Power Management under Uncertainty by Lagrangian Relaxation

N. Gröwe-Kuska[1],
W. Römisch, M.P. Nowak,
I. Wegner
Humboldt University Berlin,
Berlin, Germany.
[1]E-mail:nicole@mathematik.hu-berlin.de.

K.C. Kiwiel
Systems Research Institute,
Warsaw, Poland.
E-mail: kiwiel@ibspan.waw.pl.

Abstract: The weekly cost-optimal generation of electric power in a hydro-thermal generation system is modeled as a multistage mixed-integer stochastic program. The model incorporates uncertainties of electrical load forecasts, of inflows to pumped storage hydro plants and of fuel or electricity prices. For its solution a stochastic Lagrangian relaxation scheme is designed by assigning (stochastic) multipliers to all constraints coupling power units. Numerical results are presented for the generation system of a German utility under uncertain load. The stochastic load process is approximated by a finite number of realizations (scenarios) in scenario tree form.

Keywords: Stochastic integer programming, Lagrangian relaxation, unit commitment, bundle methods, scenario generation.

I. INTRODUCTION

In recent years there has been considerable interest in the application of mathematical modeling and optimization techniques for operating power systems and trading electricity. Much of the interest in stochastic power management models has been stimulated by the ongoing liberalization of electricity markets: electric utilities generate power in a competitive environment, generating and trading activities must be coordinated, electricity portfolios for spot and option markets become important, and the electrical load as well as electricity prices become increasingly unpredictable.

The present paper aims at optimizing generation and trading of an electric hydro-thermal based utility under data uncertainty. More specifically, we consider a power system comprising thermal units, pumped hydro storage plants and contracts for delivery and purchase. The relevant uncertain data comprise electric load, stream flows to hydro units, and fuel and electricity prices. We develop a dynamic stochastic programming model where the expected production costs are minimized subject to operational constraints. Since the model contains stochastic mixed-integer decisions and is large-scale, new questions are raised on designing solution algorithms and generating approximate scenario-based data processes. The goal of the paper is to inform the reader on how to incorporate fuel and electricity prices into the stochastic unit commitment problem, how to generate representative scenario trees for the uncertain data and how we succeded in solving the resulting large models.

The solution approach pursued in the present paper consists in a *stochastic version* of classical Lagrangian relaxation [1], which is very popular in power optimization [2], [3], [4], [5], [6], [7], [8]. Since the coupling constraints contain random variables, stochastic multipliers are needed for their dualization, and the dual problem is a nondifferentiable stochastic program. Consequently, this 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 heuristic. With a state-of-the-art bundle method for solving the dual, specialized subproblem solvers and Lagrangian heuristics, this *stochastic Lagrangian relaxation* algorithm becomes rather efficient. Our model and solution techniques are validated on the system of the German utility Vereinigte Energiewerke AG (VEAG). The VEAG generation system consists of 25 (coal-fired or gas-burning) thermal units and 7 pumped 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. Our numerical results indicate that the algorithm bears potential for solving complex real-life power scheduling models under uncertainty in reasonable time.

The stochastic power management model uses a set of scenarios to model data uncertainty. In our approach to load scenario tree generation, simulation scenarios are drawn from a SARIMA model for the load. Their empirical means and standard deviations enter a tree building scheme for the initial (binary) load scenario tree. In a final step the number of load scenarios is reduced by a scenario deletion procedure based on a suitable probability distance.

The paper is organized as follows. In §II we give a description of a our stochastic programming model. In §III we describe the stochastic Lagrangian relaxation approach together with its components and report on numerical results for the VEAG system with uncertain load. In §IV we present our procedure for generating scenario trees of the electrical load process and report on numerical tests.

II. POWER SYSTEM 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 weekly cost-optimal generation under uncertainty in electrical load, inflows in hydro units and prices for fuel or contracts.

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

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 commited capacity should exceed the load in every period by a certain amount. Other operating constraints which have to be incorporated are generating limits for thermal and hydro units. Each generating unit can only be operated within a feasible range defined by its minimum and maximum capacities. Water utilization for power generation in a pumped storage hydro unit is further limited by the storage volume in the upper and lower dam of the unit. For thermal units there are additional minimum up/down-time requirements: when a unit is switched on (off), it must remain on (off) for a certain number of time steps. Contracts for delivery and purchase have minimum up/downtime of one period.

To schedule the generation in a power system schedulers forecast the electric load for the specific time span. Since the electric load is mainly driven by meteorological parameters (temperature, cloud cover, etc.) the actual system load deviates from the prediction. Uncertainty on the generating system is not limited to that of electric load. Other sources of uncertainty are generator outages, inflows to 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. Let ξ_t collect the uncertain data in period t, $t = 1{:}T$, i.e., the load and the spinning reserve, water inflows, and coefficients of cost functions for thermal units and power contracts. Then we assume that $\{\xi_t\}_{t=1}^{T}$ is a discrete-time stochastic process. Of course, at the beginning of the scheduling horizon we only know the probability distribution of the data process $\{\xi_t\}_{t=1}^{T}$ and not its precise outcome. Nevertheless we are forced to take decisions. In practice the data forecast may be reliable until some period $t_1 \in \{1{:}T-1\}$, so that the data process $\{\xi_t\}_{t=1}^{t_1}$ is deterministic. In this context it makes sense to seek scheduling decisions which minimize the sum of the costs caused by the scheduling decisions for the time span $t = 1{:}t_1$ plus the expected generating costs for $t = t_1 + 1{:}T$ while meeting the operational constraints. The scheduling decisons of the interval $t = 1{:}t_1$ are the deterministic (first-stage) decisions, the remaining decision variables depend on the outcome of the stochastic data process. In this way, we end up with a multistage stochastic program. (For a detailed introduction to multistage stochastic programming we refer to [9].)

