Modeling of Uncertainty for the Real-Time Management of Power Systems

Nicole Gröwe-Kuska[1], Matthias P. Nowak[2], and Isabel Wegner[1]

[1] Humboldt-University Berlin, Institute of Mathematics, D-10099 Berlin, Germany
[2] SINTEF, N-7465 Trondheim, Norway

Abstract. A major issue in the application of multistage stochastic programming to model the cost-optimal generation and trading of electric power is the approximation of the underlying stochastic data processes by tree-structured schemes. We present a methodology for the generation of scenario trees for the stochastic load process from historical load profiles. The statistical modeling of the load process exploits the decomposition of the load process into a daily mean load process and a mean-corrected load series. The probability distribution of the load process over the optimization horizon is derived by using a time series model for the daily mean load process and regression models for the mean-corrected load series. We utilize the explicit representation of the distribution to compute approximate load scenarios and their probabilities. In a final step we reduce the number of load scenarios by a scenario deletion procedure. We report on the application of our approach to the cost-optimal generation of electric power in the hydro-thermal generation system of a German power utility.

1 Introduction

In industrial practice, mathematical models for the efficient generation, transmission, and distribution of electric power have been proved indispensable. The ongoing liberalization of electricity markets stimulates the interest of power utilities in developing modeling and optimization techniques for operating power systems and trading electricity under uncertainty. Utilities participating in deregulated markets observe increasing uncertainty in load (i.e., demand for electric power), inflows to reservoirs and prices for fuel and electricity on spot and contract markets. The mismatched power between actual and predicted demand may be supplied by the power system or by trading activities. The competitive environment forces the utilities to rate alternatives within a few minutes.

In the present paper we develop approximate tree-structured schemes for the stochastic load process entering a multistage mixed-integer stochastic program. It models the weekly cost-optimal generation and trading of an electric hydro-thermal based utility under data uncertainty. The relevant uncertain data may comprise electric load, stream flows to hydro units, and fuel and electricity prices. For solving the stochastic power management model a stochastic Lagrangian relaxation algorithm [15] has been designed. With a state-of-the-art bundle method for solving the dual, specialized subproblem solvers and Lagrangian heuristics, the stochastic version of classical Lagrangian relaxation becomes fairly efficient.

The stochastic power management model has emerged from a collaboration with the German utility Vereinigte Energiewerke AG (VEAG). The VEAG generation system consists of 25 (coal-fired or gas-burning) thermal units and 7 pumped-storage hydro units. Its total capacity is about 13,000 megawatts (MW), including a hydro capacity of 1,700 MW; the system peak loads are about 8,600 MW. In contrast to other hydro-thermal based utilities the amount of installed pumped-storage capacity enables the inclusion of pumped-storage plants into the optimization. An additional feature of that system is that for a planning period of one week, the changes of reservoir levels in the pumped-storage hydro units caused by stream inflows are negligible.

The paper is organized as follows. In §2 we describe the stochastic power management model and the stochastic Lagrangian relaxation approach for its solution. The model selection for the electric load is addressed in §3.2. Our procedures for generating and reducing load scenario trees are presented in §3.3 and §3.4, respectively. In §3.5 we report on numerical tests.

2 Power System Modeling

2.1 Modeling

We consider a power generation system comprising thermal units, pumped storage plants and contracts for delivery and purchase, and describe a model for its cost-optimal operation under uncertainty in electrical load (i.e., demand), stream flows in hydro units and prices for fuel or electricity.

The scheduling horizon for unit commitment is typically discretized into uniform (e.g., hourly) intervals. Accordingly, the load, stream flows and prices are assumed to be constant within each time period. The scheduling decisions for thermal units are: which units to commit in each period, and at what generating capacity. The decision variables for hydro plants are the generation and pumping levels for each period. Contracts for delivery and purchase are regarded as special thermal units. The schedule should minimize the total generation costs, subject to the operational requirements.

We use the following notation. There are T time periods. I and J are the numbers of thermal and hydro units, respectively. For a thermal unit i in period t, $u_{it} \in \{0,1\}$ is its *commitment* (1 if on, 0 if off), and p_{it} its *production*, with $p_{it} = 0$ if $u_{it} = 0$, $p_{it} \in [p_{it}^{\min}, p_{it}^{\max}]$ if $u_{it} = 1$, where $p_{it}^{\min}$ and $p_{it}^{\max}$ are the minimum and maximum capacities. Additionally, there are *minimum up/down-time requirements*: when unit i is switched on (off), it must remain on (off) for at least $\bar{\tau}_i$ ($\underline{\tau}_i$, resp.) periods. For a hydro plant j, v_{jt} and w_{jt} are its *generation* and *pumping* levels in period t, with upper bounds $v_{jt}^{\max}$ and $w_{jt}^{\max}$ respectively, and l_{jt} is the *storage* volume in the upper dam at the end of period t, with upper bound $l_{jt}^{\max}$. The water balance relates

l_{jt} with $l_{j,t-1}$, v_{jt}, w_{jt} and the *water inflow* γ_{jt}, using the *pumping efficiency* η_j. The initial and final volumes are specified by l_j^{in} and l_j^{end}.

The basic system requirement is to meet the electric load. Another important requirement is the spinning reserve constraint. To maintain reliability (compensate sudden load peaks or unforeseen outages of units) the total committed capacity should exceed the load in every period by a certain amount (e.g., a fraction of the demand). The *load* and the *spinning reserve* during period t are denoted by d_t and r_t, respectively. Typical load curves exhibit daily and weekly cycles with lows during nights and weekends, and morning and early evening peaks. Efficient operation of pumped storage hydro plants exploits such cycles by generating during peak load periods and pumping during off-peak periods.

Since the operating costs of hydro plants are usually negligible, the total system cost is given by the sum of startup and operating costs of all thermal units over the whole scheduling horizon. The *fuel cost* C_{it} for operating thermal unit i during period t has the form

$$C_{it}(p_{it}, u_{it}) := \max_{l=1,\ldots,\bar{l}} \{ a_{ilt} p_{it} + b_{ilt} u_{it} \}, \tag{1}$$

with coefficients a_{ilt}, b_{ilt} such that $C_{it}(\cdot, 1)$ is convex and increasing on R_+; note that $C_{it}(0,0) = 0$. The startup cost of unit i depends on its downtime; it may vary from a maximum cold-start value to a much smaller value when the unit is still relatively close to its operating temperature. This is modeled by the *startup cost*

$$S_{it}(u_i) := \max_{\tau=0,\ldots,\tau_i^c} c_{i\tau} \left(u_{it} - \sum_{\kappa=1}^{\tau} u_{i,t-\kappa} \right), \tag{2}$$

where $0 = c_{i0} < \ldots < c_{i\tau_i^c}$ are fixed cost coefficients, τ_i^c is the cool-down time of unit i, $c_{i\tau_i^c}$ is its maximum cold-start cost, $u_i := (u_{it})_{t=1}^T$, and $u_{i\tau} \in \{0, 1\}$ for $\tau = 1 - \tau_i^c, \ldots, 0$ are given initial values.

In electric utilities, schedulers forecast the electric load for the required time span. Since the load is mainly driven by meteorological parameters (temperature, cloud cover, etc.), the actual load deviates from its prediction. Of course, the load uncertainty increases with the length of the planning horizon. Other sources of uncertainty are generator outages, stream flows in hydro units, and prices of fuel and electricity.

To formulate a power generation model that incorporates fluctuations in stream inflows in hydro plants, and fuel and electricity prices in addition to the load uncertainty, we use a probabilistic description of uncertainty. Thus

$$\{ \xi_t := (d_t, r_t, \gamma_t, a_t, b_t, c_t) \}_{t=1}^T \tag{3}$$

is assumed to be a discrete-time stochastic process on some probability space $(\Omega, \mathcal{F}, \mathcal{P})$, where d_t, r_t and γ_t represent the load, the spinning reserve and the water inflows in period t, while a_t, b_t and c_t collect the cost coefficients of (1) and (2).

The scheduling decisions for period t are made *after* learning the realization of the stochastic variables for that period. Denote by $\mathcal{F}_t \subseteq \mathcal{F}$ the σ-field generated by $\{\xi_\tau\}_{\tau=1}^t$, i.e., the events observable till period t. Since the information on ξ_1 is complete, $\mathcal{F}_1 = \{\emptyset, \Omega\}$, i.e., ξ_1 is deterministic. By assuming $\mathcal{F}_T = \mathcal{F}$ we require that full information be available at the end of the planning horizon. The sequence of scheduling decisions $\{u_t, p_t, v_t, w_t\}$ also forms a stochastic process on $(\Omega, \mathcal{F}, \mathcal{P})$, which is assumed to be adapted to the filtration of σ-fields, i.e., *nonanticipative.* Nonanticipativity means that the decisions (u_t, p_t, v_t, w_t) may depend only on the data observable till period t, or equivalently that (u_t, p_t, v_t, w_t) is $\mathcal{F}_t$-measurable.

We now assume that we have a *discrete* distribution of the data process $\{\xi_t\}_{t=1}^T$ (cf. (3)). Its support consists of *scenarios* (i.e., realizations of $\{\xi_t\}_{t=1}^T$) that form a *scenario tree* based on a finite set of nodes $\mathcal{N}$. The *root* node $n = 1$ stands for period $t = 1$. Every other node n has a unique *predecessor* node n_- and a *transition* probability $\pi_{n/n_-} > 0$, which is the probability of n being the successor of n_-. The successors to node n form the set $\mathcal{N}_+(n)$; their transition probabilities add to 1. The probability π_n of each node n is generated recursively by $\pi_1 = 1$, $\pi_n = \pi_{n/n_-}\pi_{n_-}$ for $n \neq 1$. Nodes n with $\mathcal{N}_+(n) = \emptyset$ are called *leaves*; they constitute the *terminal* set $\mathcal{N}_T$. A scenario corresponds to a path from the root node to a leaf. The probabilities $\{\pi_n\}_{n\in\mathcal{N}_T}$ provide a distribution for the set of all scenarios. Conversely, given such scenario probabilities, the remaining node and transition probabilities are generated recursively by $\pi_n = \sum_{n_+\in\mathcal{N}_+(n)} \pi_{n_+}$, $\pi_{n_+/n} = \pi_{n_+}/\pi_n$ for $n_+ \in \mathcal{N}_+(n)$.

