Hierarchical model-based predictive control of a power plant portfolio

Edlund, Kristian Skjoldborg; Bendtsen, Jan Dimon; Jørgensen, John Bagterp

Published in:
Control Engineering Practice

DOI (link to publication from Publisher):
10.1016/j.conengprac.2011.06.002

Publication date:
2011

Document Version
Early version, also known as pre-print

Hierarchical model-based predictive control of a power plant portfolio

K. Edlund[a,*], J.D. Bendtsen[b], J.B. Jørgensen[c]

[a] *DONG Energy, Kraftværksvej 53, DK-7000 Fredericia, Denmark*
[b] *Section for Automation and Control, Aalborg University, Fredrik Bajers Vej 7, DK-9220 Aalborg SØ, Denmark*
[c] *DTU Informatics, Denmark Technical University, Building 321, office 017, DK-2800 Kgs Lyngby, Denmark*

Abstract

One of the main difficulties in large-scale implementation of renewable energy in exisiting power systems is that the production from renewable sources is difficult to predict and control. For this reason, fast and efficient control of controllable power producing units – so-called "portfolio control" – becomes increasingly important as the ratio of renewable energy in a power system grows. As a consequence, tomorrow's "smart grids" require highly flexible and scalable control systems compared to conventional power systems. This paper proposes a hierarchical model-based predictive control design for power system portfolio control, which aims specifically at meeting these demands.

The design involves a two-layer hierarchical structure with clearly defined interfaces that facilitate an object-oriented implementation approach. The same hierarchical structure is reflected in the underlying optimisation problem, which is solved using Dantzig-Wolfe decomposition. This decomposition yields improved computational efficiency and better scalability compared to centralised methods.

The proposed control scheme is compared to an existing, state-of-the-art portfolio control system (operated by DONG Energy in Western Denmark) via

*Corresponding author. Tel.:+45 9955 2963; Fax: +45 9955 0001
Email addresses: `kried@dongenergy.dk` (K. Edlund), `dimon@es.aau.dk` (J.D. Bendtsen), `jbj@imm.dtu.dk` (J.B. Jørgensen)

simulations on a real-world scenario. Despite limited tuning, the new controller shows improvements in terms of ability to track reference production as well as economic performance.

Keywords: Predictive control, Model-based control, Hierarchical control, Power systems control, Object-oriented modelling, Decoupled subsystems

---

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; Transport- og Energiministeriet, 2005; United Nations, 1998), dynamic control of power plants is becoming highly important. Indeed, the incentives for power companies 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), are stronger than ever.

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 the given power company's jurisdiction (its *portfolio*) for a specified period of time – typically from 24 hours up to a week. This 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 production plans are distributed to the generating units, where

local controllers track the plans while suppressing disturbances etc.

However, with the aforementioned increasing impact of short-term fluctuations in the supply and demand, dynamic effects at the system level are becoming increasingly inconvenient to deal with for individual generating units. Various approaches to deal with these difficulties have been presented in literature; Alvarado (2005) and Jokic (2007) deal with multiple area power system control through prices, where the network adds structure to the problem, while genetic algorithm-based (Ramakrishna and Bhatti, 2008) and fuzzy scheduling-based (Anower et al., 2006) solutions have been presented for single area problems.

Yet another 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 etc. – 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 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.

The objective of the proposed controller is to minimise deviations between sold and actual production. Furthermore, two main objectives are in focus in the design:

Scalability Future development 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 The controller must be *flexible*, such that addition of new units and maintenance of existing ones is possible. This means that the design must have a modular structure that supports *information encapsulation* and clear *communication interfaces* between the modules.

To meet these objectives, the problem is formulated as a *linear program* and solved using the socalled *Dantzig-Wolfe decomposition* (Dantzig and Wolfe, 1960; Lasdon, 2002; Dantzig and Thapa, 2002), which is a very efficient algorithm for solution of linear programs of the type considered here. Dantzig-Wolfe decomposition breaks a linear program into a number of independent subproblems and a Master Problem that coordinates the subproblems. The Master Problem sends a 'price' on a shared resource to each of the subproblems. Subject to this 'price', the optimal solution to each of the subproblems is individually computed and returned. This interchange of information continues until convergence. The Dantzig-Wolfe decomposition algorithm always converges in a finite number of iterations to the solution of the original linear program if a feasible solution exists (Dantzig and Thapa, 2002). In predictive control applications, this implies that stability can be guaranteed under mild conditions even if the algorithm has to be stopped prematurely to maintain a constant sample rate (Scokaert et al., 1999). That is, assuming the problem is feasible in the first place, it is always possible to forcefully truncate the number of iterations in case the computations are taking too long for online usage; a solution to the problem is ensured after the first iteration, although it is likely suboptimal. This is a distinct advantage over other, similar solution strategies such as Lagrange relaxation; see also (Gunnerud et al., 2009) and (Gunnerud and Foss, 2010). Dantzig-Wolfe decomposition has also been used successfully in model predictive control of chemical plants, see (Cheng et al., 2008).

Venkat et al. (2008) uses more traditional distributed MPC to solve a similar portfolio control problem (more precisely, an Automatic Generation Control problem). However, it is not clear how the Scalability and Flexibility objectives can be managed efficiently by the approach presented in that paper. These issues are addressed directly by the Dantzig-Wolfe approach presented here.

Other related solution approaches to decentralised and/or hierarchical control can be found in e.g., (Rantzer, 2009), (Beccuti et al., 2004), (Picasso et al., 2010) and (Scattolini, 2009), amongst others.

The design, is initially developed for the Western Danish power system, since

it already exhibits some of the traits outlined above: on average, about 20 per cent of the electrical energy is supplied by wind, while the rest is supplied by a mixture of fossil fuel, bio-fuels etc. The Danish power system currently has one of the highest ratios of renewable energy in the world; however, other countries are expressing their interests toward similar introduction of renewables. As a consequence, the design presented here can likely be used with minor modifications for various other systems in the future.

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, the existing portfolio controller will also be discussed briefly. 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 presents a comparison of control performances based on simulations of the actual portfolio, whereupon Section 6 sums up the contributions of this work.

The notation is mostly standard. Scalars are written in normal font, while vectors and matrices are wrtten in boldface. $(\cdot)^T$ indicates the transpose of a matrix or vector, while $\mathbf{v}\perp\mathbf{w}$ indicates that the pair of vectors $\mathbf{v}$ and $\mathbf{w}$ is orthogonal. If $\alpha = \{\alpha_i\}$ and $\beta = \{\beta_j\}$ are ordered sets of the same cardinality n, the notation $\alpha_i \perp \beta_i,\ i = 1, 2, \ldots, n$ indicates that $\alpha_i\beta_i = 0$ for each i, even if $\alpha_i \neq 0$ and $\beta_j \neq 0$ for some $i, j = 1, 2, \ldots, n$. Finally, Δ is the backward difference operator, i.e., $\Delta\mathbf{u}_k = \mathbf{u}_k - \mathbf{u}_{k-1}$, where $k-1$ and k are consecutive sample numbers and $\mathbf{u}$ is a signal vector.

