0.5 setgray0 0.5 setgray1

Warmstarting for Interior Point Methods Applied to the Long-term Power Planning Problem ⋆

Adela Pagès

Dept. of Statistics and Operations Research, Universitat Politècnica de Catalunya, 08034 Barcelona, Spain

Jacek Gondzio

School of Mathematics, University of Edinburgh, Edinburgh EH9 3JZ, UK

Narcís Nabona *

Dept. of Statistics and Operations Research, Universitat Politècnica de Catalunya, 08034 Barcelona, Spain

Abstract

The long-term planning of electricity generation in a liberalised market using the Bloom and Gallant model can be posed as a quadratic programming (QP) problem with an exponential number of linear inequality constraints called load-matching constraints (LMCs) and several other linear non-LMCs. Direct solution methods are inefficient at handling such problems and a heuristic procedure has been devised to generate only those LMCs that are likely to be active at the optimiser. The problem is then solved as a finite succession of QP problems with an increasing, though still limited, number of LMCs, which can be solved efficiently using a direct method, as would be the case with a QP interior-point algorithm. Warm starting between successive QP solutions helps then in reducing the number of iterations necessary to reach the optimiser.

The warm start technique employed herein is an extension of Gondzio and Grothey's approach to quadratic programming problems. We also propose how to initialise new variables in the problem to which a warm start technique is applied.

This study shows that warm starting requires on average 50% fewer iterations than a cold start in the test cases solved. The reduction in computation time is smaller, however.

Key words: Warmstarting, quadratic programming, long-term power generation planning, interior point method.

1 Introduction

The long-term planning of electricity generation in a liberalised market [11] using the Bloom and Gallant formulation [4] can be posed as a quadratic programming (QP) problem with an exponential number of linear inequality constraints called load-matching constraints (LMCs) and several other linear constraints (we will call them non-LMCs for short). Direct solution methods are inefficient at handling such problems and a heuristic procedure, called GP heuristic, has been devised [15] to generate only those LMCs that are likely to be active at the optimiser. The problem is then solved as a finite succession of QP problems with an increasing, though still limited, number of LMCs, which can be solved efficiently using a direct method, as would be the case with a QP interior-point algorithm. Warmstarting between successive QP solutions helps to reduce the number of iterations required to reach the optimiser. The warm start technique for interior point methods used herein is an extension of the one presented by Gondzio and Grothey [8].

Long-term energy generation planning is an issue of key importance to the operation of generation companies. It is used to budget for and plan fuel acquisitions and to provide a framework for short-term energy generation planning.

A long-term planning period (e.g. one year) is subdivided into shorter intervals, for which parameters are known or predicted. The variables to be optimised are the expected energy productions of each generating unit in each interval. In long-term planning, it is the production of each unit in the whole interval that is of relevance, rather than the generation rate per hour.

In each interval, the balance of the load must be satisfied, in addition to several other technical or economical constraints (e.g. maximum hydro generation, limits on emissions and market share constraints). A Load Duration Curve (LDC) is a practical means for representing the load of a future interval.

Bloom and Gallant [4] proposed a linear model to find the optimal method for matching the LDC of a single interval and subject to any other linear operational constraints. This model can easily be extended to the multi-interval

⋆ Partially supported by MEC Grant DPI 2005-09117-C02-01 of *Ministerio de Educación y Ciencia*, Spain

* Corresponding author. UPC Campus Nord, C5-211, c. Jordi Girona 3 – 5, 08034 Barcelona, Spain. Tel. +34 934017035, Fax +34 934015855

Email addresses: adela.pages@upc.edu (Adela Pagès), J.Gondzio@ed.ac.uk (Jacek Gondzio), narcis.nabona@upc.edu (Narcís Nabona).

URLs: http://www-eio.upc.es/~apages/ (Adela Pagès), http://www.maths.ed.ac.uk/~gondzio/ (Jacek Gondzio).

case.

When a long-term power planning problem needs to be solved for a generation company operating in a liberalised market, the company will not have a load of its own to satisfy, but will rather bid the energy 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 match the load of the whole system. The goal of a company participating in a liberalised market is to maximise its profit, which is understood as the revenue for its generation at market price minus the generation costs. Using a linear function of the market price the resulting objective function is quadratic on the expected energies.

Bloom and Gallant [4] proposed an active set methodology in which a small subset of LMCs is considered and updated at each iteration. Computational experience [14] shows that such an approach is slow to converge. Pérez-Ruiz and Conejo [16] proposed using a Dantzig-Wolfe decomposition. This approach makes it unnecessary to formulate the LMCs explicitly by generating vertices of the polyhedron defined by the load-matching and non-negativity constraints. However, solution accuracy is hard to achieve [12].

Pagès and Nabona [15] proposed a heuristic that tries to guess the active LMCs at the optimiser using a limited number of LMCs. This heuristic, called GP heuristic because it is used for solving the generation planning problem, solves a sequence of problems (as many as the number of units, at most) in which some new LMCs are included at each stage. The solution is found using interior point methods. The problems only differ in the fact that new LMCs are added, which makes the use of warm-start techniques a good strategy for obtaining the new solution more quickly.

There are many papers on warmstarting for interior point methods, mainly for linear programming. Gondzio [7] applies warm-start techniques to a cutting plane scheme where new columns are appended. Yildirim and Wright [18] give two different approaches with the theoretical worst-case analysis, where the size of the problem does not change. Gondzio and Grothey [8] describe a new version of the warm start for linear programming and problems of constant size. This paper extends the Gondzio and Grothey approach to the quadratic case and to the infeasible algorithm. The main difference between the linear and the quadratic case is that in the latter primal variables directly contribute to dual feasibility.

Benson and Shanno [2] introduce a warmstarting procedure for linear programming. The problem is modified by introducing new positive variables

whose negative values relax the nonnegativity of the original primal and dual variables. These new positive values are added to the primal and dual objective functions through ℓ_1 penalty terms. The warm-start initial point is the optimal solution of the unperturbed problem. The structure of the reduced KKT system is similar to that of the original problem, but it neither requires factorization refinement nor produces very short steps at the initial iterations. The same authors [3] have extended their warmstarting technique to nonlinear programming problems. Both in linear and in nonlinear cases, a logarithmic barrier of the relaxed variables plus their corresponding new variables is employed to develop the first order optimality conditions. Forsgren [5] develops a theoretical warm start for nonlinear programming which distinguishes between "almost active" and "almost non-active" constraints at the optimal point. For warmstarting, it is assumed that the active set does not change for the new solution. The slacks of the "almost active" constraints and the dual variables of the rest are eliminated from the Newton system.

In most of the papers, new instances do not change size. In our case new constraints (LMCs) are appended from one iteration to the next. We propose a strategy for initialising the new variables and for computing a direction to quickly recover primal and dual feasibility.

Section 2 is a brief description of the long-term electric power generation planning model. Section 3 outlines the main steps in the LMC oracle heuristic, and describes the nature of the cuts added. Section 4 considers the warm-start technique from a general point of view and Section 5 features the computational results of using warm starts and the GP heuristic method to solve the long-term power generation planning problem.

2 Problem formulation

This section is devoted to the long-term generation planning problem. The whole period considered is split into n_i intervals, because some parameters are not constant throughout the period or some constraints refer to brief lapses.

The most important part of the formulation is modelling the matching of the load for each interval. The load of a future interval is forecast using a Load Duration Curve (LDC) whose shape limits the generation of certain units. Given that the problem is set in a liberalised market framework, the only LDC that can be predicted is the system load. This makes it necessary to include a representation of all the units in the pool. As Bloom and Gallant proposed [4], the matching of the load is formulated by an exponential number of inequality constraints.