We now assume that we have a *discrete* distribution of the data process $\{\xi_t\}_{t=1}^{T}$. (This is the standard approach in multistage stochastic programming to avoid theoretical and numerical difficulties caused by multivariate continuous probability distributions.) Its support consists of *scenarios* (i.e., realizations or trajectories of $\{\xi_t\}_{t=1}^{T}$) that form a *scenario tree* based on a finite set of nodes $\mathcal{N}$ (cf. Fig. 1). Each node consists of a bundle of scenarios sharing a common history. Nodes of the scenario tree at which a bundle of scenarios branches into several disjoint bundles are called *branching points* and the intervals between them *stages*. The *root* node $n = 1$ stands for period $t = 1$. Every other node n has a unique *predecessor* node n_- and a *transition* probability $\tau_n > 0$, 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

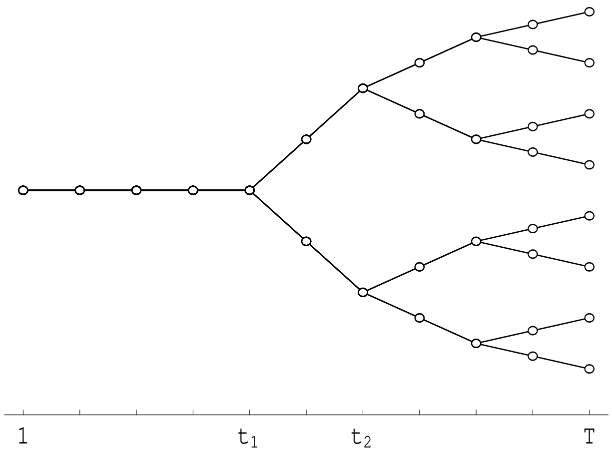

Fig. 1. Example for a scenario tree

$$\pi_1 = 1, \quad \pi_n = \tau_n \pi_{n_-} \quad \text{for } n \neq 1.$$

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

$$\pi_n = \sum_{n_+\in\mathcal{N}_+(n)} \pi_{n_+}, \quad \tau_{n_+} = \pi_{n_+}/\pi_n \quad \text{for } n_+ \in \mathcal{N}_+(n).$$

Each node n corresponds to a set of realizations of $\{\xi_t\}_{t=1}^{T}$ that coincide until the period $t(n)$, the time step associated with node n. To any node $n \in \mathcal{N}$ there is assigned a set of scheduling decisions. The sequence of scheduling decisions also form a discrete-time stochastic process. Of course, the decisions assigned to node n may depend only on the data observable till period $t(n)$, i.e. are nonanticipative. Consequently, if two scenarios are indistinguishable at some node n (i.e. they carry the same information throughout the period between the beginning of the horizon and up until $t(n)$) the corresponding decisions throughout the path from the root to node n must be the same.

A. NOTATIONS

Model Parameters	
T	number of time intervals
$\mathcal{T}$	$\{1{:}T\}$
I	number of thermal units (including contracts)
$\mathcal{I}$	$\{1{:}I\}$
J	number of pumped storage hydro units
$\mathcal{J}$	$\{1{:}J\}$
$\mathcal{N}$	node set of the scenario tree
N	$\lvert\mathcal{N}\rvert$, the total number of nodes of the scenario tree
π_n	probability of node n
$t(n)$	time step corresponding to node n
$\text{path}(n)$	path from the root to node n

d_t^n	load during period t at node n
r_t^n	spinning reserve during period t at node n
$p_{it(n)}^{\min}$	minimal output of thermal unit i during period $t(n)$
$p_{it(n)}^{\max}$	maximal output of thermal unit i during period $t(n)$
$\bar{\tau}_i$	minimum uptime of thermal unit i
$\underline{\tau}_i$	minimum downtime of thermal unit i
C_i^n	cost function at node n for operating thermal unit i
S_i^n	cost function for starting-up thermal unit i during period $t(n)$
η_j	pumping efficiency of pumped storage hydro unit j
$v_{jt(n)}^{\max}$	maximal generation level of pumped storage hydro unit j during period $t(n)$
$w_{jt(n)}^{\max}$	maximal pumping level of pumped storage hydro unit j during period $t(n)$
$l_{jt(n)}^{\max}$	maximal fill of the upper dam of pumped storage hydro unit j during period $t(n)$
γ_j^n	water inflow in pumped storage hydro unit j at node n
l_j^{in}	initial fill (in energy) of the upper dam at period 0
l_j^{end}	final fill (in energy) of the upper dam at the end of period T

Decision Variables associated with node $n \in \mathcal{N}$	
u_i^n	commitment state of thermal unit i $u_i^n \in \{0,1\}$ (1 if on, 0 if off)
$u_i^{\text{path}(n)}$	$(u_i^n)_{n \in \text{path}(n)}$
p_i^n	production level for unit i
v_j^n	generation level of pumped storage hydro unit j
w_j^n	pumping level of pumped storage hydro unit j
l_j^n	water storage in the upper reservoir of plant j at the end of time interval $t(n)$

In addition, 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 $\kappa = 1{:}t(n) - 1$.

B. MODEL

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 objective function to be minimized then is given by

$$\sum_{n \in \mathcal{N}} \pi_n \left\{ \sum_{i=1}^{I} C_i^n(p_i^n, u_i^n) + S_i^n\left(u_i^{\text{path}(n)}\right) \right\}. \qquad (1)$$

The fuel costs C_i^n associated with node n for operating the thermal unit i are piecewise linear convex, strictly monotonically increasing. The start-up costs of a thermal unit depend on the down-time of the unit. They may vary from a maximum cold-start value to a much smaller value when the thermal unit is still relatively close to its operation temperature. The down-time dependence of the start-up costs are expressed by a unit-dependent step function. Costs for the startup process of power contracts are negligible.

The minimization is subject to the following *operating constraints*:

TABLE I

SIZE OF THE SCENARIO-TREE MODEL (1)–(4) 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

- Operating ranges (1) and minimum up/down-time requirements (2a),(2b) of thermal units:

$$p_{it(n)}^{\min} u_i^n \le p_i^n \le p_{it(n)}^{\max} u_i^n, \quad u_i^n \in \{0,1\}, \quad n \in \mathcal{N},\ i \in I, \quad (2a)$$

$$u_i^{n_{-\kappa}} - u_i^{n_{-(\kappa+1)}} \le u_i^n, \quad \kappa = 1{:}\bar{\tau}_i - 1,\ n \in \mathcal{N},\ i \in I, \quad (2b)$$

$$u_i^{n_{-(\kappa+1)}} - u_i^{n_{-\kappa}} \le 1 - u_i^n, \quad \kappa = 1{:}\underline{\tau}_i - 1,\ n \in \mathcal{N},\ i \in I, \quad (2c)$$

- Operating ranges (3a) and dynamics of pumped storage hydro units (3b); water storages in the upper dam of a plant at the beginning and at the end of the scheduling period (3c).

$$\begin{aligned} 0 &\le v_j^n \le v_{jt(n)}^{\max}, \\ 0 &\le w_j^n \le w_{jt(n)}^{\max}, \quad n \in \mathcal{N},\ j \in \mathcal{J}\, j = 1{:}J, \\ 0 &\le l_j^n \le l_{jt(n)}^{\max}, \end{aligned} \qquad (3a)$$

$$l_j^n = l_j^{n_-} - v_j^n + \eta_j w_j^n + \gamma_j^n, \quad n \in \mathcal{N},\ j \in \mathcal{J}, \qquad (3b)$$

$$l_j^0 = l_j^{\text{in}}, \quad l_j^n = l_j^{\text{end}}, \quad n \in \mathcal{N}_T,\ j \in \mathcal{J}, \qquad (3c)$$