2. System description

The Danish power grid 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, consumption and production must be balanced at all times. Roughly speaking, if the consumption is larger than the production, energy will be drained from the system, making the generators slow down, and

vice versa. Such imbalances manifest themselves as deviations from the usual 50 Hz grid frequency. In order to maintain the overall balance between production and consumption, ENTSO-E is split into several regions, each governed by a Transmission System Operator (TSO) responsible for matching production with consumption and import/export into/out of the region.

2.1. Western Denmark

The major production units of the Western Danish region are shown in Figure 1.

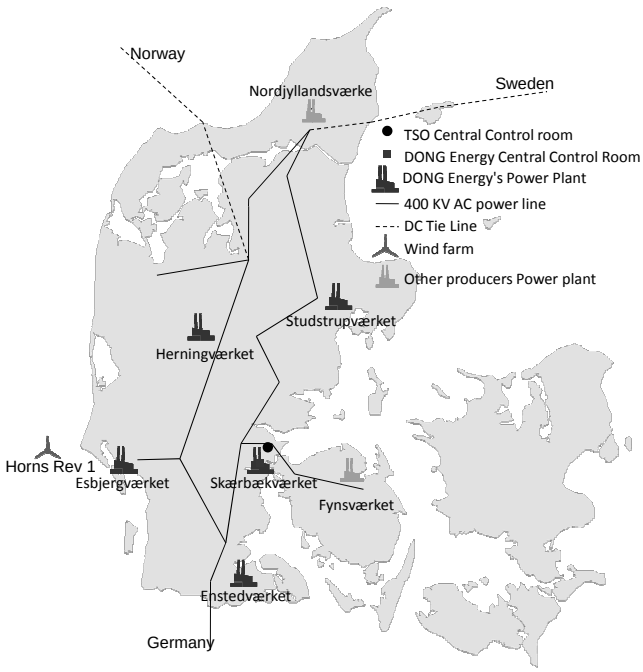

Figure 1: Within the west Danish area there are 7 sites containing large power plants comprising 9 boiler units in total with an electrical production capacity ranging from 80 MW to 650 MW; the most common size is around 400 MW. There are two major producers in the area; DONG Energy is the largest and operates a total of 6 units in the area.

Maintaining balance between production and consumption within Scandinavia is managed via energy markets such as NordPool (Nord Pool, 2010); contracts closed on the relevant energy markets yield an hourly amount of energy

that suppliers must produce within each region. The amount of energy sold is passed to a *Short-Term Load Scheduler* (STLS), which solves a *Unit Commitment problem* as mentioned in the introduction[1]. The result is a load schedule with a time resolution of 5 minutes for each individual producing unit, as shown in Figure 2.

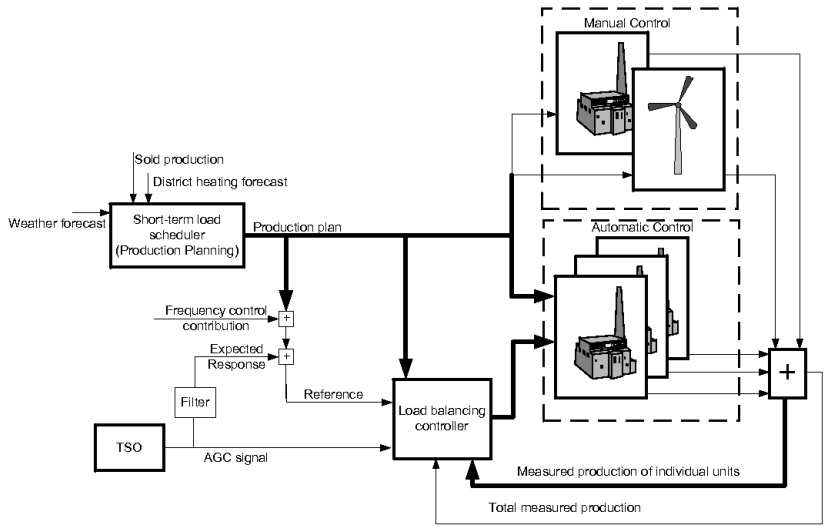

Figure 2: Diagram of the interconnection of the system. The bold lines show vectors of signals, while thin lines indicate scalar signals. The portfolio can be divided into two categories: units under manual control, which the load balancing controller cannot give corrections to, and units under automatic control, which the load balancing controller can affect.

However, even though the market provides 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. *Primary reserves* are activated in order to compensate for frequency deviations from 50 Hz; these must be very fast and are basically activated throughout the European grid as necessary. *Secondary reserves* are used to replace the primary reserves, in

[1] A more detailed description of the Short-Term Load Scheduler used in Western Denmark can be found in Jørgensen et al. (2006)

the sense that if an area creates a frequency deviation, all areas first seek to stabilise the system with the primary reserves, but the area responsible for the imbalance must bring the system back to nominal behaviour by activating secondary reserves. Each control area, including West Denmark, has secondary reserves. Finally, *tertiary reserves* are used to replace secondary reserves on a market basis. These reserves must be activated within 15 minutes of being ordered. They are activated by an operator at the TSO, who takes direct contact to an operator at an energy generation company within the region. This 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.

From a portfolio control perspective, the secondary reserves are the more interesting, so the attention will be restricted to those in the following. Further details about the reserve allocation can be found in ENTSO-E (2010).

2.2. Current controller

The current load balancing controller structure employed by DONG Energy for the Western Danish region is described in Edlund et al. (2009a); it basically serves two purposes: maintainining the internal balance among the generating assets operated by DONG Energy, and activating secondary reserves. It is an adaptation of an automatic generation control system found in Wood and Wollenberg (1996) and consists of a set of parallel PI-controllers, whose gains can be changed to accommodate changing load scenarios and constraints.

Wood and Wollenberg (1996) suggests that the individual gains should be determined from a steady-state optimisation. However, due to the conditions in the West Danish area, where the boiler units are used for load balancing and hence have to change set points very often, this 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 subsequently aims to return all the boiler units to the production schedule.

The problem with the current controller is the complexity of its many cross couplings, special rules etc., 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 may change structure over time. Furthermore, the complexity of the logics renders any form of rigorous stability or performance analysis virtually impossible. As a consequence, a novel, modular control scheme has been developed, which will be presented in the following.

3. Proposed controller structure

The structure of the proposed controller is a two layer hierarchical structure as shown in Figure 3. All parts pertaining to the individual units 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.

3.1. Assumptions

The design framework relies on a set of assumptions:

- The units can be modelled as being independent of each other, such that a change in one unit does not directly affect another unit.
- The units can be modelled as a linear dynamic system with affine constraints. The investigated models in Edlund et al. (2009b) can all be made to fit with the structure shown in Figure 4 with minor modifications.
- 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.