Let $path(n)$ denote the path from the root to node n. Then node n corresponds to a set of realizations of $\{\xi_t\}_{t=1}^T$ that coincide until the period $t(n) := |path(n)|$ associated with node n; their common value $\xi_{t(n)}$ is denoted by $\xi^n := (d^n, r^n, \gamma^n, a^n, b^n, c^n)$. Let the decisions for period t be made after learning the realization of $\{\xi_t\}_{\tau=1}^t$. The scheduling decisions (u^n, p^n, v^n, w^n) assigned to nodes n in $\mathcal{N}_t := \{n : t(n) = t\}$ are realizations of the stochastic decisions (u_t, p_t, v_t, w_t); note that $\sum_{n\in\mathcal{N}_t} \pi_n = 1$. We denote by $\mathcal{N}_{\text{first}} := \cup_{t=1}^{t_1} \mathcal{N}_t$ the set of *first-stage* nodes, where t_1 is the maximal period such that the data process $\{\xi_t\}_{t=1}^{t_1}$ is deterministic.

Let $u_i^{path(n)} := (u_i^\nu)_{\nu\in path(n)}$. We use the following notation for the sequence of predecessors of any node $n \in \mathcal{N} \setminus \{1\}$: $n_{-1} := n_-$, $n_{-(\kappa+1)} := (n_{-\kappa})_-$ if $t(\kappa) > 1$; note that $t(n_{-\kappa}) = t(n) - \kappa$ for $\kappa = 1, \ldots, t(n) - 1$. To handle initial values of the commitment decisions u_i we let $n_\kappa := \kappa - t(n)$ for $\kappa = t(n) + \tau_{\text{ini}}, \ldots, t(n)$, where $\tau_{\text{ini}} := 1 - \max_{i=1,\ldots,I}\{\,\tau_i^c, \bar{\tau}_i - 1, \underline{\tau}_i - 1\,\}$, and assume that fixed initial values $u_i^{1-\kappa}$ for $\kappa = \tau_{\text{ini}}, \ldots, 0$, are given. Then (cf. (1) and (2))

$$C_i^n(p_i^n, u_i^n) := \max_{l=1:\bar{l}} \{\, a_{il}^n p_i^n + b_{il}^n u_i^n \,\}$$

and

$$S_i^n\left(u_i^{path(n)}\right) := \max_{\tau=0:\,\tau_i^c} c_{i\tau}^n \left(u_i^n - \sum_{\kappa=1}^{\tau} u_i^{n-\kappa}\right) \tag{4}$$

are the fuel and startup costs of unit i at node n.

The *scenario-tree formulation* of the power generation problem reads:

$$\min \sum_{n\in\mathcal{N}} \pi_n \sum_{i=1}^{I} \left[C_i^n(p_i^n, u_i^n) + S_i^n\left(u_i^{path(n)}\right)\right] \quad \text{s.t.} \tag{5}$$

$$p_{it(n)}^{\min} u_i^n \le p_i^n \le p_{it(n)}^{\max} u_i^n, \quad u_i^n \in \{0,1\}, \quad n\in\mathcal{N},\ i=1,\dots,I, \tag{6a}$$

$$u_i^{n-\kappa} - u_i^{n-(\kappa+1)} \le u_i^n, \quad \kappa=1,\dots,\bar{\tau}_i - 1,\ n\in\mathcal{N},\ i=1,\dots,I, \tag{6b}$$

$$u_i^{n-(\kappa+1)} - u_i^{n-\kappa} \le 1 - u_i^n, \quad \kappa=1,\dots,\underline{\tau}_i - 1,\ n\in\mathcal{N},\ i=1,\dots,I, \tag{6c}$$

$$0 \le v_j^n \le v_{jt(n)}^{\max},\ 0 \le w_j^n \le w_{jt(n)}^{\max},\ 0 \le l_j^n \le l_{jt(n)}^{\max},\ n\in\mathcal{N},\ j=1,\dots,J, \tag{7a}$$

$$l_j^n = l_j^{n-} - v_j^n + \eta_j w_j^n + \gamma_j^n, \quad n\in\mathcal{N},\ j=1,\dots,J, \tag{7b}$$

$$l_j^0 = l_j^{\text{in}}, \quad l_j^n = l_j^{\text{end}}, \quad n\in\mathcal{N}_T,\ j=1,\dots,J, \tag{7c}$$

$$\sum_{i=1}^{I} p_i^n + \sum_{j=1}^{J}(v_j^n - w_j^n) \ge d^n, \quad n\in\mathcal{N}, \tag{8a}$$

$$\sum_{i=1}^{I}(u_i^n p_{it(n)}^{\max} - p_i^n) \ge r^n, \quad n\in\mathcal{N}. \tag{8b}$$

Here, (5) is the expected cost, (6) describes the operating ranges and minimum up/down-time requirements of thermal units, (7) models the operating ranges and dynamics of hydro units (with l_j treated as state variables), and (8) imposes the load and reserve requirements. The nonanticipativity constraint is handled *implicitly* (i.e., it is ensured automatically) by the tree-based model. Note that the model (5)–(8) forms a large-scale linear

Table 1. Size of the scenario-tree model (5)–(8) depending on the numbers of scenarios and nodes for $T = 168$, $I = 25$ and $J = 7$

S	N	Variables		Constraints	Nonzeros
		binary	continuous		
1	168	4200	6652	13441	19657
20	1176	29400	45864	94100	137612
50	2478	61950	96642	198290	289976
100	4200	105000	163800	336100	491500

mixed-integer program. For $N := |\mathcal{N}|$ nodes and $S := |\mathcal{N}_T|$ scenarios, this model involves IN binary and $(I+2J)N$ continuous decision variables, and $(2+J)N+JS$ (in)equality constraints and $(I+3J)N$ bounds for continuous variables (without taking into account the constraints of type (6b)–(6c) and the objective function). Table 1 shows how the size of the scenario-tree model (5)–(8) increases with the number of nodes and scenarios.

2.2 Lagrangian Relaxation

Recent algorithmic approaches to large-scale mixed-integer stochastic programs [1,4,5,24,31] are based on a successive decomposition into finitely many smaller subproblems. The stochastic programming model (5)–(8) is almost separable with respect to generation decisions of individual units, since only the constraints (8) couple different units. Hence, when dualizing these coupling constraints, the corresponding Lagrangian dual decomposes into much smaller subproblems, namely, into power generation models for single (thermal or hydro) units. Such a Lagrangian decomposition approach is discussed in [28] for general multistage stochastic integer programs under the label *component decomposition*. In the following, we give a brief description of this approach and of a Lagrangian relaxation algorithm for solving (5)–(8). For a more detailed presentation we refer to [7,15,25].

Let $z := (u,p,v,w)$, $N := |\mathcal{N}|$ and let $\lambda = (\lambda_1, \lambda_2) := (\lambda_1^n, \lambda_2^n)_{n\in\mathcal{N}} \in \mathbb{R}_+^N \times \mathbb{R}_+^N$ denote the Lagrange multiplier in scenario-tree form. Then (cf. [7, §4]) with the *Lagrangian*

$$
\begin{aligned}
L(z\,;\lambda) := \sum_{n\in\mathcal{N}} \pi_n \Bigg\{ & \sum_{i=1}^{I} \Big[C_i^n(p_i^n, u_i^n) + S_i^n \left(u_i^{\mathrm{path}(n)} \right) \Big] \\
& + \lambda_1^n \Big[d^n - \sum_{i=1}^{I} p_i^n - \sum_{j=1}^{J} (v_j^n - w_j^n) \Big] \\
& + \lambda_2^n \Big[r^n - \sum_{i=1}^{I} (u_i^n p_{it(n)}^{\max} - p_i^n) \Big] \Bigg\},
\end{aligned}
\tag{9}
$$

and the *dual function*

$$
D(\lambda) := \min_x \left\{ L(z\,;\lambda) \quad \text{s.t. constraints (6)–(7)} \right\}, \tag{10}
$$

the *dual problem* reads

$$
\max \left\{ D(\lambda) : \lambda \in \mathbb{R}_+^{2N} \right\}. \tag{11}
$$

Under the assumptions made on the fuel costs, the dual function D is concave and polyhedral. The dual problem is separable and its solution may be found by solving stochastic single unit subproblems. Specifically, the dual function

$$D(\lambda) = \sum_{i=1}^{I} D_i(\lambda) + \sum_{j=1}^{J} \hat{D}_j(\lambda_1) + \sum_{n \in \mathcal{N}} \pi_n \left(\lambda_1^n d^n + \lambda_2^n r^n \right), \tag{12}$$

decomposes into the *thermal subproblems*

$$\begin{aligned} D_i(\lambda) = \min_{u_i} \Big\{ \sum_{n \in \mathcal{N}} \pi_n \Big[& \min_{p_i^n} \{ C_i^n(p_i^n, u_i^n) - (\lambda_1^n - \lambda_2^n) p_i^n \} \\ & - \lambda_2^n u_i^n p_{it(n)}^{\max} + S_i^n \left(u_i^{\mathrm{path}(n)} \right) \Big] \text{ s.t. } (6) \Big\}, \end{aligned} \tag{13}$$

and the *hydro subproblems*

$$\hat{D}_j(\lambda_1) = \min_{(v_j, w_j)} \left\{ \sum_{n \in \mathcal{N}} \pi_n \lambda_1^n (w_j^n - v_j^n) \text{ s.t. } (7) \right\}. \tag{14}$$

Both subproblems represent multistage stochastic programming models for the operation of a single unit. While the thermal subproblem (13) is a combinatorial multistage program involving stochastic costs, the hydro subproblem (14) is a linear multistage model with stochastic costs and stochastic right-hand sides.