- Constraints that couple different units: balance between electric load and supply (4a); spinning reserve requirement (4b).

$$\sum_{i=1}^{I} p_i^n + \sum_{j=1}^{J} (v_j^n - w_j^n) \ge d^n, \quad n \in \mathcal{N}, \qquad (4a)$$

$$\sum_{i=1}^{I} (u_i^n p_{it(n)}^{\max} - p_i^n) \ge r^n, \quad n \in \mathcal{N}, \qquad (4b)$$

A few comments on the stochastic programming model (1)–(4) are in order. First, the modeling of the objective and the constraint set by mixed-integer terms is motivated by the far more powerful algorithmic tools that are available for mixed-integer programs. Second, for

$$\tau_{\text{ini}} := 1 - \max_{i=1:I} \{ \tau_i^c, \bar{\tau}_i - 1, \underline{\tau}_i - 1 \} \qquad (5)$$

and $\tau = \tau_{\text{ini}}{:}0$, $u_{i\tau}$ in (1) and (2b)–(2c) are replaced by fixed initial values $u_{i\tau} \in \{0,1\}$, $i = 1{:}I$. Third, the nonanticipativity of the process of scheduling decisions is handled implicitly (i.e., it is ensured automatically) by the tree-based model (1)–(4). For $N := |\mathcal{N}|$ nodes the model involves IN binary and $(I+2J)N$ continuous decision variables. Table I shows how the size of a mixed-integer LP formulation of the scenario-tree model (1)–(4) increases with the number of nodes (without taking into account the constraints of type (2b)–(2c) and the objective function). In contrast, an equivalent formulation of the stochastic program involving $S := |\mathcal{N}_T|$ scenarios (cf. §2.1 in [10]) has ITS binary and $(I+2J)TS$ continuous decision variables; note that typically $N \ll TS$.

III. Lagrangian Relaxation

The stochastic programming model (1)–(4) represents a large-scale linear mixed-integer optimization problem coupled both in time and with respect to different generation units. The model is very demanding from the computational point of view. Even latest mixed-integer programming methodology and software like CPLEX fail to solve the full problem. Therefore, algorithmic approaches to stochastic power management problems utilize decomposition techniques. In classical unit commitment, Lagrangian relaxation is very popular and has a long history. At present, suggested algorithms for solving stochastic power mamagement problems are based on one of the following Lagrangian relaxation schemes: (a) scenario decomposition [11], [12], [13], [14], [15], (b) stochastic (augmented) Lagrangian relaxation of coupling constraints [16], [17], [18], [19], [20], [21]. The approaches in (a) successively decompose the stochastic program into finitely many deterministic (or scenario) programs that may be solved by available conventional techniques. The approach of (b) hinges on a successive decomposition into finitely many smaller stochastic subproblems for which (efficient) solution techniques must be developed eventually. Due to the nonconvexity of the underlying stochastic program, the successive decompositions in (a)–(b) have to be combined with certain global optimization techniques (branch-and-bound, heuristics, etc.).

Let us now briefly describe the stochastic Lagrangian relaxation approach followed in [20], [22], [10] together with its component.

Problem (1)–(4) is almost separable with respect to units, since only constraints (4) couple different units. This structure allows us to apply a stochastic version of Lagrangian relaxation by associating a stochastic Lagrange multiplier λ with the coupling constraints (4). For convex multistage stochastic programs, this approach is justified by the general duality theory of [23]. Hence suppose momentarily the constraint $u_i^n \in \{0,1\}$ of (2a) is relaxed to $u_i^n \in [0,1]$, so that problem (1)–(4) becomes convex. Then (cf. [20, §4]) with $z := (u,p,v,w)$ and multipliers $\lambda := (\lambda^n)_{n\in\mathcal{N}} =: (\lambda_1,\lambda_2) \in \mathbb{R}_+^N \times \mathbb{R}_+^N$, where $N := |\mathcal{N}|$, the *Lagrangian*

$$
\begin{aligned}
L(z\,;\lambda) \quad := \quad & \sum_{n\in\mathcal{N}} \pi_n \Bigg\{ \sum_{i=1}^{I} \Big[C_i^n(p_i^n,u_i^n) + S_i^n\left(u_i^{\mathrm{path}(n)}\right)\Big] \qquad (6)\\
& + \lambda_1^n \Big[d^n - \sum_{i=1}^{I} p_i^n - \sum_{j=1}^{J} (v_j^n - w_j^n) \Big] \\
& + \lambda_2^n \Big[r^n - \sum_{i=1}^{I} (u_i^n p_{it(n)}^{\max} - p_i^n) \Big] \Bigg\},
\end{aligned}
$$

and the *dual function*

$$
D(\lambda) := \min_x \{ L(z\,;\lambda) \quad \text{s.t. constraints (2)–(3)} \}, \qquad (7)
$$

the *dual problem* reads

$$
\max \left\{ D(\lambda) : \lambda \in \mathbb{R}_+^{2N} \right\}. \qquad (8)
$$

The dual function D is concave and polyhedral, since the fuel costs C_i^n are polyhedral in p_i^n.

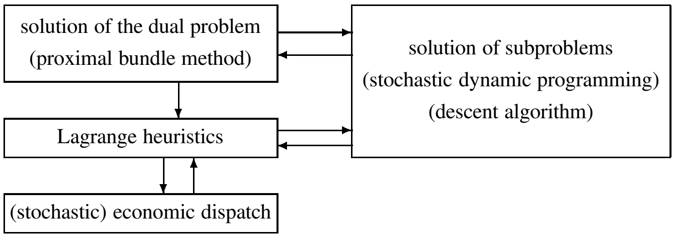

Fig. 2. Structure of the stochastic Lagrangian relaxation method

The minimization in (7) decomposes into stochastic single unit subproblems. Specifically, the dual function

$$
D(\lambda) = \sum_{i=1}^{I} D_i(\lambda) + \sum_{j=1}^{J} \hat{D}_j(\lambda_1) + \sum_{n\in\mathcal{N}} \pi_n \left(\lambda_1^n d^n + \lambda_2^n r^n\right), \qquad (9)
$$

may be evaluated by solving the *thermal subproblems*

$$
\begin{aligned}
D_i(\lambda) = \min_{u_i} \quad & \Bigg\{ \sum_{n\in\mathcal{N}} \pi_n \Big[\min_{p_i^n} \{ C_i^n(p_i^n,u_i^n) - (\lambda_1^n - \lambda_2^n) p_i^n \} \qquad (10)\\
& - \lambda_2^n u_i^n p_{it(n)}^{\max} + S_i^n\left(u_i^{\mathrm{path}(n)}\right) \Big] \text{ s.t. (2)} \Bigg\},
\end{aligned}
$$