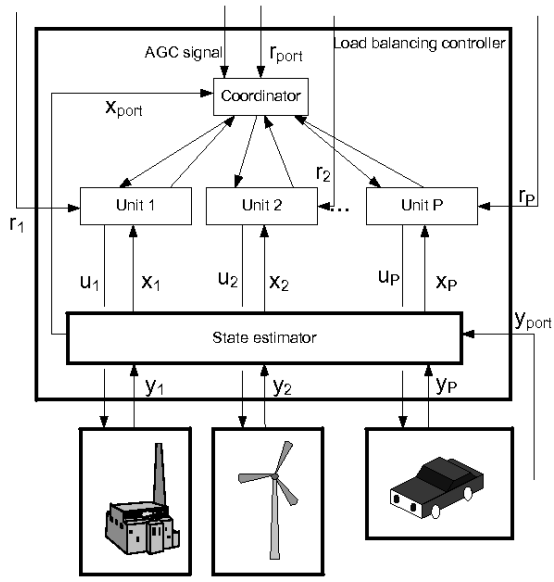

Figure 3: Sketch of the modular structure of the load balancing controller. Communication with the individual unit is handled by the independent subsystems, and portfolio communication is handled on the upper layer of the hierarchy. r_i is the reference to unit $i \in \{1, 2, \ldots, P\}$, $\mathbf{x}_i$ is an estimate of the state vector in the i'th dynamical model, 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.

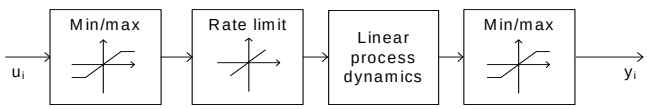

Figure 4: General structure of the units

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

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

Note that the lower layer is also responsible for state estimation tasks. As shown in Figure 4, this task is for simplicity handled by a single Kalman filter-based estimator in the current setup. In future implementations, it would be highly relevant to replace this estimator by a distributed setup, for instance following the approaches given in Mutambara (1998) or A. N. Venkat and Wright (2006); however, since the estimation is not really the focus of the current paper, the simplest solution has been chosen here.

3.2. Dantzig-Wolfe Decomposition

The hierarchical structure above encapsulates the information pertaining to each unit. Since each unit is modelled via linear dynamics and subject to individual constraints, and the objective is to track a specified reference for which deviation costs are directly proportional to the size of the deviation, we can formulate the overall MPC problem as a linear program of the form

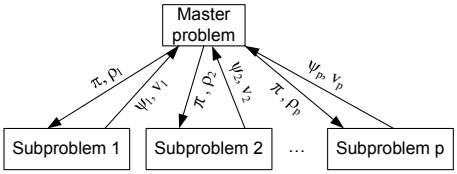

Figure 5: Concept of the Dantzig-Wolfe decomposition. The big problem is split into several smaller problems communicating with a coordinator to reach the optimum.

$$\min_{\mathbf{z}} \quad \phi = \mathbf{c}_1^T \mathbf{z}_1 + \mathbf{c}_2^T \mathbf{z}_2 + ... + \mathbf{c}_P^T \mathbf{z}_P \tag{1a}$$

$$s.t. \quad \begin{bmatrix} \mathbf{F}_1 & \mathbf{F}_2 & \dots & \mathbf{F}_P \\ \mathbf{G}_1 & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{G}_2 & & \vdots \\ \vdots & & \ddots & \\ \mathbf{0} & \dots & & \mathbf{G}_P \end{bmatrix} \begin{bmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_P \end{bmatrix} \geq \begin{bmatrix} \mathbf{g} \\ \mathbf{h}_1 \\ \mathbf{h}_2 \\ \vdots \\ \mathbf{h}_P \end{bmatrix}. \tag{1b}$$

with $\mathbf{z} = [\mathbf{z}_1^T \ \mathbf{z}_2^T \ \dots \mathbf{z}_P^T]^T \in \mathbb{R}^n$, $\mathbf{z}_i \in \mathbb{R}^{n_i}$, $\phi \in \mathbb{R}$, $\mathbf{F}_i \in \mathbb{R}^{m \times n_i}$, $\mathbf{G}_i \in \mathbb{R}^{p_i \times n_i}$, $\mathbf{g} \in \mathbb{R}^m$ and $\mathbf{h}_i \in \mathbb{R}^{p_i}$. ϕ is a functional which needs to be minimised in order to find optimum, $\mathbf{z}_i$ are decision 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 unit 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. $\mathbf{G}_i$ describes the dynamics and constraints related to the individual unit i. $\mathbf{g}$ and $\mathbf{h}_i$ are affine parts of the constraints.

Throughout the description of the decomposition without loss of generality it is assumed that the feasible region of each subproblem is closed and bounded. (Dantzig and Wolfe, 1960).

Dantzig-Wolfe decomposition builds on the well-known property of convex combinations, which states that any point in a polytope can be expressed as a convex combination of its vertices.

Indeed, using convex combinations, the polytope $\mathcal{Z}_i = \{\mathbf{z}_i | \ \mathbf{G}_i \mathbf{z}_i \geq \mathbf{h}_i\}$ can

be expressed as

$$\mathbf{z}_i = \sum_{j=1}^{M_i} \lambda_{ij} \mathbf{v}_i^j, \quad \sum_{j=1}^{M_i} \lambda_{ij} = 1, \quad \lambda_{ij} \geq 0, \quad j = 1, 2, \ldots, M_i \tag{2}$$

where $\mathbf{v}_i^j,\ j = 1, 2, \ldots, M_i$ are the vertices (extreme points) of $\mathcal{Z}_i$. Substituting (2) into (1a) and defining $f_{ij} = \mathbf{c}_i^T \mathbf{v}_i^j$ and $\mathbf{p}_{ij} = \mathbf{F}_i \mathbf{v}_i^j, i = 1, 2, ..., P;\ j = 1, 2, \ldots, M_i$, allows us to rewrite the block-angular linear program (1) as the equivalent Master Problem

$$\min_{\lambda} \quad \phi = \sum_{i=1}^{P} \sum_{j=1}^{M_i} f_{ij} \lambda_{ij} \tag{3a}$$

$$s.t. \quad \sum_{i=1}^{P} \sum_{j=1}^{M_i} \mathbf{p}_{ij} \lambda_{ij} \geq \mathbf{g} \tag{3b}$$

$$\sum_{j=1}^{M_i} \lambda_{ij} = 1, \qquad i = 1, 2, ..., P \tag{3c}$$

$$\lambda_{ij} \geq 0, \qquad i = 1, 2, ..., P;\ j = 1, 2, \ldots, M_i \tag{3d}$$

Note that the Master Problem has fewer constraints than the original problem, but the number of variables in the Master Problem is larger due to a larger number of extreme points.