In short, our method for solving the stochastic programming model (5)–(8) consists of the following ingredients:

(a) Solving the dual problem (11) by a proximal bundle method using function and subgradient information;
(b) Efficient solvers for the single unit subproblems: dynamic programming for (13) and a special descent algorithm for (14);
(c) Lagrange heuristics for determining a nearly optimal first-stage decision.

Thus, the approach is based on the same, but *stochastic*, ingredients as in the classical case: a solver for the nondifferentiable dual, subproblem solvers, and a Lagrangian heuristics. The interaction of these components is illustrated in Figure 1. They have been extensively described in previous studies [15,24] and are now briefly discussed. For a single unit, the hydro subproblem (14) can be solved by stochastic linear programming techniques that are presently available, see, e.g. [2,29]. However, a specialized descent method [24] is found to be more efficient for these problems. The outer minimization of the thermal subproblem (13) with respect to the commitment state u_i is solved by dynamic programming, while the minimization with respect to p_i is carried out explicitly. Since the dual function D is non-differentiable, the dual problem (12) has to be attacked by a subgradient-type method for concave nondifferentable maximization. The reason for employing the proximal bundle method [20] in our algorithm are its very strong convergence properties.

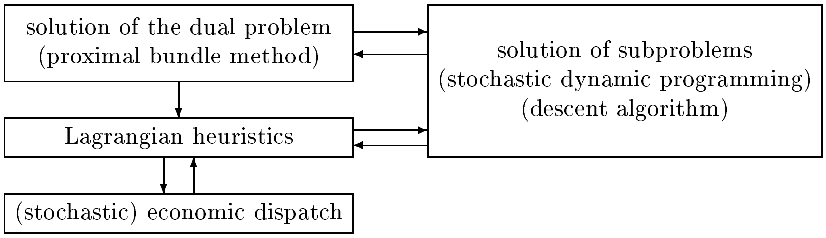

Fig. 1. Structure of the (stochastic) Lagrangian relaxation algorithm

The optimal value $D(\lambda^*)$ for (12) delivered by the bundle method provides a lower bound for the optimal cost of the model (5)–(8). In general, however, the "dual optimal" scheduling decisions $z(\lambda^*) = (u(\lambda^*), p(\lambda^*), v(\lambda^*), w(\lambda^*))$ violate the load and reserve constraints (8) such that a low-cost primal feasible solution has to be determined by a *Lagrangian heuristic*. Two Lagrangian heuristics (cf. [15,24]) have been developed that determine nearly optimal first stage decisions $\{(u^n, p^n, v^n, w^n)\}_{n \in \mathcal{N}_{\text{first}}}$ starting from the optimal multiplier λ^* and $z(\lambda^*)$. While the first heuristics provides a nearly optimal decision only at nodes $n \in \mathcal{N}_{\text{first}}$, the result of the second one is a nearly optimal solution at every node in $\mathcal{N}$.

3 Modeling of Uncertainty

3.1 Introduction

Since the stochastic power management model (5)–(8) uses a set of scenarios to model data uncertainty, new questions are raised on generating approximate scenario-based data processes. Recent scenario tree generation methods ([8]) may essentially be classified into two categories: (a) approaches that are embedded in the solution procedure of stochastic programs [6,11,14,17,19], and (b) approaches that control the goodness-of-fit of the approximation by certain distances [18,27,32]. For power management under uncertainty discrete time stochastic models are calibrated from historical time series for the load and stream flows [13,16,26,31,32]. The calibrated models can be used to simulate or select a large number of sample paths, which are combined into scenario trees. The algorithmic approaches in (a) are computationally demanding, but allow possible updates of the scenario tree structure as part of the solution procedure in the case of linear or convex stochastic programs without integrality constraints. The tree building procedures in (b) generate a tree-structured discrete distribution that minimizes the selected distance (Wasserstein distance of probability measures in [27], ℓ_1- and ℓ_2-distances of certain parameters of the distributions in [18] and [32], resp.).

Our approach to load scenario tree generation for the stochastic power management model proceeds according to the following steps:

1. Identify a statistical model of the load.
2. Generate an initial load tree.
3. Reduce the number of scenarios in the tree optimally.

These steps are explained in the following subsections.

3.2 A Statistical Model for the Electric Load

Description of the Data. The identification of a statistical model for the electric load of the VEAG generation system is based on an hourly load profile for a period of three years (1098 days). A plot of the hourly load data is displayed in Fig. 2. The historical load records show seasonal variations

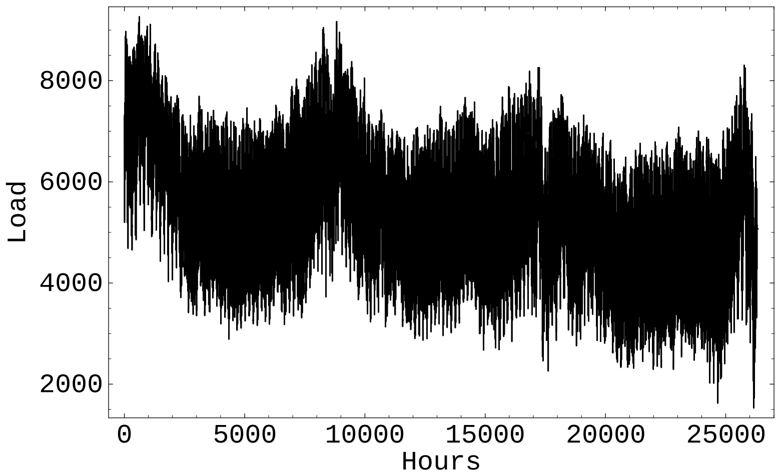

Fig. 2. Hourly load data versus time

caused by meteorological factors like temperature, cloud cover, etc. The periodic patterns complete themselves within the calendar year and are then repeated on a yearly basis. In the weekly and monthly load data there are recurring patterns of length 24 (one day) and of length 168 (one week). Interruptions of this regularity are caused by customs like public holidays or the start/end of the daylight saving time. Thus, in principle the electric load depends on the category of the day (Monday, ..., Sunday, public holiday, etc.) and on the season. Fig. 3 highlights the periodic components of our historical data.

In a first step days of a similar load pattern are identified using daily load records (24 load data of a day). To each record there we assign a day

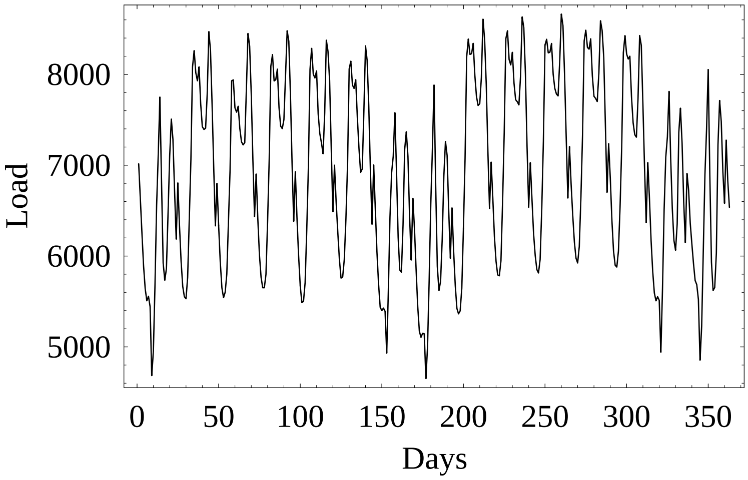

Fig. 3. Hourly load data for a period of two weeks

category (1 if it is a Monday, ..., 7 if it is a Sunday, 8 for a public holiday following a working day, 9 for days between holidays and weekends, 10 for a public holiday following a weekend or a holiday). Clustering methods from [30] are applied to answer the question whether the records can be grouped or classified into useful or informative clusters. After eliminating seasonal effects of the load records, clustering and ANOVA-tests lead to a classification of the load records into 8 categories:

category 1: Monday or working day after a public holiday
category 2: working day (Tuesday, Wednesday, Thursday)
category 3: Friday or working day before a public holiday
category 4: Saturday
categroy 5: Sunday
category 6: public holiday not following days of the categories 2,3
category 7: public holiday following days of the categories 2,3
category 8: working day between days of the categories 4-7

Transformation of Data. The statistical modeling of the load process exploits the decomposition of the load process into a *daily mean load process* and a *mean-corrected load series*. The component series are treated separately. To do this, let $x_{j\tau}$ be the observed load at time step $\tau = 1, \ldots, 24$ of day j (record j of the data base). Due to the daily mean load $\overline{d}_j := (24)^{-1} \sum_{\tau=1}^{24} x_{j\tau}$ and the mean-corrected load record $(d_{j\tau} := x_{j\tau} - \overline{d}_j)_{\tau=1}^{24}$, the historical load records are decomposed according to

$$x_{j\tau} = d_{j\tau} + \overline{d}_j \quad (\tau = 1, \ldots, 24;\ j = 1, \ldots, 1098). \tag{15}$$

The daily mean load series against the day number are plotted in Fig. 4. Fig.

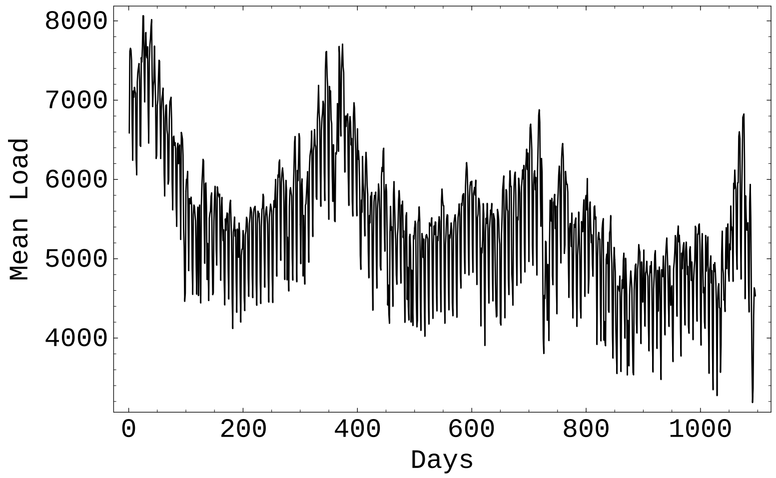

Fig. 4. Daily mean load versus the day number

5 and 6 display the mean-corrected load records $(d_{j\tau})_{\mathrm{cat}(j)=k}$ for days of the categories $k = 1, \ldots, 5$.

