A Heuristic for the Long-Term Electricity Generation Planning Problem Using the Bloom and Gallant Formulation ⋆

Adela Pagès, Narcís Nabona ∗

Dept. of Statistics and Operations Research, Universitat Politècnica de Catalunya, 08028 Barcelona

Abstract

Long-term power planning is a stochastic problem often confronted by electrical utilities in liberalized markets. One can model it for profit maximization—using market-price estimation functions for each interval—by posing it as a quadratic programming problem with some linear equalities and an exponential number of load-matching linear inequality constraints.

In order to avoid handling all the inequalities when one is attempting to solve the problem, column generation methods have been employed herein. In this paper, we describe the foundations and implementation of a heuristic that tries to iteratively guess the active set of constraints at the optimizer, alongside a normal quadratic programming solution used at each iteration. The two methods are compared and the heuristic procedure is shown to be more efficient.

Key words: Energy, Heuristics, Large Scale Optimization, Quadratic Programming, Stochastic Programming
PACS: 89.30.Ee, 07.05.Tp

1 Introduction and Motivation

Long-term energy generation planning is an issue of key importance to the operation of electricity generation companies. It is used to budget for and plan

⋆ Partially supported by MCYT Grant DPI 2002-03330 of *Ministerio de Ciencia y Tecnología*, Spain

∗ Corresponding author. Tel. +34 934017035, Fax +34 934015855
Email addresses: adela.pages@upc.edu (Adela Pagès), narcis.nabona@upc.edu (Narcís Nabona).

fuel acquisitions and to provide a framework for short-term energy generation planning.

The long-term problem is a well-known stochastic optimization problem, as several of its parameters are only known as probability distributions, such as load, the availability of thermal units, hydrogeneration and energy generations from renewable sources in general.

A long-term planning *period* (e.g., a natural year) is normally subdivided into shorter *intervals* (e.g., weeks or months), for which parameters (e.g., the load-duration curve) must be predicted, and variables (e.g., the expected energy productions of each generator unit) must be optimized. The load-duration curves (LDC's) predicted for each interval, which are equivalent to load-survival functions, are used as data for the problem, which is appropriate since load uncertainty can be suitably described using the LDC. It is assumed that the probability of failure for each thermal unit is known.

Bloom and Gallant [2] proposed a linear model (with an exponential number of inequality constraints) and used an *active set* methodology [8] to find the optimal way of matching the LDC of a single interval using thermal units, in the presence of load-matching and other operational non-load-matching constraints. These might be limits on the availability of certain fuels or on emissions. The number of load-matching constraints (*lmc*) $n_i \times 2^{n_u}$ is exponential to the number of units n_u for each of the n_i intervals considered in the problem, and gets to be very large even in moderately sized problems.

When a long-term power planning problem needs to be solved for a generation company operating in a liberalized market, the company does not have a load of its own to satisfy, but rather bids the energies produced by its units to a *market operator*, which selects the lowest-priced energy to match the load from amongst the units of bidding companies. In this case, the scope of the problem is no longer that of the generation units of a single generation company, but that of all the units of all companies bidding in the same competitive market, which matches the load of the whole system. This entails planning problems that are much larger than before and it is a reason for developing more efficient codes for solving them.

The Bloom and Gallant model has been successfully extended to multi-interval long-term planning problems, using either the active-set method [10], the Dantzig-Wolfe column generation method [5,15] or the Ford-Fulkerson column-generation method [6,13]. A quadratic model for formulating the long-term profit maximization of generation companies in a liberalized market has been proposed [11] and column generation procedures have been employed to solve it [14,12]. However, to apply quadratic programming (QP) or interior-point quadratic programming (IPQP) directly is not practical, even in moderately sized problems, due to the exponential number of linear inequalities. This paper puts forward a heuristic for building up the optimal active set of inequality *lmc*s. It employs a reduced subset of *lmc*s, which is enlarged in successive steps until the optimal active set and solution are found. Since the QP subproblems at each step of the heuristic have a moderate number of inequalities, plain QP or IPQP solvers can be employed instead of specialized column generation algorithms.

The paper is organized as follows: Section 2 describes the problem; Section 3 describes the Bloom and Gallant formulation and the solution methods employed so far; Section 4 introduces the proposed heuristic; Section 5 details how to check the feasibility of a solution; and the computational results and the conclusions follow in Sections 6 and 7.

2 The Long-Term Electricity Generation Planning Problem

2.1 The LDC

The LDC is the most sensitive technique for representing the load of a future interval. The main features of an LDC can be described using five characteristics: the duration t, the peak load power $\widehat{p}$, the base load power $\underline{p}$, the total energy $\widehat{e}$ and the shape, which is not a single parameter and is usually described using a table of durations and powers, or using a function.

The LDC for future intervals must be predicted. For a past interval, for which the hourly load record is available, the LDC is equivalent to the load-over-time curve sorted in order of decreasing power. It should be noted that in a *predicted* LDC, random events such as weather or shifts in consumption timing, which cause modifications of different signs in the load tend to cancel out, and that the LDC maintains the power variability of the load in its entirety.

2.2 Unavailability of units and the convolution method

As far as loading an LDC is concerned, the relevant parameters of a thermal unit are the power capacity c_j for the $j^{\underline{\text{th}}}$ unit (the maximum power output in MW that the unit can generate), the outage probability q_j for the $j^{\underline{\text{th}}}$ unit (the probability of the unit not being available when it is required for generating power) and a linear generation cost $\widetilde{v}_j$ for the $j^{\underline{\text{th}}}$ unit (the production cost in €/MWh).

Given a set of generator units we wish to match the demand, which is a random variable. We define e_j as the expected energy generated by the $j^{\underline{\text{th}}}$ unit over the duration t of the LDC, and can compute it as follows:

$$e_j = t(1-q_j)\int_0^{c_j} xf(x)dx = t(1-q_j)\int_0^{c_j}[1-F(x)]dx$$

where $f(x)$ is the density function of the demand (see Fig. 1). The generation of a unit is related to the demand as the unit will generate as long as there is load to supply. $F(x)$ is the distribution function of $f(x)$, and the last equality holds because x is continuous and non-negative ($f(x) = 0$ for $x < 0$).

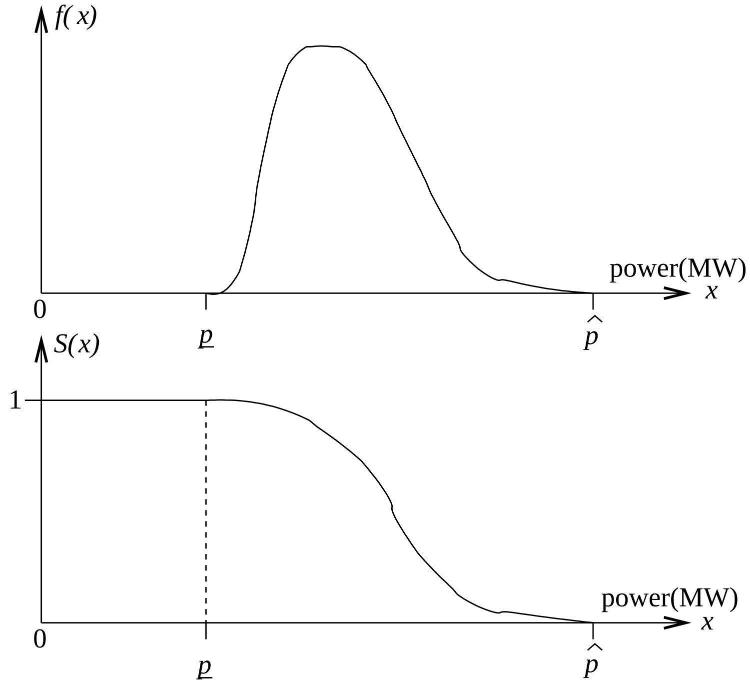

Fig. 1. Probability density function of load $f(x)$ (above), and load survival function $S(x)$ (below).

$S(x) = 1 - F(x)$ is called the load survival function and gives the probability of there being a load greater than x, $S(y) = prob(x \geq y)$ (see Fig. 1). We thus have:

$$e_j = t(1-q_j)\int_0^{c_j} S(x)dx\,.$$

Either density or survival functions can be used interchangeably, as we can derive one from the other. We prefer using the survival function, as it corresponds to the rotated and rescaled LDC.