(where we used separability and exchanged expectation with minimization over p_i) and the *hydro subproblems*

$$
\hat{D}_j(\lambda_1) = \min_{(v_j,w_j)} \left\{ \sum_{n\in\mathcal{N}} \pi_n \lambda_1^n (w_j^n - v_j^n) \text{ s.t. (3)} \right\}. \qquad (11)
$$

Both subproblems represent multistage stochastic programming models for the operation of a single unit. While the thermal subproblem (10) is a combinatorial multistage program involving stochastic costs, the hydro subproblem (11) is a linear multistage model with stochastic costs and stochastic right-hand sides.

Extending Lagrangian relaxation approaches for deterministic power management models, our method for solving the tree-based model (1)–(4) consists of the following ingredients:
(a) Solving the dual problem (8) by a proximal bundle method using function and subgradient information;
(b) Efficient solvers for the single unit subproblems: dynamic programming for (10) and a special descent algorithm for (11);
(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 Lagrange heuristics. The interaction of these components is illustrated in Figure 2. They are now briefly discussed; the interested reader is refered to [10] for a more detailed account. For a single unit, the hydro subproblem (11) is solved by a specialized descent method that generates a finite sequence of feasible hydro decisions with decreasing objective value and terminates with an optimal solution. The outer minimimum of the thermal subproblem (10) with respect to the commitment state u_i is solved by dynamic programming. Minimization with respect to p_i is done by a revised economic dispatch algorithm. Values for Lagrangian multipliers used for defining the thermal and hydro subproblems are obtained by maximizing the dual function D (cf. (9)). Since there exist subgradients od D (D is concave) the dual problem

(9) may be solved by the modern proximal bundle method [24] for concave nondifferentable maximization. The proximal bundle method has very strong convergence properties. Starting values for the Lagrangian multiplier λ we determine as follows. The initial values for the components of the multiplier λ_2 are zero. A priority list scheme of thermal units provides the initial values for the multiplier λ_1. When the bundle method delivers an optimal multiplier λ^*, the optimal value $D(\lambda^*)$ provides a lower bound for the optimal cost of the model (1)–(4). In general, however, the "dual optimal" scheduling decisions $z(\lambda^*) = (u(\lambda^*), p(\lambda^*), v(\lambda^*), w(\lambda^*))$ violate the load and reserve constraints (4) such that a low-cost primal feasible solution has to be determined by a *Lagrangian heuristics*. Two Lagrangian heuristics 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 $\mathcal{N}$.

Our first heuristic LH1 starts by computing mean values of the scenario-based stochastic processes ξ, λ^* and $l_j = l_j(\lambda^*)$, $j = 1{:}J$, i.e., we determine $\bar{\xi} = \mathbb{E}[\xi]$, $\bar{\lambda}^* = \mathbb{E}[\lambda^*]$ and $\bar{l}_j = \mathbb{E}[l_j]$. For instance, we have

$$\begin{aligned}(\bar{d}_t, \bar{r}_t, \bar{\gamma}_t, \bar{a}_t, \bar{b}_t, \bar{c}_t) &= \bar{\xi}_t = \sum_{n\in\mathcal{N}_t} \pi_n \xi^n \\ &= \sum_{n\in\mathcal{N}_t} \pi_n (d^n, r^n, \gamma^n, a^n, b^n, c^n).\end{aligned}$$

Next, replacing $\mathcal{N}$ by $\{1{:}T\}$ and ξ by $\bar{\xi}$, we consider deterministic single-scenario versions of the model (1)–(4) and the thermal subproblems (10). Then we find deterministic generation and pumping decisions v_j and w_j that satisfy the constraints (3) with l_j and γ_j replaced by $\bar{l}_j$ and $\bar{\gamma}_j$, respectively. Furthermore, deterministic on/off decisions u_i are computed by dynamic programming as solutions of the thermal subproblems (10) with the multiplier λ and the cost coefficients a, b and c replaced by $\bar{\lambda}^*$, $\bar{a}$, $\bar{b}$ and $\bar{c}$. In the next step, the hydro decisions v_j and w_j are rescheduled in order to meet, as much as possible, the modified reserve constraint

$$\sum_{i=1}^{I} u_{it} p_{it}^{\max} \geq \bar{d}_t + \bar{r}_t + \sum_{j=1}^{J} (w_{jt} - v_{jt}) \quad t = 1{:}T, \tag{12}$$

i.e., the sum of the load and reserve constraints (4a) and (4b) with d and r replaced by $\bar{d}$ and $\bar{r}$. To this end our procedure reduces the right-hand side of (12) by modifying the hydro schedules at those t where the constraint is violated and its right-hand side is largest in a certain set of neighboring time periods. This procedure is repeated several times (see also [4]). In the next step the hydro variables are fixed, and following [8] we search for binary variables u_i that satisfy the constraint (12). The main idea is to select the period t where (12) is most violated and to increase $\bar{\lambda}_t^*$ as much as necessary to switch on in the thermal subproblems just as many units as needed to satisfy (12) at t. This is repeated until the constraint (12) is satisfied in all periods. Since this technique does not distinguish between identical units that appear quite often in practice, the startup costs of such units are slightly modified. Once the binary decisions u_i are fixed, the economic dispatch algorithm (see [22] and [10]) completes LH1 by providing (deterministic) scheduling decisions $\{p_t, v_t, w_t\}$ for the whole planning horizon $t = 1{:}T$.