Model for the Daily Mean Load. The mean load depends on the category of the day and on the season. Further, there is an interaction between the mean load and meteorological factors like temperature, cloud cover etc. The meteorological impact on the daily mean demand could not be modelled because of missing meteorological parameters.

To select a suitable class of models for the daily mean load series $\{\overline{d}_j\}_{j\in \mathrm{J}}$ with $\mathrm{J} \subset \mathbb{Z} := \{0, \pm 1, \pm 2, \ldots\}$, $\{\overline{d}_j\}_{j\in \mathrm{J}}$ is considered as part of a realization of the stochastic mean load process $\{\overline{d}_j\}_{j\in \mathbb{Z}}$. A *time series model* for $\{\overline{d}_j\}_{j\in \mathrm{J}}$ is a specification of the joint distributions of $\{\overline{d}_j\}_{j\in \mathbb{Z}}$. (The term "time series" is frequently used for both the observed data and the corresponding stochastic process.) We now recall some concepts of time series analysis.

A complete time series model for a stochastic process $\{X_t\}_{t\in \mathbb{Z}}$ should specify the distribution of any random vector $(X_{t_1}, \ldots, X_{t_l})$. Often the analysis focuses on the second-order properties of $\{X_t\}$: the expected values $\mathrm{E}X_t$ and the covariances $\mathrm{cov}(X_t, X_s) := \mathrm{E}[(X_t - \mathrm{E}X_t)(X_s - \mathrm{E}X_s)]$ for all t, s. In the particular case of *Gaussian time series* all random variables X_t are normally distributed. Therefore, all the joint distributions are multivariate normal and completely characterized by the second-order properties of $\{X_t\}$. Classical time series analysis relies on the concept of stationarity. Recall that $\{X_t\}$ is *stationary* if $\mathrm{E}X_t^2 < \infty$, $\mathrm{E}X_t$ is constant and $\mathrm{cov}(X_r, X_s) = \mathrm{cov}(X_{r+t}, X_{s+t})$, $\forall r, s, t \in \mathbb{Z}$.

To select an appropriate class of time series models for observed data, their properties are analyzed first. In particular, the data graph is searched for

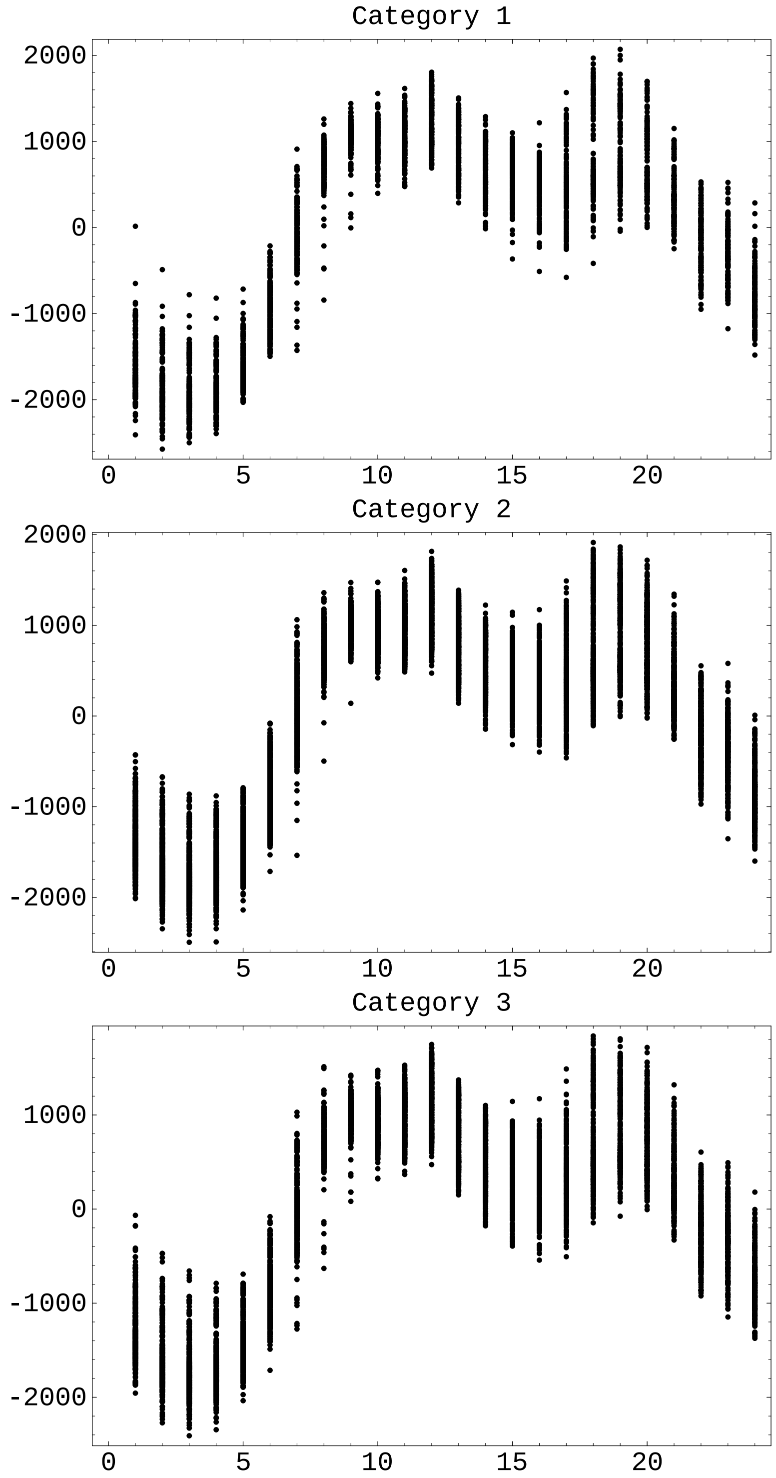

Fig. 5. Mean-corrected load records for days of category 1,2,3

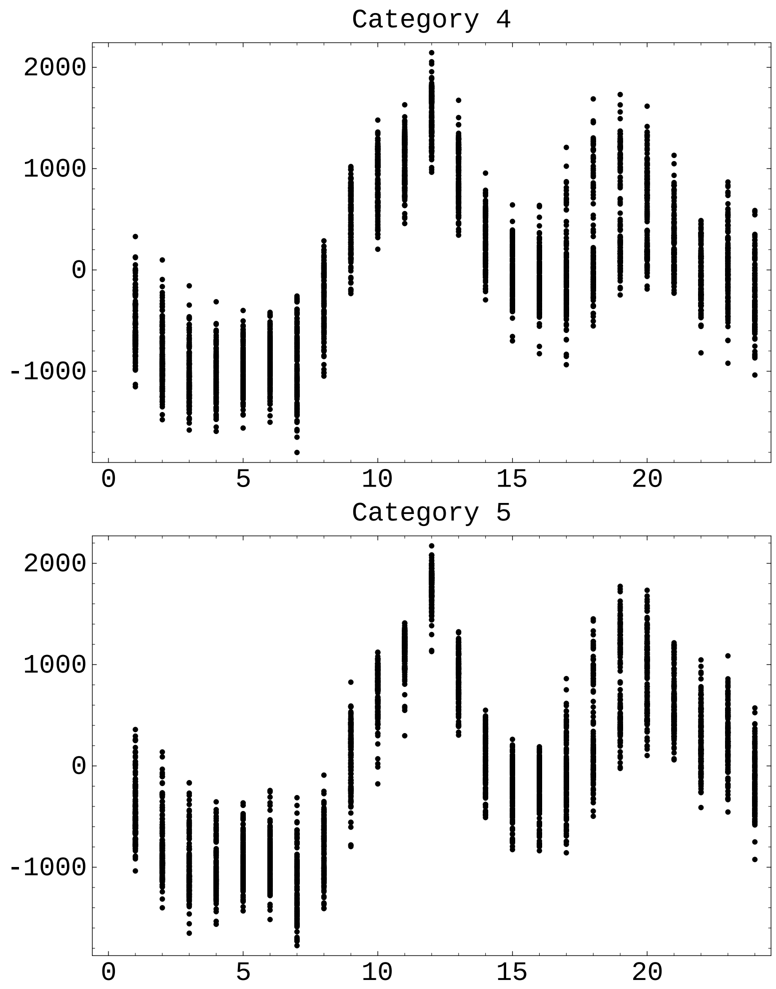

Fig. 6. Mean-corrected load records for days of category 4,5

any *seasonal* (periodic) or *trend* (nonconstant mean) components, outlying observations or sharp changes in behavior. Then suitable transformations are applied to the data to get a new stationary series (*residuals*) with zero mean and unit variance. The trend and seasonal components may be removed by estimating these components and subtracting them from the data; this is the classical decomposition model incorporating trend, a seasonal component and random noise. Another transformation is called *differencing*; it replaces $\{X_t\}$ by $\{Y_t := X_t - X_{t-s}\}$ for some *lag* $s \in \mathbb{N}$, thus eliminating a seasonal component of period s.

In the daily mean load series $\{\overline{d}_j\}_{j\in J}$ there is clearly a recurring pattern with the seasonal period of 365 (one year). There are further periodic components of length 7 (one week) and change points due to the start/end of the daylight saving time. Irregularities of the weekly patterns have been removed from the time series by replacing outlying observations by the value of the nearest day of the same category.