The long-term generation planning problem is formulated as a linearly constrained model with a quadratic objective function. A detailed explanation of the model can be found in [12].

2.1 Load Duration Curve

The LDC is the most natural technique for representing the load of a future interval. 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.

For a future interval, the base load $\underline{p}$, the peak load $\widehat{p}$, the interval energy, $\widehat{e}$, and the shape of the LDC must be predicted.

2.2 Thermal Units

The relevant parameters of a thermal unit j are the power capacity c_j (the maximum power output in MW that the unit can generate), the outage probability q_j (the probability of the unit not being available when it is required for generating power) and a linear generation cost v_j (the production cost in €/MWh). Let Ω be the set of units and n_u the number of units.

2.3 Matching of the Load

Let e_j be the expected generation of unit j. The expected generation of each unit depends on the *loading order* used to dispatch the generation in order to match the LDC.

Let ψ be a subset of units. The load survival function, $S_\psi(x)$, after the units in ψ have been loaded, where $\psi \neq \varnothing$, satisfies:

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

where χ represents any subset of ψ different from $\varnothing$. When $\psi = \varnothing$, $S_\varnothing(x)$ corresponds to the LDC rescaled and rotated. The load survival function, $S_\psi(x)$, is computed from $S_\varnothing(x)$ through convolution, as Balériaux *et al* [1] proposed.

The expected generation e_j, of unit j for a particular interval of length t and loaded after all units in ψ, is

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

Given that the load-survival function follows the property $S_\psi(x) \leq S_\theta(x)$ for $\theta \subseteq \psi$, the maximum expected generation of a unit is achieved when unit j is loaded first:

$$\overline{e}_j = t(1 - q_j) \int_0^{c_j} S_\varnothing(x)\, dx \tag{1}$$

2.4 Bloom and Gallant's model for matching the LDC

Bloom and Gallant [4] established that, in order for the expected energies e_j, $j \in \Omega$, to match the LDC, the linear inequality constraints

$$\sum_{j \in \psi} e_j \leq \widehat{e} - w(\psi) \tag{2}$$

must be satisfied for all subsets ψ of Ω. The expected unsupplied energy, $w(\psi)$, when the units in ψ have been loaded, is:

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

Given that there are $2^{n_u} - 1$ subsets of $\Omega = \{1, 2, \ldots, n_u\}$, we have an exponential number of linear inequality constraints. For 20 units, there are over one million LMCs.

These constraints ensure that any possible ordering of the units is feasible although only one of them will be optimal.

2.5 Non load-matching constraints

The objective of long-term planning is to determine the optimal *loading order* of the units and the corresponding values of the expected generations e_j at each interval (and thus match the LDC) and other operational constraints. Matching the LDC is expressed by an exponential number of LMCs. There are other constraints that must be satisfied in terms of the expected energies, 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* (non-LMCs) and are modelled through linear constraints.

2.6 Profit maximisation

In liberalised markets, generation companies 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 interested in obtaining a maximum profit, which is given by the difference between the revenue at market price and the 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.

An estimated linear market-price function with respect to load duration is calculated for each interval: $b^i + l^i t$ (t being the interval duration and b^i and $l^i < 0$ the parameters to be estimated). Taking into account the estimated duration of the expected energy generated by unit j in interval i, e^i_j/c_j, the profit (i.e. the revenue at market price minus cost) will be

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

If we add up all the intervals and units, we obtain the profit function to be maximised, which is quadratic on the generated energies. A detailed explanation can be found in [11].

2.7 Long-term electric power generation planning

The Bloom and Gallant quadratic profit-maximisation formulation extended to n_i intervals is expressed as:

$$\underset{e^i_j}{\text{maximize}} \sum_i^{n_i} \sum_j^{n_u} \left\{ \Big(b^i - v_j \Big) e^i_j + \frac{l^i}{2c_j} {e^i_j}^2 \right\} \tag{3}$$

$$\text{subject to: } \sum_{j \in \psi} e^i_j \leq \widehat{e}^i - w^i(\psi) \quad \forall \psi \subset \Omega \quad i = 1 : n_i \tag{4}$$

$$C\, e \geq d \tag{5}$$

$$e^i_j \geq \underline{0} \qquad j = 1 : n_u \qquad i = 1 : n_i \tag{6}$$

where n_i is the number of intervals, n_u is the number of units, 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; $C \in \mathbb{R}^{n_\geq \times (n_i \cdot n_u)}$ is the (multi-interval) matrix of non-LMCs, e is the vector notation of variables e^i_j and $d \in \mathbb{R}^{n_\geq}$ the corresponding right hand side vector of non-LMC inequalities.

The number of variables is equal to $n_i \times n_u$ and there are $n_i \times (2^{n_u} - 1)$ LMCs. A more detailed development of the model can be found in [11].

3 Solution method

Several approaches to solving the long-term electric power planning problem using the Bloom and Gallant formulation have been considered. A direct approach (standard QP or Interior Point QP) is of no use due to the exponential number of LMCs. The problem does not only arise from storage requirements but also from the complexity of computing the corresponding right-hand sides of (4), which is very time-consuming. These computations require numerical integration and convolutions of the load survival function. Moreover, only a few of the LMCs (at most $n_i \times n_u$ out of $n_i \times (2^{n_u} - 1)$) are active at the optimiser.

In this section we outline the main ideas behind the GP heuristic for the LMC oracle introduced in [15]. The heuristic is an iterative process in which a succession of similar problems must be solved. This section presents the problems that need to be solved and the cuts generated at each stage.

3.1 Outline of the GP heuristic

The GP heuristic exploits the fact that any feasible solution must correspond to a loading order of the units (to match an LDC, which can be different for each interval). The translation of a loading order into load-matching constraints is a set of n_u constraints nested in keeping with the loading order.

Let us recall the expression of a load-matching constraint (2):

$$\sum_{j \in \psi} e_j \leq \widehat{e} - w(\psi).$$

A set ψ of units determines a unique LMC. The body of the constraint has a 1 for each unit in ψ and the right-hand side depends on ψ. It is said that the constraint formed by the set ζ is nested by the constraint defined by θ if $\zeta \subset \theta$.

The following is an example in which the constraint defined by ζ is nested by the one defined by θ for a case with $n_u = 6$ units:

	u_1	u_2	u_3	u_4	u_5	u_6
$\zeta = \{u_2, u_4\}$	.	1	.	1	.	.
$\theta = \{u_1, u_2, u_4, u_5, u_6\}$	1	1	.	1	1	1

In general, a set of LMCs is nested if it is possible to sort each constraint of the set in such a way that each constraint is nested in the next. Similarly, the unit sequence of the loading order can be deduced from the nested set of LMCs as the unit that differs between two consecutive constraints. Following with the example, for a loading order $\{u_2, u_4, u_1, u_5, u_6, u_3\}$ the corresponding LMC coefficients are:

loading order:	u_2	u_4	u_1	u_5	u_6	u_3
LMC	1	.	.	.	.	.
	1	1	.	.	.	.
	1	1	1	.	.	.
	1	1	1	1	.	.
	1	1	1	1	1	.
	1	1	1	1	1	1

For a point at which none of the non-LMCs is active, all the n_u load-matching constraints corresponding to a certain ordering will be active. Otherwise, only a subset of LMCs will be active, although it is guaranteed that this subset will be nested [10].

