Czech Technical University in Prague
Faculty of Electrical Engineering

Bachelor Thesis

Predictive control of power balance in electric grid

Prague, 2007

Author: Ondřej Malík
Supervisor: Petr Havel

Declaration

I declare that I have created my Bachelor Thesis on my own and I have used only literature cited in the included reference list.

In Prague, _______________ ____________________

signature

Acknowledgement

I would especially like to thank my supervisor Petr Havel for guidance and for useful comments and also to people from ČEPS, a. s., for provided consultations. I would also like to thank my family and all who supported me during my studies.

Abstrakt

Problémem optimálního užívání regulačních rezerv (tzv. podpůrných služeb) z hlediska provozovatele elektrické přenosové soustavy se doposud zabývalo velmi málo studií. Účelem této práce je vytvořit takový algoritmus, který nalezne rozvrh aktivace podpůrných služeb tak, aby dodaná regulační energie pokryla předpokládaný rozdíl mezi výrobou a spotřebou elektrické energie (tzv. predikci systémové odchylky) při minimálních nákladech na použití podpůrných služeb. Jedná se tedy o specifickou regulační úlohu, která využívá metod optimálního rozvrhování. Tato úloha má některé netypické vlastnosti a omezení, vyplývající jednak z technických omezení elektrárenských bloků poskytujících podpůrné služby, a jednak z dohod stanovujích způsob jejich využití. Všechny tyto vlastnosti musí být v optimalizačním algoritmu zahrnuty. Vzhledem k tomu, že navrhovaný algoritmus zahrnuje i služby s rychlým náběhem, musí být použita odpovídající vzorkovací frekvence, což klade vysoké nároky na efektivnost algoritmu.

Navrhovaný algoritmus je založen na lineárním programování s celočíselnými proměnnými, díky nímž lze modelovat i některé vlastnosti, které by pouze pomocí spojitých proměnných byly obtížně formulovatelné. Zahrnuty jsou typické vlastnosti elektrárenských bloků jako omezení rychlosti náběhu, minimální doba po kterou musí elektrenský blok dodávat regulační energii resp. doba, po kterou nesmí být dodávka regulační energie blokem znovu obnovena, ceny za najetí a za dodanou energii, současně jsou však zahrnuty i speciální penalizace za předčasnou deaktivaci dodávky regulační energie bloku, předčasnou aktivaci regulační enegie bloku nebo za počet nájezdů bloku.

Z výsledků je patrné, že algoritmus umožňuje efektivně ovlivňovat způsob aktivace podpůrných služeb při respektování jejich vlastností jako je náběh konstantním trendem, zpoždění mezi pokynem k aktivaci a samotným náběhem služby a dalších. Provedené testy ukazují, že je použitelný i pro rozvrhování regulačních problémů o velikosti, která se blíží běžné velikosti při řízení české elektrické přenosové soustavy, a může tedy sloužit pro podporu rozhodování dispečerů provozovatele přenosové soustavy při aktivaci podpůrných služeb.

Abstract

The problem of optimal utilization of regulation reserves (provided by the Ancillary Services) from the Transmission System Operator point of view received little attention to date. The purpose of this thesis is to create an algorithm which will find a schedule of Ancillary Services activation so that their regulation energy covers the predicted deviation between the power production and the power demand (the System Deviation prediction) with minimal expense for the regulation energy. It is therefore a specific regulation problem which utlilizes methods of optimal scheduling. This problem has some uncommon properties resulting from technical limitations of generating units which provide the Ancillary Services and from agreements which specify how the services should be utilized. The scheduling algorithm must take these into account, moreover, inclusion of services with fast reaction times requires a fast sampling rate, which puts high requirements on algorithm efficiency.

The proposed algorithm is based on Mixed Integer Linear Programming, as integer variables allow for modelling of some properties that would be hard to model by using continuous variables. Included in the algorithm are typical properties of generating units such as ramp rate limitations, minimal time the reserve power of generating unit must be activated or deactivated, startup costs and costs of regulation energy, along with special penalizations for reactivating the reserve power of generating unit too early after its deactivation and deactivation of reserve power too early after its activation or for number of reserve power activations.

The results show that the algorithm allow for efficiently influencing the way the services are activated while respecting the services properties such as constant ramp rates, startup delay and other. Performed tests indicate that the algorithm is applicable even for problems with size which is close to the size of problems dealt with when controlling the Czech Transmission System and therefore may serve as a decision support tool for Transmission System Operator dispatchers.

vložit originální zadáníííííííííííííííí !!!!!

Contents

List of Figures

List of Tables

Nomenclature

k	time-sample index
j	service index
J	optimality criterion
$u_j(k)$	input of j-th service in k-th sampling interval [-]
$\Delta u_j(k)$	input difference of j-th service in k-th sampling interval [-]
$\Delta u_j^{on}(k)$	binary variable containing information whether the j-th service was brought online in k-th sampling interval [-]
$\Delta u_j^{off}(k)$	binary variable containing information whether the j-th service was brought offline in k-th sampling interval [-]
D_j^{SU}	startup delay of j-th service [minutes]
$p_j(k)$	power of j-th service in k-th sampling interval [MWh]
$\overline{p}_j$	maximal power of j-th service [MWh]
$\underline{p}_j$	minimal power of j-th service [MWh]
$r_j^{\Delta p}$	power change rate of j-th service [MW/min]
$\overline{sat}_j^p(k)$	binary variable indicating whether the power of j-th service is on its maximal level in k-th sampling interval [-]
$\underline{sat}_j^p(k)$	binary variable indicating whether the power of j-th service is on its minimal level in k-th sampling interval [-]
$\overline{usat}_j^p(k)$	product of $u_j(k)$ and $\overline{sat}_j^p(k)$ [-]
$\underline{usat}_j^p(k)$	product of $u_j(k)$ and $\underline{sat}_j^p(k)$ [-]
$\overline{E}_j$	maximal energy available from j-th service [MWh]
t_j^{SU}	startup time of j-th service [minutes]
$SDP(k)$	System Deviation Prediction for k-th sampling interval [MW]
$t_j^{on}(k)$	ammount of time that j-th service was online from last activation [minutes]
$t_j^{off}(k)$	ammount of time that j-th service was offline from last deactivation [minutes]

$\underline{t}_j^{off}$	minimal off time of j-th service [minutes]
$\underline{t}_j^{on}$	minimal on time of j-th service [minutes]
$ut_j^{on}(k)$	product of $u_j(k)$ and $t_j^{on}(k)$ [minutes]
$ut_j^{off}(k)$	product of $u_j(k)$ and $t_j^{off}(k)$ [minutes]
n_j^{SU}	number of startups of j-th service during optimization interval [-]
c_j^p	cost of 1 MWh of j-th service energy [CZK/MWh]
c_j^{SU}	cost of j-th service startup [CZK/MW]
pen^D	penalization for deviance of optimized input from System Deviation prediction [CZK/MWh]
$pen_j^{SU}(t_j^{off}(k))$	piecewise linear penalization for early startups of j-th service in k-th sampling interval, dependent on time spent offline before activation [CZK]
$pen_j^{SD}(t_j^{on}(k))$	piecewise linear penalization for early shutdowns of j-th service in k-th sampling interval, dependent on time spent online before activation [CZK]
$\overline{pen}_j^{SU}$	maximal startup penalization, incurred if the service is shut off the next sampling interval after its activation [CZK]
$\overline{pen}_j^{SD}$	maximal shutdown penalization, incurred if the service is activated the next sampling interval after its shutdown [CZK]
$\Delta upen_j^{SU}(t_j^{off}(k))$	product of Δu^{on} and $pen_j^{SU}(t_j^{off}(k))$ [CZK]
$\Delta upen_j^{SD}(t_j^{on}(k))$	product of Δu^{off} and $pen_j^{SD}(t_j^{on}(k))$ [CZK]
$pen_j^{NSU}(n_j^{SU})$	piecewise linear penalization function for number of startups of j-th service
PEN_j^{SU}	total penalization for early startups of j-th service [CZK]
PEN_j^{SD}	total penalization for early shutdowns penalization of j-th service [CZK]
PEN_j^{NSU}	total penalization for number of starts of j-th service [CZK]
T_s	sampling period [minutes]
N_{sm}	number of samples in optimization interval [-]
N_{sr}	number of services included in optimization [-]

Chapter 1

Introduction

This thesis deals with a problem of optimal utilization of regulation reserves to compensate imbalance in power grid. Its intention is to provide basic research for development of decision support tool for the Transmission System Operator (TSO) dispatchers. The power balance control presents a complex feedback control problem with structure as shown in figure 1.1. The power imbalance is a difference between energy generation and energy demand which is also referred to as System Deviation (SD). In feedback control terminology, the SD is an error which should be compensated by the regulation energy. The regulation power is a manipulated variable that is currently controlled partly by automatic controllers and partly be the TSO dispatchers. The regulation reserves are provided by Ancillary Services (AnS) which are mostly purchased from power producers, less so from power consumers.

The parts of power grid balance control which are operated by TSO are outlined red in figure 1.1. The TSO's dispatch center serves as the main controller and manipulates the regulation energy provided by following AnS:

- *Secondary regulation (SR)* which is a service provided on running generating units, so-called spinning reserve. It automatically controls the load balance based on PI regulator algorithm.

- *Tertiary Regulation (positive*[1] *(TR+) and negative*[1] *(TR-))* are also spinning reserves, which are activated on request of TSO's dispatchers. They are used to

[1]The term "positive regulation energy" will be used to describe energy that must be activated to compensate for a deficit of energy in TS. To keep this notation consistent, the "positive SD" will be the deficit of energy in TS. Therefore "negative regulation energy" will be energy used for compensating an overproduction of energy in TS and "negative SD" will mean the overproduction of energy in TS.

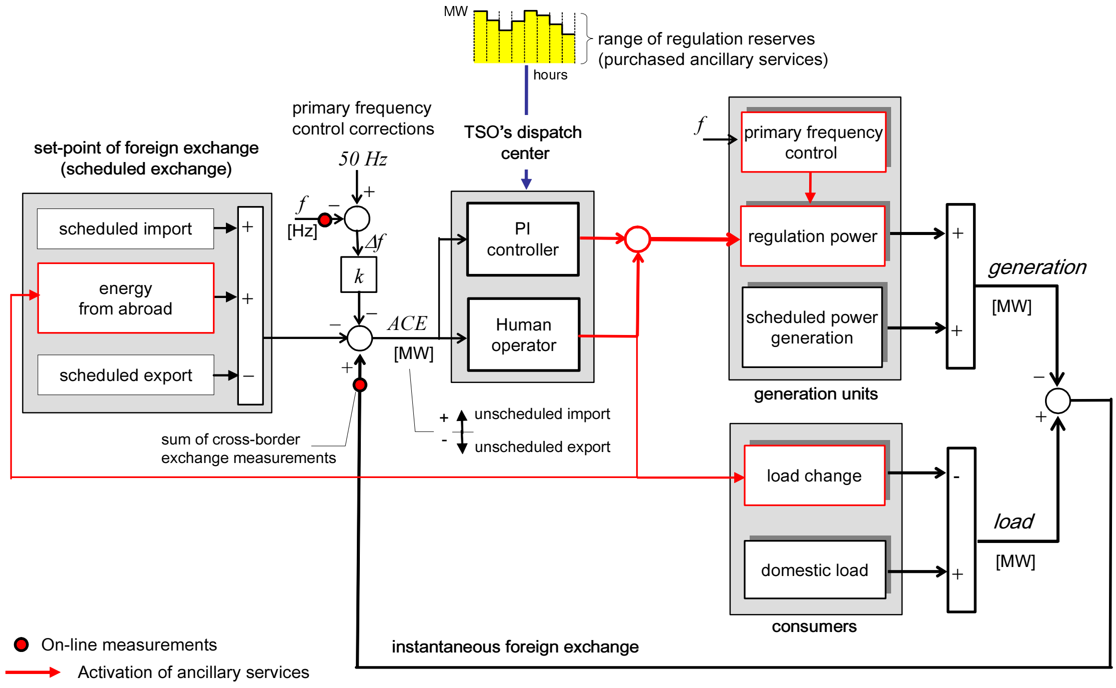

Figure 1.1: Principles of feedback control of Czech Transmission System

compensate for depleted SR energy.