TABLE II

COMPUTING TIMES AND GAPS WITH LH1 (NOA 3.0: opt_tol $= 10^{-3}$, NGRAD $= 50$)

S	N	time[s]	gap[%]	N	time[s]	gap[%]
20	1982	89	0.15	1627	94	0.10
20	1651	68	0.37	1805	85	0.07
50	4530	475	0.18	4060	274	0.10
50	4041	313	0.10	4457	288	0.43
100	9230	1183	0.11	9224	1072	0.13
100	7727	930	0.09	8867	1234	0.30

The second Lagrangian heuristic LH2 is based on the observation that usually the binary decisions in $u(\lambda^* + \varepsilon\underline{1})$ change significantly relative to $u(\lambda^*)$ even for small $\varepsilon > 0$, and ensure feasibility for ε large enough. (Here $\underline{1}$ denotes the L-vector with unit components.) Hence, LH2 starts by finding some $\varepsilon > 0$ such that $z(\lambda^* + \varepsilon\underline{1})$ satisfies all constraints (2)–(4). Then taking $u(\lambda^* + \varepsilon\underline{1})$ as a starting point, a finite sequence of binary decisions is constructed such that their components are decreasing. This is done by selecting a node $n \in \mathcal{N}$ where the available reserve capacity $\sum_{i=1}^{I} (u_i^n p_{it(n)}^{\max} - p_i^n) - r^n$ is maximal, and switching some unit i off at n and some predecessor and successor nodes. This unit i and the neighboring nodes of n are detected by stochastic dynamic programming. Next, a stochastic economic dispatch problem is solved by the descent method described in [22] and [10]. This procedure, which generates a sequence of scheduling decisions at all nodes, is continued until infeasibility is detected during economic dispatch. The heuristic terminates with the scheduling decision having minimal cost (1).

A. Numerical results

The stochastic Lagrangian relaxation algorithm was implemented in C++ except for the proximal bundle method, for which the Fortran package NOA 3.0 [25] was used as a callable library. 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). A bunch of load scenario trees was constructed as follows. Starting with a reference load scenario obtained from real-life data, $S - 1$ random branching points were selected successively to produce a scenario tree with S identical scenarios. Then a (discretized) Brownian motion was added to each node of the scenario tree. The test runs were performed on an HP 9000 (780/J280) computer with 180 MHz frequency and 768 MByte main memory under HP-UX 10.20.

First we consider the Lagrangian relaxation algorithm based on LH1. Table II shows computing times and gaps for different numbers of scenarios (S) and four randomly generated scenario trees, each having a different number of nodes (N). The gap refers to the relative difference

$$\frac{1}{D_*}\left(\sum_{t=1}^{T}\sum_{i=1}^{I} [C_{it}(p_{it}, u_{it}) + S_{it}(u_i)] - D_*\right)$$

of the cost of the scheduling decision (u, p, v, w) and the optimal value D_* of the dual problem. We note that, in general, this

TABLE III

COMPUTING TIMES AND GAPS WITH LH2 (NOA 3.0: opt_tol $= 10^{-5}$, NGRAD = 200)

S	N	NOA time[s]	total time[s]	gap[%]
1	168	10	16	0.20
5	542	65	101	0.19
10	983	128	230	0.71
21	2098	351	531	0.39
24	2175	374	695	0.83
27	2208	380	8349	0.73
32	2173	359	3337	0.66
34	3043	497	1499	0.95
39	3848	874	4092	0.82

gap does not provide a quality measure for the approximate first stage solution (it may even become nonpositive). When reading the computing times in Table II, it is worth recalling that $N = 4000$ and $N = 8000$ correspond to $100,000$ and $200,000$ binary variables in the model (1)–(4), respectively.

Table III reports computing times and gaps for the Lagrangian relaxation algorithm based on LH2 applied to test problems with different numbers S and N of scenarios and nodes of randomly generated load scenario trees. Here the gap refers to the following bound of the relative duality gap

$$\frac{1}{D_*}\left(\sum_{n\in\mathcal{N}}\pi_n\sum_{i=1}^{I}\left[C_i^n(p_i^n,u_i^n)+S_i^n\left(u_i^{\mathrm{path}(n)}\right)\right]-D_*\right).$$

Clearly, this bound provides an accuracy certificate for the approximate primal-feasible solution $\{(u^n,p^n,v^n,w^n))\}_{n\in\mathcal{N}}$.

While the "deterministic" Lagrangian heuristics LH1 requires only short computing times, this becomes quite different for the "stochastic" heuristics LH2. Table III gives more insight into the (total) computing times of different test runs. Higher computing times are always due to very many economic dispatches required by LH2. It is worth mentioning here that LH2 is quite sensitive to the accuracy of the dual solution, i.e., to the optimality tolerance of the proximal bundle method. The advantage of using LH1 consists in low running times even for mid-size scenario trees, while its drawbacks are that only first-stage solutions are provided with no accuracy bounds. The advantage of LH2 is that it produces a "stochastic" solution together with a guaranteed accuracy bound, but at the expense of higher computing times even for scenario trees of smaller size. For further information the interested reader is referred to [26].

Another test employed a load scenario tree with sixteen scenarios and 912 nodes that was generated from real-life VEAG data by the technique described in §IV. As before, we had $T = 168$, $I = 25$, $J = 7$. In effect, the scenario tree formulation of our optimization model had 22,800 binary and 41,952 continuous variables, 92,224 constraints and 242,704 nonzeros. Figure 3 provides the final output of the Lagrangian relaxation algorithm using LH2. It presents 16 realizations of load and generation levels.

IV. GENERATION OF LOAD SCENARIO TREES

To build representative scenario trees is presently an active field of research; see the survey [27]. We approximate the stochastic load process by a scenario tree within three steps:

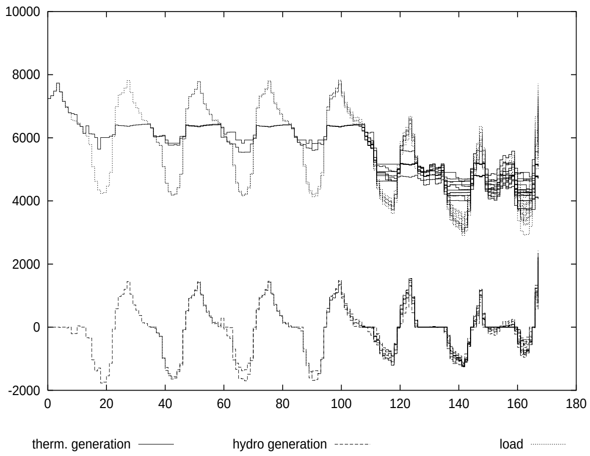

Fig. 3. Optimal stochastic solution for one week

1. Identify a statistical (time series or regression) model of the load, and use it for generating a large number of simulation scenarios.
2. Determine an initial structure of the load tree. Compute scenario values, using the sample means and standard deviations of the simulated scenarios.
3. Reduce the number of scenarios in the tree optimally.

In STEP 1 the probability distribution of the random load is modeled. For load profiles there exist advanced discrete time stochastic models (regression or time series models). They have to be calibrated from historical load profiles.

