices (shown for the case $N = 5$)

$$I_0 = \begin{bmatrix} I \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad \Psi = \begin{bmatrix} I & 0 & 0 & 0 & 0 \\ -I & I & 0 & 0 & 0 \\ 0 & -I & I & 0 & 0 \\ 0 & 0 & -I & I & 0 \\ 0 & 0 & 0 & -I & I \end{bmatrix}$$

to have the following expression

$$\Delta U = \Psi U - I_0 u_{-1} \tag{8.6}$$

Consequently, the constrained optimal control problem (8.2) may be expressed as

$$\begin{aligned}
\min_{U} \quad & \phi = c'Z + \|Z - R\|_{1,q} + \|\Delta U\|_{1,s} && \text{(8.7a)}\\
s.t. \quad & Z = \Phi x_0 + \Gamma_u U + \Gamma_d D && \text{(8.7b)}\\
& \Delta U = \Psi U - I_0 u_{-1} && \text{(8.7c)}\\
& U_{\min} \leq U \leq U_{\max} && \text{(8.7d)}\\
& \Delta U_{\min} \leq \Delta U \leq \Delta U_{\max} && \text{(8.7e)}
\end{aligned}$$

Theorem 4 (Linear Program for l_1-approximation)**.** *The l_1-approximation problem*

$$\min_{x \in \mathbb{R}^n} \quad \phi = \|Ax - b\|_1 \tag{8.8}$$

with $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$ can be represented as the linear program

$$\begin{aligned}
\min_{x,y} \quad & \phi = e'y && \text{(8.9a)}\\
s.t. \quad & -y \leq Ax - b \leq y && \text{(8.9b)}
\end{aligned}$$

with $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$, and $e = \begin{bmatrix} 1 & \dots & 1 \end{bmatrix}'$.

Proof. The l_1-approximation problem (8.8) is equivalent to $\min_{x,y} \{\phi = e'y : y \geq |Ax - b|\}$. The constraint $y \geq |Ax - b|$ may be written as the linear constraints $-y \leq Ax - b \leq y$. □

Corollary 1 (l_1-approximation as LPs in standard form)**.** *The l_1-approximation problem (8.8) may be expressed as the linear program in the form*

$$\begin{aligned}
\min_{x,y} \quad & \phi = \begin{bmatrix} 0 \\ e \end{bmatrix}' \begin{bmatrix} x \\ y \end{bmatrix} && \text{(8.10a)}\\
s.t. \quad & \begin{bmatrix} A & I \\ -A & I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \geq \begin{bmatrix} b \\ -b \end{bmatrix} && \text{(8.10b)}
\end{aligned}$$

(8.8) may also be expressed as the linear program in the form

$$\begin{aligned}
\min_{x,y} \quad & \phi = \begin{bmatrix} 0 \\ e \end{bmatrix}' \begin{bmatrix} x \\ y \end{bmatrix} && \text{(8.11a)}\\
s.t. \quad & \begin{bmatrix} b \\ -\infty \end{bmatrix} \leq \begin{bmatrix} A & I \\ A & -I \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \leq \begin{bmatrix} \infty \\ b \end{bmatrix} && \text{(8.11b)}
\end{aligned}$$

Proof. Follows by rearrangement of (8.9). □

Using Theorem 4 we may express (8.7) as

$$\min_{U,V,W} \quad \phi = c'Z + s'V + q'W \tag{8.12a}$$
$$s.t. \quad Z = \Phi x_0 + \Gamma_u U + \Gamma_d D \tag{8.12b}$$
$$\Delta U = \Psi U - I_0 u_{-1} \tag{8.12c}$$
$$U_{\min} \leq U \leq U_{\max} \tag{8.12d}$$
$$\Delta U_{\min} \leq \Delta U \leq \Delta U_{\max} \tag{8.12e}$$
$$-V \leq \Delta U \leq V \tag{8.12f}$$
$$-W \leq Z - R \leq W \tag{8.12g}$$

which by elimination of Z and ΔU is equivalent to the inequality constrained linear program

$$\min_{U,V,W} \quad \phi = c'(\Phi x_0 + \Gamma_u U + \Gamma_d D) + s'V + q'W \tag{8.13a}$$
$$s.t. \quad U_{\min} \leq U \leq U_{\max} \tag{8.13b}$$
$$\Delta U_{\min} \leq \Psi U - I_0 u_{-1} \leq \Delta U_{\max} \tag{8.13c}$$
$$-V \leq \Psi U - I_0 u_{-1} \leq V \tag{8.13d}$$
$$-W \leq \Phi x_0 + \Gamma_u U + \Gamma_d D - R \leq W \tag{8.13e}$$

This linear program along with Corollary 1 may be used to arrive at the following linear program

$$\min_{x} \quad \psi = g'x \tag{8.14a}$$
$$s.t. \quad x_l \leq x \leq x_u \tag{8.14b}$$
$$b_l \leq Ax \leq b_u \tag{8.14c}$$

with the variables and data defined as

$$x = \begin{bmatrix} U \\ V \\ W \end{bmatrix} \; x_l = \begin{bmatrix} U_{\min} \\ 0 \\ 0 \end{bmatrix} \; x_u = \begin{bmatrix} U_{\max} \\ \infty \\ \infty \end{bmatrix} \; g = \begin{bmatrix} g_u \\ s \\ q \end{bmatrix} \tag{8.15a}$$

$$A = \begin{bmatrix} \Psi & 0 & 0 \\ \Psi & I & 0 \\ \Psi & -I & 0 \\ \Gamma_u & 0 & I \\ \Gamma_u & 0 & -I \end{bmatrix} \tag{8.15b}$$

$$b_l = \begin{bmatrix} \Delta U_{\min} + I_0 u_{-1} \\ I_0 u_{-1} \\ -\infty \\ b \\ -\infty \end{bmatrix} \; b_u = \begin{bmatrix} \Delta U_{\max} + I_0 u_{-1} \\ \infty \\ I_0 u_{-1} \\ \infty \\ b \end{bmatrix} \tag{8.15c}$$

$$g_u = \Gamma_u' c \tag{8.15d}$$
$$b = R - (\Phi x_0 + \Gamma_d D) \tag{8.15e}$$

The original objective function is $\phi = \psi - c'b$ where $c'b$ is a constant.

Consequently, we may solve the constrained optimal control problem (8.2) by solution of the linear program (8.14). The coefficient matrix (8.15b) is highly structured. It is composed of the matrices Ψ and Γ_u which themselves are structured matrices. We develop a primal-dual interior-point algorithm that exploits this structure to efficiently solve the constrained optimal control problem (8.2) in the MPC.

3 Interior-Point Methods

Before proceeding to a description of the interior-point algorithm applied to (8.14), we describe the interior-point algorithm for the structural simpler linear program

$$\min_{x \in \mathbb{R}^n} \quad \phi = g'x \tag{8.16a}$$

$$s.t. \qquad Ax \geq b \tag{8.16b}$$

The algorithm and its principles are discussed in [Nocedal and Wright, 2006].

3.1 Optimality Conditions

The Lagrangian of (8.16) is

$$\mathcal{L}(x, \lambda) = g'x - \lambda'(Ax - b) \tag{8.17}$$

and a stationary point of the Lagrangian satisfies

$$\nabla_x \mathcal{L}(x, \lambda) = g - A'\lambda = 0 \tag{8.18}$$

Consequently, the first order necessary and sufficient optimality conditions may be stated as

$$g - A'\lambda = 0 \tag{8.19a}$$

$$Ax - b \geq 0 \quad \perp \quad \lambda \geq 0 \tag{8.19b}$$

in which $\perp$ is used to denote complementarity. Introduce slack variables defined as

$$s = Ax - b \geq 0 \tag{8.20}$$

and let

$$S = \begin{bmatrix} s_1 & & & \\ & s_2 & & \\ & & \ddots & \\ & & & s_m \end{bmatrix} \quad \Lambda = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_m \end{bmatrix} \tag{8.21}$$

such that the complementarity conditions $s_i\lambda_i$ for $i = 1, 2, \ldots, m$ may be stated compactly as $S\Lambda e = 0$ with $e = \begin{bmatrix} 1 & \ldots & 1 \end{bmatrix}'$. Consequently, the optimality conditions (8.19) may be stated as the systems of equations and inequalities

$$r_L = g - A'\lambda = 0 \tag{8.22a}$$

$$r_s = s - Ax + b = 0 \tag{8.22b}$$

$$r_{s\lambda} = S\Lambda e = 0 \tag{8.22c}$$

$$(s, \lambda) \geq 0 \tag{8.22d}$$

3.2 Newton Step

Given an iterate (x, λ, s) satisfying $(s, \lambda) > 0$, (8.22) may be solved by a sequence of Newton steps with modified search directions and step lengths.

The Newton direction is computed as the solution of

$$\begin{bmatrix} 0 & -A' & 0 \\ -A & 0 & I \\ 0 & S & \Lambda \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \lambda \\ \Delta s \end{bmatrix} = - \begin{bmatrix} r_L \\ r_s \\ r_{s\lambda} \end{bmatrix} \tag{8.23}$$

The structure of this linear system of equations may be utilized to solve it efficiently. Note that the second block row of (8.23) yields

$$\Delta s = -r_s + A\Delta x \tag{8.24}$$

Using that $S > 0$ and easily invertible as it is a diagonal matrix with positive entries, the third block row of (8.23) along with (8.24) yield

$$\begin{aligned} \Delta \lambda &= -S^{-1} \left(r_{s\lambda} + \Lambda \Delta s \right) \\ &= S^{-1} \left(-r_{s\lambda} + \Lambda r_s \right) - S^{-1} \Lambda A \Delta x \end{aligned} \tag{8.25}$$

Finally, the first block row of (8.23) along with (8.25) yield

$$\begin{aligned} -r_L &= -A' \Delta \lambda \\ &= \left(A' S^{-1} \Lambda A \right) \Delta x - A' S^{-1} \left(-r_{s\lambda} + \Lambda r_s \right) \\ &= \bar{H} \Delta x + \bar{r} \end{aligned} \tag{8.26}$$

in which

$$\bar{H} = A'(S^{-1}\Lambda)A \tag{8.27a}$$

$$\bar{r} = A' \left[S^{-1} (r_{s\lambda} - \Lambda r_s) \right] \tag{8.27b}$$

Consequently, (8.23) may be solved by solution of

$$\bar{H} \Delta x = -\bar{g} = -(r_L + \bar{r}) \tag{8.28}$$

for Δx and subsequent computation of Δs by (8.24) and $\Delta \lambda$ by (8.25). The next iterate in the Newton iteration is computed as

$$\begin{bmatrix} x \\ \lambda \\ s \end{bmatrix} \leftarrow \begin{bmatrix} x \\ \lambda \\ s \end{bmatrix} + \alpha \begin{bmatrix} \Delta x \\ \Delta \lambda \\ \Delta s \end{bmatrix} \tag{8.29}$$

with the step length $\alpha \in (0, \alpha_{\max}) \cap (0, 1]$ selected such that $(\lambda, s) > 0$, i.e. with the maximum step length computed as

$$s + \alpha_{\max} \Delta s \geq (1 - \tau) s \tag{8.30a}$$

$$\lambda + \alpha_{\max} \Delta \lambda \geq (1 - \tau) \lambda \tag{8.30b}$$

with $\tau \to 1$ as the iterate approaches the solution.

3.3 Predictor-Corrector Interior-Point Algorithm

To avoid being restricted to small step lengths as is often the case when (8.22) is solved directly, the complementarity conditions are modified such that the pairs $s_i\lambda_i$ decrease at the same rate for all i. Instead of solving (8.22c), we solve

$$r_{s\lambda} = S\Lambda e - \sigma\mu e = 0 \qquad \mu = \frac{s'\lambda}{m} = \frac{\sum_{i=1}^{m} s_i\lambda_i}{m} \tag{8.31}$$

for some value of $\sigma \in (0, 1]$. In Mehrotra's predictor-corrector algorithm, σ is selected adaptively based on the duality gap reduction for an affine step ($\sigma = 0$). This affine step may also be used to predict $r_{s\lambda}$ and introduce a correction such that the step direction is computed by solution of (8.23) with

$$r_{s\lambda} = S\Lambda e + \Delta S\Delta\Lambda e - \sigma\mu e \tag{8.32}$$

ΔS and $\Delta\Lambda$ are the step directions computed in the affine step ($\sigma = 0$).

3.4 Primal-Dual Interior-Point Algorithm

Algorithm 2 specifies the steps in this procedure for solution of (8.16).

The main computational efforts in Algorithm 2 are 1) formation of the matrix $\bar{H} = A'DA$ with $D = S^{-1}\Lambda$ being a diagonal matrix with positive entries on the diagonal and 2) Cholesky factorization of $\bar{H}$.

4 Interior-Point Algorithm for MPC-LP

The constrained optimal control problem (8.2) (which is equivalent with (8.14)) gives the following A-matrix and b-vector in the standard LP formulation (8.16)

$$A = \begin{bmatrix} I & 0 & 0 \\ -I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \\ \Psi & 0 & 0 \\ -\Psi & 0 & 0 \\ \Psi & I & 0 \\ -\Psi & I & 0 \\ \Gamma_u & 0 & I \\ -\Gamma_u & 0 & I \end{bmatrix} \quad b = \begin{bmatrix} U_{\min} \\ -U_{\max} \\ 0 \\ 0 \\ \Delta U_{\min} + I_0 u_{-1} \\ -(\Delta U_{\max} + I_0 u_{-1}) \\ I_0 u_{-1} \\ -I_0 u_{-1} \\ b \\ -b \end{bmatrix} \tag{8.33}$$

This A-matrix is highly structured. Therefore, we may specialize the steps in Algorithm 2 that involves operations with the A-matrix. The following theorems state the computational simplifications used in Algorithm 2 when A has the structure in (8.33). For notational convenience we use Matlab like notation in some of the theorems.

Lemma 1 (Hessian matrix, $\bar{H}$, in MPC-LP)**.** *Let A have the structure in (8.33). Let $D = diag([d_1; d_2; \ldots; d_{10}]) = \Lambda^{-1}S$ be a diagonal matrix with positive entries and let $D_i = diag(d_i)$ for $i = 1, 2, \ldots, 10$ be sub-matrices of D corresponding to the division of A in (8.33).*

Algorithm 2 Interior-point algorithm for (8.16).

Require: $(g \in \mathbb{R}^n, A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m)$
Residuals and Duality Gap:
$r_L = g - A'\lambda$, $r_s = s - Ax + b$, $r_{s\lambda} = S\Lambda e$
Duality gap: $\mu = \frac{s'\lambda}{m}$
while $Not\_Converged$ **do**
Compute $\bar{H} = A'(S^{-1}\Lambda)A$
Cholesky factorization: $\bar{H} = \bar{L}\bar{L}'$
Affine Predictor Step:
Compute $\bar{r} = A'(S^{-1}(r_{s\lambda} - \Lambda r_s))$, $-\bar{g} = -(r_L + \bar{r})$
Solve: $\bar{L}\bar{L}'\Delta x = -\bar{g}$
$\Delta s = -r_s + A\Delta x$
$\Delta\lambda = -S^{-1}(r_{s\lambda} + \Lambda\Delta s)$
Determine the maximum affine step length

$$\lambda + \alpha_{\max}\Delta\lambda \geq 0 \quad s + \alpha_{\max}\Delta s \geq 0$$

Select affine step length: $\alpha \in (0, \alpha_{\max}]$
Compute affine duality gap: $\mu_a = \frac{(\lambda + \alpha\Delta\lambda)'(s + \alpha\Delta s)}{m}$
Centering parameter: $\sigma = \left(\frac{\mu_a}{\mu}\right)^3$
Center Corrector Step:
Modified complementarity:

$$r_{s\lambda} \leftarrow r_{s\lambda} + \Delta S\Delta\Lambda e - \sigma\mu e$$

Compute $\bar{r} = A'(S^{-1}(r_{s\lambda} - \Lambda r_s))$, $-\bar{g} = -(r_L + \bar{r})$
Solve: $\bar{L}\bar{L}'\Delta x = -\bar{g}$
$\Delta s = -r_s + A\Delta x$
$\Delta\lambda = -S^{-1}(r_{s\lambda} + \Lambda\Delta s)$
Determine the maximum affine step length

$$\lambda + \alpha_{\max}\Delta\lambda \geq 0 \quad s + \alpha_{\max}\Delta s \geq 0$$

Select affine step length: $\alpha \in (0, \alpha_{\max}]$
Step: $x \leftarrow x + \alpha\Delta x$, $\lambda \leftarrow \lambda + \alpha\Delta\lambda$, $s \leftarrow s + \alpha\Delta s$
Residuals and Duality Gap:
$r_L = g - A'\lambda$, $r_s = s - Ax + b$, $r_{s\lambda} = S\Lambda e$
Duality gap: $\mu = \frac{s'\lambda}{m}$
end while
Return: (x, λ)

Then

$$\bar{H} = A'DA = \begin{bmatrix} \bar{H}_{11} & \bar{H}_{12} \\ \bar{H}_{21} & \bar{D} \end{bmatrix} \tag{8.34}$$

with the sub-matrices

$$\bar{H}_{11} = \bar{D}_1 + \Psi'\bar{D}_2\Psi + \Gamma_u'\bar{D}_3\Gamma_u \tag{8.35a}$$
$$\bar{H}_{12} = \bar{H}_{21}' = \begin{bmatrix} \Psi'\bar{D}_4 & \Gamma_u'\bar{D}_5 \end{bmatrix} \tag{8.35b}$$
$$\bar{D} = \begin{bmatrix} \bar{D}_6 & \\ & \bar{D}_7 \end{bmatrix} \tag{8.35c}$$

and

$$\begin{aligned} \bar{D}_1 &= D_1 + D_2 & \bar{D}_2 &= D_5 + D_6 + D_7 + D_8 \\ \bar{D}_3 &= D_9 + D_{10} & \bar{D}_4 &= D_7 - D_8 \\ \bar{D}_5 &= D_9 - D_{10} & \bar{D}_6 &= D_3 + D_7 + D_8 \\ \bar{D}_7 &= D_4 + D_9 + D_{10} \end{aligned}$$