Most approaches for fitting a time series to the deseasonalized data rely on linear models. *Autoregressive moving average* (ARMA) models are characterized by finite-order linear difference equations with constant coefficients. The process $\{X_t\}$ is called *ARMA*(p,q) if it is stationary and

$$X_t - \phi_1 X_{t-1} - \ldots - \phi_p X_{t-p} = Z_t + \theta_1 Z_{t-1} + \ldots + \theta_q Z_{t-q} \quad \forall t, \tag{16}$$

where $(\phi_k)_{k=1}^p$ and $(\theta_l)_{l=1}^q$ are real coefficients and $\{Z_t\}_{t\in\mathbb{Z}}$ is the *white noise* process $\mathrm{WN}(0,\sigma^2)$ with zero mean and variance σ^2, i.e., $\mathrm{E}Z_t = 0$, $\mathrm{E}Z_t^2 = \sigma^2$, $\forall t \in \mathbb{Z}$, and $\mathrm{E}Z_r Z_t = 0$ if $r \neq t$. Using the *backward shift operator* B defined by $B^\ell X_t := X_{t-\ell}$ for $t, \ell \in \mathbb{Z}$, the ARMA equations (16) can be rewritten as

$$\phi(B)X_t = \theta(B)Z_t, \quad \forall t \in \mathbb{Z}, \quad \{Z_t\} \sim \mathrm{WN}(0,\sigma^2),$$

where ϕ and θ denote the polynomials $\phi(z) = 1 - \phi_1 z - \ldots - \phi_p z^p$, $\theta(z) = 1 + \theta_1 z + \ldots + \theta_q z^q$. An ARMA$(p,q)$ process $\{X_t\}_{t\in\mathbb{Z}}$ is said to be *causal* (or *future-independent*) if there exists a real sequence $\{\psi_\ell\}$ such that $\sum_{\ell=0}^\infty \psi_\ell < \infty$ and

$$X_t = \sum_{\ell=0}^{\infty} \psi_\ell Z_{t-\ell}, \quad \forall t \in \mathbb{Z}.$$

If the differenced series $\{Y_t = (1 - B^s)X_t\}_{t\in\mathbb{Z}}$ is an ARMA(p,q) process then the model for the original series $\{X_t\}$ reads $\phi(B)(1-B^s)X_t = \theta(B)Z_t$; further, $\{X_t\}$ belongs to the class of *seasonal autoregressive integrated moving average* (SARIMA) processes if $\{Y_t\}$ is causal. General SARIMA processes are defined as follows. The process $\{X_t\}_{t\in\mathbb{Z}}$ is said to be a *SARIMA*$(p,d,q) \times (P,D,Q)_S$ *process with period* s if the differenced process $Y_t := (1-B)^d(1-B^S)^D X_t$ is the causal ARMA process

$$\phi(B)\Phi(B^s)Y_t = \theta(B)\Theta(B^s)Z_t, \quad \{Z_t\} \sim WN(0,\sigma^2),$$

where $\phi(z) = 1 - \ldots - \phi_p z^p$, $\Phi(z) = 1 - \ldots - \Phi_P z^P$, $\theta(z) = 1 + \ldots + \theta_q z^q$ and $\Theta(z) = 1 + \ldots + \Theta_Q z^Q$. Then the model for $\{X_t\}_{t\in\mathbb{Z}}$ reads $\phi(B)\Phi(B^S)(1 - B)^d(1 - B^S)^D X_t = \theta(B)\Theta(B^S)Z_t$.

To identify a suitable SARIMA model for the given time series, the differencing orders d, D, the model orders p, P, q, Q and the length S of the seasonal component must be identified. They can be discovered by inspecting the empirical autocorrelation function, the empirical counterpart of the *autocorrelation function* $\mathrm{cov}(X_\ell, X_0)/\mathrm{var}(X_0)$, $\ell \in \mathbb{Z}$; see, e.g., [3]. The model coefficients $(\phi_\ell)_{\ell=1}^p$, $(\Phi_\ell)_{\ell=1}^P$, $(\theta_\ell)_{\ell=1}^q$, $(\Theta_\ell)_{\ell=1}^Q$ and the white noise variance σ^2 can be estimated via parameter estimation procedures for ARMA processes. The maximum likelihood method produces the most efficient estimates in the particular case of Gaussian time series. Initial values for the model coefficients can be obtained by the Hannan-Rissanen algorithm (cf. [3, §5]) which solves the problem of order selection and parameter estimation for ARMA processes simultaneously.

In our case, we obtained stationary residuals after three differencing operations (two lag-364 differencing operations followed by one lag-1 differencing). The residuals were treated as part of a realization of the stochastic process

$$\{Y_j := \overline{d}_j - \overline{d}_{j-1} - 2\overline{d}_{j-364} + 2\overline{d}_{j-365} + \overline{d}_{j-728} - \overline{d}_{j-729}\}.$$

For $\{Y_j\}$ the Hannan-Rissanen algorithm from the *Mathematica Time Series Pack* [33] selected an ARMA(1,1) model that served as an initial model for the maximum likelihood method. The resulting maximum likelihood estimates for the model coefficients and random noise process are

$$\hat{\phi}_1 = 0.357756, \quad \hat{\theta}_1 = -0.639978, \quad \{Z_j\} \sim N(0, 15533.88), \quad j \in \mathbb{Z}.$$

The time series model for $\{Y_j\}$ reads $Y_j - \hat{\phi}_1 Y_{j-1} = Z_j + \hat{\theta}_1 Z_{j-1}$, $j \in \mathbb{Z}$. Accordingly, the time series model for the daily mean load process $\{\overline{\mathbf{d}}_j\}_{j\in\mathbb{Z}}$ is

$$(1 - B)(1 - B^{364})^2(1 - \hat{\phi}_1 B)\overline{\mathbf{d}}_j = (1 + \hat{\theta}_1 B)Z_j. \tag{17}$$

The above SARIMA$(1, 1, 1) \times (0, 2, 0)_{364}$ model can be converted to the ARMA$(730, 1)$ model

$$\begin{aligned}\overline{d}_j &- (1 + \hat{\phi}_1)\overline{d}_{j-1} + \hat{\phi}_1\overline{d}_{j-2} - 2\overline{d}_{j-364} + 2(1 + \hat{\phi}_1)\overline{d}_{j-365} - 2\hat{\phi}_1\overline{d}_{j-366} \\ &+ \overline{d}_{j-728} + (\hat{\phi}_1 - 1)\overline{d}_{j-729} + \hat{\phi}_1\overline{d}_{j-730} = Z_j + \hat{\theta}_1 Z_{j-1}, \quad j \in \mathbb{Z}.\end{aligned} \tag{18}$$

Model for the Mean-Corrected Load Records. We have selected polynomials to model the time dependence of the mean-corrected load records corresponding to days of the same category k, $k = 1, \ldots, 7$. More specifically, we fit models of the form

$$d_{k\tau} = \beta_{k0} + \beta_{k1}\tau + \ldots + \beta_{km_k}\tau^{m_k} + \epsilon_{km_k}, \; (k = 1, \ldots, 5; \tau = 1, \ldots, 24), \tag{19}$$

where the *error term* e_{km_k} is normally distributed with zero mean and variance $\sigma^2_{m_k}$. These models are known as *linear* or *polynomial regression models* (cf. [10]). From (19) we obtain the *predicted load records*

$$\hat{d}_{k\tau} = \beta_{k0} + \beta_{k1}\tau + \ldots + \beta_{km_k}\tau^{m_k}, \ (k = 1, \ldots, 5; \tau = 1, \ldots, 24). \qquad (20)$$

For model fitting, regression diagnostics, and forecasting we used the statistical package *S-PLUS* [30]. It remains to answer the question how we selected the degree m_k of the polynomials. This will be done in the following subsection.

Model for the Load. The statistical model for the load is obtained by combining the models for the daily mean load and the mean-corrected load records according to (15). Regression models for the mean-corrected load records corresponding to different day categories may be included into (15) by using *dummy or artificial variables* D_k, $k = 1, \ldots, 7$. These variables are defined as follows.

$$D_k := \begin{cases} 1, \text{ if the record corresponds to a day of category } k, \\ 0, \text{ otherwise,} \end{cases} \qquad (k = 1, \ldots, 7).$$

With these definitions (15) may be rewritten as

$$x_{j\tau} = D_1 d_{1\tau} + \ldots + D_7 d_{7\tau} + \overline{d}_j, \quad (\tau = 1, \ldots, 24; j = 1, \ldots, 1098). \quad (21)$$

The different time scales for the historical load records and the load process can be synchronized by an index transformation:

$$d_t = D_1 d_{1,t\%24} + \ldots + D_7 d_{7,t\%24} + \overline{d}_{[t/24]}, \quad (t \in \mathbb{Z}). \qquad (22)$$

(By $[t/24]$ and $t\%24$ we denote the (rounded down) integral part and the remainder of dividing t by 24.)

Inserting (18) and (19) into (22) we obtain the *statistical model for the load*:

$$\begin{aligned} d_t = {} & D_1 \sum_{l=0}^{m_1} \beta_{1l}(t\%24)^l + \ldots + D_7 \sum_{l=0}^{m_7} \beta_{7l}(t\%24)^l \qquad (23) \\ & + (1 + \hat{\phi}_1)\overline{d}_{[t/24]-1} - \hat{\phi}_1 \overline{d}_{[t/24]-2} + 2\overline{d}_{[t/24]-364} - 2(1 + \hat{\phi}_1)\overline{d}_{[t/24]-365} \\ & + 2\hat{\phi}_1 \overline{d}_{[t/24]-366} - \overline{d}_{[t/24]-728} - (\hat{\phi}_1 - 1)\overline{d}_{[t/24]-729} - \hat{\phi}_1 \overline{d}_{[t/24]-730} \\ & + D_1 \epsilon_{1m_1} + \ldots + D_7 \epsilon_{7m_7} + Z_j + \hat{\theta}_1 Z_{[t/24]-1}, \ (t \in \mathbb{Z}). \end{aligned}$$

To select the degrees m_k of the regression polynomials we measured the squared distance between (23) and the historical load data for the third year. The best fit we obtained for $m_k = 10$, $k = 1, \ldots, 7$.