- *Dispatch Reserve (DZ)* is provided on generating units that are offline in normal state (it is a non-spinning reserve), but are ready to receive activation request from TSO dispatchers and must be able to begin providing regulation energy in specified time. It is used for covering long term outages of generating units.

- *Domestic regulation energy from balance market (Ereg)* presents the regulation energy that may be bought on domestic balance market but isn't provided as one of aforementioned AnS.

- *Regulation energy from abroad (EregZ)* has the same properties as Ereg, but is bought on international markets.

- *Load Change* is provided by domestic consumers who offer the possibility of lowering their power demand on TSO's dispatchers request.

The Primary Regulation (PR) is not controlled directly by TSO's dispatchers, it's activation is controlled by an PI controller installed on every generating unit which provides PR.

The total generation in the Transmission System (TS) consist of aforementioned AnS (excluding Load Change) and scheduled energy generation (the energy which the power producers have scheduled to generate). The total load in TS includes the Load Change Ancillary Service and domestic load (the amount of energy that the domestic consumers demand). The difference between the generation and the load, with the the foreign energy exchange taken into account, is the SD.

In short term view, the SD behaves randomly, however, in longer term view, trends may be found in SD development. Based on observation of these trends and the actual state of power grid, it is possible to create a prediction of future SD development (System Deviation Prediction, SDP), which is generally quite precise for several hours into the future in case that no sudden outage of generating unit occurs. Therefore it is reasonable to prepare an "optimal" schedule of AnS activation to cover SDP. From the control point of view, the term "optimal" is defined as complete coverage of SD by regulation energy from AnS (i. e. reaching the balanced state of power grid). From the economic point of view, it is defined as reaching the balanced state of power grid with minimal expense for AnS activation. However, there are many units providing AnS, which have various properties and limitations, so it may be hard for a human to take all these properties into account and create a schedule which is optimal from both mentioned points of view. On the other side, an optimization algorithm incorporating all the properties and optimizing the AnS utilitation from both points of view may be developed and used as decision support tool for TSO dispatchers.

The problem of optimal usage of AnS shares many properties with the process of startup and shutdown scheduling of generating units referred to as Unit Commitment (UC). Various optimization methods have been developed to obtain optimal UC such as Langrangian Relaxation [1], Mixed Integer Linear Programming (MILP) [2, 3, 4], Quadratic Programming (QP) [5], Hybrid Particle Swarm Optimization [6] and Evolutionary Algorithm (EA) [7]. AnS were incorporated in some UC models (for example [8]), but they only presented a part of generating units energy which may be sold as regulation energy and therefore have positive effect on economic optimality. These studies may however be used to help model basic properties of AnS, such as ramp limits or minimal on and off times.

The proposed algorithm will be formulated using Mixed Integer Linear Programming and will include following AnS: SR, TR+, TR-, DZ, Ereg and EregZ. The PR won't be included as its power is low compared to uncertainty of SDP and it is an independent, self-controlled system. The Load Change won't be included because it may be activated only under certain conditions, which cannot be modeled in optimization algorithm. The emphasis will be put mainly on providing framework to model AnS properties, dynamics, and limitations.

Chapter 2

Properties of Ancillary Services

2.1 Common properties

In this section, some basic terms used throughout this thesis for description of services properties will be clarified. Figure 2.1 shows services properties associated with time on example of a typical service activation schedule:

- *Startup delay* is a time between request to activate the service and beginning of actual service startup.
- *Startup time* is a time in which the service increases its power from zero to maximal power.
- *On time* is a time between beginning of actual startup and beginning of the following shutdown.
- *Off time* is a time between beginning of actual shutdown and beginning of the following startup.

Figure 2.2 shows services properties associated with power:

- *Offline state* is a state when service power is zero.
- *Minimal power* presents a minimal power to which the service must be activated.
- *Maximal power* is a maximal output of the service.

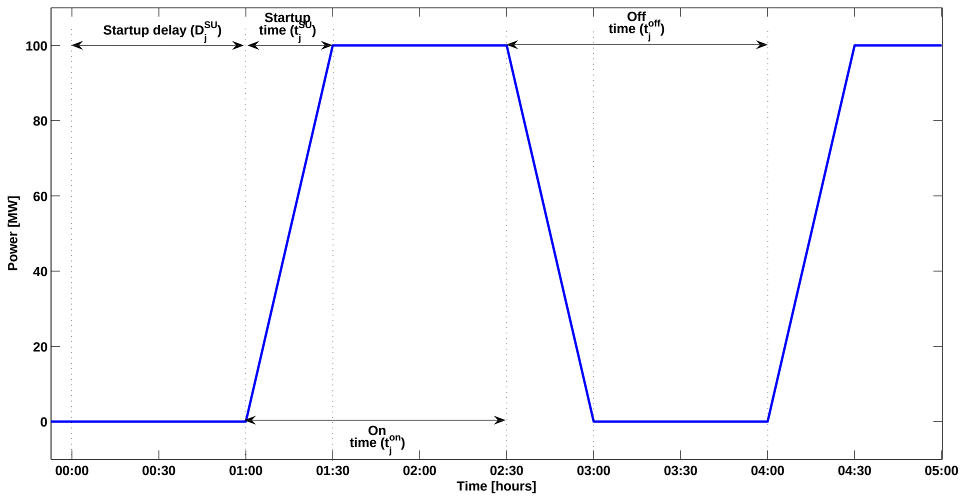

Figure 2.1: Services properties associated with time

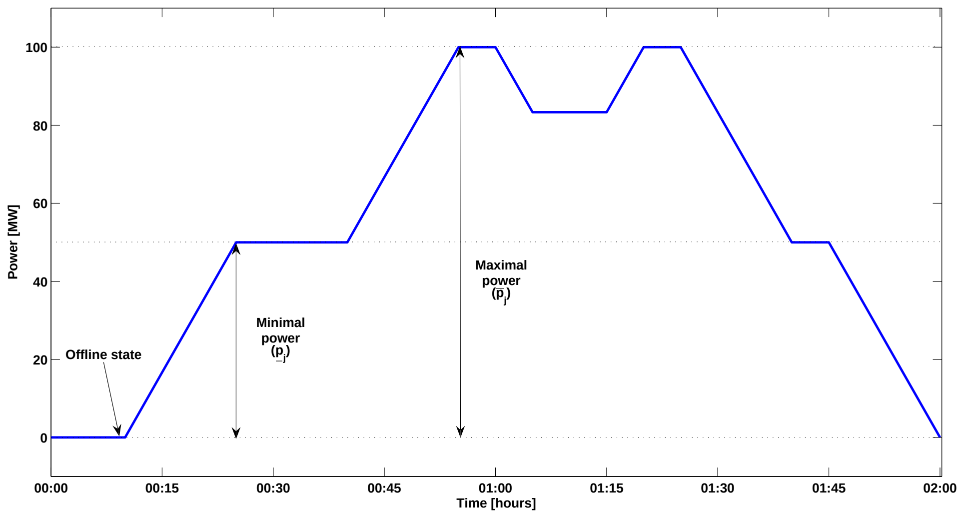

Figure 2.2: Services properties associated with power

2.2 Overview of basic services properties

Table 2.1 shows basic properties of AnS. Apart from them, some special properties, requirements and gentleman agreements are associated with each type of AnS. These are discussed below along with more precise description of each service properties. Official services purposes and properties may be found in [9], Part II.: Ancillary Services (AnS).

Type of service	**SR**	**TR**	**QS**	**DZ**	**Ereg**	**EregZ**
Startup time [minutes]	10	30	10	30, 60, 90, 360	0	
Startup delay	No	No	No	Yes	Yes	
Min ramp rate [MW/min]	2	2	N/S[a]	N/S	N/S	
Min volume [MW][b]	10	10	30	15	N/S	
Max volume [MW][b]	N/S	100	N/S	N/S	N/S	
On/Off[c]	No	Yes	No	Yes	No	
Startup costs	No	No	No	Yes	No	

Table 2.1: Basic Ancillary Services properties

[a]Not Specified

[b]The minimal and maximal volumes are limitations that specify the minimal power that must be provided on a single generating unit so that the unit may be offered as the respective AnS.

[c]On/Off service has only two operating states: If the unit is On, it is either starting up with specified ramp-up rate or is on its maximum power. If the unit is Off, it is either shutting down with specified ramp-down rate or is completely shut down. Its power output cannot remain between the minimal and maximal power

2.3 Secondary Regulation (SR)

The Union for the Coordination of Transmission of Electricity (UCTE) sets the minimal amount of SR energy that should be enough for maintaining power balance in the region (The Czech Republic in this case) in standard situations - this volume of SR energy should remain in control of automatic PI controller, primarily to handle fast load changes. However, more SR energy is usually available, so the redundant volume of SR is made available for use in proposed optimization algorithm.

The most valued feature of SR is that its activated power may be changed continuously[1] to any value between its minimal and maximal power, while respecting ramp-up and ramp-down limits. Considering this fact, SR will be modeled as one service with continuous inputs and outputs. There are no restrictions on how often power changes may occur, which again raises its value for power balance control.

2.4 Tertiary Regulation (TR)

TR is used to compensate for the depleted SR. It will be modeled as an On/Off service and each generating unit providing TR will be included separately in optimization algorithm. Because of technical limitations, TR shouldn't be deactivated sooner than one hour after its activation, unless it is absolutely inevitable. This restriction will be included in OC as a penalization for early startups or as minimal on or off time constraints.

2.5 Quick-Start 10 minute reserve (QS)

QS is a service provided mainly by pumped storage plants. It is used to quickly cover large sudden outages (when available SR is insufficient) and to compensate very fast SD changes. As it is provided on pumped storage plants, there is a natural capacity limitation associated with the amount of water stored in storage tanks. According to [9], the service must be able to operate at maximal power for at least four hours after its activation. It must be activated at least to a minimal power (set in agreement) and after reaching the minimal power, the power output may be set anywhere between the minimal and the maximal power of the activated block while respecting ramp-up and ramp-down limits. However, the number of cold starts[2] is to be kept as low as possible because it presents excessive stress on generating unit technical equipment. In addition, even the number of power changes when the service is activated shouldn't be too high as it again has negative influence on service reliability.

QS will be modeled as conjunction of two type of services - the On/Off service for the operating range from zero to minimal power and continuous service similar to SR for

[1] The term *continuous* will be used to indicate that the variable is continuous in value. The time will always be discrete as digital controller is employed.

[2] Activation of service which was offline.

the operating range between minimal and maximal power. Penalties for number of cold starts and number of power changes will be included in OC.