For the identification of a statistical model we were given an hourly load profile of one year. Because of missing meteorological parameters we could not fit regression models (cf. [15]). Alternatively, the seasonal components and the correlation structure of the load process can be described by *seasonal autoregressive integrated moving average* (SARIMA) processes. Estimation and test procedures from the *Mathematica Time Series Pack* [28] were used to identified a SARIMA$(7,0,9)\times(0,1,0)_{168}$ model for the load d_t in period t. Introducing $y_t := d_t - d_{t-168}$ it reads

$$y_t - \hat{\phi}_1 y_{t-1} - \ldots - \hat{\phi}_7 y_{t-7} = Z_t + \hat{\theta}_1 Z_{t-1} + \ldots + \hat{\theta}_9 Z_{t-9}. \quad (13)$$

The estimated model coefficients are

$$(\hat{\phi}_1,\ldots,\hat{\phi}_7) = (2.79,-4.35,5.16,-4.88,3.67,-1.92,0.50),$$

$$(\hat{\theta}_1,\ldots,\hat{\theta}_9) = (-1.27,1.53,-1.35,0.88,-0.31,-0.06,0.18,0.11,0.07).$$

Z_t, $t \in \mathbb{Z}$, are independent, normally distributed random variables with mean 0 and standard deviation 108.3.

According to the SARIMA equation (13) a large number (M) of *simulated load scenarios* (sample paths) $\tilde{d}^\ell = (\tilde{d}_t^\ell)_{t=t_1+1}^T$, $\ell = 1{:}M$, are generated using M i.i.d. realizations of Z_t, $t = t_1 - 8{:}T$, and starting values d_t, $t = t_1 - 174{:}t_1$. The *empirical means* $\bar{d}_t$ and *standard deviations* $\bar{\sigma}_t$ of the simulated load scenarios are defined by

$$\bar{d}_t = \frac{1}{M}\sum_{\ell=1}^{M}\tilde{d}_t^\ell, \quad \bar{\sigma}_t^2 = \frac{1}{M-1}\sum_{\ell=1}^{M}(\tilde{d}_t^\ell - \bar{d}_t)^2, \quad t = t_1+1{:}T.$$

In STEP 2 the branching scheme of the of the initial load scenario tree is selected, i.e., the number and position of the branching points and the branching degree in every node. The following initial structure of the load scenario tree was used for a planning horizon of one week:

- A balanced tree with 12 branching points $t_k = 12 + 12k$, $k = 1{:}11$.
- All branching points have branching degree 2; i.e., at any branching point a bundle of scenarios branches into two disjoint bundles.

Thus, the tree consists of $S := 2^{12}$ scenarios $d^s = (d_t^s)_{t=1}^T$, $s = 1{:}S$. The branching points t_k, $k = 2{:}12$, are chosen at the (normally fixed) times when already observable meteorological and load data provide the opportunity to re-adjust the unit commitment. For longer planning periods a non-equidistant position of the branching points is preferable in order to restrict the number of scenarios. By assigning two successors to any node n in $\mathcal{N}_{t_k}$, $k = 1{:}K$, it is possible to distinguish the events with the verbal description "low load" and "high load" in the time period $t = t_k + 1{:}t_{k+1}$. (For convenience of notation set $t_{K+1} := T$.) An additional event like "medium" load can easily be included, but increases the scenario number to $S = 3^K$.

It remains to specify the scenario values and their probabilities. First compute the empirical means $\bar{d}_t$, $t = t_1 + 1{:}T$, and the standard deviations $\bar{\sigma}_t$ for $t = t_k$, $k = 2{:}K+1$. The predicted load for the planning period $t = 1{:}t_1$ yields the first t_1 components for all scenarios. (If no load prediction is available one can use the empirical means for $t = 1{:}t_1$). To any scenario s, $s = 1{:}S$ there is assigned a vector $\omega^s = (\omega_k^s)_{k=2}^{K+1}$ with $\omega_k^s \in \{-1, 1\}$ for $k = 2{:}K+1$. It provides a unique description of the path in the binary tree that corresponds to scenario s. In particular, set $\omega_k^s := -1$ ($\omega_k^s := 1$) if the values of scenario s for $t = t_k + 1{:}t_{k+1}$ are realizations of the event with the verbal description "low load" ("high load") for this time span. The value of scenario s for $t = t_1{:}T$ is defined as

$$d_t^s := \bar{d}_t + \sum_{i=2}^{k-1} \omega_i^s \frac{\bar{\sigma}_{t_i}}{2^{(K+2-i)/2}} + \omega_k^s \frac{\bar{\sigma}_{t_k}}{2^{(K+2-k)/2}} \frac{t - t_{k-1}}{t_k - t_{k-1}} \qquad (14)$$

for $t = t_{k-1} + 1{:}t_k$, $k = 2{:}K+1$.

We let all scenarios have equal probabilities $S^{-1} = 2^{-K}$. (Alternative scenario probabilities might be computed from histograms of the simulated scenarios.)

A few comments on the tree construction formula (14) are in order. First, for $t = t_1 + 1{:}T$, the mean scenario value $\frac{1}{S}\sum_{s=1}^S d_t^s$ coincides with the empirical mean $\bar{d}_t$. Second, the symmetry of the load tree is consistent with the normality assumptions imposed on the time series model for the load process. Third, for $k = 1{:}K$, the events "low load" ("high load") for $t = t_k + 1{:}t_{k+1}$ are expressed in terms of scaled empirical standard deviations $\bar{\sigma}_{t_{k+1}}$. To model increasing load uncertainty, the variances $\mathrm{var}(d_t)$ of scenario values are strictly increasing with t. The extremal scenario s with $\omega_k^s = 1$ for all k has in the final period T the value

$$d_T^s = \bar{d}_T + 2^{-K/2}\bar{\sigma}_{t_2} + \cdots + 2^{-1/2}\bar{\sigma}_T .$$

Thus unrealistic ("too large") load values are avoided. Further, for $\bar{\sigma}_{t_{k+1}} \approx \bar{\sigma}$, $k = 1{:}K$, we have $\mathrm{var}(d_T) \approx \bar{\sigma}^2(\frac{1}{2^K} + \cdots + \frac{1}{2}) \approx \bar{\sigma}^2$. Finally, we add that the scenario values between the t_k's are linearly interpolated so as to save work required for computing $\bar{\sigma}_t^2$ for all $t = t_1 + 1{:}T$.

Figure 4 shows ten scenarios (including the extremal paths corresponding to "low load" and "high load" for the time span $t = t_1 + 1 : T$) of a load scenario tree generated via the scheme (14) with $2^{12} = 4096$ scenarios for a planning horizon of one week with an hourly discretization and branching points $t_k = 12 + 12k$, $k = 1{:}12$.