The procedure characterised below refers to a single interval but is applied to all intervals simultaneously. Given that the LMCs refer to a single interval, the loading order may be different in each interval. The GP heuristic has three main stages:

- Initialization stage:

  Solve the problem with the non-LMCs, the upper bound of the variable (1) and only the *all-one LMC* (of each interval): $\sum_{j\in\Omega} e_j \leq \widehat{e} - w(\Omega)$. The all-one LMC always nests any other constraint.

	u_1	u_2	u_3	u_4	u_5	u_6
list of LMCs	1	1	1	1	1	1

- Self-ordering stage:

  From the previous solution, the subset ϕ of units that generate at their maximum capacity is chosen. Then, all LMCs that are composed exclusively of units in ϕ are added to the new problem.

  The incorporation of these constraints means that the new solution will

have the best ordering for the units in ϕ, thus ensuring that any other ordering with these units in the highest position will be feasible. Computational experience shows that at most 10 units per interval are at their upper bound, $|\phi| \leq 10$, and $(2^{10} - 1)n_i$ is still an acceptable number of LMCs. For example, for a solution in which the units at the upper bound were $\phi = \{u_1, u_2, u_4\}$, the new problem would have the following LMCs:

	u_1	u_2	u_3	u_4	u_5	u_6
list of LMCs	1	1	.	.	.	.
	1	.	.	1	.	.
	.	1	.	1	.	.
	1	1	.	1	.	.
	1	1	1	1	1	1

Note that LMCs with only one unit are not explicitly stated here because they are already considered as the upper bounds of the variable.

- Iterative stage:

  The third stage consists in solving a sequence of problems in which only one new LMC (per interval) is added at each iteration. This new constraint nests the former ones except the all-one LMC and has one unit of difference with respect to the last LMC considered.

  For example, u_5 could be the next unit in the loading order and constraint

	u_1	u_2	u_3	u_4	u_5	u_6
new LMC	1	1	.	1	1	.

  which nests constraint $\{u_1, u_2, u_4\}$ would be added to the list of LMCs considered.

The heuristic requires that a range of similar problems be solved. It employs a reduced subset of LMCs and is moderately enlarged in successive steps until the optimal active set and solution are found. The use of direct methods is therefore appropriate. We solved the problems using interior point methods. Warm start techniques reduce the number of iterations needed to find the successive solutions.

4 Warm Start applied to Interior Point Methods

4.1 Interior Point Steps

Within a general framework, consider the quadratic problem

$$\begin{aligned} \text{minimize} \quad & h^T x + \tfrac{1}{2} x^T H x \\ \text{subject to} \quad & Bx \leq d \\ & x \geq 0. \end{aligned} \tag{7}$$

Any of the problems defined in the GP heuristic described in section 3.1 is covered by such a formulation. The inequality constraint is converted into an equality constraint by adding the nonnegative slack variable f. Let u and z be the Lagrange multipliers associated with constraints $Bx + f = d$ and $x \geq 0$, respectively. The first-order optimality conditions of (7) are

$$\begin{aligned} Bx + f &= d \\ -B^T u - Hx + z &= h \\ Xz &= 0 \\ Fu &= 0 \\ x, z, f, u &\geq 0. \end{aligned} \tag{8}$$

The first set of equations ensures *primal feasibility*, the second *dual feasibility* and the last two *complementarity*. We use the upper-case letters X and F to denote the diagonal matrices formed by spreading elements of vectors x and f on the diagonal of these matrices, respectively (a common notation in the interior point literature [17]).

A primal-dual interior point method relaxes the complementarity with the parameter μ and finds an approximate solution of the perturbed first-order optimality conditions

$$\begin{aligned} Bx + f &= d \\ -B^T u - Hx + z &= h \\ Xz &= \mu e \\ Fu &= \mu e \\ x, z, f, u &> 0. \end{aligned} \tag{9}$$

The first-order optimality conditions of (7) when the logarithmic barrier of

parameter μ is applied to the non-negativity of the variables are the same as (9). The parameter μ is thus commonly known as the *barrier parameter*. At each iteration of an interior-point method, the parameter μ is reduced.

The approximate solution of (9) is found by applying one iteration of the Newton method. The system to be solved at each iteration is

$$\begin{bmatrix} -H & I & -B^T \\ B & I & \\ Z & & X \\ & U & F \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta f \\ \Delta z \\ \Delta u \end{bmatrix} = \begin{bmatrix} h + B^T u + Hx - z \\ d - Bx - f \\ \mu e - Xz \\ \mu e - Fu \end{bmatrix}. \tag{10}$$

A primal step-length α_P and a dual step-length α_D are chosen to ensure that all components of the new iterate are positive.

The *central path* is the sequence of points (x, f, z, u) that depend on μ and that satisfy (9), with $(x, f, z, u) > 0$. It is proved that if the interior of the primal and dual feasible region is non-empty the central path is unique [17].

4.2 Warm Start Framework

The technique known as *warmstarting* takes advantage of some prior knowledge of the problem (for example, by drawing on a solution to a similar problem) to produce an initial point that should lead to better performance than by starting the algorithm from scratch, see for example [7].

Suppose problem (7) is solved. From its solution, and according to the GP heuristic, new constraints are generated

$$\widetilde{B}x \leq \widetilde{d}, \tag{11}$$

where $\widetilde{B}$ is the matrix of coefficients for the new constraints and $\widetilde{d}$ is its right hand side. The new problem to be solved is

$$\begin{aligned} \text{minimize} \quad & h^T x + \tfrac{1}{2} x^T H x \\ \text{subject to} \quad & Bx \leq d \\ & \widetilde{B}x \leq \widetilde{d} \\ & x \geq 0. \end{aligned} \tag{12}$$

Our goal is to find the new optimiser of (12) as fast as possible using information collected from the previous solution to (7). The first order optimality conditions for the new problem are:

$$\begin{aligned} Bx + f &= d \\ \widetilde{B}x + \widetilde{f} &= \widetilde{d} \\ -B^T u - \widetilde{B}^T \widetilde{u} - Hx + z &= h \\ Xz &= \mu e \\ Fu &= \mu e \\ \widetilde{F}\widetilde{u} &= \mu e \\ x, z, f, \widetilde{f}, u, \widetilde{u} &\geq 0, \end{aligned} \tag{13}$$

where $\widetilde{f}$ are the slack variables of the new constraints and $\widetilde{u}$ are the associated Lagrange multipliers. Again, $\widetilde{F}$ is a diagonal matrix formed from the vector $\widetilde{f}$.

The differences between the old problem (see equations (9)) and the new one (13) as regards the modified first order optimality conditions are:

- Regarding the variables, the aforementioned new primal slacks, $\widetilde{f}$, and the corresponding dual variables, $\widetilde{u}$, of the new constraints.
- Regarding the constraints, the new primal constraints, $\widetilde{B}x + \widetilde{f} = \widetilde{d}$, a new term, $\widetilde{B}^T\widetilde{u}$, in the dual constraints, and the complementarity condition for the new variables, $\widetilde{F}\widetilde{u} = \mu e$.

4.3 Restoring Primal and Dual Feasibilities

Although in practice any positive value can be employed as an initial point, the theory of the path-following algorithms requires that the iterates stay in a neighbourhood of the central path [17]. However, since this neighbourhood is rather wide, the practical requirement is that the iterates simply stay away from the boundary of the feasible region.

As an illustration, we have computed the exact points in the central path for several values of μ of a small problem. We have re-solved the problem with one new constraint. In Figure 1, both central paths are plotted. The extra constraint and the new central path are represented by the dotted lines. The small problem, with $x \in \mathbb{R}^2$, has a quadratic objective function, with $h = [-34;\ -68]$ and $H = [3\ 2; 2\ 12]$. There is one constraint: $B = [0\ \ -1]$, $d = [-5]$. The new constraint has coefficients $\widetilde{B} = [-1\ \ -1]$ and $\widetilde{d} = [-8]$.