3.3 Generation of Load Scenario Trees

An important initial decision is the choice of the number of stages and of the branching scheme for the scenario tree, i.e., the number and positions of branching levels and the branching degree in every node. We choose the following initial structure of the load scenario tree:

- A balanced tree with $K = 6$ branching periods t_k, $k = 1, \ldots, K$. The branching periods t_k, $k = 2, \ldots, K$, are equidistant within the time span $t = t_1, \ldots, T$, i.e., $t_k := t_1 + (T - t_1)(k-1)/K$, $k = 2, \ldots, K$.
- $|\mathcal{N}_+(n)| = \begin{cases} 2, n \in \mathcal{N}_{t_k} = \{n : t(n) = t_k\}, \ k = 1, \ldots, K, \\ 1, \text{otherwise.} \end{cases}$

Thus, the tree consists of $S := 2^6 = 64$ scenarios. The first branching period t_1 is defined by the length of the planning period within the scheduling horizon. The branching periods t_k, $k = 2, \ldots, K$, should correspond to the (normally fixed) times when already observable meteorological and load data provide the possibility to re-adjust the unit commitment. For the scheduling horizon of one week with an hourly discretization, $t_k = 24k$ for $k = 1, \ldots, 6$ is a reasonable choice for the generation system of the utility VEAG. For longer scheduling periods, non-equidistant branching periods would be preferable in order to restrict the number of scenarios. By assigning two successors to any node n in $\mathcal{N}_{t_k}$, $k = 1, \ldots, K$, we may distinguish the events "low load" and "high load" for periods $t = t_k + 1, \ldots, t_{k+1}$, where $t_{K+1} := T$. An additional event such as "medium load" would increase the scenario number to $S = 3^6 = 729$.

It remains to specify the realizations (scenario values) and their probabilities. Suppose the power utility supplies starting values $(d_t)_{t=-729}^{1}$. The realizations for nodes of the first stage period are given by

$$d^n := D_1 \hat{d}_{1t(n)\%24} + \ldots + D_7 \hat{d}_{7t(n)\%24} + \overline{d}_1, \ t(n) = 1, \ldots, t_1,$$

their probabilities $\pi^n := 1$. To assign realizations and probabilities to the remaining nodes we first compute the probability distribution of $(d_t)_{t=t_1+1}^{T}$. By (23), $(d_t)_{t=t_1+1}^{T}$ has a $(T - t_1)$-variate normal distribution with mean $\mu := (\mathrm{E}d_t)_{t=t_1+1}^{T}$ and covariance matrix $\Sigma := (\mathrm{cov}(d_t, d_s))_{s,t=t_1+1}^{T}$. The following properties reduce the computational effort. First,

$$\mathrm{E}(d_t - \sum_{l=1}^{7} D_l \hat{d}_{lt\%24}) = \mathrm{E}(d_{t_k} - \sum_{l=1}^{7} D_l \hat{d}_{lt_k\%24}) \tag{24}$$

for $t = t_{k-1} + 1, \ldots, t_k$, $k = 2, \ldots, K$. Second, the random vector $(v_t := d_t - D_1 \hat{d}_{1t\%24} - \ldots - D_7 \hat{d}_{7t\%24})_{t_1+1}^{T}$ has a $(T - t_1)$-variate normal distribution with mean $\hat{m} := \mu - \sum_{l=1}^{7} D_l \hat{d}_{lt_k\%24}$ and covariance matrix Σ. By (23) and (24), $\hat{m}$ and Σ are completely determined by $\mathrm{E}d_{t_k}$ and $\mathrm{cov}(d_{t_k}, d_{t_l})$ for $k, l = 1, \ldots, K$.

With $t_k = 24k$, $\mathrm{E}d_{t_k} = \mathrm{E}\overline{d}_k$ and

$$\mathrm{cov}(d_{t_k}, d_{t_l}) = \mathrm{cov}(\sum_{\nu=1}^{7} D_\nu \epsilon_{\nu m_\nu}, \sum_{\xi=1}^{7} D_\xi \epsilon_{\xi m_\xi}) + \mathrm{cov}(\overline{d}_k, \overline{d}_l)$$

for $k, l = 1, \ldots, K$. To compute $\mathrm{E}\overline{d}_k$ and $\mathrm{cov}(\overline{d}_k, \overline{d}_l)$ we use the following lemma:

Lemma 1. *Let random variables X_t, $t \geq 0$, be defined by the ARMA equation (16) with starting values $(X_{-p}, \ldots, X_{-1}) \in \mathrm{I\!R}^p$ and $Z_t \sim N(0, \sigma^2)$, $t \geq -q$. For $s, t \geq 0$ the expected value $\mathrm{E}X_t$ and the covariance $\mathrm{cov}(X_s, X_t)$ are given by*

$$\mathrm{E}X_t = \begin{cases} \sum_{k=0}^{t} a_{t-k} \left(\sum_{m=1+k}^{p} \Phi_m X_{-m+k}\right), \ t = 0, \ldots, p-1 \\ \sum_{k=0}^{p-1} a_{t-k} \left(\sum_{m=1+k}^{p} \Phi_m X_{-m+k}\right), \ t \geq p \end{cases},$$

$$\mathrm{cov}(X_s, X_t) = \sigma^2 \sum_{k=0}^{t} \sum_{m=0}^{s} a_{t-k} a_{s-m} \sum_{l=0}^{q-|k-m|} \Theta_l \Theta_{l+|k-m|},$$

where $\{a_t\}_{t=0}^{\infty}$ is defined by

$$a_0 := 1, \ \ a_k := \begin{cases} \Phi_1 a_{k-1} + \Phi_2 a_{k-2} + \ldots + \Phi_k a_0, & 1 \leq k \leq p \\ \Phi_1 a_{k-1} + \Phi_2 a_{k-2} + \ldots + \Phi_p a_{k-p}, & k > p \end{cases}.$$

Proof. By induction.

Using the computer algebra system *Mathematica* [33], the mean $\hat{m}$ and the covarince matrix Σ of the transformed random vector $(v_t)_{t=t_1+1}^{T}$ can be computed within a few seconds. The mean μ of $(d_t)_{t=t_1+1}^{T}$ can be obtained by the transformation $\mu = \hat{m} + \sum_{l=1}^{7} D_l \hat{d}_{l t_k \% 24}$.

After computing the distribution of $(d_t)_{t=t_1+1}^{T}$, a scenario tree for its approximation is constructed in a standard way. Suppose that we already assigned realizations and probabilities to all nodes of a path from the root to node $n \in \mathcal{N}$. We distinguish two cases.

(a) $t(n) \neq t_k$, $k = 1, \ldots, K$.
Then there is a single successor n_+ to n with probability $\pi_{n+} := \pi_n$ and realization $d^{n+} := \mathrm{E}(d_{t(n)+1} | d^n \ldots d^1)$, the conditional mean of $d_{t(n)+1}$ given the past realizations $d_1 = d^1, \ldots, d_{t(n)} = d^n$.

(b) $t(n) = t_k$, for some $k \in \{1, \ldots, K\}$.
Let $\mathcal{N}_+(n) = \{n_1, n_2\}$, i.e., n_1, n_2 are the successors to node n. The conditional (normal) distribution of d_{t_k+1} given $d_1 = d^1, \ldots, d_{t_k} = d^n$ is approximated by a discrete distribution with the two realizations

$$d^{n_1} = d^n + \delta_n + \sum_{l=1}^{7} D_l (\hat{d}_{l(k+1)} - \hat{d}_{lk}), \ \ d^{n_2} = d^n - \delta_n + \sum_{l=1}^{7} D_l (\hat{d}_{l(k+1)} - \hat{d}_{lk}).$$

A criterion for determining the innovation δ_n and the transition probability $\pi_{n_1/n}$ ($\pi_{n_2/n} = 1 - \pi_{n_1/n}$) is that their choice preserves the first two moments of the conditional distribution for d_{t_k+1} given the past realizations $d_1 = d^1, \ldots, d_{t(n)} = d^n$ ([23]). With the conditional mean $\mu_n := \mathrm{E}(d_{t_k+1}|d^n \ldots d^1)$ and the conditional variance $\sigma_n^2 := \mathrm{var}(d_{t_k+1}|d^n \ldots d^1)$, the innovation and the transition probability satisfies

$$\delta_n = \sqrt{(\mu_n - d^n)^2 + \sigma_n^2}, \quad \pi_{n_1/n} = \frac{(\mu_n - d^n)^2 + \sigma_n^2 + \delta_n(\mu_n - d^n)}{2\delta_n^2}.$$

The node probabilites π^{n_1} and π^{n_2}, resp. are recursively computed from the relations $\pi^{n_1} = \pi_{n_1/n}\pi^n$ and $\pi^{n_2} = (1 - \pi_{n_1/n})\pi^n$.

Fig. 7 and 8 show load scenario trees for a planning horizon of one week with an hourly discretization and branching periods $t_k = 24k, k = 1, \ldots, 6$. For the summer week (Fig. 7) the scenario probabilities vary between 0.37 and 10^{-10}, for the winter week (Fig. 8) between 0.7 and 10^{-13}. The generation of one load scenario tree took less than two minutes on an HP 9000 (780/J280) Compute-Server with 180 MHz frequency and 768 MByte main memory under HP-UX 10.20.