Proof. Follows by straightforward matrix multiplications using A in (8.35). □

Theorem 5 (Cholesky Factorization in MPC-LP)**.** *Solution of $\bar{H}x = b$ corresponds to solution of the system*

$$\begin{bmatrix} \bar{H}_{11} & \bar{H}_{12} \\ \bar{H}_{21} & \bar{D} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \tag{8.36}$$

This system may be factorized by

1. *Compute $\hat{D}_2 = \bar{D}_2 - \bar{D}_4\bar{D}_6^{-1}\bar{D}_4$*
2. *Compute $\hat{D}_3 = \bar{D}_3 - \bar{D}_5\bar{D}_7^{-1}\bar{D}_5$*
3. *Compute $\hat{H}_{11} = \bar{D}_1 + \Psi'\hat{D}_2\Psi + \Gamma_u'\hat{D}_3\Gamma_u$*
4. *Cholesky factorize $\hat{H}_{11}$: $\hat{H}_{11} = \hat{L}\hat{L}'$*

and solved by

1. *Solve $\hat{L}\hat{L}'x_1 = b_1 - \bar{D}^{-1}b_2$ for x_1 by back substitutions*
2. *Compute $x_2 = \bar{D}^{-1}\left(\bar{b}_2 - \begin{bmatrix} \bar{D}_4(\Psi x_1) \\ \bar{D}_5(\Gamma_u x_1) \end{bmatrix}\right)$*

Proof. The results are obtained by application of the Schur complement to (8.36) and the matrix definitions (8.35). □

Theorem 6 (Matrix-vector operations in MPC-LP)**.**

1. *Let A have the structure in (8.33). Let $x = [U; V; W]$. Then $Ax = [U; -U; V; W; z_1; -z_1; z_3; z_4; z_5; z_6]$ with $z_1 = \Psi U$, $z_2 = \Gamma_u U$, $z_3 = z1 + V$, $z_4 = -z_1 + V$, $z_5 = z_2 + W$, and $z_6 = -z_2 + W$.*

2. *Let A have the structure in (8.33).*
   Let $v = [v_1;\, v_2;\, \ldots;\, v_{10}]$. Then

$$A'v = \begin{bmatrix} \bar{v}_1 + \Psi'\bar{v}_2 + \Gamma_u'\bar{v}_3 \\ v_3 + v_7 + v_8 \\ v_4 + v_9 + v_{10} \end{bmatrix} \tag{8.37}$$

   with $\bar{v}_1 = v_1 - v_2$, $\bar{v}_2 = v_5 - v_6 + v_7 - v_8$, $\bar{v}_3 = v_9 - v_{10}$

Proof. Follows by straightforward matrix-vector manipulations. □

Theorem 7 (Operations with Ψ)**.** *For illustration consider Ψ for $N = 4$ and let $D = diag([d_1;\, d_2;\, d_3;\, d_4])$ be a diagonal matrix with $D_i = diag(d_i)$ for $i \in \{1, 2, 3, 4\}$ also being diagonal matrices. Then*

$$\Psi' D \Psi = \begin{bmatrix} D_1 + D_2 & -D_2 & 0 & 0 \\ -D_2 & D_2 + D_3 & -D_3 & 0 \\ 0 & -D_3 & D_3 + D_4 & -D_4 \\ 0 & 0 & -D_4 & D_4 \end{bmatrix} \tag{8.38}$$

Let $x = [x_1;\, x_2;\, x_3;\, x_4]$ then

$$\Psi x = \begin{bmatrix} x_1;\, x_2 - x_1;\, x_3 - x_2;\, x_4 - x_3 \end{bmatrix} \tag{8.39a}$$

$$\Psi' x = \begin{bmatrix} x_1 - x_2;\, x_2 - x_3;\, x_3 - x_4;\, x_4 \end{bmatrix} \tag{8.39b}$$

Proof. Straightforward matrix-matrix and matrix-vector operations with Ψ. □

The operations $\Gamma_u' D \Gamma_u$, $\Gamma_u U$, and $\Gamma_u' Z$ for some diagonal matrix D, some vector U, and some vector Z are implemented using straightforward matrix operations even though Γ_u is structured. In the current Matlab implementation $\Gamma_u' D \Gamma_u$ is the computational bottleneck. $\Gamma_u' D \Gamma_u$ is implemented using that D is a diagonal matrix but without using the structure of Γ_u.

Remark* 1** (Operations with Γ_u):*** *Γ_u is a matrix of the impulse response parameters, $\{H_{u,k} = CA^{k-1}B\}_{k=1}^{N}$. $Z = \Gamma_u U$ transfers a set of inputs $\{u_k\}_{k=0}^{N-1}$ to a set of outputs $\{z_k\}_{k=1}^{N}$ for the system ($k = 0, 1, \ldots, N-1$)*

$$x_{k+1} = Ax_k + Bu_k \qquad x_0 = 0 \tag{8.40a}$$

$$z_{k+1} = Cx_{k+1} \tag{8.40b}$$

Similarly. $U = \Gamma_u' Z$ corresponds to

$$\bar{x}_{k-1} = A'\bar{x}_k + C'z_k \qquad \bar{x}_N = 0 \tag{8.41a}$$

$$u_{k-1} = B'\bar{x}_{k-1} \tag{8.41b}$$

going backwards with $k = N, N-1, \ldots, 1$.

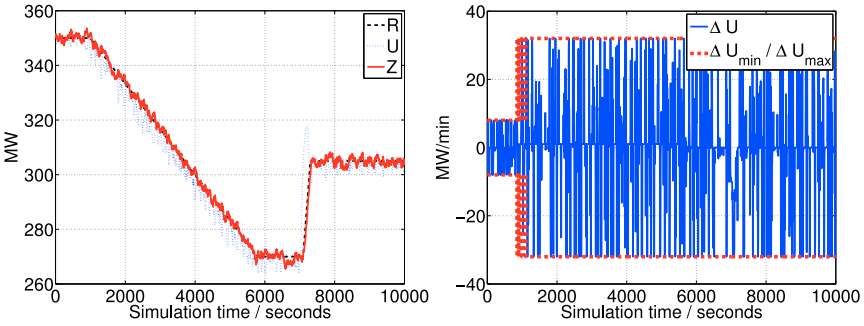


Figure 8.1: Input and output for the benchmark case.

5 Results

Using the boiler load effectuator of a power plant [Edlund et al., 2009], we test the developed interior-point MPC-LP algorithm (Algorithm 2) utilizing the structure of A in (8.33) for solution of the constrained optimal control problem (8.2). We compare our MPC-LP algorithm to the solution of (8.2) using `linprog` in Matlab's optimization toolbox and an active set LP solver applied to (8.14).

The boiler effectuator is a SISO system and we use a control horizon of $N = 50$. The number of decision variables (U, V, W) in the LP to be solved is $3N = 150$. The sampling time is $T_s = 5s$ and we run the test problem in closed-loop for 2000 samples. Figure 8.1 illustrates the benchmark case for which we have compared the tree different LP solvers. All three LP solvers give the same result indicating that our solver is implemented correct. The case study and controller tuning is chosen such that some of the constraints are usually active as indicated to the right in Figure 8.1.

As can be read from Figure 8.2, the runtime of our MPC-LP (IPmpc) solver is about one order of magnitude faster than both the active set LP solver and linprog. Furthermore, the variance of the CPU-time is much smaller for MPC-LP than for both linprog and the active set LP solver. In real-time applications it is desirable to have a predictable computing time. MPC-LP and linprog are implemented in Matlab. The active set LP solver is a highly efficient LP solver for general LPs in the form (8.14) that is implemented in Fortran and equipped with a mex-interface.

Figure 8.3 illustrates the CPU-time for the three different LP solvers for (8.2) as function of the number of decision variables in the LP (8.14) and (8.16), respectively. The interior-point MPC-LP algorithm (IPmpc) is significantly faster than the other algorithms, typically more than one order of magnitude faster.

6 Conclusion

We have developed computationally efficient primal-dual interior point algorithms for constrained optimal control problems that have linear dynamics, input constraints, rate-of-movement constraints, and objective functions containing linear stage costs and l_1-norm deviation penalties on the set-point and the input movement. MPC for dynamic

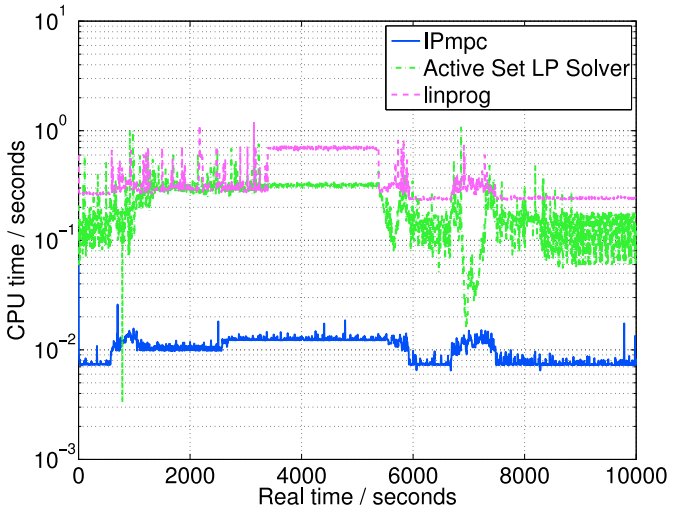


Figure 8.2: CPU times for the different LP algorithms solving (8.2).

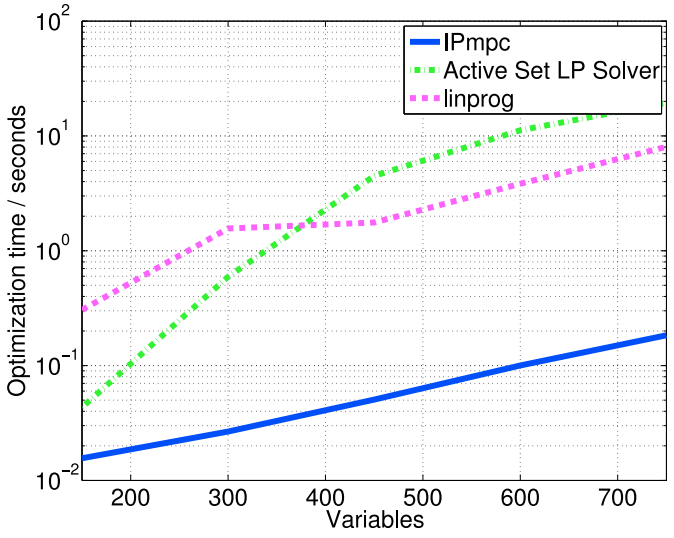


Figure 8.3: CPU-time as function of the number of decision variables in the LP corresponding to (8.2).

regulation, coordination and optimization of power generation solves such problems in real-time repeatedly. Fast and robust optimization algorithms are important in these applications. The new primal-dual interior point algorithm is implemented in Matlab and its performance is compared to an active set based LP solver and linprog in Matlab's optimization toolbox. Simulations demonstrate that the new algorithm is more than one magnitude faster than the other LP algorithms.

References

J. Allwright and G. Papavasiliou. On Linear Programming and Robust Model-Predictive Control Using Impulse-Responses. *Systems & Control Letters*, 18(2):159–164, 1992.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004. ISBN 0-521-83378-7.

J. Czyzyk, S. Mehrotra, M. Wagner, and S. J. Wright. PCx: An Interior-Point Code for Linear Programming. *Optimization Methods and Software*, 11(1):397–430, 1999.

K. Edlund, J. D. Bendtsen, S. Børresen, and T. Mølbak. Introducing Model Predictive Control for Improving Power Plant Portfolio Performance. *Proceedings of the 17th IFAC World Congress, Seoul, South Korea*, pages 6986–6991, 2008.

K. Edlund, T. Mølbak, and J. D. Bendtsen. Simple Models for Model-based Portfolio Load Balancing Controller Synthesis. *Proceedings of IFAC Symposium on power plants and power systems control 2009, Tampera, Finland*, 2009.

J. M. Maciejowski. *Predictive Control with Constraints*. Pearson Education Limited, 2002. ISBN 0-201-39823-0.

S. Mehrotra. On the Implementation of a Primal-Dual Interior Point Method. *SIAM J. Optimization*, 2(4):575–601, 1992.

A. M. Morshedi, C. R. Cutler, and T. A. Skrovanek. Optimal Solution of Dynamic Matrix Control with Linear Programming Techniques (LDMC). In *Proceedings of the 1985 American Control Conference, Boston, USA*, pages 199–208, 1985.

J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, 2nd edition, 2006. ISBN 0-387-98793-2.

C. V. Rao and J. B. Rawlings. Linear Programming and Model Predictive Control. *Journal of Process Control*, 10(2–3):283–289, 2000.

C. V. Rao, S. J. Wright, and J. B. Rawlings. Application of Interior-Point Methods to Model Predictive Control. *Journal of Optimization Theory and Applications*, 99(3): 723–757, 1998.

J. B. Rawlings and R. Amrit. Optimizing Process Economic Performance Using Model Predictive Control. In L. Magni, D. M. Raimondo, and F. Allgöwer, editors, *Nonlinear Model Predictive Control*, pages 119–138. Springer, 2009. ISBN 3-764-36297-9.

L. Vandenberghe, S. Boyd, and M. Nouralishadi. Robust Linear Programming and Optimal Control. *Proceedings of 15th IFAC World Congress, Barcelona, Spain*, 2002.

S. J. Wright. *Primal-Dual Interior-Point Methods*. SIAM, Philadelphia, 1997. ISBN 0-898-71382-4.

Y. Zhang. Solving Large-Scale Linear Programs by Interior-Point Methods under the Matlab Environment. *Optimization Methods and Software*, 10(1):1–31, 1998.

Paper E

A Dantzig-Wolfe MPC Algorithm for Power Plant Portfolio Control

Kristian Edlund & John Bagterp Jørgensen

This paper was submitted to:
International Journal of Control

Abstract

In this paper we treat the dynamic coordination of a set of constrained, dynamically independent systems seeking to achieve one common goal. This is motivated by a portfolio of multiple power generators which has to follow one production reference for the whole portfolio. A Model Predictive Controller with linear and ℓ_1-norm terms is proposed to coordinate the units. The underlying optimization problem of the proposed controller can be stated as a linear program with a block-angular constraint matrix. Dantzig-Wolfe decomposition can be successfully applied to decompose this kind of problems into several subproblems.

Simulations show that the computation time only rises linearly with the number of units in the problem rather than cubically as is the case when the problem is solved in a centralized manner with an active-set solver.

1 Introduction

Model Predictive Control (MPC) has successfully been applied in the process industries for more than thirty years [Qin and Badgwell, 2003; Lu, 2003; Froisy, 2006]. Since the description of the first MPCs based on convolution models [Richalet et al., 1978; Cutler and Ramaker, 1980], several generations of industrial and academic MPCs have lead to the formulation of state space based MPCs [Muske and Rawlings, 1993; Maciejowski, 2002; Rawlings and Mayne, 2009] and in a sense unified the theory of Model Predictive Control (MPC), Generalized Predictive Control (GPC), and Linear Quadratic Gaussian (LQG) regulation [Clarke et al., 1987; Bitmead et al., 1990; Åström and Wittenmark, 1990; Mayne et al., 2000; Jørgensen, 2005]. Linear MPC requires repeated online solution of constrained linear or quadratic optimization problems. Therefore, the computational speed and robustness of the optimization algorithms has limited the type of applications that can be controlled by MPC. MPC was originally developed for the process industries with relative slow dynamics and a low number of inputs and outputs (say less than 50). As MPC is developed for mechatronic applications with very fast dynamics, low state order models, and typically less than 3 inputs and outputs, new ways of implementing and solving the constrained optimization problem constituting the MPC have been developed [Bemporad et al., 2002; Diehl et al., 2005; Mattingley and Boyd, 2009]. Both process control and mechatronic applications use one centralized MPC to control the system. This is possible because of the low number of inputs and outputs as well as the relative low number of states in the model.

In this paper we consider a set of dynamically independent systems that must cooperate to meet a common objective. This is motivated by a portfolio of multiple power generators which has to follow one production reference for the whole portfolio. The multiple power generator problem is large scale with a very large number of independent systems. In addition, the system has fast dynamics and must be controlled with a sample rate of approximately 5 seconds. Consequently, such systems cannot be controlled by a centralized approach with one Model Predictive Controller using existing optimization algorithms due to the large size and fast dynamics. In this paper, we develop a computationally efficient algorithm for ℓ_1-norm Model Predictive Control of independent linear dynamic systems that must cooperate to meet a common objective. We utilize the special structure of the linear program representing the MPC and use Dantzig-Wolfe decomposi-

tion to obtain an efficient optimization algorithm that can solve this class of systems very fast.

In practice the computational complexity of the optimization problem scales cubically with the problem size [Ferris et al., 2007]. Therefore methods to avoid solving the centralized optimization problem must be pursued to successfully apply MPC on large-scale systems. Lower computationally complexity can be achieved through decomposition of the control problem, i.e. splitting the problem into multiple smaller problems which are easier to solve. Much research is going into decomposition of control, an extensive review of the area is found in [Scattolini, 2009].

The strategy followed in this paper is to exploit the structure of the underlying optimization problem for decomposition. The decomposition yields a two level hierarchical control structure. The structure consists of a higher level supervisor that coordinates the lower level of independent local controllers. The requirement for the method presented in this paper is that the underlying optimization problem is a linear problem with a block-angular constraint structure, i.e. a block diagonal matrix with a set of coupling constraints involving all variables. The requirement can be achieved by formulating the objective function with ℓ_1-norm and linear terms.

A system meeting these requirements can be efficiently decomposed with two methods; Lagrange relaxation [Beasley, 1993] and Dantzig-Wolfe decomposition [Dantzig and Wolfe, 1960]. Both methods significantly decrease the computation time for optimization of linear systems with a block-angular constraint structure. Both methods use an iterative scheme to solve the optimization problem. In each iteration the lagrange multipliers attached to the coupling constraints of the problem, are assumed constant. Thereby the optimization problem is reduced to a block diagonal structure and can be treated as P independent problems. The difference in the two methods is how to find the Lagrange multipliers. Lagrange relaxation computes the multipliers through heuristic methods, while Dantzig-Wolfe decomposition finds the multipliers by solving an optimization problem. [Gunnerud et al., 2009] showed that the computation time using Lagrange relaxation is very sensitive to changes in the problem, and even minor changes might result in a doubling of the computation time.