Figure 1(a) shows the central paths for the primal variables (the slack variables f are implicit). An $*$ mark highlights the points with μ equal to 10^3, 10^2, 10, 1 and 0.1. Figure 1(b) displays the dual variables in terms of μ. Note that the bottom-right plot of the dual variables corresponds to the new variable $\widetilde{u}_2$, which is present only on the new problem. This variable was not present in the old problem, therefore it does not have a solid line.

In this example, the additional constraint does not cut across the the central path but it shrinks the feasible area. Then, if the feasible area changes, the central path moves. By means of this example, we wish to illustrate that even if the starting point remains feasible for the new problem, some special steps must always be taken.

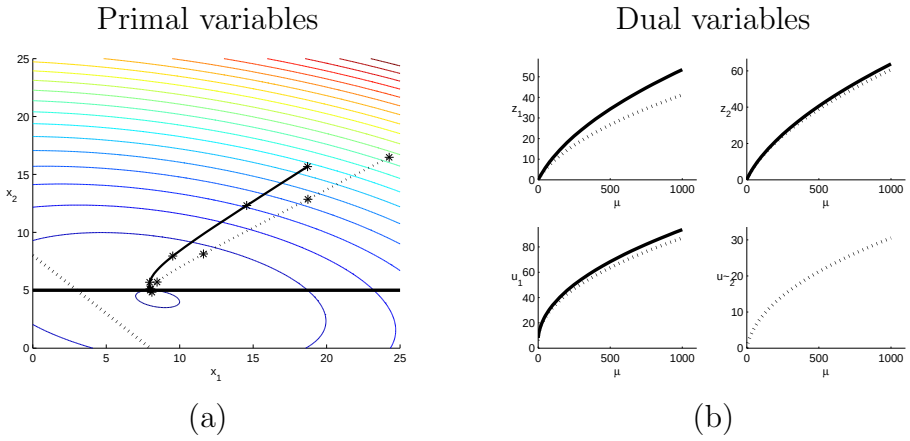

Figure 1. Change in the central path when a new constraint is introduced (dotted line)

In a general situation (when the new constraint is violated by the current point), several procedures can be implemented to recover the feasibilities. A first approach is to recover primal and dual feasibility independently. This is a reasonable approach for linear programming where primal variables do not directly contribute to the dual infeasibility. In quadratic optimisation, primal and dual feasibilities are related because of the term Hx (which depends on the primal variables) that appears in the dual feasibility constraint.

It is a well-known fact from the theory of interior-point methods [17] that the inclusion of a new constraint, even a feasible one, changes the central path. Therefore our initial goal in warmstarting is always to restore a primal, dual and central solution and to start the usual algorithm with a feasible and well-centered point.

4.4 Initial Point

- Old variables

As initial values for the old variables (x, f, z and u) we choose the first iterate from the previous solution which has primal and dual relative infeasibilities below a given threshold ϵ_c (e.g., $\epsilon_c = 0.001$) and also a small relative gap [7]. However, the complementarity products in this point are still large, $\mu > 0$. Once the usual algorithm attains such a point one or two recentering steps are carried out in order to have all complementarity products within the interval $[\gamma\mu, \mu/\gamma]$ (e.g., $\gamma = 0.1$). The stored point is known as the approximate μ-*center*.

The advantage of using a point close to optimality for warmstarting, instead of the optimal one, is that there is more room to change from the neighborhood of the former central path to that of the new one. The centering steps are a safeguard to avoid storing a point with elements that may be too close to the boundary. A start from such a point could easily get trapped.

- Barrier parameter

  No improvement to the complementarity condition is required, so we maintain the μ of the stored μ-point (old variables).

- New variables

  The only requirement in interior point methods is that the iterates have to be strictly positive. This gives a wide range of options for the initialisation of the new variables, so we try to avoid degrading the first order optimality conditions (13).

  Because new primal constraints have been added (11), the magnitude of the primal infeasibility is beyond our control.

  Given that an iterate that satisfies the modified first order optimality conditions (9) is chosen for the old variables, the dual infeasibility in the new problem is $\widetilde{B}^T\widetilde{u}$. If small positive values are chosen for the new dual variables, $\widetilde{u}$, then dual feasibility is not significantly violated. In this case, we face a trade-off: either centrality is lost or primal infeasibility is artificially increased. This approach flies in the face of the common-sense assumption that some of the new constraints will become active and the corresponding Lagrange multipliers will have a large value.

  As our objective is to reach the neighbourhood of the new central path, we assign a value to the new variables which maintains the complementarity products, $\widetilde{f}_j\widetilde{u}_j$, equal to μ. Our proposal is:

$$\widetilde{f}_j = \begin{cases} \widetilde{d}_j - \widetilde{B}_{j.}x & \text{if } \widetilde{d}_j - \widetilde{B}_{j.}x > \epsilon \\ \max(\mu/\text{mean}(\widehat{u}), \epsilon) & \text{otherwise} \end{cases} \qquad (14)$$
$$\widetilde{u}_j = \mu/\widetilde{f}_j$$

  where $\widetilde{B}_{j.}$ is row j of matrix $\widetilde{B}$.

  If the constraint is sufficiently feasible at the stored μ-point, we maintain this information. Otherwise, we take the value for the dual variable to be

the arithmetic mean of a subset $\widehat{u}$ of the old dual variables u. It has been observed that some components of u have large values compared with the others. In $\widehat{u}$ we exclude all the components such that: $u_j > \text{mean}(u) + 2 \times \text{stdev}(u)$, where $\text{stdev}(u)$ is the square root of the variance.

Parameter ϵ (e.g., $\epsilon = 100$) is a safeguard against small values of μ. This precaution is linked to the way the search direction is computed. Given that B is of different size at each iteration of the GP heuristic and $B^T B$ is block diagonal, we factorise the normal equations form of the Newton system. This form gives a system of the same size at each iteration of the heuristic and still has a good sparsity pattern. The *normal equations form* is the system that results after pivoting on the slack variables Δz, Δf and $\Delta \widetilde{f}$ and on the dual variables Δu and $\Delta \widetilde{u}$ in the Newton system for (13). The normal equations matrix takes the following form

$$\left[H + X^{-1}Z + B^T F^{-1} U B + \widetilde{B}^T \widetilde{F}^{-1} \widetilde{U} \widetilde{B}\right]. \tag{15}$$

The choice of a small $\widetilde{f}_j$ value may worsen the conditioning of the matrix (15). Consequently, it is advisable to keep the ratio $\widetilde{u}_j / \widetilde{f}_j$ of moderate size. In view of the way we choose $\widetilde{f}_j$ values, the diagonal components $\widetilde{u}_j / \widetilde{f}_j$ are safely bounded: $\widetilde{u}_j / \widetilde{f}_j = \mu / \widetilde{f}_j^2 \leq \mu / \epsilon^2$. This is a satisfactory upper bound for an $\epsilon > 1$.