The load has to be matched with the available units. Let Ω be the set of indices corresponding to the available units: Ω=$\{1, 2, \ldots, n_u\}$. The expected energy generation by each unit depends on the loading order. It is thus necessary that one knows the load survival function $S_\psi(x)$ after a subset $\psi\subset\Omega$ of units has already been loaded (survival function of still-unsupplied load — not yet eliminated by the units already loaded). Balériaux, Jamoulle and Linard de Guertechin [1] were the first to propose the following convolution:

$$S_{\psi \cup j} = q_j S_\psi(x) + (1 - q_j) S_\psi(x + c_j)\,, \tag{1}$$

which expresses the change to the load survival function caused by loading the $j^{\underline{\text{th}}}$ unit; the expected contribution of this unit being

$$e_j = t(1 - q_j) \int_0^{c_j} S_\psi(x) dx\,. \tag{2}$$

Let $S_\emptyset(x)$ be the load survival function corresponding to the LDC (prior to loading a generator unit). It is not difficult to derive, by successively applying (1), that, given a set ψ of unit indices j, the unsupplied load after loading all the units in ψ will have a survival function $S_\psi(x)$

$$S_\psi(x) = S_\emptyset(x) \prod_{m \in \psi} q_m + \sum_{\chi \subseteq \psi} \Big(S_\emptyset(x + \sum_{j \in \chi} c_j) \prod_{j \in \chi} (1 - q_j) \prod_{j \in \psi \setminus \chi} q_j \Big) \tag{3}$$

where χ represents any subset of ψ.

We can thus say from (3) that the survival function $S_\psi(x)$ of the unsupplied load is the same regardless of the order in which the units in ψ have been loaded.

The expected unsupplied energy $w(\psi)$ is computed as follows:

$$w(\psi) = \; t \int_0^{\widehat{p}} S_\psi(x)\, dx \tag{4}$$

The integration in (4) is to be carried out numerically.

2.3 Loading order and maximum expected generation of a given unit

Due to the outages of thermal units, the LDC does not coincide with the thermal units' estimated production. The installed capacity is usually higher than the peak load: $\sum_{j=1}^{n_u} c_j > \widehat{p}$.

The *generation-duration curve* (see Fig. 2) is the expected production of the thermal units over the time interval referred to by the LDC. The expected energy generated by each unit is the slice of area under the generation-duration curve that corresponds to the capacity of the thermal unit.

The probability that there will be time lapses within the time interval considered, in which, due to outages, there is not enough generation capacity to

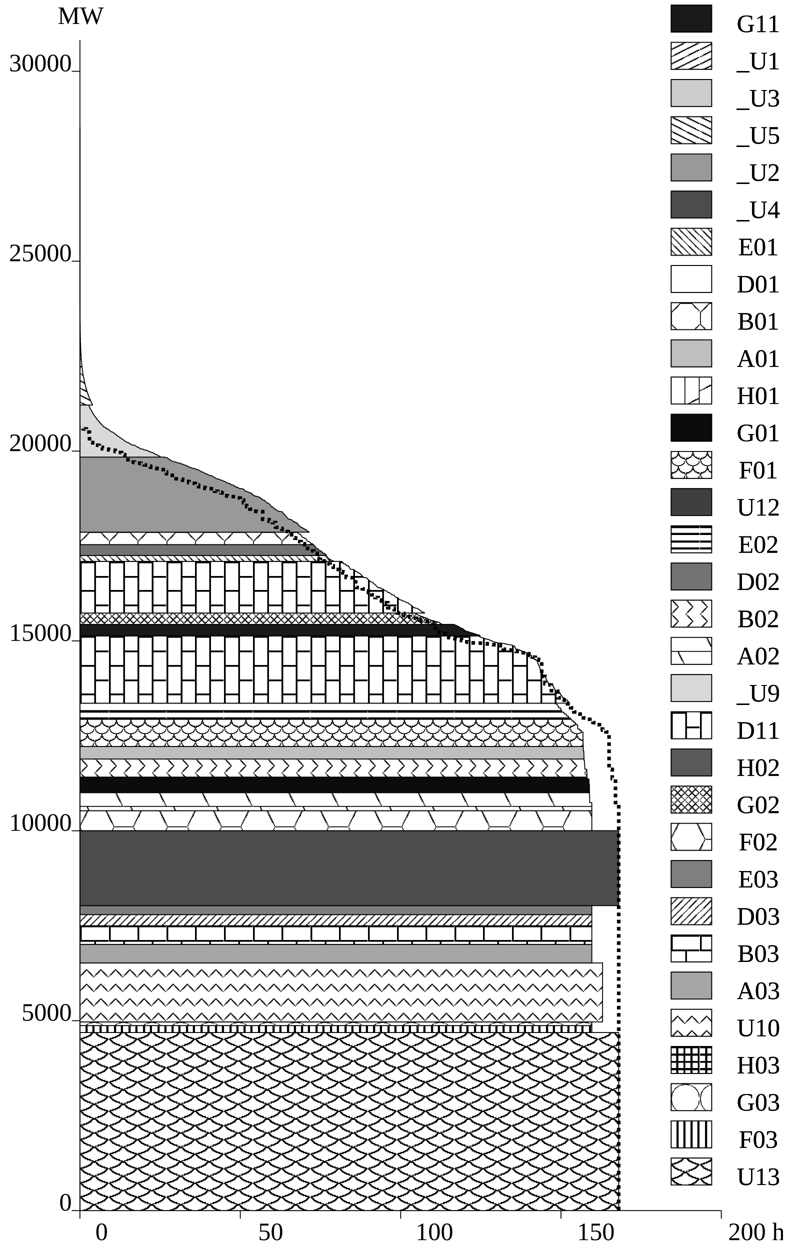

Fig. 2. Generation-duration curve and load-duration curve (dotted line) for a weekly interval in a 32 unit problem.

cover the present load, is not null. Therefore, in these cases *external energy* (from other interconnected utilities) will have to be imported and paid for at a price $\widetilde{v}_{n_u+1}$ that is higher than the most expensive unit in ownership. The peak power of the generation-duration curve is $\sum_{j=1}^{n_u} c_j + \widehat{p}$ and the area above power $\sum_{j=1}^{n_u} c_j$ is the external energy.

It is easy to verify from (1) that $S_{\psi \cup j} \leq S_{\psi}(x)\,, \forall\, x$. It then follows from (2) that loading a unit k after loading unit j will give a lower expected production e_k than if unit k were loaded just before unit j. The maximum production that can be expected of a given unit is obtained when this unit is loaded first. Thus

$$e_j \leq t(1-q_j)\int_0^{c_j} S_{\emptyset}(x)dx = \overline{e}_j\,. \tag{5}$$

The objective of long-term planning is to determine the *loading order* of the units and the corresponding values of the expected generations e_j that satisfy (2) at each interval (and thus match the LDC) and other operation constraints.

3 The Bloom and Gallant Formulation

3.1 Constraints in the Bloom and Gallant formulation

Bloom and Gallant [2] established that, in order for the expected energies e_j, $j{\in}\Omega$ to match the LDC, the linear inequality constraints

$$\sum_{j\in\psi} e_j \leq \widehat{e} - w(\psi) \qquad \forall\, \psi \subset \Omega$$

must be satisfied. Here, $w(\psi)$ is calculated as in (4). These constraints are referred to herein as *load-matching constraints* (*lmc*s).

Given that there are $2^{n_u}{-}1$ subsets of $\Omega{=}\{1,2,\ldots,n_u\}$, we have an exponential number of *lmc*s. (Thus, $2^{n_u}{-}1$ is over one million for $n_u{=}20$.)

There are other constraints that must be satisfied in terms of the e_js, such as the limited availability of fuels or emission limits over one or several intervals. These constraints are termed *non-load-matching constraints*.

3.2 Single-interval formulation that minimizes costs

The single-interval long-term planning model that minimizes production costs is then

$$\begin{aligned}
\underset{e_j}{\text{minimize}} \quad & \sum_{j=1}^{n_u+1} \tilde{v}_j \, e_j && (6)\\
\text{subject to} \quad & \sum_{j\in\psi} e_j \le \hat{e} - w(\psi) \quad \forall\, \psi \subset \Omega && (7)\\
& C\, e \ge d && (8)\\
& A\, e = s && (9)\\
& \sum_{j=1}^{n_u+1} e_j = \hat{e} && (10)\\
& e_j \ge 0 \qquad\qquad j = 1, \ldots, n_u, n_u + 1 && (11)
\end{aligned}$$

where n_u+1 is the index that represents the external energy, $C\in\mathbb{R}^{n_\ge \times n_u}$ and $d\in\mathbb{R}^{n_\ge}$ are the matrix and right-hand side (*rhs*) of non-load-matching inequality constraints, and $A\in\mathbb{R}^{n_= \times n_u}$ and $s\in\mathbb{R}^{n_=}$ are the matrix and *rhs* of non-load-matching equality constraints.