The contribution of this paper is to use Dantzig-Wolfe decomposition for computation of the dynamic calculations of model predictive control and thereby reduce the computational complexity compared to the centralized solution without relying on heuristic methods.

[Negenborn et al., 2008] and [Rantzer, 2009] applied Lagrange relaxation for decomposing problems in a model predictive control context. [Gunnerud et al., 2009] used Dantzig-Wolfe decomposition for control and planning purposes on a longer time scale, while [Cheng et al., 2008] used Dantzig-Wolfe for target calculation in a distributed model predictive controller.

There are multiple other approaches to achieve decomposition of controllers. Two of them are a decentralized control scheme, meaning that there is no communication among local controllers [Acar, 1995; Magni and Scattolini, 2006; Raimondo et al., 2007], and distributed control where local controllers have communication among each other [Jia and Krogh, 2001; Dunbar, 2007; Zhang and Li, 2007; Venkat et al., 2008]. All of these papers treat systems which are dynamically coupled rather than independent. [Keviczky et al., 2008; Dunbar and Murray, 2006] treats dynamically independent systems that needs coordination to achieve a common goal. The approach in both [Keviczky et al., 2008] and

[Dunbar and Murray, 2006] have been to communicate with the neighbours and thus limit the knowledge and size of each local controller. This approach works for systems where each subsystem has a small amount of neighbours.

This paper is organized as follows. In Section 2 we define the control problem transform it into a block-angular structured linear program. Section 3 describes Dantzig-Wolfe decomposition algorithm used for solving linear programs. Section 4 describes the power plant application motivating the development in this paper, followed by the results and comparison of the algorithm compared to the centralized solution in Section 5. Conclusions are provided in Section 6.

2 The problem

In this paper, we consider a set of dynamically independent systems that needs coordination in order to achieve one common goal.

Each independent system is described by the linear time invariant discrete state space formulation

$$\mathbf{x}_{i,k+1} = \mathbf{A}_i\mathbf{x}_{i,k} + \mathbf{B}_i\mathbf{u}_{i,k} + \mathbf{E}_i\mathbf{d}_{i,k} \tag{9.1a}$$

$$\mathbf{y}_{i,k} = \mathbf{C}_i\mathbf{x}_{i,k} \tag{9.1b}$$

In our application this system represents a number of power plants including the basic control systems. $\mathbf{d}_{i,k}$ represents the production plan, $\mathbf{u}_{i,k}$ is the correction to the production plan, and $\mathbf{y}_{i,k}$ is the produced power. We consider P power plants such that $i \in \{1, 2, \ldots, P\}$. k is the discrete time index.

The control problem that we want to investigate is

$$\begin{aligned}
\min \quad & \phi && && (9.2a)\\
s.t. \quad & \mathbf{x}_{i,k+1} = \mathbf{A}_i\mathbf{x}_{i,k} + \mathbf{B}_i\mathbf{u}_{i,k} + \mathbf{E}_i\mathbf{d}_{i,k}, && i = 1, 2, ..., P && (9.2b)\\
& \mathbf{y}_{i,k} = \mathbf{C}_i\mathbf{x}_{i,k}, && i = 1, 2, ..., P && (9.2c)\\
& \underline{\mathbf{u}_{i,k}} \leq \mathbf{u}_{i,k} \leq \overline{\mathbf{u}_{i,k}}, && i = 1, 2, ..., P && (9.2d)\\
& \underline{\Delta\mathbf{u}_{i,k}} \leq \Delta\mathbf{u}_{i,k} \leq \overline{\Delta\mathbf{u}_{i,k}}, && i = 1, 2, ..., P && (9.2e)
\end{aligned}$$

with

$$\begin{aligned}
\phi = & \sum_{k=1}^{N} \left\| \sum_{i=1}^{P} \mathbf{y}_{i,k} - \mathbf{r}_k \right\|_{1,\mathbf{Q}} + \\
& \sum_{i=1}^{P} \left[\sum_{k=1}^{N} \mathbf{q}'_{u,i,k}\mathbf{y}_{i,k} + \sum_{k=1}^{N} \|\mathbf{y}_{i,k} - \mathbf{r}_{i,k}\|_{1,\mathbf{q}_{e,i,k}} + \sum_{k=0}^{N-1} \|\Delta\mathbf{u}_{i,k}\|_{1,\mathbf{S}_i} \right]
\end{aligned} \tag{9.3}$$

ϕ is the performance function we want to minimize to find optimum. Symbols with a bar beneath, e.g., $\underline{u}$ means the lower bound, while $\overline{u}$ denotes the upper bound. $\Delta\mathbf{u}_{i,k} = \mathbf{u}_{i,k} - \mathbf{u}_{i,k-1}$. [Edlund et al., 2008] motivate and describe the details of this objective function. The first term penalizes deviations of the total power production, $\mathbf{y}_k = \sum_{i=1}^{P} \mathbf{y}_{i,k}$, from the sold power, $\mathbf{r}_k$. The second term is an economic term measuring the cost of producing power on each plant. On each plant, the third term penalizes deviation from

the scheduled power production, $\mathbf{r}_{i,k}$. The fourth term smooth the solution by penalizing rapid movements in the manipulated variables.

Model Predictive Control is often expressed using either ℓ_2-penalty functions without economic terms [Muske and Rawlings, 1993], ℓ_1-penalty functions without economic terms [Maciejowski, 2002], or using economic terms only [Rawlings and Amrit, 2009]. The objective function (9.3) contains both ℓ_1-norm penalty functions and a linear term related to cost. Therefore, the mathematical program defining the controller is a linear program. [Rao and Rawlings, 2000] demonstrate the Model Predictive Controllers containing ℓ_1-norm penalty functions may give rise to either dead-beat or idle control. While theoretically (9.2) result in this behavior, numerous simulations demonstrate that the controller performs well and provides the desired portfolio control [Edlund et al., 2008].

Control problems with ℓ_1-norm penalties such as in (9.2) can be expressed as a linear program [Edlund et al., 2009b]. This implies that (9.2) can be expressed as the block-angular structured linear program

$$\min_{\mathbf{z}} \qquad \phi = \mathbf{c}_1'\mathbf{z}_1 + \mathbf{c}_2'\mathbf{z}_2 + ... + \mathbf{c}_P'\mathbf{z}_P \tag{9.4a}$$

$$s.t. \quad \begin{bmatrix} \mathbf{F}_1 & \mathbf{F}_2 & \dots & \mathbf{F}_P \\ \mathbf{G}_1 & & & \\ & \mathbf{G}_2 & & \\ & & \ddots & \\ & & & \mathbf{G}_P \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_P \end{bmatrix} \geq \begin{bmatrix} \mathbf{g} \\ \mathbf{h}_1 \\ \mathbf{h}_2 \\ \vdots \\ \mathbf{h}_P \end{bmatrix}. \tag{9.4b}$$

With $\mathbf{z} = [\mathbf{z}_1, \mathbf{z}_2, \dots \mathbf{z}_P]$. ϕ is the functional which needs to be minimized in order to find optimum, $\mathbf{z}$ are the free variables, $\mathbf{c}_i$ are weight factors, weighing the importance of the corresponding $\mathbf{z}_i$. The constraint matrix has a block-angular structure where the block diagonal elements come from the optimization problem related to the individual power plants and the coupling constraint comes from the rewriting the first term of (9.3) into a linear problem. $\mathbf{F}_i$ is unit i's contribution to the coupling constraint. $\mathbf{G}_i$ originates from the individual subproblems constraints (9.2b)-(9.2e) and the last part of the performance function. $\mathbf{g}$ and $\mathbf{h}_i$ are the affine part of the constraints.

This paper is based on a specific performance function given in (9.3). However, the method is applicable for all control problems where the underlying control problem can be stated as (9.4).

3 Dantzig-Wolfe decomposition

The Dantzig-Wolfe decomposition is an algorithm that is very efficient for solution of linear programs with block-angular structure [Dantzig and Wolfe, 1960; Lasdon, 2002; Dantzig and Thapa, 2002]. Dantzig-Wolfe decomposition breaks the linear program (9.4) into P independent subproblems and a Master Problem (MP). The Master Problem coordinates the subproblems as illustrated in Figure 9.1. The Master Problem sends the price, π, of the shared resource to each of the subproblems. Using this price, π, each of the P subproblems computes their optimal solution. This interchange of information continues until convergence.

Throughout the description of the decomposition it is assumed that the feasible region of each subproblem is closed and bounded. This is no limitation as the decomposition can

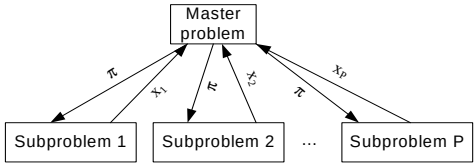


Figure 9.1: Concept of the Dantzig-Wolfe decomposition. The big problem is split into several smaller problems communicates with a coordinator to reach the optimum.

be applied to unbounded subproblems as well [Dantzig and Wolfe, 1960].

Dantzig-Wolfe decomposition builds on the theorem of convex combinations

Theorem 8. *Let $\mathcal{Z} = \{\mathbf{z} \mid \mathbf{G}\mathbf{z} \geq \mathbf{h}\}$ be nonempty, closed and bounded, i.e. a polytope. The extreme points of $\mathcal{Z}$ are denoted $\mathbf{v}^j$ with $j \in \{1, 2, ..., M\}$.*

Then any point $\mathbf{z}$ in the polytopic set $\mathcal{Z}$ can be written as a convex combination of extreme points

$$\mathbf{z} = \sum_{j=1}^{M} \lambda_j \mathbf{v}^j \tag{9.5a}$$

$$s.t. \quad \lambda_j \geq 0, \qquad j = 1, 2, ..., M \tag{9.5b}$$

$$\sum_{j=1}^{M} \lambda_j = 1 \tag{9.5c}$$

Proof. See [Dantzig and Thapa, 2002] □

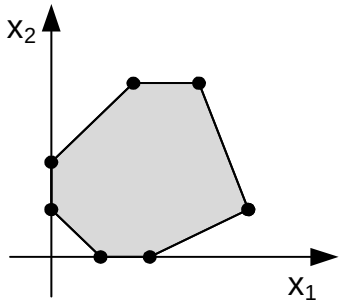


Figure 9.2: Illustration of the theorem of convex combinations. Any point in the feasible area (shaded region) can be expressed as a convex combination of the extreme points (black dots).

The theorem of convex combination says: Any point in a polytope can be expressed as a convex combination of the extreme points. This is illustrated in Figure 9.2. Any point in the shaded region can be expressed as a convex combination of the extreme points marked by a black dot.

The linear program (9.4) can be expressed as

$$\min_{z} \quad \phi = \sum_{i=1}^{P} \mathbf{c}_i' \mathbf{z}_i \tag{9.6a}$$

$$s.t. \quad \sum_{i=1}^{P} \mathbf{F}_i \mathbf{z}_i \geq \mathbf{g} \tag{9.6b}$$

$$\mathbf{G}_i \mathbf{z}_i \geq \mathbf{h}_i, \qquad i = 1, 2, ..., P \tag{9.6c}$$

Using Theorem 8, the polytope defined by (9.6c); $\mathcal{Z}_i = \{\mathbf{z}_i | \ \mathbf{G}_i \mathbf{z}_i \geq \mathbf{h}_i\}$ can be expressed as

$$\mathbf{z}_i = \sum_{j=1}^{M_i} \lambda_{ij} \mathbf{v}_i^j \tag{9.7a}$$

$$\sum_{j=1}^{M_i} \lambda_{ij} = 1 \tag{9.7b}$$

$$\lambda_{ij} \geq 0 \qquad j = 1, 2, \ldots, M_i \tag{9.7c}$$

M_i is the number of extreme points of $\mathcal{Z}_i$. Substituting (9.7) into (9.6) yields

$$\min_{\lambda} \quad \phi = \sum_{i=1}^{P} \sum_{j=1}^{M_i} f_{ij} \lambda_{ij} \tag{9.8a}$$

$$s.t. \quad \sum_{i=1}^{P} \sum_{j=1}^{M_i} \mathbf{p}_{ij} \lambda_{ij} \geq \mathbf{g} \tag{9.8b}$$

$$\sum_{j=1}^{M_i} \lambda_{ij} = 1, \qquad i = 1, 2, ..., P \tag{9.8c}$$

$$\lambda_{ij} \geq 0, \qquad i = 1, 2, ..., P;\ j = 1, 2, \ldots, M_i \tag{9.8d}$$

f_{ij} and p_{ij} are defined as

$$f_{ij} = \mathbf{c}_i' \mathbf{v}_i^j \tag{9.9a}$$

$$\mathbf{p}_{ij} = \mathbf{F}_i \mathbf{v}_i^j \tag{9.9b}$$

The Master Problem (9.8) is equivalent to the block-angular linear program (9.4). The Master Problem has fewer constraints than the original problem, but the number of variables in the Master Problem is larger due to the large number of extreme points.

The Lagrangian associated to the Master Problem (9.8) is

$$\begin{aligned}\mathcal{L}(\lambda_{ij},\pi,\rho_i,\kappa_{ij}) = &\sum_{i=1}^{P}\sum_{j=1}^{M_i} f_{ij}\lambda_{ij} \\ &- \pi'\left(\sum_{i=1}^{P}\sum_{j=1}^{M_i}\mathbf{p}_{ij}\lambda_{ij} - \mathbf{g}\right) \\ &- \sum_{i=1}^{P}\rho_i(\sum_{j=1}^{M_i}\lambda_{ij} - 1) \\ &- \sum_{i=1}^{P}\sum_{j=1}^{M_i}\kappa_{ij}\lambda_{ij}\end{aligned} \tag{9.10}$$

With π being the Lagrange multiplier of the coupling constraint (9.8b), ρ is the Lagrange multiplier for (9.8c) and κ being Lagrange multiplier for the positivity constraint (9.8d).

Consequently, the necessary and sufficient optimality conditions for the Master Problem (9.8) are

$$\nabla_{\lambda_{ij}}\mathcal{L} = f_{ij} - \mathbf{p}'_{ij}\pi - \rho_i - \kappa_{ij} = 0 \qquad i = 1,2,\ldots,P;\ j = 1,2,\ldots,M_i \tag{9.11a}$$

$$\sum_{i=1}^{P}\sum_{j=1}^{M_i}\mathbf{p}_{ij}\lambda_{ij} - \mathbf{g} \geq 0 \quad \perp \quad \pi \geq 0 \tag{9.11b}$$

$$\sum_{j=1}^{M_i}\lambda_{ij} - 1 = 0 \qquad i = 1,2,\ldots,P \tag{9.11c}$$

$$\lambda_{ij} \geq 0 \quad \perp \quad \kappa_{ij} \geq 0 \qquad i = 1,2,\ldots,P;\ j = 1,2,\ldots,M_i \tag{9.11d}$$

We notice that the conditions (9.11a) and (9.11d) imply

$$\kappa_{ij} = f_{ij} - \mathbf{p}'_{ij}\pi - \rho_i = \left[\mathbf{c}_i - \mathbf{F}'_i\pi\right]'\mathbf{v}_i^j - \rho_i \geq 0 \quad i = 1,2,\ldots,P;\ j = 1,2,\ldots,M_i \tag{9.12}$$

such that the Karush-Kuhn-Tucker conditions (KKT-conditions) for (9.8) may be stated as

$$\sum_{i=1}^{P}\sum_{j=1}^{M_i}\mathbf{p}_{ij}\lambda_{ij} - \mathbf{g} \geq 0 \quad \perp \quad \pi \geq 0 \tag{9.13a}$$

$$\sum_{j=1}^{M_i}\lambda_{ij} - 1 = 0 \qquad i = 1,2,\ldots,P \tag{9.13b}$$

$$\lambda_{ij} \geq 0 \quad \perp \quad \kappa_{ij} = \left[\mathbf{c}_i - \mathbf{F}'_i\pi\right]'\mathbf{v}_i^j - \rho_i \geq 0 \quad i = 1,2,\ldots,P;\ j = 1,2,\ldots,M_i \tag{9.13c}$$

3.1 The Dantzig-Wolfe algorithm

Large problems are characterized by a very large number of extreme points. Therefore, generation of all the extreme points in the Master Problem (9.8) can in itself be a very

challenging computational problem. The Dantzig-Wolfe algorithm overcomes this challenge using delayed column generation, i.e. it generates the extreme points for the Simplex basis algorithm when needed.

The master problem with a reduced number of extreme points is called the Reduced Master Problem (RMP) and can be expressed as

$$\min_{\lambda} \quad \phi = \sum_{i=1}^{P}\sum_{j=1}^{l} f_{ij}\lambda_{ij} \tag{9.14a}$$

$$s.t. \quad \sum_{i=1}^{P}\sum_{j=1}^{l} \mathbf{p}_{ij}\lambda_{ij} \geq \mathbf{g} \tag{9.14b}$$

$$\sum_{j=1}^{l} \lambda_{ij} = 1, \qquad i = 1, 2, ..., P \tag{9.14c}$$

$$\lambda_{ij} \geq 0, \qquad i = 1, 2, ..., P;\ j = 1, 2, \ldots, l \tag{9.14d}$$

in which $l \leq M_i$ for all $i \in \{1, 2, \ldots, P\}$. Obviously, the Reduced Master Problem can be regarded as the Master Problem with $\lambda_{i,j} = 0$ for $j = l+1, l+2, \ldots, M_i$ and all $i \in \{1, 2, \ldots, P\}$.