2.6 Dispatch Reserve (DZ)

DZ is a service mainly used for dealing with long term outages of generating units. As it is a non-spinning reserve, there is startup delay related to bringing the generating unit online. Startup cost is added to the generation costs each time the DZ unit is activated. One DZ unit should be activated at maximum two times a day. After deactivation the startup delay must again be considered before another activation, although in practice the time needed for startup soon after shutdown may be lower. DZ will be modeled as an On/Off service with startup delays 30, 60, 90 or 360 minutes according to the DZ service type. The penalizations for undesired style of activation will be used in the same way as in section 2.4.

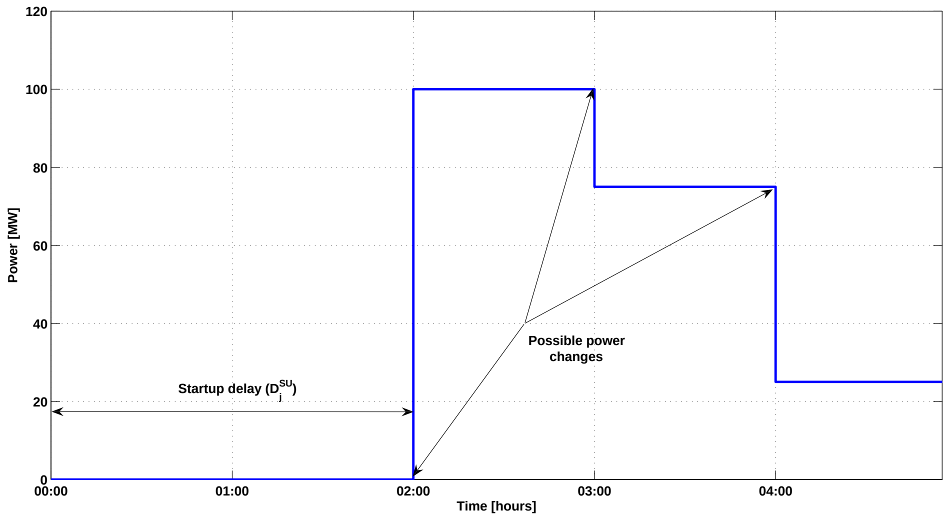

Figure 2.3: Style of Ereg and EregZ activation

2.7 Domestic energy from balance market (Ereg) and Regulation energy from abroad (EregZ)

Ereg and EregZ are specific types of AnS - they present the energy obtained on domestic balance market (in case of Ereg) or International energy market (in case of EregZ). As shown in figure 2.3, the arranged volume of Ereg or EregZ may be activated or changed only on a change of business interval. The power change is assumed instant, and the time needed to arrange regulation energy purchase is considered 2 hours.

Chapter 3

Optimal scheduling of Ancillary Services

In this chapter, the general formulations of optimality criterion and services dynamics will be given and then they will be formulated as Mixed Integer Linear Program. MIQP and MILP were considered for the problem formulation and although MIQP seems to better model the SD penalization[1] (as large SD values should be avoided at all costs which quadratic function models quite well, though nowhere is strictly defined, that the penalization should be quadratic), MILP was chosen in the end for following reasons:

- More solvers are available for solving MILP and they are much faster than MIQP solvers.
- Piecewise linear cost function may be used with MILP, therefore approximation of quadratic cost function may be modeled if needed (though it increases problem complexity resulting in performance degradation).

The optimization will be performed over the optimization interval of length $N_{sm} \cdot T_s$ [minutes] as shown in figure 3.1. The last sample until which the optimization is performed is called *Prediction horizon*.

[1]In an ideal case, if the real SD was exactly as its prediction and the services were activated according to the resulting schedule, the new SD will be the deviation between the SDP and scheduled output, as shown in figure 3.1. Therefore the term *SD penalization* will be used for the penalization of deviation of scheduled output from the SDP.

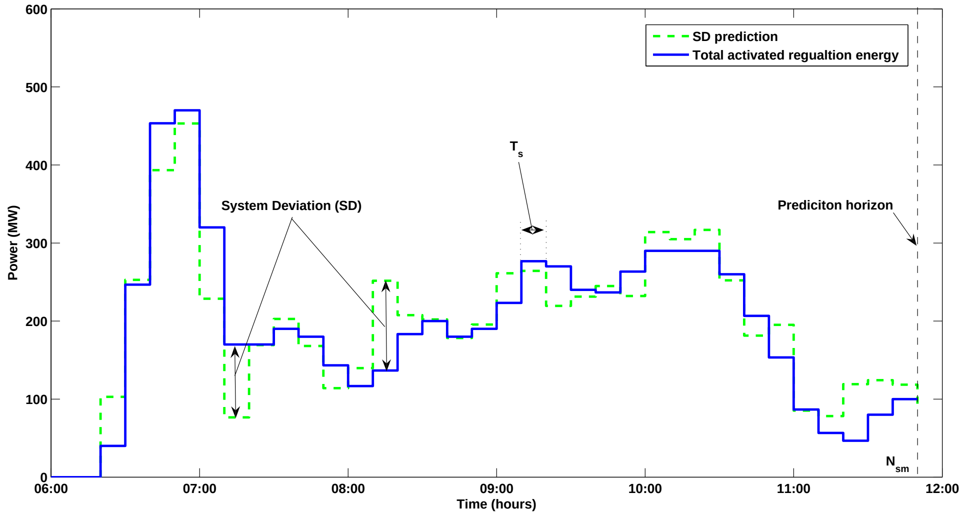

Figure 3.1: The optimization interval description

3.1 General Optimality Criterion formulation

The OC as proposed consists of real costs (the energy price and startup costs[2]) and virtual penalizations for unwanted services behaviour. The OC (3.1) contains the following:

Real costs

- *Energy costs* are costs of energy used during the services activation. The costs per MWh of energy supplied by j-th generating unit (c_j^p) are fixed in agreements between TSO and services providers.
- *Startup costs* are cost accounted for activation of services. Currently, only DZ service has these costs set; They are set per MW of activated service power, but in current state, full power must always be activated, therefore they may be computed as one time fixed cost per service activation during the optimization interval.

[2]As stated in section 2.2, real startup costs are issued only for DZ services startup.

Virtual penalizations

- *Penalization for System Deviation* is a configurable parameter of the proposed algorithm. It is a good practice to set this penalization higher than a energy price of the most expensive service as if it was set lower, the services which have higher price than this penalization will never be activated[3]. In general case, the penalization pen^D may be modeled as piecewise linear function SD to allow for more complex penalization functions.

$$
\begin{aligned}
J = &\underbrace{\sum_j \sum_k c_j^p \cdot p_j(k) +}_{\text{real energy costs}} \\
&\underbrace{+\sum_j c_j^{SU} \cdot \bar{p}_j \cdot n_j^{SU} +}_{\text{real startup costs}} \\
&\underbrace{+pen^D \cdot \sum_k \left(\sum_j p_j(k) - SDP(k) \right) +}_{\text{penalization for System Deviation}} \\
&\underbrace{+\sum_j PEN_j^{SU} +}_{\text{penalization for early startups}} \\
&\underbrace{+\sum_j PEN_j^{SD} +}_{\text{penalization for early shutdowns}} \\
&\underbrace{+\sum_j PEN_j^{NSU}}_{\text{penalization for number of startups}}
\end{aligned}
\tag{3.1}
$$

- *Penalization for early startups* and *Penalization for early shutdowns* are included to control how services are activated. They penalize such schedules in which the service is started up too early after its deactivation (or deactivated too early after its activation) to conform the activation schedules to rules set on service usage. They may be also modeled as piecewise linear functions to allow for using more complex penalization functions. Equation (3.2) defines how the Penalization for early startups is computed and equation (3.3) defines how the Penalization for early shutdowns is computed.

[3]This is only valid when linear cost function is used.

$$PEN_j^{SU} = \sum_k pen_j^{SU}\left(t_j^{off}(k)\right) \cdot \Delta u_j^{on}(k)\,,$$
$$\text{where } \Delta u_j^{on}(k) = \begin{cases} 1 \text{ if the unit was brought online in sampling interval } k \\ 0 \text{ otherwise} \end{cases} \quad (3.2)$$

$$PEN_j^{SD} = \sum_k pen_j^{SD}\left(t_j^{on}(k)\right) \cdot \Delta u_j^{off}(k)\,,$$
$$\text{where } \Delta u_j^{off}(k) = \begin{cases} 1 \text{ if the unit was brought offline in sampling interval } k \\ 0 \text{ otherwise} \end{cases} \quad (3.3)$$

- *Penalization for number of startups* is used to control service activation frequency which is also part of gentleman agreements set on service usage. It may also be modeled as piecewise linear function if required. The Penalization for number of startups is computed based on equation (3.4).

$$PEN_j^{NSU} = pen_j^{SU}\left(n_j^{SU}\right) \quad (3.4)$$

3.2 Models of services dynamics

To incorporate the service dynamics into the optimization algorithm, a model must be created. The model describes relations between consecutive sampling intervals to take into account such properties as maximal or constant ramp rates, or saturation at maximal or minimal power. Five models of services dynamics were developed for usage with scheduling algorithm:

- two for On/Off services:
  - *Model based on first order system (FO model)*,
  - *Model with constant ramp rates (CR model)*,
- *Three state model*[4] *(3S model)* for services which power may be changed while the service is activated,

[4]This model isn't used in latest version of scheduling algorithm, but is documented for possible future use.

- *Model with variable ramp rates (VR model)* for services which power may be changed continuously,
- *Hourly model (HC model)* for services which power may be altered only on changes of business intervals.

Difference equations will be used for description of models as they are simple and clear and may be used almost directly with Yalmip [10] in Matlab [11].

3.2.1 Required services dynamics

Following list sums up the dynamics that are referential for each type of AnS:

- On/Off services (TR, DZ) start up and shut down with constant power change rate defined by (3.5). They may only be activated to full power or put offline.
- SR service may continuously change its power between minimal and maximal power as long as its power changes respect maximal power change rate set by (3.5).
- QS service must be activated to at least minimal power above which the power may be changed continuously between minimal and maximal power while respecting the maximal power change rate defined by (3.5).
- Ereg and EregZ services may be activated to any power level between minimal and maximal power only only between trading hours. During the time interval between changes, the power level remains unchanged.

$$r_j^{\Delta p} = \frac{\bar{p}_j}{t_j^{SU}} \tag{3.5}$$

3.2.2 Model with variable ramp rates

The VR model is defined by (3.6).

$$\begin{aligned} p_j(k+1) &= p_j(k) + u_j(k) \cdot r_j^{\Delta p} \\ -1 &\leq u_j(k) \leq 1 \\ \underline{p}_j &\leq p_j(k) \leq \overline{p}_j \end{aligned} \tag{3.6}$$

The input $u_j(k)$ specifies the rate of service power change scheduled in k-th sampling interval, which may take any value between $\pm r_j^{\Delta u}$. $u_j(k)$ and $p_j(k)$ are continuous variables, the constraint set on $u_j(k)$ ensures, that the maximal power change rate doesn't exceed the rate specified by (3.5) and the constraint set on $p_j(k)$ bounds the service power within its minimal and maximal power.