The objective function (6) can be simplified using (10), which leads to:

$$\sum_{j=1}^{n_u} v_j \, e_j + \tilde{v}_{n_u+1}\hat{e} \qquad \text{where} \quad v_j = \tilde{v}_j - \tilde{v}_{n_u+1}$$

in which $\tilde{v}_{n_u+1}\hat{e}$ is a constant.

3.3 Multi-interval formulation that maximizes profit

As power planning for a long time period cannot take into account changes over time to some of its parameters, the time period is subdivided into shorter *intervals* in which all the parameters can be assumed to be constant. We will use superscript ^{i} to indicate the variables and parameters that refer to the $i^{\underline{\text{th}}}$ interval.

Thus, some constraints refer only to variables that occur in a single interval, while others may refer to variables that belong to different intervals. By way of example, constraints on the minimum consumption of gas may affect several or all of the intervals, while an emission limit constraint or a constraint associated with the units that make up a combined-cycle unit may refer to a single interval.

The overhauling of thermal units must be taken into account. Therefore, there will be intervals in which some units must remain idle, and the set of available units in each interval may be different. Let Ω^i be the set of units available in the $i^{\underline{\text{th}}}$ interval, and let n_u^i=$|\Omega^i|$ be the cardinality of this set.

In liberalized markets, generation companies must bid their generation to a market operator and a market price is determined every hour by matching the demand with the lowest-priced bids. Generation companies are thus no longer interested in generating power at the lowest cost, but in obtaining a maximum profit, which is given by the difference between the revenue at market price and generation cost of any bids accepted. In long-term operation, all the bids accepted over a time interval (a week, or a month) must match the LDC of the interval. Loads matched by the market operator over weeks and months can be arranged into LDC's that can be predicted for future intervals. Units participating in a liberalized market have an outage probability, and a unit outage is made up for by other units participating in the market, or from outside the pool in emergencies, at the price of unscheduled exchanges. This situation satisfies the basic hypothesis of the Bloom and Gallant formulation [2].

From now on, the LDC considered refers to the load of the entire power pool, and must be matched with the generation of the units of all generation companies participating in it. When a Specific Generation Company (SGC) participating in the pool wishes to optimize its long-term planning, it must also take into account the units of the other participants, either as single units or merged into equivalent ones. Capacity and outage probability are usually public data and generation costs of competitors' units may be estimated with sufficient accuracy.

The market price at each time interval can be estimated for each LDC, as is indicated in Fig. 3, by adjusting a linear function of market-price change to interval duration: $b^i + l^i t$ (t being the interval duration and b^i and l^i parameters to be estimated).

The duration of expected generations over each interval is conditioned by the LDC matching. An estimated linear market-price function with respect to load duration is calculated for each interval. This function is not an estimation of a price-duration curve. Taking into account the estimated duration of the expected energy generated by the $j^{\underline{\mathrm{th}}}$ unit over the $i^{\underline{\mathrm{th}}}$ interval, e^i_j / c_j, the profit (revenue at market price minus cost) of unit j in interval i will be

$$\int_0^{e^i_j/c_j} c_j \Big\{ b^i + l^i t - v_j \Big\} \mathrm{d}t = \Big(b^i - v_j \Big) e^i_j + \frac{l^i}{2c_j} {e^i_j}^2 .$$

Adding up for all intervals and units, we get the profit function to be maximized, which is quadratic in the generated energies. A more detailed development can be found in [11,12].

The objective function for each interval expresses the expected profit, which is calculated as an expected mean revenue minus the generation cost. The expected mean revenue is calculated from a predicted market-price variation w.r.t. load duration, and it is here considered that the predicted market-price

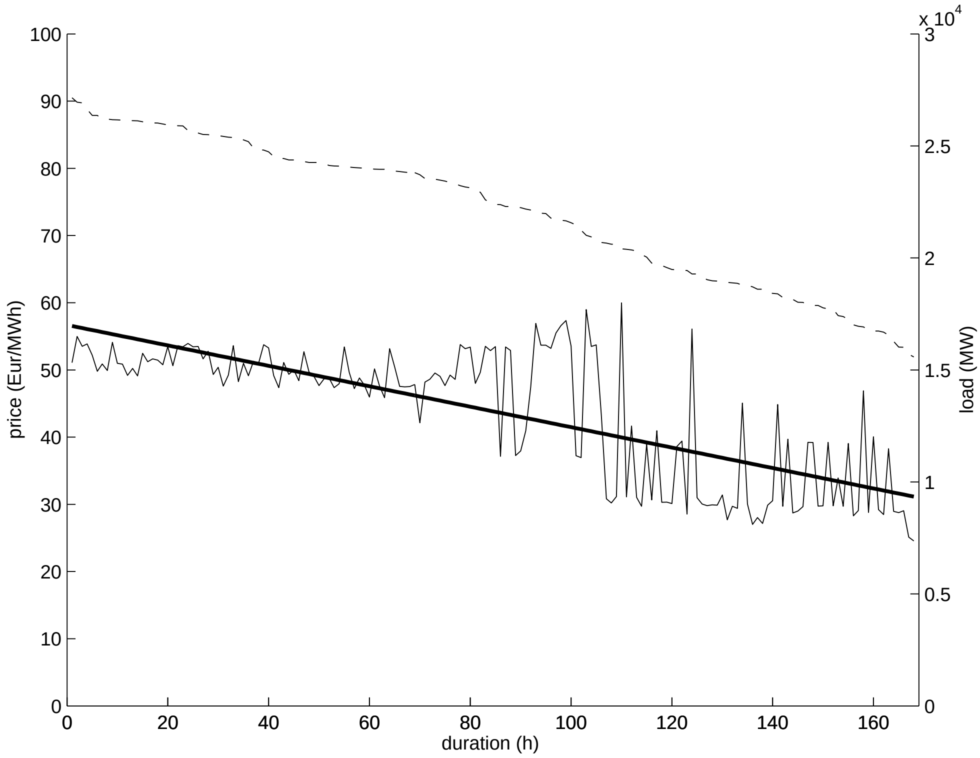

Fig. 3. Market prices ordered by decreasing load power (thin continuous curve) in weekly interval, market-price linear function (thick line), and LDC (dashed).

variation is not altered by generations in the interval, or equivalently, it considers that the influences of the different agents on the market price cancel each other out; therefore, the linear market-price change with load duration is independent from the generations. This is a reasonable assumption for a long-term horizon in an oligopolistic market, in which a price-demand correlation can easily be observed. From Figure 3 it is clear that the linear market-price change averages out the hourly changes of the market-price-to-load ratio.

The Bloom and Gallant quadratic profit-maximization formulation extended to n_i intervals, with inequality and equality non-load-matching constraints, may be expressed as:

$$\underset{e^i_j}{\text{maximize}} \sum_i^{n_i} \sum_j^{n_u} \left\{ \left(b^i - v_j\right) e^i_j + \frac{l^i}{2c_j} {e^i_j}^2 \right\} \tag{12}$$

$$\text{subject to: } \sum_{j \in \psi} e^i_j \leq \widehat{e}^i - w^i(\psi) \quad \forall \psi \subset \Omega^i \quad i = 1, \ldots, n_i \tag{13}$$

$$C^i\, e^i \geq d^i \qquad i = 1, \ldots, n_i \tag{14}$$

$$\sum_i^{n_i} C^{0i}\, e^i \geq d^0 \tag{15}$$

$$A^i\, e^i = s^i \qquad i = 1, \ldots, n_i \tag{16}$$

$$\sum_i^{n_i} A^{0i}\, e^i = s^0 \tag{17}$$

$$e^i_j \geq \underline{0} \qquad j = 1, \ldots, n_u \qquad i = 1, \ldots, n_i \tag{18}$$

where b^i and l^i are the basic and linear coefficients of the long-term market price function of the $i^{\underline{\text{th}}}$ interval; $C^i \in \mathbb{R}^{n^i_{\geq} \times n_u}$ and $d^i \in \mathbb{R}^{n^i_{\geq}}$ are the matrix and *rhs* of inequalities that refer only to interval i; $C^{0i} \in \mathbb{R}^{n^0_{\geq} \times n_u}$ and $d^0 \in \mathbb{R}^{n^0_{\geq}}$ are the matrix and *rhs* of inequalities that refer to more than one interval i; $A^i \in \mathbb{R}^{n^i_{=} \times n_u}$ and $s^i \in \mathbb{R}^{n^i_{=}}$ are the matrix and *rhs* of equalities that refer only to energies of interval i; and $A^{0i} \in \mathbb{R}^{n^0_{=} \times n_u}$ and $s^0 \in \mathbb{R}^{n^0_{=}}$ are the matrix and the *rhs* of equalities that refer to more than one interval.