The Lagrangian associated with the Master Problem (3) is

$$\begin{aligned} \mathcal{L}(\lambda_{ij}, \pi, \rho_i, \kappa_{ij}) = & \sum_{i=1}^{P} \sum_{j=1}^{M_i} f_{ij} \lambda_{ij} - \pi^T \left(\sum_{i=1}^{P} \sum_{j=1}^{M_i} \mathbf{p}_{ij} \lambda_{ij} - \mathbf{g} \right) \\ & - \sum_{i=1}^{P} \rho_i (\sum_{j=1}^{M_i} \lambda_{ij} - 1) - \sum_{i=1}^{P} \sum_{j=1}^{M_i} \kappa_{ij} \lambda_{ij} \end{aligned} \tag{4}$$

with $\pi \in \mathbb{R}^m$ being the Lagrange multiplier of the coupling constraint (3b), $\rho_i \in \mathbb{R}$ the Lagrange multiplier for (3c) and $\kappa_i \in \mathbb{R}^{M_i}$ the Lagrange multiplier for the positivity constraint (3d).

Consequently, the necessary and sufficient optimality conditions for the Mas-

ter Problem (3) are

$$\nabla_{\lambda_{ij}}\mathcal{L} = f_{ij} - \mathbf{p}_{ij}^T\pi - \rho_i - \kappa_{ij} = 0, \qquad i = 1,2,\ldots,P;\ j = 1,2,\ldots,M_i \tag{5a}$$

$$\sum_{i=1}^{P}\sum_{j=1}^{M_i}\mathbf{p}_{ij}\lambda_{ij} - \mathbf{g} \geq 0 \quad \perp \quad \pi \geq 0 \tag{5b}$$

$$\sum_{j=1}^{M_i}\lambda_{ij} - 1 = 0, \qquad i = 1,2,\ldots,P \tag{5c}$$

$$\lambda_{ij} \geq 0 \quad \perp \quad \kappa_{ij} \geq 0, \qquad i = 1,2,\ldots,P;\ j = 1,2,\ldots,M_i \tag{5d}$$

We notice that the conditions (5a) and (5d) imply

$$\kappa_{ij} = f_{ij} - \mathbf{p}_{ij}^T\pi - \rho_i = \left[\mathbf{c}_i - \mathbf{F}_i^T\pi\right]^T\mathbf{v}_i^j - \rho_i \geq 0, \quad i = 1,2,\ldots,P;\ j = 1,2,\ldots,M_i \tag{6}$$

such that the Karush-Kuhn-Tucker (KKT) conditions for (3) may be stated as

$$\sum_{i=1}^{P}\sum_{j=1}^{M_i}\mathbf{p}_{ij}\lambda_{ij} - \mathbf{g} \geq 0 \quad \perp \quad \pi \geq 0 \tag{7a}$$

$$\sum_{j=1}^{M_i}\lambda_{ij} - 1 = 0, \qquad i = 1,2,\ldots,P \tag{7b}$$

$$\lambda_{ij} \geq 0 \quad \perp \quad \kappa_{ij} = \left[\mathbf{c}_i - \mathbf{F}_i^T\pi\right]^T\mathbf{v}_i^j - \rho_i \geq 0, \quad i = 1,2,\ldots,P;\ j = 1,2,\ldots,M_i \tag{7c}$$

Large problems are characterized by a very large number of extreme points. Therefore, generation of all the extreme points in the Master Problem (3) can in itself be a very challenging computational problem. The Dantzig-Wolfe algorithm overcomes this challenge by using delayed column generation, i.e. it generates the extreme points for the underlying Simplex basis algorithm only when needed.

The master problem with a reduced number of extreme points is called the

Reduced Master Problem (RMP) and can be expressed as

$$\min_{\lambda} \quad \phi = \sum_{i=1}^{P} \sum_{j=1}^{l} f_{ij} \lambda_{ij} \tag{8a}$$

$$s.t. \quad \sum_{i=1}^{P} \sum_{j=1}^{l} \mathbf{p}_{ij} \lambda_{ij} \geq \mathbf{g} \tag{8b}$$

$$\sum_{j=1}^{l} \lambda_{ij} = 1, \qquad i = 1, 2, ..., P \tag{8c}$$

$$\lambda_{ij} \geq 0, \qquad i = 1, 2, ..., P;\ j = 1, 2, \ldots, l \tag{8d}$$

in which $l \leq M_i$ for all $i = 1, 2, \ldots, P$. 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 $i = 1, 2, \ldots, P$.

Initially, a feasible extreme point for the Master Problem (3) is needed. We may generate such a point using techniques similar to Phase I in the simplex algorithm for a standard linear program (Farris et al., 2007).

Posing the problem as a linear program will add an extra set of decision variables to the master problem originating from $\mathbf{x}_port$, these variables are denoted $\mathbf{z}_{tot}$. The 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.

In the following, it is assumed that a feasible extreme point has been computed. This feasible extreme point is used to form a Reduced Master Problem with $l = 1$. The solution to the Reduced Master Problem (8) is denoted as

λ_{ij}^{RMP}, such that a feasible solution to Master Problem (3) is

$$\lambda_{ij} = \lambda_{ij}^{RMP}, \qquad i = 1, 2, \ldots, P;\ j = 1, 2, \ldots, l \tag{9a}$$

$$\lambda_{ij} = 0, \qquad i = 1, 2, \ldots, P; j = l+1, l+2, \ldots, M_i \tag{9b}$$

This solution satisfies (7a) and (7b). To be optimal it also needs to satisfy (7c). These conditions are already satisfied for $i = 1, 2, \ldots, P$ and $j = 1, 2, \ldots, l$. It remains to verify whether they are satisfied for all $i = 1, 2, \ldots, P$ and $j = l+1, l+2, \ldots, M_i$. However, only the extreme points $\mathbf{v}_i^j$ for $i = 1, 2, \ldots, P$ and $j = 1, 2, \ldots, l$ are known.

(7c) 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

$$\psi_i = \min_{\mathbf{v}_i^j} \left[\mathbf{c}_i - \mathbf{F}_i^T \pi\right]^T \mathbf{v}_i^j, \qquad i = 1, 2, \ldots, P \tag{10}$$

$\mathbf{v}_i^j$ 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, the solution of (10) may be computed as the solution of the linear program

$$\psi_i = \min_{\mathbf{z}_i} \quad \phi = \left[\mathbf{c}_i - \mathbf{F}_i^T \pi\right]^T \mathbf{z}_i \tag{11a}$$

$$s.t. \quad \mathbf{G}_i \mathbf{z}_i \geq \mathbf{h}_i \tag{11b}$$

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. The solution to the original problem (1) can then be computed via

$$\mathbf{z}_i^* = \sum_{j=1}^{l} \mathbf{v}_i^j \lambda_{ij}, \qquad i = 1, 2, \ldots, P \tag{12}$$

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, the Reduced Master Problem may be augmented with the new extreme points, $\mathbf{v}_i^{l+1}$, obtained from solutions to the subproblems (11).

The next iteration of the algorithm starts with the solution of the new Reduced Master Problem. The algorithm will always terminate in a finite number of iterations, since the number of extreme points is finite.

Algorithm 1 summarises the Dantzig-Wolfe Algorithm for solution of the block-angular linear program (1). Note that the subproblems (14) may be solved in parallel, which is advantageous when the number of subproblems P is large. It is also worthy of note that once a feasible solution is found, the Dantzig-Wolfe Algorithm preserves feasibility of (1) in all iterations.