3.2.3 Model with constant ramp rates

The CR model precisely meets the requirements set in section 3.2.1 for On/Off services. However, the nonlinearity associated with saturation at maximal/minimal power makes the system model more complex and, as will be shown later, much less efficient than the FO model. The CR model is defined by (3.7).

$$\begin{gathered}
p_j(k+1) = p_j(k) + u_j(k) \cdot \left(1 - \overline{sat}_j^p(k)\right) \cdot r_j^{\Delta p} - (1 - u_j(k)) \cdot \left(1 - \underline{sat}_j^p(k)\right) \cdot r_j^{\Delta p} \\
u_j(k) \in \{0, 1\} \\
\overline{sat}_j^p(k) = \begin{cases} 1 \text{ if the service power was at maximum in } k \text{ - th sampling interval} \\ 0 \text{ otherwise} \end{cases} \\
\underline{sat}_j^p(k) = \begin{cases} 1 \text{ if the service power was at minimum in } k \text{ - th sampling interval} \\ 0 \text{ otherwise} \end{cases}
\end{gathered} \tag{3.7}$$

In a state when service power is between minimal and maximal power, the equation (3.7) may be rewritten into form (3.8), as both $\overline{sat}_j^P(k)$ and $\underline{sat}_j^P(k)$ are 0. Then, if the input is 1, the service is starting up with rate $r_j^{\Delta p}$ or if input is 0, the service is shutting down with rate $r_j^{\Delta p}$.

$$p_j(k+1) = p_j(k) + u_j(k) \cdot r_j^{\Delta p} - (1 - u_j(k)) \cdot r_j^{\Delta p} \tag{3.8}$$

As the power reaches its maximal level with input still being 1, the term $(1 - \overline{sat}_j^P(k))$ becomes 0 (the second part of the equation (3.7) is also 0 because the input is 1 and term $(1 - u_j(k)) \cdot \left(1 - \underline{sat}_j^p(k)\right) \cdot r_j^{\Delta p}$ therefore remains 0) and the equation (3.7) transforms into (3.9), making the power stay at maximal level. The situation at minimal level is analogous.

$$p_j(k+1) = p_j(k) \tag{3.9}$$

3.2.4 Model based on first order system

The FO model was developed to speed up the optimization process. Its main advantage is that the dynamics description is simple and no other constraints than limiting inputs to 0 or 1 are needed (if the input is 0, the service is shutting down or staying at minimal power, if it is 1, the service is starting up or remains at maximal power). The major drawback is that it doesn't precisely correspond with requirements in section 3.2.1, it is just an approximation. A compromise between correct settling time and correct ramp rate had to be found. General transfer function of first order system along with relation between natural frequency ω_n and time constant T_p is shown in equation (3.10).

$$G(s) = \frac{K \cdot \omega_n}{s + \omega_n} = \frac{\omega_n}{\omega_n} \cdot \frac{K}{\frac{1}{\omega_n} s + 1} = \frac{K}{T_p s + 1} \Rightarrow T_p = \frac{1}{\omega_n}, \tag{3.10}$$

where

K is system gain

ω_n is a natural frequency

T_p is a time constant

s is a Laplace operator

A formula for rough estimation of natural frequency needed to achieve specified settling time may be found in [12] and its inference is rewritten in (3.11) with steady state defined when the measured value settles within 5% from reference. Also inferred in (3.11) is a formula used for computation of time constant T_p.

$$e^{-\zeta \omega_n t_s} = 0.05 \Rightarrow \zeta \omega_n t_s \cong 3 \Rightarrow \omega_n = \frac{3}{\zeta t_s} \Rightarrow T_p = \frac{\zeta t_s}{3}, \tag{3.11}$$

where

ζ is a damping ratio

t_s is a settling time

As the equation (3.11) is only a rough estimation, the result obtained was slightly modified and the final transfer function of system, which best approximates the required system dynamics, was selected as (3.12) (As we don't want the system to be oscillatory, the damping ratio was set $\zeta = 1$).

$$G(s) = \frac{K}{\frac{t_s}{2.5}s + 1} = \frac{\overline{p}_j}{\frac{t_j^{SU}}{2.5}s + 1} \tag{3.12}$$

This transfer function is rewritten as differential equation in (3.13) and finally discretized using Zero Order Hold method (ZOH) to obtain the final difference equation.

$$\begin{gathered} \dot{p}_j(t) = Ap_j(t) + Bu_j(t) = \frac{2.5}{t_j^{SU}} p_j(t) + \frac{2.5\overline{p}_j}{t_j^{SU}} u_j(t) \\ \downarrow \text{ZOH} \\ p_j(k+1) = Mp_j(k) + Nu_j(k) \end{gathered} \tag{3.13}$$

where

A,B are constants describing continuous system dynamics

M,N are constants describing discrete system dynamics computed using ZOH

In figure 3.2, the FO model is compared to required dynamics. To illustrate the error which is a result of using First order dynamics model, a three services UC example is shown in figure 3.3. The real output of First order dynamics model is marked *1st order dynamics*. The output interpreted as if the schedule resulting from optimization concluded with FO model was schedule of CR model is marked *interpreted*.

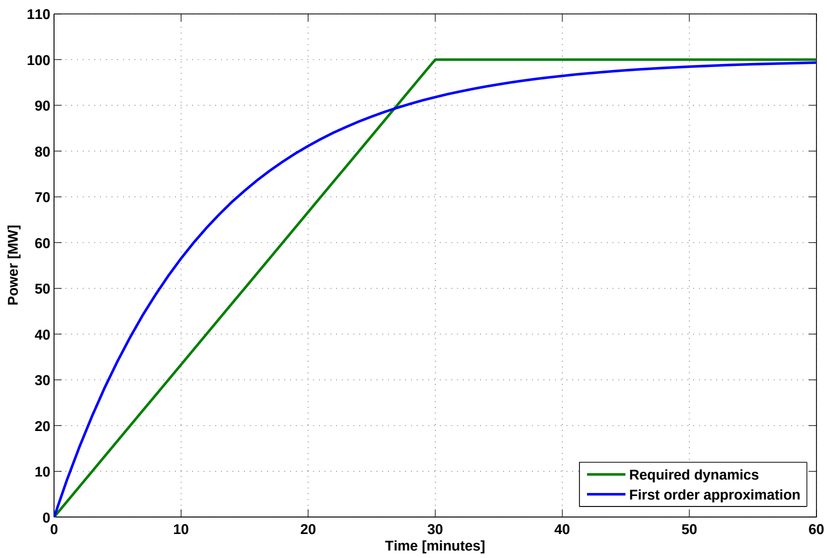

Figure 3.2: Comparison of First order dynamics model with required dynamics

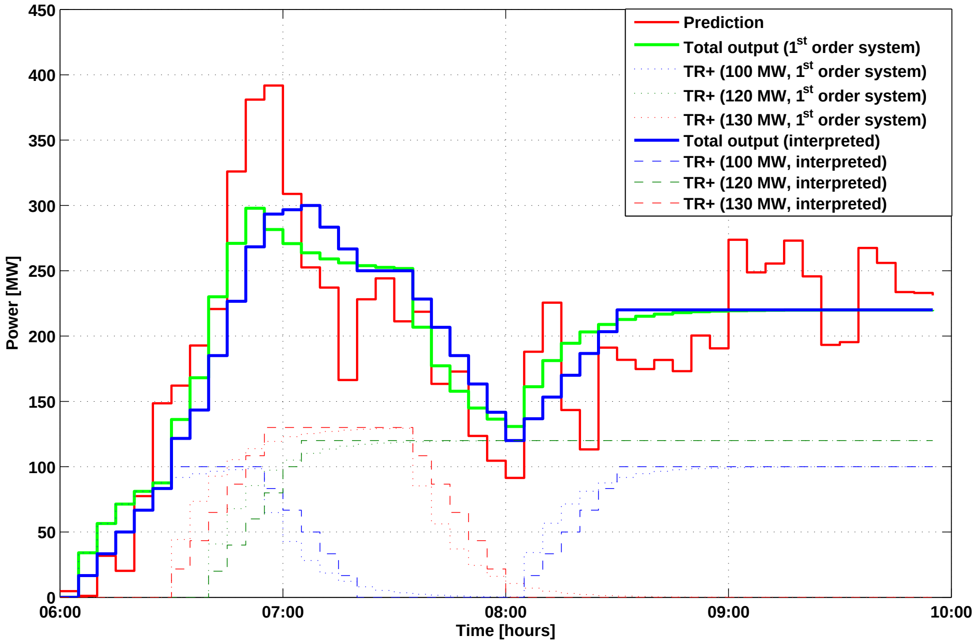

Figure 3.3: Example of three services UC for illustration of quality of First order dynamics model approximation of required dynamics

It may be concluded, that the resulting error is quite small compared to the optimization speedup which the usage of First order dynamics model brings. Note, the quality of approximation is retained as long as the service is mostly scheduled to be activated for at least its startup time t_j^{SU}. (i. e. it is not deactivated during startup). Such style of activation is, however, required for all included On/Off services, which justifies the usage of FO model.

3.2.5 Three state model

The 3S model uses constant ramp rates as specified in section 3.2.1. Its difference equation is the same as (3.6) with the only difference that the input may only take values defined by (3.14). This model isn't suitable for On/Off services as it is difficult to forbid the state, when the service remains at the level between its maximal and minimal power.

$$u_j\left(k\right)=\begin{cases}1 \text{ if the service is starting up with rate } \mathrm{r_j^{\Delta p}}\\ 0 \text{ if the service is not changing power}\\ 1 \text{ if the service is shutting down with rate } \mathrm{r_j^{\Delta p}}\end{cases} \tag{3.14}$$

3.2.6 Hourly model

This model was created for modeling Ereg and EregZ services. It is defined by (3.15): The power may be set to any level between maximal and minimal power on changes of business intervals and remains the same during the following business interval.

$$p_j\left(k+1\right)=\begin{cases}u_j\left(k\right)\cdot\overline{p}_j \text{ if } k \text{ - th sampling interval is at whole hour}\\ p_j\left(k\right) \text{ otherwise}\end{cases}$$
$$0\leq u_j\left(k\right)\leq 1 \tag{3.15}$$

3.3 Formulation as Mixed Integer Linear Program

The general linear optimization problem is defined as (3.16).

$$\begin{aligned}&\min_{\mathbf{x}}\mathbf{Cx} && \text{(optimality criterion)}\\ &\text{subject to}\\ &\qquad \mathbf{Ax}\leq\mathbf{b}, && \text{(inequality constraints)}\\ &\qquad \mathbf{A_{eq}x}=\mathbf{b_{eq}}, && \text{(equality constraints)}\\ &\text{where}\\ &\qquad \mathbf{x} \text{ is a vector consisting of integer or continuous optimization variables,}\\ &\qquad \mathbf{C} \text{ is a matrix defining costs in optimality criterion,}\\ &\qquad \mathbf{A},\mathbf{A_{eq}} \text{ are matrices defining left hand sides of constraints,}\\ &\qquad \mathbf{b},\mathbf{b_{eq}} \text{ are vectors defining right hand sides of constraints.}\end{aligned} \tag{3.16}$$

The objective function and dynamics as they were generally formulated in section 3.1 and section 3.2 were implicit formulations and contained some nonlinearities which cannot be used directly with MILP. There are, however, ways to find linear equivalents of nonlinear parts to obtain explicit formulation which is only linear combination of optimization variables. Basic techniques are described in [13] and useful advices on how to implement various features of controllers are presented in [14].