The number of variables is now $\sum_i^{n_i} n^i_u$ and there are $\sum_i^{n_i} (2^{n^i_u} - 1)$ *lmc*s.

Should constraint sets (15) and (17), which are the multi-interval constraints, be empty, the problem would be separable into n_i subproblems, one for each interval. Otherwise a joint solution must be found.

3.4 Nested active load-matching constraints in feasible points

The structure of the coefficients in the left-hand side of any *lmc* (7), in a single-interval problem, and (13), in multi-interval problems, is a row vector of ones and zeros, depending on which units there are in the subset ψ considered. Regarding these ones and zeros of the *active lmc*s, in any solution point the ones must be *nested*. For the sake of brevity, we will say that the solution point has to be *nested*, without mentioning the active *lmc*s.

Without loss of generality, we will refer to a single-interval case, as *lmc*s in multi-interval problems refer to separate intervals.

3.4.1 Nested Constraints

A *lmc* is completely determined by the set of units it refers to. We use b_ζ to represent the row vector of coefficients of the left-hand side of the *lmc* built with the set of units ζ, which can be any subset of units.

It is said that a constraint b_ζ is *nested* into b_θ, where ζ and θ are any sets of units, if $\zeta \subset \theta$.

The following is an example in which b_ζ is nested into b_θ for $n_u = 6$:

	u_1	u_2	u_3	u_4	u_5	u_6	
b_ζ	.	1	.	1	.	.	$\zeta = \{2, 4\}$
b_θ	1	1	.	1	1	1	$\theta = \{1, 2, 4, 5, 6\}$

In general, a set of constraints is nested if we can order the constraints in the set in such a way that every constraint is nested in the next.

Following the example, a set of nested constraints might be:

		u_1	u_2	u_3	u_4	u_5	u_6	
1st	b_ζ	.	1	.	1	.	.	$\zeta = \{2, 4\}$
2nd	b_θ	1	1	.	1	1	1	$\theta = \{1, 2, 4, 5, 6\}$
3rd	b_Ω	1	1	1	1	1	1	$\Omega = \{1, 2, 3, 4, 5, 6\}$

where b_ζ is nested in b_θ and b_Ω, b_θ is only nested in b_Ω and b_Ω (where the set Ω contains all the units) nests b_ζ and b_θ.

3.4.2 Nested Feasible Points

As shown in Fig. 2, all solutions must be built from a loading order, and this order implies a nested set of *lmc*s. If a unit k is at the bottom of the generation-duration curve, this means that the equation $e_k \leq \hat{e} - w(k)$ should be active. If unit l follows, equation $e_k + e_l \leq \hat{e} - w(k, l)$ should also be active, and so on. Let us consider the six-unit case:

	u_1	u_2	u_3	u_4	u_5	u_6	u_4	u_2	u_5	u_6	u_1	u_3
order	5	2	6	1	3	4	1	2	3	4	5	6
$b_{\{4\}}$	.	.	.	1	.	.	1	.	.	.	.	.
$b_{\{2,4\}}$	.	1	.	1	.	.	1	1	.	.	.	.
$b_{\{2,4,5\}}$	.	1	.	1	1	.	1	1	1	.	.	.
$b_{\{2,4,5,6\}}$	.	1	.	1	1	1	1	1	1	1	.	.
$b_{\{1,2,4,5,6\}}$	1	1	.	1	1	1	1	1	1	1	1	.
b_{Ω}	1	1	1	1	1	1	1	1	1	1	1	1

A solution point with n_u active *lmc*s, in which each of these has one more 1 than the preceding constraint, is said to have a *perfect ordering*. Units can be reordered to produce a structure of ones shaped like a flight of stairs. In our example, we would say that the solution has a perfect ordering $\{4, 2, 5, 6, 1, 3\}$. The flight-of-stairs structure is shown to the right of the table.

When there are non-load-matching constraints active there will be less than n_u *lmc*s active; still, these active *lmc*s will be nested. This brings about a solution with a *partial order*. Returning to the six-unit case:

	u_1	u_2	u_3	u_4	u_5	u_6	u_4	u_1	u_2	u_5	u_6	u_3
$b_{\{4\}}$	.	.	.	1	.	.	1	.	.	.	.	.
$b_{\{1,2,4,5,6\}}$	1	1	.	1	1	1	1	1	1	1	1	.
b_{Ω}	1	1	1	1	1	1	1	1	1	1	1	1

Here we would say that the solution has a partial ordering: $\{4, (1, 2, 5, 6), 3\}$, in which there is no ordering to the units in parentheses. When the difference in units between two successive nested active constraints is greater than 1, it is said that there is a *landing*, as in the stair structure to the right of the table. Physically, a landing means that at least one unit (inside the parentheses) has its power capacity split by other units (inside the parentheses) when they are loaded. Fig. 2 shows that unit D11 is split by units G11 and G02.

3.5 Solution methods

There are several procedures for efficiently solving a problem such as (12-18). The active set method is one that considers the subsets of the constraints of at most $n_i \times n_u$ linear equalities and active linear inequalities (as many as there are variables) and discards constraints from this set and enters new ones as the optimization proceeds [10]. There is an *oracle* associated with load-matching inequalities in the Bloom and Gallant formulation, by which the search for an entering constraint is limited to specific subsets of load matching constraints rather than the full exponential number of constraints [2].

Column generation methods such as the Ford-Fulkerson and the Dantzig-Wolfe methods find feasible points and the solution by determining a convex combination of the vertices of the polyhedron defined by all the *lmc*s [12]. When one uses the Bloom and Gallant formulation, each of these vertices corresponds to a perfect ordering (as described in Subsection 3.4.2) given an order, which is determined by certain modified costs.

In either active-set or column-generation methodologies there is no need to explicitly create the exponential number of *lmc*s (with their long-to-compute *rhs*s).

However, should we attempt to employ direct quadratic programming [8] or interior-point quadratic programming [16], we would explicitly require the exponential number of inequalities, with its exponential number of slack variables and Lagrange multipliers. This would render the solution impossible in practice for $n_u \geq 20$. It is clear, however, that there are only a reduced number of active *lmc*s at the optimizer, and that finding this optimal active set is as difficult as finding the solution.

A heuristic is put forward herein for building the optimal active set of inequality *lmc*s. This heuristic employs a reduced subset of *lmc*s and is moderately enlarged in successive steps until the optimal active set and solution are found.

4 A Heuristic for Determining the Load-Matching Constraints That Are Active at the Optimizer

The heuristic we present herein is based on the fact that each solution corresponds to a loading order, and the active *lmc*s at the optimizer must be nested. It is an iterative process in which a few *lmc*s are added in successive steps.

4.1 Definitions and Sets

The heuristic consists in solving the problem several times, although a different subset of *lmc*s is used each time instead of the complete set. Let L be a set whose elements consist of subsets of units $\zeta \subseteq \Omega$, each of which determines a *lmc* to be taken into account.

*Lmc*s for a single-interval problem (7) are referred to matricially as $Be \leq r$, and a specific constraint as $b_\varsigma e \leq r_\varsigma$. For example, for $n_u = 6$, $L = \{\{1,3,4\}, \Omega\}$ stands for the constraints $b_{\{1,3,4\}} = [1.11..]$ and $b_\Omega = [111111]$. Their right-hand sides are $r_{\{1,3,4\}} = \widehat{e} - w(\{1,3,4\})$ and $r_\Omega = \widehat{e} - w(\Omega)$ respectively. The submatrix of *lmc*s will be represented by B_L, and its corresponding right-hand side by r_L.

At each iteration the heuristic adds a new constraint, which then forms a nested subset of constraints. For this reason, one needs to keep track of the subsets of units that are already nested. If we recall that Ω is the set of all units $\{1, \ldots, n_u\}$, φ stores the subset of the units that are already nested and η is its complementary: $\eta = \Omega \setminus \varphi$. Another important set is μ, which contains those units whose values are at maximum capacity at the initialization stage.