4. Specific controller implementation

In the current system, only boiler load units are available for control purposes. The portfolio does contain a mixture of other production units, such as various smaller thermal power plants and wind turbines, but these cannot be controlled by the load balancing controller and must therefore be considered as disturbances. They have a production reference and their production is measurable, but little is known about their dynamical behaviour; they are treated below. First, however, the boiler load units are addressed.

4.1. Boiler load units

In the current setup the load balancing controller has authority over up to 6 power plant units. 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.

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, it has to be adjusted slightly - see Figure 6.

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

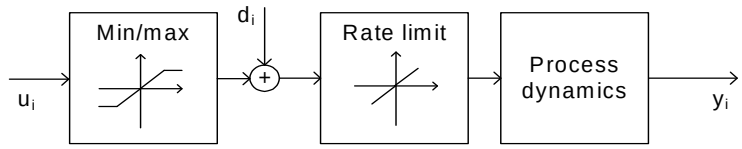

Figure 6: Concept of the boiler modelling

controller. Thus in the nominal case u_i is zero since no corrective signals are needed. The output y_i is the power production from the boiler unit.

The process dynamics is modelled as the third order system

$$H(s) = \frac{1}{(T_i s + 1)^3} \tag{15}$$

where T_i is typically around 50s, but ranging from 15s to 90s for the individual units.

In order to gain offset-free tracking, the linear models are augmented with an output error model; this gives rise to the constrained augmented discrete time state space model of the form $\mathbf{x}_{i,k+1} = \mathbf{A}_i \mathbf{x}_{i,k} + \mathbf{B}_i \mathbf{u}_{i,k} + \mathbf{E}_i \mathbf{d}_{i,k},\ y_{i,k} = \mathbf{C}_i \mathbf{x}_{i,k}$, or more specifically

$$\mathbf{x}_{i,k+1} = \begin{bmatrix} a_{1,1,i} & 0 & 0 & 0 \\ a_{2,1,i} & a_{2,2,i} & 0 & 0 \\ a_{3,1,i} & a_{3,2,i} & a_{3,3,i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \mathbf{x}_{i,k} + \begin{bmatrix} b_{1,i} \\ b_{2,i} \\ b_{3,i} \\ 0 \end{bmatrix} \mathbf{u}_{i,k} + \begin{bmatrix} e_{1,i} \\ e_{2,i} \\ e_{3,i} \\ 0 \end{bmatrix} \mathbf{d}_{i,k} \tag{16a}$$

$$y_{i,k} = \begin{bmatrix} 0 & 0 & 1 & 1 \end{bmatrix} \mathbf{x}_{i,k} \tag{16b}$$

$$\underline{u_{i,k}} \leq u_{i,k} \leq \overline{u_{i,k}} \tag{16c}$$

$$\max\{\underline{\Delta u_{i,k}} - \Delta d_{i,k}, 0\} \leq \Delta u_{i,k} \leq \min\{\overline{\Delta u_{i,k}} - \Delta d_{i,k}, 0\} \tag{16d}$$

Where k is the sample number. The elements in $\mathbf{A}_i$, $\mathbf{B}_i$ and $\mathbf{E}_i$ dependent on T_i and the sample time. $(\underline{\cdot})$ and $(\overline{\cdot})$ indicate lower and upper bounds, respectively. To avoid forcing the controller to take actions in case the production plan violates the rate of change constraints, the lower and upper rates of change constraints are modelled such that they are always non-positive and non-negative,

respectively. The upper and lower limits for the controller (16c) are set in the control system by the operator.

As mentioned in Section 2 there exists some fast reserves to stabilise the system in case of unforeseen events termed primary reserves. The power plants are capable of delivering primary reserve but amount of reserve is dependent on current production. Primary reserves are reserved on each unit, which may affect the upper and lower limit of the controller.

The rate of change constraint is dependent on the boiler load; however, in order to keep the optimisation problem linear, a linearisation based on the prediction is used in the model. More specifically, the prediction of u is used to generate rate of change constraints throughout the prediction horizon. If no prediction of u can be obtained, 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

$$\min_{\mathbf{U}_i} \quad \phi_i = \sum_{k=0}^{N-1} \left(p_{i,k+1} y_{i,k+1} + q_i |\eta_{i,k+1}| + s_i |\Delta u_{i,k}| \right) \tag{17a}$$

$$s.t. \quad \mathbf{x}_{i,k+1} = \mathbf{A}_i \mathbf{x}_{i,k} + \mathbf{B}_i u_{i,k} + \mathbf{E}_i d_{i,k}, \qquad k = 0, 1, \ldots, N-1 \tag{17b}$$

$$y_{i,k} = \mathbf{C}_i \mathbf{x}_{i,k}, \qquad k = 1, 2, \ldots, N \tag{17c}$$

$$\xi_{i,k+1} = \mathbf{A}_i \xi_{i,k} + \mathbf{B}_i u_{i,k}, \qquad k = 0, 1, \ldots, N-1 \tag{17d}$$

$$\eta_{i,k} = \mathbf{C}_i \xi_{i,k}, \qquad k = 1, 2, \ldots, N \tag{17e}$$

$$\underline{u_{i,k}} \leq u_{i,k} \leq \overline{u_{i,k}}, \qquad k = 0, 1, \ldots, N-1 \tag{17f}$$

$$\underline{\Delta u_{i,k}} \leq \Delta u_{i,k} \leq \overline{\Delta u_{i,k}}, \qquad k = 0, 1, \ldots, N-1 \tag{17g}$$

where N is the prediction horizon. $\mathbf{U}_i = [u_{i,0}, u_{i,1}, ..., u_{i,N-1}]^T$ is a vector of control signals to be determined, and $\mathbf{A}_i, \mathbf{B}_i, \mathbf{C}_i, \mathbf{E}_i$ are the system parameter matrices specified in (16a) and (16b).

$y_{i,k}$ is the total output from the boiler unit, based on controller input as

well as production plan, while $\eta_{i,k}$ is the output from the boiler unit based on controller input only (as noted above, it is desired to keep this signal small, since we assume that the short-term load scheduler provides the correct reference).

The first term in the performance function $p_{i,k+1}y_{i,k+1}$ is a linear term representing the cost of using 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 the (state-dependent) boiler efficiency; the calculations are based on the production plan alone. The term is used to weight the plants against each other when the control signals are distributed.

The term $q_i|\eta_{i,k+1}|$ is added in order to force the input signal from the controller toward zero. Conventionally, the weight would be applied directly to the input, but simulations show that the controller obtains a better behaviour when the weight is placed on the output generated by the controller input (most likely due to the slow plant dynamics).

The last term of the performance function, $s_i|\Delta u_{i,k}|$, is a penalty on rapid changes on the correction signal.

This information is managed separately within each of the units on the lower layer of the hierarchy.