Table 3.2 shows overview of optimization and auxiliary variables used in scheduling algorithm along with their length and type. The lengths are given for one service, if more services are included in algorithm, matrices with size *length* $\times N_{sr}$ are used for each variable.

The following sections will present explicit formulations of all elements which are part of OC as defined in section 3.1.

Variable	**Length**	**Type**
$u_j(k)$	N_s[a]	binary, integer or continuous (depending on model)
$\Delta u_j(k)$	$N_s - 1$	integer
$\Delta u_j^{on}(k)$	$N_s - 1$	binary
$\Delta u_j^{off}(k)$	$N_s - 1$	binary
$p_j(k)$	N_s	continuous
$\overline{sat}_j^p(k)$	N_s	binary
$\underline{sat}_j^p(k)$	N_s	binary
$\overline{usat}_j^p(k)$	N_s	binary
$\underline{usat}_j^p(k)$	N_s	binary
$t_j^{on}(k)$	N_s	integer
$t_j^{off}(k)$	N_s	integer
$ut_j^{on}(k)$	N_s	integer
$ut_j^{off}(k)$	N_s	integer
$pen_j^{SU}(t_j^{off}(k))$	N_s	continuous
$pen_j^{SD}(t_j^{on}(k))$	N_s	continuous
$\Delta upen_j^{SU}(t_j^{off}(k))$	$N_s - 1$	continuous
$\Delta upen_j^{SD}(t_j^{on}(k))$	$N_s - 1$	continuous
$pen_j^{NSU}(n_j^{SU})$	1	continuous

Table 3.2: Variables overview

[a]The indexes of inputs are $0, \ldots, N_{sm} - 1$, $u_j(0)$ is the last input preceding the optimization interval.

3.3.1 Energy costs

This part of OC (3.1) is linear dependent on service power $p_j(k)$, therefore it may be used directly with MILP. The service power explicit formulation will be given in section 3.3.6.

3.3.2 Startup costs

The startup costs are linearly dependent on number of service startups n_j^{SU}. To obtain the number of startups, $\Delta u_j(k)$ and $\Delta u_j^{on}(k)$ must first be formulated[5]. The former is defined by equation (3.17) and the latter by equation (3.18). Finally, the n_j^{SU} is just a sum of $\Delta u_j^{on}(k)$ as stated in (3.19).

The equation (3.18) is a sufficient condition to obtain $\Delta u_j^{on}(k)$ because $\Delta u_j^{on}(k)$ is a binary variable and therefore anytime the $\Delta u_j(k)$ is 1 (i. e. the service was started up in k-th sampling interval) the $\Delta u_j^{on}(k)$ must be 1 as well. Moreover, $\Delta u_j^{on}(k)$ is a part of OC through n_j^{SU}, therefore it should be minimized which ensures that $\Delta u_j^{on}(k)$ is 0 otherwise.

$$\Delta u_j\left(k\right) = u_j\left(k\right) - u_j\left(k-1\right), k = 1, \dots, N_{sm} - 1 \tag{3.17}$$

$$\Delta u_j^{on}\left(k\right) \geq \Delta u_j\left(k\right), k = 1, \dots, N_{sm} - 1 \tag{3.18}$$

$$N_j^{SU} = \sum_{k=1}^{N_{sm}-1} \Delta u_j^{on}\left(k\right) \tag{3.19}$$

3.3.3 System Deviation penalization

This part of OC contains no nonlinearities, so it can be again implemented in MILP without any changes. As the penalization function may be piecewise linear (as long as it is convex) to approximate more complex functions, the implementation of piecewise linear function according to [13] is reproduced in (3.20). The former linear cost function $\mathbf{Cx}$ is replaced with auxiliary variable z. The variable z is then constrained such that it is greater or equal than every part of piecewise linear cost function. As the variable z is minimized during optimization, the minimum is therefore right on the piecewise linear function.

[5]These formulations are only valid for On/Off services.

$$\begin{aligned}
&\min_{z,\mathbf{x}} z \\
&\text{subject to} \\
&\qquad z \geq c_i'\mathbf{x} + d_i, \quad i = 1, \dots, m, \\
&\qquad \mathbf{A}\mathbf{x} \leq \mathbf{b}, \\
&\qquad \mathbf{A}_{eq}\mathbf{x} = \mathbf{b}_{eq}, \\
&\text{where} \\
&\qquad z \text{ is an auxiliary variable representing the new cost based on} \\
&\qquad\quad \text{piecewise linear function,} \\
&\qquad c_i',d \text{ is are coefficients defining a part of piecewise linear function,} \\
&\qquad m \text{ is number of parts of piecewise linear function,} \\
&\qquad \mathbf{A},\mathbf{A_{eq}},\mathbf{b},\mathbf{b_{eq}} \text{ are matrices and vectors representing the former problem.}
\end{aligned} \tag{3.20}$$

3.3.4 Penalization for early startups and shutdowns

The penalization for early startups is computed as stated in (3.2). This equation contains two nonlinearities - first is included in power off time counter $t_j^{off}(k)$, the second is the product of piecewise linear penalization dependent on time spent offline $pen_j^{SU}(t_j^{off}(k))$ and startup indicator $\Delta u_j^{on}(k)$. The implicit formulation of $t_j^{off}(k)$ is given in (3.21).

$$\begin{aligned}
&t_j^{off}(k+1) = (1 - u_j(k)) \cdot \left(t_j^{off}(k) + 1\right) = \\
&= t_j^{off}(k) - u_j(k) \cdot t_j^{off}(k) - u_j(k) + 1, \quad k = 1, \dots, N_{sm} - 1, \\
&t_j^{off}(1) \text{ contains the time the unit was offline prior to the start} \\
&\qquad \text{of the optimization interval.}
\end{aligned} \tag{3.21}$$

To obtain explicit formulation, the product of integer variable $t_j^{off}(k)$ and binary variable $u_j(k)$ must be defined. The general way how to formulate such products is described in [14] and the particular product of aforementioned variables is given by (3.22).

$$\left.\begin{aligned}
&ut_j^{off}(k) \leq \Big(N_{sm} + t_j^{off}(1)\Big) \cdot u_j(k), \\
&ut_j^{off}(k) \geq 0, \\
&ut_j^{off}(k) \leq t_j^{off}(k), \\
&ut_j^{off}(k) \geq t_j^{off}(k) - \Big(N_{sm} + t_j^{off}(1)\Big) \cdot (1 - u_j(k)).
\end{aligned}\right\} k = 1, \ldots, N_{sm} \tag{3.22}$$

The final explicit formulation of $t_j^{off}(k)$ is stated in (3.23).

$$t_j^{off}(k+1) = t_j^{off}(k) - ut_j^{off}(k) - u_j(k) + 1, \quad k = 1, \ldots, N_{sm} - 1 \tag{3.23}$$

The product of $pen_j^{SU}(t_j^{off}(k))$ and $\Delta u_j^{on}(k)$ is formulated in

$$\left.\begin{aligned}
&\Delta upen_j^{SU}\Big(t_j^{off}(k)\Big) \leq \overline{pen}_j^{SU} \cdot \Delta u_j^{on}(k), \\
&\Delta upen_j^{SU}\Big(t_j^{off}(k)\Big) \geq 0, \\
&\Delta upen_j^{SU}\Big(t_j^{off}(k)\Big) \leq pen_j^{SU}\Big(t_j^{off}(k)\Big), \\
&\Delta upen_j^{SU}\Big(t_j^{off}(k)\Big) \geq pen_j^{SU}\Big(t_j^{off}(k)\Big) - \overline{pen}_j^{SU} \cdot \big(1 - \Delta u_j^{on}(k)\big)
\end{aligned}\right\} * \tag{3.24}$$

$$*k = 1, \ldots, N_{sm} - 1$$

Finally, the total penalization for early startups is explicitly defined by equation (3.25).

$$PEN_j^{SU} = \sum_k \Delta upen_j^{SU}\Big(t_j^{off}(k)\Big), \quad k = 1, \ldots, N_{sm} - 1 \tag{3.25}$$

The penalization for early shutdowns is formulated analogously by substituting following in preceding equations:

- $t_j^{off}(k)$ for $t_j^{on}(k)$ and $(1 - u_j(k))$ for $u_j(k)$ in (3.21),
- $t_j^{off}(k)$ for $t_j^{on}(k)$ and $ut_j^{off}(k)$ for $ut_j^{on}(k)$ in (3.22),
- $\Delta upen_j^{SU}(t_j^{off}(k))$ for $\Delta upen_j^{SD}(t_j^{on}(k))$, $pen_j^{SU}(t_j^{off}(k))$ for $pen_j^{SD}(t_j^{on}(k))$, $\overline{pen}_j^{SU}$ for $\overline{pen}_j^{SD}$ and $\Delta u_j^{on}(k)$ for $\Delta u_j^{off}(k)$ in (3.24),

and by reformulating (3.23) to (3.26) and (3.25) to (3.27)

$$t_j^{on}(k+1) = ut_j^{on}(k) + u_j(k), \quad k = 1, \ldots, N_{sm} - 1 \tag{3.26}$$

$$PEN_j^{SD} = \sum_k \Delta upen_j^{SD}\left(t_j^{on}(k)\right), \quad k = 1, \dots, N_{sm} - 1 \tag{3.27}$$

3.3.5 Penalization for number of startups

The penalization for number of startups as formulated in (3.4) is a piecewise linear function of N_j^{SU}; N_j^{SU} was explicitly formulated is section 3.3.2 and piecewise linear function was described in section 3.3.3.

3.3.6 Dynamics models

In this section, the explicit formulations of dynamics models that were described in section 3.2 will be given. The OC remains the same for all models, the constraints, however, are different. It should also be noted, that in most cases, the difference equations used for dynamics models description in section 3.2 are directly applicable with Yalmip, in this case only references to equations in section 3.2 will be provided.

3.3.6.1 Model with variable ramp rates

Variable	**Length**	**Type**	**Description**
$u_j(k)$	N_s	continuous	input of j-th service in k-th sampling interval
$p_j(k)$	N_s	continuous	power of j-th service in k-th sampling interval

Table 3.3: Variables used in model with variable ramp rates

The constraints associated with VR model are:

- *The dynamics difference equation* (3.6),
- *Input limits*,
- *Output limits*, both also included in (3.6).