The variables, which stand for expected energies, have an explicit lower bound 0 and an upper bound $\overline{e}_j$, as calculated in (5). This upper bound is part of the *lmc*s, and will be imposed at all stages of the heuristic. The ratio $\rho_j = e_j / \overline{e}_j$, will be also employed in the heuristic as a measure of how far the value of a variable lies from its maximum.

4.2 Algorithm of the Heuristic for a Single Interval

Let us recast the single-interval problem (6-11), by eliminating e_{n_u+1} through (10); changing the objective function to profit maximization, as in the multi-interval case (12); using h for the linear coefficients and H for the matrix of quadratic coefficients; and substituting the set of *lmc*s (7) by a certain subset of them:

$$\begin{aligned} \underset{e}{\text{maximize}} \quad & h'e + \tfrac{1}{2} e' H e \\ \text{subject to} \quad & B_L e \leq r_L \\ & Ce \;\; \geq d \\ & Ae \;\; = s \\ & 0 \leq e \leq \overline{e} \end{aligned} \tag{19}$$

where $B_L e \leq r_L$ is a subset of the *lmc*s (7) that are fixed at each step by using list L, setting aside the upper bounds, which are also *lmc*s that are always imposed.

The heuristic has two main parts: initialization and iterations. It starts by solving the problem (19) with no *lmc*s, except the all-one *lmc* b_Ω, which nests any other constraint, though may not be active:

$$\sum_{j\in\Omega} e_j \le \hat{e} - w(\Omega): \qquad b_\Omega e \le r_\Omega \tag{20}$$

and the upper bounds $e_j \le \overline{e}_j$.

Once this problem is solved (using any methodology), the set μ of units at its maximum capacity is built:

$$\mu = \left\{ j \in \Omega \,\middle|\, \rho_j \simeq 1 \right\},$$

The constraints added to list L are all those made up of any subset of the units in μ: $\forall\,\zeta \subseteq \mu$. Usually this is a smaller set than Ω. Otherwise, there would be no point in solving the problem using the heuristic. The constraint $b_\mu e \le r_\mu$ is considered to be the last one entered.

In the iterative part, a new constraint is added at each iteration. The new constraint nests the former ones and answers which nested constraint is most likely to be violated next. It has the same units as the last constraint entered, plus whichever other unit is proportionally nearer to its upper limit.

The outline of the heuristic is as follows:

`I) Initialization part`
- Initialize the list of load-matching constraints with the all-one constraint $[11\cdots1]$: $L = \{\Omega\}$
- Compute the upper limit for each unit $j \in \Omega$ as in (5)
- Solve the problem (19)
- Compute the ratio for each $j \in \Omega$: $\rho_j := e_j / \overline{e}_j$
- Create set μ, to contain all the units that are at their respective upper bounds:
  $\mu := \{ j \in \Omega \mid \rho_j \simeq 1 \}$
- Add to the list L all possible combinations of units in μ:
  $L := L \cup \{\forall\,\zeta \subseteq \mu\}$
- Units considered nested are $\varphi := \mu$.
- Free units still to be nested are $\eta := \Omega \setminus \varphi$

`II) Iterative part`

Repeat while $|\eta| > 1$
- Solve (19)
- Update the ratios ρ_j for each $j \in \eta$
- Find the unit u which is nearest to its upper bound among units $\in \eta$:
  $u := \left\{ j \in \eta \mid \rho_j \ge \rho_k \ \forall k \in \eta \right\}$
- Update the set of units already nested: $\varphi := \varphi \cup \{u\}$

- Update the set of units still to be nested $\eta := \eta \setminus \{u\}$
- Add the new constraint to the list:
  $L := L \cup \{\varphi\}$

end

($|\eta|$ means: cardinality of set η).

4.3 Example of Application of the Heuristic to a Small Problem

The Appendix shows the application of the heuristic to a small single-interval problem.

4.4 The Heuristic in the Multi-interval Case

The heuristic for a multi-interval problem is quite similar to the single-interval one, because *lmc*s are defined within an interval and are not related between intervals.

The heuristic is applied to each interval as in a single interval case, yet upon each iteration every interval adds a new constraint. For each interval, the variables, sets and lists $\overline{e}^i_j$, ρ^i_j, L^i,μ^i, φ^i, η^i and $S^i_\emptyset$ are the same as in the case of a single interval, but now a superscript ^{i} identifies the interval in them. For each interval i, a separate submatrix of *lmc*s, $B^i_{L^i}$, and a separate vector of *rhs*s, $r^i_{L^i}$ are considered.

The stopping criterion is that all the units of all the intervals should be nested. If one interval finishes nesting before the iterations end, no more constraints are added to that interval. A new binary variable called *stop* is used.

The multi-interval model (12-18) is now recast as:

$$\begin{array}{lll}
\underset{e}{\text{maximize}} & \sum_i^{n_i}\{h^{i\prime}e^i + \frac{1}{2}e^{i\prime}H^ie^i\} & \\
\text{subject to} & B^i_{L^i}e^i \leq r^i_{L^i} & i = 1, \ldots, n_i \\
& C^i\, e^i \geq d^i & i = 1, \ldots, n_i \\
& \sum_i C^{0i}\, e^i \geq d^0 & \\
& A^i\, e^i = s^i & i = 1, \ldots, n_i \\
& \sum_i A^{0i}\, e^i = s^0 & \\
& 0 \leq e^i \leq \overline{e}^i & i = 1, \ldots, n_i
\end{array} \quad . \qquad (21)$$

The outline of the heuristic extended to n_i intervals is:

```
I) Initialization part
```
- Let $\{i \in 1..n_i\}$ $L^i := \{\Omega^i\}$
- Let $\{i \in 1..n_i,\ j \in \Omega^i\}$ $\overline{e}^i_j := t^i(1-q_j)\int_0^{c_j} S^i_\emptyset(x)dx$
- Solve (21)
- Let $\{i \in 1..n_i,\ j \in \Omega^i\}$ $\rho^i_j := e_j/\overline{e}^i_j$
- Let $\{i \in 1..n_i\}$ $\mu^i := \{j \in \Omega^i \,|\, \rho^i_j \simeq 1\}$ and $\varphi^i := \mu^i$
- Let $\{i \in 1..n_i\}$ $\eta^i := \Omega^i \setminus \varphi^i$
- Let $\{i \in 1..n_i\}$ $L^i := L^i \cup \{\forall \zeta \subseteq \mu^i\}$
- Let $stop := false$

```
II) Iterative part
```
repeat while $stop = false$
- Solve (21)
- Let $stop := true$
- *for* $i = 1..n_i$

    if $|\eta^i| > 1$

    · Let $stop := false$
    · Let $\{j \in \Omega^i\}$ $\rho^i_j := e^i_j/\overline{e}^i_j$
    · Let $u := \{j \in \eta^i \,|\, \rho^i_j \geq \rho^i_k\, \forall\, k \in \eta^i\}$
    · Let $\varphi^i := \varphi^i \cup \{u\}$
    · Let $\eta^i := \eta^i \setminus \{u\}$
    · Let $L^i := L^i \cup \{\varphi^i\}$

    end

  end

end

4.5 Comments on the Heuristic

For the expected energies e_js, the first solution of (19) in the heuristic (or of (21) in the multi-interval case) only considers the upper bounds and the all-one *lmc* (20), but not the rest of the *lmc*s. The non-load-matching constraints and the objective function will lead a subset μ of the unit energies to their upper bounds. The heuristic then includes all possible *lmc*s regarding this subset μ in the problem, which makes these unit energies lower, so that they conform to the shape of the generation-duration curve for the LDC. Other units then increase their expected energy generation, and they will be made to conform to the generation-duration curve one by one by nesting whichever one is closest to its upper bound with those units already nested. Since every time a constraint or group of constraints is added the optimization is solved with all the non-load-matching constraints, the solution whose units are all nested will most probably be feasible and optimal. The computational results reported in Section 6 prove that this is so.

Not all the constraints added by the heuristic will generally be active at the optimizer. The nested sequence obtained by the heuristic put forward may include one or several *landings*, as not all the constraints included by the heuristic are active at the solution point.

Another issue is the number of times Problem (21) is solved. As the heuristic ends when all the units are nested (and one of the not-yet-nested units is included per iteration), there will be at most n_u iterations of the heuristic.

As regards implementation, a safeguard against the maximum number of units at the upper bound in μ should be used. The cardinality of μ^i has never exceeded ten units per interval for the test cases solved, which means that the number of *lmc*s to be added (at most $(2^{10} - 1)n_i = 1023 \times n_i$) is acceptable.