4.2. Other portfolio units

As mentioned above, the portfolio also contains a mixture of other production units. In order to include them in the controller, the portfolio is augmented with an output error model, yielding the combined portfolio output

$$x_{port,k+1} = x_{port,k} \tag{18a}$$

$$y_{port,k} = x_{port,k} \tag{18b}$$

where y_{port} is the output from all non-controlled units lumped together. Since there is no controllable or measurable input to these units, there is no input to affect the state. However, the Kalman filter used for state estimation will update the state to adjust to the measured total output from the portfolio.

The objective function for the portfolio minimisation problem is given as

$$\phi_{port} = \sum_{k=1}^{N} q_{port,k} \left| y_{port,k} + \sum_{i=1}^{P} y_{i,k} - r_{port,k} \right| \tag{19}$$

where r_{port} is the portfolio reference, i.e., the sum of references to all units in the system plus the demand from the TSO, as shown in Figure 2.

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

5. Simulation results

In order to evaluate the new controller, it will be tested against the currently running controller through simulation, using input data from 15 days of actual operation.

The current controller is implemented in Simulink^{TM}(Mathworks, 2010), and compiled so that it is able to be executed in the central control room. In other words, the new controller is compared against the *actual* existing controller, not a simplified implementation. The simulations are executed in a Simulink^{TM}simulation environment encompassing the entire portfolio.

All data management in the new controller, such as reading measurement data and constructing constraints, is implemented in Java, while the optimisation problem is handled using a purpose-built solver coded in Matlab^{TM}(Mathworks, 2010).

The dynamic parts of the boiler unit models are implemented as linear models or linear parameter varying models with constraints, as explained above. The sampling rate used in the simulations is 0.2 Hz, and the prediction horizon of the model predictive controller is chosen to 25 samples by default.

Simulations cover 25-hour sequences starting from 23:00 to midnight the following day. The first hour is then excluded from the analysis to avoid startup and settling issues, allowing to string together several 24-hour sequences from midnight to midnight for more extensive analysis.

5.1. Noise-free scenario

The controller is evaluated in two scenarios, one without measurement noise and one with simulated noise. For the noise-free case, standard deviation and mean error are used as quantitative measures for the evaluation.

Looking at the actual production (Figure 7), it is evident that both controllers tend to follow the reference well. There are periods where adhering to the primary reserve constraint causes the proposed controller temporarily to perform poorer than the current controller, however; Figure 8 shows an example of this. During these periods, there are apparently insufficient reserves available to fulfill both the primary reserve reservations and follow the reference. Rather than violating the constraints formulated in the optimisation problem, the new controller chooses a correction signal that leads to poorer reference tracking performance, but maintains the primary reserves. Removing the constraint imposed by the primary reserves improves the tracking performance, as can be seen from Figure 8.

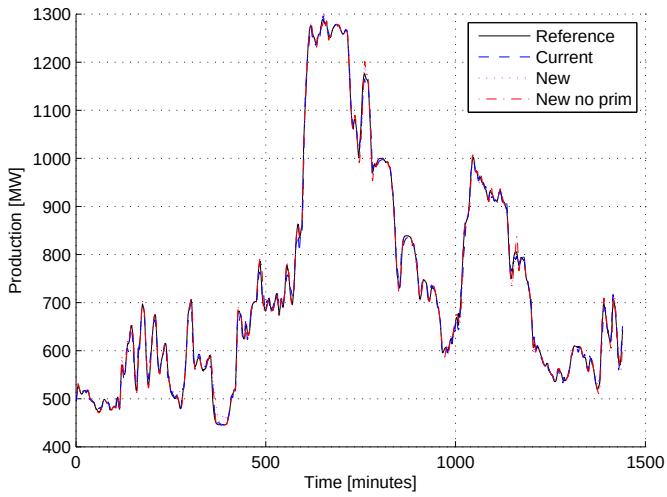

Figure 7: The production of day 3 in the scenario. Both controllers tend to follow the reference well. Both the new controller (· · ·) and the current controller (− −), follow the reference (—) well.

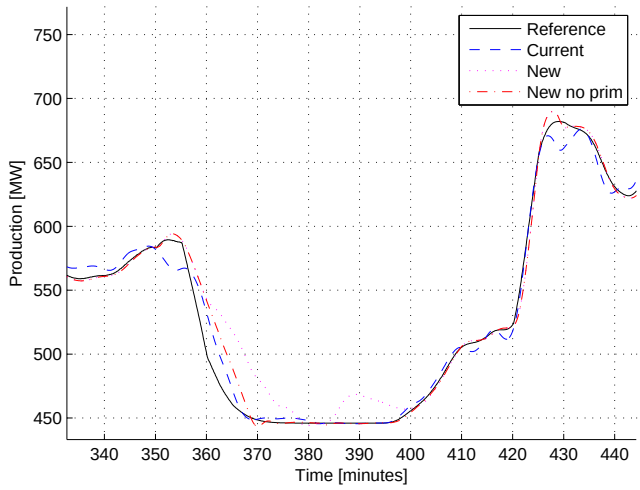

Figure 8: Section of the production on day 3, showing a period where the primary constraint is active and thus limits the new controller $(\cdots)$ from reaching the reference (—). Removing this constraint allows the new controller $(-\cdot-)$ to perform similar to the current controller $(--)$.

Switching from manual to automatic mode is handled efficiently by both controllers, as seen from Figure 9.

To evaluate the relative performance on a larger time scale, Figure 10 shows the mean error and standard deviation

$$\begin{aligned} \mu_{Day} &= \frac{1}{N_s}\sum_{k=1}^{N_s} y_{port,k} - r_{port,k} \\ \sigma_{Day} &= \sqrt{\frac{1}{N_s}\sum_{k=1}^{N_s}((y_{port,k} - r_{port,k}) - \mu_{Day})^2} \end{aligned}$$

where N_s is the number of samples in the simulation and index Day indicates the day of the scenario.

The simulations show that the mean deviation is roughly comparable in all cases, but the standard deviation is generally lower for the new controller, and removing the primary reserve constraint improves performance further, as expected. The standard deviation and mean for the entire 15-day scenario are

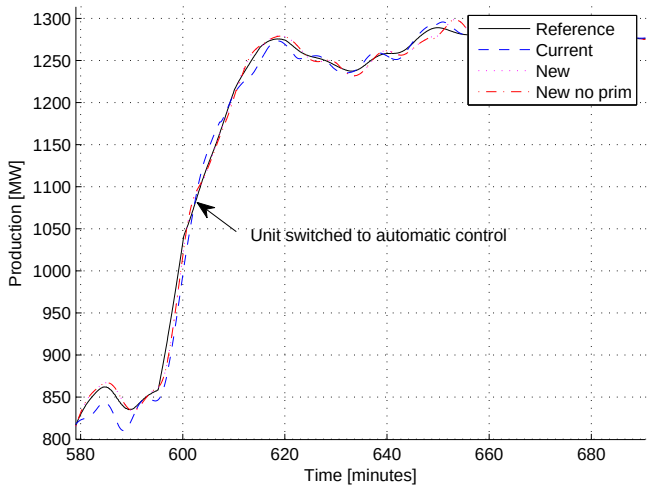

Figure 9: Section of day 3, where a power plant is switched from manual to automatic control. Both the new controller ($\cdots$) and the current controller ($--$) handle this event in a bumpless fashion and follow the reference (—) well.

given in Table 1.