Initially, a feasible extreme point to the Master Problem (9.8) is needed. We may generate such a point using techniques similar to Phase I in the simplex algorithm for a standard linear program. In the Dantzig-Wolfe algorithm one may use the procedure described in this section to a Phase I LP. In the following, we assume that a feasible extreme point has been computed. We can use this feasible extreme point to form a Reduced Master Problem with $l = 1$. We denote the solution to the Reduced Master Problem (9.14) as λ_{ij}^{RMP} such that a feasible solution to Master Problem (9.8) is

$$\lambda_{ij} = \lambda_{ij}^{RMP} \qquad i = 1, 2, \ldots, P;\ j = 1, 2, \ldots, l \tag{9.15a}$$

$$\lambda_{ij} = 0 \qquad i = 1, 2, \ldots, P; j = l+1, l+2, \ldots, M_i \tag{9.15b}$$

This solution satisfies (9.13a) and (9.13b). To be optimal it also needs to satisfy (9.13c). These conditions are already satisfied for $i = 1, 2, \ldots, P$ and $j = 1, 2, \ldots, l$. We need to verify whether they are satisfied for all $i = 1, 2, \ldots, P$ and $j = l+1, l+2, \ldots, M_i$. This is complicated by the fact that we only know the extreme points, $\mathbf{v}_i^j$ for $i = 1, 2, \ldots, P$ and $j = 1, 2, \ldots, l$.

(9.13c) is satisfied for all $i = 1, 2, \ldots, P$ and $j = 1, 2, \ldots, M_i$ if $\min_i \psi_i - \rho_i \geq 0$ where

$$\psi_i = \min_{\mathbf{v}_i^j} \left[\mathbf{c}_i - \mathbf{F}_i'\pi\right]' \mathbf{v}_i^j \qquad i = 1, 2, \ldots, P \tag{9.16}$$

$\mathbf{v}_i^j$ is an extreme point of the polytope $\mathcal{Z}_i = \{\mathbf{z}_i : \mathbf{G}_i\mathbf{z}_i \geq \mathbf{h}_i\}$. Therefore, using the Simplex Algorithm we may compute the solution of (9.16) as the solution of the linear program

$$\psi_i = \min_{\mathbf{z}_i} \quad \phi = \left[\mathbf{c}_i - \mathbf{F}_i'\pi\right]' \mathbf{z}_i \tag{9.17a}$$

$$s.t. \quad \mathbf{G}_i\mathbf{z}_i \geq \mathbf{h}_i \tag{9.17b}$$

for $i = 1, 2, \ldots, P$. These programs are called subproblems.

If $\psi_i - \rho_i \geq 0$ for all $i = 1, 2, \ldots, P$, the solution generated by the Reduced Master Problem is optimal. We can compute the solution to original problem (9.4) by

$$\mathbf{z}_i^* = \sum_{j=1}^{l} \mathbf{v}_i^j \lambda_{ij} \qquad i = 1, 2, \ldots, P \tag{9.18}$$

If $\psi_i - \rho_i < 0$ for some $i \in \{1, 2, \ldots, P\}$ then the KKT conditions are not satisfied and the solution generated by the Reduced Master Problem is not a solution to the Master Problem. In this case, we augment the Reduced Master Problem with the new extreme points, $\mathbf{v}_i^{l+1}$, obtained by solution of the subproblems (9.17).

The next iteration of the algorithm starts with the solution of the new Reduced Master Problem. The algorithm terminates in a finite number of iterations as there is a finite number of extreme points in a polytope.

Algorithm 3 summarises the Dantzig-Wolfe Algorithm for solution of the block-angular linear program (9.4). The subproblems (9.20) may be solved in parallel. This is advantageous when the number of subproblems, P, is large. In all iterations, the Dantzig-Wolfe Algorithm preserves feasibility of (9.4). In predictive control applications, this implies that stability can be guaranteed under mild condition even if the algorithm is stopped prematurely [Scokaert et al., 1999].

3.2 Computation of an initial feasible vertex

The initial feasible vertex of the Master Problem (9.8) may be computed using a Phase I simplex algorithm. The linear program

$$\min_{\alpha, \{\mathbf{z}_i, \beta_i\}_{i=1}^P} \quad \phi_I = \mathbf{e}'_\alpha \alpha + \sum_{i=1}^{P} \mathbf{e}'_{\beta_i} \beta_i \tag{9.23a}$$

$$s.t. \quad \sum_{i=1}^{P} \mathbf{F}_i \mathbf{z}_i + \mathbf{R}\alpha \geq \mathbf{g} \tag{9.23b}$$

$$\mathbf{H}_i \mathbf{z}_i + \mathbf{S}_i \beta_i \geq \mathbf{h}_i \qquad i = 1, 2, \ldots, P \tag{9.23c}$$

$$\mathbf{0} \leq \alpha \leq |\mathbf{g}| \tag{9.23d}$$

$$\mathbf{0} \leq \beta_i \leq |\mathbf{h}_i| \qquad i = 1, 2, \ldots, P \tag{9.23e}$$

with

$$\mathbf{R}_{ij} = \begin{cases} 1 & i = j \wedge \mathbf{g}_i \geq 0 \\ -1 & i = j \wedge \mathbf{g}_i < 0 \\ 0 & i \neq j \end{cases} \qquad (\mathbf{S}_i)_{p,q} = \begin{cases} 1 & p = q \wedge (\mathbf{h}_i)_p \geq 0 \\ -1 & p = q \wedge (\mathbf{h}_i)_p < 0 \\ 0 & p \neq q \end{cases}$$

may be used to compute a feasible vertex of the block angular linear program (9.4). A feasible vertex of the block angular linear program (9.4) is identical to a feasible vertex of the Master Problem (9.8) as these two linear programs are different representations of the same problem. A feasible vertex to (9.4) and (9.8) exists if the optimal value function of (9.23) is zero, i.e. $\phi_I = 0$ as this implies that a feasible vertex with $\alpha = \mathbf{0}$ and $\{\beta_i = \mathbf{0}\}_{i=1}^P$ exists. Otherwise, (9.4) is infeasible.

Algorithm 3 The Dantzig-Wolfe Algorithm for a Block-Angular LP (9.4).

1: Compute a feasible vertex of the Master Problem (9.8). If no such point exists then stop.
2: $l = 1$, Converged = false.
3: **while** Not Converged **do**
4: Solve the l'th Reduced Master Problem, RMP(l):

$$\min_{\lambda} \quad \phi = \sum_{i=1}^{P} \sum_{j=1}^{l} f_{ij} \lambda_{ij} \tag{9.19a}$$

$$s.t. \quad \sum_{i=1}^{P} \sum_{j=1}^{l} \mathbf{p}_{ij} \lambda_{ij} \geq \mathbf{g} \tag{9.19b}$$

$$\sum_{j=1}^{l} \lambda_{ij} = 1 \qquad i = 1, 2, \ldots, P \tag{9.19c}$$

$$\lambda_{ij} \geq 0 \qquad i = 1, 2, \ldots, P;\ j = 1, 2, \ldots, l \tag{9.19d}$$

and let π be the computed Lagrange multiplier associated to the linking constraint (9.19b). Let ρ_i be the computed Lagrange multiplier associated with (9.19c).
5: Solve all the subproblems ($i \in \{1, 2, \ldots, P\}$)

$$\min_{\mathbf{z}_i} \quad \phi_i = \left[\mathbf{c}_i - \mathbf{F}_i' \pi\right]' \mathbf{z}_i \tag{9.20a}$$

$$s.t. \quad \mathbf{G}_i \mathbf{z}_i \geq \mathbf{h}_i \tag{9.20b}$$

and let $(\psi_i, \mathbf{v}_i^{l+1}) = (\phi_i^*, \mathbf{z}_i^*)$ be the optimal value-minimizer pair.
6: **if** $\psi_i - \rho_i \geq 0\ \forall i \in \{1, 2, \ldots, P\}$ **then**
7: Converged = true. The optimal solution is

$$\mathbf{z}_i^* = \sum_{j=1}^{l} \lambda_{ij} \mathbf{v}_i^j \qquad i = 1, 2, \ldots, P \tag{9.21}$$

8: **else**
9: Compute the coefficients for the new columns in the RMP

$$f_{i,l+1} = \mathbf{c}_i' \mathbf{v}_i^{l+1} \tag{9.22a}$$

$$\mathbf{p}_{i,l+1} = \mathbf{F}_i \mathbf{v}_i^{l+1} \tag{9.22b}$$

10: $l \leftarrow l + 1$
11: **end if**
12: **end while**

It is trivial to find an initial feasible vertex to (9.23), i.e. $\{\alpha = |\mathbf{g}|, \{\mathbf{z}_i = \mathbf{0}, \beta_i = |\mathbf{h}_i|\}_{i=1}^{P}\}$ is a feasible vertex of (9.23). Therefore, the phase I block angular program may be parameterized similar to the Master Problem (9.8) and solved using a Dantzig-Wolfe algorithm similar to Algorithm 3. The subproblems in the phase I part of the Dantzig-Wolfe algorithm are

$$\min_{\alpha} \quad \phi_{I,0} = \left[\mathbf{e}_v - \mathbf{R}'\pi\right]' \alpha \tag{9.24a}$$

$$s.t. \quad \mathbf{0} \leq \alpha \leq |\mathbf{g}| \tag{9.24b}$$

and

$$\min_{\mathbf{z}_i, \beta_i} \quad \phi_{I,i} = (-\mathbf{F}_i'\pi)'\mathbf{z}_i + \mathbf{e}'_{\beta_i}\beta_i \tag{9.25a}$$

$$s.t. \quad \mathbf{G}_i\mathbf{z}_i + \mathbf{S}_i\beta_i \geq \mathbf{h}_i \tag{9.25b}$$

$$\mathbf{0} \leq \beta_i \leq |\mathbf{h}_i| \tag{9.25c}$$

for $i \in \{1, 2, \ldots, P\}$. This implies that a feasible vertex may be computed by solution of a number of relatively small subproblems, (9.24) and (9.25), and solution of a Reduced Master Problem using delayed column generation.

It should be noted that the computation of a feasible vertex of (9.8), i.e. solution of (9.23) by the Dantzig-Wolfe algorithm, is of approximately the same computational complexity as the computation of the optimal solution when a feasible vertex is available. This means that we can utilise the block-angular structure efficiently in the computation of a feasible vertex. It also means that just finding a feasible vertex may be just as expensive as computing the optimal solution. Therefore, if a feasible vertex is readily available, it should be used directly instead of applying a phase I simplex procedure.

Just as several alternatives exists for determining a feasible vertex in linear programming by the simplex algorithm, several alternatives to the phase I procedure in the Dantzig-Wolfe algorithm exists. The alternatives are mainly based on determining a feasible vertex by replacing (9.23) with a ℓ_1-norm or ∞-norm regression problem in the phase I procedure.

4 Application

Section 2 defines the general control problem for P plants with independent dynamics collaborating to achieve a common objective, i.e. (9.2)-(9.3). In this section, we apply this optimization model for control of a power generating portfolio.

[Edlund et al., 2009a] defines the word "effectuator" to describe an entity which *is a process or part of a process in a power system that represents control actions with associated dynamics and actuation costs allowing the power output to be manipulated*. [Edlund et al., 2009a] also derives dynamic models for several types of effectuators.

Figure 9.3 illustrates a power portfolio. A power portfolio consists of multiple different effectuators. An effectuator can be a traditional thermal power plant or a wind farm. In Denmark, thermal power plants are combined heat and power plants (CHP). In the future, even electric vehicles may be used as effectuators.

Each effectuator has its own regulatory control system. Therefore, the manipulated variable, u_i, of an effectuator is its power production reference. The output, y_i, from

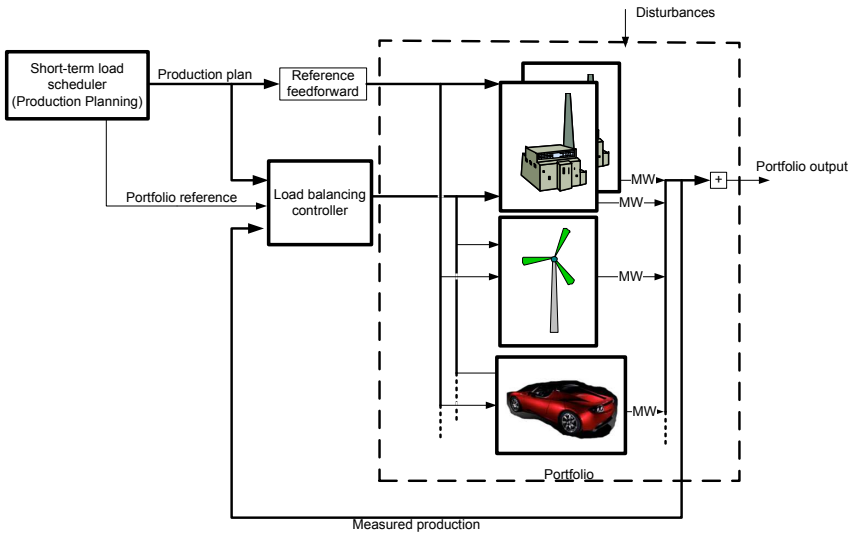


Figure 9.3: Sketch of the system we want to control. Bold lines represent vectors

an effectuator is the produced power by that effectuator. The power portfolio controller consists of a short term load scheduler (STLS) and a load balancing controller. The short term load scheduler makes a static economic optimization of the system and uses the solution to generate references for each effectuator. Due to fault-tolerant considerations, these references are sent directly to the effectuators rather than to the load balancing controller. If the dynamic part of the portfolio control crashes due to hardware error, this architecture implies that the effectuators still have a reference. [Jørgensen et al., 2006] describe the details of this power portfolio controller.

The short term load scheduler provides a 5-minute 24 hour ahead based production plan for each effectuator. However, a power production system is complex: Disturbances and deviations from the production plan does occur. Therefore, a dynamic load balancing controller is added to the system. The load balancing controller seeks to minimize the deviation between sold and actual power production.

4.1 Modelling of an effectuator

In the following, we only consider boiler load effectuators. This choice is made to simplify the presentation. A third order transfer function describes the boiler load effectuator dynamics sufficiently well

$$y_i(s) = \frac{1}{(\tau_i s + 1)^3}(u_i(s) + d_i(s)) \tag{9.26}$$

y_i is the produced power, d_i is the setpoint from the STLS, and u_i is the correction computed by the load balancing controller. Therefore, $u_i + d_i$ is the desired power production setpoint sent to the regulatory system of the effectuator. τ_i is the time constant for the boiler. Different boilers may have different time constants. Besides the dynamic part, the

model includes an upper and lower bound on the output as well as a constraint on how fast power production can be changed. The effectuator has a control system which enforces these constraints. Therefore they are treated as input constraints in the model.

Using the performance function defined by [Edlund et al., 2008], the optimization problem for each effectuator $i \in \{1, 2, \dots P\}$ can be stated as

$$\min \quad \phi_i = \sum_{k=1}^{N} ||y_{i,k} - r_{i,k}||_{1,q_{e,i,k}} + \sum_{k=1}^{N} q'_{u,i,k} y_{i,k} + \sum_{k=0}^{N-1} ||\Delta u_k||_{1,q_{\Delta u,i,k}} \tag{9.27a}$$

$$s.t. \quad \mathbf{x}_{i,k+1} = \mathbf{A}_i \mathbf{x}_{i,k} + \mathbf{B}_i u_{i,k} + \mathbf{E}_i d_{i,k} \qquad k = 0, 1, \dots, N-1 \tag{9.27b}$$

$$y_{i,k} = \mathbf{C}_i \mathbf{x}_{i,k} \qquad k = 1, 2, \dots, N \tag{9.27c}$$

$$\underline{u_{i,k}} \leq u_{i,k} \leq \overline{u_{i,k}} \qquad k = 1, 2, ..., N \tag{9.27d}$$

$$\underline{\Delta u_{i,k}} \leq \Delta u_{i,k} \leq \overline{\Delta u_{i,k}} \qquad k = 1, 2, ..., N \tag{9.27e}$$

The first term of the objective function, $\sum_{k=1}^{N} ||y_{i,k} - r_{i,k}||_{1,q_{e,i,k}}$, is added to penalize deviations from the static optimization determined by the STLS. In the nominal situation the effectuators should adhere to the production plan, since it is made in an optimal fashion utilizing information which is not available for the load balancing controller. $q_{u,i,k}$ is the marginal cost of the effectuator, which is the cost of producing an extra megawatt hour, the marginal cost includes fuel cost and wear on the plant, but does not include static cost such salary to operators which is independent of the power production. The second term, $\sum_{k=1}^{N} q'_{u,i,k} y_k$, is therefore an economic term measuring the cost of producing power on each plant. The third term, $\sum_{k=0}^{N-1} ||\Delta u_k||_{1,q_{\Delta u,i,k}}$ smooth the solution by penalizing rapid movements in the manipulated variables.

The constraints for the optimization problem comes from the linear model and the constraints of the model.

4.2 Portfolio modelling

So far P independent problems have been described, in order for the portfolio to work as a whole some problem or optimization problem for the whole portfolio has to be derived.

The goal of the portfolio is to minimize the overall deviation between the total reference and the total production

$$\min \quad \phi_t = \sum_{k=1}^{N} \|y_{tot,k} - r_k\| \,. \tag{9.28}$$

there are no constraints in the portfolio goal, however there are also no means of actually affecting the output of this optimization goal since $y_{tot,k} = \sum_{i=1}^{P} y_{i,k}$ is the sum of outputs from the effectuators. Since the portfolio uses the information from the effectuators they cannot be solved separately but needs to be combined. (9.2) represents the optimal control problem that combines the portfolio control problem (9.28) with the control problem for each effectuator (9.27). It should be noted that the portfolio reference, r_k in (9.28), is not necessarily equal to the sum of the effectuator references, $r_{i,k}$ in (9.27).

4.3 Finding an initial feasible solution for the master problem

Rewriting (9.28) into a linear program will add an extra set of decision variables to the master problem, called $\mathbf{z}_{tot}$. These variables acts similar to slack variables in the sense that if they are large enough the problem will become feasible. In this case it means that if a feasible solution can be found to all sub problems, a feasible solution to the Master Problem exists.