The application of the heuristic is limited, in theory, to cases in which, as a result of the Initialization part, there are not all that many units (e.g., $|\mu|{\leq}18$) whose expected energies are at its upper limit, which are the most efficient ones. However, in practice, $|\mu|$ can still be higher because there is no need to include an exponential number ($2^{|\mu|}{-}1$) of load-matching inequality constraints in the Initialization part, because all constraints corresponding to any subset $\nu{\subset}\mu$ such that $\sum_{j\in\nu} c_j{\leq}\underline{p}$, where $\underline{p}$ is the base power of the LDC (also termed the LDC knee point), need not be included since they are a linear combination of the upper bounds of the units' expected energies, as can be easily deduced from the Expressions (1) and (2). Moreover, as is common engineering practice, many units of similar characteristics can be merged into a few equivalent units with no significant loss of quality in the results, which leads to a reduced number of efficient units; fuel availability limits and other constraints mean that many of the most efficient units do not reach their maximum energy output in all intervals.

In the heuristic, $|\eta^i|{>}1$ is considered to be a condition for including one more nesting constraint, and not ${>}0$, because, when only one unit is still to be nested, the energy-balance equation (20), which is already included, also nests the remaining unit.

Though any quadratic programming solver could be employed to solve subproblems (19), (or (21) in the multi-interval case) in the heuristic, one that could take advantage of the previous solution to start solving the next subproblem, in which one or more *lmc*s will have been added, would be more efficient. Using an Interior Point solver with *warm-start procedures* [9] could speed up the process of reaching the next optimal solution.

The heuristic put forward includes neither a feasibility nor an optimality check.

5 Checking the Feasibility of a Solution Found through the Heuristic

Though the heuristic presented refers to a single interval, it applies to any interval in a multi-interval problem. Before starting, we will enumerate several of the properties of $S(x)$.

From (1), it is easy to confirm that

$$S_{\zeta}(x) \geq S_{\theta}(x) \quad \forall\, \zeta \subseteq \theta$$

For any subset of units ψ, the right-hand side (7) can be rewritten as:

$$\widehat{e} - w(\psi) = t \sum_{j \in \psi} (1 - q_j) \int_0^{c_j} S_{\{\cup^{\tilde{j}} k \in \psi_o\}}(x) dx$$

where $\tilde{j}$ indicates the unit preceeding j in the loading order ψ_o (set ψ with elements in loading order); $\{\cup^{\tilde{j}} k \in \psi_o\}$ is the subset made up of the union of unit indices in ψ_o up to $\tilde{j}$ in the loading order; and $S_{\{\cup^{\tilde{j}} k \in \psi_o\}}(x)$ is the load survival function with which unit j has been loaded in loading order. As shown in (3) and (4), the right-hand side is not related to any order, so the sum can be done in any order.

Without a loss of generality, we can consider that the loading order for the units in any subset $\chi \subset \Omega$ will be based on the loading order of the solution for all the units in Ω. Let us now consider, as an example of the use of this notation, the subset of units $\chi \subset \Omega$:

set Ω	:	1	2	3	4	5	6	subset χ	:	2	3	4	6
solution order Ω_o	:	5	2	6	1	3	4	order χ_o	:	2	6	3	4

$$\begin{aligned}\widehat{e} - w(\chi) = \widehat{e} - w(\{2,3,4,6\}) = &\\ t\Big[(1 - q_2)\int_0^{c_2} S_{\emptyset}(x)dx + (1 - q_3)\int_0^{c_3} S_{\{2,6\}}(x)dx + &\\ (1 - q_4)\int_0^{c_4} S_{\{2,6,3\}}(x)dx + (1 - q_6)\int_0^{c_6} S_{\{2\}}(x)dx\Big]\,. &\end{aligned}$$

5.1 *Feasibility with a perfect ordering entails complete feasibility*

A solution with a perfect ordering implies n_u nested active *lmc*s, which is a completely determined system and can be solved using forward substitution.

For unit $j \in \Omega$, e_j could be calculated as

$$e_j = t(1 - q_j) \int_0^{c_j} S_{\{\cup^{\tilde{j}} k \in \Omega_o\}}(x) dx \,. \tag{22}$$

<u>Lemma 5.1</u>: *A solution with perfect ordering is feasible.*

Proof:

To check the feasibility of a perfect ordering solution Ω_o, we can consider any other (inactive) *lmc* χ that only involves the units in subset $\chi \subseteq \Omega$. Using (22), the alternative *lmc*

$$\sum_{j \in \chi} e_j = t \sum_{j \in \chi} (1 - q_j) \int_0^{c_j} S_{\{\cup^{\tilde{j}} k \in \Omega_o\}}(x) dx \leq t \sum_{j \in \chi} (1 - q_j) \int_0^{c_j} S_{\{\cup^{\tilde{j}} k \in \chi_o\}}(x) dx$$

is also satisfied as $S_{\{\cup^{\tilde{j}} k \in \Omega_o\}}(x) \leq S_{\{\cup^{\tilde{j}} k \in \chi_o\}}(x)$ $\forall x$, given that, for the same loading order, for any $\tilde{j}$, the units already loaded in the case of subset χ_o will be the same or less than in the case of using subset Ω_o. Therefore, a solution with a perfect ordering is feasible $\square$.

5.2 Potentially violated constraints with feasibility with respect to a partial ordering

In a solution with partial ordering, there can be perfectly ordered subsets of units and others that correspond to a split (which form a *landing* in the stair-like structure of the ones in the ordered active *lmc*s).

<u>Lemma 5.2</u>: *In a partial ordering solution, the only load-matching constraints that could possibly be violated correspond to constraints made up of all the units nested prior to a landing, plus the units that form the landing.*

Proof:

The perfectly ordered units correspond to a set of nested *lmc*s that can be reordered so that they present a flight-of-stairs structure. By subtraction of successive nested constraints, the expected value of a perfectly ordered unit is found to correspond to Expression (22). As shown in Lemma 5.1, the expected energies of these units are feasible for any *lmc*.

When a subset of units does not have a perfect ordering (at least one unit is split by others in the subset, and they form a *landing* in the stair structure), nothing prevents an ordering of these units from being infeasible, taking into account the set of units that has already been loaded.

This means that one has to check the *lmc*s of all the combinations of the units that correspond to the *landing*, which is exponential $2^{n_l}-1$, if there are n_l units in the *landing*. The heuristic does not perform these feasibility checks. They were performed using separate programs to verify the results given by the heuristic.

6 Computational Results

6.1 Test cases

Table 6.1 contains the dimensions of the test cases solved, including the number of units, intervals and non-load matching constraints for each test case. The total length of the period ($\sum_i t^i$) in weeks, and the week number in the first year of the first interval, which is always one week long in all test cases, are also included.

Table 1
Characteristics of test cases solved and active constraints at solution point

case	n_u	n_i	1st week	$\sum t^i$	$n_{\leq}$	*active* $n_{\leq}$	*consid.* n_{lmc}	*active* *up.bo.*	*active* n_{lmc}
ltp01	13	11	10	94	9	7	552	59	35
ltp02	15	11	10	94	43	11	181	22	55
ltp03	17	11	10	94	66	12	205	23	54
ltp04	18	11	10	94	77	19	222	24	58
ltp05	45	11	10	94	40	39	10614	108	43
ltp06	63	11	10	94	222	97	3729	82	196
ltp07	18	52	10	52	321	246	956	69	184
ltp08	25	27	10	53	190	64	914	76	115
ltp09	52	15	45	59	90	70	1971	32	31
ltp10	29	8	10	52	61	40	260	9	42
ltp11	33	13	31	52	34	19	522	12	24
ltp12	67	15	45	59	329	294	1636	74	258

These problems are realistic cases taken from the Spanish liberalized power pool. Each case refers to a specific generation company participating in the Spanish electricity market, and it includes the units of the specific generation company in full detail, plus those of all the competitors participating in the market, merged into a certain number of different units. Several of the cases may refer to the same specific generation company (e.g., cases ltp06 and ltp08) but the competitors are merged into greater or smaller number of units in different cases. The following (linear) non-load matching constraints were considered: the expected hydrogeneration in several basins; limitations on the availability of some fuel types; the minimum generation time over a time period by certain units (in order to qualify for a power warranty bonus in the Spanish pool regulations); market share constraints on the specific generation company; and special-regime minimum-generation limits [11].

6.2 Solutions through a column generation method and through the heuristic

The programs were run on a SPECfp2000 310 processor of a Hewlett Packard Netserver LC2000 U3.