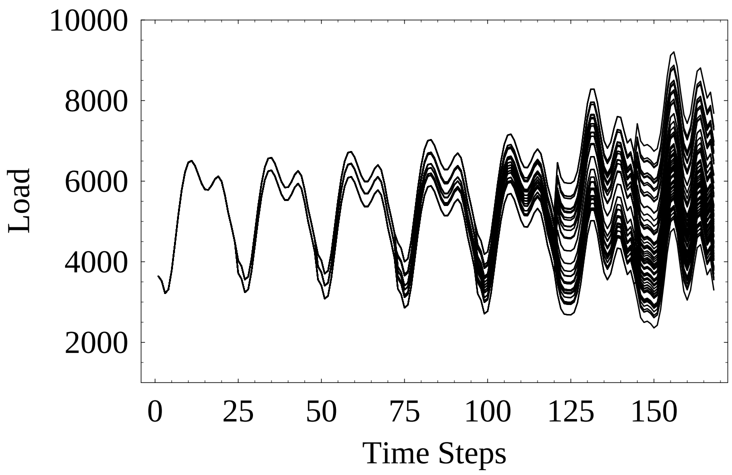

Fig. 7. Load scenario tree for one week in summer

A few comments on the tree construction procedure are in order. First, the construction of the load scenario tree is consistent with the normality assumption imposed on $(d_t)_{t=t_1+1}^T$ by the statistical model (23). $\mathcal{N}$ preserves the mean and covariance matrix. In particular, the scenario tree resembles the correlation structure of the time series model. Second, in general the two

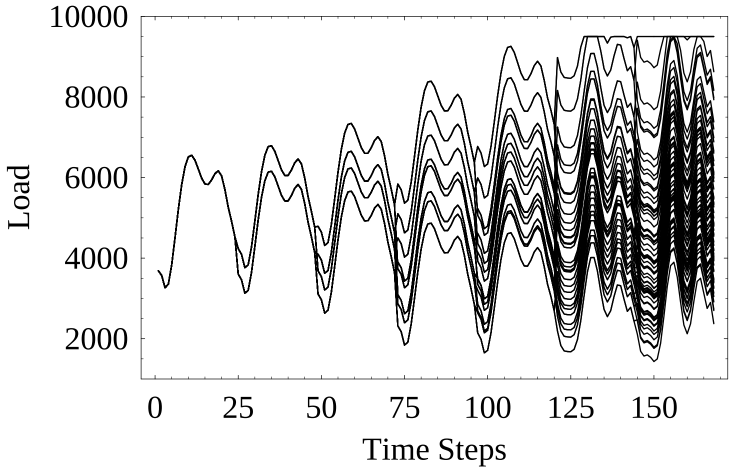

Fig. 8. Load scenario tree for one week in winter

transistion probabilities for a branching point are different. As a consequence, the probabilities of the scenarios in the tree differ. Third, the construction of the load scenario tree does not prevent unrealistic ("too large") load values. Load values may exceed the maximum capacity of the thermal system. Empirical results have shown that they are related to very small node probabilities ($< 10^{-10}$). In order to avoid computational difficulties these load values are replaced by the maximum capacity of the thermal system.

3.4 Optimal Reduction of Load Scenario Trees

The initial scenario tree generated in Section 3.3 may contain too many nodes to solve the power management model (5)–(8) within acceptable time. Therefore, one often incorporates a procedure to reduce the number of scenarios of the initial tree [9,34].

The load scenario tree provides a first approximation of the distribution of the discrete-time stochastic load process. Our reduction concept determines a scenario subset of prescribed cardinality and a probability measure based on this set that is closest to the initial approximation in terms of a natural (or canonical) probability metric. Quantitative stability results for stochastic programs (cf. [9]) indicate that the Fortet-Mourier metrics ζ_h, $h \geq 1$, are canonically associated for a multistage stochastic program like (5)–(8).

Let δ_ω denote the probability measure on $\mathbb{R}^T$ having unit mass at $\omega \in \mathbb{R}^T$. The initial scenario tree represents a discrete probability distribution P carried by scenarios $\omega_i \in \Omega$ with weights $p_i > 0$, $\sum_{i=1}^{S} p_i = 1$. A reduced scenario tree is obtained by deleting all scenarios ω_j, $j \in J$ belonging to some index set $J \subset \{1, \ldots, S\}$ and by assigning new probabilistic weights

q_j to each scenario ω_j, $j \notin J$. Let Q denote the corresponding probability distribution, i.e.

$$P = \sum_{i=1}^{S} p_i \delta_{\omega_i}, \quad Q = \sum_{j \notin J} q_j \delta_{\omega_j}.$$

The Fortet-Mourier metric ζ_h, $h \geq 1$, of the discrete probability measures P, Q may be derived by solving the dual transportation problem

$$\zeta_h(P,Q) = \max \left\{ \sum_{i=1}^{S} p_i u_i + \sum_{j \notin J} q_j v_j \; : \; u_i + v_j \leq c_h(\omega_i, \omega_j) \right\},$$

where $c_h(\omega, \omega') := \max\{1, \|\omega\|^{h-1}, \|\omega'\|^{h-1}\}\|\omega - \omega'\|$ and $\|\cdot\|$ is the Euclidean norm on $\mathbb{R}^T$. Since the two measures P and Q have the same support $\{\omega_s\}_{s=1}^{S}$, but different weights, upper and lower bounds for $\zeta_h(P,Q)$ can be derived [9].

Now, let D_J be the distance of P to a closest probability distribution having support $\{\omega_s : \; s = 1, \ldots, S, \; s \notin J\}$, i.e., corresponding to deleting all scenarios of P belonging to some index set J:

$$D_J := \min\{\zeta_h(\sum_{i=1}^{S} p_i \delta_{\omega_i}, \sum_{j \notin J} q_j \delta_{\omega_j}) \; : \; q_j \geq 0 \,, \sum_{j \notin J} q_j = 1\}.$$

Then we have (cf. [9])

$$D_J = \sum_{i \in J} p_i \min_{j \notin J} c_h(\omega_i, \omega_j).$$

The optimal weights $(q_j)_{j \notin J}$ for the scenarios remainimg in the reduced tree are

$$\bar{q}_j = p_j + \sum_{\substack{i \in J \\ j(i)=j}} p_i \;\; \forall j \notin J \tag{25}$$

where $j(i) \in \arg\min_{j \notin J} c_h(\omega_i, \omega_j)$ for $i \in J$.

An optimal rule for reducing P to a measure Q with a prescribed number $\tilde{S}$ of scenarios is given as the solution of the combinatorial optimization problem

$$\min\{D_J = \sum_{i \in J} p_i \min_{j \notin J} c_h(\omega_i, \omega_j) \; : \; J \subset \{1, ..., S\} \,, \#J = S - \tilde{S}\}. \tag{26}$$

Explicit solutions to (26) are available for the cases $\tilde{S} = S - 1$ (single scenario deletion) and $\tilde{S} = 1$ (keeping only one scenario) [9]. Upper and lower bounds for (26) yield heuristic reduction strategies for the general case. The *forward selection* algorithm recusively determines the indices u_j, $j = 1, \ldots, \tilde{S}$ for the scenarios in the reduced tree. It uses the lower bound

$$\min\{D_J : J \subset \{1, ..., S\} \,, \#J = S - \tilde{S}\} \leq \sum_{i \in J_u} p_i \min_{j \notin J_u} c_h(\omega_i, \omega_j),$$

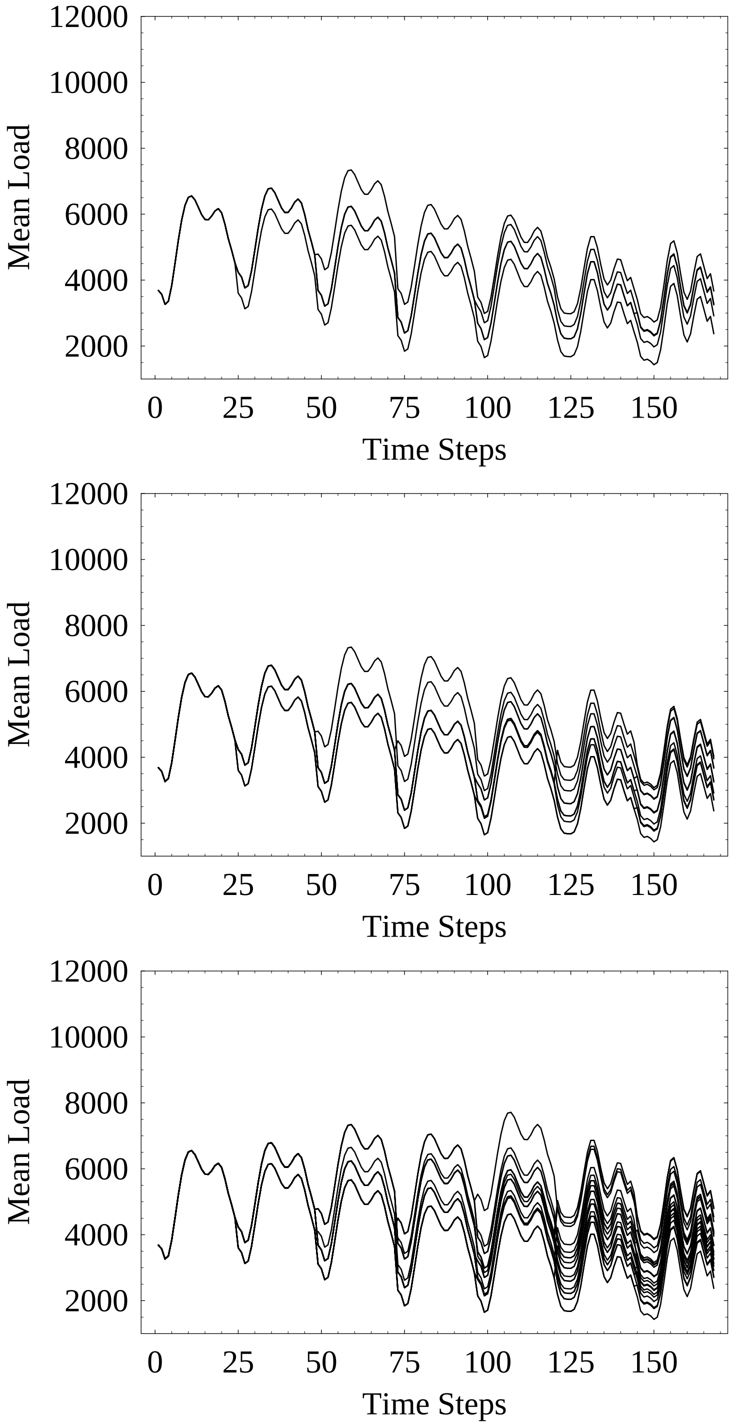

Fig. 9. Reduced load scenario tree with 5, 10 and 20 scenarios

where $J_u = \{1, \dots, S\} \setminus \{u_1, \dots, u_{\tilde{S}}\}$. The indices u_j, $j = 1, \dots, \tilde{S}$ are chosen recursively such that

$$u_j \in \arg \min_{l \notin \{u_1, \dots, u_{j-1}\}} \sum_{\substack{i=1 \\ i \notin \{u_1, \dots, u_{j-1}, l\}}}^{S} p_i \min_{u \in \{u_1, \dots, u_{j-1}, l\}} c_h(\omega_u, \omega_i).$$

Optimal probabilistic weights $(q_j)_{j \notin J_u}$ are computed from (25) (with J replaced by J_u). Fig. 9 shows the reduced load scenario trees with 5, 10 and 20 scenarios obtained by the forward selection algorithm with $c_h(\omega, \omega') := \|\omega - \omega'\|$.