3.3.6.2 Model with constant ramp rates

Variable	Length	Type	Description
$u_j(k)$	N_s	binary	input of j-th service in k-th sampling interval
$p_j(k)$	N_s	continuous	power of j-th service in k-th sampling interval
$\overline{sat}_j^p(k)$	N_s	binary	maximal power saturation indicator
$\underline{sat}_j^p(k)$	N_s	binary	minimal power saturation indicator
$\overline{usat}_j^p(k)$	N_s	binary	product of $u_j(k)$ and $\overline{sat}_j^p(k)$
$\underline{usat}_j^p(k)$	N_s	binary	product of $u_j(k)$ and $\overline{sat}_j^p(k)$

Table 3.4: Variables used in model with constant ramp rates

In order to obtain explicit formulation of CR model (expanded version of dynamics (3.7) is given in (3.28)), the saturation indicators must first be defined and the products of saturation indicators and input must be formulated.

$$\begin{aligned} p_j\left(k+1\right) &= p_j\left(k\right) + u_j\left(k\right)\cdot\left(1-\overline{sat}_j^p\left(k\right)\right)\cdot r_j^{\Delta p} - \left(1-u_j\left(k\right)\right)\cdot\left(1-\underline{sat}_j^p\left(k\right)\right)\cdot r_j^{\Delta p} = \\ &= p_j\left(k\right) + \left(u_j\left(k\right) - u_j\left(k\right)\cdot\overline{sat}_j^p\left(k\right)\right)\cdot r_j^{\Delta p} - \\ &- \left(1-u_j\left(k\right)-\underline{sat}_j^p\left(k\right)+u_j\left(k\right)\cdot\underline{sat}_j^p\left(k\right)\right)\cdot r_j^{\Delta p}, \quad k=1,\ldots,N_{sm} \end{aligned} \tag{3.28}$$

As Yalmip has built-in *implies* operator, the specification of saturation indicators as stated in (3.7) may be directly implemented as shown in (3.29) or using a method described in [14].

$$\left.\begin{aligned} p_j\left(k\right) \geq \overline{p}_j &\Rightarrow \overline{sat}_j\left(k\right) = 1, \\ p_j\left(k\right) < \overline{p}_j &\Rightarrow \overline{sat}_j\left(k\right) = 0, \\ p_j\left(k\right) \leq \underline{p}_j &\Rightarrow \underline{sat}_j\left(k\right) = 1, \\ p_j\left(k\right) > \underline{p}_j &\Rightarrow \underline{sat}_j\left(k\right) = 0. \end{aligned}\right\} k=1,\ldots,N_{sm} \tag{3.29}$$

The product of two binary variables, input $u_j(k)$ and maximal power saturation indicator $\overline{sat}_j^p(k)$ may be, according to [14], expressed as (3.30). The product of input $u_j(k)$ and minimal power saturation indicator $\underline{sat}_j^p(k)$ is formed by substituting $\overline{sat}_j^p(k)$ for $\underline{sat}_j^p(k)$ and $\overline{usat}_j^p(k)$ for $\underline{usat}_j^p(k)$ in equation (3.30).

$$\left.\begin{aligned} -\overline{sat}_j^p(k) + \overline{usat}_j^p(k) \le 0, \\ -u_j(k) + \overline{usat}_j^p(k) \le 0, \\ u_j(k) + \overline{sat}_j^p(k) + \overline{usat}_j^p(k) \le 1. \end{aligned}\right\} k = 1, \ldots, N_{sm} \tag{3.30}$$

The final explicit formulation of CR model is shown in equation (3.31).

$$\begin{aligned} p_j(k+1) = p_j(k) + \left(u_j(k) - \overline{usat}_j^p(k)\right) \cdot r_j^{\Delta p} - \\ - \left(1 - u_j(k) - \underline{sat}_j^p(k) + \underline{usat}_j^p(k)\right) \cdot r_j^{\Delta p}, \quad k = 1, \ldots, N_{sm} \end{aligned} \tag{3.31}$$

The constraints associated with CR model are:

- *The dynamics difference equation* (3.31),
- *Input limits* are completely defined by constraining the inputs to be binary,
- *Saturations indicators* defined by (3.29),
- *Products of input and saturation indicators* defined by (3.30).

3.3.6.3 Model based on first order system

Variable	Length	Type	Description
$u_j(k)$	N_s	binary	input of j-th service in k-th sampling interval
$p_j(k)$	N_s	continuous	power of j-th service in k-th sampling interval

Table 3.5: Variables used in model based on first order system

The constraints associated with FO model are:

- *The dynamics difference equation* (3.13),
- *Input limits* are completely defined by constraining the inputs to be binary.

3.3.6.4 Three state model

Variable	Length	Type	Description
$u_j(k)$	N_s	integer	input of j-th service in k-th sampling interval
$p_j(k)$	N_s	continuous	power of j-th service in k-th sampling interval

Table 3.6: Variables used in three state model

The constraints associated with 3S model are:

- *The dynamics difference equation* (3.6),
- *Input limits* defined by (3.14),
- *Output limits* defined by (3.6).

3.3.6.5 Hourly model

Variable	Length	Type	Description
$u_j(k)$	N_s	continuous	input of j-th service in k-th sampling interval
$p_j(k)$	N_s	continuous	power of j-th service in k-th sampling interval

Table 3.7: Variables used in hourly model

The formulation given in (3.15) may be implemented directly or in matrix form (3.32) (the equation (3.32) is an example of system dynamics during two trading hours). The inputs during trading hours remain unused and have no effect on service output.

The constraints associated with hourly model are:

- *The dynamics difference equation* (3.15) or matrix form (3.32),
- *Input limits* defined by (3.15).

$$\mathbf{p_j} = \mathbf{S} \cdot \mathbf{u} = \begin{bmatrix} \overline{p}_j & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \overline{p}_j & 0 & 0 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & \overline{p}_j & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \cdots & 0 & \overline{p}_j & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ \vdots \\ u_{N_{sm-1}} \end{bmatrix} \tag{3.32}$$

where

$$\mathbf{p_j} = \begin{bmatrix} p_1 \\ \vdots \\ p_{N_{sm}} \end{bmatrix}$$ is a vector of j - th service power in each sampling interval,

$\mathbf{S}$ is $N_{sm} \times N_{sm-1}$ a matrix defining system reaction to inputs,

$\mathbf{u}$ is a vector of inputs.

3.3.7 Other constraints

3.3.7.1 Startup delay

The startup delay may be included by directly altering the difference equation as shown in equation (3.33) or when transformed into matrix form, equation (3.34) may be applied. In matrix form, including startup delay involves simply shifting the elements of the original matrix[6] down by D_j^{SU} rows, while the matrix retains its original dimensions, therefore last D_j^{SU} rows of original matrix are left out.

$$p_j\left(k + D_j^{SU} + 1\right) = p_j\left(k + D_j^{SU}\right) + S \cdot u_j(k), \quad k = 1, \ldots, N_{sm} - D_j^{SU} - 1, \tag{3.33}$$

where

S is a constant defining service reaction to inputs.

$$\mathbf{p_j} = p_j(1) + \mathbf{S} \cdot \mathbf{u} = = p_j(1) + \begin{bmatrix} 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ D_j^{SU} \left\{ \vdots \right. & & & & & \vdots \\ 0 & & & & & \vdots \\ r_j^{\Delta p} & \ddots & & & & \vdots \\ \vdots & \ddots & \ddots & & & \vdots \\ r_j^{\Delta p} & \cdots & r_j^{\Delta p} & 0 & \cdots & 0 \end{bmatrix} \cdot \begin{bmatrix} u_1 \\ \vdots \\ u_{N_{sm-1}} \end{bmatrix} \tag{3.34}$$

where

$\mathbf{p_j} = \begin{bmatrix} p_1 \\ \vdots \\ p_{N_{sm}} \end{bmatrix}$ is a vector of j - th service power in each sampling interval,

$p_j(1)$ is an initial power of j - th service,

$\mathbf{S}$ is a $N_{sm} \times N_{sm-1}$ matrix defining system reaction to inputs,

$\mathbf{u}$ is a vector of inputs.

3.3.7.2 Minimal on/off time

Apart from Early shutdown penalization and Early startup penalization, the constraint which directly sets the minimal time the service must remain online (offline) before de-

[6] The original matrix is lower triangle matrix shifted one row down to reflect the fact, that input at k-th sampling interval affects the output in next sampling interval.

activation (activation) may be included. It uses fewer auxiliary variables than Early shutdown and startup penalizations, therefore should be less computationally intensive. It should be noted, however, that setting this strict constraint leads to feasibility problems when using freely available solvers as it dramatically decreases the amount of feasible schedules.

The Minimal on/off time constraint is implemented using Off time counter $t_j^{off}(k)$ and On time counter $t_j^{on}(k)$ as defined by equation (3.35).

$$-\frac{t_{on}}{T_s}t_j^{on}\left(k\right) \leq \Delta u_j\left(k\right) \leq \frac{t_{off}}{T_s}t_j^{off}\left(k\right), \quad k = 1, \ldots, N_{sm} - 1 \tag{3.35}$$

3.3.7.3 Quick-Start 10 minute reserve maximal energy

Due to limited capacity of water reservoirs of pumped storage plants providing QS, the maximal energy constraint must be included in algorithm, defined by (3.36).

$$\sum_k p_j\left(k\right) \leq \overline{E}_j, \quad k = 1, \ldots, N_{sm} \tag{3.36}$$

Chapter 4

Results

This chapter presents results achieved using proposed scheduling algorithm. As the resulting optimization problem is generally large-scale, the efficient commercial solver CPLEX [15] was used for all measurements. Freely available solvers exhibited severe problems in finding the optimal solution, probably mostly because they lack heuristics the CPLEX has.

4.1 Comparison of dynamics models performance

To perform comparison of three basic dynamics[1] models in terms of performance, the test case with 6 services was set up in following way:

- 6 units of TR type are included (30 minutes startup time, no startup delay, on/off operation, energy price averaging around 1500 CZK/MWh),
- the optimization was performed over 6 hour optimization interval,
- the SD penalization was set to 10000 CZK/MWh,
- to restrict excessive activation, penalization for number of startups during the optimization interval is used - the first startup is not penalized, the others are penalized by 100000 CZK/startup.

[1]The CR, FO and 3S models were selected as basic as they may be used to model TR and DZ, which present the greatest part of optimization problem as each generating unit providing TR or DZ is included separately

- the sampling period was changed from 2 minutes up to 30 minutes, effectively changing the problem size in terms of variables and constraints,
- the optimization was stopped as soon as the solution was 5 percent from optimal solution or after 5 minutes. The resulting optimality gap of the obtained solution from optimum is recorded in result tables along with the optimization time.

The tables 4.1 - 4.5 show the test results for each sampling time that was tested, while the table 4.6 compares the optimization times.

	FO model	**3S model**	**CR model**
Binary variables	1074	1074	5370
Continuous variables	3402	3402	3402
Constraints	8952	8964	30444
Regulation energy costs [th. CZK]	1703.9	1681.2	N/A
Deviation penalty [th. CZK]	2602.9	2121.1	N/A
Total costs [th. CZK]	4306.9	3802.4	N/A
Resulting optimality gap [%]	5	8.84	N/A
Execution time [s]	76.2	300	N/A

Table 4.1: Test case with 6 services (Sampling interval: 2 minutes)

	FO model	**3S model**	**CR model**
Binary variables	426	426	2130
Continuous variables	1350	1350	1350
Constraints	3552	3564	12084
Regulation energy costs [th. CZK]	1726.1	1708.3	1728.8
Deviation penalty [th. CZK]	2198.8	1883.3	2616.9
Total costs [th. CZK]	3925	3591.5	4345.8
Resulting optimality gap [%]	5	7.54	27.73
Execution time [s]	13.31	300	300

Table 4.2: Test case with 6 services (Sampling interval: 5 minutes)

	FO model	3S model	CR model
Binary variables	210	210	1050
Continuous variables	666	666	666
Constraints	1752	1764	5964
Regulation energy costs [th. CZK]	1694.2	1623.9	1595.5
Deviation penalty [th. CZK]	1891.4	1589.8	2009.8
Total costs [th. CZK]	3585.6	3213.7	3605.3
Resulting optimality gap [%]	5	5	19.92
Execution time [s]	9.4	196	300

Table 4.3: Test case with 6 services (Sampling interval: 10 minutes)