Table 9 compares the solution obtained using a Column Generation method (CGM) to that obtained by the heuristic, using either *AMPL* [7] to program the heuristic and *Cplex* [4] as a quadratic programming solver, or with the heuristic coded in C and calling a quadratic primal-dual interior point code (IPM)) [16]. The results include the required CPU time, the objective function value, the total solver iterations and the iterations made by the heuristic, and the number of *rhs* terms calculated using the heuristic or the number of vertices (vx in the table heading) calculated by the column generation procedure. Obviously, the number of *rhs* terms calculated is equivalent to the number of *lmc*s considered by the heuristic.

Optimality detection in column generation methods is based on Lagrange multiplier positivity. Objective function values quite close to the optimal one are reached before the optimality condition is strictly satisfied, at an iteration in which the norm of the negative multipliers is small; from the iteration in which feasibility is reached, the objective function monotonically increases very slowly until it reaches the final value. Given that the heuristic yields objective function values that are very close to but not exactly equal to the optimal column generation solution, it may not be fair to compare CPU requirements for the heuristic and for a column generation procedure, if one considers that optimality is strictly guaranteed. Only the iterations needed to reach, in column generation, the same precision as the heuristic are required to compare the two procedures. In the test cases reported, this occurs roughly after 85% of the iterations needed to ensure optimality.

It must be borne in mind that it takes much longer to perform arithmetic calculations in an *AMPL* script, such as those of the *rhs*s of the *lmc*s (13) being considered, than it does to perform the same calculations in a user-developed code in C. This explains the long computation time required by the *AMPL+Cplex* solution as compared to the CPU time required by the code developed in C, when the same heuristic is being implemented.

On the right in Table 6.1 there is a comparison of the inequalities included in the model and of how many of them are active at the solution point. Column $n_{\geq}$ stands for the number of non-load-matching constraints of the model and the subsequent column shows the number of active ones. Equivalent results are displayed for the *lmc*s. The total number of these constraints that have been considered in the heuristic procedure is shown in n_{lmc}. The column headed by *active up.bo.* shows the number of units at their upper bound at the end of the procedure, which has a significant influence on the total number of *lmc*s (in the Initialization part). There is a close correlation between column *consid.* n_{lmc} and *active up. bo.*. The last column shows the number of independent active *lmc*s at the solution point.

6.3 Feasibility and optimality of the solutions found using the heuristic

Checking feasibility requires computing all the constraints nested between any two consecutive active *lmc*s that determine a *landing*. For large examples, there can be intervals in which a large number of units form a *landing*. An example of this is case *ltp05*, where, in the first interval, there is a landing made up of 38 units out of a total of 45. This amounts to $2^{38}-2$ computations of *rhs*'s, which would lead to many days of computation.

The feasibility of the solutions obtained using the heuristic of all the small cases ($n_u \leq 25$) has been thoroughly verified. For large cases ($n_u > 25$), the solution obtained using the heuristic can only be compared to that obtained using a column generation procedure, which is optimal (and feasible). It is hard to derive the active *lmc*s from a column generation solution, so the comparison is effected through the objective function value and the generation value at the solution.

Table 9 shows the objective function values obtained using column generation and those obtained using the heuristic. It can be seen that the objective function values agree at least up to the 7th figure, which is sufficiently good.

7 Conclusions

- We have presented a model of long-term electric generation planning in a competitive market using the Bloom and Gallant formulation.

- We expressed long-term profit maximization of generation utilities in a liberalized market as a quadratic objective function of expected energy generations.
- We addressed issues involved in the calculation of the expected generation and expected unsupplied generation relevant to the *rhs*s of load-matching constraints and for the feasibility in nested sets of active set load-matching equations.
- We explained the principles of loading orderings and nested sets of active load-matching constraints in detail.
- Perfect and partial orderings in unit loadings have been characterized.
- The principles of the heuristic were established and the heuristic was described as applied to single-interval and multi-interval cases.
- The requirements for verifying the feasibility of solutions obtained using the heuristic were established.
- The computational results, including the following, were presented:
  · Twelve realistic test cases of different characteristics and sizes.
  · The results yielded by a column-generation method and the computation time required.
  · The results yielded by and the computational requirements of two implementations of the heuristic: one coded in a script of the *AMPL* modeling language and the commercial barrier quadratic solver *Cplex*, and the other coded entirely in C with a primal-dual interior-point quadratic programming solver coded also in C.
- The computational results suggest that:
  · The heuristic is reliable because it provides feasible optimal solutions. Feasibility was checked using special purpose programs, for all cases except the exceedingly large ones. The objective function values coincide up to seven decimal figures with those (optimal and feasible) obtained using column generation methods.
  · The code in C of the heuristic and of the interior-point quadratic solver is faster than the implementation that uses *AMPL+Cplex*, due to the *AMPL* script's inefficiency at handling the heavy arithmetic calculations required for computing the *rhs*s.
  · The heuristic is much more efficient than the column-generation methods.

8 Appendix: Application of the Heuristic to a Small Single-Interval Problem

This section illustrates the application of the heuristic to a test problem presented by Conejo [3] and also used by Bloom and Gallant [2]. Although it has a linear objective function and null outage probabilities the heuristic works in the same fashion. It has n_u=9 units, 5 thermal units, and two storage units, which are modeled with two generators each, one for the charging side and another for the discharging one. An extra unit represents the external source (see Table 2). It is a single-interval problem ($n_i = 1$) with a duration of t=8. The AMPL data and solution files are available at `http://www-eio.upc.es/~apages`.

Table 2
Small problem data

	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{ext}
capacity	2	0.4286	2.5714	1	1	1	1	1	1	∞ (100)
cost	2	10	12	13	15.5	0	0	0	0	20
outage prob.	0	0	0	0	0	0	0	0	0	0

There are two linear equality non-load matching constraints for representing the storage units:

$$0.7e_6 + e_7 = 5.6$$
$$0.75e_8 + e_9 = 6.0$$

Units must match the LDC shown in figure 4.

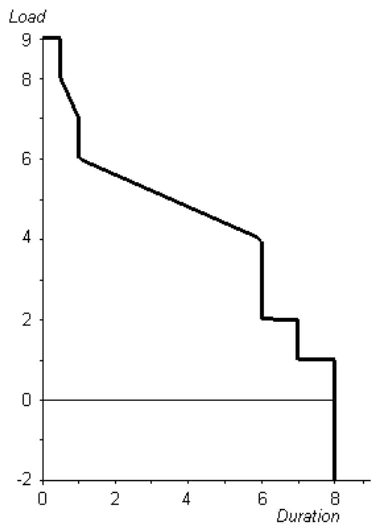

Fig. 4. Load Duration Curve

8.1 Initialization part

Solve the problem with the all-one *lmc*: $L_0 := \Omega$, $\Omega := \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8, u_9\}$:

Table 3
Solution obtained in the initialization part

	$L_0 := \Omega$								
	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9
e_j	16	3.43	16.83	0	0	8	0	8	0
$\bar{e}_j$	16	3.43	20.57	8	8	8	8	8	8
ρ_j	1	1	0.82	0	0	1	0	1	0

From Table 3, $\mu := \{u_1, u_2, u_6, u_8\}$ is the set of units that are generating at its maximum capacity ($\rho \simeq 1$). All possible *lmc*s formed with units in μ are considered in the model (see Table 4, L_1).

Next *lmc*s must nest units $\varphi := \mu = \{u_1, u_2, u_6, u_8\}$. The free units still to be nested are $\eta := \Omega \setminus \varphi := \{u_3, u_4, u_5, u_7, u_9\}$

8.2 Iterative part

As in the Initialization part 4 units out of 9 were fixed, there will be 5 more iterations.

Table 4
Solution obtained in the 1st iteration of the iterative part

	$L_1 := L_0 \cup \{\{u_1, u_2\}, \{u_1, u_6\}, \{u_1, u_8\}, \{u_2, u_6\}, \{u_2, u_8\}, \{u_6, u_8\}, \{u_1, u_2, u_6\}, \{u_1, u_2, u_8\}, \{u_1, u_6, u_8\}, \{u_2, u_6, u_8\}, \{u_1, u_2, u_6, u_8\}\}$								
	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9
e_j	16	2.57	17.93	0	0	8	0	7	0.75
$\bar{e}_j$	16	3.43	20.57	8	8	8	8	8	8
ρ_j	1	0.75	**0.87**	**0**	**0**	1	**0**	0.875	**0.09**

From the units in η , the nearest to its upper bound is $u := u_3$ (see Table 4), which leads us to consider the constraint: $\{u_1, u_2, u_3, u_6, u_8\}$. Update φ, η and L_2 accordingly:

$$\varphi := \quad \{u_1, u_2, u_3, u_6, u_8\} \qquad \eta := \quad \{u_4, u_5, u_7, u_9\}$$

From Table 5, $u := u_4$ is the unit in η with the highest value of ρ_j.