4.5 Recovering step

Primal and dual feasibility may be violated at the proposed initial point, but the complementarity products are uniform. From this point, the usual interior point algorithm would try to reduce the infeasibilities and would also attempt to approach optimality by reducing the barrier parameter μ. However, as this initial point was built artificially, we propose to retain the parameter μ for a few iterations and concentrate on reducing primal and dual infeasibilities. The system that should be solved is

$$\begin{bmatrix} -H & & I & -B^T & -\widetilde{B}^T \\ B & I & & & \\ \widetilde{B} & & I & & \\ Z & & & X & \\ & U & & & F \\ & & \widetilde{U} & & & \widetilde{F} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta f \\ \Delta \widetilde{f} \\ \Delta z \\ \Delta u \\ \Delta \widetilde{u} \end{bmatrix} = \begin{bmatrix} h + B^T u + \widetilde{B}^T \widetilde{u} + Hx - z \\ d - Bx - f \\ \widetilde{d} - \widetilde{B}x - \widetilde{f} \\ \mu e - Xz \\ \mu e - Fu \\ \mu e - \widetilde{F}\widetilde{u} \end{bmatrix}. \tag{16}$$

This system is equivalent to the one that would be solved by the usual infeasible interior point method applied to (13) except that in this case no

improvement to complementarity condition is required.

4.6 Weighted Newton Step

It is unrealistic to assume that a full step in the direction obtained from (16) will be performed, for the reason that variables must remain positive. It may occur that only a very small step is allowed, because of poor scaling of either the data of the problem or the variable values, and then the amount of infeasibility absorbed is very small. When this occurs, our proposal is to perturb the direction and apply multiple centrality correctors [6] in order to provide a better chance for improvement in primal and/or dual feasibilities.

Rather than solving (16) in one go, we split the right-hand side into three parts:

$$\begin{bmatrix} \tau_D \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ \tau_P \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ \tau_\mu \end{bmatrix}, \tag{17}$$

where

$$\tau_D = \begin{bmatrix} h + B^T u + \widetilde{B}^T \widetilde{u} + Hx - z \end{bmatrix}, \ \tau_P = \begin{bmatrix} d - Bx - f \\ \widetilde{d} - \widetilde{B}x - \widetilde{f} \end{bmatrix}, \ \tau_\mu = \begin{bmatrix} \mu e - Xz \\ \mu e - Fu \\ \mu e - \widetilde{F}\widetilde{u} \end{bmatrix} \tag{18}$$

and solve the system for each part (using the same factorisation). We obtain the directions Δ_D, Δ_P and Δ_μ, respectively. Note that Δ_P only attempts to recover primal feasibility: it leaves the dual infeasibility and the complementarity unchanged. Analogously, Δ_D attempts to recover dual feasibility only, while Δ_μ does not alter the infeasibilities at all and concentrates on driving the complementarity products close to the barrier parameter μ.

An *estimate* of the amount of infeasibility to be absorbed can be computed from these directions. If any of the estimates are not satisfactory we propose to scale the direction and apply multiple-centrality correctors. Scaling a direction with a factor β ($\beta < 1$) is equivalent to reducing the infeasibilities in the right hand side of (16) by this factor. The advantage of solving the system in three steps is that β can be chosen after solving the system.

4.6.1 Estimates of the infeasibility to be absorbed

In linear programming, the reduction rate of the infeasibilities is 1 minus the step size, which may be different for the primal and the dual direction. Namely, if step α_P is performed on the primal direction, then primal infeasibility is reduced $(1 - \alpha_P)$ times. Analogously, if the step α_D is carried out, then dual infeasibility is reduced $(1 - \alpha_D)$ times.

In quadratic programming, the same results apply if primal and dual step lengths are the same. However, the usual practice is to choose different step sizes for the primal and dual directions, α_P and α_D. While primal infeasibility is monotonically decreasing, the dual one behaves erratically. Given that the initial point was feasible (or nearly feasible) in the previous problem

$$\tau_D^{old} = h + B^T u + Hx - z \approx 0,$$

the warm-starting point in the new problem will satisfy

$$\begin{aligned}\tau_D^{new} &= h + B^T u + \widetilde{B}^T \widetilde{u} + Hx - z \\ &= (h + B^T u + Hx - z) + \widetilde{B}^T \widetilde{u} \\ &= \tau_D^{old} + \widetilde{B}^T \widetilde{u} \approx \widetilde{B}^T \widetilde{u}\,.\end{aligned}$$

After the step in Newton direction the dual infeasibility will be the following:

$$\begin{aligned}\widehat{\tau}_D^{new} &= h + B^T(u + \alpha_D \Delta u) + \widetilde{B}^T(\widetilde{u} + \alpha_D \Delta \widetilde{u}) + H(x + \alpha_P \Delta x) - (z + \alpha_D \Delta z) \\ &= h + B^T u + \widetilde{B}^T \widetilde{u} + Hx - z + \alpha_D B^T \Delta u + \alpha_D \widetilde{B}^T \Delta \widetilde{u} + \alpha_P H \Delta x - \alpha_D \Delta z \\ &\approx \widetilde{B}^T \widetilde{u} + \alpha_D (B^T \Delta u + \widetilde{B}^T \Delta \widetilde{u} - \Delta z) + \alpha_P H \Delta x\,.\end{aligned}$$

Extending the linear case to the quadratic case, but only to compute the estimates of the infeasibility to be absorbed, we propose:

- As estimates of the primal infeasibility to be absorbed, α_P^P, which is the maximum step length with the primal variables along Δ_P.
- As estimates of the dual infeasibility to be absorbed, α_D^D, which is the maximum step length with the dual variables along Δ_D.

Direction Δ_μ could be omitted if we only intended to recover feasibility. Moreover, if the initial point is well centred τ_μ will be nearly 0 in all its components, at least in the first iteration.

4.6.2 Final direction

When an estimate, either α_P^P or α_D^D, is very small, this is a sign that certain components of this direction are not satisfactory. We hope that by using a weighted direction plus a few multiple centrality correctors, the total reduction in primal and dual infeasibilities will be larger.

Given that the final direction will be a composition of the three directions, we try a more ambitious target, weighting each direction according to:

$$\beta_P = \min(\kappa\alpha_P^P, 1)\ \beta_D = \min(\kappa\alpha_D^D, 1)\ \beta_\mu = 1 \text{ or } 0 \tag{19}$$

with $\kappa > 1$. On the basis of previous computational experience and [8] we propose to use $\kappa = 2$, but there may be other cases in which a larger κ is more suitable. The final composite direction Δ is:

$$\Delta = \beta_P\Delta_P + \beta_D\Delta_D + \beta_\mu\Delta_\mu$$

to which some centrality correctors are applied.

4.7 Outline of a warm-start iteration

The warm start in interior point methods, as opposed to the cold start, produces an initial point using some prior knowledge of the problem (or a similar one), with the idea of making the usual algorithm perform better. Therefore, when considering a warm start for the following problem, we must store a near-optimal and well-centered μ-point.

From the optimal solution, new cuts are appended to the problem. The new variables, both primal and dual, are initialized as in (14). From this point the following warm-start procedure is carried out and produces the initial point.

Warm-start procedure

i Compute the primal and dual infeasibilities, τ_P and τ_D, and the centrality deviations, τ_μ.

ii Compute the directions needed to recover primal and dual feasibility and centrality, solving system (16) with right-hand sides (17)-(18): Δ_P, Δ_D, Δ_μ.

iii Compute the usual Newton direction: $\Delta = \Delta_p + \Delta_d + \Delta_\mu$ and the maximum primal and dual step lengths α^P and α^D. If $\alpha^P > \epsilon_s$ and $\alpha^D > \epsilon_s$ then go to (vii).

iv Compute the maximum step length and the estimates of the infeasibility to be absorbed:

$$\Delta_P \rightarrow \alpha_{\mathbf{P}}^{\mathbf{P}}, \alpha_P^D$$

$$\Delta_D \rightarrow \alpha_D^P, \alpha_{\mathbf{D}}^{\mathbf{D}}$$

v Compute the direction weights:

$$\beta_P = \min(\kappa\alpha_P^P, 1)$$

$$\beta_D = \min(\kappa\alpha_D^D, 1)$$

$$\beta_\mu = 1$$

vi Compute the predictor composite direction:

$$\Delta = \beta_P\Delta_P + \beta_D\Delta_D + \beta_\mu\Delta_\mu$$

vii Apply multiple centrality correctors to Δ.

viii Compute the step lengths on Δ and update the point.

ix Compute primal and dual infeasibilities, τ_P and τ_D.

x If $\|\tau_P\| > \epsilon_w$ or $\|\tau_D\| > \epsilon_w$ and there has been a significant reduction of the infeasibilities, repeat from (i)

Otherwise switch to the usual infeasible interior point method.