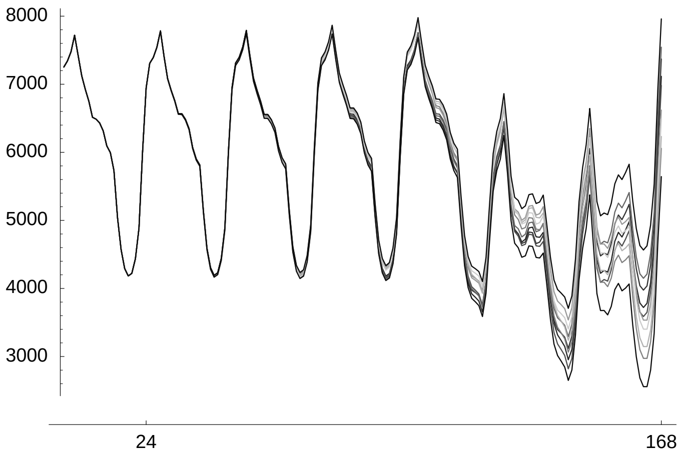

Fig. 4. Ten selected scenarios of a load scenario tree for one week

Since the mixed-integer model (1)–(4) is large even for relatively few nodes, the number of scenarios of the initial scenario tree has to be reduced in STEP 3. Our reduction argument is based on certain probability metrics that measure the distance between the initial discrete approximation of the distribution underlying the load and the reduced one.

The scenario reduction procedure works as follows:

1. Initialization: Set $S' := S$.

Compute the Euclidian distances c between all scenarios in the initial scenario tree:

$$c(d^{s_1}, d^{s_2}) := \sqrt{\sum_{t=t_1}^{T} \left(d_t^{s_1} - d_t^{s_2}\right)^2}, \quad s_1, s_2 \in \{1 : S'\} \qquad (15)$$

2. Select scenario $s^* \in \{1, \ldots, S'\}$, such that

$$\pi_{s^*} \min_{s \neq s^*} c(d^s, d^{s^*}) = \min_{m=1,\ldots,S'} \pi_m \min_{s \neq m} c(d^s, d^m). \qquad (16)$$

and delete scenario d^{s^*} from the tree.

3. Update the probabilities of the scenarios in the reduced tree: Set $S' := S' - 1$ and

$$\pi_s := \begin{cases} \pi_{\bar{s}} + \pi_{s^*}, & \text{if } s = \bar{s} \text{ for some } \bar{s} \in \mathrm{Arg}\min_{s \neq s^*} c(d^s, d^{s^*}) \\ \pi_s, & \text{if } s \neq \bar{s} \end{cases} \qquad (17)$$

4. Stopping criterion:

If ($S' > N$) then goto 2., else STOP.

A. Example of load scenario generation

To test our approach, we generated a load scenario tree via the scheme (14) for an hourly discretized time horizon of one week ($T = 168$) with branching points $t_k = 12 + 12k$, $k = 1{:}12$ (cf. Fig. 4). The initial number of scenarios $S = 4096$ was reduced to 16 by applying the scenario reduction rule.

Figure 5 shows the position of the shifted supports ($d_t^s -$

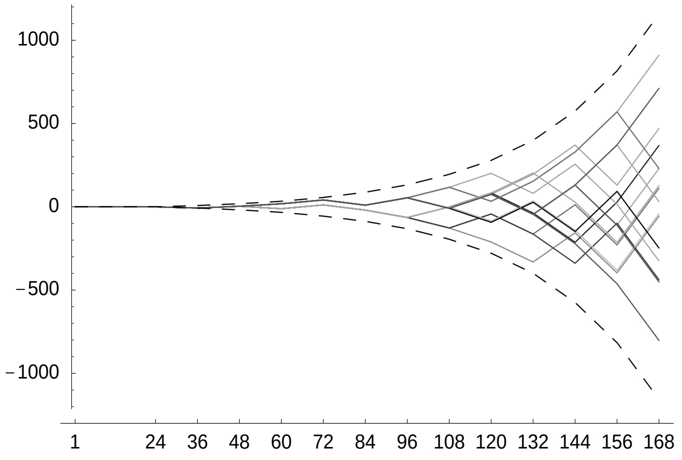

Fig. 5. Shifted supports of the reduced scenario tree

$\bar{d}_t)_{t=1}^{168}$, $s = 1{:}16$, of the reduced scenario tree within the extremal paths of the initial scenario tree indicated by dashed lines, with grey levels proportional to scenario probabilities. The probabilities β_s, $s = 1{:}16$, assigned to scenarios in the reduced tree vary between 0.04 and 0.11.

Acknowledgment

The authors are grateful to Darinka Dentcheva (Lehigh University; formerly Humboldt-University Berlin), Svetlozar T. Rachev (University of California at Santa Barbara and University Karlsruhe) and Rüdiger Schultz (Gerhard-Mercator University Duisburg) for useful discussions on the material of this paper, and to G. Schwarzbach, J. Thomas and J. Krause (VEAG Vereinigte Energiewerke AG, Berlin) for providing us with the data used in our test runs.

References

[1] C. Lemaréchal, "Lagrangian decomposition and nonsmooth optimization: Bundle algorithm, prox iteration, augmented Lagrangian," in *Nonsmooth Optimization, Methods and Applications*, F. Giannessi, Ed., pp. 201–216. Gordon and Breach, Philadelphia, 1992.

[2] D. Bertsekas, G. S. Lauer, N. R. Sandell, and T. A. Posberg, "Optimal short-term scheduling of large-scale power systems," *IEEE Trans. Automat. Control*, vol. AC-28, pp. 1–11, 1983.

[3] S. Feltenmark and K. C. Kiwiel, "Dual applications of proximal bundle methods, including Lagrangian relaxation of nonconvex problems," *SIAM J. Optim.*, vol. ?, 1999, To appear.

[4] R. Gollmer, A. Möller, W. Römisch, R. Schultz, G. Schwarzbach, and J. Thomas, "Optimale Blockauswahl bei der Kraftwerkseinsatzplanung der VEAG," in *Optimierung in der Energieversorgung II*, VDI-Berichte 1352, pp. 71–85. VDI, Düsseldorf, 1997.

[5] C. Lemaréchal and A. Renaud, "Dual-equivalent convex and nonconvex problems," Research Report, INRIA, Rocquencourt, 1996.

[6] G. B. Sheblé and G. N. Fahd, "Unit commitment literature synopsis," *IEEE Trans. Power Systems*, vol. 9, pp. 128–135, 1994.

[7] A. J. Wood and B. F. Wollenberg, *Power Generation, Operation, and Control*, Wiley, New York, second edition, 1996.