	FO model	3S model	CR model
Binary variables	138	138	690
Continuous variables	438	438	438
Constraints	1152	1164	3924
Regulation energy costs [th. CZK]	1704.5	1715.7	1572.1
Deviation penalty [th. CZK]	1567.9	1219.4	1694.0
Total costs [th. CZK]	3272.4	2935.1	3266.1
Resulting optimality gap [%]	5	5	15
Execution time [s]	10.1	207.6	300

Table 4.4: Test case with 6 services (Sampling interval: 15 minutes)

	FO model	3S model	CR model
Binary variables	66	66	330
Continuous variables	210	210	210
Constraints	552	564	1884
Regulation energy costs [th. CZK]	1712.5	1791	1668.2
Deviation penalty [th. CZK]	1101.7	798.1	1021.3
Total costs [th. CZK]	2814.2	2589.1	2689.5
Resulting optimality gap [%]	5	5	5
Execution time [s]	5	36.9	37.5

Table 4.5: Test case with 6 services (Sampling interval: 30 minutes)

	FO model		3S model		CR model	
T_s **[minutes]**	**Time [s]**	**Gap [%]**	**Time [s]**	**Gap [%]**	**Time [s]**	**Gap [%]**
2	76.2	5	300	8.84	N/A	N/A
5	13.3	5	300	7.54	300	27.73
10	9.4	5	196	5	300	19.92
15	10.1	5	207.6	5	300	15
30	5	5	36.9	5	37.5	5

Table 4.6: Performance comparison of the test case with 6 services

As the table 4.6 shows, FO is by far the best performing model in terms of optimization speed. This is interesting since the problem size is approximately on par with the 3S model. It may therefore be concluded, that the model performance is not dependent just on the problem size (the number of variables and constraints) but it also depends on the problem structure. The model based on first order approximations profits from being linear and simple. The CR model failed to produce any solution within 5 minutes for 2 minutes sampling period and in other cases was very inefficient. This may be attributed to high number of binary variables, overall larger size and structure with many nonlinearities.

When comparing the total costs (OC value) achieved by models, the 3S model generates the schedules with the lowest total costs, but this is mainly because it doesn't correctly model On/Off services and resulting schedules contain states when the service power remain on the level between the minimal and maximal power. This advantage allows for more precise SDP compensation. The higher costs produced by FO model may result from the fact that it is only an approximation of required dynamics and the total costs were computed as if all models had precisely the required dynamics (see comparison in section 3.2.4).

4.2 Influence of penalizations on services activation and optimization speed

This section documents how the implemented penalizations influence the activation schedule and how height of penalizations impact optimization speed. All tests were performed in following way:

- the first order approximation model was used,
- the constant ramp rate interpretation of FO model schedules wasn't used, because it is not precise in schedules with excessive services activation,
- only one service with TR characteristics was included (maximal power 100 MW, startup time 30 minutes, no startup delay, energy price 1500 CZK/MWh),
- SD penalization was set to 10000 CZK/MWh,
- the search was stopped as soon as the solution was 0,5 percent from optimal solution or after 5 minutes.

4.2.1 Penalization for number of startups

The figure 4.1 shows that the penalization for number of startups is able to effectively control the frequency of services activation[2].

The performance of optimization increases with increasing penalization as shown in table 4.8 - As the penalization increases, it represents a larger portion of the final criterion value; The equilibrium between the penalization for number of startups and SD penalization is then more clear as another startup may decrease the deviation from SDP and its penalization, but will bring much higher penalization for number of startups. On the other side, lowering the number of startups will decrease the respective penalization, but may bring much higher penalization for SDP. If the penalization for number of startups is lower, the equilibrium point isn't clearly defined and it is harder for optimizer to find the optimal balance between the two penalizations.

[2] Also considered as startup is the state, when the service is reactivated during shutdown process.

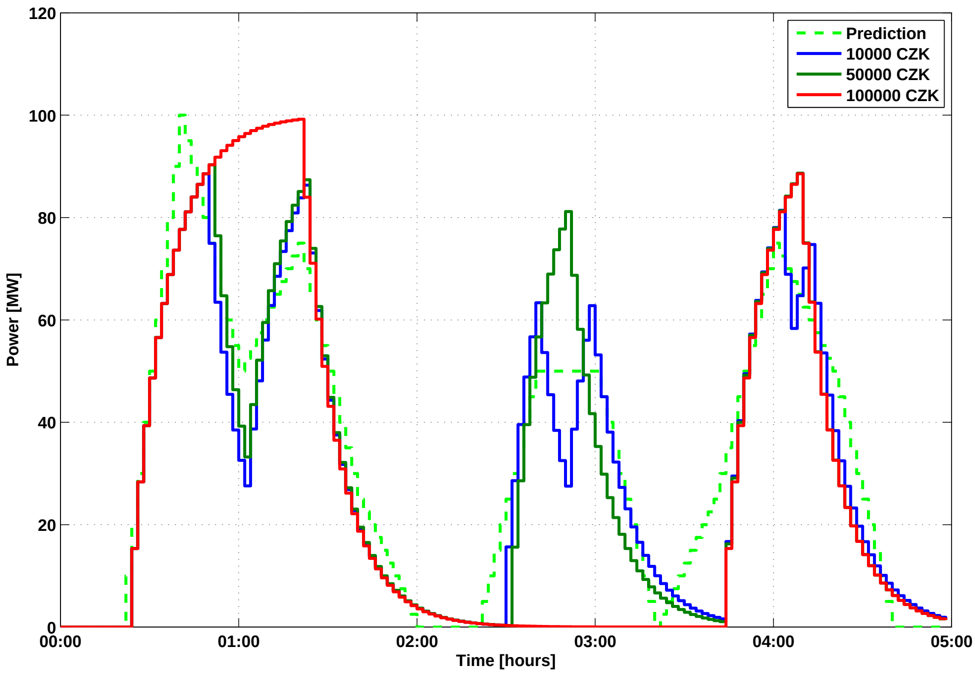

Figure 4.1: Service activation depending on penalization for number of startups

Penalization [CZK]	Time [s]
10000	300
50000	42.2
100000	10.2

Table 4.7: Performance depending on penalization for number of startups

4.2.2 Penalizations for early startups and shutdowns

The effect of penalizations for early startups and shutdowns on activation schedule is about the same as that of penalization for number of startups as may be seen in figure 4.2. The difference, though, is that the penalization for early startups or early shutdowns should prevent shutting the service down or starting the service up too early more reliably - the penalization for number of startups may allow early startup or shutdown if startup of the respective service is not needed anymore throughout the optimization interval.

The performance, when this penalization is used, is rather low as the penalization implementation contains many nonlinearities and requires a lot of auxiliary variables, many of which are binary. As may be seen in table 4.8, the optimization failed to finish in five minutes in all cases and the resulting optimality gap of obtained solution from the optimum is relatively high.

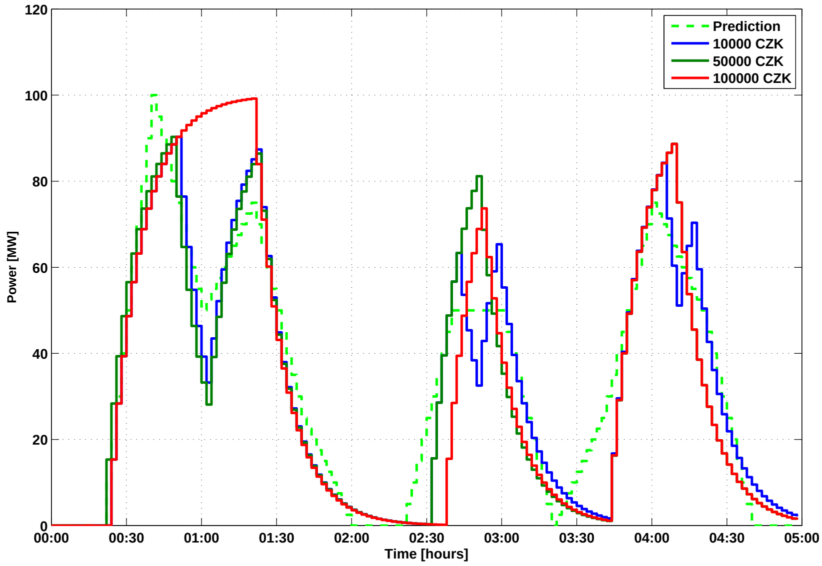

Figure 4.2: Service activation depending on penalization for early startups/shutdowns

Penalization [CZK]	Time [s]	Gap [%]
10000	300	21.2
50000	300	23.6
100000	300	28.9

Table 4.8: Performance depending on penalization for early startups/shutdowns

4.2.3 Performance depending on minimal on/off time constraints

The minimal on/off time constraints are another way to influence style of services activation. It sets the minimal amount of time the service must remain online or offline.

The figure 4.3 shows that these constraints are correctly implemented and the requirements on minimal on or off time are met.

Performance-wise, the optimization reached the desired optimality gap quickly with CPLEX (see table 4.9), though with freely available solvers, this kind of constraint brings problems with finding any feasible solution.

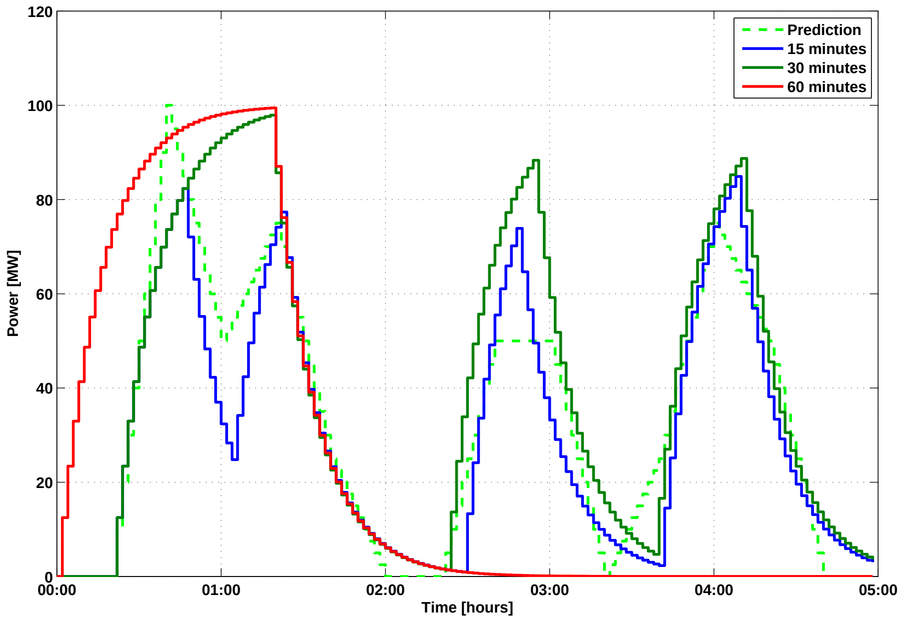

Figure 4.3: Service activation depending on minimal on/off time

Min On/Off time [minutes]	Time [s]
15	20.8
30	13
60	7.8

Table 4.9: Performance depending on minimal on/off time

4.3 Real test case

Finally, to verify the algorithm performance, the eleven services test case was created, having following properties:

- eleven services were included as shown in table 4.10,
- the optimization was performed over 6 hours optimization interval,
- the sampling period was set to 10 minutes,
- price for deviance of scheduled output from SDP was set to 10000 CZK/MWh,
- the minimal on and off times constraints were used to control services activation in the first test (these were set to 60 minutes for TR, 90 minutes for DZ and none for QS (the times applies for both - the on and the off time) and startup penalizations in the second test (500000 CZK for second and following startups for TR and DZ, 20000 CZK in case of QS),
- the penalization for DZ startups was increased by 20000 CZK for every startup (including the first one),
- the search was stopped as soon as the solution was 0,5 percent from optimal solution or after 5 minutes.