3.5 Numerical Testing Results

For numerical tests we considered the hydro-thermal power system of VEAG (with $T = 168$, $I = 25$ and $J = 7$) under uncertain load (i.e., the remaining data were deterministic). Previous tests [15,24] of our implementation of the stochastic Lagrangian relaxation algorithm have proved its potential for solving the stochastic power management model within a few minutes.

In this testing, we combined the stochastic Lagrangian relaxation algorithm with the tree generation and reduction technique. The methods for solving the subproblems and the heuristics are kept identical. Again, the test runs were performed on an HP 9000 (780/J280) Compute-Server with 180 MHz frequency and 768 MByte main memory under HP-UX 10.20. First we generated a load scenario tree (cf. §3.3) for an hourly discretized time horizon of one week ($T = 168$) with branching points $t_k = 24k$, $k = 1, \dots, 6$ (cf. Fig. 8). The initial number of scenarios $S = 64$ was reduced by applying the forward scenario selection rule of §3.4 with $c_h(\omega, \omega') := \|\omega - \omega'\|$. Due to the nonconvexity of the underlying stochastic programs, primal solutions for (5)–(8) are obtained by a heuristic method. This is the reason for comparing the solutions of the dual problem. Table 2 reports the objective and computing time of the dual stochastic problem (11) for different numbers of scenarios ($\tilde{S}$) in the reduced tree, each having a different number of nodes (N). The reduced trees with $\tilde{S} = 5, 10, 20$ have been presented in Fig. 9. It can be seen that the optimal values of (11) for reduced scenario trees slowly converge to the objective of the full scenario tree problem ($S = 64$). Rough approximations of the full scenario tree generally require only short computing times. For $\tilde{S} < 10$ the scenarios are highly concentrated and the true objective is therefore underestimated. The approximation of the initial scenario tree improves for $\tilde{S} > 20$, resulting in a better approximation of the objective.

Acknowledgement. The authors wish to thank W. Römisch (Humboldt-University Berlin) for many useful suggestions. We are grateful to G. Schwarzbach, J. Thomas and J. Krause (VEAG Vereinigte Energiewerke AG Berlin)

Table 2. Objective values and computing times of the dual stochastic problem (11) for different numbers of scenarios ($\tilde{S}$) in the reduced tree

$\tilde{S}$	N	objective	time [s]	$\tilde{S}$	N	objective	time [s]
1	168	2.83589e+07	10.83	15	1166	2.84102e+07	109.40
2	313	2.81617e+07	28.41	20	1387	2.84189e+07	141.59
3	386	2.82691e+07	28.71	25	1656	2.84164e+07	186.07
4	411	2.83139e+07	35.32	30	1829	2.84159e+07	186.32
5	532	2.83874e+07	40.33	35	2122	2.84146e+07	259.52
6	581	2.83942e+07	57.84	40	2295	2.84248e+07	301.57
7	678	2.83964e+07	55.95	45	2492	2.84219e+07	276.72
8	703	2.83999e+07	58.71	50	2689	2.84149e+07	321.04
9	728	2.84124e+07	63.24	55	2838	2.84207e+07	379.66
10	801	2.84071e+07	83.61	60	2987	2.84181e+07	374.44
Initial tree:				64	3111	2.84210e+07	487.27

for their outstanding cooperation over many years. Further thanks are due to J. Dupačová (Charles University Prague) and her former students A. Henclová, M. Kotyzová and J. Muková for the collaboration in modeling load profiles and to K.C. Kiwiel (Polish Academy of Sciences, Warsaw) for the permission to use the NOA 3.0 package.

References

1. Bacaud, L., Lemaréchal, C., Renaud, A., Sagastizábal, C.: Bundle methods in stochastic optimal power management: A disaggregated approach using preconditioners. Comput. Optim. Appl. (to appear)
2. Birge, J.R.: Stochastic programming computation and applications. INFORMS J. Comput. **9** (1997) 111–133
3. Brockwell, P.J., Davies, R.A.: Introduction to Time Series and Forecasting. Springer, New York, 1996
4. Carøe, C.C., Schultz, R.: Dual decomposition in stochastic integer programming. Oper. Res. Lett. **24** (1999) 37–45
5. Carpentier, P., Cohen, G., Culioli, J.-C., Renaud, A.: Stochastic optimization of unit commitment: A new decomposition framework. IEEE Trans. Power Systems **11** (1996) 1067–1073
6. Consigli, G., Dempster, M.A.H.: Dynamic stochastic programming for asset-liability management. Ann. Oper. Res. **81** (1998) 131–161
7. Dentcheva, D., Römisch, W.: Optimal power generation under uncertainty via stochastic programming. Stochastic Programming Methods and Technical Applications, K. Marti and P. Kall, eds., Lecture Notes in Economics and Mathematical Systems 458, Springer-Verlag, Berlin, 1998, pp. 22–56
8. Dupačová, J., Consigli, G., Wallace, S.W.: Scenarios for multistage stochastic programs. Ann. Oper. Res. **100** (2001) (to appear)

9. Dupačová, J., Gröwe-Kuska, N., Römisch, W.: Scenario reduction in stochastic programming: An approach using probability metrics. Preprint 00-09, Institut für Mathematik, Humboldt-Univ. Berlin, Berlin, Germany, 2000
10. Draper, N., Smith, H.: Applied Regression Analysis. Hafner, New York, second ed., 1981
11. Edirisinghe, N.C.P.: Bound-based approximations in multistage stochastic programming: Nonanticipativity aggregation. Ann. Oper. Res. **85** (1999) 173–192
12. Feltenmark, S., Kiwiel, K.C.: Dual applications of proximal bundle methods, including Lagrangian relaxation of nonconvex problems. SIAM J. Optim. **10** (2000) 697–721
13. Fleten, S.-E., Wallace, S.W., Ziemba, W.T.: Hedging electricity portfolios via stochastic programming. IMA Volumes in Mathematics and its Applications, Springer-Verlag (to appear)
14. Frauendorfer, K.: Barycentric scenario trees in convex multistage stochastic programming. Math. Programming **75** (1996) 277–293
15. Gröwe-Kuska, N., Kiwiel, K.C., Nowak, M.P., Römisch, W., Wegner, I.: Power management in a hydro-thermal system under uncertainty by Lagrangian relaxation. Preprint 99-19, Institut für Mathematik, Humboldt-Universitt Berlin, 1999 and IMA Volumes in Mathematics and its Applications, Springer-Verlag (to appear)
16. Henclová, A., Kotyzová, M., Muková, J.: Analysis of the demand for electric energy. Semainar work (in Czech), Department of Probability and Mathematical Statistics, Charles University Prague, 2000
17. Higle, J.L., Sen, S.: Stochastic Decomposition; A Statistical Method for Large Scale Stochastic Linear Programming. Kluwer, Dordrecht, 1996
18. Høyland, K., Wallace, S.W.: Generating scenario trees for multi-stage decision problems. Management Sci. (to appear)
19. Infanger, G.: Monte Carlo (importance) sampling within a Bender's decomposition for stochastic linear programs. Ann. Oper. Res. **39** (1992) 69–95
20. Kiwiel, K.C.: Proximity control in bundle methods for convex nondifferentiable minimization. Math. Programming **46** (1990) 105–122
21. Kiwiel, K.C.: User's guide for NOA 3.0: A Fortran package for convex nondifferentiable optimization. tech. rep., Systems Research Institute, Warsaw, 1994
22. Lemaréchal, C.: Lagrangian decomposition and nonsmooth optimization: Bundle algorithm, prox iteration, augmented Lagrangian. Nonsmooth Optimization, Methods and Applications, F. Giannessi, ed., Gordon and Breach, Philadelphia, 1992, pp. 201–216
23. Miller, A.C., Rice, T.R.: Discrete approximations of probability distributions. Management Sci. **29** (1983) 352–362
24. Nowak, M.P.: Stochastic Lagrangian Relaxation in Power Scheduling of a Hydro-Thermal System Under Uncertainty. PhD thesis, Institut für Mathematik, Humboldt-Univ. Berlin, Berlin, Germany, 2000
25. Nowak, M.P., Römisch, W.: Stochastic Lagrangian relaxation applied to power scheduling in a hydro-thermal system under uncertainty. Ann. Oper. Res. **100** (2001) (to appear)
26. Philpott, A.B., Craddock, M., Waterer, H.: Hydro-electric unit commitment subject to uncertain demand. European Journal of OR 125(2000), 410–424

27. Pflug, G.C.: Scenario tree generation for multiperiod financial optimization by optimal discretization. Math. Programming **89** (2001) 251–271
28. Römisch, W., Schultz, R.: Multistage stochastic integer programs: An introduction. submitted to the present volume
29. Ruszczyński, A.: Decomposition methods in stochastic programming. Math. Programming **79** (1997) 333–353
30. S-PLUS User's Guide Version 4.5. Data Analysis Products Division, MathSoft, Seattle, 1998
31. Takriti, S., Krasenbrink, B., Wu, L.S.-Y.: Incorporating fuel constraints and electricity spot prices into the stochastic unit commitment problem. Oper. Res. **48** (2000) 268–280
32. Vitoriano, B., Cerisol, S., Ramos, A.: Generating scenario trees for hydro inflows. Proceedings of the 6th International Conference Probabilistic Methods Applied to Power Systems PMAPS 2000, Volume 2, INESC Porto, 2000
33. Wolfram Research: Mathematica Time Series Pack: Reference and User's Guide. Champaign, IL, 1995
34. Zenios, S.A., Shtilman, M.S.: Constructing optimal samples from a binomial lattice. J. Inform. Optim. Sci. **14** (1993) 125–147