In this warm-start procedure (step `x`), we consider the reduction of the infeasibility to be significant if the infinity norm of the primal and dual infeasibilities at the current iteration has been reduced by at least 40% compared with that in the previous iteration.

5 Computational Results

The warm-start approach presented was tested for the solution of the long-term generation planning using the GP heuristic for the LMC oracle. These problems are realistic test cases from the Spanish liberalised power pool. Some of the units represent a single generation unit while some others merge a group of similar generation units. The data of the test cases is available from [13].

5.1 Implementation issues

The code used was implemented in C programming language. The warm-start procedure is an extension of the infeasible primal-dual interior point method [17]. The interior point algorithm applies the Mehrotra's predictor-corrector direction [9] with Gondzio's multiple-centrality correctors [6]. The usual algorithm was slightly modified in order to store the μ-point: when the algorithm attains a point which is a good candidate for the warm-start procedure, this point is recentered and stored for later use in the warm-start function if a new problem has to be solved.

The warm-start procedure allows the user to choose some parameters. The values that gave the best results when we were solving the instances were:

- $\epsilon_c = 0.001$: if relative primal and dual infeasibilities and the relative duality gap are smaller than ϵ_c then perform some centering steps and store the μ-point.
- $\kappa = 2$: scaling directions parameter used in (19).
- $\beta_\mu = 1$: the centering direction usually helps.
- $\epsilon_s = 0.1$: if the Newton step does not make enough progress ($\alpha^P < \epsilon_s$ or $\alpha^D < \epsilon_s$) then compute the weighted Newton step.
- $\epsilon_w = 10^{-4}$: if relative primal and dual infeasibilities are less than ϵ_w the warm start procedure stops.

5.1.1 Problem description

Table 1 shows the basic statistics of each problem: number of units, n_u, number of intervals, n_i, and number of non-LMCs, $n_{\geq}$. Thus, the number of variables is $n_i \times n_u$. The total number of LMCs that the complete model has is $n_i \times (2^{n_u} - 1)$. Note the reduced subset of LMCs employed by the heuristic shown in the column *consid.* n_{lmc}, of which *active* n_{lmc} are active at the solution point. The number of LMCs generated is highly related to the number of active upper bounds (1) because the self-ordering stage of the GP heuristic adds a new LMC for any combination of the variables at their upper bound (see columns *active up. bound* and *consid.* n_{lmc}).

5.1.2 GP heuristic statistics

The number of problems re-solved during the solution of each case with the GP heuristic is reported in Table 2. The first problem generated by the heuristic is solved from a cold start and the following ones are fed with an initial warm-start point, derived from the previous solution.

			non-LMCs	LMCs (all intervals)		
				active	*active*	*consid.*
case	n_u	n_i	$n_{\geq}$	n_{lmc}	*up. bound*	n_{lmc}
ltp_01	13	11	9	24	59	171
ltp_02	15	11	43	48	22	117
ltp_03	17	11	66	46	23	137
ltp_04	18	11	77	51	24	147
ltp_05	45	11	40	34	108	2293
ltp_06	63	11	222	176	82	2484
ltp_07	18	52	321	130	69	703
ltp_08	25	27	190	85	76	411
ltp_09	52	15	90	27	32	622
ltp_10	29	8	61	39	9	213
ltp_11	33	13	34	23	12	317
ltp_12	67	15	329	243	74	794
ltp_13	56	52	32	88	218	2386
ltp_14	13	27	25	39	127	138
ltp_15	14	27	24	40	40	169
ltp_16	12	27	25	33	15	182
ltp_17	16	27	24	40	37	241
ltp_18	18	27	17	77	42	318
ltp_19	19	27	25	80	112	354

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

In each iteration of the GP heuristic a number of LMCs are added to the problem. We reoptimise when one or more of the LMCs generated are infeasible, but the new problem contains all previously generated constraints. In Table 2, the *new LMCs* column shows the average number of new LMCs appended to each problem being reoptimized. The adjacent columns show the average number of infeasible LMCs at the optimal point, $inf*$, and at the stored μ-point, inf^{μ}. The average is calculated with respect to the number of problems solved using a warm start, shown in the column *num prb re-solved*.

Table 3 details the relative infeasibility (of the newly added LMCs), which is only calculated for information purposes, at each iteration of the GP heuristic

	num prb	LMCs		
	re-solved	new	inf *	inf $^{\mu}$
ltp_01	4	40.5	26.5	3.3
ltp_02	7	15.0	6.3	2.0
ltp_03	9	13.9	4.8	3.1
ltp_04	10	13.5	4.7	3.1
ltp_05	1	1983.0	1983.0	1697.0
ltp_06	1	1983.0	1983.0	1099.0
ltp_07	6	108.5	12.5	2.0
ltp_08	6	62.5	4.7	0.0
ltp_09	5	106.0	1.2	0.0
ltp_10	7	28.3	2.7	0.1
ltp_11	3	90.7	1.0	0.0
ltp_12	0	0.0	0.0	0.0
ltp_13	2	1127.5	1.0	0.0
ltp_14	2	48.0	3.0	2.5
ltp_15	4	31.5	1.5	1.0
ltp_16	1	91.0	1.0	1.0
ltp_17	9	21.6	1.2	0.4
ltp_18	12	23.9	4.0	3.5
ltp_19	9	31.7	3.2	0.1

Table 2
Number of problems re-solved starting from a warm-start point, and average number of added LMCs and infeasible LMCs at the optimizer and at the stored μ-point.

where at least one of the new LMCs is infeasible and the new problem must be re-solved (see the columns heading for the explicit definition of relative infeasibility). The 0 value indicates that the norm of the infeasibility is smaller than 10^{-6}. There is a significant difference between iteration 2, which is the self-ordering stage, and the following ones corresponding to the iterative stage (see Section 3.1).

During the self-ordering stage (iteration 2) of the GP heuristic a number of LMCs are added so that the generation of the units that, at the initialization stage, were generating at their maximum capacity conform to the shape of the LDC. The subset of the LMCs added that become active generally means the deepest cut in the application of the GP heuristic. The following LMCs

Table 3
2-norm of the relative infeasibility of the new constraints at each iteration where the problem is re-solved.