The task of finding an initial feasible solution to the Master Problem is thereby reduced to finding a feasible solution to all subproblems with $\pi = \mathbf{0}$. Once a solution to all subproblems are found $\mathbf{z}_{tot}$ has to fulfill $z_{tot,k} \geq |\sum_{i=1}^{P} y_{i,k} - r_k|$. Since the right hand side is known, finding a solution for this inequality is trivial and result in an initial feasible solution to the Master Problem.

5 Results

[Edlund et al., 2008] implement (9.2) as a centralized MPC using a sample time of 5 seconds. We also use a sample time of 5 seconds and compare the power portfolio balancing MPC (9.2) implemented using the Dantzig-Wolfe Algorithm described in this paper to a centralized implementation using standard linear programming [Edlund et al., 2008]. We investigate the effect of the prediction horizon, N, as well as the effect of the number of effectuators, P, on the computing time for the Dantzig-Wolfe based MPC and the centralized MPC.

In this section problems of different size will be treated. Each of the Effectuators have $3N$ optimization variables and $8N$ constraint equations. The portfolio adds $2N$ constraints to the optimization problem. Where N is the prediction horizon. So with p effectuators we have a centralized optimization problem with $3Np$ optimization variables and $8Np + 2N$ constraints. And when using Dantzig-Wolfe decomposition we have p optimization problems with $3N$ optimization variables and $8N$ constraints, in addition to the RMP with $2N$ constraints and a variable number of optimization variables, depending on the number of iterations needed.

For finding the solution an active-set LP-solver is used which scales cubically with the number of optimization variables. The solver is used for solving both the centralized problem and the Dantzig-Wolfe decomposition, so the solution times are comparable.

The largest deviation in the elements of the optimal point between the centralized solution and the decomposed is of in the magnitude of 10^{-6}. That is within the expected precision of the algorithm and it is therefore concluded that the two algorithms does converge to the same optimal point.

Figure 9.4 shows the average execution time per sample as a function of the number of effectuators.

The centralized solution scales approximately cubically with the number of effectuators. The extra overhead is created from a mixture of the optimizer using more iterations to converge and Matlab™creating more overhead when the problem size grows. The Dantzig-Wolfe decomposition scales almost linearly with the number of effectuators. Part of the overhead in this algorithm comes from the fact that the RMP grows faster when a large number of subsystems are used since a multi column generation scheme is used. The subproblems in the Dantzig-Wolfe decomposition algorithm can be run in parallel. The figure shows a lower bound time estimation for a parallel version of Dantzig-Wolfe

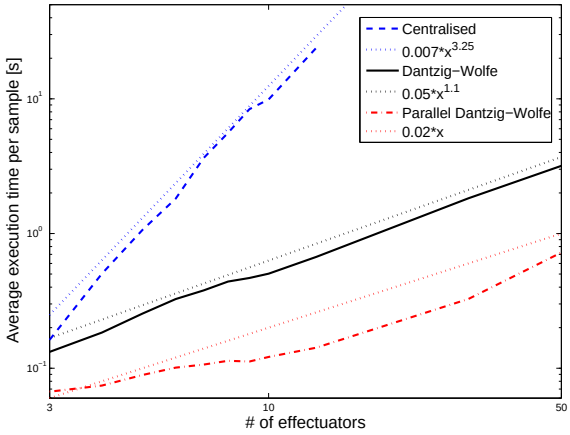


Figure 9.4: Average execution time as a function of the number of effectuators

where the the time for optimization of the subsystems is calculated as the time of the slowest system. In reality there will be some overhead from the communication. The solution time of a parallel real implementation will be somewhere in between the seriel and fully parallel version.

The average execution time as a function of the prediction horizon is shown in Figure 9.5. As expected the execution time still scales cubically with the prediction horizon, since an increased prediction horizon means an increased subproblem size.

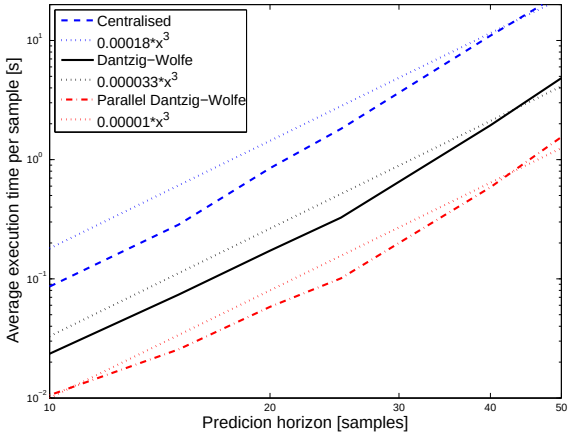


Figure 9.5: Average execution time as a function of the prediction horizon

6 Conclusion

We have treated a power portfolio system consisting of multiple dynamically independent power plants which cooperates to achieve a total production. A Model Predictive Controller with linear and ℓ_1-norm terms has been proposed to coordinate the units. The underlying optimization problem of the proposed controller can be stated as a linear program with a block-angular constraint matrix (9.4). Dantzig-Wolfe decomposition can be successfully applied to decompose this kind of problems, yielding a two layer hierarchical optimization structure. The decomposition has a logical physical interpretation as the lower layer consist of independent local controllers for each of the power plants. The upper level is the supervisor which coordinate the individual units.

The Dantzig-Wolfe decomposition updates and solves the subproblems in an iterative manner, and will converge to the same optimum as the centralized solution within a finite number of iterations. Theoretically the number of iterations may be high, but the simulation results show fast convergence in practical computations. The computation time scales linearly with the number of units in the problem, rather than cubically cubically as would be the case if the problem is solved in a centralized manner with an interior point solver. Dantzig-Wolfe decomposition yields faster solution times than the centralized solution even for small number of effectuators.

The proposed Dantzig-Wolfe algorithm scales cubically with the size of the subproblems in the as the interior point algorithm, i.e. when the number of constraints or the prediction horizon increases. When the the prediction horizon grows there is algorithms which has linear scaling [Rao et al., 1998], this can be exploited when the subproblem size grows. However,it is not investigated in this paper.

References

L. Acar. Boundaries of the Receding Horizon Control for Interconnected Systems. *Journal of Optimization Theory and Applications*, 84(2):251–271, 1995.

K. J. Åström and B. Wittenmark. *Computer-controlled Systems*. Prentice Hall, 3rd edition, 1990. ISBN 0-133-14899-8.

J. E. Beasley. *Modern heuristic techniques for combinatorial problems*. John Wiley & Sons, Inc., New York, NY, USA, 1993. ISBN 0-470-22079-1.

A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. The explicit linear quadratic regulator for constrained systems. *Automatica*, 38(1):3–20, 2002.

R. R. Bitmead, M. Gevers, and V. Wertz. *Adaptive Optimal Control. The Thinking Man's GPC*. Prentice-Hall, New York, NY, USA, 1990. ISBN 0-130-13277-2.

R. Cheng, J. F. Forbes, and W. S. Yip. Dantzig-Wolfe decomposition and plant-wide MPC coordination. *Computers and Chemical Engineering*, 32(7):1507–1522, 2008.

D. W. Clarke, C. Mohtadi, and P. S. Tuffs. Generalized Predictive Control. Part 1: The Basic Algorithm. *Automatica*, 23(2):137–148, 1987.

C. R. Cutler and B. L. Ramaker. Dynamic Matrix Control - A Computer Control Algorithm. In *Joint American Control Conference*, 1980.

G. B. Dantzig and M. N. Thapa. *Linear Programming 2: Theory and Extensions*. Springer-Verlag, 2002. ISBN 0-387-98613-8.

G. B. Dantzig and P. Wolfe. Decomposition Principle for Linear Programs. *Operations research*, 8(1):101–111, 1960.

M. Diehl, H. G. Bock, and J. P. Schlöder. A Real-Time Iteration Scheme for Nonlinear Optimization in Optimal Feedback Control. *SIAM Journal on Control and Optimization*, 43(5):1714–1736, 2005.

W. B. Dunbar. Distributed Receding Horizon Control of Dynamically Coupled Nonlinear Systems. *IEEE Transactions on Automatic Control*, 52(7):1249–1263, 2007.

W. B. Dunbar and R. M. Murray. Distributed Receding horizon control for multi-vehicle formation stabilization. *Automatica*, 42(4):549–558, 2006.

K. Edlund, J. D. Bendtsen, S. Børresen, and T. Mølbak. Introducing Model Predictive Control for Improving Power Plant Portfolio Performance. *Proceedings of the 17th IFAC World Congress, Seoul, South Korea*, pages 6986–6991, 2008.

K. Edlund, T. Mølbak, and J. D. Bendtsen. Simple Models for Model-based Portfolio Load Balancing Controller Synthesis. *Proceedings of IFAC Symposium on power plants and power systems control 2009, Tampera, Finland*, 2009a.

K. Edlund, L. E. Sokoler, and J. B. Jørgensen. A Primal-Dual Interior-Point Linear Programming Algorithm for MPC. *Proceedings of 48th IEEE Conference on Decision and Control, Shanghai, China*, pages 351–356, 2009b.

M. C. Ferris, O. L. Mangasarian, and S. J. Wright. *Linear Programming with Matlab*. SIAM, 2007. ISBN 0-898-71643-8.

J. B. Froisy. Model predictive control Building a bridge between theory and practice. *Computers and Chemical Engineering*, 30(10–12):1426–1435, 2006.

V. Gunnerud, B. Foss, B. Nygreen, R. Vestb, and N. C. Walberg. Dantzig-Wolfe decomposition for real-time optimization - applied to the Troll west oil rim. *Proceedings of International Symposium on Advanced Control of Chemical Processes, Istanbul, Turkey*, 2009.

D. Jia and B. Krogh. Distributed model predictive control. *Proceedings of the 2001 American Control Conference, Arlington, USA*, pages 2767–2772, 2001.

C. Jørgensen, J. H. Mortensen, T. Mølbak, and E. O. Nielsen. Modelbased Fleet Optimization and Master Control of a Power Production System. In *Proceedings of IFAC Symposium on Power Plants and Power Systems Control 2006, Kananaskis, Canada*, 2006.

J. B. Jørgensen. *Moving Horizon Estimation and Control*. Ph.d. thesis, Technical University of Denmark, Kgs. Lyngby, Denmark, 2005.

T. Keviczky, F. Borelli, K. Fregene, D. Godbole, and G. J. Balas. Decentralised Receding Horizon Control and Coordination of Autonomous Vehicle Formations. *IEEE Transactions on Control Systems Technology*, 16(1):19–33, 2008.

L. S. Lasdon. *Optimization theory for large systems*. Dover Publications Inc., 2nd edition, 2002. ISBN 0-486-41999-1.

J. Z. Lu. Challenging control problem and emerging technologies in enterprise optimisation. *Control Engineering Practice*, 11(8):847–858, 2003.

J. M. Maciejowski. *Predictive Control with Constraints*. Pearson Education Limited, 2002. ISBN 0-201-39823-0.

L. Magni and R. Scattolini. Stabilizing decentralized model predictive control of nonlinear systems. *Automatica*, 42(7):1231–1236, 2006.

J. Mattingley and S. Boyd. Automatic Code Generation for Real-Time Convex Optimization. In Y. Eldar and D. P. Palomar, editors, *Convex Optimization in Signal Processing and Communication*. Cambridge University Press, Cambridge, UK, 2009.

D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained Model Predictive Control: Stability and Optimality. *Automatica*, 36(3):789–814, 2000.

K. R. Muske and J. B. Rawlings. Model Predictive Control with Linear Models. *AIChE Journal*, 39(2):262–287, 1993.

R. R. Negenborn, B. De. Schutter, and J. Hellendoorn. Multi-agent model predictive control for transportation networks: Serial versus parallel schemes. *Engineering Applications of Artificial Intelligence*, 21(3):353–366, 2008.

S. J. Qin and T. A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, 11(7):733–764, 2003.

D. M. Raimondo, L. Magni, and R. Scattolini. Decentralized MPC of nonlinear systems: An input-to-state stability approach. *International Journal of Robust and Nonlinear Control*, 17(17):1651–1667, 2007.

A. Rantzer. Dynamic Dual Decomposition for Distributed Control. *Proceedings of the 2009 conference on American Control Conference, St. Louis, USA*, pages 884–888, 2009.

C. V. Rao and J. B. Rawlings. Linear Programming and Model Predictive Control. *Journal of Process Control*, 10(2–3):283–289, 2000.

C. V. Rao, S. J. Wright, and J. B. Rawlings. Application of Interior-Point Methods to Model Predictive Control. *Journal of Optimization Theory and Applications*, 99(3): 723–757, 1998.

J. B. Rawlings and R. Amrit. Optimizing Process Economic Performance Using Model Predictive Control. In L. Magni, D. M. Raimondo, and F. Allgöwer, editors, *Nonlinear Model Predictive Control*, pages 119–138. Springer, 2009. ISBN 3-764-36297-9.

J. B. Rawlings and D. Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, 2009. ISBN 0-975-933770-9.

J. Richalet, A. Rault, J. L. Testud, and J. Papon. Model Predictive Heuristic Control: Applications to Industrial Processes. *Automatica*, 14(5):413–428, 1978.

R. Scattolini. Architectures for distributed and hierarchical Model Predictive Control. *Journal of Process Control*, 19(5):723–731, 2009.

P. O. M. Scokaert, D. Q. Mayne, and J. B. Rawlings. Suboptimal Model Predictive control (Feasibility implies Stability). *IEEE Transactions on Automatic Control*, 44(3):648–654, 1999.

A. N. Venkat, I. A. Hiskens, J. B. Rawlings, and S. J. Wright. Distributed MPC strategies with application to power system automatic generation control. *IEEE Transaction on Control Systems Technology*, 16(6):1192–1206, 2008.

Y. Zhang and S. Li. Networked model predictive control based on neighbourhood optimization for serially connected large-scale processes. *Journal of Process Control*, 17 (1):37–50, 2007.

Paper F

Hierarchical model-based predictive control of power plant portfolio

Kristian Edlund, Jan Dimon Bendtsen & John Bagterp Jørgensen

This paper was submitted to:
Control Engineering Practice

Abstract

This paper describes a design method for developing a flexible hierarchical model-based predictive controller (MPC) for power system portfolio control. The design objectives are flexibility and computational scalability – since the portfolio will grow significantly in the future.

The method yields a model predictive controller with a two layer hierarchy and clearly defined interfaces. The same hierarchical structure is achieved on the underlying optimization problem utilising Dantzig-Wolfe decomposition. Decomposition yields improved computational efficiency and better scalability compared to centralised methods.

Through simulations on a real scenario the new controller shows improvements in ability to track reference production and economic performance.

1 Introduction

With the recent (and ongoing) liberalisation of the energy market [Ringel, 2003], increasing fuel prices, and increasing political pressure toward the introduction of more sustainable energy into the market [UCTE, 2007; Danish Ministry of Transport and Energy, 2005; United Nations, 1998], dynamic control of power plants is becoming highly important. More than ever, power companies must be able to adapt their production to uncontrollable fluctuations in consumer demands as well as in the availability of production resources, e.g., wind power, at short notice [UCTE, 2007].

Historically, static optimisation of load distribution among power production units, so-called *unit commitment*, has been the norm [Padhy, 2004; Salam, 2007]. Unit commitment refers to determining the combination of available generating units and scheduling their respective outputs to satisfy the forecast demand with the minimum total production cost under the operating constraints enforced by the system under a power company's jurisdiction (its *portfolio*) for a specified period of time – typically from 24 hours up to a week. The optimisation problem is of high dimension and combinatorial in nature, and can thus be difficult to solve in practice. Results using Heuristic methods [Johnson et al., 1971; Viana et al., 2001], Mixed Integer Programming [Dillon et al., 1978], Dynamic Programming [Ayuob and Patton, 1971] and Lagrangian Relaxation [Aoki et al., 1987; Shahidehpour and Tong, 1992], have been reported in literature.

Once a solution to the unit comment problem, i.e., a static schedule, has been found, the load plans are distributed to the generating units. Each unit is then responsible for following its load plan, and must handle disturbances etc. locally, implying the necessity of local power plant controllers, wind farm controllers etc.

However, with the aforementioned increasing impact of short-term fluctuations in the supply and demand, dynamic effects at the system level will become increasingly inconvenient to deal with for individual generating units. Yet, system-wide *dynamic portfolio control* is a fairly new concept in the field of power production. So far, to the best of the authors' knowledge, no results have been reported except [Jørgensen et al., 2006; Edlund et al., 2009a, 2008].

Furthermore, an additional difficulty that will have to be faced in tomorrow's 'smart grids' is the addition of many more power plants of various types, with different dynamics – e.g., decentralised bio-mass fired thermal units, solar farms and so forth – which means that scalability of the control system is set to become an important issue.

This paper presents a novel, object-oriented design for such a dynamic portfolio controller, which is able to handle dynamic disturbances at the system level as well as the non-static configuration of generating units, i.e., the fact that not all units are active at all times. It is based on model-based predictive control (see e.g. [Rossiter, 2003] and/or [Rawlings and Mayne, 2009] for a comprehensive review) and utilises a decomposed solution scheme tailored specifically to the problem at hand to solve the optimisation problem (see [Edlund and Jørgensen, nd] for further details). Tests based on actual operation data from the Danish power grid indicate that the proposed controller is able to improve the load following capabilities of the system compared to existing solutions.

The objective of the new controller is to minimise deviations between sold and actual production. When designing the controller two objectives have been in focus.

Scalability The future of the power system will require the controller to be able to coordinate more units, therefore the method must be scalable in terms of computational complexity.

Flexibility To create a controller which is flexible, such that addition of new units and maintenance of existing ones is possible. This means designing a modular structure with good encapsulation of information and clear communication interfaces between modules.

The outline of the rest of the paper is as follows. In Section 2 an overview of the Danish power system is given, including a brief account of the system services the producers must provide. For comparison purposes, Section 2.3 briefly explains the existing portfolio controller. Next, Section 3 presents the proposed control design method and Section 4 uses the design method for designing a controller for the current portfolio. Section 5 present the comparison of the control performances of the simulations, while Section 6 sums up the contributions of this work.