Table 5
Solution obtained in the 2nd iteration of the iterative part

$L_2 := L_1 \cup \{u_1, u_2, u_3, u_6, u_8\}$									
	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9
e_j	16	2.57	14.18	3.75	0	8	0	7	0.75
$\overline{e}_j$	16	3.43	20.57	8	8	8	8	8	8
ρ_j	1	0.75	0.69	**0.47**	**0**	1	**0**	0.875	**0.09**

$$\varphi := \quad \{u_1, u_2, u_3, u_4, u_6, u_8\} \qquad \eta := \quad \{u_5, u_7, u_9\}$$

Table 6
Solution obtained in the 3rd iteration of the iterative part

$L_3 := L_2 \cup \{u_1, u_2, u_3, u_4, u_6, u_8\}$									
	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9
e_j	16	3.43	14.18	2.25	0.88	7.57	0.30	6.57	1.07
$\overline{e}_j$	16	3.43	20.57	8	8	8	8	8	8
ρ_j	1	1	0.69	0.28	**0.11**	0.95	**0.04**	0.82	**0.13**

We repeat the procedure with unit $u := u_9$ because it is the one in η with the highest value of ρ (see Table 6). This gives

$$\varphi := \quad \{u_1, u_2, u_3, u_4, u_6, u_8, u_9\} \qquad \eta := \quad \{u_5, u_7\}$$

Table 7
Solution obtained in the 4th iteration of the iterative part

$L_4 := L_3 \cup \{u_1, u_2, u_3, u_4, u_6, u_8, u_9\}$									
	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9
e_j	16	3.43	14.18	2.25	0.88	7.48	0.37	6.67	1
$\overline{e}_j$	16	3.43	20.57	8	8	8	8	8	8
ρ_j	1	1	0.69	0.28	**0.11**	0.93	**0.05**	0.83	0.125

Finally, the last constraint to be added to the model is the one that has $u := u_5$ (see Table 7), which gives the optimal solution of Table 8.

$$\varphi := \quad \{u_1, u_2, u_3, u_4, u_5, u_6, u_8, u_9\} \qquad \eta := \quad \{u_7\}$$

Note that u_7 is left but that it would give the all-one *lmc* equation that is already in the model.

Table 8
Optimal solution: 5th iteration of the iterative part

$L_5 := L_4 \cup \{u_1, u_2, u_3, u_4, u_6, u_8, u_9\}$									
	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9
e_j	16	3.43	14.18	2.25	0.89	7.29	0.5	6.86	0.86
$\overline{e}_j$	16	3.43	20.57	8	8	8	8	8	8
ρ_j	1	1	0.69	0.28	0.11	0.91	0.06	0.86	0.11

References

[1] H. Balériaux, E. Jamoulle, and F. Linard de Gertechin. Simulation de l'explotation d'un parc de machines thermiques de production d'électricité couplé à des stations de pompage. *Revue E de la Soc. Royale Belge des Electriciens*, 7:225–245, 1967.

[2] J.A. Bloom and L. Gallant. Modeling dispatch constraints in production cost simulations based on the equivalent load method. *IEEE Transactions on Power Systems*, 9(2):598–611, 1994.

[3] A.J. Conejo. Optimal utilization of electricity storage reservoirs: efficient algorithms embedded in probabilistic production costing models. unpublished MSc thesis, EECS Dept., M.I.T., Cambridge, MA, USA, August 1987.

[4] CPLEX Optimization Inc. *Using the CPLEX Callable Library. User's Manual, version 4.0*, 1985.

[5] G.B. Dantzig and P. Wolfe. Decomposition principle for linear programmes. *Operations Research*, 8:101–110, 1960.

[6] L.R. Ford and D.R. Fulkerson. A suggested computation for maximal multicommodity network flows. *Management Science*, 5:97–101, 1958.

[7] R. Fourer, D.M. Gay, and B.W. Kernighan. *AMPL a modeling language for mathematical programming.* Boyd and Fraser Publishing Company, Danvers, MA 01293, USA, 1993.

[8] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization.* Academic Press, London, G.B., 1981.

[9] J. Gondzio. Warm start of the primal-dual method applied in the cutting-plane scheme. *Mathematical Programming*, 83:125–143, 1998.

[10] N. Nabona, C. Gil, and J. Albrecht. Long-term thermal power planning at VEW ENERGIE using a multi-interval Bloom and Gallant method. *IEEE Transactions on Power Systems*, 16(1):69–77, 2001.

[11] N. Nabona and A. Pagès. Long-term electric power planning in a competitive market using the Bloom and Gallant procedure and a modeling

language. Technical Report DR2002/24, Dept. Estadística i Inv. Operativa, Univ. Politèc. de Catalunya, 08028 Barcelona, 2002. Downloadable at `http://www-eio.upc.es/~gnom`.

[12] N. Nabona and A. Pagès. Long-term electric power planning in liberalized markets using the Bloom and Gallant formulation. In A. Alonso-Ayuso, E. Cerdá, L.F. Escudero, and R. Sala, editors, *Optimización bajo Incertidumbre*, pages 185–211. Tirant lo Blanch, Valencia 46010, Spain, 2004. ISBN 84-8456-100-3.

[13] A. Pagès. Procediment de Bloom i Gallant de planificació a llarg termini d'unitats tèrmiques de generació elèctrica usant la descomposició de Dantzig-Wolfe. Final-year project, Facultat de Matemàtiques i Estadística, Univ. Politècnica de Catalunya, 08028 Barcelona, 2002.

[14] A. Pagès and N. Nabona. Long-term electric power planning using the Bloom and Gallant's linear model. A comparison of solution techniques. Technical Report DR2002/25, Dept. Estadística i Inv. Operativa, Univ. Politèc. de Catalunya, 08028 Barcelona, 2002.

[15] J. Pérez-Ruiz and A.J. Conejo. Multi-period probabilistic production cost model including dispatch constraints. *IEEE Transactions on Power Systems*, 15(2):502–507, 2000.

[16] S.J. Wright. *Primal-dual interior-point methods.* SIAM, Philadelphia, PA 19104-2688, USA, 1997.

Table 9
Comparison of the solution methods

case	n_u	n_i	solver	CPU (sec.)	Obj. Fun.	Iter. total	Iter. heur.	rhs/vx evals.
ltp01	13	11	CGM	12	9536489725.21	258	-	182
			AMPL	2418	9536489727.70	153	10	552
			IPM	3	9536489598.95	303	10	552
ltp02	15	11	CGM	66	10961049053.36	1248	-	466
			AMPL	1595	10961049157.01	240	16	181
			IPM	3	10961049204.67	415	16	181
ltp03	17	11	CGM	128	10977720295.23	1934	-	578
			AMPL	1978	10977720268.52	274	18	205
			IPM	4	10977720279.39	471	18	205
ltp04	18	11	CGM	145	10979064725.32	2064	-	630
			AMPL	2260	10979064719.30	299	19	222
			IPM	5	10979064722.63	517	19	222
ltp05	45	11	CGM	576	8840160415.23	1815	-	1073
			IPM	100	8840159192.23	1589	37	10614
ltp06	63	11	CGM	2992	10667293261.44	11889	-	2357
			IPM	221	10667293157.78	2301	58	3729
ltp07	18	52	CGM	36809	5862769730.00	8440	-	3896
			AMPL	9273	5862769728.49	289	19	956
			IPM	81	5862769733.04	488	19	956
ltp08	25	27	CGM	16210	7077804321.54	4441	-	2041
			AMPL	8907	7077804305.06	470	26	914
			IPM	39	7077804323.51	876	26	914
ltp09	52	15	CGM	31964	5759806317.00	4956	-	6044
			IPM	80	5759806306.82	1706	53	1971
ltp10	29	8	CGM	345	5267939450.94	2062	-	1247
			AMPL	3192	5267939453.50	499	30	260
			IPM	6	5267939454.83	792	30	260
ltp11	33	13	CGM	1028	4868047080.54	1004	-	1068
			IPM	13	4868047082.54	931	34	522
ltp12	67	15	CGM	70748	5122060086.80	49757	-	18202
			IPM	410	5122061191.26	2020	67	1636