	Noise-free		Noisy	
	σ [MW]	μ [MW]	σ [MW]	μ [MW]
Measurements	-	-	14.25	-1.51
Current	8.49	-0.52	15.59	-1.78
New	7.49	0.38	17.05	0.32
New no primary	6.54	-0.33	-	-

Table 1: Standard deviation and mean throughout the Scenario

5.2. Noisy scenario

Since the scenario discussed above is based on 15 days of actual operation, *actual* input and output sample sequences have been recorded for each boiler unit. One can thus estimate a noise sequence for the scenario as

$$y_n = y_{meas} - y_{sim} \tag{20}$$

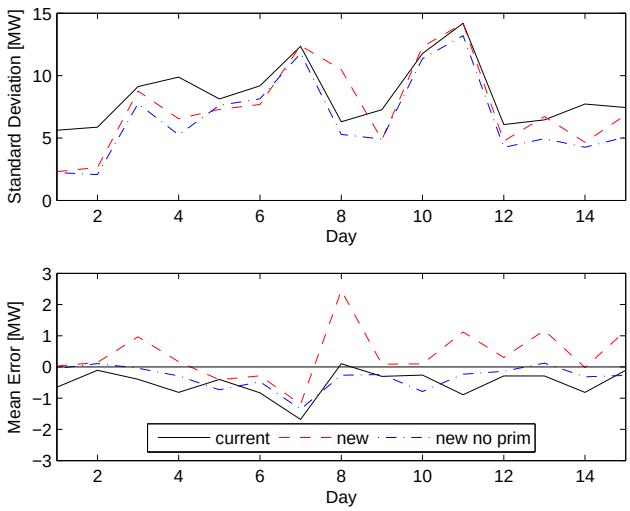

Figure 10: Performance results, noise-free scenario. The top plot shows the standard deviation from the STLS schedule generated by the current controller (—), the new controller (−−) and the new controller with the primary reserve constraints removed (−·−) on a daily basis, while the bottom plot shows the corresponding mean error. The new controller generally produces a lower standard deviation in the output error.

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 and is thus not white. Nonetheless, it is chosen to use this noise sequence in order to make the simulation scenario resemble the actual scenario as closely as possible; that is, realistic occurrences of failures, large steady state offsets etc., are included in the input data used for the simulation. The measurements from other units in the portfolio are applied directly in the simulation as well.

The mean and standard deviation are once again used as quantitative measures for controller performance. Furthermore, the price difference between the controllers can be calculated given fuel costs and deviation prices.

The standard deviation produced by the new controller is slightly higher than for the current controller as was the case in the noise-free scenario. The

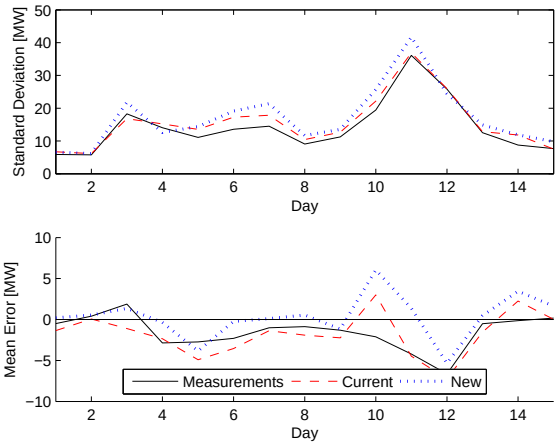

Figure 11: Standard deviation (top) and mean error (bottom) for the controllers in the noisy scenario. The simulated performance of both the current controller (−−) and the new controller (· · ·) is closely comparable to the actual operation, although the new controller is slightly better at following the reference.

deviations produced by both controllers are slightly higher than the measurement data, which is likely caused by the noise generation scheme.

The mean error is larger in the noisy scenario compared to the noise-free one. Though not consistently lower, the average error computations shown in Table 1 indicate that the new controller is overall better at following the reference than the current controller.

Figure 12 shows the price difference between the two controllers. In the price comparison, the base load fuel costs are deducted, such that following the planned production without any reserve activation results in a price of zero. Deviations from the planned production will give increased/decreased fuel costs as well as imbalance costs from the TSO. As the figure shows, the new controller performs better in terms of income most of the time; the current controller only earns more on days 8–10 in this scenario.

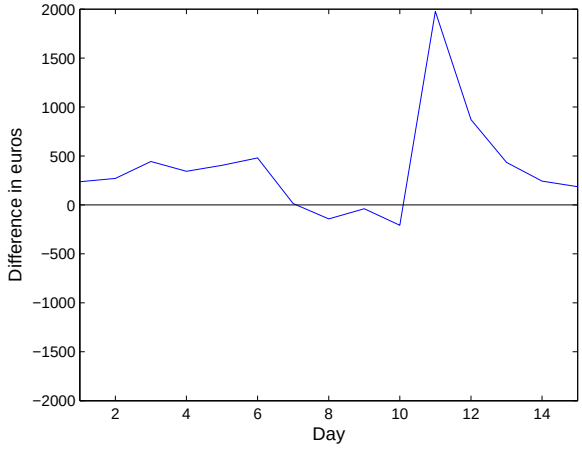

Figure 12: Price difference between the current controller and the new controller. Positive difference means that the new controller is cheaper (i.e., earns more money for the portfolio owner).

	Total costs (€)
Current Controller	100628
New Controller	106140
Difference	5512 / 5.5%

Table 2: Cost comparison for the two controllers. Positive numbers mean an income.

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 lowering the execution time further.

Power plants	Centralised [s]	Dantzig-Wolfe [s]
2	2191	3154
3	3845	3379
4	6506	3956

Table 3: Execution time for 25-hour simulation as a function of the number of active power plants. Both a centralised MPC and a Dantzig-Wolfe MPC

Table 3 shows a comparison of execution times between a centralised MPC implementation and a Dantzig-Wolfe MPC for a 25 hour period with a different amount of active power plants. The Centralised version of the algorithm is fastest with two units while the Dantzig-Wolfe implementation is faster from 3 units and up. The difference in execution time increases rapidly with an increased number of power plants.

Prediction horizon [samples]	Execution time [s]
5	691
15	1245
25	3602
35	12960

Table 4: Execution time for 25-hour simulation as a function of the prediction horizon

Table 4 shows the execution time as a function of the prediction horizon for a day where 4 units are being controlled. 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-hour simulation can be performed in just about an hour.

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 the other is that the algorithm may benefit from better handling of fast vs. slow unit dynamics (compared to the prediction horizon). This is a subject of future research.

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 terms of number units under control. Therefore two design objectives were in focus: flexibility and computational 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 using the Dantzig-Wolfe decomposition algorithm. The decomposition of the optimisation problem also gave a computationally scalable controller, in the sense that 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 causes the computational complexity to grow cubically with the number of states.