2 System description

To start with the broad perspective, the system is a part of the ENTSO-E, which is the electrical grid covering the mainland of Europe, from Portugal in the west to Romania in the east; within this grid, balance between consumption and production must be maintained at all times. Roughly speaking, if the consumption is larger than the production, energy will be pulled out of the system, making the generators slow down from the usual 50Hz and thus a drop in the system frequency can be observed.

In order to maintain the overall balance between production and consumption ENTSO-E is split into several regions governed by a Transmission System Operator (TSO) who is responsible for matching production with consumption and import/export from the region.

The area treated in this paper is the west Danish area, which is connected synchronously to Germany and asynchronously to Norway and Sweden. The major production units are shown in Figure 10.1.

In order to keep balance between production and consumption, a hierarchical scheme based on time horizons is used. Balance between the production and consumption is currently maintained by changing the production, but in some cases consumption might be

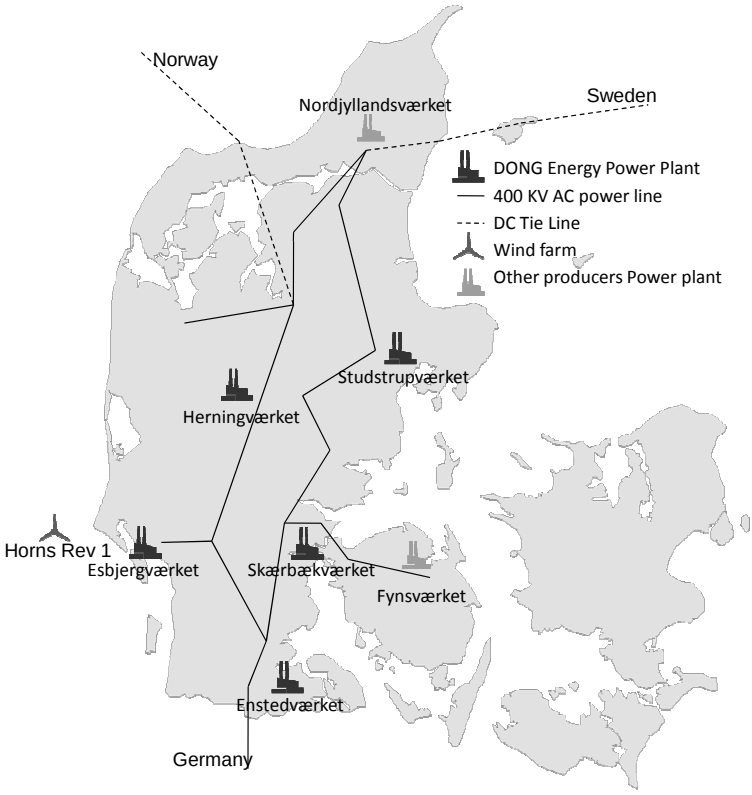


Figure 10.1: Within the west Danish area there are 7 sites containing large power plants comprising 9 boiler units in total with an electrical production capacity ranging from 80 MW to 650 MW; the most common size is around 400 MW. There are two major producers in the area; DONG Energy is the largest and operates a total of 6 units in the area.

changed as well to achieve the goal. To this end, we introduce the term *effectuator*, which encompasses all power producing and power consuming units capable of participating in load balancing control, as follows:

Definition 5. An effectuator is a process or part of a process in a power system that represents control actions with associated dynamics and actuation costs allowing the power output to be manipulated.

2.1 Energy Market and Production Planning

The first effort to balance production and consumption happens when the energy is sold on the energy market, which in the Danish case is Nord Pool [Nord Pool, 2010]. An auction determines the energy price throughout the area for each hour of the following day. The results of the auction yields an amount of energy to be produced each hour. As depicted in Figure 10.2, the sold production is used by the short term load scheduler (STLS)

together with weather forecasts, district heating demand forecasts and constraints such as minimum amount of biomass fuel, to solve a unit commitment problem and compute a 5-minute based schedule for all production units in the portfolio that DONG Energy operates, 24 hours ahead. A more detailed description of the STLS can be found in [Jørgensen et al., 2006].

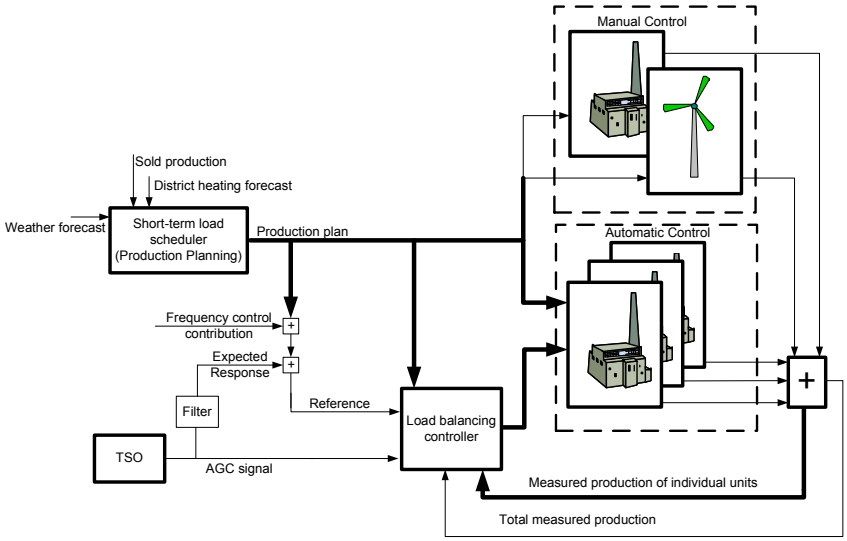


Figure 10.2: Diagram of the interconnection of the system. The bold lines show vectors of signals. The portfolio can be divided into two groups. A manual control which the load balancing controller cannot give corrections to, and an automatic control group which the load balancing controller can affect.

2.2 Reserves

Even though the market gives a good estimate of the demand for the following day, there will be deviations during the day due to disturbances, inaccurate predictions, weather, etc. Therefore, three levels of control have been established to balance production and consumption. In order to execute the control it is required that a certain production capacity is kept in reserve for that purpose. On the shortest time scale is the primary reserve, which is used to avoid system collapse. This is then followed up by slower reserves to bring the system back to the nominal state. The time scale for activation is shown in Figure 10.3.

When the system frequency deviates from 50 Hz, this reserve is to be activated proportional to the system frequency deviation. The reserve has to be activated within 30 seconds after a deviation occurs. Details about the reserve can be found in [ENTSO-E, 2010]. In case of large frequency deviations the primary reserves will be activated throughout the whole European grid.

Secondary reserves are used to replace the primary reserves and help restore the system frequency when they are activated. Each control area, including West Denmark, has secondary reserves. The control area which hosts an imbalance should seek to activate

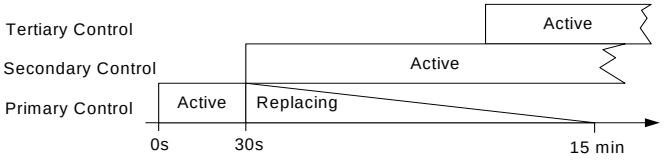


Figure 10.3: Timescale for reserve activation

secondary reserves in order to reject the disturbance. That means that if an area creates a frequency deviation, all areas seek to stabilise the system with the primary reserves, but the area responsible for the imbalance has to bring the system back to nominal behaviour by activating secondary reserves. The secondary reserves can in many cases be activated before a frequency deviation occurs, however. The secondary reserves are activated by the TSO providing an activation signal (AGC Signal on Figure 10.2) to the power producing company. The power company is then expected to deliver a filtered response to the activation signal.

The last reserves that may be activated to stabilise the system frequency are the tertiary reserves. These reserves must be activated within 15 minutes from being ordered. They are activated by an operator at the TSO by taking contact to the operator at the central control room for the energy generation companies. The additional ordering of energy, will most often be added into the STLS, which will then generate and broadcast a new production plan to the units.

2.3 Current controller

The current load balancing controller structure is described in [Edlund et al., 2009a] and is an adaptation of an automatic generation control system from [Wood and Wollenberg, 1996]. It consists of a set of parallel PI-controllers, where the gains of each PI-controller can be changed to accommodate the changing load scenario and constraints.

The mechanism for determining the individual gains is in [Wood and Wollenberg, 1996] proposed to be a steady state optimisation. However, due to the conditions in the West Danish area where the boiler units are not used for base load, but rather changing load very frequently, the optimisation approach has been deemed infeasible. Instead the gains are determined by a logic based mechanism, where each unit is prioritised by the operator for both negative and positive corrections. The logic then utilise the boiler unit with highest priority first, and after usage return all boilers to the production plan.

Besides the main control loop, there is a lot of logic in the controller for handling bumpless transfer between automatic and manual control, and other features in an attempt to make the controller as close to optimal as possible. The result is a huge control structure with many cross couplings.

The problem with the current controller is the complexity of the cross couplings, which means that modifying one part of the controller often affects other parts of the controller in a way that the designer cannot predict. Thus, while the performance of the controller is quite adequate for the existing system, the current structure is not suited for portfolios that change structure over time. Furthermore, the complexity of the logics makes any form of rigorous stability or performance analysis virtually impossible. As a

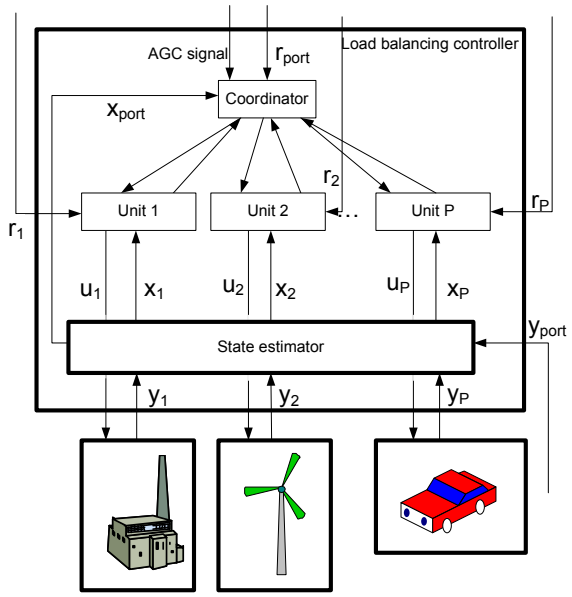


Figure 10.4: Sketch of the modular structure of the load balancing controller. Communication with the individual effectuator is handled by the independent subsystems, and portfolio communication is handled on the upper layer of the hierarchy. r_i is the reference to effectuator $i \in \{1, 2, \ldots, P\}$, x_i is the state estimate, y_i is the measured output, and u_i is the controller correction. For the portfolio there is a reference r_{port}, state estimate x_{port} and a total measured production y_{port}. The references come from the production planning.

consequence, a novel, modular control scheme has been developed.

3 Proposed controller structure

The structure of the proposed controller is a two layer hierarchical structure as shown in Figure 10.4. All parts pertaining to the individual effectuators in the controller are placed in the lower layer separated from one another, allowing them to be modified, removed or adding new ones without affecting the other units. Above is a coordination layer coordinating the individual units to achieve the portfolio goal of minimising deviations.

Model Predictive Control (MPC) has been chosen as the controller scheme, since the system is a constrained MIMO system where knowledge of the future references are available.

The design framework relies on a set of assumptions:

- The effectuators can be modelled as independent of each other, such that a change in one effectuator does not directly affect another effectuator.
- The effectuators can be modelled as a linear dynamic model with affine constraints. The investigated models in [Edlund et al., 2009b] can all with minor modifications

be modelled with the structure shown in Figure 10.5. However, other kinds of linear

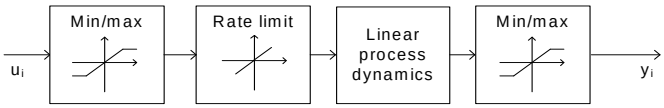


Figure 10.5: General structure of the effectuators

input, output and state constraints fits into the modelling framework as well.

- The underlying optimisation problem in the MPC can be stated as a linear program, which means the corresponding objective function must consist of linear and ℓ_1-norm terms.

Each object in the lower layer of the hierarchy contains a constrained linear model and an objective function for the optimal operation of the effectuator which together form a constrained linear programming problem. Furthermore it contains all communication with the physical unit. The information that has to be sent to the upper layer is how the output of the effectuator will affect the portfolio output, meaning a prediction of the power production/consumption of the unit.

The upper layer contains a constrained linear model of the portfolio excluding the individually modelled effectuators, as well as an objective function of the optimal operation of the portfolio. The upper layer also handles communication with surrounding systems, for instance obtaining the portfolio reference (the load schedule).

3.1 Solving the optimisation problem

The hierarchical structure encapsulates the information pertaining to each unit. However, one challenge persists: MPC relies on solving an optimisation problem at each sample. This is a challenge of the MPC framework, since solving the optimisation problem usually grows cubically with the problem size. Therefore one of the design challenges has been to create an optimisation problem which can be encapsulated in the same hierarchical structure as well as being scalable.

For solving the optimisation problem a Dantzig-Wolfe decomposition approach has been taken [Dantzig and Wolfe, 1960; Dantzig and Thapa, 2002]. The decomposition technique has been adapted to the MPC context in [Edlund and Jørgensen, nd], where details of the algorithm are also described.

Dantzig-Wolfe decomposition can only be applied to linear problems. The performance function for the whole problem is assumed to be chosen as a mixture of linear and ℓ_1-norm terms, which can be rewritten into a linear program suitable for Dantzig-Wolfe decomposition.

An important consequence of this forced choice of performance function and constraints is the solution, i.e., the point where the performance function attains its extremum, must either be at an extreme point of the feasible set, or the solution of an unconstrained problem.

When the optimisation problem is composed from the effectuator optimsation problems and the portfolio optimisation problem it can be rewritten into a linear program with the structure

$$\min_{\mathbf{z}} \quad \phi = \mathbf{c}_1^T\mathbf{z}_1 + \mathbf{c}_2^T\mathbf{z}_2 + ... + \mathbf{c}_P^T\mathbf{z}_P \tag{10.1a}$$

$$s.t. \quad \begin{bmatrix} \mathbf{F}_1 & \mathbf{F}_2 & \dots & \mathbf{F}_P \\ \mathbf{G}_1 & & & \\ & \mathbf{G}_2 & & \\ & & \ddots & \\ & & & \mathbf{G}_P \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_P \end{bmatrix} \geq \begin{bmatrix} \mathbf{g} \\ \mathbf{h}_1 \\ \mathbf{h}_2 \\ \vdots \\ \mathbf{h}_P \end{bmatrix}. \tag{10.1b}$$

with $\mathbf{z} = [\mathbf{z}_1, \mathbf{z}_2, \dots \mathbf{z}_P] \in \mathbb{R}^n$, $\mathbf{z}_i \in \mathbb{R}^{n_i}$, $\phi \in \mathbb{R}$, $\mathbf{F}_i \in \mathbb{R}^{m \times n_i}$, $\mathbf{G}_i \in \mathbb{R}^{p_i \times n_i}$, $\mathbf{g} \in \mathbb{R}^m$ and $\mathbf{h} \in \mathbb{R}^{p_i}$. ϕ is the functional which needs to be minimised in order to find optimum, $\mathbf{z}_i$ are the free variables, $\mathbf{c}_i$ are weight factors, weighing the importance of the corresponding $\mathbf{z}_i$. The constraint matrix has a block-angular structure where the block diagonal elements come from the effectuator optimisation problem and the coupling constraint comes from the portfolio linking the problem together. $\mathbf{F}_i$ is unit i's contribution to the coupling constraint. $\mathbf{G}_i$ originates from the individual effectuators optimisation problem. $\mathbf{g}$ and $\mathbf{h}_i$ are the affine part of the constraints. Ignoring the coupling constraints the program consist of P independent problems

$$\min_{\mathbf{z}_i} \quad \phi = \mathbf{c}_i^T\mathbf{z}_i \tag{10.2a}$$

$$s.t. \quad \mathbf{G}_i\mathbf{z}_i \geq \mathbf{h}_i \tag{10.2b}$$

Dantzig-Wolfe decomposition builds on the theorem of convex combinations

Theorem 9. *Let $\mathcal{Z} = \{\mathbf{z} \in \mathbb{R}^n \mid \mathbf{G}\mathbf{z} \geq \mathbf{h}\}$ with $\mathbf{G} \in \mathbb{R}^{m \times n}$ and $\mathbf{h} \in \mathbb{R}^m$ be nonempty, closed and bounded, i.e. a polytope. The extreme points of $\mathcal{Z}$ are denoted $\mathbf{v}^j$ with $j \in \{1, 2, ..., M\}$.*

Then any point $\mathbf{z}$ in the polytopic set $\mathcal{Z}$ can be written as a convex combination of extreme points

$$\mathbf{z} = \sum_{j=1}^{M} \lambda_j \mathbf{v}^j \tag{10.3a}$$

$$s.t. \quad \lambda_j \geq 0, \qquad j = 1, 2, ..., M \tag{10.3b}$$

$$\sum_{j=1}^{M} \lambda_j = 1 \tag{10.3c}$$

Proof. See [Dantzig and Thapa, 2002] □

Using the theorem on (10.2) and substituting it into (10.1) yields

$$\min_{\lambda} \quad \phi = \sum_{i=1}^{P} \sum_{j=1}^{M_i} f_{ij} \lambda_{ij} \tag{10.4a}$$

$$s.t. \quad \sum_{i=1}^{P} \sum_{j=1}^{M_i} \mathbf{p}_{ij} \lambda_{ij} \geq \mathbf{g} \tag{10.4b}$$

$$\sum_{j=1}^{M_i} \lambda_{ij} = 1, \qquad i = 1, 2, ..., P \tag{10.4c}$$

$$\lambda_{ij} \geq 0, \qquad i = 1, 2, ..., P;\ j = 1, 2, \ldots, M_i \tag{10.4d}$$

With M_i being the number of extreme points of subproblem i. f_{ij} and $\mathbf{p}_{ij}$ are defined as

$$f_{ij} = \mathbf{c}_i^T \mathbf{v}_i^j \tag{10.5a}$$

$$\mathbf{p}_{ij} = \mathbf{F}_i \mathbf{v}_i^j \tag{10.5b}$$

Equation (10.4) is denoted the Master Problem. The idea is to only generate the extreme points needed for the optimisation instead of generating all extreme points which can be even more computationally complex due to the size of the problem. Assuming an initial feasible solution is available for (10.4), a Reduced Master Problem can be set up and expanded through iteration with more extreme points. At iteration l the Reduced Master Problem is defined as

$$\min_{\lambda} \quad \phi = \sum_{i=1}^{P} \sum_{j=1}^{l} f_{ij} \lambda_{ij} \tag{10.6a}$$

$$s.t. \quad \sum_{i=1}^{P} \sum_{j=1}^{l} \mathbf{p}_{ij} \lambda_{ij} \geq \mathbf{g} \tag{10.6b}$$

$$\sum_{j=1}^{l} \lambda_{ij} = 1, \qquad i = 1, 2, ..., P \tag{10.6c}$$

$$\lambda_{ij} \geq 0, \qquad i = 1, 2, ..., P;\ j = 1, 2, \ldots, l \tag{10.6d}$$

in which $l \leq M_i$ for all $i \in \{1, 2, \ldots, P\}$. Obviously, the Reduced Master Problem can be regarded as the Master Problem with $\lambda_{i,j} = 0$ for $j = l + 1, \ldots, M_i$ and all $i \in \{1, 2, \ldots, P\}$.