	$\|\max(0,(\widetilde{d}_j - \widetilde{B}_{j\cdot}x^*)/\widetilde{d}_j)\|_2$										
	ite 2	ite 3	ite 4	ite 5	ite 6	ite 7	ite 8	ite 9	ite 10	ite 11	ite 12
ltp_01	4.1e-03	1.5e-05	4.9e-05	6.8e-05							
ltp_02	1.4e-03	4.1e-05	6.5e-05	5.5e-05	4.7e-05	2.9e-05	6.3e-05				
ltp_03	1.4e-03	6.0e-06	5.0e-06	3.1e-05	5.4e-05	4.5e-05	3.8e-05	2.3e-05	5.4e-05		
ltp_04	1.4e-03	6.0e-06	6.0e-06	3.1e-05	5.4e-05	6.2e-05	4.1e-05	3.4e-05	2.1e-05	5.1e-05	
ltp_05	1.7e-02										
ltp_06	1.7e-02										
ltp_07	2.0e-06	8.0e-06	8.0e-06	5.4e-05	6.3e-05	1.2e-04					
ltp_08	8.0e-03	0	0	0	0	0					
ltp_09	1.0e-06	1.0e-06	0	0	0						
ltp_10	1.2e-02	3.8e-03	3.0e-06	1.0e-06	0	0	0				
ltp_11	3.0e-06	0	0								
ltp_12											
ltp_13	0	0									
ltp_14	2.9e-05	2.3e-04									
ltp_15	3.5e-05	1.6e-04	3.9e-05	3.4e-05							
ltp_16	1.3e-02										
ltp_17	7.4e-05	1.0e-05	3.5e-05	3.4e-05	4.2e-05	3.4e-05	3.6e-05	1.8e-05	2.9e-05		
ltp_18	9.9e-03	1.2e-02	4.6e-05	6.0e-06	4.0e-06	7.2e-05	1.4e-04	3.9e-05	2.8e-05	1.3e-05	4.0e-06
ltp_19	0	0	0	0	0	0	0	0	0		

added in the iterative stage of the GP heuristic append only one more unit at a time, the one with the highest ratio of generated energy with respect to maximum capacity. The LMC then reduces the energies of the former and the

last added unit; the cut is generally more shallow as more units are added in the iterative stage.

5.2 Warm-start results

The results in Table 4 show time (in seconds) and the number of iterations obtained by solving the long-term power planning instances with the warm start technique. The way the interior-point solver begins is compared here: using a cold start or a warm start. The first problem solved (of the GP heuristic described in Section 4) is always initialized with a cold start (all the variable values are 10^6).

The column headed *prb ws* shows the number of problems where the warm start procedure was applied, and therefore these are the compared solutions. The adjacent column shows the results obtained using the cold start solution: time and average number of iterations executed by the interior-point solver. The warm start results are detailed next: solution time, the average number of warm start iterations (*ws*) as displayed in the warm start procedure in Section 4.7, the average number of iterations required to find the new solution using the usual interior-point method (*ipm*), and the sum of both columns (*total*), giving the total number of iterations. The total number of iterations should be compared to the number of iterations needed by a cold start solution.

The last part of Table 4 shows the variations in computation time and in the number of iterations between the solutions obtained using a cold or warm start. The bottom row shows the average for the test cases computed. On average, 15 iterations are required to solve each subproblem if we start from an arbitrary point (cold start) and 7 iterations if we use information from the previous solution (warm start). On average, the warm-start routine performs 1 iteration (and 6 iterations of the usual interior-point algorithm). The average saving on interior point iterations is around 50%.

The difference in time in most of the instances is a matter of fractions of a second (see Table 4). Considering that the timing function is inexact (the time given is computed with the Linux *time* command, adding up user and system time), the difference in time is almost imperceptible. We further analyzed how much of the time is employed by the routines that solve the problems (including the warm start routine when applicable) using the `-pg` option of the gcc compiler and the `gprof` profiler [1]. Figure 2 shows the percentage of time employed by the interior-point solver function plus the warm start routine (where applicable). In 17 of the 19 cases, the time employed to solve the problems represents less than 30% of the total time (for the cold start

[1] We wish to thank Andreas Grothey for his guidance on this issue

	prb	Cold start		Warm start				Difference	
	ws	time	ite	time	ws	ipm	total	time	ite
ltp_01	4	1.00	15.5	0.80	1.5	8.3	9.8	0.20	5.75
ltp_02	7	0.70	15.1	0.62	1.9	5.7	7.6	0.08	7.57
ltp_03	9	0.74	14.0	0.71	1.1	7.1	8.2	0.03	5.78
ltp_04	10	0.81	14.3	0.77	1.1	6.7	7.8	0.04	6.50
ltp_05	1	2.66	18.0	2.59	1.0	11.0	12.0	0.07	6.00
ltp_06	1	4.03	21.0	3.94	1.0	12.0	13.0	0.09	8.00
ltp_07	6	4.17	13.8	2.72	1.7	5.3	7.0	1.45	6.83
ltp_08	6	2.14	16.2	1.38	0.5	5.0	5.5	0.76	10.67
ltp_09	6	2.91	17.8	2.20	0.8	4.8	5.6	0.71	12.23
ltp_10	7	0.77	15.1	0.68	1.1	4.0	5.1	0.08	10.00
ltp_11	6	1.19	15.7	1.12	0.0	4.7	4.7	0.07	11.00
ltp_12	0	3.49	-	3.47	-	-	-	-	-
ltp_13	1	6.74	14.0	6.54	0.0	4.0	4.0	0.20	10.00
ltp_14	2	0.48	15.5	0.50	0.5	6.5	7.0	-0.02	8.50
ltp_15	4	0.57	14.5	0.70	1.5	4.0	5.5	-0.14	9.00
ltp_16	1	0.50	12.0	0.48	2.0	4.0	6.0	0.01	6.00
ltp_17	9	0.75	14.1	0.63	0.8	4.8	5.6	0.12	8.55
ltp_18	12	1.08	13.9	0.99	1.3	7.4	8.7	0.08	5.25
ltp_19	9	1.76	16.1	1.20	0.7	5.1	5.8	0.55	10.33
avg		*1.92*	*15.4*	*1.69*	*1.0*	*6.1*	*7.2*	*0.24*	*8.22*

Table 4
Time and average number of iterations done with a cold start or with a warm start.

solution), being the average at 17.14% for the cold start and 11.17% for the warm start solution. From the rest of the time, most of it is spent on the computation of the LMC right-hand sides. Consequently, the scope for time savings resulting from the use of warm start technique is very limited.

To complete the analysis, Table 5 shows the percentage of time used by the interior-point solver (t_{IP}) within the solution of the GP heuristic (t_{GP}), the percentage of time used by the warm-start routine (t_{WS}) within the interior-point algorithm and the percentage of time used by the warm-start routine within the solution of the case. The conclusion drawn here is that on average:

	% t_{IP}/t_{GP}	% t_{WS}/t_{IP}	% t_{WS}/t_{GP}
ltp_01	4.9	13.3	0.65
ltp_02	6.5	25.0	1.62
ltp_03	12.8	11.8	1.50
ltp_04	14.8	13.3	1.97
ltp_05	3.5	3.1	0.11
ltp_06	9.7	4.2	0.41
ltp_07	53.5	18.3	9.82
ltp_08	29.5	6.3	1.87
ltp_09	9.4	11.4	1.07
ltp_10	9.3	20.0	1.85
ltp_11	3.6	0	0
ltp_12	8.4		
ltp_13	4.1	0	0
ltp_14	2.8	0	0
ltp_15	3.6	14.3	0.51
ltp_16	1.8	0.0	0.00
ltp_17	7.2	9.5	0.68
ltp_18	11.7	16.3	1.91
ltp_19	15.7	9.8	1.53
avg	*11.19*	*9.81*	*1.24*

Table 5
Statistics on the percentage of time used for the warm-start solution.