The current implementation of the proposed controller relies on a single Kalman filter for state estimation for the whole system. A logical future expansion of the design will be to incorporate distributed estimation with the same hierarchical structure as the controller. This is not considered to be a significant challenge as long as the units in the portfolio remain largely independent of each other.

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 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 again likely caused by the constraint. In the noisy case, it was possible to calculate the cost of the production in the portfolio. Here, it was observed that 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. Indeed, if the estimated values are extrapolated, the accumulated extra earnings from the new controller amount to approximately 100,000 € per year. This is of course not entirely realistic, but it is at least clear that, all other things being equal, the new controller is likely to improve the portfolio owner's earnings noticeably compared to the existing controller.

One final remark is that the currently implemented controller has matured over the cause of nearly a decade. 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 even further improved performance compared to the results presented here.

A. N. Venkat, I. A. Hiskens, J. B. R., Wright, S. J., Dec. 2006. Distributed output feedback mpc for power system control. In: Proceedings of Conf. Decision and Control. pp. 4039–4045.

Alvarado, F. L., 2005. Controlling power systems with price signals. Decision Support Systems 40 (3-4), 495–504.

Anower, M. S., Sheikh, M. R. L., Hossain, M. F., Rabbani, M., Nasiruzzaman, A. B. M., 2006. Fuzzy gain scheduling of an agc in a single area power sys-

tem. International Conference on Electrical and Computer Engineering, 2006. ICECE '06, 9–12.

Aoki, K., Satoh, T., Itoh, M., Ichimori, T., Masegi, K., 1987. 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.

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

Beccuti, A. G., Geyer, T., Morari, M., 2004. Temporal lagrangian decomposition of model predictive control for hybrid systems. In: Proceedings of IEEE Conf. Decision and Control.

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

Dantzig, G. B., Thapa, M. N., 2002. Linear Programming 2: Theory and Extensions. Springer-Verlag.

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

Dillon, T. S., Edwin, K. W., Kochs, H. D., Taud, R. J., 1978. 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.

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

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

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

Farris, M. C., Mangasarian, O. L., Wright, S. J., 2007. Linear Programming with MATLAB. SIAM.

Gunnerud, V., Foss, B., 2010. Oil production optimization – A piecewise linear model, solved with two decomposition strategies. Computers and Chemical Engineering 34, 1803–1812.

Gunnerud, V., Foss, B., Nygreen, B., Vestb, R., Walberg, N. C., 2009. 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.

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

Jokic, A., 2007. Price-based optimal control of electrical power systems. Ph.D. thesis, Technische Universiteit Eindhoven.

Jørgensen, C., Mortensen, J. H., Mølbak, T., Nielsen, E. O., 2006. 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.

Lasdon, L. S., 2002. Optimization theory for large systems, 2nd Edition. Dover Publications Inc.

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

Mutambara, A. G. O., 1998. Decentralized Estimation and Control for Multi-sensor Systems, 1st Edition. CRC Press Inc.

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

Padhy, N. P., 2004. Unit commitment - a bibliographical survey. IEEE Transactions on Power Systems 19 (2), 1196–1205.

Picasso, B., De Vito, D., Scattolini, R., Colaneri, P., 2010. An mpc approach to the design of two-layer hierarchical control systems. Automatica 46, 823–831.

Ramakrishna, K. S. S., Bhatti, T. S., 2008. Automatic generation control of single area power system with multi-source power generation. Proceedings of the Institution of Mechanical Engineers – Part A – Power & Energy 220 (1), 1–11.

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

Rawlings, J. B., Mayne, D. Q., 2009. Model Predictive Control: Theory and Design. Nob Hill Publishing.

Ringel, M., 2003. Liberalising European electricity markets: Opportunities and risks for a sustainable power sector. Renewable and Sustainable Energy Reviews 7 (6), 485–499.

Rossiter, J., 2003. Model-based Predictive Control: A Practical Approach. CRC Press LLC.

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

Scattolini, R., 2009. Architectures for distributed and hierarchical model predictive control. Journal of Process Control 19 (5), 723–731.

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

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

Transport- og Energiministeriet, 2005. Energistrategi 2025 - Perspektiver frem mod 2025. Tech. rep., Transport- og Energiministeriet.

UCTE, 2007. Final Report System Disturbance on 4 November 2006. http://www.ucte.org/pdf/Publications/2007/Final-Report-20070130.pdf.

United Nations, 1998. Kyoto protocol to the United Nations framework convention on climate change. Tech. rep., United Nations.

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

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

Wood, A. J., Wollenberg, B. F., 1996. Power Generation, Operation and control, 2nd Edition. John Wiley & sons, inc.

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

Compute a feasible vertex of the Master Problem (3). If no such point exists then stop.

$l \leftarrow 1$, *converged* $\leftarrow$ **false**.

while not *converged* **do**

Solve Reduced Master Problem (RMP) l:

$$\min_{\lambda} \quad \phi = \sum_{i=1}^{P} \sum_{j=1}^{l} f_{ij} \lambda_{ij} \tag{13a}$$

$$s.t. \quad \sum_{i=1}^{P} \sum_{j=1}^{l} \mathbf{p}_{ij} \lambda_{ij} \geq \mathbf{g} \tag{13b}$$

$$\sum_{j=1}^{l} \lambda_{ij} = 1 \qquad i = 1, 2, \ldots, P \tag{13c}$$

$$\lambda_{ij} \geq 0 \qquad i = 1, 2, \ldots, P;\ j = 1, 2, \ldots, l \tag{13d}$$

$\pi \leftarrow$ Lagrange multiplier for (13b).

$\rho_i \leftarrow$ Lagrange multiplier for (13c).

for $i = 1$ **to** P **do**

Solve subproblem i:

$$\min_{\mathbf{z}_i} \quad \phi_i = \left[\mathbf{c}_i - \mathbf{F}_i^T \pi\right]^T \mathbf{z}_i \tag{14a}$$

$$s.t. \quad \mathbf{G}_i \mathbf{z}_i \geq \mathbf{h}_i \tag{14b}$$

$(\psi_i, \mathbf{v}_i^{l+1}) \leftarrow (\phi_i^*, \mathbf{z}_i^*)$ {optimal value-minimizer pair}

end for

if $\psi_i - \rho_i < 0$ for any $i \in \{1, 2, \ldots, P\}$ **then**

$f_{i,l+1} \leftarrow \mathbf{c}_i^T \mathbf{v}_i^{l+1}$, $\mathbf{p}_{i,l+1} \leftarrow \mathbf{F}_i \mathbf{v}_i^{l+1}$ {coefficients for new columns in RMP}

$l \leftarrow l + 1$

else

for $i = 1$ **to** P **do**

$\mathbf{z}_i^* \leftarrow \sum_{j=1}^{l} \lambda_{ij} \mathbf{v}_i^j$

end for

converged $\leftarrow$ **true**

return $\mathbf{z}^*$

end if

end while