Solving the Reduced Master Problem yields a Lagrange multiplier, π, for the coupling constraint (10.6b). This can be interpreted as a 'price' for a shared resource, in this case the portfolio deviation. New extreme points are generated by solving subproblems defined as

$$\min_{\mathbf{z}_i} \quad \phi = \left[\mathbf{c}_i - \mathbf{F}_i^T \pi\right]^T \mathbf{z}_i \tag{10.7a}$$

$$s.t. \quad \mathbf{G}_i \mathbf{z}_i \geq \mathbf{h}_i \tag{10.7b}$$

for $i \in \{1, 2, \ldots, P\}$. These originate from (10.2), but the objective function is updated with $-\mathbf{F}_i^T \pi$ where π is given by the Reduced Master Problem in order to generate different extreme points based on the updated price. $\mathbf{F}_i$ is the effect the effectuator will have on the portfolio output.

The algorithm will then iterate over these steps until convergence is reached.

When using a Dantzig-Wolfe decomposition, the Master Problem and the subproblems are defined using the exact same structure as shown in Figure 10.4. The Reduced Master Problem (10.6) is solved in the upper layer, and the subproblems (10.7) are solved in the lower layer of the hierarchy.

The Dantzig-Wolfe algorithm scales almost linearly as a function of the number of subproblems, rather than cubically when solving one centralised problem.

Currently a standard Kalman filter is used for state estimation; it communicates the states to each subproblem and the Master Problem. The current solution means that the modelled units which are not in control need to send an output prediction at the beginning of each sample for use in the Master Problem. The communication between upper and lower layer is shown in Figure 10.6.

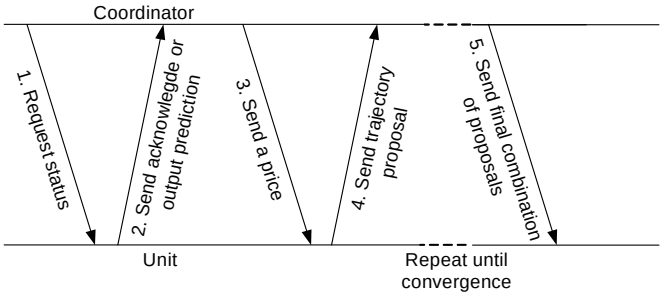


Figure 10.6: Communication timeline between coordinator and each unit during each sample

The developed controller has an object oriented structure with a clear interface between the layers and a clear communication scheme. The controller structure can be described as a UML diagram as in Figure 10.7. As long as the implementations of the effectuators adhere to the defined interface, the implementations can be chosen freely without having to change the framework . The interface is defined by the communication needs of the Dantzig-Wolfe decomposition.

If the information of one unit needs to be updated, it is easy to shut down that part of the controller, update it and set it back into control without having to shut down the entire controller. If the coordination layer needs maintenance the controller will clearly loose its ability to minimise deviation, but the communication with the actual units can be maintained and thus the current input can be maintained instead of ramping down to zero input.

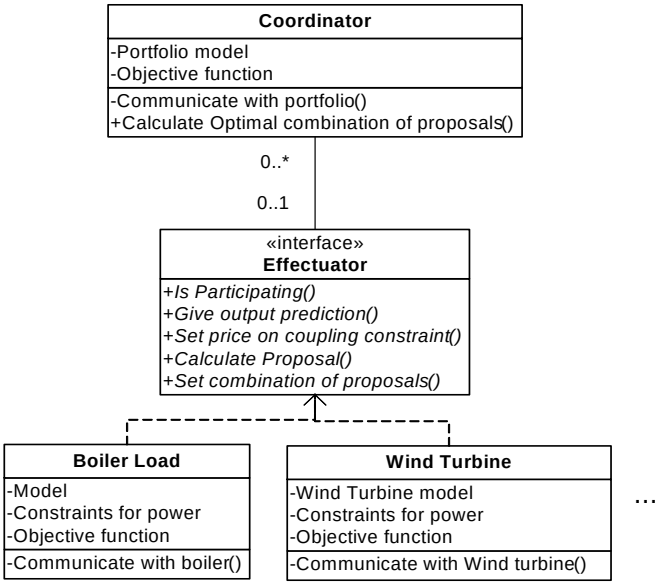


Figure 10.7: UML diagram of the controller structure. The defined interface allows for a flexible implementation of the specific effectuators.

4 Specific controller implementation

In the current system, only boiler load units are available for control purposes, and the specific implementation in this paper is limited to include those; however other effectuators can be included in a straightforward manner.

As outlined above, the individual boilers can be modeled separately, as the actions in one boiler does not affect the others. They are only coupled through the objective to follow the overall portfolio reference and activating secondary resources. A constrained linear model for each boiler is derived in the following, along with a performance function for each.

4.1 Boiler load units

In the current controller there are between 0 and 6 power plant units in control. These will all be modeled in a similar fashion. A simple model of the boiler has been derived in [Edlund et al., 2009b], but in order to fit it into the linear control scheme developed here, some assumptions must be made. The modeling concept is shown in Figure 10.8. The model derived here is for use in the controller and thus all constraints are formulated to fit into the controller which gives corrective signals to the boiler units.

The model has two input signals, d_i is the input signal coming from the production plan and u_i is the input signal coming from the load balancing controller. Thus in the nominal case u_i is zero since no corrective signals are needed.

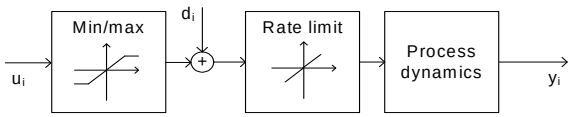


Figure 10.8: Concept of the boiler modelling

The process dynamics is modelled as the third order system

$$H(s) = \frac{1}{(T_i s + 1)^3} \tag{10.8}$$

where T_i is the time constant of effectuator i.

In order to gain offset-free tracking, the linear models are augmented with an output error model such that the constrained augmented discrete time state space model becomes

$$\mathbf{x}_{i,k+1} = \begin{bmatrix} a_{1,1,i} & 0 & 0 & 0 \\ a_{2,1,i} & a_{2,2,i} & 0 & 0 \\ a_{3,1,i} & a_{3,2,i} & a_{3,3,i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{i,k} + \begin{bmatrix} b_{1,i} \\ b_{2,i} \\ b_{3,i} \\ 0 \end{bmatrix} \mathbf{u}_{i,k} + \begin{bmatrix} e_{1,i} \\ e_{2,i} \\ e_{3,i} \\ 0 \end{bmatrix} \mathbf{d}_{i,k} \tag{10.9a}$$

$$\mathbf{y}_{i,k} = \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix} \mathbf{x}_{i,k} \tag{10.9b}$$

$$\underline{u_i} \leq u_i \leq \overline{u_i} \tag{10.9c}$$

$$\max\{\underline{\Delta u_i} - \Delta d_i, 0\} \leq \Delta u_i \leq \min\{\overline{\Delta u_i} - \Delta d_i, 0\} \tag{10.9d}$$

The elements in $\mathbf{A}_i$, $\mathbf{B}_i$ and $\mathbf{E}_i$ are dependent on T_i and the sample time. Symbols with an bar beneath, e.g., $\underline{u}$ means the lower bound, while $\overline{u}$ denotes the upper bound. The upper rate of change constraint is modelled such that it is always non-negative and vice versa, to avoid forcing the controller to take actions in case the production plan violates the rate of change constraint. The upper and lower limits for the controller (10.9c) are set in the control system by the operator.

The rate of change constraint is dependent on the boiler load. A typical form of the rate of change constraint as a function of the boiler load is depicted in Figure 10.9.

To linearise the constraint the prediction of u is used to generate rate of change constraints throughout the prediction horizon. If no prediction of u exists, it is assumed to be zero.

In case the operator changes the upper or lower bound such that the current control signal violates the limits, the limit is ramped down with the maximum allowed rate of change. This measure is taken to avoid infeasible optimisation problems.

The optimisation problem for each boiler unit is formulated as

$$\min_{\mathbf{U}} \quad \phi_i = \sum_{k=0}^{N-1} p_{i,k+1} y_{i,k} + ||y_{i,u,k+1}||_{1,q_{i,k+1}} + ||\Delta u_{i,k}||_{1,s_{i,k}} \tag{10.10a}$$

$$s.t. \quad \mathbf{x}_{i,k+1} = \mathbf{A}_i \mathbf{x}_{i,k} + \mathbf{B}_i u_{i,k} + \mathbf{E}_i d_{i,k}, \qquad k = 0, 1, \ldots, N-1 \tag{10.10b}$$

$$y_{i,k} = \mathbf{C}_i \mathbf{x}_{i,k}, \qquad k = 1, 2, \ldots, N \tag{10.10c}$$

$$\underline{u_{i,k}} \leq u_{i,k} \leq \overline{u_{i,k}}, \qquad k = 0, 1, \ldots, N-1 \tag{10.10d}$$

$$\underline{\Delta u_{i,k}} \leq \Delta u_{i,k} \leq \overline{\Delta u_{i,k}}, \qquad k = 0, 1, \ldots, N-1 \tag{10.10e}$$

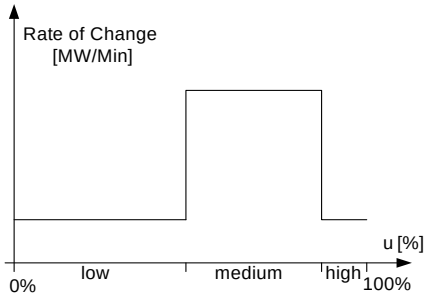


Figure 10.9: Actual rate of change constraint as a function of boiler load. This state dependency is not captured in the constraint (10.9d), but a linearisation based on the prediction is used in the model.

where $\mathbf{U}_i = [u_{i,0}, u_{i,1}, ..., u_{i,N-1}]^T$, and $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i, \mathbf{E}_i$ given in (10.9a) and (10.9b).

The first term in the performance function $p_{i,k-1}y_{i,k}$ is a linear term representing the cost of the boiler unit. The weight $p_{i,k+1}$ is the marginal cost, i.e., the cost for producing energy on the boiler unit. The price is calculated based on the fuel prices and boiler efficiency. The efficiency is state-dependent. For the calculations of $p_{i,k+1}$, it is based on the production plan alone.

It is assumed that the production plan from the STLS is optimal, and thus in the nominal case the correction signal from the load balancing controller should be zero. In order to avoid that the load balancing controller maximises the production of the cheapest units, and minimise the production of the most expensive units the term $||y_{u,k-1}||_{1,q_{y,k-1}}$ is added. This term penalises the part of the output coming from the controller corrections.

The last term of the performance function is a penalty on rapid changes on the correction signal.

This optimisation problem is the controller for unit i, and this information is stored in each of the effectuators on the lower layer of the hierarchy.

Primary reserve handling

Figure 10.10 shows an example of the maximum reserve available in both up and down direction as a function of the unit load. Reserves available for the positive and negative corrections are shown in the right and left half planes respectively.

Currently the Frequency Control Scheduler makes reservations of the reserves periodically. That means it might reserve 5MW of positive correction and 10MW of negative correction power from a specific unit at a given time.

It is chosen to give first priority to the Frequency Control Scheduler, and let it make the reservations. Once the reservations are known, the upper and lower bound for the unit can be determined, such that the reserved primary reserve can be delivered. These upper and lower bounds are enforced on the load balancing controller along with upper and lower bounds set by the operator.

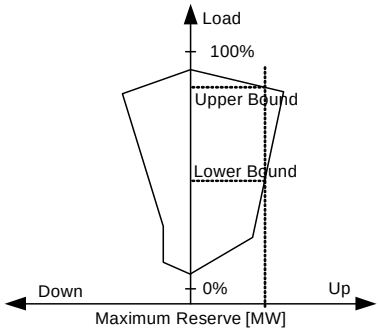


Figure 10.10: Primary Reserves as a function of unit load. On the y-axis is the unit load. On the x-axis the maximum possible primary reserve that can be delivered at that boiler load in both positive and negative direction. Dotted lines show a positive reserve reservation and the derived upper and lower input bounds for the controller. Similar reservation can be made at the same time for negative reserves.

Automatic / manual control and fall-back strategies

The boiler load effectuators can be in either automatic control or manual control mode. As explained earlier, in automatic control mode the controller can give corrective control signals to the unit. The units can also switch from Manual to automatic control, and vice versa. This event is assumed external and non-predictable; however, it is observable.

If the effectuator switches from automatic to manual control while $u \neq 0$ the strategy is to ramp the control signal toward zero with a predefined slope. This is done on both unit and in the controller, so in case of communication errors the behaviour of the unit can be predicted. The same fall-back strategy is used in case of faults in the effectuator.

These fall back strategies along with the control status are all handled in the lower layer of the hierarchy.

4.2 Portfolio modelling

The portfolio is comprised of the boiler load units modeled previously and a mixture of other production units. These other production units are various smaller thermal power plants and some wind turbines. They have a production reference and their production is measurable, but little is known about their dynamical behaviour. They are considered a disturbance in this context.

In order to include them in the controller the portfolio model consists only of an output error model and the output of the portfolio is the sum of all the production units, such that the model becomes

$$x_{port,k+1} = x_{port,k} \tag{10.11a}$$
$$y_{port,k} = x_{port,k} \tag{10.11b}$$

The optimisation problem for the portfolio is based on reference tracking and is given

as

$$\min_{U} \sum_{k=1}^{N} \left\| y_{port,k} + \sum_{i=1}^{P} y_{i,k} - r_{port,k} \right\|_{1,q_{port,k}} \tag{10.12a}$$

(10.12b)

where y_{port} is the output from the units lumped together and denoted portfolio. r_{port} is the portfolio reference which is the sum of references to all units in the system plus the demand from the TSO as shown in Figure 10.2. k is the sample number and N is the prediction horizon.

This optimisation problem is placed in the upper layer of the hierarchy.

5 Results

In order to evaluate the new controller it will be tested against the currently running controller through simulation in a scenario stretching throughout a month of real operation.

The current controller is implemented in Simulink™[Mathworks, 2010], and compiled so it is able to be executed in the central control room. In other words, it is the *actual* controller and not some simplified implementation of the controller the comparison is performed against. In order to test the new developments and maintenance of the current controller, models have been developed in Simulink™to be able to test the whole system. Since there already exists a test environment it has therefore been an obvious choice to make the comparison in Simulink™.

The new controller is implemented in mixture of Java for all the data handling such as reading measurement data and constructing constraints, and Matlab™[Mathworks, 2010] for solving the optimisation problem.

The dynamic part of the boiler unit models are implemented as linear models or linear parameter varying models. Besides the dynamic of the boiler unit, parts of the control system operating the boiler unit has been implemented, it means that all upper/lower bounds, rate of change constraints, correction for district heating and parasitic consumption are implemented in the models along with a lot of the logic controlling the switch from manual to automatic mode and vice versa.

The simulation environment runs at the same sample time as the current controller, i.e., 0.5s, and since the sample rate of the newly developed controller is 5s a ZOH approach will be taken. The data is saved with a 5s sample time for both controllers for analysis purposes.

Simulations cover 25-hour sequences starting from 23:00 to midnight the following day. In the analysis section the first hour is discarded, so the analysis covers 24-hour sequences from midnight to midnight. The first hour is then used to avoid startup and settling issues influencing the analysis, allowing to string together several sequences for more extensive analysis.

The controller is evaluated in both a noise-free and a noisy scenario.

5.1 Noise-free scenario

For the noise-free case standard deviation and mean error are used as quantitative measures for the evaluation.

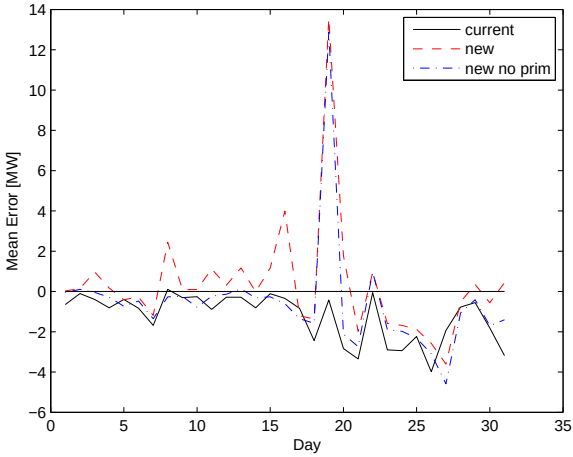


Figure 10.11: Mean error for the controllers in a scenario without noise. The figure shows the current controller (solid), the new controller (dashed) and the new controller with the primary reserve constraints removed (dash dotted). The results are for the individual days. Day 19 is omitted from the analysis due to missing measurement data for the scenario.