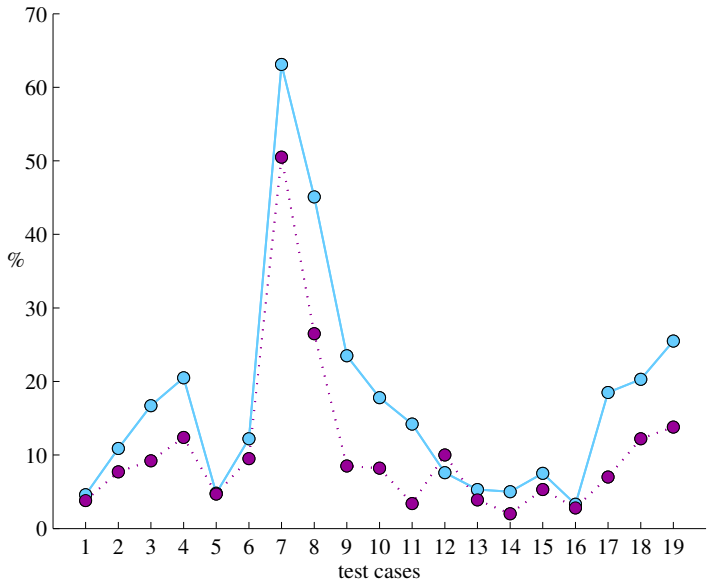

Figure 2. Percentage of time used by the interior-point solver in the GP heuristic (cold start represented by solid line and warm start by dotted line)

- about 90% of the time is devoted to computing LMC right-hand sides,
- the warm-start routine stands for about 10% of time used by the interior-point solution, and
- the warm-start routine, in a global vision, uses 1% of the GP heuristic solution time.

5.2.1 Importance of the μ-point

The analysis of the results presented in Table 4 reveals that warmstarting in interior point methods works well. Table 6 shows the average number of iterations saved by carrying out the warm-start procedure instead of simply making a straightforward start from the stored μ-point. It is worth noting that in some cases, rather than carrying out a previous step, it is better to quite simply feed the usual algorithm with the stored μ-point as the initial point. These results support the idea that making sure that the stored approximate μ-center has good centrality properties pays off.

The small advantage presented by the warm-start procedure in the test cases solved may be a consequence of the shallowness of the appended cuts (LMCs). When problems arise with deeper cuts, the warm-start technique should present an advantage.

Another conclusion that can be drawn from the analysis of the results is that the number of warm-start iterations not only depends on the magnitude of the primal and dual infeasibility, but also on the quality of the stored μ-point:

	Cold start		Warm start		μ-point	
	time	ite	time	ite	time	ite
ltp_01	1.00	15.5	0.80	9.8	0.96	9.0
ltp_02	0.70	15.1	0.62	7.6	0.66	8.9
ltp_03	0.74	14.0	0.71	8.2	0.66	7.9
ltp_04	0.81	14.3	0.77	7.8	0.76	8.3
ltp_05	2.66	18.0	2.59	12.0	2.66	11.0
ltp_06	4.03	21.0	3.94	13.0	3.78	12.0
ltp_07	4.17	13.8	2.72	7.0	3.81	12.3
ltp_08	2.14	16.2	1.38	5.5	2.10	10.4
ltp_09	2.91	17.8	2.20	5.6	2.16	6.0
ltp_10	0.77	15.1	0.68	5.1	0.57	6.5
ltp_11	1.19	15.7	1.12	4.7	0.94	4.7
ltp_12	3.49	-	3.47	-	3.44	-
ltp_13	6.74	14.0	6.54	4.0	6.55	4.0
ltp_14	0.48	15.5	0.50	7.0	0.49	6.5
ltp_15	0.57	14.5	0.70	5.5	0.66	6.3
ltp_16	0.50	12.0	0.48	6.0	0.56	6.0
ltp_17	0.75	14.1	0.63	5.6	0.61	5.3
ltp_18	1.08	13.9	0.99	8.7	0.90	10.8
ltp_19	1.76	16.1	1.20	5.8	1.55	13.6
avg	*1.92*	*15.4*	*1.69*	*7.2*	*1.78*	*8.3*

Table 6
Comparison of several initializing procedures.

the larger μ is, the easier it is to warm start interior point method.

6 Conclusions

- The solution through the GP heuristic in [15] of the long-term generation planning modeled with the Bloom and Gallant formulation requires solving a succession of QP problems with an increasing number of inequality constraints, where warm starting can be applied.
- The warm start procedure of Gondzio and Grothey [8] has been extended to the quadratic case. The idea is to absorb the primal and dual infeasibility in

several steps maintaining the complementarity products close to μ. When the steps along the usual Newton direction are small, the Newton system is solved in three steps, giving a direction to recover primal feasibility, another to recover dual feasibility and the third one is to recover the centrality. We try to reduce the amount of infeasibility using a weighted Newton direction. The weights are based on estimates of the primal and dual infeasibility reduction.

- We have also proposed a way to initialise the new variables. It maintains the same μ for the new point.
- The weighted Newton warm start produces savings with respect to the cold start both in the number of iterations (53% on average) and in CPU time (12% on average).
- Initialising the new variables in the way put forward and then continuing with normal interior-point iterations is also advantageous with respect to cold start, because on average the savings in iterations are of 46% and the savings in CPU time are of 7%.
- As regards CPU time, it must be borne in mind that on average 90% of it is devoted to the calculation of the right-hand sides of the newly created constraints. In the general context of interior-point methods for QP the CPU time savings are proportional to the number of iterations saved. However, in this specific application the vast majority of time is spent on evaluating the right-hand sides of the LMCs. Therefore, leaving out the calculations of the right-hand sides, the savings resulting from the use of the warmstarting procedure would be more significant.

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] H. Y. Benson and D. F. Shanno. An exact primal-dual penalty method approach to warmstarting interior-point methods for linear programming. *Computational Optimization and Applications*, 38(3):371–399, 2007.

[3] H.Y. Benson and D.F. Shanno. Interior-point methods for nonconvex nonlinear programming: regularization and warmstarts. *Computational Optimization and Applications*, 40(2):143–189, 2008.

[4] 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.

[5] A. Forsgren. On warm starts for interior methods. In Springer Boston, editor,

System Modeling and Optimization, volume 199/2006 of *IFIP International Federation for Information Processing*, pages 51–66. Springer, 2006.

[6] J. Gondzio. Multiple centrality corrections in a primal-dual method for linear programming. *Computational Optimization and Applications*, 6(2):137–156, 1996.

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

[8] J. Gondzio and A. Grothey. Re-optimization with the primal-dual interior point method. *SIAM Journal on Optimization*, 13(3):842–864, 2003.

[9] S. Mehrotra. On the implementation of a primal-dual interior point method. *SIAM Journal on Optimization*, 2(4):575–601, November 1992.

[10] N. Nabona. On the loading order in the matching of an electric load-duration curve by generation units. Technical Report DR2006/03, Dept. Estadística i Inv. Operativa, Univ. Politèc. de Catalunya, Barcelona, C. Jordi Girona, 1-3 08034 Barcelona, 2006. Available at http://www-eio.upc.es/research/gnom/.

[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, Barcelona, 2002.

[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. Cerdà, E. Escudero, and R. Sala, editors, *Optimización bajo Incertidumbre*, chapter 8, pages 185–211. Tirant lo Blanch, Valencia 46010, Spain, 2004. ISBN 84-8456-100-3. Available at http://www.tirant.es.

[13] GNOM Group on Numerical Optimization and Modeling. http://www-eio.upc.es/research/gnom.

[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, Barcelona, 2002.

[15] A. Pagès and N. Nabona. A heuristic for the long-term electricity generation planning problem using the Bloom and Gallant formulation. *European Journal of Operational Research*, 181(3):1245–1264, 2005.

[16] 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.

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

[18] E. A. Yildirim and S. J. Wright. Warm-start strategies in interior-point methods for linear programming. *SIAM Journal on Optimization*, 12(3):782–810, 2002.