The problem with minimal on and off time constraints consisted of 385 binary variables, 2154 continuous variables and 5676 constraints. The optimization was stopped after five minutes with resulting optimality gap of 26.8 percent, yielding a schedule shown in figure 4.4. The optimality gap is quite high, but it should be also noted, that the gap 30 percent was reached after two minutes, after which the solution shown only minor improvements. The resulting schedule is feasible, respects all constraints and covers the system deviation prediction quite well.

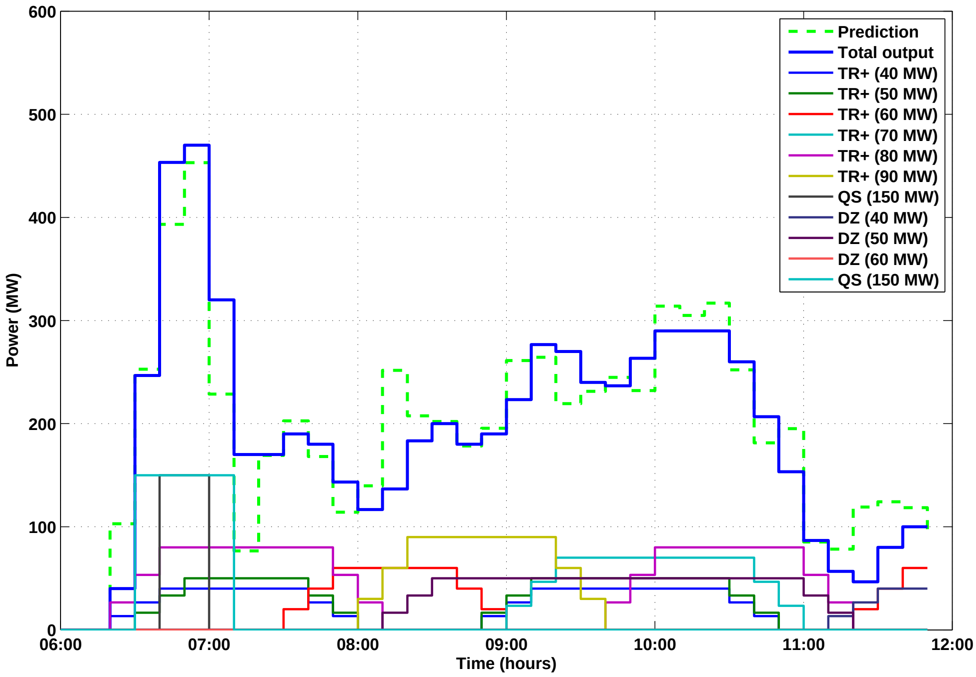

Figure 4.4: Eleven services real test case (minimal on and off time constraints)

With startup penalization used, the model consisted of 385 binary variables, 1191 continuous variables and 2526 constraints. After five minutes, optimality gap of 16 percent was reached, with resulting schedule as shown in figure 4.5. The only unwanted element in the schedule is the activation of TR (90 MW) for only 20 minutes at the beginning of optimization interval.

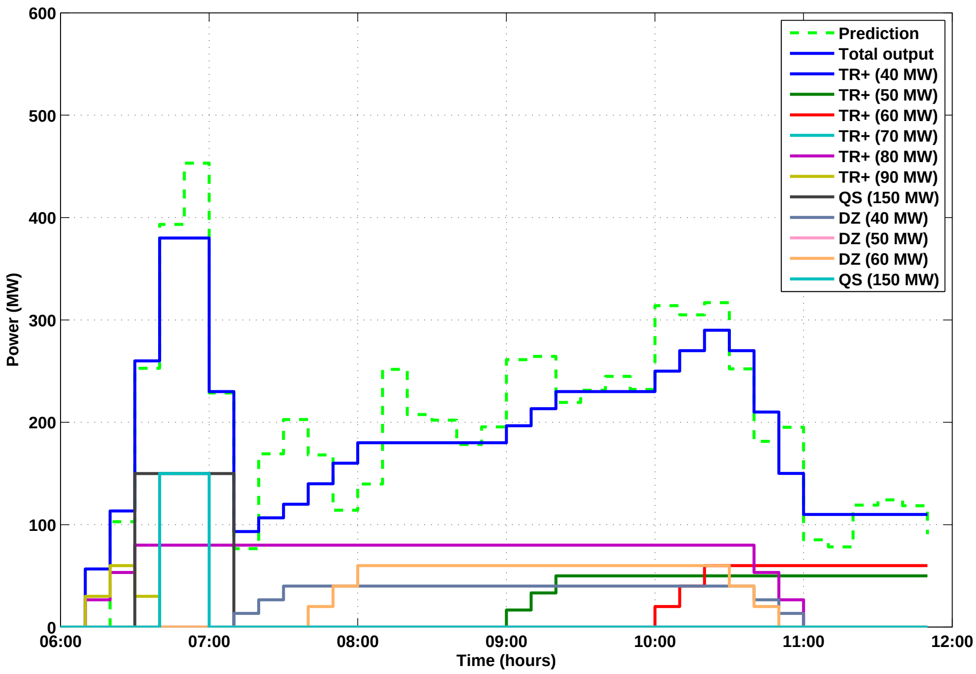

Figure 4.5: Eleven services real test case (startup penalizations)

Service type	$\overline{p}$ **[MW]**	c^p **[CZK/MWh]**	t_j^{SU} **[minutes]**	D_j^{SU} **[minutes]**
TR	40	1500	30	0
TR	50	1400	30	0
TR	60	1600	30	0
TR	70	1700	30	0
TR	80	1550	30	0
TR	90	1650	30	0
QS	150	3500	10	0
QS	150	3400	10	0
DZ	40	1700	30	60
DZ	50	1650	30	30
DZ	60	1750	30	90

Table 4.10: Eleven units test case services properties

Chapter 5

Conclusion

A formulation of algorithm which optimizes Ancillary Services utilization was presented with intention to provide a support decision tool for Transmission System Operator dispatchers. The proposed algorithm deals with properties specific for Ancillary Services such as rules associated with the way the services should be activated or on/off type of operation as well as with typical generating units properties as ramping limits or minimal on and off times.

Three basic models were developed to model different types of services behaviour - the model with constant ramp rates, the model based on first order system and the three state model. Their performance was tested in a test case, with the model based on first order system coming out as the best performing in terms of speed. The three state model doesn't currently meet the requirements of on/off service behaviour, but proved good performance in terms of both the speed and the quality of achieved solution. It will be therefore retained for possible future use. The model with constant ramp rates models the required dynamics precisely, but its performance is poor due to high number of constraints and binary variables.

Three ways to limit services activation were proposed - the penalization for number of startups, the penalization for early startups and shutdowns and constraints specifying minimal on and off times. Influence of these penalization on helping the schedules meet the services activation rules and limitations was successfully presented. The penalization for number of startups performed very well as concerns the optimization speed, although when used alone, it may occasionally produce schedules which don't comply with services activation rules. The constraints specifying the minimal on and off times also showed a solid optimization speed and ensured that the required minimal on and off times were correctly held in resulting schedules. Their drawback is that with larger problems, they may lead to feasibility problems. The utilization of penalizations for early startups and shutdowns also showed to be efficient in producing schedules which comply with services activation rules, but the optimization speed was very low due to the need of many auxiliary variables.

The efficiency of the proposed algorithm was shown on a real test case study, proving that in current state, it is able to produce feasible schedules which may serve the intended purpose. However, use of commercial solver is recommended for optimization as freely available solvers fail to optimize the problem in reasonable time.

In this thesis, the effort was put to create general framework for modeling properties, dynamics and limitations of Ancillary Services. The problem was formulated generally and in more detail than is needed for real utilization - the factors as quality and the sampling period of System Deviation prediction, and the services included in optimization will play important role in real application, probably resulting in substantial optimization speed increase.

References

[1] Baldick, R. The Generalized Unit Commitment. *IEEE Transactions on Power Systems*, 10, No.1:465–475, 1995.

[2] Restrepo, J. F. and Galiana, F. D. Unit Commitment With Primary Frequency Regulation Constraints. *IEEE Transactions on Power Systems*, 20, No.4:1836–1842, 2005.

[3] Arroyo, J. M. and Conejo, A. J. Optimal Response of a Thermal Unit to an Electricity Spot Market. *IEEE Transactions on Power Systems*, 15, No.3:1098–1104, 2000.

[4] Carrión, M and Arroyo, J. M. A Computationaly Efficient Mixed-Integet Linear Formulation for the Thermal Unit Commitment Problem. *IEEE Transactions on Power Systems*, 21, No.3:1371–1378, 2006.

[5] Sawa, T., Sato, Y., Tsurugai, M., and Onishi, T. Security Constrained Integrated Unit Commitment Using Quadratic Programming. In *Power Tech*, volume 165, Lausanne, Switzerland, 2007.

[6] Ting, T., Rao, M. V. C., Loo, C. K., and Ngu, S. S. Solving Unit Commitment Problem Using Hybrid Particle Swarm Optimization. *Journal of Heuristics*, 9:507–520, 2003.

[7] Georgopoulou, H. A. and Giannakoglou, K. C. A Two-Level Method for Solving Power Generating Unit Commitment Problems. In *Power Tech*, volume 238, Lausanne, Switzerland, 2007.

[8] Li, Z. and Shahidehpour, M. Security-Constrained Unit Commitment for Simultaneous Clearing of Energy and Ancillary Services Market. *IEEE Transactions on Power Systems*, 20, No.2:1079–1088, 2005.

[9] ČEPS, a.s. Praha. Kodex přenosové soustavy [online]. ⟨Revision 07, January 2007, http://www.ceps.cz/doc/kodex/revize07.zip⟩, 2007.

[10] J. Löfberg. Yalmip : A toolbox for modeling and optimization in MATLAB [online]. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004. ⟨http://control.ee.ethz.ch/ joloef/yalmip.php⟩.

[11] The MathWorks MATLAB Website, 2007, [online]. ⟨http://www.mathworks.com/products/matlab/⟩.

[12] Franklin, G. F., Powell, J. D., and Emami-Naeini, A. *Feedback Control of Dynamic Systems*. Pearson Prentice Hall, 2006.

[13] Bertsimas, D. and Tsitsiklis, J. N. *Introduction to Linear Optimization*. Athena Scientific, 1997.

[14] Bemporad, A. and Morari, M. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35:407–427, 1999.

[15] The ILOG CPLEX Website, 2007, [online]. ⟨http://ilog.com/products/cplex/⟩.

Appendix A

Contents of the included CD

Included in this thesis is a CD with following contents:

- Directory Thesis:
  - bt_predictive_power_balance_control.pdf - the electronic version of this Bachelor Thesis
- Directory Scripts:
  - average.m - a script for averaging datasets
  - generatePieceWiseCostFunction.m - a script for generating set of constraints to model piecewise cost functions
  - getTimeInterval.m - a script for selecting intervals from large data sets
  - interpretInput.m - a script for interpreting the first order model schedules
  - optimize.m - a script performing the actual optimization