Figure 10.11 shows the mean error

$$\mu = \frac{1}{N_s} \sum_{k=1}^{N_s} y_{port,k} - r_{port,k} \tag{10.13}$$

with N_s being the number of samples in the simulation. Figure 10.12 shows the standard deviation

$$\sigma = \sqrt{\frac{1}{N_s} \sum_{k=1}^{N_s} ((y_{port,k} - r_{port,k}) - \mu)^2} \tag{10.14}$$

on a daily basis. Analysis shows that the constraints from primary reserves are limiting the controller in periods.

For analysis purposes the new controller the new controller is simulated without the primary constraints to compare the standard deviation and mean in a similar scenario to the current controller. The standard deviation is generally lower for the new controller, although there is a period from day 20 to 28 where the standard deviation is lower for the current controller.

The standard deviation and mean for the scenario are given in Table 10.1. The standard deviations are within 10% of each other, so even though there are differences they are close to each other.

Looking at the actual production (Figure 10.13), it is evident that both controllers tend to follow the reference well. However in some cases, adhering to the primary reserve constraint causes the proposed controller temporarily to perform poorer then the current controller, as can be seen from Figure 10.14. The interpretation is that there are

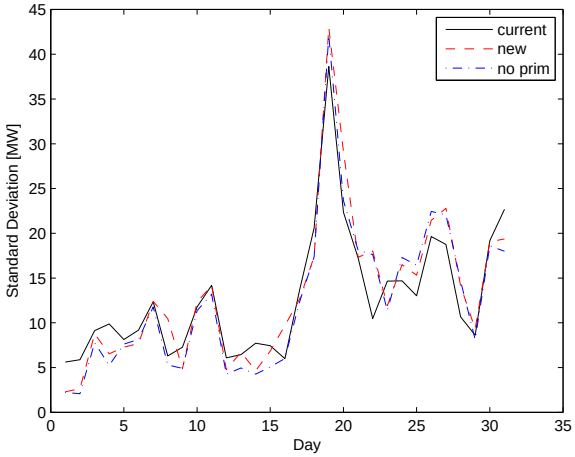


Figure 10.12: Standard deviation for the controllers in a scenario without noise. The figure shows the current controller (solid), the new controller (dashed) and the new controller with the primary reserve constraints removed (dash dotted). The results are for the individual days. Day 19 is omitted from the analysis due to missing measurement data for the scenario.

	Noise-free		Noisy	
	σ [MW]	μ [MW]	σ [MW]	μ [MW]
Measurements	-	-	17.74	-3.27
Current	11.98	-1.26	23.11	-2.78
New	12.21	-0.12	25.72	0.29
New no primary	11.41	-1.02	-	-

Table 10.1: Standard deviation and mean throughout the whole month of simulation

missing reserves to fulfill both the primary reserve reservations and follow the reference. Removing the constraint imposed by the primary reserves improved the performance significantly as can also be seen from Figure 10.14. Furthermore, switching from manual to automatic mode is handled efficiently by both controller, as seen from Figure 10.15.

5.2 Noisy scenario

For each boiler unit the input and output sample sequences of the original scenario are known. One can thus estimate a noise sequence for the scenario as

$$y_n = y_{meas} - y_{sim} \tag{10.15}$$

This noise is applied to the output of the model of the boiler unit. Since the noise is generated based on closed loop measurements, it is filtered by the controller in the loop rather than white noise.

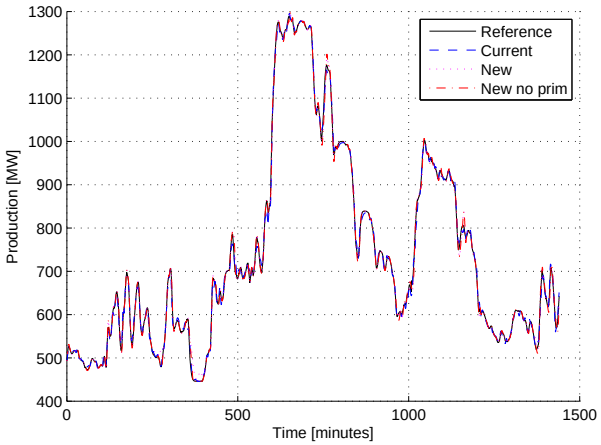


Figure 10.13: The production of day 3 in the scenario. Both controllers tend to follow the reference well. Both the new controller (dotted line) and the current controller (dashed line), follow the reference (solid line) well.

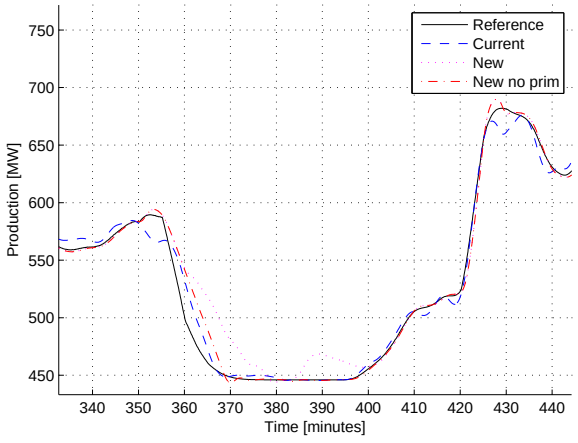


Figure 10.14: Section of the production on day 3, showing a period where the primary constraint is active and thus limits the new controller (dotted line) from reaching the reference (solid line). Removing this constraint make the new controller (dash dotted line) perform similar to the current controller (dashed line).

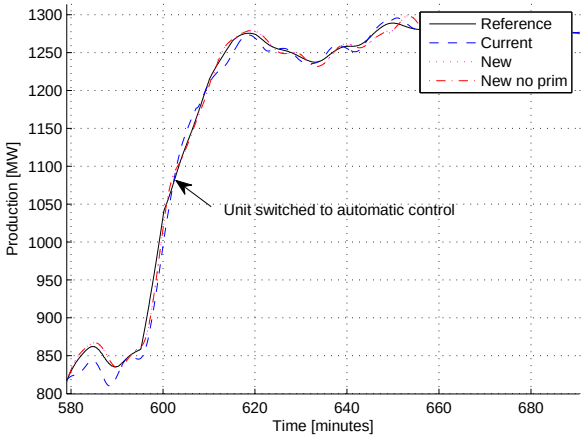


Figure 10.15: Section of day, where a power plant is switched from manual to automatic control. Both the new controller (dotted line) and the current controller (dashed line) handle this event in a bumpless fashion and follow the reference (solid line) well.

This noise generation is chosen in order for the simulation scenario to resemble the actual scenario as closely as possible including failures. The measurements from the units modelled as the portfolio are applied directly to the simulation without filtering.

The mean and standard deviation are once again used as quantitative measures for controller performance. The price difference between the controllers can be calculated given fuel costs and deviation prices.

The standard deviation on the new controller is higher than the current controller as was the case in the noise-free scenario. Both are significantly higher than the measurement data which is likely caused by the noise generation scheme as shown in Figure 10.17. The trend in standard deviation is the same for both controllers and measurement data. As seen in the noise-free scenario, the current controller has a slightly better performance than the proposed.

The mean error is larger in the noisy scenario compared to the noise-free. Though not consistently lower, the average shown in Table 10.1 shows that the new controller is an order of magnitude closer to zero mean error compared to the current controller.

Figure 10.18 shows the price difference between the two controllers. Analysing the price shows that on most days the new controller performs better in terms of income. On Day 20 the primary reserves limits the controller such that a large deviation occurs over a long period of time, which is detrimental for the earnings of the controller. On average the difference is 240 €/day, which means an earning of almost 90.000 €/year.

5.3 Execution time

The benefits of using Dantzig-Wolfe decomposition, besides the very logical decomposition, is that the execution time scales almost linearly with the number of units in control and that the problem can be easily distributed amongst multiple processors and thus low-

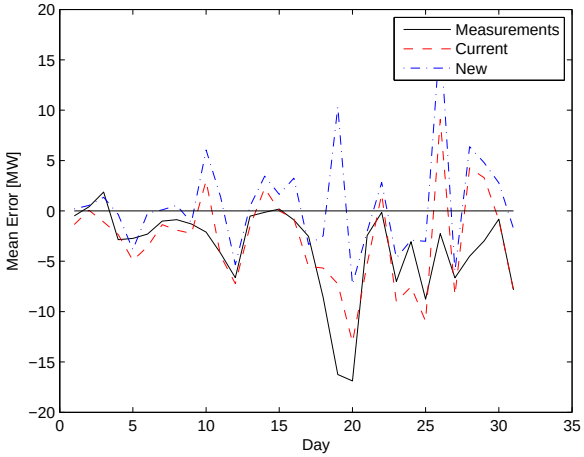


Figure 10.16: Mean error for the controllers in a noisy scenario. The figure shows the current controller (solid), the new controller (dashed) and the new controller with the primary reserve constraints removed (dash dotted). The results are for the individual days. Day 19 is omitted from the analysis due to missing measurement data for the scenario.

ering the execution time further.

As a benchmark of execution time, day 24 is chosen, where 4 units are controlled.

Prediction horizon [samples]	Execution time [s]
5	691
15	1245
25	3602
35	12960

Table 10.2: Execution time as a function prediction horizon

Table 10.2 shows the execution time as a function of the prediction horizon. These simulations are performed on a Dual Core Intel Xeon machine running at 2.53GHz with 4GB RAM and using Windows Vista as operating system. A 25 hours simulation can be performed in just about an hour. This timing is including the simulation.

Increasing the prediction horizon significantly increase the execution time of the controller. This has two explanations, one is the obvious that the problem size grows, and it was shown in [Edlund and Jørgensen, nd] that the execution time grows cubically with the prediction horizon. Secondly the algorithm may benefit from better handling of fast vs. slow unit dynamics (compared to the prediction horizon). This is subject of future research.

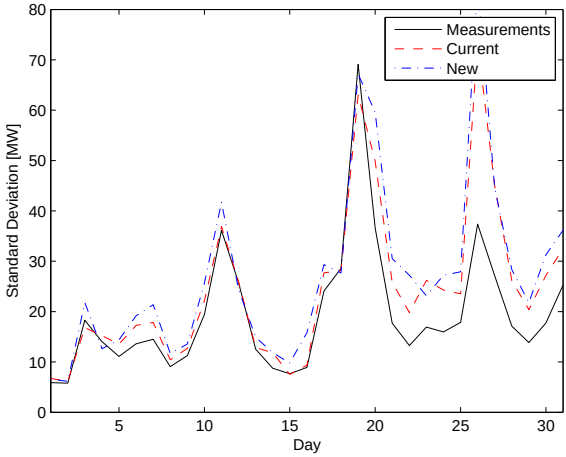


Figure 10.17: Standard deviation for the controllers in a noisy scenario. The figure shows the current controller (solid), the new controller (dashed) and the new controller with the primary reserve constraints removed (dash dotted). The results are for the individual days. Day 19 is omitted from the analysis due to missing measurement data for the scenario.

6 Conclusion

The aim for this paper was to develop a controller design method for developing a controller for power system portfolio control. In the future the portfolio is likely to grow significantly in the number of units under control. Therefore two design objectives were in focus: flexibility and computationally scalability.

The controller design involves a model predictive controller with a two layer hierarchy and some clearly defined interfaces. The underlying optimisation problem from the MPC controller was split into the same hierarchical structure by use the Dantzig-Wolfe decomposition algorithm. The decomposition of the optimisation problem also gave a computationally scalable controller. The Dantzig-Wolfe decomposition scales linearly in computational complexity with the number of units in control, and the optimisation problem is distributable over several computers. Solving the same optimisation problem in a centralised fashion would yield a cubic scalability.

The proposed controller design relies on one Kalman filter for state estimation of the whole system. The complexity of the matrix multiplications grow cubically with the problem size. This could prove to be a limiting factor for the scalability in the proposed control design. However, the computation time spent on the Kalman filter in the implementation is insignificant compared to the computation time spent by the controller. A logical future expansion of the design will be to incorporate distributed estimation with the same hierarchical structure as the controller.

The controller was tested in simulations both with and without noise. The newly developed controller has an extra constraint added compared to the current controller, in order to ensure primary reserves. In the noiseless case the newly developed controller was

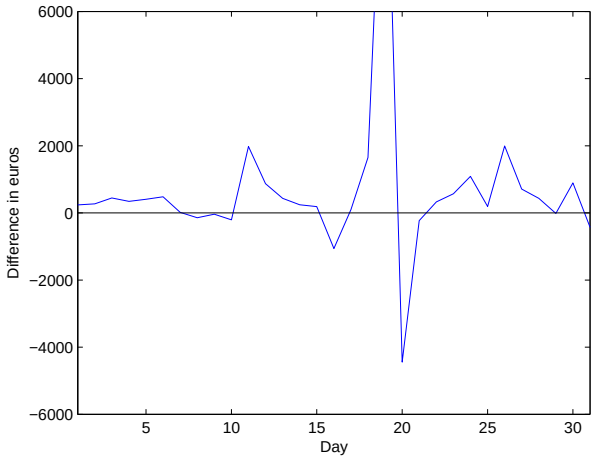


Figure 10.18: Price difference between the current controller and the new controller. Positive difference means that the new controller is cheaper (earns more money for DONG Energy).

tested both with and without the extra constraint. Without the extra constraint the standard deviation and mean was lowered compared to the currently implemented controller, when the extra constraint was added the standard deviation rose to a level above the current implementation.

In the noisy case the standard deviation was again higher than the currently implemented controller, which is likely caused by the constraint again. In the noisy case it was possible to calculate the cost of the production in the portfolio. The newly developed controller gave an economical gain compared to the current controller due to a better distribution of control action among the participating units.

One remark to make, is that the currently implemented controller has matured over the cause of years. In comparison the new controller has been implemented and tested through simulation for a very short time. It is therefore likely that the implementation and further development of the newly developed method will yield an improved performance compared to the results of this paper.

References

K. Aoki, T. Satoh, M. Itoh, T. Ichimori, and K. Masegi. Unit Commitment in a Large-Scale Power System including Fuel Constrained Thermal and Pumped-Storage Hydro. *IEEE Transactions on Power Systems*, 2(4):1077–1084, 1987.

A. K. Ayuob and A. D. Patton. Optimal thermal generating unit commitment. *IEEE Transactions on Power Apparatus and Systems*, 90(4):1752–1756, 1971.

Danish Ministry of Transport and Energy. Energistrategi 2025 - Perspektiver frem mod 2025. Technical report, Transport- og Energiministeriet, 2005.

G. B. Dantzig and M. N. Thapa. *Linear Programming 2: Theory and Extensions*. Springer-Verlag, 2002. ISBN 0-387-98613-8.

G. B. Dantzig and P. Wolfe. Decomposition Principle for Linear Programs. *Operations research*, 8(1):101–111, 1960.

T. S. Dillon, K. W. Edwin, H. D. Kochs, and R. J. Taud. Integer Programming Approach to the Problem of Optimal Unit Commitment with Probabilistic Reserve Determination. *IEEE Transactions on Power Apparatus and Systems*, PAS-97(6):2154–2166, 1978.

K. Edlund and J. B. Jørgensen. Dantzig-Wolfe Algorithm for Power Plant Portfolio Control. *Submitted for International Journal of Control*, nd.

K. Edlund, J. D. Bendtsen, S. Børresen, and T. Mølbak. Introducing Model Predictive Control for Improving Power Plant Portfolio Performance. *Proceedings of the 17th IFAC World Congress, Seoul, South Korea*, pages 6986–6991, 2008.

K. Edlund, J. D. Bendtsen, and P. Andersen. Structural Stability Analysis of a Rate Limited Automatic Generation Control System. *Proceedings of European Control Conference, Budapest, Hungary*, pages 4534–4539, 2009a.

K. Edlund, T. Mølbak, and J. D. Bendtsen. Simple Models for Model-based Portfolio Load Balancing Controller Synthesis. *Proceedings of IFAC Symposium on power plants and power systems control 2009, Tampera, Finland*, 2009b.

ENTSO-E. ENTSO-E Operational handbook. `http://www.entsoe.eu/index.php?id=57`, 2010.

R. Johnson, H. Happ, and W. Wright. Large Scale Hydro-Thermal Unit Commitment-Method and Results. *IEEE Transactions on Power Apparatus and Systems*, PAS-90(3): 1373–1384, 1971.

C. Jørgensen, J. H. Mortensen, T. Mølbak, and E. O. Nielsen. Modelbased Fleet Optimization and Master Control of a Power Production System. In *Proceedings of IFAC Symposium on Power Plants and Power Systems Control 2006, Kananaskis, Canada*, 2006.

Mathworks. `http://www.mathworks.com/`, 2010.

Nord Pool. `http://www.nordpool.com`, 2010.

N. P. Padhy. Unit Commitment - A Bibliographical Survey. *IEEE Transactions on Power Systems*, 19(2):1196–1205, 2004.

J. B. Rawlings and D. Q. Mayne. *Model Predictive Control: Theory and Design*. Nob Hill Publishing, 2009. ISBN 0-975-933770-9.

M. Ringel. Liberalising European electricity markets: Opportunities and risks for a sustainable power sector. *Renewable and Sustainable Energy Reviews*, 7(6):485–499, 2003.

J. A. Rossiter. *Model-based Predictive Control: A Practical Approach*. CRC Press LLC, 2003. ISBN 0-849-31291-4.

S. Salam. Unit Commitment Solution Methods. *Proceedings of the World Academy of Science, Engineering and Technology, Barcelona, Spain*, 26:600–605, 2007.

S. M. Shahidehpour and S. K. Tong. A scheduling model for the daily operation of an electric power system. *Applied Mathematical Modelling*, 16(5):226–244, 1992.

UCTE. Final Report System Disturbance on 4 November 2006. `http://www.ucte.org/pdf/Publications/2007/Final-Report-20070130.pdf`, 2007.

United Nations. Kyoto protocol to the United Nations framework convention on climate change. Technical report, United Nations, 1998.

A. Viana, J. P. de Sousa, and M. Matos. Simulated annealing for the unit commitment problem. *Proceedings of 2001 IEEE Porto Power Tech Proceedings*, pages 4–8, 2001.

A. J. Wood and B. F. Wollenberg. *Power Generation, Operation and control*. John Wiley & sons, inc., 2nd edition, 1996. ISBN 0-471-58699-4.