[8] G. Zhuang and F. D. Galiana, "Towards a more rigorous and practical unit commitment by Lagrangian relaxation," *IEEE Trans. Power Systems*, vol. 3, pp. 763–773, 1988.

[9] J. R. Birge and F. Louveaux, *Introduction to Stochastic Programming*, Springer, New York, 1997.

[10] N. Gröwe-Kuska, K. C. Kiwiel, M. P. Nowak, W. Römisch, and I. Wegner, "Power management in a hydro-thermal system under uncertainty by lagrangian relaxation," Preprint 19-99, Institut für Mathematik, Humboldt-Univ. Berlin, Berlin, Germany, 1999, submitted to the IMA Volumes in Mathematics and its Applications, Springer Verlag.

[11] C. C. Carøe and R. Schultz, "Dual decomposition in stochastic integer programming," *Oper. Res. Lett.*, vol. 24, pp. 37–45, 1999.

[12] C. C. Carøe and R. Schultz, "A two-stage stochastic program for unit commitment under uncertainty in a hydro-thermal power system," Preprint 98-13, Konrad-Zuse-Zentrum für Informationstechnik, Berlin, Germany, 1998.

[13] A. Løkketangen and D. L. Woodruff, "Progressive hedging and tabu search applied to mixed integer $(0,1)$ multi-stage stochastic programming," *J. Heuristics*, vol. 2, pp. 111–128, 1996.

[14] S. Takriti, J. R. Birge, and E. Long, "A stochastic model for the unit commitment problem," *IEEE Trans. Power Systems*, vol. 11, pp. 1497–1508, 1996.

[15] S. Takriti, B. Krasenbrink, and L. S.-Y. Wu, "Incorporating fuel constraints and electricity spot prices into the stochastic unit commitment problem," Research Report RC 21066, Mathematical Sciences Dept., IBM Research Division, T.J. Watson Research Center, Yorktown Heights, New York, 1997, To appear in Ann. Oper. Res.

[16] L. Bacaud, C. Lemaréchal, A. Renaud, and C. Sagastizábal, "Disaggregated bundle methods in stochastic optimal power management," Research Report ?, INRIA, Rocquencourt, 1999.

[17] P. Carpentier, G. Cohen, and J.-C. Culioli, "Stochastic optimal control and decomposition-coordination methods, part II: Application," in *Recent Developments in Optimization*, R. Durier and C. Michelot, Eds., Lecture Notes in Economics and Mathematical Systems 429, pp. 88–103. Springer-Verlag, Berlin, 1995.

[18] P. Carpentier, G. Cohen, J.-C. Culioli, and A. Renaud, "Stochastic optimization of unit commitment: A new decomposition framework," *IEEE Trans. Power Systems*, vol. 11, pp. 1067–1073, 1996.

[19] A. Renaud, "Optimizing short-term operation of power generation: A new stochastic model," Lecture presented at the VIII International Conference on Stochastic Programming, Vancouver (Canada), Aug. 1998.

[20] D. Dentcheva and W. Römisch, "Optimal power generation under uncertainty via stochastic programming," in *Stochastic Programming Methods and Technical Applications*, K. Marti and P. Kall, Eds., Lecture Notes in Economics and Mathematical Systems 458, pp. 22–56. Springer-Verlag, Berlin, 1998.

[21] W. Römisch and R. Schultz, "Decomposition of a multi-stage stochastic program for power dispatch," *Z. Angew. Math. Mech.*, vol. 76, pp. 29–32, 1996, Suppl. 3.

[22] M. P. Nowak and W. Römisch, "Stochastic Lagrangian relaxation applied to power scheduling in a hydro-thermal system under uncertainty," Preprint 98-36, Institut für Mathematik, Humboldt-Univ. Berlin, Berlin, Germany, 1998, Submitted to Ann. Oper. Res.

[23] R. T. Rockafellar and R. J.-B. Wets, "The optimal recourse problem in discrete time: L^1-multipliers for inequality constraints," *SIAM J. Control Optim.*, vol. 16, pp. 16–36, 1978.

[24] K. C. Kiwiel, "Proximity control in bundle methods for convex nondifferentiable minimization," *Math. Programming*, vol. 46, pp. 105–122, 1990.

[25] K. C. Kiwiel, "User's guide for NOA 3.0: A Fortran package for convex nondifferentiable optimization," Tech. Rep., Systems Research Institute, Warsaw, 1994.

[26] M. P. Nowak, *Stochastic Lagrangian Relaxation in Power Scheduling of a Hydro-Thermal System Under Uncertainty*, Ph.D. thesis, Institut für Mathematik, Humboldt-Univ. Berlin, Berlin, Germany, 1999, submitted.

[27] J. Dupačová, G. Consigli, and S. W. Wallace, "Scenarios for multistage stochastic programs," *Ann. Oper. Res.*, vol. ?, 1999, To appear.

[28] Wolfram Research, Champaign, IL, *Mathematica Time Series Pack: Reference and User's Guide*, 1995.

Nicole Gröwe-Kuska graduated from Charles University, Prague, in 1987. She recieved the Ph.D. degree in Applied Mathematics from the Humboldt University, Berlin, in 1995. Since 1987 she was with Humboldt University, Berlin, Delft University of Technology and the Konrad-Zuse Zentrum für Informationstechnik, Berlin. In 1997 she joined the Institute of Mathematics at Humboldt University Berlin. Her research activities are in the field of stochastic optimization, with a special emphasis on real life applications for power systems.

Krzysztof C. Kiwiel graduated from Warsaw University of Technology, where he recieved his Ph.D. in mathematics in 1982, and the habilitation in 1988. Since 1982, he is with the Systems Research Institute, Warsaw. His main research activities are in the field of nondifferentiable optimization. He is the author of the book "Methods of Descent for Nondifferentiable Optimization" (Lecture Notes in Mathematics 1133, Springer-Verlag Berlin, 1985). In 1999 he recieved the research fellowship of the Alexander von Humboldt Foundation.

Matthias P. Nowak graduated from Humboldt University, Berlin, in 1996. Since 1996, he has been with the Ph.D. program *Geometrie und Nichtlineare Analysis* at the Humboldt-University Berlin. His research activities are in the field of optimization of large scale systems.

Werner Römisch graduated from Humboldt University, Berlin, where he recieved his Ph.D. in mathematics in 1975, and the habilitation in 1984. His main interests are Stochastic Optimization and Numerical Analysis of Stochastic Equations with a special emphasis on power systems optimization.

Isabel Wegner graduated from Humboldt University, Berlin, in 1999. In 1999, she joined the Ph.D. program *Geometrie und Nichtlineare Analysis* at the Humboldt-University Berlin where she is now interested in stochastic programming.